Methods for Incorporating Biological Information into the Statistical Analysis of Gene Expression Microarray Data

Debbie Leader

A thesis submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Statistics.

The University of Auckland New Zealand.

December 2009

Abstract

Microarray technology has made it possible for researchers to simultaneously measure the expression levels of tens of thousands of genes. It is believed that most human diseases and biological phenomena occur through the interaction of groups of genes that are functionally related. To investigate the feasibility of incorporating functional information and/or constraints (based on biological and technical needs) into the classification process two approaches were examined in this thesis.

The first of these approaches investigated the effect of incorporating a pre-filter into the gene selection step of the classifier construction process. Both simulated and real microarray datasets were used to assess the utility of this approach. The pre-filter was based on an early method for determining if a gene had undergone a biologically relevant level of differential expression between two classes. The genes retained by the pre-filter were ranked using one of five standard statistical ranking methods and the most highly ranked were used to construct a predictive classifier. To generate the simulated data a selection of different parametric and non-parametric techniques were employed. The results from these analyses showed that when the constraints that the pre-filter contains were placed on the classification analysis, the predictive performance of the classifiers were similar to when the pre-filter was not used.

The second approach explored the feasibility of incorporating sets of functionally related genes into the classification process. Three publicly available datasets obtained from studies into breast cancer were used to assess the utility of this approach. A summary of each gene-set was derived by reducing the dimensionality of each gene-set via the use of Principal Co-ordinates Analysis. The reduced gene-sets were then ranked based on their ability to distinguish between the two classes (via Hotelling's T^2) and those most highly ranked were used to construct a classifier via logistic regression. The results from the analyses undertaken for this approach showed that it was possible to incorporate function information into the classification process whilst maintaining an equivalent (if not higher) level of predictive performance, as well as improving the biological interpretability of the classifier.

Acknowledgements

I could not have embarked on the journey that this thesis has taken me on without the Bright Futures Enterprise Scholarship I was awarded from the TEC in collaboration with Pacific Edge Biotechnology Ltd. Of equal importance to the completion of this thesis was the ongoing guidance, support and inspiration provided by my supervisor Dr. Micheal A. Black, even after he moved to Dunedin.

I would also like to thank my parents, Donna and Andrew Leader for their continued support, encouragement and for taking care of me when I felt that the world was coming to an end.

I wish to thank the Department of Biochemistry at the University of Otago for hosting me during my stay in Dunedin, in particular I wish to thank Moira Robinson, Dr. Fabienne Lecomte, Payal Diwadkar, Dr. Nattanan (Joy) Panjaworayan and Dr. Amonida Zadissa who made my time in Dunedin both informative and enjoyable.

I would also like to thank my fellow PhD sufferer and great friend Dr. Sarah Song for her lively conversions and continued support.

Finally I wish to thank Associate Professor Brian H. McArdle for taking on the official duty of being my main supervisor when Mik joined the Department of Biochemistry at the University of Otago half way through my PhD and the Department of Statistics at the University of Auckland for supporting me in my decision to move to Dunedin so as to remain under the supervision of Dr. Micheal A. Black.

Contents

List of Figures

List of Tables

1

Introduction

A little over a hundred years after Freidrich Miescher (1871) identified deoxyribonucleic acid (DNA), the concept of the "Human Genome Initiative" was formed (Charles DeLisi and David A. Smith, HGP, 1985). Two years later, in 1987, with the support of several million US dollars, the advisory committee of the U.S. Department of Energy put forth a 15-year multidisciplinary plan to map and sequence the entire human genome (Lim, 2002). It was at this point that Dr. Hwa A. Lim, of the Supercomputer Computations Research Institute, coined the neologism "Bio-informatique" to give a name to the emerging field that encompassed aspects of a number of existing areas, such as Biological Sciences (genetics/genomics), Computer Science, Mathematics and Statistics. Over the following year the term morphed into the neologism that is used today, namely "Bioinformatics" (Lim, 2002).

The field of Bioinformatics can be defined as the organisation, analysis and storage of genomic data, information and results. One aspect that defines this field is the use of computational and statistical methods applied to whole genome data, which may come from one or more organisms. The use of statistical methods to analyse DNA microarray data has been well documented (Alizadeh et al., 2000; Dudoit et al., 2002b; Speed, 2003; Bair and Tibshirani, 2004; John et al., 2008). However, it has only been in recent years that more advanced multivariate techniques have begun to be applied to microarray data (e.g., the use of Principal Components Analysis in gene selection (Wang and Gehan, 2005) and Discriminant Function Analysis for classification purposes

(Nguyen and Rocke, 2002; McLachlan et al., 2004; Shen et al., 2006)).

The existing literature on microarray data analysis brings to light several issues regarding the dimensionality of microarray data and the effect it has on model building and the computational costs involved. Hastie and Tibshirani (2004) provide an excellent description of how several of the more popular and computer intensive microarray analysis methods can be computationally reduced in order to speed up the analysis process. Two major issues that relate to the dimensionality of microarray data are those of gene selection, when using microarray gene expression data for classification purposes, and the ability of the produced classification models to be generalised. The issue of gene selection is of particular importance as many of the statistical classification methods that are used in the analysis of microarray data require there to be a greater number of observations than variables, which is not the case in most microarray experiments. Methods for gene selection are numerous and well documented (including the use of the Signal to Noise Ratio (Golub et al., 1999), the SAM adjusted t-test (Tusher et al., 2001) and the moderated t-statistic (Smyth, 2004)), however many of these methods share the weakness that they do not take into account the possibility of any interaction between the genes. It is only recently that methods that take this interaction into account have been introduced (e.g., Wang and Gehan, 2005). Gene selection is arguably one of the most important steps in the analysis of microarray gene expression data for classification purposes and there is a lot of room for methodology development and refinement in this area. Generalisability of the produced classification models can sometimes be achieved, either by taking a larger number of observations (i.e. more samples/arrays, which in many cases is not feasible due to ethical or monetary restraints) through the appropriate use of cross-validation, or by insuring that the models are validated on independent data.

The development of a new and successful prognostic test, for differentiating between patients who have a good long term prognosis and those who have a poor long term prognosis, that utilises multivariate techniques has enormous potential commercial benefit and would be a significant contribution to the development of individualised patient treatment plans. In order to achieve this goal the use of multivariate and univariate statistical techniques will be investigated in combination with novel approaches to the incorporation of biological pathway and co-regulation information. This includes the incorporation of biological information into the gene selection and classification processes via the use of methods based on biological knowledge and multivariate methods such as principal co-ordinates analysis (Gower, 1966) and Hotelling's T^2 statistic (Hotelling, 1933).

1.1 Molecular Biology

As interest lies in the analysis of expression levels of genes, it is important to first understand what genes are and where they come from. This begins with DNA. DNA consists of a long series of paired nucleotides, also known as base pairs, in the form of a double helix (Lewin, 1997; Berg et al., 2002). There are four nucleotides, Adenine (A), Thymine (T), Guanine (G) and Cytosine (C). As Adenine can only pair with Thymine, and Guanine can only pair with Cytosine, this yields two possible base pairings. DNA is wound tightly around histones to form chromosomes, which are found in the nucleus of an organism's cells (Lewin, 1997). Genes are sections of DNA that are responsible for the creation of proteins, which are chains of amino acids folded into specific forms. Proteins determine the structure, function and regulation of the cells of an organism (Berg et al., 2002). The process of protein production is referred to as the "Central Dogma of Molecular Biology" (Crick, 1966). There are two steps to the Central Dogma, *transcription* and *translation*, as shown in Figure 1.1.

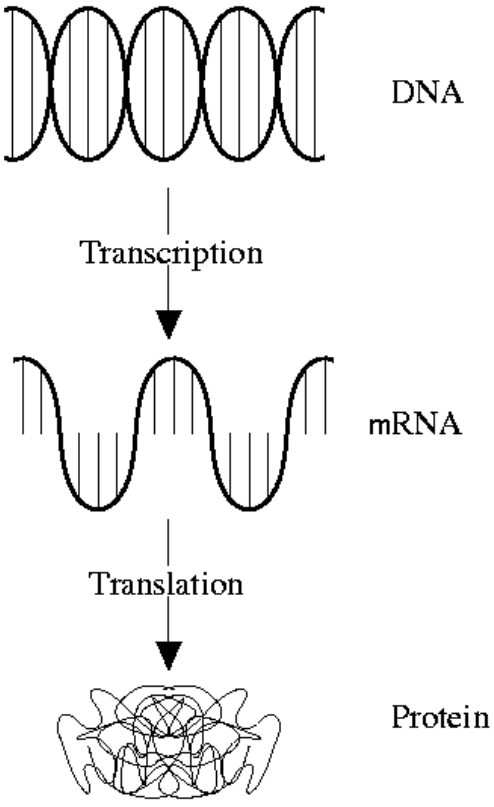

Figure 1.1: The Central Dogma of Molecular Biology. The transcription of DNA into mRNA followed by the translation of the mRNA into protein.

Transcription is the process of making a copy of a section of DNA (a gene) and converting it into messenger RNA (mRNA) (Lewin, 1997). Ribonucleic Acid (RNA) is a

single stranded version of DNA. It has the same bases as DNA except that Thymine (T) is replaced by Uracil (U) (Berg et al., 2002). Once the mRNA is complete it travels out of the nucleus but remains within the ribosome of the cell where translation can now occur (Brenner et al., 1961). Translation is the process of converting mRNA into proteins (Lewin, 1997). In order to describe this process one must start with the mRNA. The bases on the mRNA are grouped in triplets. Each triplet or codon represents a building block (amino acid) of a protein (Lewin, 1997). At the ribosome, amino acids are attached to each of the codons. There are twenty different amino acids, so each amino acid can be represented by a number of different codons (Lewin, 1997). At its most basic, translation is the construction of a chain of amino acids, according to the ordering of the codons in the mRNA. Once the chain has been formed and any post-translational modifications have been made, it is folded into a specific structure, which helps to determine the end function of the protein (Berg et al., 2002). When the structure is complete the protein is ready for use. A gene is said to be *expressing* if it is producing proteins (Berg et al., 2002). Thus to get an indication of the level of expression, it is common practice to measure the level of mRNA being transcribed (Shalon et al., 1996).

1.2 Genomic Technologies

There are several types of bioinformatic data, of which protein based data and nucleic acid based data are two common types.

One of the most popular forms of nucleic acid based data is that generated by DNA microarray experiments (Schema et al., 1995; Shalon et al., 1996).There are several types of microarray data, including spotted arrays (Schema et al., 1995; Shalon et al., 1996), photolithographic arrays (Fodor et al., 1991), arrays produced by ink-jet printing methods (Blanchard et al., 1996), and bead-based systems (Yang et al., 2001). Microarray technology has been used in many different areas, including the study of cancers (Alizadeh et al., 2000), the monitoring of expression levels of genes that are involved in drug metabolism (Debouck and Goodfellow, 1999; Scherf et al., 2000) and the analysis of plant (Ruan et al., 1998) and animal development (White et al., 1999).

Here the focus is on spotted and photolithographic arrays, as these are the technologies that were used in the development of the methodology presented in this thesis.

1.2.1 High Throughput Genomics: Expression Microarrays

The following is a description of the process and technology involved in generating gene expression data from spotted and photolithographic microarrays. This is followed by

a brief description of the technology involved in generating gene expression data from inkjet printed and bead-based microarrays.

Spotted Microarray Experimental Process

Consider an experiment in which it is of interest to compare the gene expression from two different treatment groups, using spotted microarrays (e.g., comparing cancerous tissue to non-cancerous tissue). The first step of the spotted array experimental process is to determine the genes of interest and create the spotted microarray(s). This is accomplished via the robotic printing of single stranded DNA of the genes of interest (Duggan et al., 1999). There are a number of ways to generate/obtain these probes, the experimenter can use expressed sequence tag (EST) libraries (Gerhold and Caskey, 1996; Marra et al., 1998; Quackenbush, 2001; Wolfsberg and Landsman, 2001) to generate the probes they are interested in or order pre-made oligonucleotides, which are then spotted onto the arrays. It is also possible for the experimenter to purchase pre-printed microarrays from a commercial source.

The next steps are to extract mRNA from each of the two treatment groups, purify it and transcribe it into cDNA. In order for this to bind to the denatured DNA on the array, the mRNA must be reverse transcribed into single-stranded complementary DNA (cDNA) these are known as the targets (Duggan et al., 1999). During this process a fluorescent marker can be attached to each transcript, so that it is possible to identify which genes are expressing in each of the treatment groups. Two of the most common fluorescent markers in use are Cy5, (generally represented in images as red) and Cy3 (generally represented in images as green) (Eickhoff et al., 2002).

Once each of the the two samples has been labelled with its fluorescent marker (i.e., one sample labeled with Cy3 and the other with Cy5) they are combined and pipetted onto the microarray. Here the labelled cDNA targets created from the mRNA samples bind to the denatured DNA probes on the array corresponding to the gene from which the mRNA was transcribed. This binding process is known as hybridization and theoretically targets should only bind to their complementary probes. That is, mRNA-derived targets should only bind to the probe representing the gene from which the mRNA transcript was produced (McLachlan et al., 2004).

After hybridization has occurred, the next step is to wash the slide(s) so as to remove any labelled cDNA which attached to the surface of the slide instead of hybridizing to a spot. The slide(s) are then spun dry in a centrifuge in preparation for scanning. When the slides have dried, the last step is to scan them using lasers of different wavelengths so as to excite the labels to fluorescence (Duggan et al., 1999).

The results of the scanning yields two fluorescent images, one for each channel of the

array (red and green). The images are saved as 16-bit Tiff files which allow for 2^{16} shades in each channel (Schermer, 1999). These image files are usually translated into text files that contain information on the median and average foreground and background intensities for each spot, in each channel (red and green in this example). Also included is information regarding the scanner used, the time and date the image was created, the position of each spot on the array, the ID number and gene name (where it is known), along with spot specific quality control information (Schermer, 1999). The foreground intensity refers to the fluorescent intensity within the circumference of the spot, where the probes were fixed. The background intensity summarises the fluorescence intensity of any labelled cDNA that stuck to the slide rather than hybridizing to the probes.

The two Tiff file images are often overlaid to produce a composite image containing three possible colours, red, green and yellow (McLachlan et al., 2004), which represents the ratio of the intensity at each spot. If treatment group one was marked with Cy5 (red) and treatment group two was marked with Cy3 (green), then a yellow spot would imply that the gene was expressed equally in both conditions, a green spot would imply that the gene was over-expressed in treatment group two and a red spot would imply that the gene was over-expressed in treatment group one. The intensity of the fluorescence at each spot relates to how much labelled cDNA hybridized there, thus the brighter the spot the more cDNA was hybridized (Shalon et al., 1996). This process is illustrated in figure 1.2 (Black, 2002).

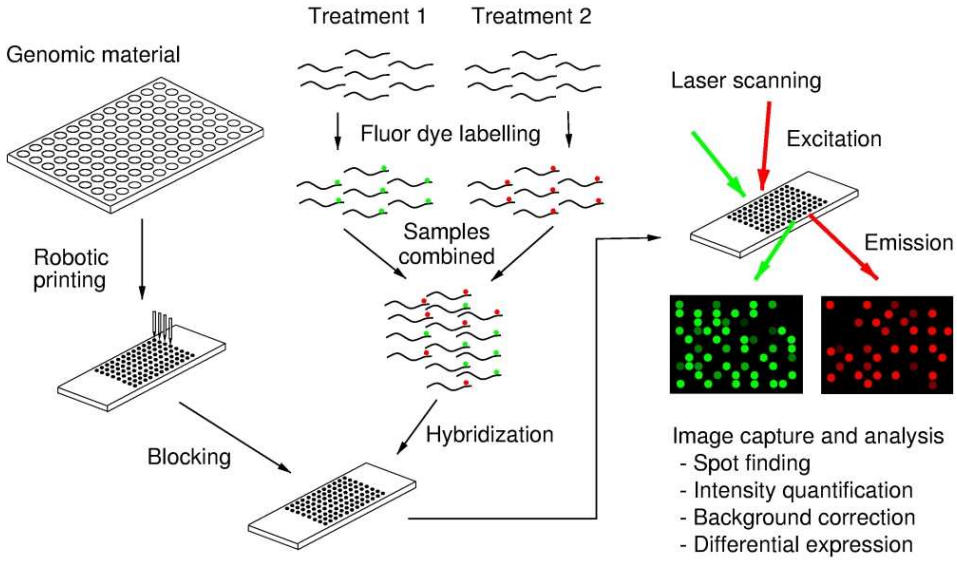

Figure 1.2: Spotted Array Experimental Process (Black, 2002). Genomic material is printed onto the slide, and fluorescently labelled cDNA from the two treatment groups is hybridized to the genomic material. The slide is then laser scanned to produce two images with are combined to give ratios of the intensity of fluorescence at each spot.

Spotted Microarray Technology

The first generation of spotting devices with stepper motors was developed in the laboratory of Hans Lehrach at the Imperial Cancer Research Fund in London between the years 1987 and 1991 (Eickhoff et al., 2002). These devices print thousands of tiny spots of genetic material onto a small surface area such as a microscope slide. These devices achieved spotting densities of approximately 400 spots per cm^2. By 1992 the second generation of spotting devices emerged (Eickhoff et al., 2002), increasing the spotting density to 2500 spots per cm^2. Second generation machines have been constantly upgraded to the point that by 2002 they were using "more accurate and faster drives with better encoders, providing higher sample throughput and superior reproducibility" (Eickhoff et al., 2002).

Two types of pins have been developed, namely "solid" and "split" pins. Eickhoff et al. (2002) state that they both have a delivery range of 0.5 to 5 nl depending on the dimensions of the pins. In first generation machines solid pins were made of stainless steel with a diameter of 0.9mm and a flat print tip. Eickhoff et al. (2002) also state that by 2002 solid pins had evolved to incorporate conical and cylindrical print tips, the use of titanium and tungsten as construction material and a reduction in pin tip diameter to 100μm. Solid pins have the advantage of being easy to clean and sterilize and can perform thousands of sample transfers without loss of spotting performance, however they can only address 1 slide or filter for spotting per loading procedure. As a result this is time consuming, especially when a single spot needs addressing several times.

To combat the time problems associated with solid pins split pins were developed. These have an enclosed cavity and the first generation of these could hold up to 5μl of liquid per loading (achieved by capillary forces), meaning that more than 10 slides could be addressed per loading. Because of their additional complexity over solid pins, they are more difficult to clean and sterilize and their production costs can be up to 100 times higher (Eickhoff et al., 2002). Advancements in pin tip design now give users the ability to specify the diameter of the spot and the number of spots that the pin can address per loading. For example, TeleChem International Inc. offer a wide selection of pin tips under their Stealth Technology brand which can address anywhere between 25 to 915 spots per loading, with spot diameters ranging from 62.5μm to 600μm achieving spot densities for four pins of between 144 and 10,000 spots per cm^2.

An alternative to pins is the so called drop-on-demand method. In principle this method is similar to that of an ink-jet printer. Samples are pipetted onto the array with a multichannel micro-dispensing robot. Each nozzle in this robot can dispense single or multiple drops and each drop has a volume of 100pl. Eickhoff et al. (2002)

reported that this method increased the spotting density to 3000 spots per cm^2 and that it has a high degree of automation on glass slides compared to nylon membranes. MicroFab Technologies Inc. advertise that they fabricate drop-on-demand piezoelectric microdispensing devices that can produce spots with diameters of 25μ to 100μm and volumes of 10pl to 0.5nl at a rate of up to 4,000 spots per second.

Nylon membranes were first used as a support surface because they were proven to have good DNA binding capacity and could be reused up to 10 times. However they have an inherent fluorescent signal which prohibits the use of all fluorescence-based detection methods. Thus glass slides with their low to non-existent fluorescent signal were introduced as a support surface. These slides are coated with a substance to minimise the spreading of the printed spots and in some cases give the surface a positive charge which can help position the DNA on the surface so as to increase the efficiency of cross-linking. Common slide coatings include polylysine, epoxysilanate, aldehyde and various proprietary coatings.

Photolithographic Arrays

One of the most popular forms of photolithographic arrays in current use are those produced by Affymetrix. These arrays are produced by building each probe base by base on the slide, via a process called photolithography (Fodor et al., 1991). At each step of the process a different mask is used in combination with a nucleic acid base so that the next base to be added only couples with the oligonucleotide sequences that require that particular base. The coupling is accomplished via the photolithography process pioneered by Fodor et al. (1991), where light is used to aid in the joining of the bases. This process is repeated until each probe consists of 25 nucleic acid bases (McGall and Fidanza, 2001). For each probe on the array there is a Perfect Match (PM) sequence and a Mis-Match (MM) sequence which differs by one base in the middle of the sequence from the PM sequence (McLachlan et al., 2004). The MM sequence is used to estimate the level of non-specific binding (i.e., noise level) associated with the corresponding PM probe. By subtracting the intensity level associated with the MM sequence from that of the PM sequence, one can obtain an estimated intensity level for the probe. A gene is represented on these arrays by a set of PM and MM probe pairs, known as a probe set, which usually consists of 11-20 probe pairs (McLachlan et al., 2004). To obtain an estimate of the expression value for a probe set the intensities associated with the probes belonging to that probe set are combined. A number of algorithms have been proposed for this (e.g., MAS 5.0 (Hubbell et al., 2002), RMA (Irizarry et al., 2003a,b), gc-RMA (Wu et al., 2004), PLIER (Affymetrix, 2005) and

dChip (Li and Wong, 2001)). For a more detailed explanation of the fabrication process see Stekel (2003). The targets, or labeled cDNA, can then be applied to the slide in a similar manner as for spotted arrays. Two differences between spotted arrays and Affymetrix photolithographic arrays are that Affymetrix arrays use a different type of fluorescent label, biotin, and only one sample is hybridized to each Affymetrix array.

Inkjet Printed Arrays and Bead-Based Arrays

Blanchard et al. (1996) proposed the development of 'Inkjet Printing' of microarrays. This process uses the technology behind standard inkjet printers to build oligonucleotides and whole cDNA strings base by base directly onto the surface of a microarray slide. The oligonucleotide sequences that are required can be given to the printing system or synthesizer in the form a text file which is converted to instructions for the printer. This allows for any set of oligonucleotides to be constructed which can then be placed at any position on the array (Southern, 2001). To create a different array Southern (2001) states that it is as simple as changing the sequence file inputted to the synthesizer. Cooley et al. (2001) provide a detailed description of a procedure for the fabrication of oligonucleotide microspot arrays using inkjet printing technology. This procedure uses a 10-channel inkjet print head to deposit up to 10 oligonucleotide probes directly onto the array surface simultaneously (Cooley et al., 2001). The printing system is controlled and operated using a single control program on a computer. This program allows the user to determine the operating parameters of the printing system, such as how fast the slide is moved between print deposits and how much bioactive solution is deposited at each print, and what printing pattern is to be executed (Cooley et al., 2001). Inkjet printing has the ability to produce a higher consistency across slides and a higher quality of spot uniformity than those produced by spotted array techniques (Grigorenko, 2002).

Bead-based arrays are a technology platform that utilises microspheres. Denatured DNA is immobilised onto the microspheres which are then submerged into the fluorescently labeled mRNA for hybridization. Yang et al. (2001) presented the technology platform "BeadsArray" which makes use of multiple colour-coded microspheres and the Luminex^{100} LabMAP system commercially available from the Luminex Corporation. Bead-based arrays work in a similar way to spotted microarrays except that the denatured DNA is immobilised to fluorescent microspheres rather than to a microarray, the microspheres are then placed into the fluorescently labeled mRNA for hybridization. Once hybridization has occurred the microspheres are passed through a device which identifies and records each microsphere and the fluorescence intensity from the labelled

mRNA associated with the microsphere.

The Luminex Corporation provides sets of up to 100 unique microspheres, each with a different fluorescent colour code, along with well plates which contain 96 wells. Each microshpere can be thought of as a "spot" on a microarray slide or chip. With one set of microspheres per well this gives the potential to examine 9600 unique genes/transcripts in a single plate. Naciff et al. (2005) used the Luminex xMAP system to design a microsphere-based high-throughput gene expression assay to determine estrogenic potential. They reported that the trade-off for the increase in throughput and decrease in cost which were observed when compared with existing microarray technology was a reduction in the total number of transcripts that could be analysed in a single experiment. Specifically, the gene expression assay developed by Naciff et al. (2005) can detect up to 100 different transcripts, 1 transcript per unique microshpere, whereas, the Affymetrix RG_U34A rat chip they used for comparison contains 8799 probe sets, representing approximately 5300 known genes, on a single slide or chip. Jacobson et al. (2006) reported on the variation on the fluorescent measurement of the Luminex^{100} (Luminex Corporation) and how this may affect the analytic quantification.

Another bead-based array is available from Illumina Incorporated, who offer two bead-based systems, which are based on 3-micron silica beads that randomly self assemble in microwells on either planar silica slides or fiber optic bundles. These bead-based systems can be applied to a variety of RNA and DNA analyses (Steemers and Gunderson, 2005). Gunderson et al. (2004) provide a description of the assembly and decoding of a fiber optic bundle bead array offered by Illumina Incorporated, where they concentrated on decoding 1520 unique bead types/probe sequences with 49,777 wells per fiber-optic-bundle and 96 fiber-optic-bundles arranged in such a fashion so as to match the layout of a standard 96-well microtiter plate. Information on how to extend this decoding process for a larger number of unique bead-types/probe sequences was also provided.

1.2.2 Expression Data from Spotted and Affymetrix Microarrays

Spotted Microarrays

A major goal in the analysis of gene expression data from spotted microarrays is the detection of genes that are differentially expressed between two treatment groups. That is, the level of activity for these genes is different between the two treatment groups. Section 1.2.1 described the generation of intensity values for genes using Cy5 and Cy3

fluorescent markers, where treatment one was labelled with Cy5 and treatment two was labelled with Cy3 and both treatments were represented on each microarray.

When Genepix (Axon Instruments, Union City, CA) is used to produce the images from a microarray experiment, 8 intensity values are generated for each gene on the microarray, these are the mean and median foreground and background intensities for each gene, for each of the two fluorescent markers. These values are saved in GenePix Results (GPR) files along with additional information pertaining to the acquisition of the images including the date and time the images were acquired, the type of scanner and version of GenePix software used and information that identifies each gene in terms of its placement on the array. The median foreground and background intensity values for the genes are more commonly used as they are less affected by outliers. To be able to analyse the median intensity values more easily a certain amount of pre-processing and normalisation is needed to remove unwanted experimental effects arising from the different characteristics of the Cy3 and Cy5 dyes or from spatial biases across the array. This can also deal with any overt background noise produced by labelled cDNA that stuck to the slide rather than hybridizing to the probes. It can further account for the severely right skewed nature of the raw median intensity values across all the genes on the array, which is due to the large number of genes that show low intensity values. Doing this yields two intensity values for each gene, one for each of the treatment groups denoted by the two fluorescent markers Cy5 and Cy3.

The ratio of these two intensities (Cy5/Cy3) is known as the fold change. Using this ratio a gene is said to be 2-fold up-regulated in expression activity if it has a Cy5 intensity that is twice as large as its associated Cy3 intensity. Similarly a gene is said to be 2-fold down-regulated in expression activity if it has a Cy3 intensity that is twice as large as its associated Cy5 intensity. A gene is sometimes considered to be differentially expressed between two treatment groups if it has a fold value that is greater than or equal to a pre-determined number (e.g., genes that are 2-fold or more may be considered as being differentially expressed).

As part of the pre-processing of the median intensity values background correction is performed to remove any noise coming from labelled cDNA that fixed to the slide rather than hybridizing to a spot. This is simply done by subtracting the median background intensity from the median foreground intensity for each spot in each channel. It is not always appropriate to do this as the signal from the background can be larger than that in the foreground for some of the spots on the microarray (Eisen and Brown, 1999). In this situation there are several options available. One of these options is to overlook background correction and transform the foreground intensities directly. If one takes this option, however, it is often wise to perform the analysis using the background corrected data first and then the non-background corrected data to see if

this step has had a large impact on the results of the analysis. Another option is to set any background corrected values that are less than zero to one or any small positive number, so as to enable a log transformation.

Another important part of the pre-processing of the median intensity values is to transform the raw Cy5 and Cy3 median intensities. This transformation is needed because these values are severely right skewed and in most cases the variability is not constant at all the intensity levels. To correct this it is common to take the $\log_2$ of each of the Cy5 and Cy3 intensities. Performing this transformation brings the distribution of the intensities to an approximate bell-shape. This means that a gene that is 2-fold up-regulated corresponds to a $\log_2$ ratio of 1 and a gene that is 2-fold down-regulated corresponds to a $\log_2$ ratio of -1. Genes are often considered as having undergone differential expression if their $\log_2$ ratio is greater than or equal to 1 or less than or equal to -1, while genes that aren't differential expressed have a $\log_2$ ratio of zero. This criterion is itself a possible method for testing to see if a gene has undergone differential expression. Nevertheless the data still contains some experimental effects, such as spatial and dye effects, which can mean that using this criterion as a method for testing for differential expression can result in some rather arbitrary results. The major concern when using this criterion as a method for testing for differential expression is that it does not take into account any variability within or between the gene expression levels. It is therefore recommended that any experimental effects be dealt with prior to analysis, although it is possible to simultaneously deal with experimental effects whilst analysing the data (e.g., via the use of ANOVA).

The two main experimental effects that need to be dealt with before any analysis can take place are those arising from spatial bias and the different characteristics of the Cy3 and Cy5 dyes. The general term "Normalisation" is used to describe a collection of methods that can be applied to the raw and transformed data so as to aid in the removal of any experimental effects. One of the earliest normalisation methods was to simply divide the unlogged background corrected data in each channel by the median for that channel. This method, as with almost all normalisation methods, relies on the assumption that only a small number of genes in the experiment would undergo differential expression. Since then there have been further advances into different techniques that can be used to normalise microarray data but for the purpose of this summary only ANOVA normalisation (Kerr et al., 2000), lowess/loess normalisation (Dudoit et al., 2002b), pin-tip loess normalisation (Smyth and Speed, 2003) and the normalisation methods available in version 2.9.13 of the R package **limma** (Smyth, 2005) will be described.

ANOVA Normalisation

This is a linear model approach to normalising the data which is quite straight forward and easy to implement. The idea is to take the logged data in each channel and subtract the mean for that channel. This method works well when the experimental effects are linear but it falls over when confronted with an experimental effect that is non-linear, for example, a non-linear dye effect. This effect can be seen quite easily in a MA plot (Dudoit et al., 2002b) (e.g., Figure 1.3 (a)), where M is the $\log_2$ fold change for each gene (plotted on the y-axis) and A is the average $\log_2$ intensity for each gene (plotted on the x-axis). The effect that ANOVA normalisation has on the MA plot is to centre the data about zero on the y-axis, without affecting the shape of the data cloud.

One of the more prevalent experimental designs is that known as the dye-swap experiment (Kerr et al., 2000). In this design each treatment is assigned to each dye, each dye is present on each array, each treatment is present on each array and each gene is represented on each array. That is, if n is the number of arrays in the experiment, $n/2$ of the arrays have trt1 labeled with Cy5 and trt2 labeled with Cy3, and the other $n/2$ arrays the dyes are swapped. This ensures that the factors *array*, *dye*, *treatment* and *gene* are balanced. As each of the factors are balanced their effects can be separated thus making it possible to estimate the importance of each factor and determine if there is any interaction between the factors. Let X_{ijklm} denote the fluorescent intensity for each spot on each array and let $\mathrm{Y}_{ijklm} = log_2(\mathrm{X}_{ijklm})$ for all the intensities that are greater than zero (if an intensity is zero then set Y_{ijklm} to zero). If Y_{ijklm} can be modeled using a simple linear model where the *array*, *dye*, *treatment* and *gene* effects are additive in nature and there is associated with each data point some random error ϵ_{ijklm} that can't be predicted, then Kerr et al. (2000) demonstrate that it is possible to model the intensities from the spots on the microarray using an Analysis of Variance (ANOVA) model.

The use of ANOVA models allows for both normalisation and analysis to be performed in a single step, whilst properly accounting for the degrees of freedom used in the normalisation step and the easy incorporation of random effects for the experimental factors such as array-to-array variation (Wolfinger et al., 2001). Dudoit et al. (2002b) point out that ANOVA normalisation cannot deal with plate, spatial and intensity dependent effects. For these types of effects they recommend the use of a local regression technique such as lowess or loess. If there are replicate spots, however, for each of the probes, then any spatial/pin effects could be accounted for when using ANOVA Normalisation. This is due to the ANOVA approach being driven by the design/layout of the probes on the array.

Lowess/Loess Normalisation

These methods are known as local regression techniques and scatterplot smoothers. They can be used to remove intensity-dependent and spatial effects without affecting any information related to differential expression.

Lowess was originally proposed by Cleveland (1979). It is a locally weighted scatterplot smoother. The method places a 'window' centred about x on the scatterplot, the data points that fall within this 'window' are weighted such that those closest to x have a higher weight, then a weighted regression is used to predict the value at x. The window is then moved along to centre about $x + 1$ and the process repeated. The width of the window can be controlled by the smoothing parameter f (a value between 0 and 1), that represents the proportion of data that is to be contained in the window. Cleveland et al. (1992) proposed Loess as an extension of Lowess to 3 or 4 dimensions. This extension gives the possibility to fit polynomial surfaces to a data cloud. However it can also be applied in 2 dimensions just as with Lowess.

Dudoit et al. (2002b) proposed that for microarray analysis lowess or loess normalisation be applied directly to the MA plot. Once the smoother has been applied to the MA plot each M value has its associated fitted value subtracted thus removing unwanted experimental effects from the data. Figure 1.3 shows an example of this process on the mouse data contained in the R package **SMA** developed by Dudoit et al. (2002b).

Smyth and Speed (2003) extended loess normalisation so that it fits a loess smoother for each pin used in the production of the array. This is done so as to correct for any differences between the pins as well as broad spatial inconsistencies. The method is known as Print-Tip Loess Normalisation and it is the default method in **limma**'s within-array normalisation function. An example of this method can be seen in Figure 1.4, again using the mouse data contained in the R package **SMA** developed by Dudoit et al. (2002b). In addition to print-tip loess Smyth and Speed (2003) also give global loess, robust spline normalisation and alternative methods which can be invoked in **limma** for use on appropriate array data, such as that derived from Agilent arrays for global loess normalisation and arrays with only a small number of spots per print-tip group. Smyth et al. (2005) describe robust spline normalisation as an empirical Bayes compromise between print-tip and global loess normalisation with five parameter regression splines used instead of loess curves. Smyth and Speed (2003) also describe methods for normalising the individual intensities of a two-colour array via "between array" or "separate channel" normalisation. These methods are based on quantile normalisation procedures and are a prerequisite of individual channel analysis methods. Quantile normalisation procedures are used to ensure that the intensities have the same empirical distribution across a combination of arrays, channels and the A values calculated in the MA plot.

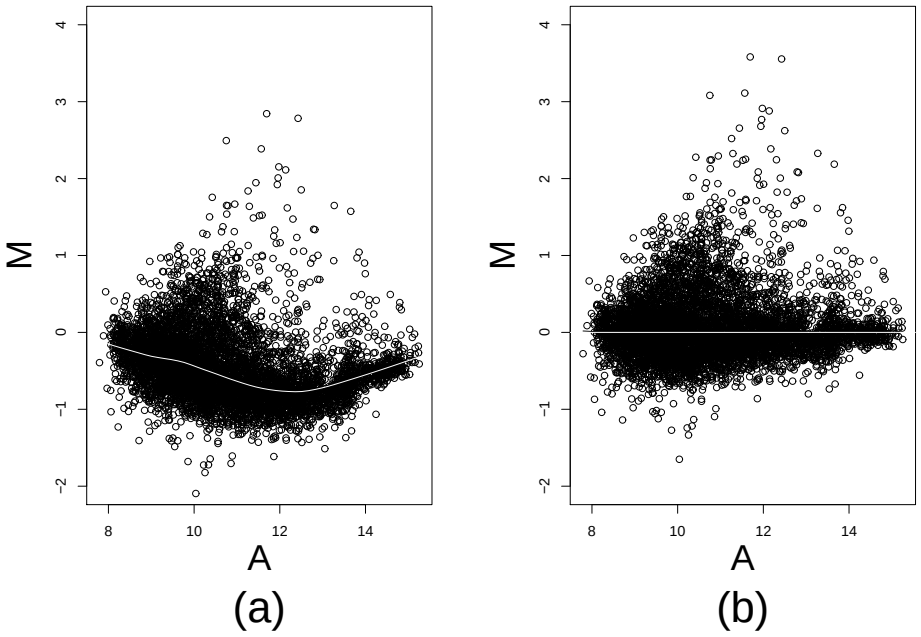

Figure 1.3: Loess Normalisation Process on the Mouse Data Contained in the R package **SMA** developed by Dudoit et al. (2002b). (a) Shows the pre-normalised M & A values with the loess smoothed line produced by the `plot.smooth.line` function in the **SMA** package (Dudoit et al., 2002b). (b) Shows the loess normalised M values *versus* the same A values.

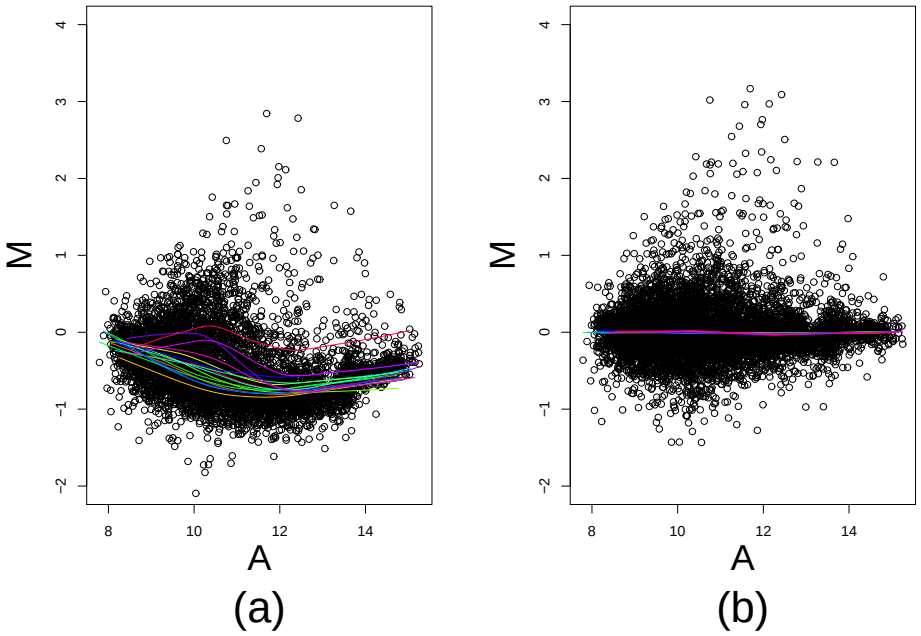

Figure 1.4: Print-Tip Loess Normalisation Process on the Mouse Data Contained in the R package **SMA** developed by Dudoit et al. (2002b). (a) Shows the pre-normalised M & A values with a loess line for each print-tip. (b) Shows the print-tip loess normalised M values *versus* the same A values.

Affymetrix Microarrays

As with spotted microarray data a major goal in the analysis of gene expression data from Affymetrix microarrays is the detection of genes that are differentially expressed between two treatments or phenotypes. A description of how intensity values are generated from Affymetrix arrays was given in Section 1.2.1, where each array represents a single sample under one treatment or phenotype and each gene on the array is represented by a probe set that consists of a group of paired probes, the Perfect Match (PM) and Mismatch (MM) sequences. The latter of these is used to detect any non-specific binding and background noise that might occur during hybridization.

For each probe on the array there are two values, these are the intensity levels associated with the PM and MM sequences for each probe. These intensity values are contained within a CEL file, which also includes information pertaining to the calculation of these values, including the pixel coordinates of each cell on the array, the total number of cells on the array, the algorithm used to create the CEL file along with its parameters, which cells have been masked by the user and which cells the software has labeled as "outliers". As with spotted Microarrays a certain amount of pre-processing and normalisation is required to remove any systematic variation that is present across the arrays so that comparisons between arrays can be performed on a level playing field. One of the first steps taken in pre-processing Affymetrix data is to adjust for any non-specific binding and/or background noise present on the array (Hubbell et al., 2002). This can be accomplished by subtracting the intensity level of the MM sequences from those of the PM sequences for each probe. However, when the intensity level for the MM sequences are greater than that for the PM sequences an alternative measure of background noise needs to be derived. The MAS 5.0 software, provided by Affymetrix (Hubbell et al., 2002), calculates an Ideal Mismatch (IM) intensity for this situation by imputing an ideal value based on the average ratio of all the probe pairs in the probe set. Once this step has been performed the intensity values are then transformed using a $\log_2$ transformation to stabilise the variance (Hubbell et al., 2002).

At this point the $\log_2$ probe intensities associated with the same probe set can be combined to give an estimate of the expression value for each gene so that further normalisation between the arrays is performed using these values or alternatively, between array normalisation can be carried out on the $\log_2$ probe intensities. The algorithm used in the MAS 5.0 procedure applies the first of these options, calculating the average expression level of the probes in each probe set. These average expression values are then scaled so that each array has the same overall average value. This method works well when the relationship between arrays is linear but can fail when confronted with a non-linear relationship (Bolstad et al., 2003).

The robust multi-chip average (RMA) (Irizarry et al., 2003a,b) and guanine cytosine robust multi-chip average (gc-RMA) (Wu et al., 2004) algorithms were designed to normalise and summarise the background corrected probe intensities derived from Affymetrix microarrays. These two algorithms are implemented in the R packages **affy** (Irizarry et al., 2003b; Gautier et al., 2004) and **gcrma** (Wu et al., 2004), which can be obtained through the Bioconductor website (http://bioconductor.org/packages/release/bioc/). Both algorithms follow a three step process that takes as inputs the raw intensity data (i.e., both the PM and MM intensity values for each probe) from each array. In the first step background correction is performed. To achieve this RMA uses a linear model that ignores the MM intensities which as Wu et al. (2004) point out, does not adequately adjust for non-specific binding. This method is based on a convolution model where the observed probe signal (S) is split into two components, the signal (X) and the background (Y) so that S = X + Y (Bolstad, 2008). This model assumes that the signal follows an Exponential distribution with parameter α and the background follows a Normal distribution with parameters μ and σ^2. Bolstad (2008) shows that the expected value of X given the observed signal is calculated to provide a background corrected value for each probe. This expectation is given by:

$$E(\mathrm{X}|\mathrm{S}) = a + b\Big(\frac{\phi(a/b)}{\Phi(a/b)}\Big)$$

where $a = s - \mu - \sigma^2\alpha$ and $b = \sigma$. The parameters in this model are estimated via a non-parametric procedure, which starts by fitting a density estimator to the observed PM probe intensities. This density is used to estimate the mode of the distribution, then points above the mode are used to estimate the Exponential parameter α and the points below the mode are used to estimate the parameters μ and σ^2 for the Normal distribution (Bolstad, 2008). gc-RMA uses probe sequence information, specifically the position of each base type (T, A, C or G) along the probe, to estimate the level of non-specific binding associated with the probes (i.e., probe affinity to non-specific binding) which it then uses in the computation of the background adjusted intensity (Wu et al., 2004). In the second step quantile normalisation is used at the probe level to ensure that the empirical distribution of intensity values is the same for each array (Bolstad et al., 2003). The algorithm that implements this normalisation method first requires that the intensities to be normalised are arranged in a matrix such that each row represents a probe and each column represents an array (Bolstad, 2008). The algorithm starts by sorting the probe intensities for each array from lowest to highest, then at each quantile the intensities are averaged across each row in the sorted dataset. These average intensities are then substituted as the intensity for each specific quantile in each column of the matrix. Finally each column is reordered to return each element

to its original position (Bolstad, 2008). Finally the normalised probe intensities in each probe set are summarised to obtain an expression value for each gene represented on the array, both the RMA and gc-RMA algorithms achieve this by using the median polish algorithm (Bolstad, 2008). This procedure uses a robust multi-chip linear model which is fitted on a probe set by probe set basis. Bolstad (2008) shows the model to be of the form:

$$log_2(y_{nij}) = \mu_n + \theta_{nj} + \alpha_{ni} + \varepsilon_{nij} \tag{1.1}$$

where $log_2(y_{nij})$ is the probe intensity for the ith probe pair in the nth probe set on the jth array on the log_2 scale. Bolstad (2008) shows that for any given probe set n this model is subject to the following constraints; $\text{median}_j(\theta_{nj}) = \text{median}_i(\alpha_{ni}) = 0$ and $\text{median}_j(\varepsilon_{nj}) = \text{median}_i(\varepsilon_{ni}) = 0$. The log_2 expression values are given by $\hat{\beta}_{nj} = \hat{\mu}_n + \hat{\theta}_{nj}$ (Bolstad, 2008). The algorithm used to obtain the parameters in 1.1, starts by forming a matrix of the log_2 probe intensities for each probe set such that probes are presented in rows, arrays are presented in columns and an additional row and column are added containing the row and column effects, which are initially set to zero, so that the matrix is of the form:

$$\left(\begin{array}{ccc|c} \varepsilon_{11} & \dots & \varepsilon_{1N_A} & \mathrm{a}_1 \\ \vdots & \ddots & \vdots & \vdots \\ \varepsilon_{I_n1} & \dots & \varepsilon_{I_nN_A} & \mathrm{a}_{I_n} \\ \hline \mathrm{b}_1 & \dots & \mathrm{b}_{N_A} & \mathrm{m} \end{array}\right)$$

Next, the median across all of the columns except the last column (containing the row effects) is calculated for each row, these values are then subtracted from each element in their respective row and added to the last column. The median across all the rows except the bottom row (containing the column effects) is then calculated for each column, these values are then subtracted from each element in their respective column and added to the bottom row. These two steps are repeated iteratively until the changes converge to zero. When this occurs $\hat{\mu}_n = m$, $\hat{\theta}_{nj} = b_j$, $\hat{\alpha}_{ni} = a_i$ and the ε_{ij} elements in the final matrix will be the estimated residuals for the nth probe set (Bolstad, 2008).

As the median polish algorithm does not naturally provide standard error estimates Bolstad (2008) describes an alternative approach to summarisation. This method is known as the probe level model (PLM), which fits the same model displayed in equation 1.1, where α_{ni} is the probe effect and the errors are assumed to be independent and identically distributed and the only model constraint is that the sum over i of the probe effects is equal to zero. Bolstad (2008) fits this model by robust regression using M-estimation and points out that although this method is somewhat slower than the median polish algorithm it does, however, provide natural standard error estimates.

Another popular normalisation method for Affymetrix data is cyclic-loess normalisation (Bolstad et al., 2003), which can be applied iteratively to either probe intensities or expression values for each gene from two arrays at a time. This method is based on the MA plot, where M is the difference between the two logged intensity/expression values and A is the average of these two values and is implemented in the R package **affy** (Irizarry et al., 2003b; Gautier et al., 2004).

1.3 Summary

The first part of this chapter provided a brief overview of the molecular biology which defines what genes are, where they come from and the process by which genes are directly involved in the production of proteins. This process is known as the Central Dogma of molecular biology it consists of two steps, a section of DNA (a gene) is firstly transcribed into mRNA which is then translated into protein in the second step. A gene that completes this process is said to be expressed and by measuring the amount of mRNA being transcribed one can get an indication of the expression level of the gene.

Section 1.2.1 of this chapter gave a description of the processes and technology involved in generating gene expression data from four types of high throughput expression microarrays, namely spotted arrays, photolithographic arrays, arrays produced by inkjet printing methods, and bead-based systems. These microarrays give the researcher the ability to assess the expression levels of tens of thousands of genes per sample in a quick and efficient manner, using a variety of different experimental designs. Once the gene expression data has been generated, a certain amount of preprocessing is required to clean and prepare the expression data for the analysis process. Section 1.2.2 described a selection of the more common methods used to achieve this preprocessing for expression data obtained from spotted microarrays and the photolithographic microarrays produced by Affymetrix. These methods, known as normalisation techniques, are used to aid in the removal of systematic variation which may be present both within and across the microarrays so that comparisons made between the microarrays can be performed on a level playing field and questions posed at the beginning of the experiment can be investigated.

2

Tools for the Analysis of Gene Expression Data

This chapter is split into four sections. The first section provides a review of the relevant literature published in recent years, focusing in particular on methods for gene/feature selection, binary variable classification methods, simulation, and bias reduction methods suitable for gene expression microarrays. The second section describes in more detail some of the multivariate statistical methodology discussed in section one, including discriminant function analysis, principal components and principal co-ordinates analysis, two cluster analysis techniques, two machine learning classification algorithms and a selection of resampling-based methods. Sections three and four provide a brief overview of databases for the storage and annotation of gene expression data, and software applications for the analysis of microarray data.

2.1 Methods that Have Been Used Recently in Bioinformatic Applications

The methods reviewed in this section focus on those that are relevant to the analysis methods explored in the subsequent chapters of this thesis, namely classification methods, gene selection methods, bias reduction methods and simulation methods suitable for gene expression microarray data.

2.1.1 Classification Methods

The classification of gene expression data to aid, not only in the prediction of relapse in tumor patients, but also the identification of genes that are co-regulated in subgroups of a given population, is an important aspect of microarray data analysis. Early forays into the analysis of microarray gene expression data concentrated mainly on unsupervised classification methods, in particular the use of hierarchical clustering to group together patients and/or genes with similar attributes (Allison et al., 2006). One of the seminal analyses of microarray data was performed by Golub et al. (1999), who used DNA microarray data to differentiate between acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL). Using a weighted voting system they classified new samples as either AML or ALL depending on how similar the new sample was to the centroid for each class (i.e., AML or ALL) in a set of "informative genes". These genes were chosen based on their correlation with an "ideal" expression pattern which was uniformly highly expressed in one class and had uniformly low expression levels in the other class.

Alizadeh et al. (2000) used microarray data as a basis to classify tumours with the aim of detecting clinically significant subtypes of cancer. In particular they were interested in differentiating between different types of diffuse large B-cell lymphoma. Hierarchical Cluster Analysis (using Centroid Linkage on the Pearson correlation distance matrix of the expression data) was used to detect and classify tumours to different subtypes of B-cell lymphoma based on gene expression.

Hedenfalk et al. (2001) used microarray data to classify breast cancer tumour samples into two classes according to the presence or absence of two different genetic mutations. This was achieved via the use of a compound covariance predictor with the misclassification rates estimated using leave-one-out cross validation (LOOCV). The class labels were permuted to determine the significance of the LOOCV results. Since then an exhaustive amount of documentation has been produced on the application and adaptation of statistical classification methods for gene expression data.

Dudoit et al. (2002a) provided a comparison of the performance abilities of three well known discrimination methods on three publicly available datasets. The discrimination methods compared were; the k nearest neighbour (k-NN) classifier (Fix and Hodges, 1951), binary classification trees implemented via CART (Breiman et al., 1984) and linear discriminant analysis (LDA) which included Fisher's LDA (Fisher, 1936) and the maximum likelihood discriminant rules diagonal quadratic discriminant analysis (DQDA) and diagonal linear discriminant analysis (DLDA) (Day and Kerridge, 1967). They showed that the more simple classification methods (e.g. DLDA and k-NN) outperformed the more sophisticated methods on the three publicly available data sets they worked with. These were the leukemia data of Golub et al. (1999), the lymphoma data

of Alizadeh et al. (2000) and the NCI60 dataset first published by Ross et al. (2000).

Hastie and Tibshirani (2004) explored a class of techniques that can reduce the computation time/cost of a selection of statistical classification and regression models when implemented for the analysis of gene expression data. The method proposed was based on the quadratic regularisation of linear models, which makes use of the singular value decomposition (SVD) of the data matrix (Golub and Van Loan, 1983). In doing so it can overcome the dilemma of there being substantially more genes than samples (i.e., $p \gg n$) in microarray data as well as reduce the computational time, as one need only invert an $n \times n$ matrix (comprising the "eigengenes" derived from the SVD of the original data set) instead of the $p \times p$ matrix generated by the ridge regression solution which adds a quadratic penalty to the least-squares fitting criterion associated with standard linear regression models. (i.e., the quadratic regularisation of a linear model). Hastie and Tibshirani (2004) showed that by fitting quadratically regularised models with linear predictors to the $n \times n$ matrix of eigengenes derived from the SVD of the original data set in place of the original data set, it is possible to perform all the aspects of model evaluation, in particular cross validation. This was shown for a selection of regression and classification methods including; Logistic regression, generalised linear models, Cox proportional hazards models, regularised Linear Discriminant Analysis, Neural Networks, Linear Support Vector Machines and Euclidean distance methods. Hastie and Tibshirani stated that one of the major drawbacks of using quadratically regularised linear models for gene expression data is that the solutions involve a combination of all of the genes, so no gene selection is performed and no single gene or sub-set of genes can be identified as being of more importance to the classification process. This is a particular drawback when the goal of an analysis is to identify a small sub-set of genes that can be used in a clinical setting to build a prognostic tool with a high level of accuracy.

Support Vector Machines (SVMs), proposed by Vapnik (1995) as a supervised learning methodology used for both classification and regression, have become a popular choice of classifier for microarray data. One of the reasons for its popularity is that SVMs have been shown to out-perform other classification learning algorithms used in the classification of microarray data, and that they are efficient enough to handle data where the number of variables greatly outweighs the number of samples (Statnikov et al., 2005). Using SVMs to classify samples based on gene expression data is well documented and includes research performed by, for example, Brown et al. (2000) who used SVMs to classify genes into functional roles, and Ramaswamy et al. (2001) who used a multi-class classifier based on the SVM algorithm to determine if it was possible to differentiate between different types of cancer tumours based solely on gene expression data. They concluded by stating that even though the multi-class classifier was highly

accurate, it was not perfect, and that improvement may be achieved by increasing the number of samples in the training set. Zhu and Hastie (2004) compared penalised logistic regression with SVMs for cancer classification, observing that both classification methods seemed to perform at a similar level. Statnikov et al. (2005) reviewed several multi-category classification methods for microarray gene expression cancer diagnosis, finding that multi-category support vector machines were the most effective classifiers in terms of performance accuracy in cancer diagnosis from gene expression data. More recently Hu et al. (2006) investigated the effect of combining different gene selection methods with two benchmark classification algorithms, one of these being SVMs. They observed that in general the performance of both the classifiers was improved when combined with a gene selection method.

Recent literature explores the use of a summary of expression values for genes that are known to be functionally or biologically related, known as gene sets. Svensson et al. (2006) used gene sets based on Gene Ontology terms (see Section 2.3 for a description) and a nearest centroid classification method to discriminate between cancer patients with and without susceptibility to severe late radiotherapy toxicity. They achieved a classification rate of 67% in a small (12 patients) independent validation set.

2.1.2 Gene Selection Methods

Feature selection is arguably one of the most important steps in the analysis of gene expression data. One reason for this is that the number of variables (genes) considerably outweighs the number of samples (arrays) in any given experiment. A wealth of literature is available on proposed feature/gene selection methods, from non-statistical methods such as fold-change cut-offs (Chu et al., 1998), to univariate and multivariate statistical techniques, such as t-test based methods (Cui and Churchill (2003), Tusher et al. (2001), Smyth (2005)) and multivariate extensions to these such as Hotelling's T^2 (Lu et al., 2005).

Early methods for gene selection were based on expression fold change alone (Chu et al., 1998) which does not take into account the variability inherent in gene expression data. Since then more statistically rigorous methods have been proposed or adapted for the selection of differentially expressed genes. These include the t-test (Cui and Churchill, 2003), the SAM adjusted t-test (Tusher et al., 2001), moderated t-statistics (Smyth, 2005), the Pearson and/or Spearman correlation coefficient (Cho, 2002) and many other ranking statistics. An example of applying gene selection methods in a practical setting can be seen in the analysis conducted by Wang et al. (2004) who, in the course of their analysis of gene expression data from 74 patients with Dukes' B colon cancer, used the t-test with Bonferroni corrections for multiple comparisons, to identify genes that showed different expression levels between two patient subgroups

obtained via a hierarchical cluster analysis of the samples using average linkage. The gene with the smallest Bonferroni corrected p-value was selected as the indicator to assign patients into the two subgroups. In this paper they were also interested in identifying gene-markers that could best discriminate between the relapse and disease free patients. To this end, the t-test was used to select genes that best discriminated between the two classes. Univariate Cox proportional hazards regression was also used to identify genes that were correlated with the clinical variable of disease free time (i.e., survival time). Genes discovered by both of these methods were used to build a classification model that accurately predicted the likelihood of a patient relapsing.

All of the gene selection methods described so far are essentially univariate in nature as the statistic is calculated on a gene-by-gene basis and the genes are then ranked by these statistics. For this reason the incorporation of the multivariate nature of gene expression data into the gene selection and classification process has become increasingly popular in recent years, with the adaption of many multivariate techniques for the analysis of gene expression data (Chilingaryan et al. (2002), Szabo et al. (2002)) including the use of Hotelling's T^2 (Lu et al., 2005), Principal Components Analysis (Wang and Gehan, 2005) and Multi-dimensional scaling techniques (Lai et al., 2006) in the gene selection procedure.

Lu et al. (2005) proposed the use of Hotelling's T^2 statistic, in combination with a multiple forward search algorithm, for detecting differential expression between two groups of genes. The algorithm works by systematically finding subsets of genes that maximise the Hotelling's T^2 statistic until all the subsets that produce a p-value smaller than a pre-specified significance level are found. Validation of this method was achieved using a spike-in HGU95 dataset from Affymetrix. Application of this method to two gene expression datasets based on human liver and breast cancers showed the advantages of using their method over the gene selection methods used in the original analyses of the two datasets, not only in the observed increase in the number of genes detected as differentially expressed, but also an increase in the predictive ability of the genes found.

Wang and Gehan (2005) took a different approach to incorporating the multivariate nature of gene expression data into the gene selection process via the use of principal components analysis (PCA). This method starts by finding the minimum number of principal components (M) that can be used to reproduce the general features of the original data as closely as possible. Then using a backward elimination process the number of genes in the data set is reduced, one gene at a time, until the "distance" between the M principal component scores of the reduced dataset and the M principal component scores of the original dataset has reached a point where removing another gene greatly increases this "distance". Validation of this gene selection method was attained using a subset of the NCI60 database, which consists of expression values for

more than 9,000 genes in 60 human cancer cell lines from 9 cancer types. The subset used contained only the cell lines for two types of cancer (leukemia and renal cancer) and 1,375 genes which Scherf et al. (2000) deemed as being able to cluster most of the cell lines according to the nine phenotypes. Further validation of this gene selection method was inferred when the classification model produced using the genes selected using the proposed method gave the same misclassification rate and used fewer genes in the construction of the model when compared to the classification model produced based on all of the 1,375 genes. Wang and Gehan (2005) also favourably compared their gene selection method with the t-test based method and Jolliffe's method (Jolliffe, 1972), which selects genes with the largest absolute coefficients from each of the M leading principal components. Lai et al. (2006) provided a warning that gene selection methods that are overly-complex are prone to over-training and can reduce the predictive performance. Lai et al. (2006) compared several univariate and multivariate gene selection methods and concluded that in five of the seven data sets they used, the univariate methods out-performed the multivariate methods.

Leung and Cavalieri (2003) state that genes never act alone in a biological system - they work in a cascade of networks, thus implying that a gene that is highly expressed will have several biologically related genes that are also highly expressed. This means that lists produced by ranking methods could contain groups of genes that are related/correlated with one another, creating redundancy in the produced list. Several researchers have explored the idea of combining selection methods with the aim of selecting a small subset of non-redundant genes. Wang et al. (2005b) provided an example of this by proposing a combination of a univariate ranking method with hierarchical clustering to produce "marker" genes that give the same or better leave-one-out cross validation accuracy in the classification methods they employed when compared to genes chosen via the ranking method alone. Hu et al. (2006) also looked at combining gene selection methods. They consider the combination of four different ranking methods with two wrapper methods, SVM with recursive feature elimination and the tree based classification method C4.5 and found that, in general, the performance of the classification methods improved with the use of certain ranking methods. Their results implied that the combination of ranking method and classifier has an impact on performance. Of the four ranking methods they considered only one of the methods improved the performance of both classifiers over a range of different data sets and reduced the number of genes/variables used in the classifier.

There has also been an increase in the use of sets of genes that are functionally related in the search for expression differences between different phenotypes. One such example of this is reviewed by Shi and Walker (2007) in which they described the method known as Gene Set Enrichment Analysis (GSEA) (Mootha et al., 2003; Subramanian

et al., 2005) which uses a two step process to determine if a predefined set of genes is differentially expressed between two levels of a clinical outcome of interest for two phenotypes. This process results in the calculation of an "enrichment score" which indicates the level to which the set of genes is associated to the phenotype. Shi and Walker (2007) go on to describe a selection of extensions and alterations to this process as well as discussing some of the algorithm's limitations.

Curtis et al. (2005) gave a description of the advantages of using pathway-based tests for differential expression between differing phenotypes, stating that methods such as these have the ability to identify more subtle changes in expression levels compared to gene lists that are a result of univariate statistical analyses. They also noted that these methods can be used with a variety of different biological data types and go so far as to say that these methods will see increasing use as a way to integrate data from different biological sources. Curtis et al. (2005) also compared the performance of three pathway-based methods to determine if they produce similar results. The three methods they compared aimed to determine the enrichment of a given pathway or classification in a list of genes. These types of methods involve some form of comparison of a list of genes with the genes in a pathway or the genes used in a prognostic classifier, to determine if there is a larger overlap between the two lists of genes than would be expected by chance. The first method they looked at was the standardised difference score (or z score). This is a "hit-counting" method based on the hypergeometric distribution. The second method compared was the binomial approximation to the hypergeometric distribution which calculates the probability of there being a specific number of genes from a given pathway in the gene list of interest. The third method was the Gene Set Enrichment Analysis (GSEA) method (Mootha et al, 2003; Subramania et al, 2005). This method involves working down a list of genes (usually all the genes on the array) that has been ordered by some measure of differential expression, comparing each gene in turn to the pathway of interest. If the gene appears in the pathway then the enrichment score is increased, if the gene doesn't appear in the pathway the enrichment score is decreased. As the list of genes is descended the maximum enrichment score is retained as the measure of enrichment for that pathway in the list of genes. These three methods were compared using two datasets from different mouse microarray platforms. Curtis et al. (2005) found a good degree of overlap between the binomial and z score analyses. Their results showed similar p-values even though one method calculated these directly and the other used a permutation method to obtain p-values. They also reported a smaller degree of overlap between GSEA and the other two methods, stating that GSEA showed more pathways as being down-regulated compared to the other two methods.

2.1.3 Bias Reduction Methods for Microarray Data

Obtaining bias free inferences is a very important aspect of all statistical analyses and, as such, methods for avoiding, assessing and correcting for bias in the analysis of microarray gene expression data need to be taken into account. Ambroise and McLachlan (2002) discussed the assessment and correction of selection bias in the classification process of gene expression data, emphasising the necessity of applying any cross validation or bootstrap methodology external to the gene selection step to avoid introducing selection bias into the prediction error of the classification model built using the selected genes. They compared the apparent error (AE) rate, or re-substitution error rate, and the leave-one-out cross validation error rate applied after gene selection (LOOCVES) to the LOOCV error rate applied so that gene selection is performed within each step of the cross validation procedure (LOOCVIS) and the .632 bootstrap (B.632) error estimate using two publicly available datasets and two different classification methods (backward elimination with SVM and forward selection with Fisher's Rule). In all four combinations of dataset and classifier the AE and LOOCVES error rates were markedly smaller and hence more biased than the LOOCVIS and B.632 error rates, thus demonstrating how important it is to be aware of how the gene selection procedure is applied within the classification process. To illustrate their point even further they used one of the datasets to generate 20 "no-information" training sets by permuting the class labels, then by using a recursive feature elimination method a SVM classifier was built for each dataset and the average AE, LOOCVES, LOOCVIS and B.632 error rates were calculated and compared. The two methods that did not account for selection bias (AE and LOOCVES) gave average error rates close to zero whereas the two methods that did correct for selection bias (B.632 and LOOCVIS) gave average error rates that were consistent with the fact that the classifiers were based on non-informative training sets. The authors further emphasized that in order to obtain unbiased classification error rates the test set must be truly independent of the feature/gene selection process.

Molinaro et al. (2005) give a detailed comparison of several methods for estimating the "true" classification error rate when gene selection is present. These methods included the more traditional training and test splits (i.e., split sample) where a given proportion, p, of the data is set aside as test data ($p = 1/3$ and $1/2$ were tested in this paper), v-fold cross validation with $v = 2$, 5 and 10, leave-one-out cross validation (LOOCV), the .632$^+$ bootstrap technique and Monte Carlo cross validation (MCCV), where for a sample of size n, a given proportion, p, and a specified number of repetitions r, the sample is split so that np of the observations are randomly selected and set aside as the test data for each of the r repetitions (i.e., repeated split sample). To compare these re-sampling methods Molinaro and colleagues used a combination of three different data types (1 simulated dataset, 2 microarray datasets and 1 proteomic

dataset) and four different classifiers, namely; Linear Discriminant Analysis (LDA), Diagonal Discriminant Analysis (DDA), Nearest Neighbour (NN) and Classification Trees (CART). Each data set was randomly split into learning and validation sets so that the learning set consisted of either 40, 80 or 120 samples (dataset permitting) with equal numbers of cases and controls (the two classes). The learning sets were then split again according to each of the aforementioned resampling methods, in turn, into training and test sets so as to optimise each classifier. To give a baseline to compare these resampling methods to, each classifier was built on the learning set and the error rate calculated using the validation set. Molinaro and colleagues produced the following conclusions:

- That in the presence of small sample sizes the split sample and 2-fold cross validation methods perform poorly, attributing this to an increase in bias which resulted from using a smaller sized training set which can drastically reduce the ability to produce an effective classifier.
- That when observing the mean squared error and bias, LOOCV generally gave the best performance.
- That in almost all settings the prediction error estimates obtained using 10-fold cross validation were very similar to those obtained using LOOCV.
- That when a dataset is not highly separable (i.e., the distributions of the two classes significantly overlap one another) the $.632^{+}$ bootstrap prediction error estimate gave the best performance.
- That there was not enough of a difference in the bias and mean squared error estimates between the MCCV and v-fold cross validation methods to warrant the use of MCCV over v-fold cross validation.
- And that as the sample size increased the observed differences between the resampling methods decreased.

They also noted that the differences between the resampling methods decreased as the amount to which the distributions of the two classes overlapped increased, that is the separability of the classes decreased.

Varma and Simon (2006) recommend that to avoid introducing bias in the process of developing a classification model using cross validation, any comparative selection of the parameters needs to be done within the training step of the cross validation method applied, so that the test data is not used in optimising the classifier and thus remains independent. Varma and Simon proposed that to obtain cross validation error estimates that are an unbiased estimate of the true error a nested cross validation method should be adopted, that is, in the training set within a given cross validation method, an

additional layer of cross validation should be used to optimise the classifier of interest. The internal cross validation method need not be the same cross validation method as the external method. Varma and Simon demonstrated a nested cross validation method that used 10-fold cross validation as the internal method and leave-one-out cross validation as the external method. Using simulated data Varma and Simon (2006) demonstrated the effect on the bias of the error estimates when using a nested cross validation method to calculate the error estimates in comparison to using a single cross validation method to optimise a classifier. The classifiers used in this comparison were the Shrunken Centroids Classifier and a SVM classifier. Two types of data were simulated, one set representing no difference between the two classes (i.e., the "null" data) and the other representing the case where a selection of the simulated genes are differentially expressed between the two classes (i.e., the "non-null" data). They showed that when using the "null" data the bias of the error estimates decreased substantially when using a nested cross validation method (10-fold cross validation within leave-one-out cross validation) compared to that obtained using a single cross validation method (10-fold cross validation) to select the model parameters that minimise the error estimate.

2.1.4 Simulating Microarray Data

Using simulated microarray gene expression data to evaluate the performance of new and existing analysis methodologies can not only reduce the cost involved in testing analysis methods on real microarray data, it can also give the researcher a better grasp of how well the analysis method performs under a controlled setting and if the analysis method performs to a satisfactory level to warrant its use on real microarray gene expression data.

Nykter et al. (2006) proposed a method for simulating microarray data that takes into account the biological and manufacturing processes that can affect the quality of real microarray data. The proposed simulation method consists of six different modules, each targeting a specific issue in the production of microarray data. Taken sequentially each module deals with a particular aspect of the generation of microarray data, from determining the number and locations of each spot on the array and the type of array (user can choose between spotted two-channel microarrays or oligonucleotide based single-channel microarrays), through to the modeling of biological errors related with, for example, sample preparation and the measurement errors incurred that can arise due to the limitations of the measurement technology used. One of the modules is solely concerned with modeling the process in which microarrays are manufactured, taking into account not only variations in the size and position of individual spots but also variations in the configuration of groups of spots (i.e., subarrays within the microarray)

which can result in the subarrays shifting from the ideal rectangular layout. The effects induced by the hybridization, scanning and image reading steps in the generation of microarray data are taken into account by this simulation method via a separate module for each of these steps. Each of these modules focuses on incorporating the effect each step has on microarray data into the simulation process so that the resulting data shows realistic biological and statistical characteristics. Nykter et al. (2006) have ensured that at every step of this simulation method the end-user can control how the simulated data is produced by, for example, either adjusting the parameters of the default effect models or by replacing an error model with a different one. One feature of this simulation method is that it produces images similar to those obtained using real microarray slides, so in the module that deals with the image reading step it is possible to use any software system the end-user wishes. The flexibility of this simulation method means that it can be used to simulate data from a variety of different settings and so can be used for evaluating different kinds of analyses. Nykter et al. (2006) provided a software package that implements their simulation method in Matlab. This package is available online as a supplement to their paper, however, efforts made to implement this software during the course of the research conducted for this thesis were unsuccessful. This was due to an error in compatibility between the software package and the Matlab version, which unfortunately could not be resolved.

Singhal et al. (2003) proposed a simulation method which mimics data from a single channel microarray experiment comparing "normal" tissue to "diseased" tissue to find genes that are significantly differentially expressed between the two states. Starting with data representing a baseline "normal" sample ("Normal-1"), taken from any available microarray hybridization experiment, be that a single experiment/array or an average of experiments/arrays, this simulation method assigns a defined change to the expression levels of each gene in "Normal=1", by choosing the percentage of genes to be changed and the degree of fold change to be used. The default is that 5% of the genes will have a two-fold change in expression. The new data is labelled as "Cancer-1" and information pertaining to how each gene was changed is retained as an "answer key" for assessing the ability of the algorithm of interest to rediscover these changes. The final step in this simulation method is to impose three layers of variability onto both "Normal-1" and "Cancer-1" in an effort to account for and include the random biological inter-array variation, the random technical intra-array variation and the systematic inter-array variation seen in actual microarray experiments. Systematic inter-array technical variability such as the variability that results from differences in hybridization conditions between arrays is the first type of variability that Singhal et al. (2003) account for. This is achieved by multiplying each of the starting samples ("Normal-1" and "Cancer-1") by a different multiplicative factor creating two new samples ("Normal-2"

and "Cancer-2") this is repeated until the desired number of samples for each state is reached. The multiplicative factors are chosen so that the average change in the overall expression levels between each of the new samples and the overall expression level of the starting samples is ± 20%. To model random intra-array technical variability Singhal et al. (2003) then add or subtract a randomly generated value to each gene in each sample, that is up to 30% above or below the average expression level for each sample. Finally they impose biological variability via the addition of a randomly generated value that is up to 30% above or below the expression level of each gene in each sample so that, each gene in a given sample is changed by the same proportion/percentage and that this proportion/percentage is different for each sample. Singhal et al. (2003) showed that data simulated using this method showed very similar characteristics to data that was derived from an actual experiment. They then used their simulated data to compare and evaluate three normalisation methods and three gene selection procedures.

Several researchers have used more simplistic methods to simulate data that mimics gene expression data, these include the use of the Normal or Multivariate Normal distributions using different means for the different states, combining distributions to form a mixed distribution to capture the mixture of differentially expressed and equivalently expressed genes in the "disease" state. The parameters for these distributions may be based on observed data either by directly calculating the observed means and variances or by fitting a model to the data. Varma and Simon (2006) used the Multivariate Normal distribution to simulate gene expression data, changing only the mean for genes set as differentially expressed. Molinaro et al. (2005) simulated genes that were differentially expressed using a mixture of two Multivariate Normal distributions with different means, the same sample variance-covariance structure and a mixing probability of 0.5.

2.2 Statistical Methods Suitable for the Analysis of Gene Expression Data

This section provides details on how a selection of analysis methods are implemented. The analysis methods were chosen based on their popularity in the field, their suitability for analysing microarray gene expression data and relevance to the analysis methods explored in subsequent chapters of this thesis.

2.2.1 Multivariate Analysis Methods

Discriminant Function Analysis

In microarray analysis it is sometimes of interest to find a way to allocate arrays/samples into one of two or more predefined groups. For example, allocating samples into either recurrence or non-recurrence of a disease. Discriminant Function Analysis is the name given to a group of techniques that can be used to facilitate such an assignment. Two of the more popular and well documented of these techniques are discussed here; Linear Discriminant Function Analysis and Quadratic Discriminant Function Analysis. The discriminant function that underpins all of the discriminant function analysis techniques was proposed by Fisher (1936).

Linear Discriminant Function Analysis

Linear Discriminant Function Analysis (LDA) works by calculating a linear function for each group based on the allocation of data in the training set into the "closest" group. LDA makes two assumptions, firstly that the data follows the multivariate normal distribution. Secondly, and more important, that the variance-covariance matrices of each group are the same, so that a single pooled variance-covariance matrix can be used. McArdle (2001) gives a good intuitive explanation of LDA, starting with two groups in unidimensional space then extending to three groups in two dimensional space. In multidimensional space the first step in LDA is to adjust for the variance-covariance structure in the data, which can be done by rescaling the space by the inverse of the within group covariance matrix. McArdle (2001) suggests that the easiest way to achieve this adjustment is to rescale the distance matrix directly, that is, to use Mahalanobis distance rather than Euclidean distance as a measure of closeness. This adjustment sphericises the data clouds, as demonstrated in Figure 2.1. The next step is to construct a 'boundary' (or boundaries) between the groups, which can be adjusted to take into account prior probabilities for each group if they are known. These boundaries are derived from the discriminant function (Fisher, 1936). For example, if there are two groups and an observation is equally likely to belong to either group then the boundary

can be constructed to lie halfway between the two group means. A new observation will then be allocated to a group depending on which side of the boundary it lies on.

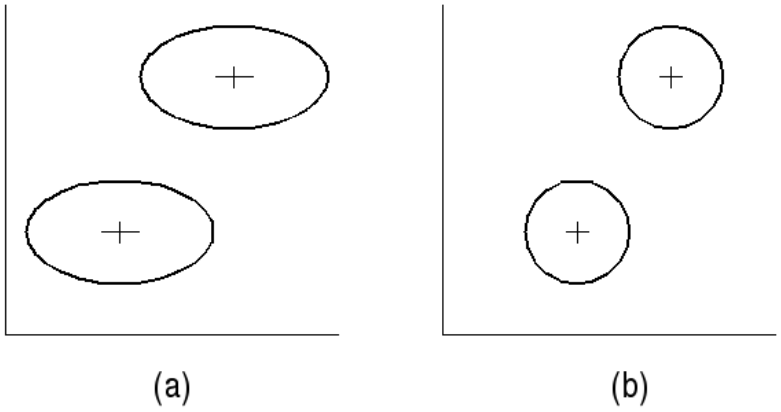

Figure 2.1: Graphical Example of the Difference Between (a) Euclidean and (b) Mahalanobis Spaces.

Quadratic Discriminant Function Analysis

As with LDA, Quadratic Discriminant Function Analysis (QDA) assumes that the data conforms to a multivariate normal distribution, however, instead of using the pooled variance-covariance matrix QDA uses the individual variance-covariance matrices of each group in calculating where the boundaries will lie, bearing in mind any prior group membership probabilities. This means the boundaries are quadratic in nature instead of being linear as an in LDA. Figure 2.2 illustrates the difference between LDA and QDA. It would seem then that if the variance-covariance matrices of the groups differ QDA would be a more appropriate method to use over LDA, however McArdle (2001, Chapter 13, page 15) points out that QDA is "very sensitive to small sample sizes". This is because the variance-covariance matrix for each group has to be calculated.

Seber (1984) summarised a number of studies that looked at the two group situation and made three main recommendations:

1. If the covariance differences between the groups are small and there are less than seven variables, LDA and QDA generally produce the same results.

2. If the groups each have less than twenty five observations in them and the covariance differences between the groups are large and/or there is a large number of variables, then LDA is more appropriate. However if there are large covariance differences between the groups and a large number of variables then neither method may be appropriate, in which case a non-parametric approach should be taken.

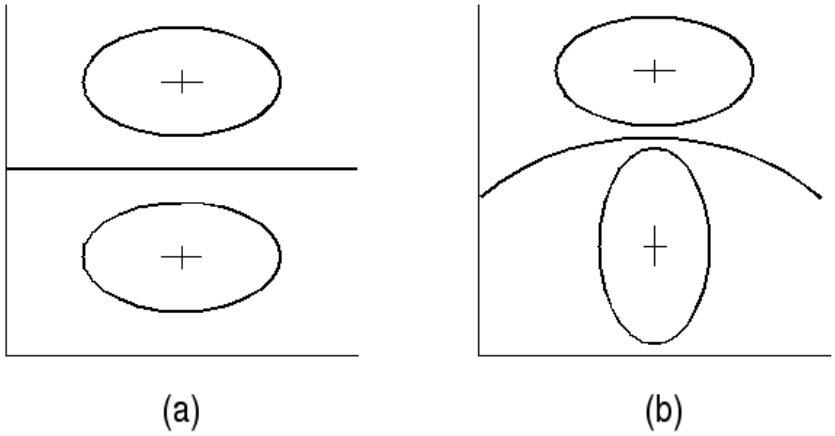

Figure 2.2: Graphical Example of the Difference Between (a) LDA and (b) QDA.

3. If the covariance differences between the groups are large and there are more than six variables and sufficiently large sample sizes then, QDA will perform better than LDA.

Seber (1984) also provided some recommended sample sizes for the two group case when considering the use of QDA. These recommendations were that if there are four variables, each group should have twenty five observations and that for every two additional variables an extra twenty five observations are needed. Seber (1984) also noted that if there are more than one hundred observations in each group then QDA will be generally superior.

Principal Components Analysis

Proposed by Pearson (1901) and later developed by Hotelling (1933), Principal Components Analysis (PCA) is one of the most well known exploratory multivariate analysis methods. It wasn't until the advent of the electronic computer, however, that PCA became widely used and that it is now available in virtually every statistical computer package (Jolliffe, 1986).

McArdle (2001) and Jolliffe (1986) describe PCA as a technique that aims to reduce the dimensionality of an unstructured sample of multivariate data whilst keeping as much of the initial variability as possible, thereby losing only a fraction of the information contained in the data. This reduction is done via an eigenvalue-eigenvector decomposition of a symmetric matrix (either the correlation matrix or the variance-covariance matrix), however, for numerical accuracy singular value decomposition of the original data matrix is recommended by Mardia et al. (1979). The easiest way to describe this method is through an example. Table 2.1 lists an imaginary dataset consisting of two variables and twelve observations that roughly conform to a multivariate

normal distribution with a negative trend. Figure 2.3 gives a graphical description of how PCA works for two variables, based on the variance-covariance matrix. Figure 2.3(a) is a scatterplot of the raw data. The first step in PCA is to centre the data, that is subtract the variable means, this can be seen in Figure 2.3(b). The second step is to rotate the axes so that one of them lines up with the major trend in the data, this new axis is known as Principal Component 1 (PC1) and the second (orthogonal) axis is PC2 (and so on for higher dimensions). This rotation is depicted in Figure 2.3(c). The last step is to project the data points onto the new axes as shown in Figure 2.3(d). Looking at PC1 in Figure 2.3(d) it is clear that PCA has effectively reduced this data set from two dimensions to one dimension, whilst leaving the relative positions of the observations largely unchanged. What this means is that by looking at the first few PC's relative to the old axes, it is possible to describe the major trend in the data in a dimensionally reduced space. This plot is known as a reduced space plot.

Table 2.1: Example Data for PCA

x1	x2
0.20	4.8
0.25	4.9
0.60	4.1
2.00	3.5
2.20	4.2
3.00	3.1
3.10	2.5
3.80	2.1
3.90	1.6
4.10	2.6
4.20	1.7
4.90	1.1

Given that microarray data has as many variables as there are genes on the array(s), which can reach into the tens of thousands, this method is certainly useful for reducing the dimensionality of microarray data and thereby aiding in the discovery of any major trends in the data (Bair and Tibshirani (2004), Zhang et al. (2004) and Wang and Gehan (2005)).

Metric Multidimensional Scaling

Metric Multidimensional Scaling (Schoenberg, 1935; Young and Householder, 1938; Torgerson, 1952) is the name given to a group of techniques that aim to "portray the data's structure in a spatial fashion (that is) easily assimilated by the relatively untrained human eye" (Young and Hamer, 1987). These techniques all start with what

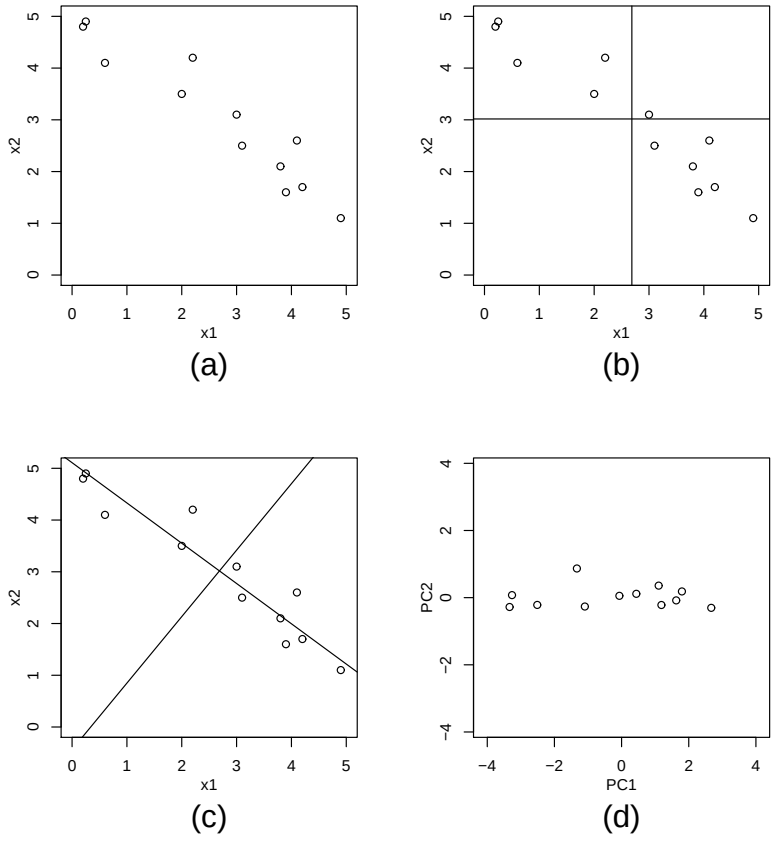

Figure 2.3: Graphical Example of PCA for Two Variables. (a) shows the data in Table 2.1, (b) has the means for x1 and x2 superimposed as axes, (c) shows these axes rotated so that the first axis has the most variation and (d) shows the second principal component versus the first principal component.

is known as a dissimilarity matrix. This is a matrix which consists of the inter-point distances between each observation and every other observation in the data set.

A classic Metric Multidimensional Scaling technique is that known as Principal Co-ordinates Analysis (PCO) (Gower, 1966) which stems from the work of Schoenberg (1935), Young and Householder (1938) and Torgerson (1952). McArdle (2001) describes PCO as a method that "finds a configuration of points that optimally reproduces the distance matrix in a Euclidean space of fewer dimensions". McArdle (2001) gives the following intuitive explanation of PCO using a Euclidean dissimilarity matrix:

> ... try to imagine a cloud of points hanging in a space of unknown dimensionality. These points have no coordinates since as yet there are no axes (only distances were given). The problem is to impose a set of axes on the space and then locate the points on them.

By transforming the dissimilarity matrix into the form of a cross-product matrix (YY^T) for a set of coordinates (Y), an eigenvalue-eigenvector decomposition of the

transformed matrix can be done. As McArdle (2001) points out this decomposition is also used in PCA (Section 2.2.1), where the aim was to use the eigenvectors to project the original data points into a reduced space. With PCO the eigenvectors (once scaled by the square root of the eigenvalues) are the vectors of the observation scores in the reduced space (i.e., the principal axes). This is because PCO uses a dissimilarity matrix rather than the original data matrix to calculate the cross-product matrix that is then reduced via an eigen decomposition. It is also possible to derive the eigenvectors and absolute value of the eigenvalues of the cross-product matrix by determining the singular value decomposition of the original column mean-centered data matrix. To expand on this, if X is an $n \times p$ rectangular data matrix, with n samples and p variables and has had the column means subtracted from each column (i.e., centered by column means) then by singular value decomposition

$$X = V\Delta U^T$$

where V are the eigenvectors of XX^T the cross-product matrix derived from the dissimilarity matrix of X used in PCO, where the distances are measured between the rows, Δ is the diagonal matrix of the square root of the eigenvalues and U are the eigenvectors of X^TX. The principal component scores for PCO are obtained by $V\Delta$ and the projection matrix can be derived from a simple algebraic manipulation of the equation above to obtain $V\Delta$ in terms of X and U:

$$\begin{aligned} X &= V\Delta U^T \\ XU &= V\Delta(U^TU) \\ XU &= V\Delta \end{aligned}$$

so that the projection matrix is U. A new column mean centered data matrix Z $(m \times p)$ can then be projected into the reduced space by the matrix multiplication of Z by the projection matrix (i.e., ZU). As with PCA the first principal axis describes the most variation or major trend in the data cloud, the second principal axis the second largest variation, and so forth. Gordon (1981) makes the interesting point that if the dissimilarity matrix is Euclidean-based then the results of a PCO will be identical to those obtained from a PCA on the covariance matrix of the data matrix from which the dissimilarity matrix was calculated. McArdle (2001) also notes that although PCO works best on Euclidean dissimilarity matrices it can also give meaningful results on non-Euclidean dissimilarity matrices.

Cluster Analysis

Cluster Analysis can be split into two areas, methods that partition the data, and methods that seek to find and uncover any hierarchical structure that already exists in the data. Both of these areas span a diverse range of techniques, the more popular of these are described below. Cluster analysis differs from some of the other techniques discussed so far, in that there is no knowledge of any predefined groups or group structure in the data. It can therefore be thought of as an exploratory tool that is looking to see if there is any natural grouping or hierarchical structure in a data set. Most clustering techniques use some type of distance or dissimilarity measure to search for or partition the data set into clusters or groups. Choosing which clustering method and distance measure to use needs to be given a considerable amount of thought, taking into consideration not only the nature of the data being analysed but also the objectives of the analysis, as using different combinations of clustering technique and distance measure can produce different clusterings of the data being analysed.

Partitioning Methods

One of the more popular partitioning methods available is that known as k-means (MacQueen (1967); Hartigan (1975); Hartigan and Wong (1979)), where k is the number of clusters that the data is to be partitioned into. This value is usually predefined, however it can be quite prudent to run the analysis with different values of k ($k \in$ {2:Number of observations}) and choose the most appropriate (or natural) number of clusters from the results (Everitt et al., 2001). As with other partitioning methods k-means has three general steps:

(i) choose the number of clusters wanted.

(ii) assign an initial allocation of the data to these clusters.

(iii) reallocate the data between the clusters until there is no further "improvement" to the resulting clusters.

The k-means clustering algorithm (MacQueen (1967); Hartigan (1975); Hartigan and Wong (1979)) implemented in R (R Development Core Team, 2005), uses the within-class sum of squares from the centres of the k predefined clusters as a measure of "improvement" (Venables and Ripley, 1999). The aim is to minimise this value by swapping pairs of data points between clusters. Once the number of clusters has been settled on and the optimal reallocation of the data has been found, the cluster solution can be superimposed onto a reduced space plot arising from a PCA on the data.

Hierarchical Methods

There are two types of hierarchical methods, agglomerative techniques (Ward, 1963) and divisive techniques (Sonquist and Morgan (1963); MacNaughton-Smith (1965); MacNaughton-Smith et al. (1964)). Agglomerative techniques start from the individual data points and fuse them together to get clusters until there is only one cluster consisting of all the data points. Divisive techniques start with the entire data set as a cluster and split it into groups until there are as many groups as there are individual data points (Hand, 1981). Of these two techniques Everitt et al. (2001, page 56) suggests that "agglomerative procedures are probably the most widely used of the hierarchical methods". Kaufman and Rousseeuw (1990), however, point out that divisive techniques reveal the main structure of a data set from the outset which can be of greater use to the analyst.

This fusing and splitting is generally done with distance or similarity thresholds. For agglomerative methods the distance threshold is set at zero to start with and is steadily increased. Five of the most popular methods to measure distance in agglomerative techniques are; Single Linkage (Sneath, 1957), Complete Linkage (Sorensen, 1948), Average Linkage (Sokal and Michener, 1958), Centroid Linkage (Sokal and Michener, 1958), and Ward's method (Ward, 1963).

Single Linkage (also known as the Nearest Neighbour method) defines the distance between two clusters to be the minimum distance from one data point in one cluster to another data point in another cluster. This is illustrated in Figure 2.4(a). Everitt et al. (2001) point out that using this method with a large data set will often result in unbalanced and elongated clusters. They also note that the method does not take into account any structure in the clusters.

As Aldenderfer and Blasfield (1984) state, Complete Linkage (also known as the Furthest Neighbour method) is the logical opposite of Single Linkage. As depicted in Figure 2.4(b) the linkage rule for Complete Linkage states that any data point that is to be included into an existing cluster must be within a certain distance of all the members of that cluster (Sokal and Michener, 1958). Aldenderfer and Blasfield (1984) state that this method has a tendency to produce "relatively compact and hyper-spherical clusters composed of highly similar cases". Everitt et al. (2001) adds the fact that as with Single Linkage this method does not take into account any structure in the clusters.

Average Linkage was developed as an "antidote to the extremes" of the complete and single linkage methods (Aldenderfer and Blasfield, 1984). The distance between two clusters is defined to be the average distance between pairs of data points, where one data point is in one cluster and the other in an other cluster, as can be seen in Figure 2.4(c). Everitt et al. (2001) note that this method tends to join clusters that

have small variances and that unlike single and complete linkage, this method does take into account the structure of the clusters. They also remark on the robustness of this method.

Centroid Linkage defines the distance between two clusters to be the Euclidean distance between the centroids (or mean vectors) of each cluster (Figure 2.4(d)). Obviously this assumes that the data points can be represented in Euclidean space. While it is possible to use a different measure of distance for this method, Anderberg (1973) has shown that the results would not be interpretable in regards to the raw data.

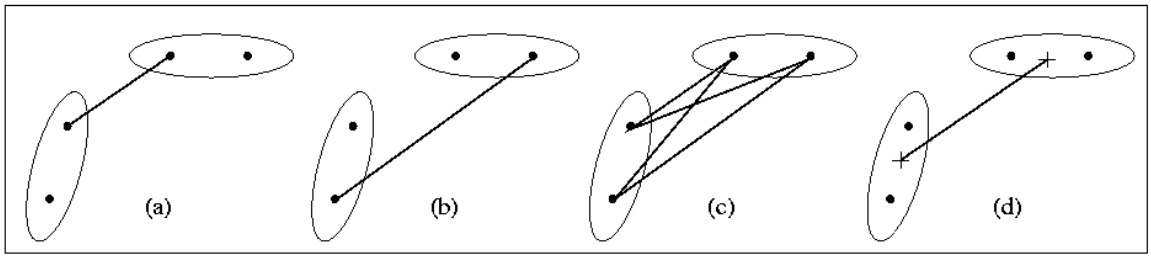

Figure 2.4: Graphical Examples of Linkage Methods (a) Single Linkage, (b) Complete Linkage, (c) Average Linkage and (d) Centroid Linkage (McArdle, 2001).

Ward's method joins two clusters together based on the size of an error sum of squared distances, with the aim of increasing the overall within-group sum of squared distances by the smallest amount possible at each fusion (Ward, 1963). This is equivalent to minimising the sum of squared within-group deviations from the centroids, that is, minimising the trace of the pooled within-group covariance matrix (Gordon, 1981).

MacNaughton-Smith et al. (1964) proposed a splitting method for the divisive methods that works by first finding the data point which is the most isolated (or furthest away) from the rest of the data points in the whole data set, using a proximity matrix. This point is then used as a seed for the new splinter cluster. The second step is to find all the data points that are "closest" to the seed. These data points form the new splinter cluster. The third step is to find the largest distance between two data points in each of the clusters, the cluster that has the largest of these distances is chosen and steps one to three are repeated. This process continues until the required number of clusters is reached or each cluster only contains one data point.

Kaufman and Rousseeuw (1990, page 254) described this method as follows:

> the mechanism somewhat resembles the way a political party might split up due to inner conflicts: firstly the most discontented member leaves the party and starts a new one, and then some others follow him until a kind of equilibrium is attained. So we first need to know which member disagrees most with the others.

The results obtained from any of the hierarchical methods can be displayed using a dendrogram. This is the technical name given to a binary tree diagram (see Figure 2.5 for an example), where the length of the branches represents the distance at which the clusters were joined and each node represents a cluster (Everitt et al., 2001). The longer the branch is the larger the distance is between the two clusters.

Figure 2.5: An Example of a Dendrogram. The length of each branch represents the distance at which the two clusters were joined and each node represents a cluster (Everitt et al., 2001).

2.2.2 Univariate Analysis Methods

Logistic Regression

Logistic regression is a member of a class of models known as generalised linear models (McCullagh and Nelder, 1989). It is used in a wide range of research areas including medical research, biological and social sciences and in marketing applications. Logistic regression models are characterised by four assumptions which Venables and Ripley (1999) state as being:

- There is a response variable which is observed independently at fixed values of the explanatory variables.

- The explanatory variables only influence the distribution of the response variable via a single linear function which is known as the *linear predictor*.

- The distribution of the response variable has a density function of the form

$$f(y_i;\theta_i,\varphi) = \exp[A_i\{y_i\theta_i - \gamma(\theta_i)\}/\varphi + \tau(y_i,\varphi/A_i)]$$

  where φ is a scale parameter, A_i is a known prior weight and the parameter θ_i is dependent upon the *linear predictor*.

- The mean of the response variable is a smooth invertible function of the *linear predictor*. The inverse of this function is known as the *link function*.

In the case of logistic regression the response variable is assumed to have a binomial distribution and is characterised as being binary in nature, usually taking the values 1 or 0 depending on whether the event of interest has occurred or not. The most common link function for this type of model is the logit (Berkson, 1944) which represents the natural logarithm of the odds of the event of interest occurring.

The aim of logistic regression is to assess the probability of the event of interest occurring given the observed values of the explanatory variables and can be formulated in the following way:

$$E(y) = Pr(Y = 1|X_i = x_i \text{ for all } i = 1:k) = \frac{e^{\beta_0+\beta_1x_1+...+\beta_kx_k}}{1+e^{\beta_0+\beta_1x_1+...+\beta_kx_k}}$$

which with some additional algebra becomes:

$$ln\left(\frac{Pr(Y = 1|X_i = x_i \text{ for all } i = 1:k)}{1 - Pr(Y = 1|X_i = x_i \text{ for all } i = 1:k)}\right) = \beta_0 + \beta_1x_1 + ... + \beta_kx_k$$

where Y is the binary response variable and x_1 to x_k are the k explanatory variables which can be numeric or categorical variables. The log odds model is fitted by default when using the `glm()` function in the **R** package **MASS** (Venables and Ripley, 2002) with the family option set to binomial. This function will return an estimate of the natural logarithm of the odds of the event of interest occurring given the observed values of the explanatory variables, which can then be back transformed to give an estimate of the probability of the event of interest occurring given the explanatory variables.

To use logistic regression as a classifier one can round the predicted probability of the event occurring to 1 or 0 to attain the predicted class/group. Alternatively if there is prior knowledge to suggest that the model is more likely to predict a higher or lower probability then instead of using a 0.5 rounding threshold another value could be used. For example one could set the threshold to be the lowest predicted probability of those in the training set that belong to class/group 1.

Choosing an appropriate model is an important issue when constructing any regression model. Two popular model selection criteria that can be used to aid in the

construction of a suitable logistic regression model (and indeed many other types of models) are Akaikes Information Criterion (AIC) (Akaike, 1974) and the Bayesian Information Criterion (BIC) (Schwarz, 1978). Both of these criteria are known as penalised likelihood criteria as they both use the likelihood of the estimated model in their formulation. AIC is formulated as the difference between twice the number of parameters in the model (k) and two times the natural log of the maximum value of the likelihood function (L) for the estimated model (i.e., AIC $= 2 \times k - 2 \times ln(L)$). The formula for BIC is similar except that the first term is replaced by the number of parameters in the model (k) multiplied by the natural log of the number of samples used to construct the model (n). The choice of which of these two model selection criterion should be used is a highly debatable issue (Weakliem, 1999; Li and Nyholt, 2001; Burnham and Anderson, 2004; Celeux, 2005; Bouchard and Celeux, 2007). Burnham and Anderson (2004) state that this choice should be based on the philosophical context of what is assumed about the reality of the research topic, the approximation of the models to this topic and what is intended of the model-based inference. It has been stated that the main aim of AIC is to reduce the predictive error of the model (Kieseppa, 2001) and that the main aim of BIC is to increase the interpretability of the data and resulting model (Weakliem, 1999). To this end Burnham and Anderson (2004) point out that for small to moderate sample sizes BIC tends to select models that are more parsimonious than AIC.

Survival Analysis

Survival analysis is most commonly thought of as the study of time to event data, where individuals (or objects) are observed over time and what is of interest is the time at which a particular event occurs. For example, how long a patient survives after diagnosis of a life threatening disease. Survival data display two characteristics that differentiate them from other types of data, the first of these is that survival times are positive with a right skewed distribution and secondly these values are often censored. For a survival time to be censored it means that the exact time at which the event of interest occurred is unknown (Lee and Go, 1997; Oakes, 2001; Bewick et al., 2004).

If T denotes the time until the event of interest occurs for a randomly selected individual from the population of interest, then the probability that T is greater than a specific time (denoted t) is known as the survivor function $[S(t) = Pr(T > t)]$. For a given sample of survival times the Kaplan-Meier estimator (Kaplan and Meier, 1958) can be used to estimate the survival function as follows

$$\hat{S}(t) = \prod_{j:t_j \leq t} \left(\frac{n_j - d_j}{n_j} \right)$$

where n_j is the number of individuals at risk at time t_j and d_j is the number of individuals observed to fail at time t_j (i.e., the number of individuals for whom the event of interest has occurred at time t_j).

Plotting the Kaplan-Meier estimates against time gives a graphical depiction of the survival rate of the observed data over time. The plot itself is a series of horizontal steps decreasing in magnitude with small vertical lines denoting censored observations. This provides a visual method for comparing the survival functions of two or more groups. Figure 2.6 shows the Kaplan-Meier curves for the breast cancer dataset analysed by Wang et al. (2005a). For the purpose of this example the dataset was split into two groups based on the amount of estrogen receptor present, the first group consisted of those samples with over 10fmol of estrogen receptor per μg of protein (labeled as ER positive) and the second group consisted of the remaining samples (labeled as ER negative).

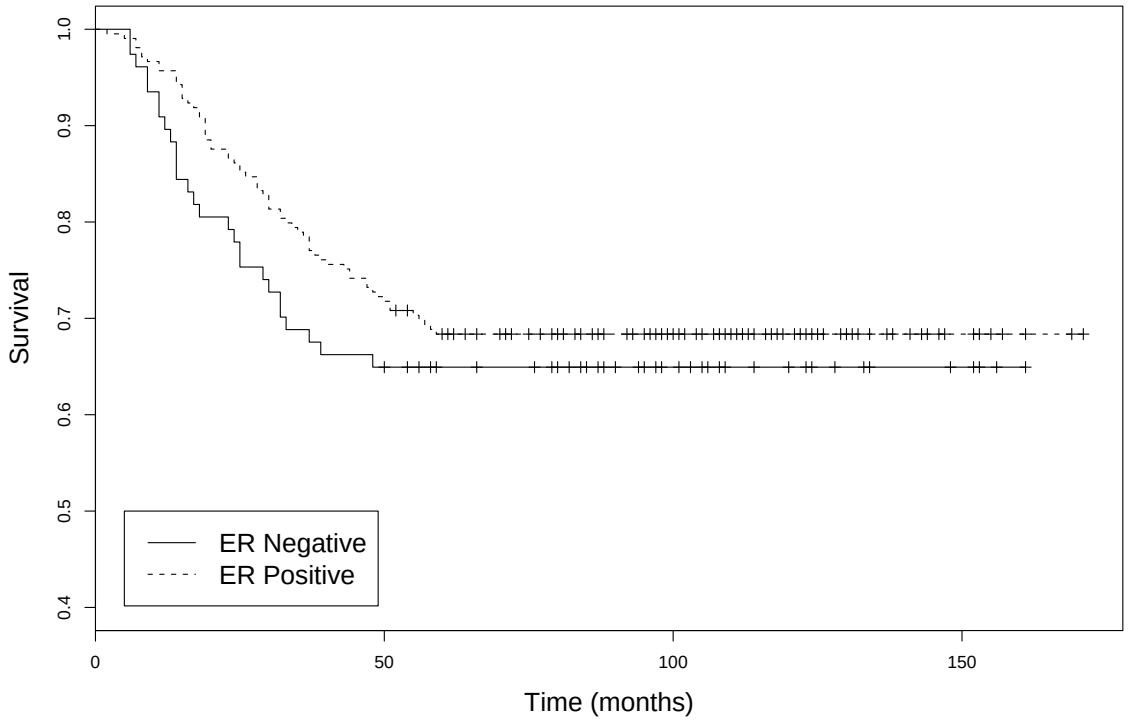

Figure 2.6: Kaplan-Meier survival curves for the Wang et al. (2005a) dataset which was split into two groups based on the estrogen receptor status.

To test if an observed difference between two Kaplan-Meier curves is statistically significant one can use the log-rank test, also known as the Mantel-Haenszel test (Mantel and Haenszel, 1959; Peto and Peto, 1972). This test is based on the Chi-square test for two way tables of counts and was specifically designed to be used with non-informative right censored survival data. The statistic is constructed by computing the difference

between the observed and expected number of failures in one of the two groups at each observed event time. The weighted sum of these differences is then calculated to obtain an overall summary across all the time points where there is an event giving the following formula

$$Z = \frac{\sum_{j=1}^{J}(O_{1j} - E_j)}{\sqrt{\sum_{j=1}^{J} V_j}}$$

where

- $j = 1, ..., J$ are the distinct times of the observed events in either group,
- O_{1j} is the observed number of events in group one at time j,
- $E_j = O_j \frac{N_{1j}}{N_j}$ where O_j is the total number of events observed at time j, N_{1j} is the number of patients "at risk" in group 1 at time j and N_j is the total number of patients "at risk" at time j, and
- $V_j = \frac{O_j(N_{1j}/N_j)(1-N_{1j}/N_j)(N_j-O_j)}{N_j-1}$

For the Kaplan-Meier curves in Figure 2.6 the p-value obtained using this test is 0.385 (calculated using the `survdiff()` function in the **R** package **survival** (Therneau and Lumley, 2007)).

Estimating Distribution Parameters

The simulation of log scaled gene expression data from two classes, via the use of the Normal distribution can require the estimation of the distribution parameters. To generate realistic simulated data it is possible to derive the distribution parameters from available log scaled expression data, where gene expression levels have been measured for a number of samples from two classes. If $X_{ij} \sim N(\mu_{1i}, \sigma^2)$ represents the log scaled expression level of gene i on array j in class 1, $Y_{ij} \sim N(\mu_{2i}, \sigma^2)$ represents the log scaled expression level of gene i on array j in the second class, and $\mu_{1i} = \mu_{2i} = \mu_i \sim N(0, \tau^2)$, then σ^2 and τ^2 can be estimated using the Log-Normal Normal model implemented in the `emfit()` function in the **R** package **EBarrays** (Kendziorski et al., 2006). The algorithm implemented in the function `emfit()` attempts to characterise the probability distribution of the observed expression measurements for each gene across all the samples via the use of a mixture model (Kendziorski et al., 2003, 2006). The Log-Normal Normal model is applicable when the observed log scaled expression measurements are assumed to have arisen from a Gaussian distribution with a gene-specific mean μ_g and a sampling variance σ^2 which Kendziorski et al. (2003) treats as being common to all genes. Kendziorski et al. (2003) define μ_g to be normally distributed with an underlying mean μ_0 and variance τ^2. The parameters σ^2, μ_0 and τ^2

are estimated by numerically optimising the marginal log-likelihood for the Log-Normal Normal model (Kendziorski et al., 2003).

Alternatively, if $\mu_{1i} = \pi_0 N(\mu_0, \tau_0^2) + (1 - \pi_0)N(\mu_1, \tau_1^2)$ then to estimate π_0, μ_0, τ_0^2, μ_1, and τ_1^2 a 2-component mixture model could be fitted to the observed gene-specific means from one of the two classes in the available log scaled gene expression dataset. The **R** package **mclust** (Fraley and Raftery, 2006) contains a function, `Mclust()`, that uses a model based clustering approach to derive a mixture model consisting of G components for the inputted data (Fraley and Raftery, 2002). This model can then be optimised using the Expectation-Maximisation (EM) algorithm (Dempster et al., 1977), which is implemented in the `em()` and `me()` functions within this package (Fraley and Raftery, 2002, 2006). The EM algorithm is generally used to iteratively compute the maximum likelihood estimates of the parameters in a probabilistic model which depends on latent variables (Dempster et al., 1977). The algorithm alternates between two steps the expectation (E) step and the maximisation (M) step. The E step computes an expectation of the likelihood by including the latent variables as if they were observed (Dempster et al., 1977). The M step, computes the maximum likelihood estimates of the parameters by maximising the expected likelihood found in the E step. The estimates for the parameters found in the M step are then feed back into the E step and the process is repeated (Dempster et al., 1977). The `em()` function starts with the expectation step whereas the `me()` function starts with the maximisation step. The number of components in the mixture model can either be inputted as a fixed value (e.g., G=2) or it can be specified as a range of values (e.g., G=1:9). When the latter is true the algorithm uses the Bayesian Information Criterion or BIC to choose the mixture model with the optimal number of components (Fraley and Raftery, 2002, 2006). In the case where a 2-component mixture model consisting of univariate normal distributions is to be derived for a dataset, as in the example above, the `Mclust()` and `meV()` functions can be used to generate optimal estimates for π_0, μ_0, τ_0^2, μ_1, and τ_1^2 from the gene-specific means from one of the classes in the observed log scaled expression data.

2.2.3 Machine Learning Methods

Support Vector Machines

Support Vector Machines (SVM) (Vapnik 1995;1998) are learning systems that use a hypothesis space of linear functions in a high dimensional feature space, trained with a learning algorithm from optimisation theory that implements a learning bias derived from statistical learning theory (Cristianini and Shawe-Taylor, 2000).

SVM's are based on the ideas of *learning methodology*, which is the approach of using examples to manufacture a learning program. For example, in the case of classification "Supervised Learning" is when both the inputs and their associated outputs (known as the training set) are used to develop a set of classification rules, which are then tested on an independent set of inputs and their outputs (known as the test set).

Support Vector Machines can be used for both classification problems and regression problems. Cristianini and Shawe-Taylor (2000) give a detailed description of how SVM's can be applied in both of these two problems. For the classification problem they describe the aim to be that of constructing a "computationally efficient way of learning 'good' separating hyperplanes in a high dimensional feature space" which will optimise a given set of "generalisation bounds" and be "able to deal with sample sizes of the order 100,000". Cristianini and Shawe-Taylor (2000) also have a chapter dedicated to applications of SVM's in which they discuss, among other areas of application, the use of SVM's for the "automatic categorisation of gene expression data from DNA microarrays".

The "maximal margin classifier" is the original (and perhaps most simple) SVM model. This method works by finding a linear hyperplane that can separate data belonging to two different classes whilst maximising the distance of the training samples to the hyperplane, that is, maximising the margin (Varma and Simon, 2006). Figure 2.7 provides a graphical representation of the maximal margin SVM classifier for discriminating between two classes. The margin is the distance between the linear hyperplane, shown as a solid line, and the support vectors which are determined by the data points nearest to the hyperplane, represented by the dashed lines. Its major drawback for practical use is that it only works for data that can be separated linearly in the feature space. However, by mapping the original data to a higher dimension via a kernel function and formulating the optimisation problem in a way that penalises misclassifications, it is possible to analyse data that cannot be separated linearly in the feature space (Hastie et al. (2001), Statnikov et al. (2005)).

Neural Networks

Hastie et al. (2001) describe Neural Networks (McCulloch and Pitts (1943); Widrow

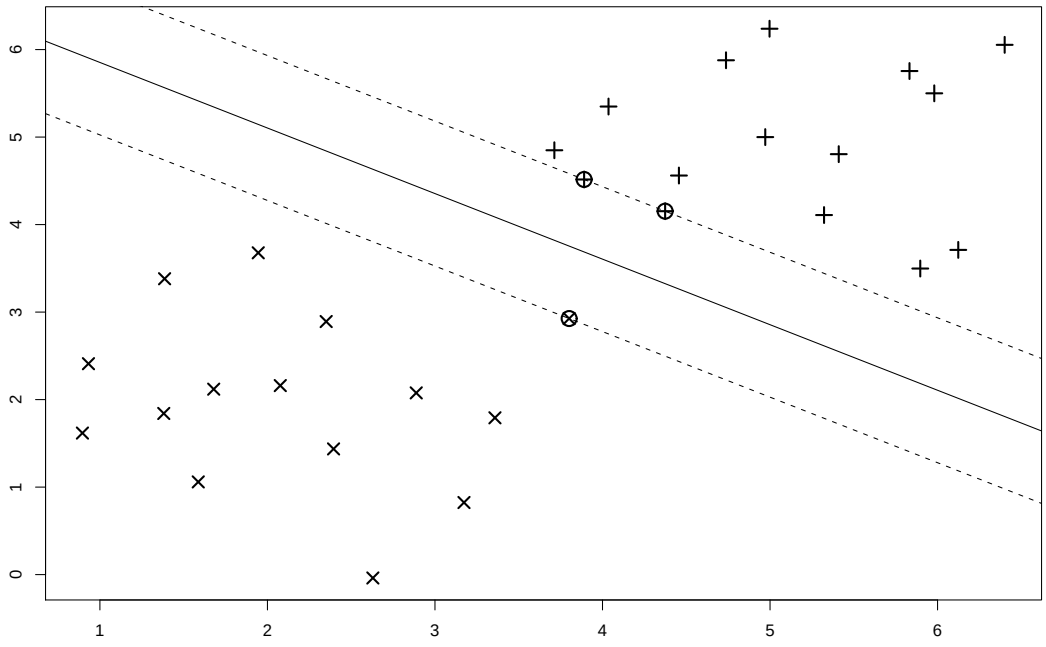

Figure 2.7: Example of the maximal margin SVM classifier. The margin is the distance between the linear hyperplane (the solid line) and the support vectors, indicated by the dashed lines.

and Hoff (1960)) as two-stage regression or classification models which are usually represented by a *network diagram*. An example can be seen in Figure 2.8, which depicts a feed-forward Neural Network with only a single hidden layer. The $\mathbf{X}_i$'s $(i = 1 : p)$ form what is known as the input layer, the $\mathbf{Z}_m$'s $(m = 1 : M)$ the hidden layer and the $\mathbf{Y}_k$'s $(k = 1 : K)$ the output layer. As the lines in the figure suggest and Hastie et al. (2001) explain, the $\mathbf{Y}_k$'s are a linear combination of the $\mathbf{Z}_m$'s, each of which is in turn a linear combination of the $\mathbf{X}_i$'s. This linear combination of the $\mathbf{X}_i$'s is known as the *activation function*. Both Venables and Ripley (1999) and Hastie et al. (2001) state that the activation function is usually taken to be the logistic function in either $e^v/(1 + e^v)$ or $1/(1 + e^{-v})$ form.

Venables and Ripley (1999) give the following formula for the one hidden layer, feed-forward neural network:

$$y_k = \phi_o\Big(\alpha_k + \sum_h w_{mk}\phi_m\Big(\alpha_m + \sum_i w_{im}x_i\Big)\Big)$$

Where ϕ_m is the activation function, α_k and α_m are known as bias terms and ϕ_o is the output function which according to (Hastie et al., 2001) "allows a final transformation of the vector of outputs". From the description given above it is clear that Neural Networks can be applied to both regression and classification. In the classification case,

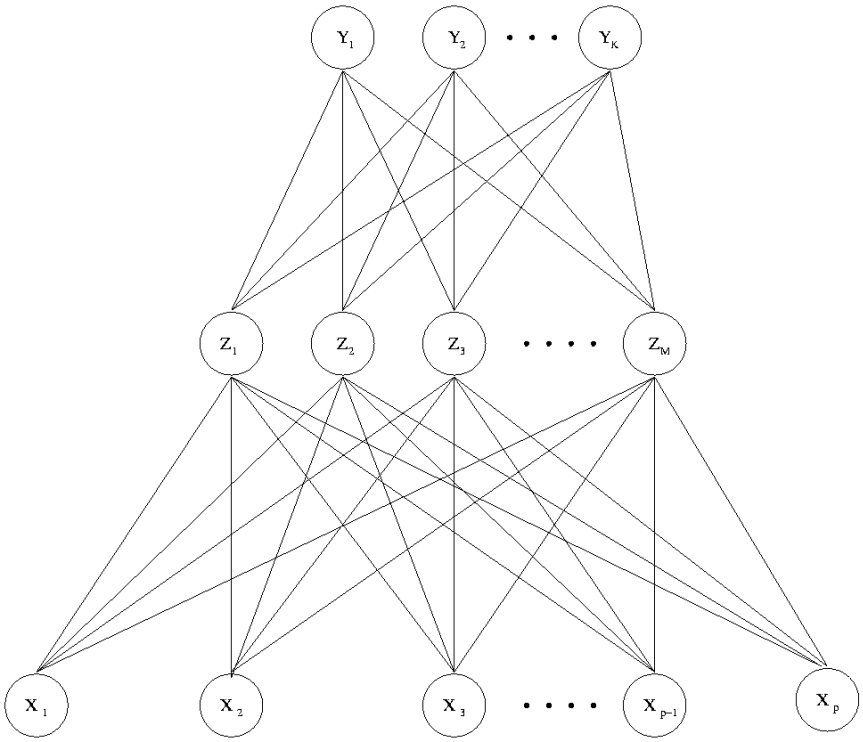

Figure 2.8: A Single Hidden Layer, Feed-Forward Neural Network (Hastie et al., 2001). The middle row is know as the hidden layer which forms a link between the input layer (bottom row) and the output layer (top row). This link is known as the *activation function*.

where there are K-classes and each output $\mathbf{Y}_k$ $(k = 1 : K)$ is coded as a 0-1 variable, the kth output is modeling the probability of class k, as stated by Hastie et al. (2001). As a word of warning Hastie et al. (2001) indicate that Neural Networks are not as effective when one is interested in describing the "physical process that generated the data and the roles of individual inputs".

2.2.4 Resampling-Based Methods

Permutation Tests

Permutation Tests, also known as Randomisation Tests (Fisher, 1935; Pitman, 1937a,b, 1938) are statistical tests in which the data are repeatedly divided between treatments. A test statistic (e.g., t or F) is computed for each data division, and the proportion of the data divisions with a test statistic value at least as large as the value obtained from the initial experiment determines the significance of the initial results (Edgington, 1980). Edgington (1980) also emphasized that Permutation Tests are really just a different way of calculating p-values for all kinds of test statistics, when it is not possible to use "conventional significance tables" or parametric methods. Good (2000) commented that Permutation Tests are one of the most powerful statistical methods as they are not only flexible and robust when there is missing data or test assumptions are violated but

that they also give the analyst the ability to choose the most appropriate test statistic for the analysis being performed.

Permutation Tests rely on two assumptions; (1) that the underlying distribution of the test statistic is symmetric and (2) that the observations are exchangeable with respect to groups. Pesarin (2001) states that if the second assumption is not satisfied "it may be useful to employ bootstrap techniques, which are less demanding in terms of assumptions".

The Jackknife (or Leave-one-out) Method

The idea behind the Jackknife was first proposed by Quenouille (1949) for the estimation of the bias of an estimate. Tukey (1958) named the procedure the "Jackknife" when he realised its potential for estimating the standard error of an estimate. The Jackknife therefore is a technique for estimating the bias and standard error of an estimate. It does this by the process of leave-one-out. What this means is that for a given sample and estimator, the estimates of the bias and standard error of the estimator are calculated using all the subsets obtained by leaving one of the sample observations out at each turn. Efron and Tibshirani (1993, page 141) give the following definition of the Jackknife:

> Suppose we have a sample $\mathbf{x} = (x_1,x_2,...x_n)$ and an estimator $\hat{\theta} = s(\mathbf{x})$. We wish to estimate the bias and standard error of $\hat{\theta}$. The jackknife focuses on the samples that *leave out one observation at a time*:
>
> $$\mathbf{x}_{(i)} = (x_1, x_2, ...x_{i-1}, x - i + 1, ...x_n)$$
>
> for $i = 1, 2, ...n$, called *jackknife samples*. The ith jackknife sample consists of the data set with the ith observation removed. Let
>
> $$\hat{\theta}_{(i)} = s(\mathbf{x}_{(i)})$$
>
> be the ith *jackknife replication* of $\hat{\theta}$.
>
> The jackknife estimate of bias is defined by
>
> $$\hat{\text{bias}}_{\text{jack}} = (n-1)(\hat{\theta}_{(.)} - \hat{\theta})$$
>
> where
>
> $$\hat{\theta}_{(.)} = \sum_{i=1}^{n} \hat{\theta}_{(i)}/n.$$

The jackknife estimate of standard error is defined by

$$\hat{\text{se}}_{\text{jack}} = [\frac{n-1}{n}\sum(\hat{\theta}_{(i)} - \hat{\theta}_{(.)})^2]^{1/2}.$$

As Efron and Tibshirani (1993) point out, however, the jackknife can "fail miserably" when the statistic $\hat{\theta}$ is not smooth. What they meant by smooth is that if a small change is made to the data this causes the statistic to change by only a small amount. An example of a non-smooth statistic is the median.

The Bootstrap

First introduced by Efron (1979) the bootstrap is an extension of the Jackknife method (Tukey, 1958) for estimating the bias and standard error of an estimate. The bootstrap method works on the basis of re-sampling (with replacement) the observations, to form a 'bootstrap sample' which is of the same size, n, as the original sample. This re-sampling can be done over the entire sample set or within subgroups. Efron and Tibshirani (1993) provide the following algorithm for estimating the bootstrap standard error of an estimate:

1. Select B independent bootstrap samples $\mathbf{x}^{*1}$, $\mathbf{x}^{*2}$,...,$\mathbf{x}^{*B}$, each consisting of n data values drawn with replacement from $\mathbf{x}$.
2. Evaluate the *bootstrap replication* corresponding to each bootstrap sample,
$$\hat{\theta}^*(b) = s(\mathbf{x}^{*b}) \qquad b = 1, 2, ..., B$$
3. Estimate the standard error $\text{se}_F(\hat{\theta})$ by the sample standard deviation of the B replications

$$\hat{\text{se}}_B = \{\sum_{b=1}^{B}[\hat{\theta}^*(b) - \hat{\theta}^*(.)]^2/(B-1)\}^{1/2},$$

$$\text{where} \qquad \hat{\theta}^*(.) = \sum_{b=1}^{B}\hat{\theta}^*(b)/B.$$

Bootstrap replication refers to the estimate for the chosen parameter of interest (for example the mean or any test statistic) evaluated on the bootstrap sample. Efron and Tibshirani (1993) recommend the number of different bootstrap samples used (B) should be in the range of 25 - 200 for estimating a standard error.

The bootstrap method can also be used to estimate the bias of an estimate. As Efron and Tibshirani (1993) point out, the bias can be calculated using the same set

of bootstrap samples as was used in the calculation of the bootstrap standard error. In step three of Efron and Tibshirani (1993) algorithm for estimating the bootstrap standard error, they simply replace the formula for $\hat{\text{se}}_B$ with the following formula for the bias.

$$\text{Bias}_{Boot} = \hat{\theta}^*(.) - \hat{\theta}$$

where $\hat{\theta}^*(.) = \sum_{b=1}^{B} \hat{\theta}^*(b)/B$ is the average apparent error rate for the bootstrap samples and $\hat{\theta}$ is the observed apparent error rate.

The average apparent error rate for the bootstrap samples ($\hat{\theta}^*(.)$) is typically based on approximately 63.2% of the dataset (i.e., $0.632 \times n$, where n is the sample size) which, as Ambroise and McLachlan (2002) point out, implies that this average is upwardly biased. To counteract this bias Efron (1983) proposed the $B.632$ estimator which weights both the observed apparent error rate and the average apparent error rate for the bootstrap samples so that the unbiased estimate of the true apparent error is:

$$B.632 = 0.632 \times \hat{\theta}^*(.) + 0.368 \times \hat{\theta}$$

When a classifier over-fits the data, due either to the nature of the classifier or as a direct consequence of the classifier being built on a dataset with a relatively larger number of variables than samples (eg., microarray data), Ambroise and McLachlan (2002) recommend using the 0.632^+ estimator (Efron and Tibshirani, 1997) to obtain an unbiased estimate of the true apparent error rate. Efron and Tibshirani (1997) defined the 0.632^+ estimate as:

$$B.632^+ = \omega \times \hat{\theta}^*(.) + (1 - \omega) \times \hat{\theta}$$

where the weight ω is calculated based on the 'no-information error rate' one would obtain if there truly were no difference between the two classes.

2.3 Bioinformatic Databases

Databases are repositories of information, generally stored on computers. They contain an organised series of records that can be searched through and displayed on the computer screen, downloaded or emailed to a specified address. There are a large number of databases dedicated to bioinformatic data and information, covering topics such as gene function and sequence, as well as analysis methods and software. The databases that are described in this section provide researchers with access to publicly available gene expression data, tools for the annotation, analysis and visualisation of such data, and links to supporting literature. The following terms are often encountered when

dealing with bioinformatic databases.

- *Accession Number:* From the glossary of the National Centre for Biotechnology Information online handbook, an accession number is defined as "a unique identifier given to a sequence when it is submitted to one of the DNA repositories" (NCBI, 2002). An accession number is a combination of letters and numbers, for example D15979 (Sasaki et al., 1994). The Oxford English Dictionary defines annotation to be "a note added to anything written, by way of explanation or comment" (Simpson, 2005). The annotation of a gene sequence starts with its accession number and is built up as information becomes available, for example information on the gene's function.

- *Gene ontologies* (GO) are sets of defined terms that are networked and restricted to those used in the field of biology (Ashbuner et al., 2000). GO terms can be used as "Keywords" when conducting database searches across several databases, so that the search results will be consistent. GO terms are organised over three principal areas; molecular function, biological process and cellular component. For a gene to be assigned to a particular GO term the RNA or protein encoded by that gene must exhibit an attribute in at least one of these three areas relating to that term.

Examples of Bioinformatic Databases

www.ncbi.nlm.nih.gov

The National Centre for Biotechnology Information.

This is a division of the National Library of Medicine which provides integrated access to all the publicly available genetic sequence information and associated annotation. It also provides citations and abstracts from published literature referenced by the genetic sequence records. One of the NCBI's offspring databases in collaboration with the National Library of Medicine (NLM), is PubMed (www.ncbi.nlm.nih.gov/pubmed). This database includes all the MEDLINE® records with some additional records and where possible links to electronic full-text articles are also included.

Another of the NCBI's offspring databases is Entrez Gene, formerly known as Locus Link. Entrez Gene focuses on three areas: genomes that have been completely sequenced, those that have a active research community that can contribute gene-specific information to the database, and those that are scheduled for intense sequence analysis in the near future (Maglott et al., 2005). Entrez Gene aims to provide traceable unique species specific identifiers (GeneID) for genes and to give information associated with

those identifiers for the unrestricted use of the public (Maglott et al., 2005). The information that Entrez maintains includes chromosomal localisation, gene products and their attributes (for example protein interactions), associated markers, phenotypes, interactions and a large number of links to citations, sequences, variation details and external databases (Maglott et al., 2005). The most direct way to search through this information is to submit a query directly from the NCBI home page, as can be done with PubMed.

http://www.geneontology.org/

Gene Ontology Consortium

The Gene Ontology Consortium is a collaborative project that was developed to "address the need for consistent descriptions of gene products in different databases (gene product is the RNA or protein encoded by a given gene). This was achieved by the ongoing development of three ontologies that describe gene products in terms of their associated biological processes, cellular components and molecular functions in a species-independent manner" (Ashbuner et al., 2000). This database allows the user to search by GO term, gene name or protein name. If the user searches by gene or protein name, the results are the Gene Product, Data source, Associated terms, Aspect (i.e., whether it is a biological process, molecular function or a cellular component) and links to the database UniProt (Apweiler et al., 2004) for details on proteins. If a user searches by GO term, the results are the GO term, Aspect, definition, links to some more details and a graphical view of the information. The Gene Ontology Consortium uses both manual and automated methods to assign genes to GO Terms based on two principles. The first of these is that every assignment be attributable to a source (eg., a literature reference, or computational analysis, or another database). The second principle is that the type of supporting evidence found in the source for the assignment must be included in the annotation. They also stress that assignments should reflect the normal function, process or localisation component of the gene and that in the event of uncertainty in the assignment of a gene, that the gene be associated with two GO Terms, where one term could be a parent term of the other. To ensure that the ontologies provided are kept up to date, they are continuously updated, with new versions made available on a monthly basis.

http://www.genome.jp/kegg/

KEGG

The Kyoto Encyclopedia of Genes and Genomes (KEGG)(Kanehisa (1997); Kanehisa and Goto (2000)) has developed from research projects at the Kanehisa Laboratory

of Kyoto University Bioinformatic Center. KEGG is split into three groups, Generalised KEGG, Specialised KEGG and Personalised KEGG. Generalised KEGG consists of four databases; KEGG PATHWAY (for pathway information), KEGG GENES (for genomic information), KEGG LIGAND (for chemical information) and KEGG BRITE (which contains the KEGG Orthology). Specialised KEGG is broken down into three arena, KEGG for specific organisms, KEGG for selected research areas (covering microarray data analysis and glycome informatics) and KEGG for software development (including XML representation of KEGG pathways). Personalised KEGG allows the user to download a stand-alone program for microarray data analysis (KegArray) and a stand-alone program for drawing compound/glycan structures.

http://david.niaid.nih.gov/david

DAVID (Database for Annotation, Visualisation and Integrated Discovery)

DAVID is a web-based tool that gives users access to functional annotation databases. These annotations are generally derived from Entrez Gene at the National Centre for Biotechnology Information. DAVID uses the Entrez accession numbers to link gene accession systems such as Genbank and Unigene (see below for details). It also uses Affymetrix probe identifiers to link to biological annotations including gene names. DAVID has five tools available for use; Annotation Tool, GO Charts, KeggCharts, Domain Charts and EASEonline.

The Annotation Tool is an automated method for the functional annotation of genome-scale datasets. On submitting a list of gene accession numbers, this annotation tool will produce by default, where available, the official gene symbol, name and functional summary as included in the Entrez Gene (Locus Link) report given by NCBI, it will also report the unique and stable numeric identifier for the curated genetic loci. There are six other options available in the annotation tool, these are the Genbank, OMIM, Refseq, Unigene, Gene Ontology and Affy Description. The first of these gives the accession number corresponding to the nucleotide sequence represented by the given probe set. The second gives links to textual information, references, MEDLINE, sequence records in the Entrez Gene system and to additional related resources at NCBI and its collaborators. Refseq provides reference sequence standards for the naturally occurring molecules of the Central Dogma. Unigene is an experimental system that automatically partitions GenBank sequences into a set of non-redundant gene-oriented clusters, where each cluster contains sequences that represent a unique gene. Gene Ontology gives links to the gene ontology consortium (Ashbuner et al., 2000), where a dynamic controlled vocabulary can be applied to the functions of genes and proteins. Lastly the Affy Description is the probe set description provided by Affymetrix, relating to the particular probes used on their GeneChip microarray system.

The GO Charts tool produces gene ontology charts for a given list of genes. There are three types of available classifications; Biological Process, Molecular Function and the Cellular Component. Each type of classification has seven levels determining the coverage and specificity sort. The user can choose the minimum number of "hits threshold", any whole number between 1 and 10, whether the results should be ordered alphabetically or by the number of hits and whether the results should be displayed in a new window or in the existing one.

The KeggCharts tool is a visualisation tool that graphically displays the distribution of differentially expressed genes among metabolic pathways. The Domain Charts tool does a similar thing but among functional protein domains instead of metabolic pathways.

The EASEonline tool provides statistical methods for discovering enriched biological themes within gene lists. This tool is only for use on Windows operating systems. The tool performs three basic functions for any given list of genes: annotation, customisable linking to online tools, and biological "theme" finding using gene category over-representation analysis. It includes a function to automatically download and pass annotation from public sources into local files so that gene lists can be analysed without having to transmit them over the Internet, and thus impinge on data confidentiality.

http://www.ncbi.nlm.nih.gov/geo

GEO (Gene Expression Omnibus)

GEO is a public repository which archives and distributes high throughput data, such as microarray data and molecular abundance data which has been submitted by the scientific community (Edgar et al., 2002). Currently GEO stores over three billion individual measurements which have been derived from over 200 organisms in the investigation of a wide range of biological issues and phenomena (Barrett et al., 2007). GEO also provides a collection of user-friendly web-based tools which can be used to query, visualise, explore and download the experiments and expression data it stores.

Within GEO there are five basic record types; *platform*, *sample*, *series*, *dataset* and *profile*. The first three of these are supplied by the submitter and form a detailed description of the experiment as a whole, including; the experimental process, a description of the platform used in the experiment, any extra handling or processing that the data was subjected to, any additional information pertaining to the analysis (e.g., clinical outcomes, and a link to published literature associated with the experiment), and the contact details of the submitter. Each submitted dataset is assigned a unique GEO accession number (e.g., GSE4922) which can be submitted into one of the search

algorithms within GEO to retrieve the *series* record for the dataset. This record provides details on the experiment and contributor with links to the published literature associated with the dataset, versions of the expression data and any clinical and supplementary information that can be downloaded. Similarly the platform used in the experiment and each individual sample in the experiment are also assigned a unique ID number, which can be used to retrieve the record associated with a given platform or sample.

The latter two record types (*dataset* and *profile*) are assembled by the curators of the database. The *dataset* record consists of a curated collection of samples that are biologically and statistically comparable, that is, their associated measurements (e.g., expression values) were obtained using the same platform (e.g., Affymetrix hgu133a) and can be assumed to have been calculated in an equivalent manner (i.e., any background processing and/or normalisation are consistent). These records, each with its own unique ID number, provide a pool of data that can be analysed using the suite of data visualisation and analysis tools provided within GEO. These tools allow the user to visualise a set of samples via a heatmap with dendrograms derived from a hierarchical or k-means cluster analysis of the measurements and/or samples. They also allow the user to perform basic comparative analyses (using a t-test) between two groups of samples to find genes that are differentially expressed. The *profile* record is derived from the *dataset* record and consists of all the expression values or measurements for an individual gene across all of the samples in a given *dataset.*

In addition to searching the database using a unique ID or accession number, users are able to browse through a list of the contents currently held by GEO broken down by record type, search by gene symbol or keyword, or by using the link to the NCBI's basic local alignment search tool (BLAST).

2.4 Software for the Analysis of Bioinformatic Data

www.r-project.org

R

Developed by Ihaka and Gentleman (1996) R is an open-source implementation of the S language, which was developed at Bell Labs by John Chambers and colleagues (Becker et al., 1988), R is an environment within which statistical techniques and graphics can be implemented. It is a comprehensive computing language which allows the user to build functions, written in the R dialect of S, C, C++ or Fortran, or to extend the environment with the addition of "packages" or libraries that contain source code for analytic and graphical functions (e.g., **survival** (Therneau et al., 1990) which

contains source code for the analysis of survival data). These "packages" can be found through the Comprehensive R Archive Network (http://cran.stat.auckland.ac.nz) and many other Internet sites that cover a vast range of modern statistical techniques (e.g., Bioconductor, see below). R compiles and runs on a wide range of platforms including Unix, Linux, Windows and MacOS, and is available to download freely under the terms of the Free Software Foundation's GNU General Public License from the above website.

www.bioconductor.org

Bioconductor

Bioconductor (Gentleman et al., 2004) is an open source and development software project that provides tools for the analysis and understanding of genomic data. These tools can be run in R (Ihaka and Gentleman, 1996) and there are also some plug-ins available for Excel. This project emerged from the Biostatistics Unit of the Dana Farber Cancer Institute at the Harvard Medical School and the Harvard School of Public Health, in the early part of 2001. Its broad objectives, taken from Gentleman et al. (2004), are to;

- provide access to a wide range of powerful statistical and graphical methods for the analysis of genomic data
- facilitate the integration of biological meta-data in the analysis of experimental data, such as literature data from PubMed and annotation data from LocusLink
- allow the rapid development of extensible, scalable and interoperable software
- promote high-quality documentation and reproducible research and provide training in computational and statistical methods for the analysis of genomic data.

Bioconductor has a large number of packages that can be used for spotted microarray data. From pre-processing packages that can help with normalisation, as described in Chapter 1, and gene selection right through to packages that can be used for annotation building and packages for interacting with Gene Ontology. Bioconductor also has a selection of experimental and meta data looking at leukemia, ovarian, breast and prostate cancer.

Downloading Bioconductor can be achieved through the command line in R using one of two functions. The first, `biocLite()`, downloads and by default installs a small subset of the available packages on the bioconductor website, Table 2.2 gives an overview of this subset. Descriptions are quoted from the Release 2.0 Package Index and the Developmental Package Index pages found on the Bioconductor web-page given above. The function `biocLite()` also allows for the downloading and installation of specified

bioconductor and CRAN packages. The other function, `getBioC()`, works in a similar manner as `biocLite()` but with a much larger list of default packages. A full list of the packages that have been released and that are available to be downloaded can be found at:

www.bioconductor.org/packages/release/BiocViews.html

Table 2.2: Default Subset of packages installed when using the `biocLite()` function, with a brief description quoted from the Release 2.0 Package Index and the Developmental Package Index pages found on the Bioconductor web-page

Package Name	Description
affy	Methods for Affymetrix Oligonucleotide Arrays.
affydata	Affymetrix Data for Demonstration Purpose.
affyPLM	Methods for fitting probe-level models.
annaffy	Annotation tools for Affymetrix biological metadata.
annotate	Associate experimental data in real time to biological meta data from web databases such as GenBank, LocusLink and PubMed. Process and store query results. Generate HTML reports of analyses.
Biobase	Base functions for Bioconductor. These Functions are needed by many for the other packages.
Biostrings	String objects representing biological sequences, and matching algorithms.
DynDoc	Dynamic document tools.
gcrma	Background Adjustment Using Sequence Information.
genefilter	Tools for sequentially filtering genes using a wide variety of filtering functions. Examples of filters include: number of missing values, ANOVA p-value and Cox model p-values.
geneplotter	Contains functions for plotting genetic data.
hgu95av2	Affymetrix Human Genome U95 Set Annotation Data (hgu95av2).
limma	Linear models for microarray data.
marray	Includes functions for data input, diagnostic plots, normalisation and quality checking for two-colour spotted microarray data.
matchprobes	Tools for sequence matching of probes on arrays.
multtest	Multiple testing procedures for controlling the family-wise error rate and the false discovery rate (FDR). Tests are based on t- or F-statistics for 1 and 2 factor designs. Permutation procedures are available to estimate adjusted p-values.
ROC	Utilities for ROC, with microarray focus.
vsn	Normalisation and variance stabilizing transformations for both Affymetrix and cDNA array data.
xtable	Export tables to LaTeX or HTML.

3

Gene Selection for Class Prediction

3.1 Motivation

It is well known that the particular genes that are used in a predictive analysis can have a major impact on the ability to determine the likelihood of a particular clinical outcome of interest, for example, relapse in cancer tumours (Ein-Dor et al., 2005). A wide range of statistical and non-statistical methods have been used to facilitate gene selection, from test statistics (such as Student's t-test (Cui and Churchill, 2003) and the Mann-Whitney test (Chen et al., 2007)) to more straightforward methods, such as the use of fold-change cut offs (Chu et al., 1998). In this chapter a method for gene selection is proposed which incorporates a pre-filter (based on fold-change magnitude) with standard statistical ranking methods, for example the t-test statistic or the SAM adjusted t-statistic. The main goal for developing this pre-filter is to provide a method for removing genes that could not be used in a diagnostic test. One example of such a situation is a test that relies on polymerase chain reaction (PCR), and thus requires genes to have undergone a biologically relevant level of differential expression in order for it to succeed. The proposed pre-filter makes the assumption that the expression data being analysed arises from dual channel spotted microarray experiments where each sample is compared to a reference sample. The effect of this pre-filter on the ability to determine the likelihood of a particular clinical outcome of interest (eg., tumor relapse) is compared with the effect obtained from standard approaches to gene selection. This methodology is illustrated using simulated data and a melanoma dataset provided by Pacific Edge Biotechnology Ltd.

A typical microarray classification study usually has the goal of predicting class

membership for a given binary clinical outcome such as tumour relapse. This goal can be based on gene expression data from a dual channel spotted microarray experiment, where each sample from both classes is compared to a reference sample. To predict class membership a classification algorithm, which includes a variable selection component is usually employed (Cho and Won, 2003; Jirapech-Umpai and Aitken, 2005; Hu et al., 2006). This type of classification algorithm usually starts by selecting genes that show a high level of discriminatory ability between the two classes, which can be achieved by using a ranking method such as Student's t-test (Cui and Churchill, 2003) and the Mann-Whitney test (Chen et al., 2007). These highly discriminatory genes are then used to construct a classifier, the performance of which is then validated on an independent dataset.

3.2 Issues in Gene Selection for Classifying Microarray Data

With the large number of gene selection methods available, one major issue is which selection method is the most appropriate for the microarray data that is to be analysed and the classification method that is to be used. Jirapech-Umpai and Aitken (2005) stated, in their comparison of six of the eight gene selection methods available in the RankGene software produced by Su et al. (2003), that the choice of gene selection method can have a significant effect on the predictive performance of the classification method employed. Hu et al. (2006) similarly observed that, of the four gene selection methods they compared (using two classification methods which contain an internal gene selection procedure) the combination of a given gene selection method with one of the classifiers increased the predictive performance of the classifier, whereas when the same gene selection method was used in combination with the other classifier the predictive performance of that classifier decreased when analysing the same dataset. Chen et al. (2007) pointed out that different gene selection methods can give different rankings even for the same objective, such as the development of a classifier for class prediction or the identification of differentially expressed genes for a follow-up study.

Once a gene selection method has been chosen it is important that it be implemented in such a way as not to introduce any bias into the analysis. Ambroise and McLachlan (2002) highlighted the issue of bias in microarray variable selection. In their article, they demonstrated how bias could be introduced into a classification analysis if the selection of genes to be used to build a classification/prediction model is performed using the entire dataset. The main point that Ambroise and McLachlan (2002) made was that gene selection should occur in the training stage of the classification/prediction process so as to avoid bias at the validation stage of the analysis.

When the chosen gene selection method has been implemented one of the next issues is how many of the selected genes should be included in the classifier. Hu et al. (2006) noted that if the number of genes has been heavily reduced by the gene selection method then the performance of the classifier may also be reduced as genes that may be relevant from a classification point-of-view may have been removed in the gene selection process. They also pointed out that classifiers built on a smaller number of selected genes do not necessarily have a higher predictive performance in comparison with classifiers built on a larger number of selected genes. This is because genes that are deemed to be less informative by the gene selection technique can improve the predictive performance of a classifier if they are related to genes that are deemed to be highly informative by the gene selection method. On the other hand, having too many genes in the classifier can lead to over-fitting and a decrease in the predictive performance of the classifier on independent test/validation data (Ransohoff, 2004). This implies that a balance must be struck between too few selected genes and too many selected genes. As stated by Wang et al. (2005b) typically the top 50-100 highly informative genes are selected as variables in the classifier. The list of genes used in a classifier is known as a *gene-list* or *gene signature*.

The first of two further issues that one must be aware of is that when a gene-list has been compiled it may not be the only optimally performing signature, as it is possible to obtain a different gene-list that performs just as well as the original list by simply using a different subset of the samples for gene selection (Ein-Dor et al., 2005). Secondly, there may also be an element of redundancy in the gene-list due to some of the genes being highly correlated with each other. A large amount of research has been conducted on gene selection methods that produce lists of non-redundant marker genes that are highly relevant in classifying microarray data. Some examples of these methods were proposed by Hanczar et al. (2003), Jaeger et al. (2003) and Wang et al. (2005b). The method proposed by Wang et al. (2005b) combines one of three gene selection methods, that are based on ranking genes via a given statistic, with hierarchical clustering on the ranked genes and collapsing the resulting clusters to form marker genes.

3.3 Gene Selection Techniques

A large number of gene selection techniques for the detection of differentially expressed genes have been developed specifically for microarrays. This section describes some of these methods, focusing on gene selection methods that rank genes via the use of a given statistic, and which could be used as variable selection methods in the process of predictor construction. When using gene selection techniques to identify significantly differentially expressed genes in large datasets it is important to correct for multiple

comparisons.

Fold-Change (FC) Cut Offs

When working with dual channel spotted microarray expression data the fold-change of a gene, as described in Chapter 1, is generally the ratio of the activity of a given gene under two different conditions. For example, the ratio of the activity of a gene in two different phenotypes. To use fold-changes in a gene selection method one would keep genes that satisfy the following condition,

$$|\frac{1}{n}\sum_{j=1}^{n} log_2(\mathrm{FC}_{ij})| > \mathrm{c}$$

where, FC_{ij} is the fold-change for the ith gene in the jth array, $j = 1, ..., n$ and $i = 1, ..., p$ and c is a pre-determined cut off value (generally set to 1).

The first of two examples of using fold-change cut-offs as a gene selection method comes from a study exploring changes in gene expression during sporulation (which is the formation of spores for reproduction) in budding yeast. Chu et al. (1998) used a criterion that is equivalent to a 3-fold change cut off for a single time point, or an average 2.2-fold change cut off across the entire time course, to determine whether a gene had undergone a significant change in mRNA levels during sporulation. They reported that of the approximately 6200 known protein-encoding genes in the yeast genome at that time, this criterion deemed over 1000 of these genes to be changing significantly during sporulation.

The second example comes from Cheadle et al. (2003) who compared the use of fold changes (calculated from globally normalised data) as a method for predicting significant changes in gene expression with z ratios and z and t statistical tests using z score transformed data. When the significance threshold of the fold changes was set at $\geqslant 2$ they observed that the number of genes that were predicted as undergoing significant differential expression was substantially smaller than the numbers predicted by the other methods.

Allison et al. (2006) stress that even though fold change cut-off was the first method used to determine if genes were differentially expressed, it is not an appropriate test statistic for "true" inference. They give two reasons for this statement, the first is based on the fact that using this gene selection method does not account for the variability in the data, which is a highly important issue in predictor construction, and secondly because this gene selection method does not give any level of "confidence" associated with its results, such as a p-value or confidence interval.

Mann-Whitney Test

The Mann-Whitney Test (Wilcoxon (1945), Mann and Whitney (1947)) is a non-parametric significance test which tests for the equality of two population medians. It works by ranking all of the data values and finding the sum of the rankings for the class with the smaller sample size (T_o). This value is then used to calculate a test statistic (U_o) which, along with the sample sizes, is used to evaluate a p-value. Once the p-value of each gene has been calculated, these values are then ranked and the desired number of genes with the smallest p-values are used in the predictor. The test statistic for the Mann-Whitney Test is given by,

$$U_o = n_1 n_2 + \frac{1}{2} n_1 (n_2 + 1) - T_o$$

where n_1 and n_2 are the sample sizes of classes 1 and 2 respectively. The p-value is then calculated as twice the probability of getting a test statistic greater than or equal to U_o, assuming that the two population medians are truly equal. The desired number of genes with the smallest p-values are then retained for further investigation.

One of the strengths associated with using this test as a gene selection method is that it is a non-parametric test and is quick to calculate.

t-Test

Proposed by Gosset in 1908, the t-test is a parametric significance test which tests for the equality of two population means. It works on the assumption that the populations from which the two samples are drawn are normally distributed with different means and variances. The degrees of freedom that are used in calculating the p-value for this test depend on the number of samples for each gene. Once the p-values have been calculated, the desired number of genes with the smallest p-values are then retained for further investigation. The test statistic for the t-test is given by,

$$t_{oi} = \frac{\bar{g}_{1i} - \bar{g}_{2i}}{SE(\bar{g}_{1i} - \bar{g}_{2i})}$$

where $\bar{g}_{1i}$ is the average expression level of the ith gene in class 1 and $\bar{g}_{2i}$ is the average expression level of the ith gene in class 2. $SE(\bar{g}_{1i} - \bar{g}_{2i})$ is the standard error for the difference between the average expression level of the ith gene in classes 1 and 2.

Using the t-test as a gene selection method has a number of advantages and disadvantages. Advantages of using this method are that it is a well known and statistically rigorous test and it is also quick to calculate. A disadvantage of this method is that one of the assumptions of the t-test is that the distribution of data in the underlying population from which each of the samples is derived is normal (Sheskin, 1997).

SAM Adjusted t-Test

Statistical Analysis of Microarrays (SAM) was developed by Tusher et al. (2001) as a method for determining the significance of changes in expression levels between different biological states, whilst taking account of the very large number of genes being compared. The SAM Adjusted t-test is based on the Student's t-test (Gosset, 1908) with a constant, s_o, added to the pooled standard error in the denominator of the test statistic $d(i)$. This constant is chosen so as to minimise the coefficient of variation of the test statistic $d(i)$, which is computed as a function of the pooled standard error (Tusher et al., 2001). The new test statistic is calculated for each gene and the results sorted and the desired number of genes with the largest test statistics are used in the predictor. The test statistic for the SAM Adjusted t-test is given by,

$$d(i) = \frac{\bar{g}_1(i) - \bar{g}_2(i)}{s(i) + s_o}$$

where $s(i)$ is the pooled standard error across classes 1 and 2 for gene i.

Signal to Noise Ratio

Proposed for use in microarray analysis by Golub et al. (1999) the Signal to Noise Ratio (SNR) is also based on Gosset's (1908) t-test statistic, where the denominator is replaced with the sum of the standard deviations for each class instead of the pooled standard error. The new test statistic is calculated for each gene, the results sorted and the desired number of genes with the largest test statistics are used in the predictor.

$$SNR_i = \frac{\bar{g}_{1i} - \bar{g}_{2i}}{\sigma_{1i} + \sigma_{2i}}$$

Where $\bar{g}_{1i}$ is the average expression level for the ith gene in class 1 and σ_{1i} is the sample standard deviation of the expression levels for the ith gene in class 1. Similarly $\bar{g}_{2i}$ and σ_{2i} are the average and sample standard deviation for the expression levels for the ith gene in class 2.

The Signal to Noise Ratio penalises genes that have large variances in both classes to a greater degree than genes with a large variance in one class and a smaller variance in the other class. In doing so it may exclude genes that are important to the classification process. For this method to work effectively the data in each class should arise from similar distributions, in particular the Normal distribution. This can be seen as a major disadvantage of this gene selection method as the distribution of each class need not be similar.

Moderated t-statistic

Proposed by Smyth (2004) and implemented in the R package **limma** (Smyth, 2005), this method uses linear models and an empirical Bayes shrinkage method to estimate the fold change and standard errors for each gene. These models can take into account the correlation structure provided by technical replicates (e.g., duplicate arrays). This method works by first calculating the average correlation within the technical replicates if they exist (using the `duplicateCorrelation()` function in the **limma** package (Smyth, 2005)), it then uses this along with the experimental design and blocking structure of the technical replicates to build a linear model on a gene-by-gene basis to estimate fold change and standard errors. Once this has been done an empirical Bayes method is applied to moderate the standard errors and finally the genes are sorted in descending order by the moderated t-statistic, $\tilde{t}_{gj}$. To help describe how the unknown coefficients, β_{gj} from the same model fitted on a per gene basis, and the unkown variances σ_g^2 vary across the genes, prior distributions are assumed for these unknown parameters. For σ_g^2 the prior information that describes how the variances are expected to vary across the genes is equivalent to a prior estimator s_o^2 with d_o degrees of freedom (i.e. $\frac{1}{\sigma_g^2} \sim \frac{1}{d_o s_o^2}\chi^2_{d_o}$). For any given j, it is assumed that the contrast β_{gj} is non-zero with probability p_j. This probability is the expected proportion of genes that are truly differentially expressed. The prior information that describes the expected distribution of log-fold changes for genes that are differentially expressed (i.e., the nonzero coefficients) is assumed to be equivalent to a prior observation equal to zero with an unscaled variance υ_{oj} (i.e., $\beta_{gj}|\sigma_g^2$, $\beta_{gj} \neq 0 \sim N(0, \upsilon_{oj}\sigma_g^2)$) (Smyth, 2004). The per-gene moderated t-statistic is defined as,

$$\tilde{t}_{gj} = \frac{\widehat{\beta}_{gj}}{\tilde{s}_g\sqrt{\upsilon_{gj}}}$$

where $\widehat{\beta}_{gj}$ is the estimate of contrast β_{gj} for gene g, $\tilde{s}_g^2$ is the posterior variance and υ_{gj} is the jth diagonal element in the unscaled covariance matrix pre and post multiplied by the contrast matrix. Smyth (2004) states that, when the prior degrees of freedom, d_o, are zero the moderated t-statistic, $\tilde{t}_{gj}$, reduces to the ordinary t-statistic, and when the prior degrees of freedom tend to infinity the moderated t-statistic becomes proportional to the coefficient $\widehat{\beta}_{gj}$. The total degrees of freedom associated with the moderated t-statistic are the observed degrees of freedom for a given gene plus the prior degrees of freedom. This reflects the additional information that is borrowed from all of the genes, to aid in the inference for each individual gene (Smyth, 2004).

One of the major advantages this method has over the previously discussed methods is that, through the use of a well formulated linear model and empirical Bayes shrinkage, this gene selection method can take into account the correlation structure provided by

duplicate arrays for the same sample. The function `duplicateCorrelation()` which achieves this incorporation uses an algorithm that estimates the correlation between the technical replicates for each gene and then combines these values to obtain a common values across the genes (Smyth, 2005). This information can then be incorporated into a common correlation model which is fitted to the expression data for each gene (Smyth, 2005). For this model to be useful Smyth (2005) state that the per gene correlations need to be sufficiently stable (i.e., the per-gene correlations are reasonably constant).

3.4 Pre-Filtering Options

Possible advantages in using a pre-filter stem from the advantages of using a gene selection method. As Hu et al. (2006) pointed out, a good gene selection method can increase the accuracy of a classifier by removing irrelevant/noise genes thus drastically reducing the size of the dataset. Several pre-filtering methods have been proposed that aim to deal with the issue of redundancy in the produced gene-list. Three examples come from Hanczar et al. (2003), Jaeger et al. (2003) and Wang et al. (2005b). These methods involve combining gene expression values to form marker genes via different applications of cluster analysis methods, which are then used as variables in classifiers. In this approach, however, a single variable is no longer an individual gene but a combination of several genes. This has major implications when one is interested in finding individual genes that can be used in a prognostic tool in a clinical setting (e.g., in identifying a small collection of genes whose expression is to be measured by a non-array based diagnostic technology). The pre-filtering method described here works on a simpler notion of combining two gene selection methods, both of which deal with genes on an individual basis.

Proposed Filtering Method

Researchers often prefer genes to have undergone a certain level of differential expression so that when using another technology which cannot reliably detect subtle differences in expression, such as real time PCR, it is easier to determine in relation to a reference sample whether a gene is present in a new sample and if it is present, how active that gene is in the new sample. The gene selection method proposed in this chapter uses a fold change cut-off method, similar to that described in Section 3.3, as a pre-filter to remove genes that have not undergone a biologically relevant level of differential expression. The use of a fold-change cut-off method limits this pre-filter to situations where each sample is compared to a reference on each array (i.e., expression data obtained from two channel spotted microarrays with a reference design). The pre-filter is then followed by any of the statistical ranking methods described in Section 3.3 and

the most highly ranked genes are then used to construct a classifier. The proposed gene-selection method was designed specifically for gene expression data obtained from a spotted microarray experiment where each sample across two classes was compared with a reference sample. The pre-filter was designed so that it only retains a gene if at least $\alpha\%$ of the arrays/samples in either class have fold-change values above the biologically deemed level of expression. The $\alpha\%$ threshold was chosen to account for the possibility of the presence of subgroups within the dataset. These limits may be related to the dataset being studied, as in this case, or they could be based on prior information and expert knowledge. For example, when the goal of a two channel spotted microarray analysis (based on a large sample size) is to find genes that can be used in a diagnostic test which requires genes to have undergone a fold-change of at least 2 relative to a reference sample, the pre-filter could be applied so that a gene is only retained if at least 50% of the samples in either class have a fold-change of 2 or more. This is equivalent to the $\log_2$ Fold-Change for at least 50% of the samples for a given gene lying outside the range -1 to 1.

Figure 3.1 contains three plots representing the simulated $\log_2$ expression level for three genes in relation to a reference sample ($\log_2$ fold change), across multiple samples from two different classes. Plot (a) shows gene 1 as having a higher level of expression in class 1 compared to the reference sample and a similar level of expression in class 2 compared to the reference sample, implying that gene 1 is differentially expressed between the two classes, with gene 1 being more highly expressed in class 1. Plot (b) shows gene 2 as having a lower level of expression in class 1 compared to the reference sample and a similar level of expression in class 2 compared to the reference sample, implying that gene 2 is differentially expressed between the two classes, with gene 2 being more under expressed in class 1. Plot (c) shows gene 3 to have a similar level of expression in both classes compared to the reference sample, implying that gene 3 is equivalently expressed between the two classes. Using the parameters stated above in the pre-filter the genes represented in plots (a) and (b) in Figure 3.1 would be retained, whereas the gene represented in plot (c) would be discarded.

This pre-filter will also retain genes which exhibit an absolute fold change greater than two in both classes. Once all the genes that are deemed not to have undergone a biologically relevant level of differential expression have been removed, a statistical ranking method, such as the moderated t-statistic, is then used to rank the remaining genes in order of their ability to distinguish between the two classes. This approach aims to produce a set of genes that, via a classifier, can effectively and efficiently discriminate between two classes whilst producing a more biologically meaningful interpretation of the list of genes (or gene signature) produced by the classifier.

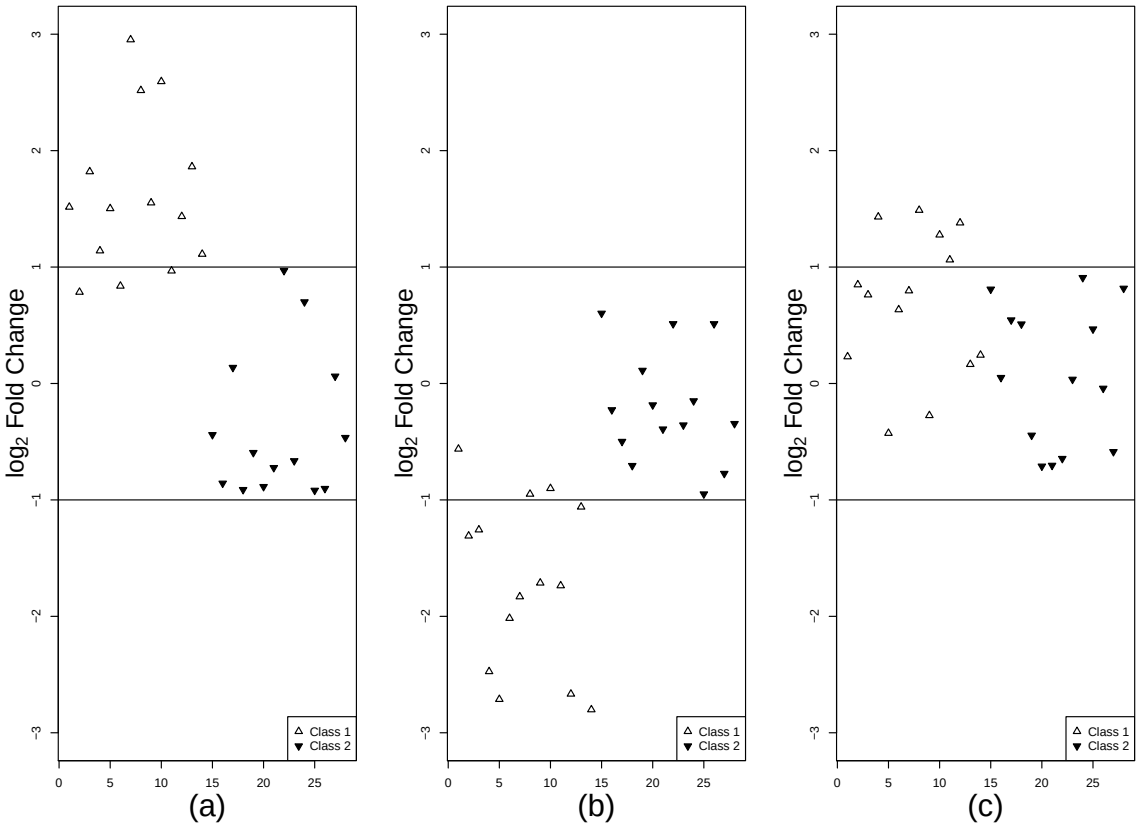

Figure 3.1: Visualisation of the pre-filter method with range set at -1 to 1 and α equal to 50%. Each panel represents the simulated $\log_2$ fold change with respect to a reference sample for one of three genes across multiple samples from two classes. Using the parameters stated above the genes represented in plots (a) and (b) would be retained by the pre-filter whereas the gene represented in plot (c) would not be retained.

3.5 Predictor Construction

There were two important criteria that had to be taken into account when a classifier was chosen to assess the effect on the predictive performance when the pre-filter was used in combination with the five standard gene selection methods, described in Section 3.3, compared to when the pre-filter was not used. The first of these was that the classifier must be considered as a benchmark classification method that has a reported high level of success in this field to date. The second criteria was that the classifier should have been shown to be compatible with the standard gene selection methods described in Section 3.3 that were used to rank the genes.

Classification Method

Hu et al. (2006) described the Support Vector Machine (SVM) classifier as a benchmark classification algorithm, which can not only handle high dimensional datasets but has also been shown to outperform other existing classification methods (Furey et al. (2000),

Cho and Won (2003) and Fan and Ren (2006)). As a result of their analyses Hu et al. (2006) concluded that of the gene selection methods they tested, the Signal to Noise Ratio was a compatible gene selection method for the SVM classifier. Ma et al. (2005) reported on the success of using non-parametric gene selection methods with the SVM classifier. In particular they showed that a good prediction accuracy could be attained when using the Mann-Whitney U statistic in combination with the SVM classifier. Chu and Wang (2005) showed that a high level of predictive performance can be attained when using the t-test based gene selection method with a SVM classifier.

Proposed by Vapnik (1995), Support Vector Machines work by finding the "maximal margin hyperplane" in a linear kernel-induced feature space. For two classes it constructs a rule than best separates the two classes using the variables given, as described in Chapter 2. Linear SVM's can produce the probability of a given observation belonging to each class (i.e., good and poor response). Standard practice is to classify the observation to the class with the largest probability. If two dual channel spotted microarrays were produced for each observation (i.e., technical replication, such as that seen in the microarray data used in this chapter) and the SVM is fitted using all of the data so that for each replication the probability of belonging to each class has been calculated, then one possible way to use the information from both of these replications is to take the average of the two probabilities generated for each class, and classify the observation to the class with the larger average probability.

Model Fit

The classification models used in this chapter were constructed in the following way. A linear SVM classification model was fitted using the top 2 ranked genes found via each of the five statistical gene selection methods described in Section 3.3, and those five gene selection methods combined with the pre-filter as described in Section 3.4 (10 models in total). For each model the proportion of samples that were correctly classified by the classifier (the classification rate), the proportion of samples that belong to class 1 that were correctly classified as belonging to class 1 by the classifier (sensitivity) and the proportion of samples that belong to class 2 that were correctly classified as belonging to class 2 by the classifier (specificity) were calculated by leaving the data for one sample out at a time (i.e., by using leave-one-out cross-validation (LOOCV)). This was repeated using the top 3 ranked genes found via the gene selection and pre-filter methods in the linear SVM classification model, continuing until models using the top 100 ranked genes were fitted. The model with the highest classification rate plus the smallest number of genes used to construct the model was chosen and the classification rate, sensitivity and specificity were recorded for comparison across the 10 methods.

As all of the data is used to choose the "best" model, that is by choosing the

model with the fewest number of genes that gives the highest classification rate under LOOCV, selection bias has been introduced. To account for this bias two different bias adjustment techniques were employed, these were the $B.632$ bootstrap (Efron, 1983) and nested Leave-One-Out or Double Leave-One-Out (Varma and Simon, 2006; Cawley and Talbot, 2006). Where stated, in the next two sections, bootstrap bias calculations (See Chapter 2 for details) were performed for the chosen models for each of the gene selection and pre-filter methods to remove the selection bias incurred when using the above Leave-One Out (LOO) classification method. In brief, for the LOO classification rates obtained using the above classifier, the $B.632$ bootstrap (Efron, 1983) estimate was calculated as the weighted sum of the average bootstrap error rate (taken over at least 500 bootstrap samples) and the observed error rate. This estimate was then subtracted from 1 to obtain an estimate of the unbiased LOO classification rate. Another way to achieve unbiased performance statistics is to leave each observation out in turn and apply Leave-One-Out on the remaining data, building a classification model and testing it on the observation left out. This is known as nested Leave-One-Out or Double Leave-One-Out (DLOO) (Varma and Simon, 2006; Cawley and Talbot, 2006). This method was also performed in some of the analyses in this chapter, as a way to achieve unbiased performance results. However, due to the overly computationally intensive nature of this method it was only used in a small number of analyses, where only one dataset was analysed.

3.6 Melanoma Data

The microarray gene expression data used in this chapter were obtained from a microarray experiment conducted by John et al. (2008) in which tissue samples were taken from tumours surgically removed from 29 patients diagnosed with stage IIIB and IIIC melanoma. Sixteen of the 29 patients exhibited "poor" post-surgery response (rapid tumor progression to stage IV melanoma, class 0), while the remaining thirteen patients exhibited a "good" response (tumor progression to stage IV melanoma was greater than 24 months, class 1). The average time to tumor progression for the patients in the "poor" response class was 4 months, in the "good" response class this was 40 months. Two dual channel spotted microarrays were produced for each patient giving rise to technical replication, each microarray compared the tissue sample for a given patient with a reference sample and consisted of 32,016 spots from 30,888 distinct oligonucleotide probes which were obtained from MWG Biotech. The reference sample was comprised of a variety of tumors including melanoma cell lines and normal tissues (John et al., 2008). The expression data was extracted using the GenePix v6.0 software (Axon Instruments, Union City, CA) and normalised using Lowess print-tip

normalisation (described in Chapter 1).

The goal of John et al. (2008) was to develop a gene-expression based tool for determining prognosis in stage III melanoma. To this end, 21 differentially expressed genes were found using the Mann-Whitney test with multiple testing adjusted for via the false discover rate controlling method of Benjamini and Hochberg (John et al., 2008). To confirm the microarray results for the 21 differentially expressed genes John et al. (2008) performed quantitative PCR (qPCR) for each of these genes. They found that 15 of the 21 differentially expressed genes showed a strong cross-platform correlation. These 15 genes were used to construct two predictive scores, one based on the microarray data and the other based on the qPCR data. When the microarray based predictor was applied to an independent validation set consisting of 10 stage III tumour samples a classification rate of 90% was achieved. When the qPCR based predictor was applied to a different independent validation set consisting of 14 stage III tumour samples a classification rate of 85% was achieved. John et al. (2008) concluded that via the use of microarray expression data and qPCR it was possible to produce a gene expression profile which was capable of predicting clinical outcome in what was previously believed to be a homogeneous group of stage III melanoma patients.

The goal of the analysis presented here is to compare the effect on the performance statistics and the rankings of the top 100 genes when the technical replication in the data was taken into consideration throughout the analysis, and when it is only taken into consideration in the classification step. The details and results from this comparative analysis are presented in Section 3.8.

3.7 Simulated Datasets

The simulation of data that accurately mimics log scaled normalised microarray gene expression data is challenging (Nykter et al., 2006). This is because microarray gene expression data is a complex layering of several factors and steps involved in the data production process including slide preparation, hybridization, and biological dependencies between genes that are correlated with one another, resulting in a complex dataset that generally contains a high level of noise. A number of methods for simulating gene expression data have been proposed (e.g., Singhal et al. (2003) and Nykter et al. (2006)). These range from very simplistic methods that make a large number of assumptions to simplify the simulation process (such as assuming that the genes are not correlated and that log scaled normalised gene expressions can be modeled by a Multivariate Normal distribution with a constant variance (Varma and Simon, 2006)), through to complex methods that take into account the steps involved in producing microarray gene expression data and the biological relationships and dependencies between the genes (Nykter

et al., 2006).

The simulation methods used in this chapter range from very simple methods that simulate log scaled normalised gene expression data from the Normal distribution with the same variance but different means, to methods that fit two-component mixture models with estimates derived from the analysis of the melanoma data described in the previous section. With the aim of producing simulated data that closely mimicked that of the melanoma data, a total of six simulation methods were developed. These methods are split into two groups, initial methods which were simplistic, for which only one dataset was generated and only a few gene selection methods were applied, and the second group consisting of more complex data driven methods for which multiple datasets were generated per simulation method. In addition to these six simulation methods, the simulation method proposed by Nykter et al. (2006) (described in Chapter 2) was investigated, however efforts to implement this simulation method failed due to an error in compatibility between the software package and Matlab.

3.7.1 Initial Simulation Methods

Simulation Method 1

The first and most simple simulation method used the Normal distribution with the variance set to 1 for all genes and varied the mean from 0 for genes set to be differentially expressed (DE). One dataset (Sim 1a) with a total of 60 arrays, equally spilt between two classes, with 10,000 genes was simulated with 500 genes set as DE. All the simulated expression values for the genes in class 1 where randomly generated from the Standard Normal distribution (i.e., N(μ=0, σ^2=1)). The 500 simulated expression values for the genes in class 2 that were set as DE were randomly generated from six Normal distributions with means of -2, -1.5, -1, 1, 1.5 and 2, where 50 expression values were simulated from each of the distributions with means equal to -2 and 2 and 100 expression values were simulated from each of the four other distributions. The remaining 9500 gene expression values where simulated from the Standard Normal distribution. The means for the DE genes were changed to -1, -0.5, -0.2, 0.2, 0.5 and 1 respectively and another dataset (Sim 1b) was generated as above. The SVM classifier was applied to both of these datasets using LOO and DLOO in combination with the t-test gene selection methods and the pre-filter combined with the Mann-Whitney and t-test gene selection methods.

Simulation Method 2

The second simulation method was set up in the following way. If X_{ij} is the expression level of gene i on array j in class 1, and Y_{ij} is the expression level of gene i on array j

in class 2 then, $X_{ij} \sim N(\mu_{1i}, \sigma^2)$ and $Y_{ij} \sim N(\mu_{2i}, \sigma^2)$. If gene i shows equal expression in both classes then $\mu_{1i} = \mu_{2i} = \mu_i \sim N(\mu_o, \tau^2)$. If gene i shows differential expression between the two classes then μ_{1i} and μ_{2i} are considered to arise independently from a $N(\mu_o, \tau^2)$ distribution. For this simulation method μ_o was set at zero and σ^2 and τ^2 were estimated via a Log-Normal Normal (LNN) model, implemented in the R package **EBarrays** (Kendziorski et al., 2006). This model was fitted to the melanoma data (see Section 3.6 for a full description of this data). A dataset containing a total of 60 arrays, equally spilt between two classes, and 10,000 genes with a probability of a gene being DE set at 0.05, was simulated using this method. This method was repeated with $\mu_{1i} = 0$ and then again with the LNN model applied to the melanoma data with the technical replicate for each observation removed and the probability of a gene being DE (PDE) also being estimated by the LNN model.

Simulation Method 3

The third simulation method utilises the same notation as that used in the second simulation method. The difference here is that if gene i shows differential expression between the two classes then $\mu_{2i} = \mu_{1i} + \Delta_i$ where $\mu_{1i} \sim N(\mu_0, \tau^2)$ and $\Delta_i \sim N(0, \eta^2)$. The parameters for these distributions were calculated in the following way; η^2 was estimated directly from the melanoma data as the variance of the differences between the class means for each gene, whereas μ_0, σ^2 and τ^2 where all estimated from the same LNN model that was fitted to the melanoma data with the technical replicate for each observation removed.

One dataset was simulated using these parameters with the PDE set at 0.05. This dataset contained 10,000 simulated genes and 60 simulated arrays which were equally split between the two classes of "good" and "poor" response.

3.7.2 Data Driven Simulation Methods

Simulation Method 4

In an attempt to mimic the melanoma data more effectively the fourth simulation method derived μ_0, σ^2 and τ^2 directly from the melanoma data in the following way:

- $\hat{\sigma}^2 = \left(\sum_{\substack{k \\ classes}} \sum_{\substack{i \\ genes}} \sum_{\substack{j \\ arrays}} (x_{kij} - \bar{x}_{ki.})^2 \right) / \left(\sharp arrays \times \sharp genes - 1 \right)$ where $\bar{x}_{ki.}$ is the mean expression level of gene i in class k.
- $\hat{\mu}_o = \left(\sum_i \bar{x}_{1i.} \right) / \left(\sharp genes \right)$ The mean of all the mean expression levels in class 1.
- $\hat{\tau}^2 = var(\bar{x}_{ki.})$ the variance of all the mean expression levels in class 1 and class 2.

- $\hat{\eta}^2 = var([\bar{x}_{1i.} - \bar{x}_{2i.}])$ the variance of the differences between the mean expression levels in the two classes, as per the third simulation method.

For this simulation method five hundred datasets were generated. Each dataset consisted of 10,000 synthetic genes, 60 synthetic arrays (equally split into two classes) and the probability that a given gene is differentially expressed was set to 0.05, thus approximately 500 of the 10,000 synthetic genes are differentially expressed.

Simulation Method 5

The fifth simulation method was a minor change on the fourth simulation method where instead of simulating equal numbers of arrays for the two classes, the class deemed to be good response (class 1) contained 45 simulated arrays and the poor response class (class 2) contained only 15 simulated arrays. Further changes to this method comprised of fixing the distribution parameters at different values to assess the effect on the number of simulated genes set as DE and the number of these DE genes being retained by the classifier after using the pre-filter combined with the SAM adjusted t-test gene selection method. Four different sets of parameter values were tested (Table 3.1 gives these values compared to the initial values), 100 datasets were simulated for each of the four sets of parameters and LOO SVM with the pre-filter combined with the SAM adjusted t-test gene selection method were applied to each dataset.

Table 3.1: The first column shows the initial parameter values used in simulation method five. The columns labeled A - D show the four different sets of parameter values used to determine if any improvement in the average number of differentially expressed genes retained in the classified and/or the average classification rate could be attained using simulation method six.

Parameter	Initial Values	A	B	C	D
$\hat{\sigma}^2$	0.1485	0.1	0.09	0.09	0.09
$\hat{\mu}_o$	-0.01899	0.02	0.02	0.02	0.02
$\hat{\tau}^2$	0.2626	0.2	0.19	0.19	0.19
$\hat{\eta}^2$	0.0231	0.2	0.2	0.2	0.2
Probability of Differential Expression	0.05	0.05	0.05	0.07	0.09

Simulation Method 6

To try to capture the nature of real microarray data more adequately the sixth simulation method used two-component mixture models to estimate both μ_{1i} and Δ_i the distribution parameters for these two values were estimated via the model based clustering method applied to the melanoma data. This method is implemented in the R

package **mclust** (Fraley and Raftery, 2006). The mean expression levels for the genes in the "good" response class in the melanoma data were fitted with a 2-component mixture and the estimates used in the formulation of μ_{1i}. The estimates obtained from modeling the observed differences between the mean expression levels for the genes in the "good" and "poor" response classes in the melanoma data, via a 2-component mixture model were used in the formulation of Δ_i, so that,

$$\mu_{1i} = \hat{\pi}_0 N(\hat{\mu}_0, \hat{\tau}_0^2) + (1 - \hat{\pi}_0) N(\hat{\mu}_1, \hat{\tau}_1^2)$$

$$\Delta_i = \hat{\rho}_0 N(\hat{\mu}_2, \hat{\eta}_0^2) + (1 - \hat{\rho}_0) N(\hat{\mu}_3, \hat{\eta}_1^2)$$

with σ^2 estimated directly from the melanoma data, as was done in the fourth simulation method. This method was refined by simulating σ^2 from an inverse Gamma distribution with parameters estimated by fitting a Gamma distribution to the inverse of the observed variances of the genes in the "good" response class in the melanoma data (see Figure 3.2). The inverse Gamma distribution was chosen as it is the conjugate prior that is placed on the variance of the Normal distribution, in a Bayesian setting, and is thus a reasonable candidate for simulating σ^2.

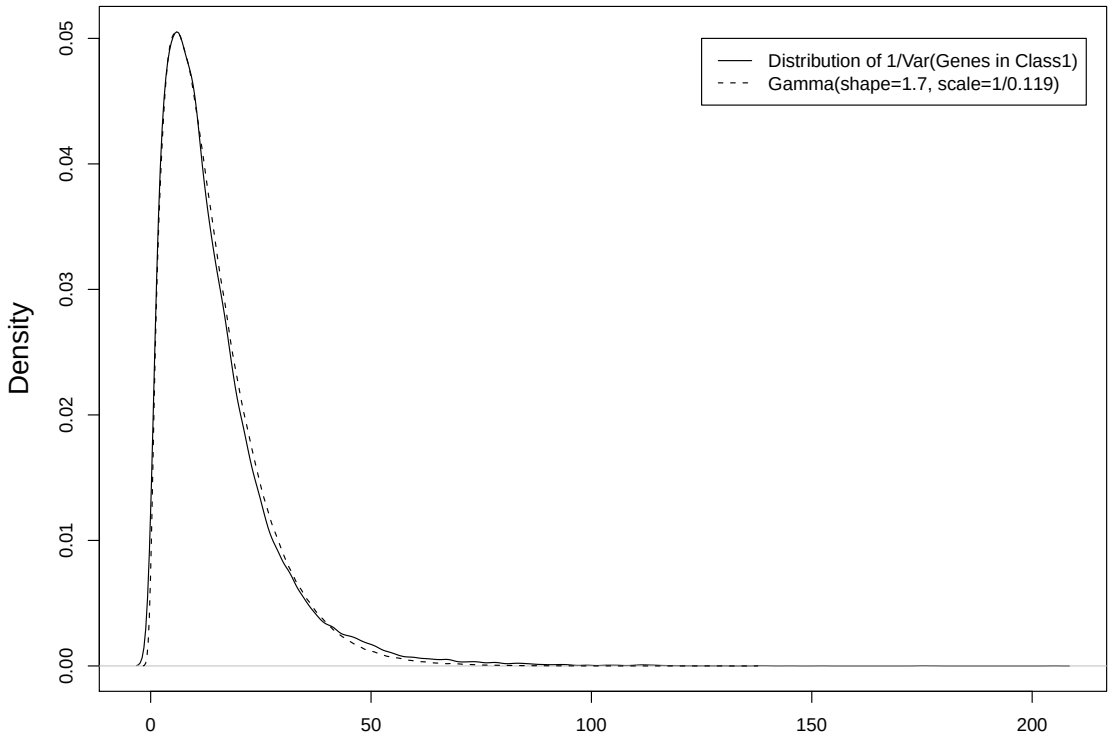

Figure 3.2: Estimating σ^2 by fitting a Gamma distribution to the inverse of the observed variances of the genes in the "good" response class in the melanoma data (class 1).

The mixture component on Δ_i was dropped in favour of the distribution used in the third simulation method, namely $N(0, \eta^2)$, and finally a minor change to the formulation of the two-component mixture model for μ_{1i}. This was done so as to pull out the tails of the corresponding distribution to give a better likeness to the distribution of the observed gene means for the "good" response class from the melanoma data. This change included setting $\hat{\pi}_0$ to 0.75 and dividing $\hat{\tau}_0^2$ in the first term by 2 so that

$$\mu_{1i} = \hat{\pi}_0 N(\hat{\mu}_0, \hat{\tau}_0^2/2) + (1 - \hat{\pi}_0) N(\hat{\mu}_1, \hat{\tau}_1^2)$$

where $\hat{\mu}_o$, $\hat{\tau}_0^2$, $\hat{\mu}_1$ and $\hat{\tau}_1^2$ were taken from a two-component mixture model fitted to the observed gene means in the "good" response class from the melanoma data without the technical replicates for each to the observations. An initial one hundred datasets where simulated using this method. Each of these datasets consisted of:

- 10,000 synthetic genes,
- 60 synthetic arrays split into two classes the "good" response class containing 30 synthetic arrays and the "poor" response class containing a mixture of 15 "good" and 15 "poor" response synthetic arrays,
- the probability that a given gene was differentially expressed was set at 0.05, thus approximately 500 of the 10,000 synthetic genes were seen to be differentially expressed.

The reason for making a heterogeneous "poor" response class was to test if a classifier constructed using genes retained by the pre-filter in combination with any of the five gene selection methods would perform at the same level as a classifier constructed using genes retained by any of the five gene selection methods.

3.7.3 Simulation Results

When the pre-filter was used in the following analyses, α was set at 50% and the range values on the $\log_2$ Fold-Change was set at -1 to 1, representing a Fold-Change of 2 or more. Results from varying these parameters are discussed in Section 3.8. The classifier was constructed as described in Section 3.5 *Model Fit*.

3.7.4 Initial Simulation Results

Simulation Method 1

The Leave-One-Out (LOO) and Double-Leave-One-Out (DLOO) results for the first simulated dataset, Sim 1a, are displayed in Tables 3.2 and 3.3. The DLOO was used to overcome the selection bias incurred by using all of the data to select the number of

genes that produces a classifier with the highest LOOCV classification rate. The gene selection methods used to analyse this simulated dataset were the t-test and the pre-filter in combination with the t-test, and the pre-filter combined with the Mann-Whitney test. The Mann-Whitney test was only considered in combination with the pre-filter to provide a comparison with the pre-filter combined with the t-test in the event of there being a difference in the performance results for these two methods. If a substantial difference had been seen, then the Mann-Whitney test would have been used by itself to see if the difference was due to the gene selection technique. In Table 3.4 both the LOO and DLOO performance results are given for the application of the pre-filter in combination with the Mann-Whitney test to the second simulated dataset Sim 1b. The perfect and near perfect performance statistics from these analyses indicates that the simulated data generated using this method did not capture the inherent noise in "real" gene expression data adequately. These results indicate that the assumptions imposed by this method were not appropriate when considering the nature of microarray gene expression data, and that this simulation method may be overly simplistic.

Table 3.2: Comparison of the SVM leave-one-out (LOO) classifier prediction results achieved for simulation method 1a

		Pre-Filter Combined with:	
	t-Test	t-Test	Mann-Whitney
	90 models with	90 models with	88 models with
Sensitivity	1	1	1
Specificity	1	1	1
Classification Rate	1	1	1

Table 3.3: Comparison of the SVM double LOO (DLOO) classifier prediction results achieved for simulation method 1a

		Pre-Filter Combined with:	
	t-Test	t-Test	Mann-Whitney
Sensitivity	1	1	0.93
Specificity	0.97	0.97	0.97
Classification Rate	0.98	0.98	0.95

Table 3.4: Comparison of the SVM LOO and DLOO classifier prediction results achieved for simulation method 1b

	Pre-Filter Combined with Mann-Whitney	
	LOO	DLOO
	76 models with	
Sensitivity	1	1
Specificity	1	0.97
Classification Rate	1	0.98

Simulation Method 2

When the Log-Normal Normal (LNN) model was fitted to the melanoma data (see Section 3.6 for a full description of this data) it gave the following values; $\hat{\sigma}^2 = 0.0738$ and $\hat{\tau}^2 = 0.1297$. These estimates were used to generate a single dataset containing 60 arrays, equally split into two classes, and 10,000 genes with a probability of a gene being differentially expressed (DE) set at 0.05. This simulation method was repeated with $\mu_{1i} = 0$ to generate a second dataset and then again with the LNN model applied to the melanoma data with the technical replicate for each observation removed and the probability of a gene being DE (PDE) also being estimated by the LNN model, which produced the following estimates; $\hat{\mu}_o = -0.0149$, $\hat{\sigma}^2 = 0.0765$, $\hat{\tau}^2 = 0.1259$ and pde = 0.0077.

All three of the simulated datasets generated using the second simulation method and the two modifications to that method showed perfect LOO performance statistics when the SVM classifier was applied with the pre-filter in combination with the Mann-Whitney gene selection method. The first simulated dataset using simulation method 2 showed perfect LOO performance statistics for every model built using the SVM classification method described in Section 3.5. The second simulated dataset only allowed 22 of the 10,000 simulated genes past the gene selection process (which used the pre-filter combined with the Mann-Whitney test) thus limiting the number of models created to 21. All but one of these models produced perfect LOO performance statistics. The third simulated dataset produced perfect LOO performance statistics for 80 of the 99 models constructed using the SVM classifier described in Section 3.5 when used in combination with the pre-filter and Mann-Whitney gene selection method. The results from these three simulated datasets imply that this simulation model, with the two variations in the parameters, did not generate simulated data that adequately captured the complexity of microarray gene expression data.

Simulation Method 3

When the parameters for the third simulation method were generated as described in Section 3.7.1 the following estimates were obtained; $\eta^2 = 0.0231$, $\hat{\mu}_o = -0.0149$, $\hat{\sigma}^2 = 0.0765$ and $\hat{\tau}^2 = 0.1259$. These estimates were then used to generate a single dataset that consisted of 60 arrays, equally split into two classes, and 10,000 genes with a probability of a gene being differentially expressed (DE) set at 0.05.

This dataset produced perfect LOO performance statistics in 41 of the 99 models generated using the SVM classifier, as shown in Figure 3.3. The pre-filter in combination with the Mann-Whitney gene selection method was used in the gene selection step, prior to the construction of the classification models. The approximate 95% asymmetric confidence intervals displayed in Figure 3.3 as vertical black lines, were calculated using

the R function `binconf(n*p,n,alpha=0.05,method="wilson")` from the R package **Hmisc** (Harrell Jr et al., 2009). Where `p` was the sensitivity, specificity or classification rate for a given classifier and `n` was the number of samples/arrays used in the calculation of `p` (either 30 for the sensitivity and specificity or 60 for the classification rate). Wilson's method (Wilson, 1927) for calculating the 95% asymmetric confidence intervals was chosen because Agresti and Coull (1998) have shown that this method tends to produce confidence intervals that have coverage probabilities that are closer to the specified coverage level than the other method available in the R function `binconf()`. The ability to easily distinguish between the two classes implies that the simulated data did not adequately capture the inherent noise in "real" gene expression data.

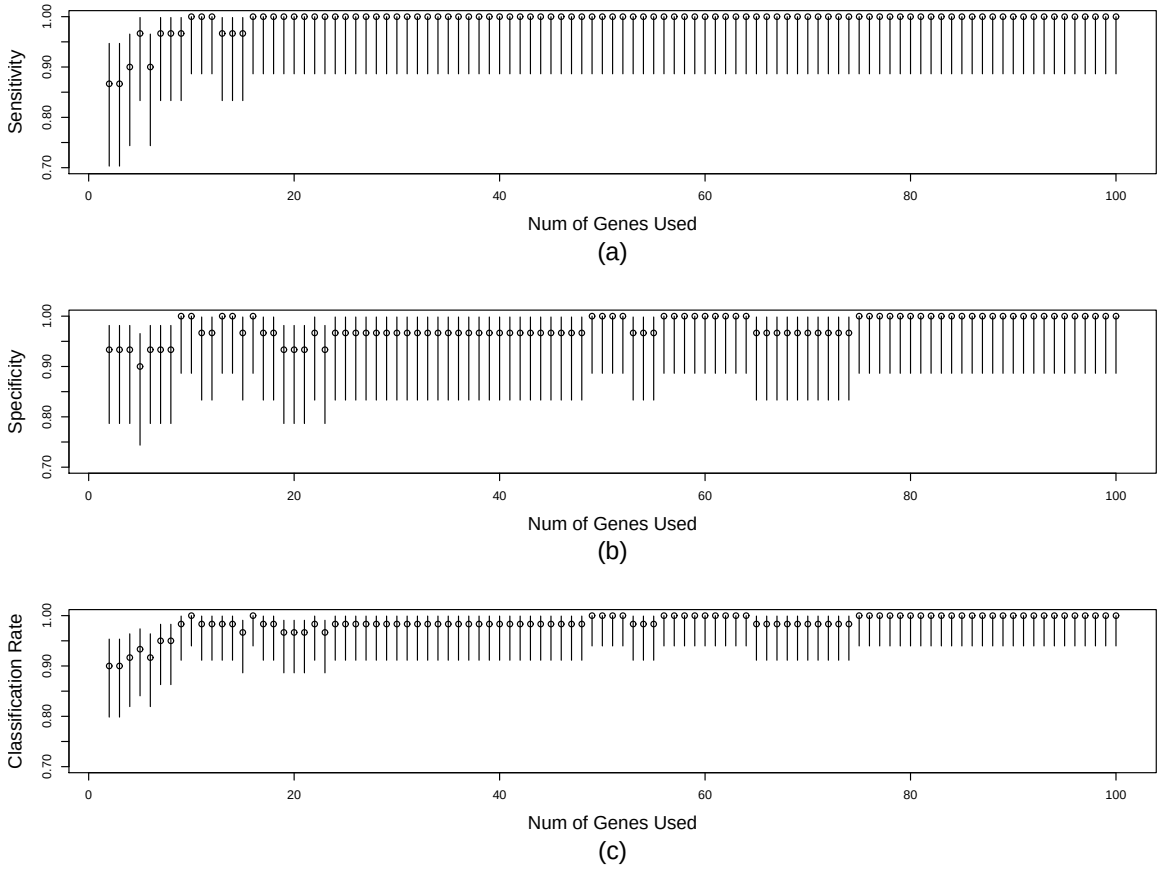

Figure 3.3: Performance statistics obtained when applying the SVM classifier used in combination with the pre-filter and Mann-Whitney gene selection method to simulated data generated using Simulation Method 3. The horizontal axis shows the number of genes used in each classifier and the black lines represent the approximate 95% asymmetric confidence intervals for each classifier. Plot (a) shows the proportion of "poor" response samples that were correctly classified (i.e., the sensitivity), (b) shows the proportion of "good" response samples that were correctly classified (i.e., the specificity) and (c) shows the proportion of all the samples that were correctly classified (i.e., the classification rate) for each of the 99 classifiers.

3.7.5 Data Driven Simulation Results

Simulation Method 4

When the parameters for the fourth simulation method were generated as described in Section 3.7.2 the following estimates were obtained: $\hat{\sigma}^2 = 0.1485$, $\hat{\tau}^2 = 0.2626$, $\hat{\mu}_o = -0.01899$, and $\hat{\eta}^2 = 0.023096$. These parameters were used to generate 500 simulated datasets, each consisting of 10,000 genes, 60 arrays equally split between two classes and the probability of a given gene being differentially expressed was set to 0.05, thus approximately 500 of the 10,000 genes were set as differentially expressed.

As a large number of datasets were simulated for this simulation method it was decided to use the $B.632$ bootstrap bias correction method as it is less computationally intensive than accounting for the bias by using DLOO. Tables 3.5 and 3.6 summarise the average LOO $B.632$ Bootstrap bias corrected classification results obtained when applying the five gene selection methods with and without the pre-filter prior to the construction of the classification models. These results were based on 500 simulated datasets generated using simulation method 4. In order to calculate a bias adjusted estimate a bootstrap sample was generated for each simulated dataset. The bootstrap classification rate for each method was calculated using the samples in the simulated dataset that were not in the bootstrap sample. The average bootstrap classification rate was then taken across the 500 simulated datasets and used in calculating the $B.632$ bootstrap corrected estimate of the classification rate.

For each of the ten methods approximately the same amount of bias was removed from the LOO results (average bias estimate of 0.13 with an approximate 95% confidence interval of 0.12 to 0.14). For each of the average classification rates a 95% confidence interval (CI) was calculated using a standard approach for calculating a confidence for a single mean. That is $\bar{\theta} \pm t_{df} \times se(\bar{\theta})$, where $\bar{\theta}$ is the average performance statistic, t_{df} is the appropriate quantile from the Student's t distribution and $se(\bar{\theta})$ is the standard error of the average classification rate. This method was also used to calculate 95% confidence intervals for the average number of genes set as differentially expressed, the average number of differentially expressed genes that were retained by the gene selection methods, and for the average number of genes used in the classifier.

From Tables 3.5 and 3.6 it is clear that the bias corrected average classification rates were slightly lower and more variable when the pre-filter was used in combination with the five "out-of-the-box" gene selection methods described in Section 3.3. These tables also show that there was a large drop in the average number of the differentially expressed genes being retained by the gene selection methods when combined with the pre-filter. The average number of genes retained by the classifiers was nearly half of that observed when the pre-filter was not used in combination with gene selection methods,

and the average classification rate of the classifiers were only slightly lower when the pre-filter was used in combination with the gene selection methods. One reason for the large drop in the number of differentially expressed genes retained by the selection methods when the pre-filter is used, is that a large number of the differentially expressed genes exhibit only a small change in expression between the two classes (lower than the threshold set by the pre-filter).

The one exception to this was the moderated t-statistic where, on average, a very large number of the differentially expressed genes were being retained both in combination with the pre-filter and without the pre-filter. This increase in the number of differentially expressed genes being retained by the moderated t-statistic is due in part to the way the statistic is calculated. The classification rates for these two gene selection methods were seen to be similar to the classification rates observed for the other gene selection methods. This implies that although this gene selection method was able to identify a larger proportion of the genes that were set to be differentially expressed, these genes were of no additional benefit to the classifier.

To assess the stability of these results a further 500 datasets were generated to which the pre-filter in combination with the Mann-Whitney and then with the moderated t-statistic were applied. The LOO classification results obtained from these analyses were combined with those in Table 3.6. When these combined results were compared to those obtained using only 500 datasets the only difference was a very slight narrowing of the confidence intervals. This implies that the generation of 500 datasets was enough to provide stable results.

To test how the pre-filter in combination with the Mann-Whitney gene selection method fared when there was no differential expression between any of the genes a simulated dataset was generated using simulation method 4 with genes in both classes arising independently from the same distribution. As only one dataset was simulated DLOO was used to provide unbiased performance statistics. The DLOO results from this dataset produced a classification rate of 58% with Sensitivity and Specificity rates of 63% and 53% respectively (95% asymmetric confidence intervals, obtained using the `binconf()` function, for these estimates were [0.45, 0.70], [0.45, 0.78] and [0.36, 0.69] respectively). As both classes contained the same number of samples, these observed performance results fall within what would be expected if a sample was classified solely by chance (i.e., a 50-50 chance of being correctly classified), thereby implying that when the pre-filter was used in combination with the Mann-Whitney gene selection method, the classifier did not "invent" a difference between the two classes where there was none. When a similar dataset was generated with 5% of the genes showing differential expression the DLOO results showed a classification rate of 70%, sensitivity of 57% and specificity of 83% (95% asymmetric confidence intervals, obtained using the `binconf()`

function, for these estimates were [0.57, 0.80], [0.40, 0.73] and [0.66, 0.92] respectively). This implies that for this simulated dataset, when the pre-filter was used in combination with the Mann-Whitney gene selection method, the classifier was able to differentiate between the two classes with a higher degree of certainty than if there were no difference between the two classes.

Simulation Method 5

For the fifth simulation method 500 datasets were generated, to which the SAM adjusted t-test gene selection method and the pre-filter combined with the SAM adjusted t-test gene selection method were applied. Initially only one gene selection method, with and without the pre-filter, was applied to these 500 simulated datasets to determine if any improvement in the classification rate could be attained when using the pre-filter. If this had been the case then the other gene selection methods would have been applied. The choice of using the SAM adjusted t-test gene selection method was based on the knowledge that of the five gene selection methods considered it is one of the two methods specifically designed for gene expression data. The average LOO classification rate when using the SAM adjusted t-test to select genes was 94% (95% CI 93.6% to 94.3%) compared to 86% (95% CI 85% to 87%) when the pre-filter was used in combination with the SAM adjusted t-test gene selection method. The average number of genes retained in the classification models that were set as differentially expressed decreased from 27/499, when the SAM adjusted t-test gene selection method was used to 9/499 when the pre-filter was used in combination with the SAM adjusted t-test. These results suggest that when the pre-filter was used in combination with the SAM adjusted t-test a slightly lower number of the genes retained by the gene selection method were set as differentially expressed (DE) and the predictive ability of the classifier was slightly lower.

A further 100 datasets were generated using this simulation method, to which the pre-filter in combination with the SAM adjusted t-test was applied. The average number of DE genes retained in the classification process as a proportion of the average number of genes set as DE was approximately 2% (8/499). When the parameters used to generate these datasets were varied (see Table 3.1 for these values) this proportion dropped to 1% for all four sets of parameter changes. Perfect (i.e., 100%) average LOO classification rates were obtained for all four sets of parameter changes. These results imply that although an improvement in the average LOO classification rate could be attained using this simulation method, the average number of genes retained in the classifier that were set as DE was approximately halved and that these classifiers could be over-fitting the data.

Table 3.5: Comparison of the average SVM LOO $B.632$ bootstrap bias corrected prediction results obtained when each of the five gene selection methods were applied to 500 simulated data sets generated using simulation method 4. 95% confidence intervals (CI) are provided in brackets.

	t-Test	SAM t-Test	Mann-Whitney	Signal to Noise Ratio	Mod t-Stat
Average Number of Genes Chosen to be DE (95% CI)	502 (500, 504)	499 (497, 501)	500 (498, 502)	499 (497, 501)	502 (500, 504)
Average Number of DE Genes Retained by Selection Methods (95% CI)	86 (85, 86)	83 (83, 84)	76 (75, 76)	85 (85, 86)	173 (168, 177)
Average Number of Genes Retained by Classifier (95% CI)	31 (29, 32)	31 (29, 32)	31 (29, 32)	30 (29, 32)	31 (30, 33)
Average Classification Rate with 95% CI	0.98 (0.98,0.98)	0.98 (0.98,0.98)	0.97 (0.97,0.98)	0.98 (0.98,0.98)	0.98 (0.98,0.99)
Bias Corrected Average Classification Rate	0.85	0.87	0.84	0.84	0.87
Standard Deviation of Classification Rate	0.03	0.02	0.03	0.03	0.02

Table 3.6: Comparison of the average SVM LOO $B.632$ bootstrap bias corrected prediction results obtained when the pre-filter was combined with each of the five gene selection methods and were applied to 500 simulated data sets generated using simulation method 4. 95% confidence intervals (ECI) are provided in brackets.

	Combined with:				
	t-Test	SAM t-Test	Mann-Whitney	Signal to Noise Ratio	Mod t-Stat
Average Number of Genes Chosen to be DE (95% CI)	501 (499, 503)	501 (499, 503)	499 (498, 501)	499 (497, 501)	499 (498, 501)
Average Number of DE Genes Retained by Selection Methods (95% CI)	24 (24, 25)	24 (24, 24)	22 (22, 23)	24 (24, 24)	156 (152, 160)
Average Number of Genes Retained by Classifier (95% CI)	18 (16, 19)	17 (16, 18)	17 (16, 19)	17 (16, 19)	18 (16, 19)
Average Classification Rate with 95% CI	0.86 (0.86,0.87)	0.87 (0.86,0.87)	0.85 (0.84,0.86)	0.86 (0.85,0.86)	0.86 (0.85,0.86)
Bias Corrected Average Classification Rate	0.73	0.73	0.72	0.73	0.73
Standard Deviation of Classification Rate	0.07	0.07	0.07	0.07	0.07

Simulation Method 6

When the two-component mixture model, described in Section 3.7.2, was fitted to the observed gene means in the "good" response class from the melanoma data without the technical replicates for each of the observations the following estimates for the sixth simulation method were obtained; $\hat{\mu}_o = 0.0182$, $\hat{\tau}_0^2 = 0.0773$, $\hat{\mu}_1 = -0.365$, and $\hat{\tau}_1^2 = 2.058$. These parameter estimates were used to generate 100 simulated datasets as described in Section 3.7.2.

Tables 3.7 and 3.8 show the average LOO $B.632$ bootstrap bias corrected classification results when applying the five statistical gene selection methods (described in Section 3.3) with and without the pre-filter prior to classification construction. The aim of these analyses was to determine whether the classifiers could identify the heterogeneity present in the "poor" response class.

In order to calculate a bias adjusted estimate, a bootstrap sample was generated for each of 100 simulated datasets produced using simulation method 6. The bootstrap classification rate for each method was calculated using the samples in the simulated dataset that were not in the bootstrap sample. The average bootstrap classification rate was then taken across the 100 simulated datasets and used in calculating the $B.632$ bootstrap corrected estimate of the classification rate. For each of the ten methods approximately the same amount of bias was removed from the LOO results (average bias estimate of 0.13 with an approximate 95% confidence interval of 0.11 to 0.14).

For each of the average classification rates a 95% confidence interval (CI) was calculated using a standard approach for calculating a confidence for a single mean. That is $\bar{\theta} \pm t_{df} \times se(\bar{\theta})$, where $\bar{\theta}$ is the average performance statistic, t_{df} is the appropriate quantile from the Student's t distribution and $se(\bar{\theta})$ is the standard error of the average classification rate. This method was also used to calculate 95% confidence intervals for the average number of genes set as differentially expressed, the average number of differentially expressed genes that were retained by the gene selection methods and for the average number of genes used in the classifier. These confidence intervals are displayed in parentheses beneath their associated estimate in Tables 3.7 and 3.8.

These tables show that the $B.632$ bootstrap bias corrected average LOO classification rates for all of the gene selection methods were slightly lower when the gene selection methods were used in combination with the pre-filter than when they were used on their own. These classification rates and their associated 95% confidence intervals indicate that the classifiers were identifing the presence of the two subgroups within the "poor" response class. Tables 3.7 and 3.8 also show that the average number of differentially expressed genes retained by the classification process was slightly lower when the pre-filter was used compared to when it was not used in combination with all of the gene selection methods. Although fewer DE genes were being retained in

the classifier, the predictive ability of the classifier was only slightly reduced when the pre-filter was used in combination with the gene selection methods.

To check the stability of these results a further 400 datasets were simulated. The signal to noise ratio both in combination with the pre-filter and by itself were applied to these extra datasets and the classification results were compared with those obtained using the first 100 simulated datasets. When the pre-filter was not used in combination with the signal to noise ratio the average number of differentially expressed genes that where retained by the gene selection method reduced from 52 (95% CI 51 to 53) to 51 (95% CI 50 to 51) when the number of simulated datasets was increased. The average classification rate and the average number of genes used in the classifier both remained unchanged, but their respective 95% CIs were slightly narrower when 500 datasets were simulated (0.77 to 0.78 for the average classification rate and 17 to 20 for the average number of genes used in the classifier). When the pre-filter was used in combination with the signal to noise ratio the average classification rate increased from 0.68 (95% CI 0.66, 0.69) to 0.69 (95% CI 0.69 to 0.70) and the average number of genes used in the classifier decreased from 19 (95% CI 15 to 22) to 18 (95% CI 17 to 20) when a large number of datasets were simulated. These comparisons showed little to no difference between the two sets of results implying that the results obtained using 100 simulated datasets were reasonably stable.

Table 3.7: Comparison of the average SVM LOO $B.632$ bootstrap bias corrected prediction results obtained when each of the five gene selection methods were applied to 100 simulated data sets generated using simulation method 6. 95% confidence intervals (CI) are provided in brackets.

	t-Test	SAM t-Test	Mann-Whitney	Signal to Noise Ratio	Mod t-Stat
Average Number of Genes Chosen to be DE (95% CI)	496 (492, 501)	496 (492, 501)	496 (492, 501)	499 (494, 503)	496 (492, 501)
Average Number of DE Genes Retained by Selection Methods (95% CI)	51 (50, 52)	45 (44, 46)	43 (42, 44)	52 (51, 53)	50 (49, 51)
Average Number of Genes Retained by Classifier (95% CI)	20 (17, 24)	22 (18, 25)	21 (18, 25)	18 (15, 21)	20 (16, 23)
Average Classification Rate with 95% CI	0.78 (0.76,0.79)	0.77 (0.75,0.78)	0.78 (0.77,0.80)	0.78 (0.76,0.79)	0.78 (0.76,0.79)
Bias Corrected Average Classification Rate	0.63	0.62	0.63	0.63	0.63
Standard Deviation of Classification Rate	0.08	0.09	0.08	0.08	0.09

Table 3.8: Comparison of the average SVM LOO $B.632$ bootstrap bias corrected prediction results obtained when the pre-filter was combined with each of the five gene selection methods and were applied to 100 simulated data sets generated using simulation method 6. 95% confidence intervals (CI) are provided in brackets.

	Combined with:				
	t-Test	SAM t-Test	Mann-Whitney	Signal to Noise Ratio	Mod t-Stat
Average Number of Genes Chosen to be DE	499	499	499	499	499
(95% CI)	(494, 503)	(494, 503)	(494, 503)	(493, 503)	(494, 503)
Average Number of DE Genes Retained by Selection Methods	9	9	8	9	9
(95% CI)	(8, 9)	(8, 9)	(8, 9)	(8, 9)	(8, 10)
Average Number of Genes Retained by Classifier	19	19	17	19	18
(95% CI)	(15, 22)	(16, 22)	(14, 20)	(15, 22)	(15, 21)
Average Classification Rate with 95% CI	0.68 (0.66,0.69)	0.67 (0.66,0.69)	0.68 (0.66,0.70)	0.68 (0.66,0.69)	0.67 (0.66,0.69)
Bias Corrected Average Classification Rate	0.57	0.57	0.57	0.57	0.57
Standard Deviation of Classification Rate	0.08	0.09	0.09	0.08	0.08

Discussion of Performance of Filtering and Pre-Filtering Methods in Simulated Data

The results from the fourth simulation method (Tables 3.5 and 3.6) indicated that when the pre-filter was used in combination with the gene selection methods, on average, fewer differentially expressed genes were retained by the gene selection methods than when the pre-filter was not used. Comparing the average classification rates when the pre-filter was used in combination with the five gene selection methods to those obtained when the pre-filter was not used, a small difference was observed. These results imply that fewer DE genes were being retained by the classifier when the pre-filter was used, and that the predictive performance of the classifier suffered slightly when the pre-filter was used. When the sixth simulation method was used (results in Tables 3.7 and 3.8) these trends were repeated with a similar decrease observed in both the average number of differentially expressed genes retained by the gene selection methods and the average classification rates. The decrease in the average number of DE genes retained by the gene selection methods was not surprising. This is because for both simulation methods four and six, genes were simulated to have a $\log_2$ fold change of approximately zero and the amount that was added to genes that were deemed to be differentially expressed between the two classes was chosen from a normal distribution with a mean of zero. This means that a large proportion of the DE genes wouldn't pass the constraint in the pre-filter which is only interested in the largest fold change values. In practical terms, these results mean that when the pre-filter was used in combination with the five standard filtering methods on simulated data a slight loss in the predictive ability of the constructed classifiers was observed in comparison to the classifiers constructed using the five standard filtering methods. These results also mean that when using the pre-filter a smaller number of highly ranked genes are required to attain approximately the same level of predictive ability observed when the pre-filter was not used. To investigate the performance of these methods in a real microarray data set, the predictor construction process described here was applied to the melanoma data described in Section 3.6.

3.8 Application of the Proposed Methodology to a Melanoma Dataset

To test the proposed filtering methodology on the melanoma data, the technical replication (i.e., the duplicate array for each sample) needed to be considered in relation to both the gene selection and classification stages of the analysis. To test the effect of overlooking the technical replication in this data on the ranking of the genes the moderated t-statistic gene selection method was used, as it has the ability to take into account the technical replication in the data. This was achieved by firstly determining the average correlation within the duplicate arrays using the `duplicateCorrelation()` function in the **limma** package. A linear model was then fitted which placed a constraint on the correlations between the genes within each set of duplicate arrays by using the average correlation calculated by the `duplicateCorrelation()` function. This linear model was created using the `lmFit()` function in the **limma** package. The model also used the number of duplicate arrays for each sample as a blocking factor. This model was then submitted to an empirical Bayes method, using the `eBayes()` function in the **limma** package, in order to moderate the standard deviations between the genes. The final model was then used as an input to the `topTable()` function in the **limma** package. This function calculates, among other values, the moderated t-statistic for each gene in the model, producing a ranked list of genes from most significant to least, based on either the empirical Bayes log odds of differential expression, the adjusted p-value for the moderated t-statistic or the absolute value of the moderated t-statistics. Multiple comparisons are adjusted for based on the false discovery rate.

Comparing the genes that were ranked in the top 100 using the moderated t-statistic (Mod T-Stat) with and without factoring in the technical replication in the data, there was a 63% overlap between the two lists. The majority of the top 100 genes only changed in their rank by one or two places up or down the ranking. A similar pattern was observed when the moderated t-statistic was used in combination with the pre-filter, where there was an overlap of 82% between the two lists of the top 100 ranked genes. Table 3.9 shows the LOO results for the moderated t-statistic gene selection method with and without the combination of the pre-filter and with and without factoring in the technical replication at the gene selection stage. The approximate 95% asymmetric confidence intervals displayed in Table 3.9 were calculated using the `binconf()` function where `p` was either the sensitivity, specificity or classification rate and `n` was the number of samples used in the calculation of `p` (16 for the sensitivity, 13 for the specificity or 29 for the classification rate). As the proportion of samples correctly classified, when taking the technical replication into account and when ignoring it at the gene selection stage, were obtained using the same data, a paired t-test for the difference between two

proportions was conducted. This tested the null hypothesis of there being no difference between the proportion of samples correctly classified when the technical replication was taken into account at the gene selection stage ($\hat{p}_1 = 0.79$ (23/29), Table 3.9) and when the technical replication was ignored at the gene selection stage ($\hat{p}_2 = 0.69$ (20/29), Table 3.9). The t-test statistic t_0 was calculated as the ratio of the difference between the two estimated proportions (i.e., $\hat{p}_1 - \hat{p}_2$) and the standard error of the difference between the two estimated proportions, with observations (either correctly or incorrectly classified when the technical replication was accounted for or ignored) arising from the same sample in the melanoma dataset treated as pairs. The standard error is derived from the information obtained by constructing a two-way table of counts comparing the number of observations correctly classified when the technical replication was taken into account at the gene selection stage to those correctly classified when the technical replication was ignored at the gene selection stage (Table 3.9). Wild and Seber (1993) provide some of the background theory that uses this information to produce the formula for the standard error, as given below. Wild and Seber (1993) state that by using this standard error formula the resulting t-statistic is the square root of that obtained using the non-parametric McNemar test (McNemar, 1947) for paired data. Thus the standard error is given by,

$$se(\hat{p}_1 - \hat{p}_2) = \sqrt{\frac{(p_{12} + p_{21}) - (\hat{p}_1 - \hat{p}_2)^2}{n}}$$

where p_{12} is the proportion of samples that where correctly classified when the technical replication was taken into account at the gene selection stage but incorrectly classified when the technical replication was ignored at the gene selection stage. p_{21} is the proportion of samples that were correctly classified when the technical replication was ignored at the gene selection stage but incorrectly classified when the technical replication was taken into account at the gene selection stage, and $n = 29$ (the sample size). This test gave a test statistic of $t_0 = 1.8292$ and a p-value of 0.0674 when applied to the results obtained when the moderated t-statistic was used at the gene selection stage. A similar test was carried out when the pre-filter was used in combination with the moderated t-statistic, this gave a test statistic of $t_0 = 0$ and a p-value of 1. As neither of these were significant at the 5% level, there was little or no evidence to suggest that there was a difference between the classification rates when the technical replication in the data was overlooked at the gene selection step, and when the technical replication was taken into account for this dataset. As more interest lay in the general applicability of the proposed pre-filter method, it was felt that the technical replication in the data could be overlooked in the gene selection step, whilst being taken into account in the classification step. This was achieved by using a combination of the class probabilities

for each replicate to assign an observation to either of the two classes, as described in Section 3.5.

To test if changing the parameters of the pre-filter (i.e., α and the range/Fold Change cut-off) had any effect on the classification rate when the pre-filter was used in combination with the Mann-Whitney gene selection method, eight analyses were conducted with the range/Fold Change cut-off and α set at different values. The first set of analyses fixed the range at -1 to 1 (i.e., Fold Change cut-off equals 2) and varied α. The second set of analyses fixed α at 50% and varied the range/Fold Change cut-off. Figure 3.4 shows the classification rates along with approximate 95% confidence intervals when (a) α is varied for a fixed range of -1 to 1 and when (b) the range is varied for a fixed α of 50%. The approximate 95% asymmetric confidence intervals were calculated using the `binconf()` function with `n` = 29. As both of these plots show the highest classification rate is obtained when using $\alpha = 50\%$ and a range of -1 to 1 (i.e., Fold Change greater than 2), it was decided to continue using these parameter values.

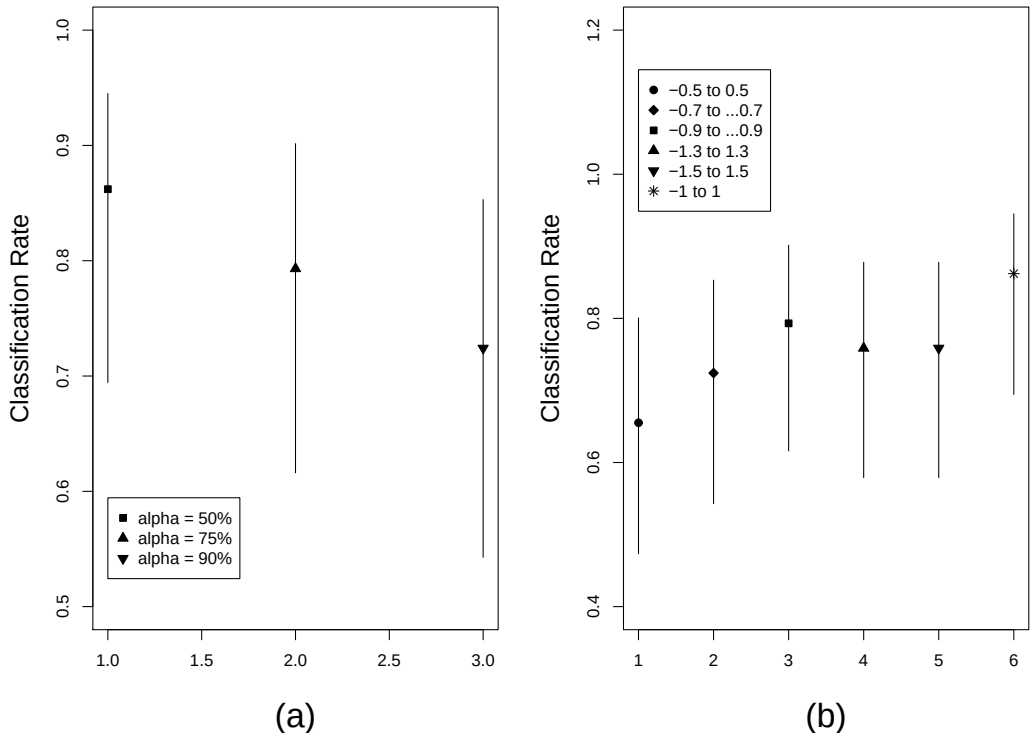

Figure 3.4: Comparison of the classification rates with approximate 95% asymmetric confidence intervals obtained when the parameters in the pre-filter were changed. Plot (a) shows the effect of varing α for a fixed range of -1 to 1 and plot (b) shows the effect when the range is varied for a fixed α of 50%.

Applying the Linear SVM classifier and the LOO method described in Section 3.5 to the melanoma data in combination with the five statistical gene selection methods described in Section 3.3, and then with these methods combined with the pre-filter,

Table 3.9: Comparing whether the SVM LOO classification was correct (Yes, below) when the moderated t-statistic was used to account for the technical replication in the melanoma data at the gene selection stage, compared to those obtained when the technical replication was overlooked at the gene selection stage. The comparison was also considered with the combination of the moderated t-statistic and the pre-filter. Approximate 95% asymmetric confidence intervals are included in brackets for the proportion correctly classified by each method.

		Moderated T-Statistic					Moderated T-Statistic Combined with Pre-Filter		
		Without Technical Replicates					Without Technical Replicates		
		Yes	No				Yes	No	
With Technical Replicates	Yes	20	3	23 (0.62,0.90)	With Technical Replicates	Yes	24	1	25 (0.69,0.95)
	No	0	6	6		No	1	3	4
		20 (0.52,0.83)	9	29			25 (0.69,0.95)	4	29

yielded the performance results in Tables 3.10 and 3.11. The approximate 95% asymmetric confidence intervals displayed in these tables were obtained using the `binconf()` function with `n` = 16, 13 and 29 for the sensitivity, specificity and classification rates observed.

Tables 3.10 and 3.11 also include the number of genes used in the chosen model and the permutation p-value. 500 different permutations of the dataset were generated, upon which classification models using the LOO classification process were constructed. The permutation p-value gives the proportion of times that the classification rate from these 500 models was greater than that observed in the chosen model, and thus provides an indication of the predictive reliability of the chosen models. As can be seen in Table 3.10, when the pre-filter was not used in combination with the five gene selection methods all of the permutation p-values were large. This implies that it was possible to find a better classifier by simply permuting the class labels and that none of the gene selection methods were able to provide a list of genes that were useful in a classifier. However, when the pre-filter was used in combination with these gene selection methods, 4 out of 5 of these permutation p-values became highly significant.

As these results still contain an amount of selection bias due to the model selection method used in the classification procedure employed, an extra leave-one out step was added to the LOO method creating a Double-Leave-One-Out (DLOO) method. As this proved to be overly computationally intensive it was only applied to three of the five statistical gene selection methods both in combination with the pre-filter and without the pre-filter. These three methods were chosen as they represent three standard approaches to gene selection (two parametric methods, of which one is tailored for gene expression data, and one non-parametric method) and they were all computationally quick to evaluate.

From Tables 3.12 and 3.13, it is clear that there was a marked improvement in the performance statistics when the t-test and Mann-Whitney gene selection methods were combined with the pre-filter and a slight improvement in the performance statistics when the SAM adjusted t-test gene selection method was combined with the pre-filter. The low classification rates observed when the t-test and Mann-Whitney gene selection methods were implemented without the use of the pre-filter (Table 3.12) imply that a large amount of over-fitting in the internal LOO step of the analysis was occurring. This could indicate that within this dataset there were a large number of genes that displayed a low level of fold change and could be used to discriminate between the two classes within the training data but that this ability did not generalise to the test data. The approximate 95% asymmetric confidence intervals displayed in Tables 3.12 and 3.13 were obtained using the `binconf()` function with `n` = 16, 13 and 29 for the sensitivity, specificity and classification rates observed. The number of genes retained

by the classifiers in the internal LOO were not kept.

A less computationally intensive method of correcting for the selection bias was to apply the $B.632$ bootstrap bias correction method to the LOO results. Tables 3.14 and 3.15 contain the bias adjusted sensitivity, specificity and classification rates for each of the five statistical gene selection methods with and without the use of the pre-filter. From Table 3.15 it is clear that when the t-test, Signal-to-Noise Ratio and Mann-Whitney gene selection methods are combined with the pre-filter they perform slightly better than when used on their own (Table 3.14). That is, the 18 gene predictor using the pre-filter method with the t-test as the gene selection method produced a classification rate of 64% with an approximate 95% confidence interval of 46% to 82% (sens = 71% & spec = 60%) compared with the classification rate of 49% with an approximate 95% confidence interval of 30% to 68%, of the 68 gene predictor using only the t-test as the gene selection method. These results imply that when the pre-filter was used a slight improvement in the predictive ability of the classifier could be achieved for this dataset. The approximate 95% asymmetric confidence intervals displayed in Tables 3.14 and 3.15 were obtained using the `binconf()` function with `n` = 16, 13 and 29 for the sensitivity, specificity and classification rates observed.

As a comparison to the LOO bootstrap bias corrected models a more simplistic classification model was used that involved training the model on all of the data, calculating the resubstitution/apparent performance statistics and then applying the $B.632$ bootstrap bias correction method. The performance statistics when the pre-filter was combined with the standard gene selection methods showed very little change to the performance statistics obtained when the pre-filter was not used. The similarity observed between these two sets of results could be due to the over-fitting that was encountered by using all of the dataset to train and test the classifiers. This could also be the reason why the improvement in the classifiers observed in the DLOO and LOO methods were not seen in this method. The results from this method are presented in Tables 3.16 and 3.17 with approximate 95% asymmetric confidence intervals obtained using the `binconf()` function with `n` = 16, 13 and 29 for the bias adjusted sensitivity, specificity and classification rates obtained. Comparing the bias corrected results obtained when using LOO (Tables 3.14 and 3.15) to those obtained using the more simplistic classification model (Tables 3.16 and 3.17), the bias corrected classification rates obtained when using LOO without the pre-filter (Table 3.14) are considerably lower than when the simplistic classification model was used (Table 3.16). This trend was also observed, but to a lesser extent, when the bias adjusted classification rates obtained when the pre-filter was used were compared between the two classification construction methods (Tables 3.15 and 3.17). The discrepancy between these results could be due to the way the classifiers were constructed, the amount of bias and over-fitting that was incurred

during the construction of the classifiers, and the $B.632$ bias correction only accounting for the bias that was introduced by choosing the classifier with the smallest number of genes that had the largest classification rate in the "test" data.

The results from this set of analyses imply that by using the pre-filter in combination with the gene selection methods the predictive performance of the classifier was not hindered. In the cases where the pre-filter was used in combination with the t-test and Mann-Whitney gene selection methods two of the four sets of analyses showed a significant improvement in the average classification rates over those obtained when the pre-filter was not used (Tables 3.10, 3.11, 3.12 and 3.13).

Table 3.10: Comparison of the SVM LOO prediction results without bias corrections obtained when each of the five statistical gene selection methods were applied to the Melanoma data. Approximate 95% asymmetric confidence intervals are included in brackets.

	t-Test	SAM t-Test	Mann-Whitney	Signal to Noise Ratio	Mod t-Stat
Number of genes Retained	68	7	53	17	7
Sensitivity	0.63	0.81	0.63	0.69	0.81
(95% Confidence Interval)	(0.39,0.82)	(0.57,0.93)	(0.39,0.82)	(0.45,0.86)	(0.57,0.93)
Specificity	0.38	0.77	0.38	0.31	0.77
(95% Confidence Interval)	(0.17,0.64)	(0.50,0.92)	(0.17,0.64)	(0.13,0.57)	(0.50,0.92)
Classification Rate	0.52	0.79	0.52	0.52	0.79
(95% Confidence Interval)	(0.35,0.69)	(0.61,0.90)	(0.35,0.69)	(0.35,0.69)	(0.61,0.90)
permutation p-value	0.86 (430/500)	0.12 (60/500)	0.83 (415/500)	0.824 (412/500)	0.118 (59/500)

Table 3.11: Comparison of the SVM LOO prediction results without bias corrections obtained when the pre-filter was used in combination with each of the five gene selection methods and were applied to the Melanoma data. Approximate 95% asymmetric confidence intervals are included in brackets.

	Combined with:				
	t-Test	SAM t-Test	Mann-Whitney	Signal to Noise Ratio	Mod t-Stat
Number of genes Retained	18	50	23	23	41
Sensitivity	0.88	0.81	0.88	0.88	0.88
(95% Confidence Interval)	(0.65,0.97)	(0.57,0.93)	(0.65,0.97)	(0.65,0.97)	(0.65,0.97)
Specificity	0.85	0.77	0.85	0.77	0.85
(95% Confidence Interval)	(0.58,0.96)	(0.50,0.92)	(0.58,0.96)	(0.50,0.92)	(0.58,0.96)
Classification Rate	0.86	0.79	0.86	0.83	0.86
(95% Confidence Interval)	(0.69,0.94)	(0.61,0.90)	(0.69,0.94)	(0.66,0.93)	(0.69,0.94)
permutation p-value	0.008 (4/500)	0.102 (51/500)	0 (0/500)	0.06 (30/500)	0.012 (6/500)

Table 3.12: Comparison of the SVM DLOO prediction results obtained when three of the gene selection methods were applied to the Melanoma data. Approximate 95% asymmetric confidence intervals are included in brackets bellow each result.

	t-Test Filter	SAM t-Test	Mann-Whitney
Sensitivity	0.44 (7/16)	0.69 (11/16)	0.38 (6/16)
(95% Confidence Interval)	(0.23,0.67)	(0.44,0.85)	(0.18,0.61)
Specificity	0.15 (2/13)	0.31 (4/13)	0.15 (2/13)
(95% Confidence Interval)	(0.04,0.42)	(0.13,0.58)	(0.04,0.42)
Classification Rate	0.31	0.52	0.28
(95% Confidence Interval)	(0.17,0.49)	(0.34,0.69)	(0.15,0.46)

Table 3.13: Comparison of the SVM DLOO prediction results obtained when the Pre-Filter was used in combination with each of the three chosen gene selection methods and were applied to the Melanoma data. Approximate 95% asymmetric confidence intervals are included in brackets bellow each result.

	t-Test	SAM t-Test	Mann-Whitney
Sensitivity	0.81 (13/16)	0.69 (11/16)	0.75 (12/16)
(95% Confidence Interval)	(0.57,0.93)	(0.44,0.86)	(0.51,0.90)
Specificity	0.62 (8/13)	0.46 (6/13)	0.77 (10/13)
(95% Confidence Interval)	(0.36,0.82)	(0.23,0.71)	(0.50,92)
Classification Rate	0.72	0.59	0.76
(95% Confidence Interval)	(0.54,0.85)	(0.41,0.74)	(0.58,0.88)

Table 3.14: Comparison of the SVM LOO prediction results with $B.632$ bootstrap bias corrections obtained when each of the five gene selection methods were applied to the Melanoma data. Approximate 95% asymmetric confidence intervals are included in brackets.

	t-Test	SAM t-Test	Mann-Whitney	Signal to Noise Ratio	Mod t-Stat
Number of genes					
Retained	68	7	53	17	7
Sensitivity	0.58	0.65	0.58	0.60	0.66
(95% Confidence Interval)	(0.35,0.78)	(0.41,0.83)	(0.35,0.78)	(0.36,0.80)	(0.42,0.84)
Specificity	0.42	0.56	0.42	0.39	0.56
(95% Confidence Interval)	(0.20,0.67)	(0.31,0.78)	(0.20,0.67)	(0.18,0.65)	(0.31,0.78)
Classification Rate	0.49	0.59	0.49	0.49	0.60
(95% Confidence Interval)	(0.32,0.66)	(0.41,0.75)	(0.32,0.66)	(0.32,0.66)	(0.42,0.76)

Table 3.15: Comparison of the SVM LOO prediction results with $B.632$ bootstrap bias corrections obtained when the pre-filter was used in combination with each of the five gene selection methods and were applied to the Melanoma data. Approximate 95% asymmetric confidence intervals for the classification rates are included in brackets.

	Combined with:				
	t-Test	SAM t-Test	Mann-Whitney	Signal to Noise Ratio	Mod t-Stat
Number of genes Retained	18	50	23	23	41
Sensitivity	0.71	0.68	0.70	0.71	0.70
(95% Confidence Interval)	(0.47,0.87)	(0.44,0.85)	(0.46,0.87)	(0.47,0.87)	(0.46,0.87)
Specificity	0.60	0.56	0.59	0.57	0.60
(95% Confidence Interval)	(0.34,0.81)	(0.31,0.78)	(0.33,0.81)	(0.32,0.79)	(0.34,0.81)
Classification Rate	0.64	0.61	0.64	0.63	0.64
(95% Confidence Interval)	(0.46,0.79)	(0.43,0.76)	(0.46,0.79)	(0.45,0.78)	(0.46,0.79)

Table 3.16: Comparison of the SVM apparent prediction results with $B.632$ bootstrap bias corrections obtained when each of the five gene selection methods were applied to the Melanoma data. Approximate 95% asymmetric confidence intervals are included in brackets.

	t-Test	SAM t-Test	Mann-Whitney	Signal to Noise Ratio	Mod t-Stat
Number of genes Retained	6	6	16	27	5
Sensitivity	0.72	0.72	0.72	0.72	0.72
(95% Confidence Interval)	(0.48,0.88)	(0.48,0.88)	(0.48,0.88)	(0.48,0.88)	(0.48,0.88)
Specificity	0.65	0.67	0.66	0.66	0.66
(95% Confidence Interval)	(0.39,0.85)	(0.40,0.86)	(0.39,0.85)	(0.39,0.85)	(0.39,0.85)
Classification Rate	0.67	0.68	0.67	0.68	0.68
(95% Confidence Interval)	(0.49,0.81)	(0.50,0.82)	(0.48,0.81)	(0.50,0.82)	(0.50,0.82)

Table 3.17: Comparison of the SVM apparent prediction results with bootstrap bias corrections obtained when the pre-filter was used in combination with each of the five gene selection methods and were applied to the Melanoma data. Approximate 95% asymmetric confidence intervals are included in brackets.

	t-Test	SAM t-Test	Mann-Whitney	Signal to Noise Ratio	Mod t-Stat
Number of genes					
Retained	12	9	10	23	9
Sensitivity	0.75	0.73	0.74	0.75	0.73
(95% Confidence Interval)	(0.51,0.90)	(0.49,0.89)	(0.50,0.89)	(0.51,0.90)	(0.49,0.89)
Specificity	0.66	0.69	0.65	0.67	0.68
(95% Confidence Interval)	(0.39,0.85)	(0.42,0.87)	(0.39,0.85)	(0.40,0.86)	(0.41,0.87)
Classification Rate	0.69	0.70	0.69	0.70	0.69
(95% Confidence Interval)	(0.51,0.83)	(0.52,0.84)	(0.51,0.83)	(0.52,0.84)	(0.51,0.83)

3.9 Discussion of the Performance of the Methodology when Applied to both Simulated Data and the Melanoma Data

When the pre-filter was used in combination with the five standard gene selection methods on the simulated datasets produced using the fourth and sixth simulation methods, the average classification rates were slightly lower than when the pre-filter was not used (Tables 3.5, 3.6, 3.7 and 3.8). However when the pre-filter was used in combination with the five standard gene selection methods on the melanoma data the average classification rates were seen to be higher for most of the gene selection methods than those obtained when the pre-filter was not used. A high level of overlap was observed when comparing the 95% asymmetric confidence intervals for the average classification rates obtained when the pre-filter was used in combination with the majority of the gene selection methods on the melanoma data, to those derived when the pre-filter was not used on the melanoma data. The exceptions to this were when the pre-filter was used in combination with the t-test and Mann-Whitney gene selection methods, where the predictive performance (i.e., the classification rate) of the SVM classifier showed a significant improvement over that obtained when the pre-filter was not used (Tables 3.10, 3.11, 3.12 and 3.13). This observed increase in the predictive performance implies that when the pre-filter was used in combination with the t-test and Mann-Whitney gene selection methods, they each produced highly ranked genes that were more suitable for use in a linear SVM classifier, than when the other gene selection methods were used in combination with the pre-filter.

When the methodology was applied to the melanoma data the number of genes retained by the classifiers ranged from between 9 to 50 genes, over the 5 gene selection methods, when the pre-filter was used (Tables 3.11, 3.15 and 3.17) and from between 6 to 68 genes, over the 5 gene selection methods, when the pre-filter was not used (Tables 3.11, 3.14 and 3.16). Comparing these to the average number of genes retained by the classifiers when using the simulated data (ranging from between 17 to 19 when the pre-filter was used (Tables 3.6 and 3.8) and between 18 to 31 when the pre-filter was not used (Tables 3.5 and 3.7)) a larger degree of variability in the number of genes retained by the classifiers was observed when the methodology was applied to the melanoma data.

The discrepancy between the simulation results and the real data example is most likely due to the way the simulated data sets were created. In particular, for simulation methods four and six, genes were simulated to have a $\log_2$ fold change of approximately zero and the amount that was added to genes that were deemed to be differentially expressed between the two classes was chosen from a normal distribution with a mean of

zero. This means that a large proportion of the DE genes wouldn't pass the constraint in the pre-filter which is only interested in the largest fold change values. Another contribution to the discrepancy between the simulation results and those obtained using the melanoma data could be due to the technical replicates within the melanoma data, even though it was shown that these replicates could be over-looked at the gene selection stage, thus giving rise to a structural difference between the two types of data. In additional to this structural difference, the correlation structure that was present in the real data was not built into the simulated data. However, as the pre-filter was designed to be applied in a per-gene basis the correlation structure of the dataset is not taken into account and should not be a cause for the differences observed between the results obtained from the simulated and real datasets.

The results obtained from the analysis of both the simulated and melanoma data imply that when the pre-filter was used in combination with the gene selection methods the classification rates, produced from a SVM classifier, were at least as good as the classification rates produced when the pre-filter is not used in combination with the gene selection methods.

The main reason for using the pre-filter was to remove genes that were deemed not to have undergone a biologically relevant level of differential expression and could therefore not be used in a diagnostic test that requires genes to have a reasonable-sized fold change in order for it to succeed (e.g., polymerase chain reaction based diagnostic tests). The results from the analyses undertaken in this chapter have shown that when this constraint was placed on the analysis, via the use of the pre-filter, the predictive performance of the classifier was not detrimentally impacted.

4

Incorporating Functional Information into the Classification Process

4.1 Motivation

The incorporation of biological information into the microarray analysis process has become increasingly important. This is partly due to the realisation that, under the assumption that major changes in biological mechanisms are associated with poor outcomes, groups of co-regulated genes that perform specific and important biological functions need not necessarily be highly ranked on a per-gene basis when using standard approaches to detecting differences in gene expression levels (Leung and Cavalieri (2003), Barry et al. (2005), Shi and Walker (2007)). It is also highly desirable to be able to provide a biologically meaningful interpretation of the analysis results, rather than simply a list of gene names that may be unrelated in terms of biological function (Pham et al., 2006).

The incorporation of such information is well documented in terms of detecting differentially expressed genes (e.g., Goeman et al. (2004), Curtis et al. (2005), Barry et al. (2005), and Mootha et al. (2003)). Known as "gene-set" or "pathway analysis" methods, the main goal of these techniques is to detect significant changes in expression for groups of functionally related genes. In general these methods use pre-defined gene-sets, with pre-existing biological knowledge used to place genes into groups representing a common function to calculate a statistic which assesses the difference in expression between experimental conditions for each pre-defined gene-set (Curtis et al., 2005). Two popular sources of such functional information are The Kyoto Encyclopedia of Genes and Genomes (KEGG) (Kanehisa and Goto, 2000) and the Gene Ontology project (GO)

(Ashbuner et al., 2000). As described in Section 2.3, KEGG and GO are bioinformatic databases containing information on the biological, functional and molecular properties of genetic data.

In 2003, Todd Golub and colleagues published what can now be seen as a breakthrough paper in the area of gene set analysis (Mootha et al., 2003). Their statistical methodology, entitled Gene Set Enrichment Analysis (GSEA), was the first to explicitly use pre-defined gene-sets to detect significant changes in expression for groups of functionally related genes. Following this trend, and adding to the growing collection of gene-set analysis methods, Barry et al. (2005) proposed Significance Analysis of Function and Expression (SAFE), which follows a similar process to that of GSEA. A method for incorporating correlation into the analysis process was proposed by Goeman et al. (2004), who made use of the covariance structures of the gene-sets being studied and the clinical outcome of interest, in the formulation of their gene-set analysis method known as global test. More recently another gene-set analysis method has been proposed by Kong et al. (2006) and Song (2009). A software implementation of this is available from Bioconductor, entitled PCOT2 (Song and Black, 2007), this software incorporates inter-gene correlation into a method that aims to identify gene-sets that show significant differences in their expression levels between two classes. The aim of this chapter is to extend these ideas into the classification of biologically distinct samples, by incorporating sets of functionally related genes directly into the classification process. The approach developed in this chapter utilises principal co-ordinates analysis (PCO) to create a summary of the activity levels within a set of functionally related genes. These summaries are then used as explanatory variables in the classification and prediction process. This procedure is illustrated via application to three publicly available breast cancer data sets.

4.2 Gene Set Analysis - Overview

Curtis et al. (2005) provide a review of several existing gene-set/pathway analysis methods and a comparison of three commonly used methods (The binomial distribution, z scores and GSEA) in terms of their performance on two unpublished mouse gene expression data sets from two different array platforms, spotted microarrays and Affymetrix microarrays. As described in Section 2.1.2, the z-score (or standardised difference score) is a "hit-counting" method based on the hyper-geometric distribution which determines the enrichment (i.e., the amount of significant over-representation) of a given pathway or gene-set in a list of genes. The binomial distribution method calculates the probability of there being a specific number of genes from a given pathway in the gene list of interest. They concluded that the results using the binomial distribution and z-scores

were similar and that there was some overlap between these two methods and GSEA. The gene-set analysis methods described by Curtis et al. (2005) work on a per-gene basis, that is, these methods compare each gene in a given list of ranked genes to the genes in a specific pathway or functional classification, to determine if there are a greater number of matches between the two lists than would be expected by chance (Curtis et al., 2005). A alternative approach to gene-set analysis involves the use of correlation. Two such methods are the global test proposed by Goeman et al. (2004) and PCOT2 (Song and Black, 2007). The following sections describe these two correlation based gene-set analysis methods and the per-gene approaches of GSEA (Mootha et al., 2003) and SAFE (Barry et al., 2005).

4.2.1 Per-gene methods: GSEA and SAFE

The GSEA algorithm was designed to determine if pre-defined sets of genes are differentially expressed in different phenotypes (Shi and Walker, 2007). The concept behind the GSEA algorithm has been described by Segal et al. (2003), who provide an algorithm that integrates gene expression data with DNA sequence data using a unified probabilistic graphical model. The goal of this algorithm was to identify sets of genes (i.e., modules) that were co-regulated within a set of experiments, through a common sequence motif profile or combination of sequence motif profiles from the DNA sequence data. A sequence motif is a sequence of nucleotides or amino-acids that is repeatably observed in different DNA sequences and is believed to be of biological significance. This algorithm, which the GSEA algorithm is based on, allows genes to be re-assigned to different modules with similar patterns, and the addition or removal of motifs simultaneously. Segal et al. (2003) evaluated their algorithm using two gene expression data sets and it was shown to have an advantage over standard attempts to relate sequence motifs and gene expression data.

In a practical application studying the cyclin D1 oncogene Lamb et al. (2003) used the concept behind the GSEA algorithm to determine if the expression patterns of an experimentally defined set of target genes that were induced/activated by both the normal and a mutated cyclin D1 gene, were significantly correlated with the levels of cyclin D1 in fourteen different human tumours taken from a database of gene expression profiles.

Shi and Walker (2007) provide an overview of the original GSEA algorithm, a selection of extensions and alterations of the algorithm as well as discussing some limitations of the method. The original algorithm, proposed by Mootha et al. (2003), was designed for the situation of a two-category outcome, and is implemented in a two step process. Further extensions to the algorithm now allow the clinical variable to be continuous (Shi and Walker, 2007). In the first step of the algorithm the individual genes are

ranked using an association score (e.g., t-test, Signal-to-Noise Ratio or Wilcoxon test) to generate a list of genes ranked by their association with the outcome of interest, then in the second step the ranks of the genes in a predefined gene set are compared to the ranks of the genes not in the given gene set, to give an "enrichment score". It is assumed that if a gene set is related to the outcome of interest or phenotype, then the genes in that gene set will tend to have higher association scores than genes in gene-sets that are not related to the phenotype, and so the enrichment score will be larger. The Kolmogorov-Smirnov statistic (Massey, 1951) is used to calculate the enrichment score. For multiple testing the original algorithm allowed the permutation p-values to be adjusted by the Bonferroni method to control the Family Wise Error Rate (Subramanian et al., 2005).

Subramanian et al. (2005) proposed a number of modifications to the original GSEA algorithm, the most notable of these being the introduction of a weighting factor that reduced the enrichment score (ES) for gene-sets showing a high level of enrichment near the middle of the list of genes which had been ranked by their association with the outcome of interest. Shi and Walker (2007) point out that, based on the assumption that gene-sets of real interest would have the largest differences between the two categories in the outcome of interest, the reason for incorporating such a weighting factor was because the original enrichment score had the undesirable characteristic of finding gene-sets whose genes differed greatly from a uniform distribution regardless of whether this happened at the top or the middle of the ranked list of genes. Shi and Walker (2007) also state that some of the other modifications that Subramanian and colleagues proposed included; the option of using the False Discovery Rate (FDR) as a method for adjusting the permutation p-values for multiple comparisons, a way of identifying which members of a gene-set contributed the most to the enrichment score and that they also included a database of pre-defined gene-sets. Shi and Walker (2007) describe three other extensions to GSEA, including the Absolute Enrichment (AE) algorithm of Saxena et al. (2006), which ranks the genes by the absolute difference between the expression levels of each gene in the two categories of the outcome of interest, so that gene-sets that have some genes significantly up regulated and some genes significantly down regulated in response to a given perturbation can be detected.

The Parametric Analysis of Gene Set Enrichment (PAGE) proposed by Kim and Volsky (2005), who suggest replacing the permutation test used to calculate the p-values with the expected normal distribution of the average score of the genes in the gene-set, based on the Central Limit Theorem. However, as pointed out by Shi and Walker (2007) this may not be appropriate as the number of genes in a gene-set required to give an adequate convergence to the normal distribution is in most cases larger than the number of genes known to belong to a given gene-set, and furthermore, the permutation test is

generally more powerful than the normal approximation for non-normally distributed populations, like that of gene expression data. Shi and Walker (2007) also describe some of the extensions Jiang and Gentleman (2007) offer. These include the ability to take into account other factors or covariates that can affect gene expression values, via the incorporation of linear regression into the GSEA algorithm. They also included several alternative measures of association and provide a way to analyse genes that have correlated expression values in the context of GSEA. Some of the limitations of GSEA that Shi and Walker (2007) point out include the need to ensure that there are no less than 8-10 genes per gene-set, no less than 8 samples in each of the two categories of the outcome of interest and that the expression values of each gene are standardised before GSEA can be used with any degree of confidence in the resulting analysis.

Significance analysis of function and expression (SAFE) proposed by Barry et al. (2005) is based on a similar two step process as GSEA. In the first step a "local" statistic is calculated that measures the association between the expression profile of a given gene and the response. By default the absolute value of the t-statistic is used, then in the second step a "global" statistic is calculated which assesses how the distribution of the local statistics within a category differs from local statistics outside that category. The global statistic can be calculated using either the Wilcoxon Rank Sum statistic or the Kolmogorov-Smirnov statistic. To assess the significance of the global statistic the values of the clinical outcome of interest (i.e., the response variable) are permuted, to reflect the underlying design of the experiment, and both the local and global statistics are recalculated. The p-value is then calculated as the proportion of permuted global statistics that are greater than or equal to the observed global statistic. This value is then corrected for multiple comparisons using either the family wise error rate or the false discovery rate. By performing this permutation the correlation structure both within and across the levels of the clinical outcome of interest is preserved and taken into consideration. For easy implementation Barry et al. (2005) have incorporated this algorithm into a publicly available **R** package named **safe** which can be downloaded from the Bioconductor webpage (http://bioconductor.org/packages/release/bioc/).

4.2.2 Correlation-based Methods: PCOT2 and Global Test

As pointed out by Leung and Cavalieri (2003) it has been known for some time now that genes do not act alone in a biological system, thus a gene that is highly expressed will have several biologically related genes that are also highly expressed. This implies that genes that are seen to be correlated with each other may be biologically related. Exploiting this inter-gene correlation provides an alternative perspective to the standard approach of identifying sets of genes that are differentially expressed in different phenotypes. Two such methods that make use of the inter-gene correlation are the

global test proposed by Goeman et al. (2004), and PCOT2 proposed by Kong et al. (2006) and Song and Black (2007). The global test (Goeman et al., 2004) achieves this by calculating a test statistic that reflects the correlation between the covariance matrix of the gene-expression profiles between the samples, and the covariance matrix of the clinical outcome of interest between the samples. A large test statistic implies that these two matrices show a higher degree of correlation than if the test statistic were small. The global test statistic is based on the empirical Bayes generalised linear model and so allows for the clinical outcome to be either a continuous measure (e.g., Survival time) or a discrete class label (e.g., recurrent and non-recurrent classes). Global test was developed to answer the question, "is the global expression pattern of a group of genes significantly related to the clinical outcome of interest?" (Goeman et al., 2004).

PCOT2 (Song and Black, 2007) is a two-stage permutation-based method that utilises inter-gene correlation information in order to detect significant differences in the activity levels of predefined gene-sets between two phenotypes (e.g., recurrent and non-recurrent classes, or patients with two different sub-types of a particular cancer). The first stage consists of reducing the dimensionality of the gene-set(s) via principal co-ordinates analysis, keeping the first two principal coordinates by default, while in the second stage the Hotelling's T^2 statistic is calculated for each of the reduced gene-sets and the associated permutation p-value is calculated by permuting the class labels of the two classes. These p-values are adjusted for multiple comparisons using the False Discovery Rate of Benjamini and Yekutieli (2001). Song (2009) compares PCOT2 to global test and concludes that there is very little difference between the results for the two models in the case where the clinical outcome is a discrete two-level class label.

4.2.3 Defining Gene-Sets

Gene-sets can be defined in a number of ways, one way of defining a gene-set is as a group of genes that are either functionally or biologically related. Two examples of gene-sets following this definition are GO categories defined by the Gene Ontology project (Ashbuner et al., 2000) and KEGG pathways as defined in the Kyoto Encyclopedia of Genes and Genomes (Kanehisa and Goto, 2000). As Curtis et al. (2005) state, the formation of GO categories, or GO Terms, is based on the hierarchical classification of genes with similar functions. Because of this hierarchical structure a gene that belongs to a particular GO Term will automatically belong to all of that GO Term's parent Terms in the hierarchical structure. This means that genes that are involved in, for example, apoptosis (programmed cell death), are also involved in cell death, one of its parent terms, and so on up the hierarchy, as represented in Figure 4.1. The Kyoto Encyclopedia of Genes and Genomes uses molecular interactions and reactions to group genes into pathways describing metabolic cellular processes and human diseases so that,

for example, genes that interact with one another at a molecular level to affect the rate of apoptosis in a biological system are grouped together to form a pathway. These pathways are represented graphically: Figure 4.2 shows the graphical representation of the apoptosis pathway.

Another way of defining gene-sets is to construct gene-networks, which are groups of genes that are linked to one another based on some measure of similarity or association. Luo et al. (2007) describe a method for creating co-expression networks using a two step process where the correlation matrix of the gene expression data matrix is first calculated and then pairs of genes are linked together if the correlation between their expression values is greater that a pre-determined threshold. Along those lines, Song (2009) has explored an alternative method for defining gene-sets based on using clusters from a hierarchical cluster analysis on the correlation matrix of gene expression data, so that a gene-set consists of genes that are highly correlated with one another. This method works by firstly removing any genes whose expression values are not correlated above a pre-defined threshold with any of the other genes in the dataset. The retained genes are then submitted to a hierarchical clustering algorithm which uses the correlation matrix of the retained genes as the similarity matrix on which to form the clusters. Once the hierarchical clustering is complete the results can be represented by a dendrogram or tree, where the height of the branches are proportional to the correlation level at which the clusters were formed and consequently gives a measure of the correlation of the genes in the clusters. Within the **pcot2** Bioconductor package, it is possible to graphically visualise the dendrogram and the correlation matrix together before the dendrogram is cut, as a guide to determining the number of clusters or height at which to cut the dendrogram.

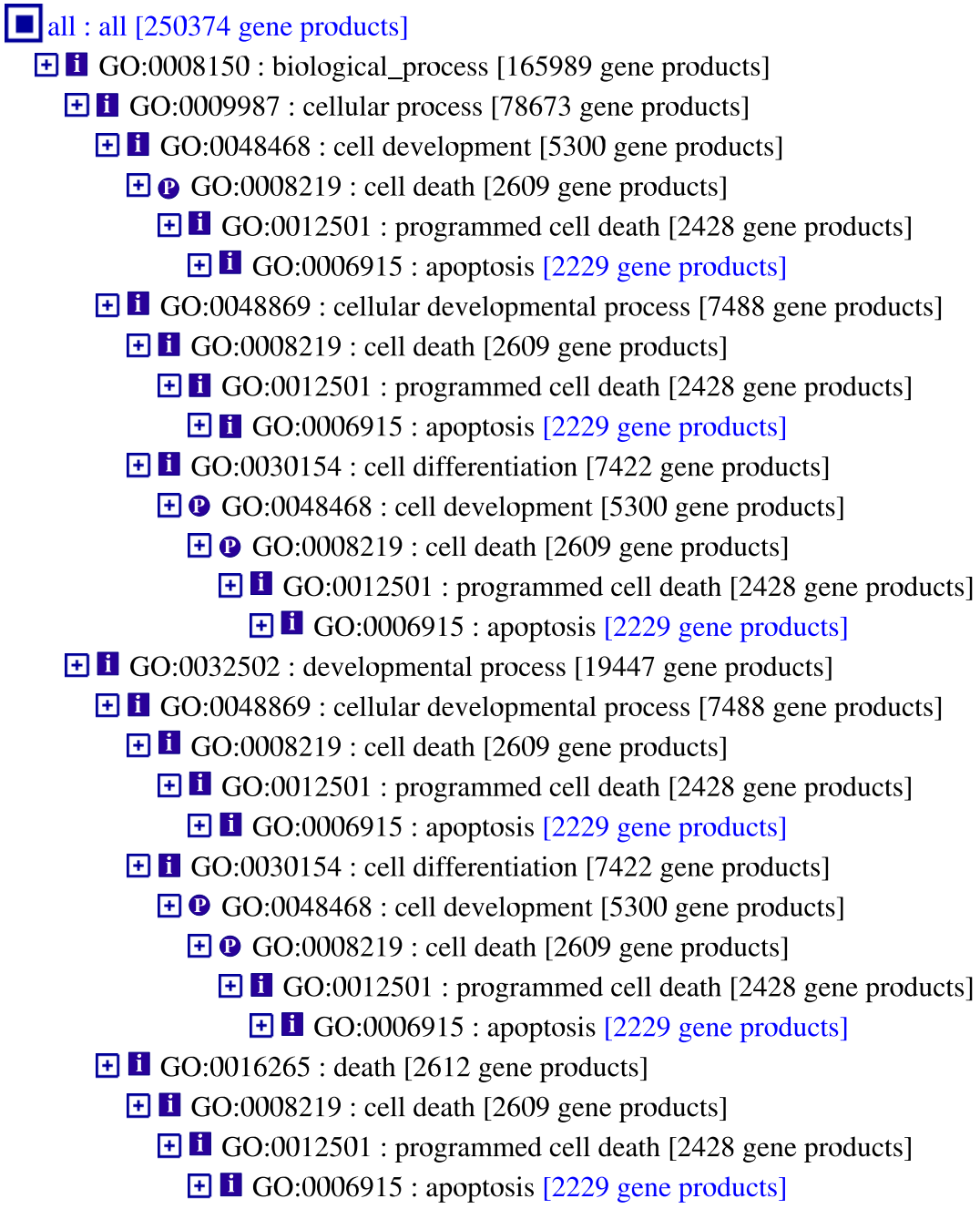

Figure 4.1: The GO hierarchical Structure for apoptosis. (Ashbuner et al. (2000), Release Date: 2008-05-15)

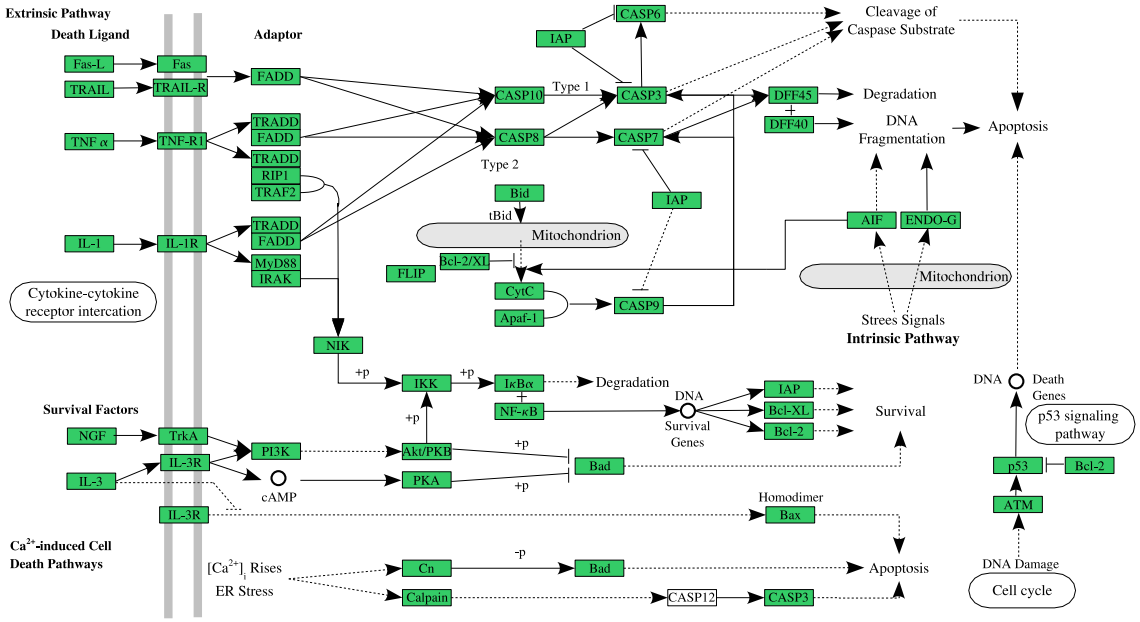

Figure 4.2: A graphical representation of the KEGG pathway for apoptosis (Kanehisa and Goto (2000); Kanehisa et al. (2006, 2008), Entry Number 04210, Release Date 4/14/04)

4.3 Extending Gene Set Analysis to Classification

Svensson et al. (2006) suggest that using a classification approach that is based on gene-sets may better accommodate existing genetic heterogeneity within a group of patients than an approach that works on a gene-by-gene basis. They argue that the combined effect of genes that are functionally related may be significant in terms of being able to classify and predict the clinical outcome of interest, whereas the effect of the individual genes may not be significant. Therefore, if it is known that a pre-defined gene-set is associated with a particular clinical outcome then it is a natural progression to ask if the gene-set can be used to predict that clinical outcome.

By using a gene-set approach to classification it may be possible to discover gene-sets that have predictive power across different datasets. As pointed out by Ein-Dor et al. (2005), when gene lists are obtained from classification methods based on a gene-by-gene approach it is possible to generate a large number of gene lists all of which give a suitably high degree of classification/predictive accuracy. However, there may be little or no overlap between these gene lists, and the genes within these list could be unrelated in terms of their biological function, thus providing little insight into the biology separating the classes.

To test the idea that groups of functionally related genes can be used as predictors in microarray classification problems, a methodology has been developed here that facilitates gene-set based classification, closely paralleling the approaches described above for detecting significant changes in gene set activity. This methodology utilises principal co-ordinates analysis (PCO) to create a summary of the expression levels of given gene-sets, and then uses these summaries as explanatory variables in the classification and prediction process. This procedure is illustrated via application to the three breast cancer datasets of Wang et al. (2005a), Ivshina et al. (2006) and Loi et al. (2007), that are publicly available via the Gene Expression Omnibus (GSE2034, GSE4922 and GSE6532 respectively).

4.4 Breast Cancer Microarray Data

The breast cancer tumour data used in this chapter was obtained from microarray experiments using Affymetrix Human U133A GeneChips, which provides expression levels for 22,283 probes, representing 13076 genes in the human genome. The data were downloaded from the Gene Expression Ominbus (GEO) database on the NCBI website (Edgar et al., 2002), with series entries: GSE2034 (Wang et al., 2005a), GSE4922 (Ivshina et al., 2006) and GSE6532 (Loi et al., 2007).

Wang et al. (2005a) used a probe-based classification method in a training set of 115 patients, to identify a 76-probe list that showed 93% sensitivity and 48% specificity

in an independent test set that consisted of 171 lymph-node-negative patients, where the clinical outcome of interest was the development of distant metastasis within five years. The method Wang et al. (2005a) employed started by dividing the training data into two subgroups based on the amount of estrogen receptor present, genes were then selected within these two subgroups via the use of univariate Cox proportional-hazards regression and a bootstrap technique to reduce the effect of multiple testing. Once the 76-probe list was identified, a relapse score was calculated for each patient's risk of distant metastasis using a linear combination of the expression values weighted by the standardised Cox regression coefficients. A threshold at the classification point was used to ensure a sensitivity of 100% and the highest specificity in the training data. The functionality of the 76-probe list was assessed using pathway analysis, and Wang et al. (2005a) state that although 18 of the 76 probes did not have any known function, several pathways were well represented in the 58 probes that were associated with known biochemical activities, including cell death, cell cycle and immune response.

In a large study on the histological grading of breast cancer tumours Ivshina et al. (2006) aimed to test the hypothesis that a gene expression signature that was capable of differentiating between high and low-grade tumours could also be used to detect differences in medium grade tumours. In their analysis of breast cancer expression data from three independent sources, they identified a gene expression signature of 264 probes that could effectively distinguish between high and low-grade tumours. They also discovered that a subset of this signature could separate the medium grade tumours into two highly discriminated subgroups: one that portrays very similar survival outcomes as those for patients with high grade tumours and the other subgroup whose survival outcomes are very similar to patients with low grade tumours. The 264 probes in the gene expression signature were the maximum number of probes that could accurately differentiate between the low and high grade tumours in one of the three data sets. The classification method employed to find this gene signature was the prediction analysis of microarrays (PAM) algorithm proposed by Tibshirani et al. (2002). The data used to generate the gene expression signature, called the "Uppsala Cohort" in Ivshina et al. (2006), was comprised of gene expression data from 249 primary invasive breast cancer tumours, where the expression data was obtained from Affymetrix Human U133A microarrays which provided expression levels for 22,283 probes.

The third dataset was first published by Loi et al. (2007) who used this dataset, along with six other datasets, to determine if histological grading of breast cancer tumours could assist in the classification of different molecular subtypes within the molecular profiles of breast cancer tumours that were identified as being estrogen receptor positive (ER-positive). Using a gene expression grade index, which they had proposed in an earlier paper (Sotiriou et al., 2006), they were able to identify two ER-positive molecular

subtypes which they showed to be clinically distinct across the seven datasets. The previously unpublished dataset consisted of gene expression data, evaluated on Affymetrix Human U133A microarrays, for 268 patients of which all but five were identified as being ER-positive and all of which were treated with tamoxifen.

Table 4.1 provides a summary of some of the clinical and pathological characteristics available for the three datasets. These include a summary of the age of the patients in each dataset at the time of each study, the estrogen receptor status of the tumours, where a tumour was classified as being ER-positive if the tumour sample consisted of greater than 10fmol/μg protein for the GSE2034 and GSE6523 datasets and greater than 0.05fmol/μg DNA for the GSE4922 dataset. The histological gradings of the tumours are also represented in Table 4.1, although it was not clear from Wang et al. (2005a) which grading system was used for the GSE2034 dataset: the other two datasets used the Nottingham Grading System (Elston and Ellis, 1991). This system classifies tumours to three grades based on a microscopic evaluation of the tumour cells which aims to quantify the aggressive behaviour of the tumour. Grade 1 (Low) represents tumours that are well-differentiated and slow-growing, grade 2 (Medium) represents tumours that are moderately differentiated and grade 3 (High) represents tumours that are poorly differentiated and highly proliferative. The other clinical variable in Table 4.1 is the clinical outcome used as the class variable in the classification models in the following section. These class variables split the data into two classes, those who suffered a relapse or recurrence of the cancer within a given time and those who remained disease free within that time. For the GSE2034 dataset the class variable was based on the clinical event of metastasis within 5 years, for the GSE4922 dataset this was recurrence of any type or death due to cancer within 10.5 years and for the GSE6532 dataset the class variable was based on the clinical event of relapse within 14 years.

Table 4.1: Clinical and pathological characteristics of the three datasets. Values for GSE2034 were adapted from Table 1 in Wang et al. (2005a) and supplementary information associated with the GSE2034 dataset obtained through the Genebank Gene Expression Ominbus database. Values for the GSE4922 and GSE6532 datasets were obtained from the clinical information available through the Genebank Gene Expression Ominbus database as supplementary information associated with each datasets.

	GSE2034 (286 Patients)	GSE4922 (249 Patients)	GSE6532 (268 Patients)
Age			
≤ 40	36	19	1
$41-55$	129	60	45
$56-70$	89	92	152
>70	32	78	70
Clinical Outcome	Metastasis within 5 years	Recurrence of any type or death within 10.5 years	Relapse within 14 years
Yes (1)	93	89	90
No (0)	193	160	178
Estrogen Receptor Status	>10fmol/μg protein	>0.05fmol/μg DNA	>10fmol/μg protein
Positive	209	211	263
Negative	77	34	5
Unknown	-	4	-
Histological Grade	Unknown	Nottingham Grading System	Nottingham Grading System
Low	148	68	50
Medium	42	126	131
High	7	55	47
Unknown	89	-	40

4.5 Methods

4.5.1 Gene-Based Classification Method

In order to provide a baseline level of performance against which to compare a gene-set based classification tool, a standard gene-based classifier that can be used to predict relapse was constructed in the following way for each dataset. First the data were randomly split into training and test sets, so that the training set was twice as large as the test set for the GSE4922 and GSE6532 datasets (i.e., a two to one training to test ratio). The GSE2034 dataset was split into training and test sets stratified by the estrogen receptor level in the tumour, in a effort to match the training and test split used by Wang et al. (2005a) (data are described in Section 4.4). The genes in the training set were then ranked using the absolute T-statistic (assuming unequal variance) and the top 20 ranked genes, with the use of forward selection, were used to build a logistic regression model that minimised the Bayesian Information Criterion. This criterion was chosen as it is less likely to choose a model that is over parameterised compared to using the Akaike Information Criterion (Celeux, 2005). Logistic regression was chosen as it can be seen as a baseline method representing a standard statistical approach. Once the optimal model had been found in the training data, it was used to classify the test data, allowing calculation of the classification rate, the proportion of each class correctly classified (i.e., the Sensitivity for the recurrent class and the Specificity for the non-recurrent class) and the number of genes used in the classification model. Survival analysis was then used to test for a difference between the survival times of the predicted classes for the test data. The survival analysis p-value and performance statistics for the test data were retained.

The use of a ranking statistic that is specifically designed for gene expression data (e.g., SAM) was considered, however as the main interest was in establishing a baseline performance result, a simple and computationally efficient statistical procedure was selected. The use of a threshold at the classification point to improve the performance statistics was also investigated. Section 4.6 discusses the results from applying these methods to three publicly available breast cancer datasets.

To provide a level of confidence for the performance statistics obtained using this baseline classification method, two types of confidence intervals were calculated depending on how the results were reported. When average performance statistics were obtained 95% confidence intervals for these averages were obtained using a standard approach for calculating a confidence for a single mean. That is $\bar{\theta} \pm t_{df} \times se(\bar{\theta})$, where $\bar{\theta}$ is the average performance statistic, t_{df} is the appropriate quantile from the Student's t distribution and $se(\bar{\theta})$ is the standard error of the average performance statistic. The second type of confidence interval calculated was for when median performance statis-

tics were obtained. For each of the median performance statistics a 95% empirical confidence interval (ECI) was calculated by choosing the 95% quantile range that was the shortest. The quantile ranges were based on the distribution of the observed performance statistics. Three quantile ranges were considered these were the 0%-95%, 2.5%-97.5% and 5%-100% quantile ranges. This method of generating confidence intervals was chosen for two reasons. Firstly, it was chosen due to the highly skewed nature of the observed performance statistics when a large number of repeated random splits of the datasets into training and test sets were performed. The second reason was because it gave a good description of the distribution of the observed performance statistics over the large number of repeated random splits of the datasets into training and test sets.

4.5.2 Gene-Set-Based Classification Method

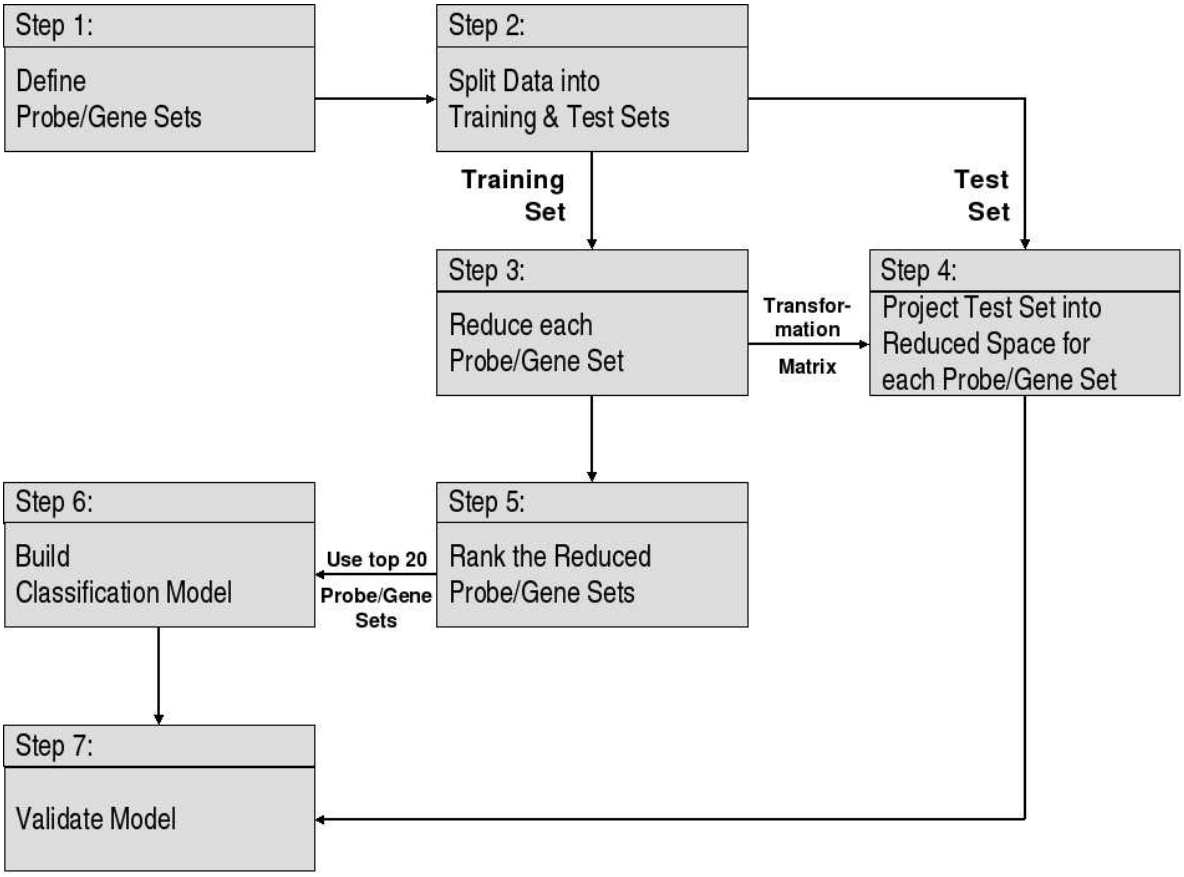

Figure 4.3: Flow Chart of Proposed Gene-Set-Based Classification Methodology

Figure 4.3 represents the gene-set methodology proposed as a series of steps in a flow chart. In step one the gene-sets were defined using a suitable method: the methods used to test this methodology consisted of two pre-defined gene-set methods (namely

KEGG pathways and GO Terms) and the gene-set definition method proposed by Song (2009) that forms gene-sets based on the correlation levels of the gene expression values in the dataset. In the second step the data were split into training and test sets. A 2/3 training, 1/3 test split was used to test this methodology with 11 fold cross-validation used as a comparison. Eleven folds were chosen as 11 divided evenly into the sample size of one of the three datasets. Step three reduces the dimensionality of each gene-set, based solely on the training sets from step two, to 3 dimensions (or the smallest number of dimensions that captures/preserves most of the variation in the dataset, as determined by the cumulative sum of the eigenvalues). This reduction was achieved via the use of the Singular Value Decomposition (SVD) of the transpose of the original expression matrix (putting genes/probes in columns) to calculate the principal component scores and a transformation matrix for Principal Co-Ordinates analysis (PCO) (see Chapter 2 for further details). In step four the test data were projected into the reduced space for each gene-set using the transformation matrix derived in the previous step. In step five Hotelling's T^2 statistic was used to rank the reduced gene-sets in the training set. Step six consisted of building a classification model. To test this methodology logistic regression with forward selection which drew from the top 20 ranked gene-sets from step five was used to construct the classifier(s). The decision to draw from only the top 20 ranked gene-sets was based on a need to keep the focus on only those gene-sets that showed a high level of separability between the two classes of the response variable and to limit the computational time involved in building each classifier. The explanatory variables used in the regression were either the first principal component or the first two principal components (PCs) for each of the top 20 ranked gene-sets and the response variable was the class labeling. The classes were binary in nature, representing a clinical outcome of interest (e.g., recurrence of cancer within a given time frame). Logistic regression was chosen as it can be seen as a baseline method representing a standard statistical approach, as well as to ensure that the classification method used in this methodology was the same as that used in the gene based method. As all of the model selection and refinement procedures also took place in this step, the final classification/prediction model was defined independently of the test data, so that in step seven the classification model could be validated using the independent test set by calculating the classification rate, the proportion of each class correctly classified (i.e., the sensitivity for the recurrent class and the specificity for the non-recurrent class), and the number of gene-sets used in the classification model. Survival analysis was then used to test for a difference between the survival times of the predicted classes for the test data, and the resulting p-value along with the performance statistics for the test data were retained.

To give a more accurate estimate of the true performance statistics, steps two to

seven were repeated a number of times by randomly splitting the dataset into different training and test sets in step two and then proceeding with steps three to seven to obtain estimates of the performance statistics on the test data for each iteration of this cycle by repeating these six steps.

To provide a level of confidence for the performance statistics obtained using this classification method, two types of confidence intervals were calculated depending on how the results were reported. When either a single random split of a dataset or 11 fold cross-validation was used in obtaining the performance statistics 95% asymmetric confidence intervals were obtained using the R function `binconf(n*p,n,alpha=0.05,method="wilson")` from the R package **Hmisc** (Harrell Jr et al., 2009). `p` was the sensitivity, specificity or classification rate for the classifier and `n` was the number of samples/arrays used in the calculation of `p`. The second type of confidence interval calculated was for when median performance statistics were obtained. For each of the median performance statistics a 95% empirical confidence interval (ECI) was calculated by choosing the 95% quantile range that was the shortest. The quantile ranges were based on the distribution of the observed performance statistics. Three quantile ranges were considered these were the 0%-95%, 2.5%-97.5% and 5%-100% quantile ranges. This method of generating confidence intervals was chosen for two reasons. Firstly, it was chosen due to the highly skewed nature of the observed performance statistics when a large number of repeated random splits of the datasets into training and test sets were performed. The second reason that this method was chosen, was because it gave a good description of the distribution of the observed performance statistics over the large number of repeated random splits of the datasets into training and test sets.

KEGG-Based Method

The KEGG-based method uses KEGG pathway annotation to define the gene-sets. Using the **R** package **KEGG** (Liu et al., 2006) and the `getImat()` function in the **R** package **pcot2** (Song and Black, 2007) an identification matrix indicating which genes are associated with each of the pathways in the KEGG database was produced, where each row in the identification matrix refers to a gene in the dataset and each column refers to a pathway in the KEGG database. If gene i is associated with pathway j then the (i^{th}, j^{th}) element in the identification matrix will be 1, if gene i is not associated with pathway j then this element will be 0. KEGG pathways with a minimum of 10 genes per pathway were retained for use as gene-sets in the proposed methodology. The threshold on the minimum number of genes per pathways could be set at any value, the reason 10 was chosen for these analyses was that this threshold removed all of the smaller pathways, leaving only the medium and large sized pathways for further analysis

(Novak and Jain, 2006; Wang et al., 2008).

In addition to using KEGG pathway annotation to define the gene-sets, several other modifications to the general methodology were tested. These included: using a pre-filter on the standard deviations of the probes before defining the probe-sets (i.e., finding the KEGG pathways associated with the retained probes), and, ranking pathways stratified by the estrogen receptor status of the patients.

GO Term Based Method

The GO Term based method used GO Terms as defined by the GO Consortium for creating gene-sets (Ashbuner et al., 2000). As GO Terms can contain anywhere from one to over 6,000 probes per term, only those with between 10 and 100 probes associated with them were chosen to be submitted into the proposed methodology. This range was chosen so as to focus attention on GO Terms with a medium sized set of probes/genes. During the analysis this range was increased to include GO Terms with up to 1000 probes per term. Other variations to the methodology included only using GO Terms that were associated with Cell Cycle (due to the importance of cell proliferation in cancer relapse), and the use of a pre-filter on the standard deviation of the probes before selecting the GO Terms to be used in the analysis.

Correlation-Based Method

The gene-set definition method used in the correlation based method was proposed by Song (2009). The gene-sets are based on the resulting clusters from a hierarchical cluster analysis on the correlation matrix of gene expression data so that a gene-set consists of genes that are highly correlated with one another. To use this correlation-based method for defining gene-sets the general methodology (Figure 4.3) was slightly modified. Step two (splitting the dataset into training and test sets), was performed first so that the gene-set definition method could be applied to the training set, thus maintaining the independence of the test set so that selection bias was not incurred. Once the gene-sets/clusters were defined, the training set for each gene-set was dimensionally reduced according to Step three of the general methodology. For each gene set the genes in the test set that belonged to each gene set were projected into the reduced space using the transformation matrix derived in the previous step. Then the remaining steps of the methodology were applied.

The correlation-based gene-set definition method first transforms the probe expression values to gene expression values, by deriving the average of the expression levels for the probes that are associated with the same gene. It then discards genes that do not have an absolute correlation of at least a value/threshold that is greater than the "background" correlation level in the data. A way to find this value/threshold would

be to permute both the genes and samples (i.e., the rows and columns) of the training data, for each class, and find the maximum correlation, repeat this 1000 times, or more, then find the maximum or 95^{th} percentile for each class and use the maximum of those as the threshold. Here the threshold was set at 0.8 for these analyses, as this value was greater than the 95^{th} percentiles for the three datasets analysed. The distance matrix, based on correlation, of the reduced dataset was calculated and used in a hierarchical clustering algorithm to construct gene-sets that consisted of genes that were highly correlated. The linkage method used here was the Complete Linkage method so that the minimum correlation between any pair of genes in a cluster was directly proportional to the value/height at which the tree/dendrogram was cut. The use of a different linkage method would create slightly different clusterings, thus the choice of linkage method needs to be considered in combination with the objective of the analysis.

For the data in this analysis it was of interest to construct gene-sets that contained genes that were highly correlated with one another, thus the complete linkage method was used (for a description of this linkage method see Chapter 2). When cutting the resulting dendrogram/tree to form the clusters a cutoff of 0.3 was chosen so that the minimum correlation within each cluster was 0.7 and clusters were retained if they contained at least 10 genes (so as to remove the smallest clusters). These clusters formed the gene-sets that were then submitted into the proposed methodology.

4.5.3 Comparison of Gene-Set Definition Methods

There are a number of advantages and limitations to using the gene-set definition methods described above, especially when comparing pre-defined gene-sets, such as KEGG Pathways and GO Terms, to data driven gene-set definition methods such as the correlation based method proposed by Song (2009).

One of the major advantages of using a data driven gene-set definition method is that gene-sets are tailored to the data being analysed, and thus provide an insight into the area being studied that may not be possible if pre-defined gene-sets were used. A disadvantage of this data driven approach is the possibility that gene-sets found to be statistically significant may not be consistent across analyses of the same data by different researchers.

Pre-defined gene-sets such as KEGG Pathways and GO Terms can be used "off the shelf" as all the work and time involved in defining these gene-sets has already been expended. This also means that gene-sets are consistent across analyses by different researchers. However, with this advantage comes the assumption that when the gene-sets were defined the genes were correctly assigned to the appropriate gene-sets. To provide a standardised process to the assignment of genes to GO Terms the Gene Ontology Consortium uses both manual and automated methods which are based on

two main principles. The first of these is that every assignment be attributable to a source, be that a literature reference, computational analysis or another database. The second principle is that the type of supporting evidence found in the source for the assignment be included in the annotation. Another limitation of the use of pre-defined gene-sets is that there may not exist any pre-defined gene-sets suitable for the type data being studied or that many of the genes being studied may not have been assigned to any of the available pre-defined gene-sets.

4.6 Results

Gene-Based Method

The gene based method described in Section 4.5.1 was first applied to the GSE2034 dataset. In an effort to reproduce the results of Wang et al. (2005a) the data were split into training and test sets that mimicked those used by Wang et al. (2005a), who first split the data into two subgroups based on tumour estrogen receptor status, and then each of these subgroups were separated into training and test sets. Gene selection was applied within each of the two training sets stratified by estrogen receptor status; the genes retained in the two training sets were merged and used to build a classifier on the combined training sets and then validated on the combined test sets. However, efforts to reproduce the results of Wang et al. (2005a) were unsuccessful here, even when the 76-probe list defined by Wang et al. (2005a) was used and when different classification algorithms were employed. See Tables 4.2 and 4.3 for results. Two possible reasons for not being successful in reproducing the results of Wang et al. (2005a) are that the exact training and test sets used by Wang et al. (2005a) were not available, and secondly the GSE2034 dataset that was downloaded from the Gene Expression Omnibus showed slightly different clinical outcomes than reported by Wang et al. (2005a), suggesting that the clinical data had been updated since the paper was published.

Figure 4.4 (a) displays a Venn diagram portraying the overlap between the three lists of probes retained in the 1,000 models generated for each of the three datasets when applying the gene based method. Each model was created by treating the probes as being independent from one another and using a different random split of each dataset into training and test sets. As is evident in this figure, there were only 7 probes that were common to all three probe lists. Table 4.4 shows the median performance statistics for the probe-based method. From this table it is clear that the performance statistics obtained when using the probe-based method were very similar across the three datasets, with the median classification rates being 0.62, 0.6 and 0.64 for the GSE2034, GSE4922 and GSE6532 datasets respectively, and none of the median survival p-values being statistically significant. For comparison purposes this method was repeated using the

average expression level of probes that were associated with the same gene. The performance statistics, displayed in Table 4.5, were seen to be consistent across the three datasets, with the median classification rates for the GSE2034, GSE4922 and GSE6532 datasets being 0.63, 0.6 and 0.64 respectively and none of the survival p-values being significant at the 5% level. Figure 4.4 (b) displays a Venn diagram portraying the overlap between the three lists of genes retained in the 1,000 models generated for each dataset; as is evident in this figure there were only 9 genes that were common to all three gene lists. Comparing the performance results from these two sets of analyses, it can be seen that for each dataset there was little or no difference between the performance statistics when treating probes independently and when averaging the expression level of probes associated with the same gene. As the same random splits were used to produce the results in Tables 4.4 and 4.5 a paired t-test for the difference between the classification rates (probes vs. genes) for each dataset could be performed. Once adjusting for multiple comparisons none of the paired t-tests provided a significant result, thus implying that there was no significant difference between the classification rates obtained using the probe-based method and those obtained when using the average expression level of probes that were associated with the same gene. The results were also similar across the three datasets, with the median classification rates being between 60% to 64% and the median survival analysis p-values ranging from 0.1161 to 0.2708.

To ascertain how much of an effect changing the splitting method had on the performance statistics, 11 fold cross-validation was used in combination with the GSE2034 dataset. The classifiers were constructed using the average expression level of probes that were associated with the same gene. This method attained a classification rate of 65% (approximate 95% asymmetric confidence interval of 59% to 70%) with sensitivity and specificity rates of 29% (21%, 39%) and 82% (76%, 87%) respectively, a survival p-value of 0.0249, and from the 11 classifiers constructed an average of 5 genes were used in the classifiers. Comparing these results to those in Table 4.5 the main effect of changing the splitting method was on the survival p-value which showed a large improvement when 11 fold cross-validation was used. This improvement in the survival p-value could be a result of using all of the samples in the GSE2034 dataset to test for a difference between the survival times of the two predicted classes, and that the classifiers were constructed based a large proportion (approximately 91%) of the samples in the dataset when 11 fold cross validation was used. In contrast, the median survival p-value displayed in Table 4.5 was based on the predicted classes of only one third of the samples in the GSE2034 dataset over 1,000 random splits of this dataset into training and test sets with two thirds of the dataset used to construct the classifiers.

Table 4.2: Comparison of the average performance statistics obtained when the probe-based method was used to construct a linear SVM classifier on the Wang et al. (2005a) dataset, with and without probe selection being stratified by estrogen receptor (ER) status. The performance statistics were based on 10 random splits of the dataset into training and test sets and the t-statistic was used to rank the probes. This table displays the maximum of the average classification rates obtained along with the average sensitivity and specificity for the classifiers associated with the maximum average classification rate. 95% empirical confidence intervals are shown in parentheses.

	Probe Selection Stratified by ER Status	
	No	Yes
Number of Unique Probes in union of Top 20 probes from each Random Split	181	186
Overlap with 76-probe list from Wang et al. (2005a)	5	8
Number of Probes in Model	99	2
Sensitivity	0.32 (0.27, 0.36)	0.12 (0.03, 0.21)
Specificity	0.78 (0.75, 0.81)	0.94 (0.88, 0.99)
Classification Rate	0.63 (0.62, 0.64)	0.67 (0.65, 0.69)

Table 4.3: Comparison of the median performance statistics obtained when the probe-based method was used to construct a classifier, based on logistic regression, on the Wang et al. (2005a) dataset, with and without the use of a threshold at the point of classification. The performance statistics were based on 10,000 random splits of the dataset into training and test sets with only the probes contained in the 76-probe signature published by Wang et al. (2005a) used in the classifiers. The threshold value at which a test sample was predicted as belonging to class "1" (recurrent) was chosen as the minimum predicted class "1" probability attained by the recurrent samples in the training set that were correctly classified. Approximate 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

	Threshold at Classification Point Used	
Median	No	Yes
Sensitivity	0.50 (0.34, 0.66)	0.38 (0.21, 0.55)
Specificity	0.66 (0.55, 0.77)	0.77 (0.65, 0.86)
Classification Rate	0.61 (0.53, 0.68)	0.64 (0.57, 0.71)

Table 4.4: Summary of the median performance statistics obtained when the probe-based method was used to construct a classifier, based on logistic regression, for each of the three datasets. The performance statistics were based on 1,000 random splits of each dataset into training and test sets. 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

Median	GSE2034	GSE4922	GSE6532
Number of Probes Kept	5 (3, 9)	6 (3, 10)	5 (3, 8)
Sensitivity	0.34 (0.20, 0.48)	0.36 (0.18, 0.56)	0.36 (0.17, 0.56)
Specificity	0.76 (0.65, 0.86)	0.74 (0.59, 0.87)	0.79 (0.66, 0.90)
Classification Rate	0.62 (0.56, 0.68)	0.60 (0.49, 0.70)	0.64 (0.55, 0.73)
Survival p-value	0.15 (0, 0.8789)	0.27 (0.00001, 0.8978)	0.12 (0, 0.8258)

Table 4.5: Summary of the median performance statistics obtained when the gene-based method was used to construct a classifier, based on logistic regression, for each of the three datasets. The performance statistics were based on 1,000 random splits of each dataset into training and test sets. To obtain an estimate for the expression level of each gene, the expression levels for all of the probes associated with the same gene were averaged. 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

Median	GSE2034	GSE4922	GSE6532
Number of Genes Kept	5 (3, 9)	6 (3, 9)	5 (3, 8)
Sensitivity	0.34 (0.20, 0.48)	0.35 (0.18, 0.57)	0.36 (0.19, 0.54)
Specificity	0.77 (0.65, 0.86)	0.74 (0.59, 0.87)	0.79 (0.66, 0.91)
Classification Rate	0.63 (0.56, 0.69)	0.60 (0.51, 0.70)	0.64 (0.55, 0.73)
Survival p-value	0.14 (0, 0.8717)	0.27 (0.00001,0.9384)	0.12 (0, 0.7753)

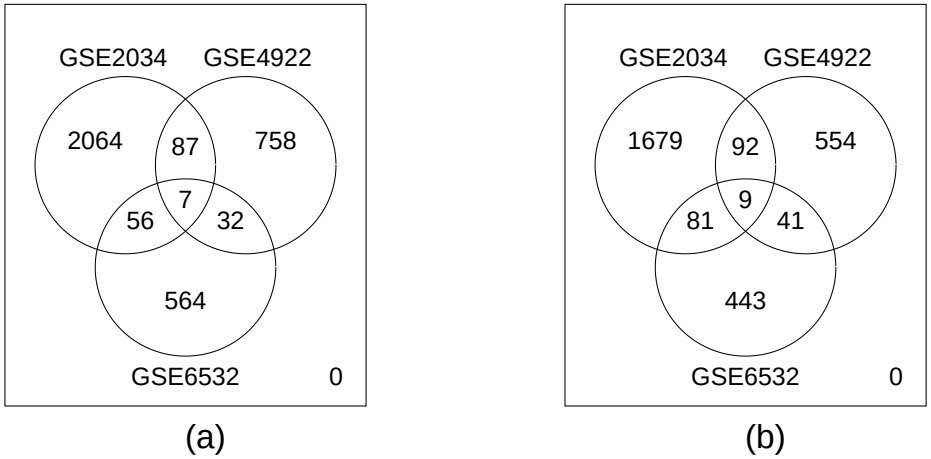

Figure 4.4: Venn diagrams displaying the overlap between (a) the three probe lists generated using the probe-based baseline method and (b) three gene lists generated using the gene-based baseline method

KEGG-Based Method

The first step in the KEGG-based method was to extract the KEGG pathways that were associated with each of the three datasets. There were two ways of achieving this, the first was to extract KEGG pathways based on the Affymetrix probe IDs, treating the probes independently, the second way was to transform the dataset so that the expression levels of probes associated with the same gene were averaged to give an estimate of the expression level for that gene and then use the gene symbols as the basis for extracting the KEGG pathways. When treating the probes independently the number of KEGG pathways with at least 10 probes per pathway was 170 for all three datasets and when the gene symbols were used there were 158 KEGG pathways with at least 10 genes per pathway for all three datasets. This equality of KEGG pathways between the three datasets was due to all three datasets being based on the same platform, namely Affymetrix Human U133A microarrays. Approximately 27% (3,576) of the genes that were represented on the microarrays for these three datasets were associated with at least one KEGG pathway, of these genes 3,539 were contained in the 158 KEGG pathways used in the following analyses.

The methodology, as outlined in Figure 4.3, was then applied, repeating steps two to seven 1,000 times for each dataset using both the probe-based KEGG pathways and the gene based KEGG pathways. The classification models were first built using only the first principal component of each of the reduced KEGG pathways that were highly ranked as the explanatory variables and then with the first two principal components as the explanatory variables. Tables 4.7 and 4.8 show the median performance statistics for these two sets of analyses when basing the KEGG pathways on the Affymetrix probe IDs, and Tables 4.9 and 4.10 show the median performance statistics when the KEGG pathways were based on the gene symbols of the transformed data. There is very little difference between the results using the probe based KEGG pathways and the gene based KEGG pathways. Within both of these definitions of the KEGG pathways, using the first two principal components as the explanatory variables seemed to increase the median sensitivity for each dataset. The median survival p-values for the GSE6532 dataset were seen to decrease when the first two principal components were used as explanatory variables (Tables 4.8 and 4.10). This decrease was also seen for the GSE2034 dataset when the gene symbols were used to extract the KEGG pathways (Table 4.10). For each dataset two lists of KEGG pathways were obtained. This was achieved by combining all of the KEGG pathways that were retained in the 1,000 classification models constructed using the first two principal components as explanatory variables. One list when KEGG pathways were extracted using the probe IDs and the other when the gene symbols were used. Comparing the three lists of KEGG pathways retained when the probe IDs were used, there was an overlap of 82 pathways out of

a total of 134 retained pathways (this overlap is represented in the Venn diagram in Figure 4.5 (a)). When the gene symbols were used to extract KEGG pathways there was an overlap of 80 pathways (out of 149) between the three lists of retained KEGG pathways, this overlap is represented in the Venn diagram in Figure 4.5 (b).

Using the Affymetrix probe ID based method of extracting KEGG pathways two modifications to the general methodology were tested on the GSE2034 dataset. For both of these modifications the classifiers where constructed using the first principal component of each of the highly ranked KEGG pathways as explanatory variables. The two modifications were:

- The ranking of pathways within the estrogen receptor (ER) status of each sample, to assess any improvement in the performance statistics. When the performance statistics from one random split of the GSE2034 dataset for this method were compared to those obtained when the ER status of the samples was not used to stratify the pathways for ranking, no significant difference was seen between the classification rate and specificity for the two sets of performance statistics (Table 4.6). A slight improvement in the sensitivity was observed when the ER status was used to stratify the pathways for ranking.

- A pre-filter on the standard deviations of each probe, disregarding class information, was also considered. Using the 5th, 10th, 15th, 20th and 25th percent quantiles of all the standard deviations from every probe in the dataset, probes with standard deviations below these quantiles where removed from the dataset prior to analysis. These pre-filters were applied in order to investigate the effect on the performance statistics when probes with small standard deviations were excluded from the analysis. Table 4.11 gives a summary of the results for these 5 methods applied to the GSE2034 dataset, which showed no change in the median classification rate.

To ascertain how much of an effect changing the splitting method had on the performance statistics 11 fold cross-validation was used in combination with the GSE2034 dataset with the gene-sets defined as KEGG pathways that contained at least 10 genes per pathway. The classifiers were constructed using the first two principal components of the top ranked KEGG pathways. This method attained a classification rate of 66% (approximate 95% asymmetric confidence interval of 60% to 71%) with sensitivity and specificity rates of 27% (19%, 37%) and 84% (78%, 89%) respectively, a survival p-value of 0.0094, and from the 11 classifiers constructed an average of 6 KEGG pathways were used in the classifiers. Comparing these results to those in Table 4.10 the main effect of changing the splitting method was on the survival p-value which showed a large improvement when 11 fold cross-validation was used. This improvement in the survival

p-value could be a result of using all of the samples in the GSE2034 dataset to test for a difference between the survival times of the two predicted classes, and that the classifiers were constructed based on a large proportion (approximately 91%) of the samples in the dataset when 11 fold cross validation was used. In contrast, the median survival p-value for the GSE2034 dataset displayed in Table 4.10 was based on the predicted classes of only one third of the samples in this dataset. The median was calculated over 1,000 random splits of this dataset into training and test sets, where two thirds of the samples in this dataset were used to construct the classifiers.

Eleven fold cross validation in combination with the GSE2034 dataset was also used to investigate the effect of changing the number of explanatory variables (i.e., ranked KEGG pathways that contained at least 10 genes per pathway) that were used as a source group to construct the classifier. This involved changing the number of top ranked KEGG pathways in this source group from 20 to 10, 15 and 30. The results from these analyses, displayed in Table 4.12, showed that the highest classification rate and specificity were attained when the top 20 ranked KEGG pathways were used to construct the classifiers. The main effect that changing the number of top ranked KEGG pathways in the source group had was on the survival p-value which was at its lowest when the source group contained 10 top ranked KEGG pathways (0.0089) and at its highest when there were 15 pathways in the source group (0.1186).

Table 4.6: Comparing the effect on the performance statistics, from one random split of the GSE2034 dataset into training and test sets, when the reduced KEGG pathways (extracted using the probe IDs) were ranked within estrogen receptor (ER) status to when the reduced KEGG pathways were ranked ignoring ER status. The classification models were constructed using the first principal component of the highly ranked reduced pathways as explanatory variables. 95% asymmetric confidence intervals are shown in parentheses.

	Ranked Within ER Status	
	No	Yes
Number of Pathways Kept	3	3
Sensitivity	0.06 (0.03, 0.13)	0.23 (0.16, 0.33)
Specificity	0.83 (0.77, 0.88)	0.85 (0.79, 0.89)
Classification Rate	0.58 (0.52, 0.64)	0.65 (0.59, 0.70)

Table 4.7: Summary of the median performance statistics obtained when the first principal component of the top ranked KEGG pathways (extracted using the probe IDs) were used to construct a classifier, using logistic regression, for each of the three datasets. The performance statistics were based on 1,000 random splits of each dataset into training and test sets. 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

Median	GSE2034	GSE4922	GSE6532
Number of Pathways Kept	2 (1, 4)	1 (1, 4)	2 (1, 4)
Sensitivity	0.20 (0, 0.36)	0.15 (0, 0.32)	0.17 (0, 0.33)
Specificity	0.88 (0.78, 1)	0.87 (0.75, 1)	0.88 (0.78, 1)
Classification Rate	0.66 (0.57, 0.74)	0.61 (0.52, 0.70)	0.64 (0.55, 0.73)
Survival p-value	0.23 (0, 0.9101)	0.39 (0, 0.9275)	0.30 (0, 0.9317)

Table 4.8: Summary of the median performance statistics obtained when the first two principal components of the top ranked KEGG pathways (extracted using the probe IDs) were used to construct a classifier, using logistic regression, for each of the three datasets. The performance statistics were based on 1,000 random splits of each dataset into training and test sets. 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

Median	GSE2034	GSE4922	GSE6532
Number of Pathways Kept	5 (2, 10)	4 (1, 8)	5 (2, 9)
Sensitivity	0.29 (0.12, 0.47)	0.25 (0.10, 0.42)	0.32 (0.17, 0.50)
Specificity	0.81 (0.70, 0.92)	0.77 (0.63, 0.90)	0.81 (0.70, 0.92)
Classification Rate	0.64 (0.56, 0.72)	0.58 (0.49, 0.67)	0.65 (0.56, 0.73)
Survival p-value	0.23 (0, 0.9078)	0.46 (0.0003, 0.9580)	0.11 (0, 0.8288)

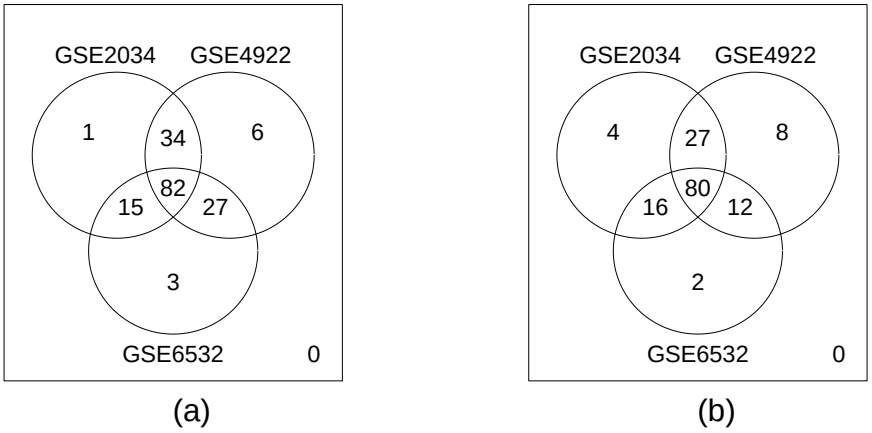

Figure 4.5: Venn diagrams displaying the overlap between (a) the three KEGG pathway lists obtained when the first two principal components of the top ranked KEGG pathways (extracted using the probe IDs) were used to construct the classifiers, and (b) the three KEGG pathway lists obtained when the first two principal components of the top ranked KEGG pathways (extracted using the gene symbols) were used to construct the classifiers.

Table 4.9: Summary of the median performance statistics obtained when the first principal component of the top ranked KEGG pathways (extracted using the gene symbols) were used to construct a classifier, using logistic regression, for each of the three datasets. The performance statistics were based on 1,000 random splits of each dataset into training and test sets. 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

Median	GSE2034	GSE4922	GSE6532
Number of Pathways Kept	2 (1, 5)	1 (1, 3)	2 (1, 4)
Sensitivity	0.21 (0, 0.37)	0.17 (0, 0.32)	0.18 (0.03, 0.36)
Specificity	0.88 (0.77, 1)	0.87 (0.75, 1)	0.88 (0.78, 1)
Classification Rate	0.66 (0.58, 0.74)	0.61 (0.53, 0.70)	0.64 (0.56, 0.73)
Survival p-value	0.23 (0, 0.9092)	0.31 (0, 0.9213)	0.25 (0, 0.9288)

Table 4.10: Summary of the median performance statistics obtained when the first two principal components of the top ranked KEGG pathways (extracted using the gene symbols) were used to construct a classifier, using logistic regression, for each of the three datasets. The performance statistics were based on 1,000 random splits of each dataset into training and test sets. 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

Median	GSE2034	GSE4922	GSE6532
Number of Pathways Kept	6 (3, 12)	5 (2, 10)	4 (2, 8)
Sensitivity	0.32 (0.19, 0.60)	0.27 (0.12, 0.43)	0.28 (0.13, 0.46)
Specificity	0.80 (0.70, 0.90)	0.75 (0.63, 0.90)	0.82 (0.70, 0.93)
Classification Rate	0.65 (0.56, 0.72)	0.58 (0.48, 0.67)	0.63 (0.55, 0.73)
Survival p-value	0.13 (0, 0.8683)	0.48 (0, 0.9465)	0.17 (0, 0.8882)

Table 4.11: Comparing the effect on the performance statistics of the KEGG-based classifier when each of the five standard deviation pre-filters were applied to the training data prior to defining the KEGG pathways (which were extracted using the probe IDs). The performance statistics were based on 1,000 random splits of the GSE2034 dataset into training and test sets. 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

Quantile	Number of Std. Dev. $\leq$ Quantile	Median Number of Pathways Kept	Median Sensitivity	Median Specificity	Median Classification Rate
5%	1115	2 (1, 4)	0.20 (0, 0.36)	0.88 (0.77, 0.98)	0.66 (0.57, 0.74)
10%	2229	2 (1, 4)	0.21 (0, 0.36)	0.88 (0.78, 1)	0.66 (0.57, 0.74)
15%	3343	2 (1, 4)	0.21 (0, 0.36)	0.88 (0.78, 1)	0.66 (0.57, 0.74)
20%	4457	2 (1, 5)	0.21 (0, 0.36)	0.88 (0.78, 1)	0.66 (0.57, 0.74)
25%	5571	2 (1, 4)	0.21 (0, 0.36)	0.88 (0.78, 1)	0.66 (0.57, 0.74)

Table 4.12: Comparing the effect on the performance statistics of the KEGG-based classifier (with the pathways extracted using the gene symbols) when the number of top ranked pathways used as a source to construct the classifiers was varied. The performance statistics were attained using 11 fold cross validation on the GSE2034 dataset with the fist two principal components used to construct the classifiers. Approximate 95% asymmetric confidence intervals for the performance statistics are shown in parentheses.

	Number of Top Ranked KEGG Pathways in Source Group			
	10	15	20	30
Average number of Pathways Retained by the 11 classifiers	4	5	6	9
Sensitivity	0.28 (0.20, 0.38)	0.26 (0.18, 0.36)	0.27 (0.19, 0.37)	0.29 (0.21, 0.39)
Specificity	0.83 (0.77, 0.88)	0.81 (0.75, 0.86)	0.84 (0.78, 0.89)	0.81 (0.45, 0.86)
Classification Rate	0.65 (0.59, 0.70)	0.63 (0.57, 0.68)	0.66 (0.60, 0.71)	0.64 (0.58, 0.69)
Survival p-value	0.0089	0.1186	0.0094	0.0324

GO Term-Based Method

Using GO to annotate the Affymetrix Human U133A probes resulted in 1,256 GO Terms with between 10 and 100 probes per term. Approximately 87% (19,489) of the probes in the three datasets were associated with at least one GO Term, of these probes 13,191 were associated with the 1,256 GO Terms described above. These GO Terms were used to define the gene-sets in Step 1 of the general methodology, outlined in Figure 4.3, and the subsequent steps of the methodology were applied with only the first principal component of each of the highly ranked GO Terms being used in the construction of the classification models, with steps two to seven repeated 1,000 times. Table 4.13 compares the median performance statistics from this application of the methodology to all three of the datasets. From this table it is clear that the median performance statistics for the GSE2034 and GSE6532 datasets showed a very slight improvement over those for the GSE4922 dataset. The median survival p-value for the GSE2034 dataset was the smallest of the three median values and was just under half that obtained for the GSE4922 dataset, however none of these p-values could be considered as being significant. The choice not to compare these results with those that would have been obtained if 11 fold cross-validation was used as the splitting method, was based on the comparisons made in the previous section. That is, the main effect of changing to this splitting method was an improvement in the results from the comparison of the survival curves for the predicted classes. Comparing the lists of GO Terms retained in the classification models for these three datasets there were 23 (out of 644) GO Terms in common between the three lists. This overlap is represented in the Venn diagram in Figure 4.6. This method was repeated with the addition of a pre-filter on the standard deviations of all of the probes in the dataset. Probes were discarded if their standard deviation was less than the 25th, 35th, 45th, 55th, 65th and 75th percent quantiles of all the probe standard deviations in the dataset. These pre-filters were applied to the GSE2034 dataset in order to investigate the effect on the performance statistics when probes with small standard deviations were excluded from the analysis. Table 4.14 gives a summary of the results when the six pre-filters were used. From this table it is clear that the use of the pre-filters had little or no effect on the performance statistics.

Table 4.15 shows the median performance statistics for the GSE4922 dataset when the range of probes per GO Term was increased from 10-100 to 10-1,000 and the first two principal components of the highly ranked GO Terms were used to construct the classification models. Increasing the range on the number of probes per GO Term showed only a slight, but insignificant improvement in most of the median performance statistics compared to the median performance statistics for the smaller range. The exception to this was the median sensitivity which showed a slight decrease in performance. When the 95% empirical confidence intervals for these median sensitivity values

were compared a high degree of overlap was observed, implying that this observed difference was not significant.

With the increased range on the number of probes per GO Term and using both the first and second principal components in the construction of the classification models, it was of interest to see if using only GO Terms associated with Cell Cycle resulted in any improvement in the median performance statistics for the GSE4922 dataset. However, as can be seen in Table 4.16, even though there was a slight improvement in the median specificity and the median classification rate, the median sensitivity showed a slight decrease compared to the performance statistics obtained when using all the GO Terms with between 10 to 1,000 probes per GO Term.

Table 4.13: Summary of the median performance statistics obtained when the first principal component of the top ranked GO Terms (which contained between 10 and 100 probes per term) were used to construct a classifier, using logistic regression, for each of the three datasets. The performance statistics were based on 1,000 random splits of each dataset into training and test sets. 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

Median	GSE2034	GSE4922	GSE6532
Number of GO Terms Kept	3 (1, 5)	2 (1, 5)	2 (1, 4)
Sensitivity	0.24 (0.08, 0.43)	0.22 (0, 0.36)	0.22 (0.06, 0.42)
Specificity	0.85 (0.73, 0.95)	0.81 (0.68, 0.94)	0.85 (0.71, 0.96)
Classification Rate	0.65 (0.56, 0.73)	0.59 (0.51, 0.70)	0.64 (0.55, 0.72)
Survival p-value	0.26 (0, 0.8863)	0.44 (0, 0.9503)	0.33 (0, 0.9142)

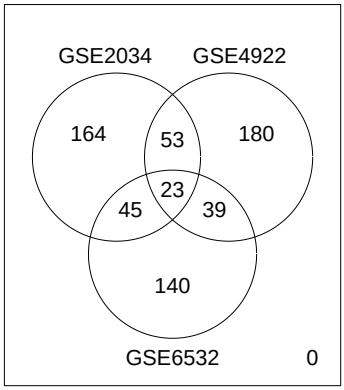

Figure 4.6: A Venn diagram displaying the overlap between the three GO Term lists obtained when the first principal component of the top ranked GO Terms (which contained between 10 and 100 probes per term) were used to construct the classifiers.

Table 4.14: Comparing the effect on the performance statistics of the GO Term based classifier when each of the six standard deviation pre-filters were applied to the training data prior to defining the GO Terms (which contained between 10 and 100 probes per term). The performance statistics were based on 1,000 random splits of the GSE2034 dataset into training and test sets.

Quantile	Number of Std. Dev. $\leq$ Quantile	Median Number of Pathways Kept	Median Sensitivity	Median Specificity	Median Classification Rate
25%	5571	3 (1, 5)	0.24 (0.08, 0.42)	0.85 (0.73, 0.95)	0.65 (0.56, 0.73)
35%	7799	3 (1, 5)	0.24 (0.06, 0.42)	0.85 (0.74, 0.95)	0.66 (0.57, 0.73)
45%	10027	2 (1, 5)	0.24 (0.08, 0.42)	0.86 (0.75, 0.97)	0.66 (0.57, 0.74)
55%	12256	3 (1, 5)	0.24 (0.07, 0.42)	0.86 (0.75, 0.97)	0.66 (0.57, 0.74)
65%	14484	2 (1, 5)	0.24 (0.07, 0.41)	0.87 (0.75, 0.97)	0.67 (0.58, 0.74)
75%	16712	2 (1, 4)	0.24 (0.08, 0.43)	0.88 (0.75, 0.97)	0.67 (0.59, 0.75)

Table 4.15: Comparing the median performance statistics of the GO Term based classifier when the range on the number of probes per GO Term was increased from 10-100 to 10-1,000. The classification models were built using the first two principal components of the top ranked GO Terms as explanatory variables and the median performance statistics were based on 1,000 random splits of the GSE4922 dataset into training and test sets. 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

	GSE4922 Number of Probes Per GO-Term	
Median	10-100	10-1000
Number of GO Terms Kept	8 (4, 13)	7 (3, 12)
Sensitivity	0.52 (0.35, 0.67)	0.33 (0.17, 0.54)
Specificity	0.55 (0.43, 0.65)	0.74 (0.60, 0.85)
Classification Rate	0.54 (0.42, 0.64)	0.59 (0.48, 0.70)
Survival p-value	0.41 (0.0003, 0.9296)	0.38 (0.00002, 0.9474)

Table 4.16: Comparison of the median performance statistics obtained when only GO Terms associated with cell cycle were used in the classifier to those obtained based on all of the GO Terms with between 10 and 1,000 probes per term. The comparison was made using the GSE4922 dataset which was randomly split 1,000 times into training and test sets. The classifiers were constructed using the first two principal components of the top ranked GO Terms as explanatory variables. 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

Median	Cell Cycle Only	All GO Terms
Number of GO Terms Kept	2 (1, 5)	7 (3, 12)
Sensitivity	0.19 (0.03, 0.39)	0.33 (0.17, 0.54)
Specificity	0.84 (0.70, 1)	0.74 (0.60, 0.85)
Classification Rate	0.60 (0.52, 0.69)	0.59 (0.48, 0.70)
Survival p-value	0.43 (0, 0.9448)	0.38 (0.00002, 0.9474)

Correlation-Based Method

This method defined gene-sets by firstly converting the probe expression data into gene expression data, through averaging the expression levels of probes associated with the same gene. Genes whose expression values were not correlated, at the pre-determined level of 0.8, with any of the other genes in the dataset were then removed. The retained genes were then submitted to a hierarchical clustering algorithm which used the correlation matrix of the retained genes as the similarity matrix on which to form the clusters and resulting dendrogram/tree. The dendrogram was then cut at a given point/height and clusters which contained at least 10 genes were retained. These clusters formed the gene-sets that were then submitted into the proposed methodology. Using the GSE4922 dataset and only one random split of the data into training and test sets three different linkage methods were tested in the gene-set definition step of the methodology, as described above and in Section 4.5.2. These were, Complete Linkage, Single Linkage and Average Linkage, along with a combination of different cutting points for the resulting dendrogram and the use of one minus the absolute value of the correlation between the genes as the distance/similarity measure.

When the gene-sets were defined using Complete Linkage, three cut points were considered, h=0.15, 0.2 and 0.3. When cutting the dendrogram at h=0.15 and 0.2 no improvement over the null model could be found for the chosen split of the data. This was not the case when cutting at h=0.3 in the gene-set definition step. Results for these models are shown in the first column of Tables 4.17 and 4.18. When Single Linkage was used in the definition of gene-sets, the same three cut points of were tested. The results are shown in column two of Tables 4.17 and 4.18 for gene-sets that showed any improvement over the null model. The third linkage method, Average Linkage, was used in combination with cut points of h= 0.15, 0.2 and 0.3. The results for gene-sets that showed any improvement over the null model are displayed in column three of Tables 4.17 and 4.18. Comparing the classification rates and survival p-values from these two tables it was seen that using Single Linkage with a cut point of h=0.15 and using the first principal component of the highly ranked clusters to construct the classifier gave the smallest p-value (0.00005, Table 4.17) and one of the highest classification rates (0.69 with an approximate 95% asymmetric confident interval of (0.63, 0.74), as shown in Table 4.18). It was also noted that all of the survival p-values were significant when only the first principal component was used to construct each classifier and that as the cutting point for the tree was increased the number of test samples that were predicted as being recurrent also increased. This increase had the general effect of decreasing the classification rate slightly and increasing the survival p-values.

To test if clusters derived from only one of the classes could be used to efficiently distinguish between both classes, clusters were created within each of the two classes

in the training data such that clusters formed using the recurrent data ("R") in the training data were used to build a classification model using all of the training data. This model was then used to predict all of the test data. This was then repeated with clusters derived using the non-recurrent data ("NR") in the training data. The clusters were constructed using the Average Linkage method with cut points of h=0.15, 0.2, 0.25 and 0.3 being considered. The classification models were constructed using both the first principal component and the first two principal components of the highly ranked clusters as explanatory variables. Tables 4.19 and Tables 4.20 display the results from these analyses. From these tables it is clear that all of these methods resulted in models that only predicted a small number of subjects as recurrent. To resolve this the following alterations/additions where investigated with complete linkage and cutting the tree at h = 0.3

1. Standardising the genes to $\mu = 0$ and $\sigma = 1$

2. Standardising the genes to $\mu = 0$ and $\sigma = 1$ and using a threshold at the classification point so that the sensitivity in the training data was greater than 50%

3. Standardising the genes to $\mu = 0$ and $\sigma = 1$ and creating binary explanatory variables by considering whether the training samples have values less than or greater than the mean of the 1st and/or 2nd PC in the training data.

The results from these analyses can be seen in Tables 4.21 to 4.23. Tables 4.21 and 4.22 show the results from applying the first and second alterations, listed above, to the GSE4922 dataset using only one random split of the dataset into a training and a test set. The results displayed in Table 4.21 were derived from a classifier built using the first principal component of the top ranked clusters as explanatory variables, and the results in Table 4.22 were obtained from a classifier that was constructed using the first two principal components of the top ranked clusters as explanatory variables. From both of these tables, it is clear that the number of test samples that were predicted as being from the recurrent ('R') class dramatically increased when the threshold, described in the second item of the list above, was incorporated into the classification model. By incorporating the threshold a very noticeable decrease was observed in the survival p-value (well under half of that obtained without the use of the threshold) along with a slight decrease in the classification rate for the test set.

The proposed methodology was applied to all three of the datasets with the genes standardised to $\mu = 0$ and $\sigma = 1$. The classification models were constructed using only the first principal component of the highly ranked clusters. The median performance results were based on 100 random splits of each dataset into training and test sets. The median performance results (displayed in Table 4.24) showed a similar level of median

classification rate across the three datasets, with median sensitivity values substantially smaller than the median specificity values for each dataset. Over a third of the classification models produced for the GSE2034 dataset predicted all of the samples in the test set as belonging to the non-recurrent class. This resulted in a very low median sensitivity, a very high median specificity and only being able to obtain survival p-values for those models which produced predicted samples in both classes. Only the median survival p-value for the GSE6532 dataset showed any evidence of a difference between the two predicted classes (0.0491 compared to 0.1466 and 0.3786 for the GSE2034 and GSE4922 datasets respectively, as shown in Table 4.24).

From these results it was decided to apply the methodology with 100 random splits of the datasets using only the first principal component of the highly ranked clusters to construct the classification models with the genes standardised to $\mu = 0$, $\sigma = 1$ and using a threshold at classification point so that the sensitivity in the training data was greater than 50%. This was because this method showed a substantial increase in the number of test samples being predicted as recurrent whilst not over complicating the classifier and producing a reasonably high classification rate and substantially smaller survival p-values. The median performance statistics from applying this method to all three of the datasets, displayed in Table 4.25, show similar median sensitivity, specificity and classification rates across the three datasets, with the GSE6532 dataset showing a slightly higher median specificity and classification rate. This dataset also gave the smallest median survival p-value, significant at the 5% level, compared with the other two datasets whose median survival p-values were 0.1342 and 0.0949 for the GSE2034 and GSE4922 datasets respectively. The median survival p-values obtained for the GSE4922 and GSE6532 datasets when incorporating the threshold at the classification point were substantially smaller (4.25) than those obtained when the threshold was not used (Table 4.24). The median sensitivity values obtained for all three of the datasets when the threshold was incorporated also showed a vast improvement (4.25) over those obtained when the threshold was not used (Table 4.24).

Comparing the clusters used in the classification models from the first 20 random splits of the data into training and test sets, the GSE2034 dataset used 30 clusters, the GSE6532 dataset used 99 clusters and the GSE4922 dataset used 32 clusters. The decision to compare the clusters used in the classifiers from only the first 20 random splits of the datasets was based on avoiding the laborious task of analysing each of the 780 clusters used in all of the classification models on a manual basis. The gene symbols for the genes contained in each of these clusters were submitted to GATHER, a web-based annotation resource available from Duke University (http://gather.genome.duke.edu, Chang and Nevins (2006)), where the GO Terms associated with the genes in each cluster were attained and sorted by the level of association with the cluster, so that GO

Terms that were more highly associated with the cluster were nearer the top of the list of GO Terms.

The total number of GO Terms associated with at least one of the clusters in the GSE2034 dataset was 253, for the GSE6532 dataset this was 527 and for the GSE4922 dataset there were 372 GO Terms associated with at least one of the clusters used in the first 20 classification models. Comparing all of the GO Terms associated with each of the clusters used in the first 20 classification models for each dataset, there were 218 out of 561 GO Terms common to all three datasets, as shown in Figure 4.7 (a). The GO Terms; *physiological process*, *cellular process*, and *cellular physiological process* were seen to be associated with every cluster across all three of the datasets. As these three GO Terms are each associated with over ten thousand genes it is not surprising that they would be associated to some degree with all of the clusters used in the classification models from the first 20 random splits of each dataset into training and test sets.

Due to the association levels of the GO Terms in these lists ranging from highly significant to almost no significant association, it was of more interest to focus on the GO Terms that were highly associated with each cluster. To achieve this the ten most highly associated GO Terms for each of the clusters within each dataset were extracted and compared. There was a total of 186 GO Terms in the combination of these three extracted lists, of these GO Terms 39 were common to all three of the datasets, as shown in Figure 4.7 (b), where the most frequently appearing GO Terms for the GSE2034 dataset were *defense response*, *immune response* and *response to biotic stimulus* all of which were highly associated with 23/30 of the clusters in this dataset. The GSE6532 dataset was also seen to have the GO Terms; *immune response* and *defense response* appearing the most frequently, each of these being highly associated with 48 out of the 99 clusters used in the first 20 classifiers for this dataset. The most frequently appearing GO Terms for the GSE4922 dataset, highly associated with 22 out of the 32 clusters, were; *M phase mitotic cell cycle*, *mitotic cell cycle*, *M phase*, *nuclear division* and *mitosis*. Including the eight GO Terms stated above only five other GO Terms appeared as being highly associated with at least half of the clusters used in the first 20 classifiers for at least one of the three datasets. These were *organismal physiological process*, *cellular process*, *antigen processing*, *cell cycle* and *cell proliferation*. Figure 4.8 compares the proportion of clusters that each of these 13 GO Terms were highly associated with for each of the three datasets. It can be seen from this figure that, of those GO Terms that were highly associated with at least 50% of the clusters for the GSE4922 dataset all of them were seen to be associated with a considerably smaller proportion of the clusters for both the GSE2034 and GSE6532 datasets. Figure 4.8 also shows that with the exception of *cellular process* and *antigen processing*, those GO Terms that were highly associated with a high proportion of the clusters for the

GSE2034 dataset were also highly associated with a reasonably high proportion of the clusters for the GSE6532 dataset.

Eleven fold cross-validation was used as the splitting method for the three datasets to determine if using a different splitting method affected the performance results. The classification models were constructed using the first principal component of each of the highly ranked clusters. The genes were standardised to $\mu = 0$, $\sigma = 1$ and a threshold was used at classification point so that the sensitivity in the training data was greater than 50%. The performance statistics obtained when 11 fold cross vaildation was used, as displayed in Table 4.26, were similar across the three datasets and only slightly better than those obtained using the 2:1 training/test set splitting method (Table 4.25). As seen in the KEGG based method the main effect of using 11 fold cross validation was observed when comparing the survival p-values obtained when 11 fold cross validation was used to those obtained using the 2:1 training/test set splitting method. In this case there was a very large improvement in the survival p-values all of which showed an extremely high level of significance when 11 fold cross validation was used.

Comparing the clusters used in the 11 classification models constructed for each dataset, the GSE2034 dataset used a total of 16 clusters, the GSE4922 dataset used a total of 13 clusters and the GSE6532 dataset used 57 clusters. The gene symbols for the genes contained in each of these clusters were submitted to GATHER (as for the other method) and the GO Terms associated with the genes in each cluster were attained and sorted by the level of association with the cluster, so that GO Terms that were more highly associated with the cluster were at the top of the list of GO Terms. The total number of GO Terms associated with at least one of the clusters in the GSE2034 dataset was 253, for the GSE4922 dataset this was 288 and for the GSE6532 dataset there were 513 GO Terms associated with at least one of the clusters used in the 11 classification models that were constructed. Comparing all of the GO Terms associated with each of the clusters used in the 11 classification models for each dataset, there were 156 out of 561 GO Terms common to all three datasets, as shown in Figure 4.9 (a). The GO Terms; *physiological process*, *cellular process*, and *cellular physiological process* were again seen to be associated with every cluster across all three of the datasets. Comparing the ten most highly associated GO Terms for each of the clusters within each dataset there was a total of 131 GO Terms in the combination of these three extracted lists. Of these GO Terms 15 were common to all three of the datasets, as shown in Figure 4.9 (b), where the most frequently appearing GO Terms for the GSE2034 dataset were *defense response*, *immune response*, *response to biotic stimulus* and *organismal physiological process* all of which were highly associated with 11 out of 16 of the clusters used in the classification models for this dataset. The most frequently appearing GO Terms for the GSE4922 dataset, highly associated with 11 out

of the 13 clusters used in the 11 classification models, were; *M phase mitotic cell cycle*, *mitotic cell cycle*, *M phase*, *nuclear division*, *mitosis*, *cell cycle* and *cell proliferation*. For the GSE6532 dataset there were only three GO Terms that appeared as being highly associated with most of the clusters, these were *defense response* (associated with 28/57 of the clusters), *response to biotic stimulus* (27/57) and *immune resp* (24/57). Including the eleven GO Terms stated above only four other GO Terms appeared as being highly associated with at least half of the clusters used in the 11 classifiers for at least one of the three datasets. These were *cellular process*, *cytokinesis*, *spindle organization and biogenesis* and *mitotic spindle organization and biogenesis*. Figure 4.10 compares the proportion of clusters that each of these 15 GO Terms were highly associated with for each of the three datasets. It can be seen from this figure that the trends observed in Figure 4.8 are evident in this figure as well.

Using the 11 fold cross validation method described above with the GSE2034 dataset, the pre-determined correlation level (set at 0.8) a gene had to attain with at least one other gene in order to be retained for further analysis was varied. This was performed to ascertain if a slight change to this threshold had any effect on the performance statistics. Thresholds of 0.7, 0.75 and 0.85 were tested and no significant change in any of the performance statistics was observed when compared to the results for the GSE2034 dataset in Table 4.26 where this threshold was pre-defined to be 0.8.

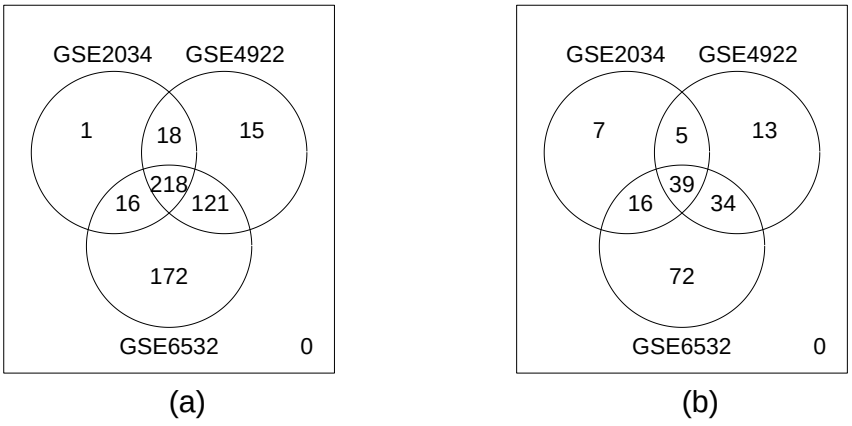

Figure 4.7: Venn diagrams displaying the overlap between the GO Terms associated with each of the clusters used in the classifiers for the first 20 random splits of each dataset. The classifiers were constructed using the first principal component of the top ranked clusters, with the incorporation of a threshold at the point of classification to ensure that the sensitivity in the training set was greater than 0.50 and the genes were standardised to $\mu = 0$, sd $= 1$. (a) compares all of the GO Terms associated with each of the clusters, (b) compares only the top 10 highly associated GO Terms for each of the clusters.

Table 4.17: Comparing the performance statistics of three linkage methods and three cutting points of the resulting dendrograms, for the correlation-based method for classifier construction applied to the GSE4922 dataset. One random split of the dataset into training and test sets was used and the classifier was constructed using only the first principal component of the top ranked clusters as explanatory variables. Approximate 95% asymmetric confidence intervals for the classification rates are shown in parentheses.

	Complete Linkage	Single Linkage		Average Linkage	
	cut at h=0.3	h=0.15	h=0.2	h=0.2	h=0.3
Number of Clusters Kept	1	1	1	1	1
Number of Test Samples Predicted as 'R'	9	5	8	7	9
Classification Rate	0.69	0.69	0.67	0.66	0.69
	(0.63, 0.74)	(0.63, 0.74)	(0.61, 0.73)	(0.60, 0.71)	(0.63, 0.74)
Survival p-value	0.0042	0.00005	0.0387	0.0260	0.0093

Table 4.18: Comparing the performance statistics of three linkage methods and three cutting points of the resulting dendrograms, for the correlation-based method for classifier construction applied to the GSE4922 dataset. One random split of the dataset into training and test sets was used and the classifier was constructed using the first two principal components of the top ranked clusters as explanatory variables. Approximate 95% asymmetric confidence intervals for the classification rates are shown in parentheses.

	Complete Linkage	Single Linkage			Average Linkage		
	cut at h=0.3	h=0.15	h=0.2	h=0.3	h=0.15	h=0.2	h=0.3
Number of Clusters Kept	2	1	1	2	1	1	2
Number of Test Samples Predicted as 'R'	23	9	9	13	6	8	23
Classification Rate	0.61	0.69	0.69	0.54	0.67	0.67	0.59
	(0.55, 0.67)	(0.63, 0.74)	(0.63, 0.74)	(0.48, 0.60)	(0.61, 0.73)	(0.61, 0.73)	(0.53, 0.65)
Survival p-value	0.2020	0.0042	0.0037	0.1860	0.0530	0.0062	0.3970

Table 4.19: Comparing the performance statistics when the classifier was constructed using clusters derived from the recurrent (R) training samples to the performance statistics obtained when the classifier was constructed using clusters derived from the non-recurrent (NR) training samples. The correlation-based method using Average Linkage in the construction of the clusters was applied to the GSE4922 dataset in each case. The performance statistics were also compared across three different cutting points of the resulting dendrograms. One random split of the dataset into training and test sets was used and the classifier was constructed using only the first principal component of the top ranked clusters as explanatory variables. Approximate 95% asymmetric confidence intervals for the classification rates are shown in parentheses.

	"R"			"NR"		
	cut at h=0.2	h=0.25	h=0.3	h=0.2	h=0.25	h=0.3
Number of Clusters Kept	1	1	1	1	1	1
Number of Test Samples Predicted as 'R'	8	9	7	7	9	9
Classification Rate	0.65	0.66	0.66	0.66	0.69	0.69
	(0.59, 0.71)	(0.60, 0.72)	(0.60, 0.72)	(0.60, 0.72)	(0.63, 0.74)	(0.63, 0.74)
Survival p-value	0.1188	0.0236	0.0646	0.0206	0.0093	0.0093

Table 4.20: Comparing the performance statistics when the classifier was constructed using clusters derived from the recurrent (R) training samples to the performance statistics obtained when the classifier was constructed using clusters derived from the non-recurrent (NR) training samples. The correlation-based method using Average Linkage in the construction of the clusters was applied to the GSE4922 dataset in each case. The performance statistics were also compared across three different cutting points of the resulting dendrograms. One random split of the dataset into training and test sets was used and the classifier was constructed using the first two principal components of the top ranked clusters as explanatory variables. Approximate 95% asymmetric confidence intervals for the classification rates are shown in parentheses.

	"R"				"NR"			
	cut at h=0.15	h=0.2	h=0.25	h=0.3	h=0.15	h=0.2	h=0.25	h=0.3
Number of Clusters Kept	1	2	2	2	1	1	2	1
Number of Test Samples Predicted as 'R'	6	8	20	16	6	8	18	9
Classification Rate	0.67	0.67	0.63	0.63	0.67	0.67	0.63	0.69
	(0.61, 0.73)	(0.61, 0.73)	(0.57, 0.69)	(0.57, 0.69)	(0.61, 0.73)	(0.61, 0.73)	(0.57, 0.69)	(0.63, 0.74)
Survival p-value	0.0530	0.0062	0.0884	0.1340	0.0530	0.0062	0.1490	0.0093

Table 4.21: Comparison of the performance statistics obtained when the correlation-based method was applied to one random split of the GSE4922 dataset into training and test sets, with and without the use of a threshold at the point of classification to ensure that the sensitivity in the training set was greater than 0.50. Complete linkage was used in the construction of the clusters and the genes were standardised to $\mu = 0$, sd $= 1$. The classifiers were constructed using the first principal component of the top ranked clusters as explanatory variables. Approximate 95% asymmetric confidence intervals for the classification rates are shown in parentheses.

	Without Threshold	With Threshold
Number of Clusters Kept	1	1
Number of Test Samples Predicted as 'R'	10	33
Classification Rate in test set	0.65 (0.59, 0.71)	0.59 (0.53, 0.65)
Survival p-value	0.2180	0.0979

Table 4.22: Comparison of the performance statistics obtained when the correlation-based method was applied to one random split of the GSE4922 dataset into training and test sets, with and without the use of a threshold at the point of classification to ensure that the sensitivity in the training set was greater than 0.50. Complete linkage was used in the construction of the clusters and the genes were standardised to $\mu = 0$, sd $= 1$. The classifiers were constructed using the first two principal components of the top ranked clusters as explanatory variables. Approximate 95% asymmetric confidence intervals for the classification rates are shown in parentheses.

	Without Threshold	With Threshold
Number of Clusters Kept	1	1
Number of Test Samples Predicted as 'R'	16	38
Classification Rate in test set	0.60 (0.54, 0.66)	0.58 (0.51, 0.64)
Survival p-value	0.4630	0.1440

Table 4.23: Comparison of the performance statistics obtained when the correlation-based method was applied to one random split of the GSE4922 dataset into training and test sets, with the classifier built using binary explanatory variables created using the means of the first principal component and then the first two principal components of the top ranked clusters in the training set. The complete linkage method was used in the construction of the clusters, with the genes standardised to $\mu = 0$, sd $= 1$ and a threshold used at the point of classification to ensure that the sensitivity in the training set was greater than 0.50. Approximate 95% asymmetric confidence intervals for the classification rates are shown in parentheses.

	Using 1st PC	Using 1st and 2nd PC
Number of Clusters Kept	1	2
Threshold Value	0.45	0.42
Number of Test Samples Predicted as 'R'	41	41
Classification Rate in test set	0.57 (0.51, 0.63)	0.57 (0.51, 0.63)
Survival p-value	0.1550	0.2060

Table 4.24: Summary of the median performance statistics obtained when the first principal component of the top ranked clusters from the correlation-based method were used to construct a classifier for each of the three datasets. Complete linkage was used to define the clusters and the genes were standardised to $\mu = 0$, sd $= 1$. The performance statistics were based on 100 random splits of each dataset into training and test sets. 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

Median	GSE2034	GSE4922	GSE6532
Number of Clusters Kept	1 (1, 3)	1 (1, 4)	4 (2, 7)
Sensitivity	0.03 (0, 0.30)	0.11 (0, 0.28)	0.26 (0.07, 0.44)
Specificity	0.98 (0.89, 1)	0.90 (0.79, 1)	0.88 (0.78, 1)
Classification Rate	0.68 (0.63, 0.76)	0.60 (0.52, 0.69)	0.67 (0.58, 0.75)
Survival p-value	0.1466 (0, 0.8459)	0.3786 (0, 0.9243)	0.0491 (0, 0.8730)

Table 4.25: Summary of the median performance statistics obtained when the first principal component of the top ranked clusters from the correlation-based method were used to construct a classifier, which incorporated a threshold at the point of classification to ensure that the sensitivity in the training set was greater than 0.50, for each of the three datasets. Complete linkage was used to define the clusters and the genes were standardised to $\mu = 0$, sd $= 1$. The performance statistics were based on 100 random splits of each dataset into training and test sets. 95% empirical confidence intervals for the median performance statistics are shown in parentheses.

Median	GSE2034	GSE4922	GSE6532
Number of Clusters Kept	1 (1, 3)	1 (1, 4)	4 (2, 7)
Threshold Value	0.365 (0.327, 0.410)	0.385 (0.350, 0.445)	0.410 (0.340, 0.488)
Sensitivity	0.48 (0.32, 0.65)	0.46 (0.25, 0.66)	0.44 (0.23, 0.64)
Specificity	0.68 (0.61, 0.82)	0.68 (0.59, 0.74)	0.77 (0.65, 0.85)
Classification Rate	0.61 (0.54, 0.70)	0.59 (0.51, 0.67)	0.66 (0.56, 0.75)
Survival p-value	0.1342 (0, 0.8434)	0.0949 (0.0003, 0.9200)	0.0265 (0, 0.7267)

Table 4.26: Summary of the 11 fold cross validation performance statistics obtained when the first principal component of the top ranked clusters from the correlation-based method were used to construct a classifier, which incorporated a threshold at the point of classification to ensure that the sensitivity in the training set was greater than 0.50, for each of the three datasets. Complete linkage was used to define the clusters and the genes were standardised to $\mu = 0$, sd $= 1$. Approximate 95% asymmetric confidence intervals for the performance statistics are shown in parentheses.

	GSE2034	GSE4922	GSE6532
Average Number of Clusters Kept	1	1	5
Average Threshold Value	0.367	0.385	0.400
Sensitivity	0.52 (0.42, 0.62)	0.49 (0.39, 0.59)	0.54 (0.44, 0.64)
Specificity	0.71 (0.64, 0.77)	0.72 (0.65, 0.78)	0.79 (0.72, 0.84)
Classification Rate	0.64 (0.58, 0.69)	0.64 (0.58, 0.70)	0.70 (0.64, 0.75)
Survival p-value	4.5e-04	3.1e-06	5.3e-09

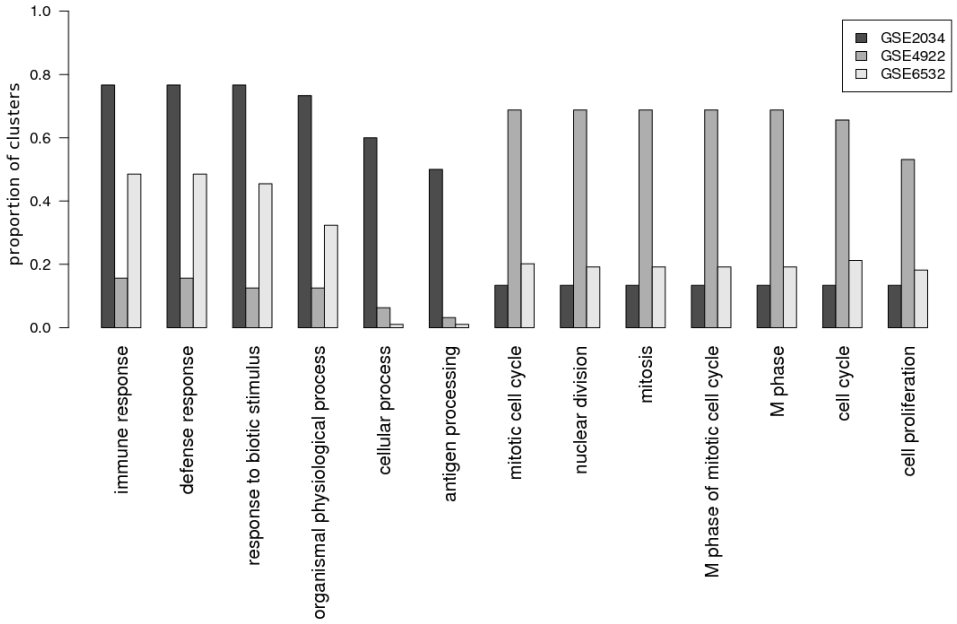

Figure 4.8: There were 13 GO Terms that were highly associated with at least 50% of the clusters used in the classifiers from the first 20 random splits for at least one of the three datasets. This barchart compares the proportion of clusters that each of these 13 GO Terms were highly associated with for each of the three datasets. The classifiers were constructed using the first principal component of the top ranked clusters, with the incorporation of a threshold at the point of classification to ensure that the sensitivity in the training set was greater than 0.50 and the genes were standardised to $\mu = 0$, sd $= 1$.

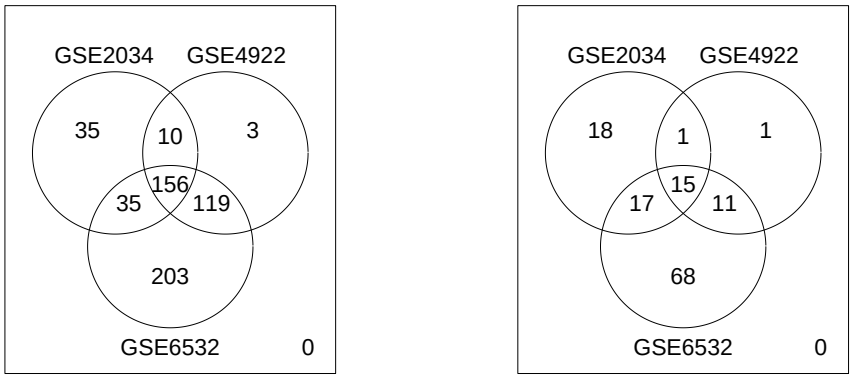

Figure 4.9: Venn diagrams displaying the overlap between the GO Terms associated with each of the clusters used in the 11 classifiers for each dataset. The classifiers were constructed using the first principal component of the top ranked clusters, with the incorporation of a threshold at the point of classification to ensure that the sensitivity in the training set was greater than 0.50 and the genes were standardised to $\mu = 0$, sd $= 1$. (a) compares all of the GO Terms associated with each of the clusters, (b) compares only the top 10 highly associated GO Terms for each of the clusters.

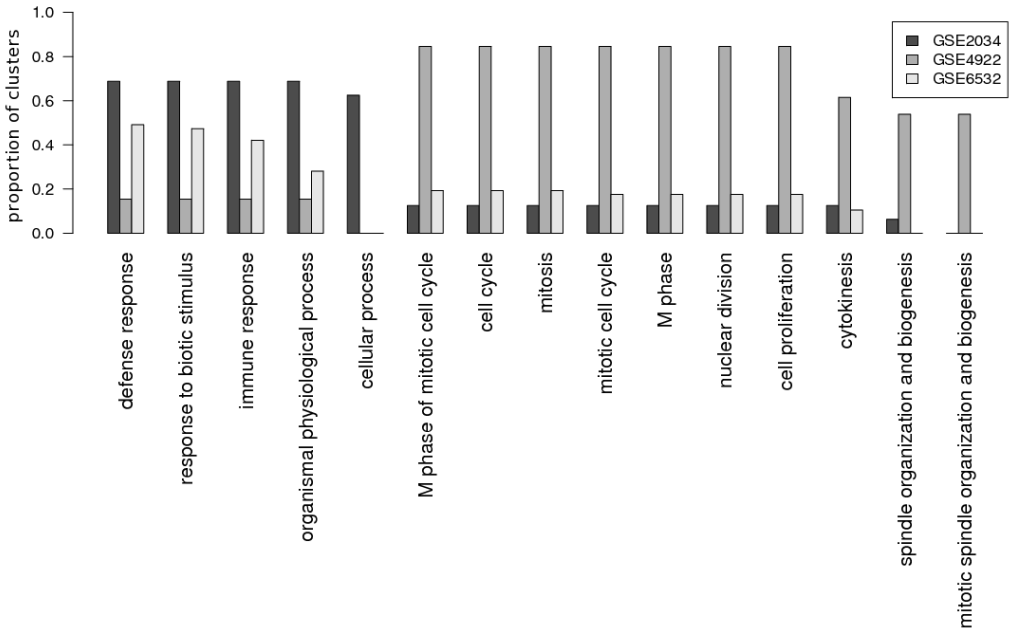

Figure 4.10: There were 15 GO Terms that were highly associated with at least 50% of the clusters used in the 11 classifiers for at least one of the three datasets. This barchart compares the proportion of clusters that each of these 15 GO Terms were highly associated with for each of the three datasets. The classifiers were constructed using the first principal component of the top ranked clusters, with the incorporation of a threshold at the point of classification to ensure that the sensitivity in the training set was greater than 0.50 and the genes were standardised to $\mu = 0$, sd $= 1$.

4.7 Comparison of Results

Probe-Based Data

In the analyses where the probes were treated as being independent of one another, the baseline method of analysing the GSE2034 dataset on a probe-by-probe basis encountered difficulties when trying to produce performance results similar to those published by Wang et al. (2005a), obtaining a maximum average classification rate of only 0.67 (Table 4.2). When the gene based method described in Section 4.5.1 (replacing genes with probes) was applied to each of the three datasets, with median performance statistics based on 1,000 random splits of each dataset into training and test sets, the performance results were seen to be consistent across the three datasets, with the median classification rates for the three datasets ranging from 0.60 to 0.64 (Table 4.4). However the probes used to construct the classifiers for each dataset were markedly different, with only a very small degree of overlap between the three datasets (7/3568, Figure 4.4 (a)).

Using the proposed gene-set based methodology with the Affymetrix probe IDs used to extract gene-sets based on KEGG pathways, the median performance statistics were consistent across the three datasets regardless of whether only the first principal component or the first two principal components of the highly ranked gene-sets were used to construct the classifiers (Tables 4.7 and 4.8). Only a slight improvement to the median survival p-value for the GSE6532 dataset resulted when the classifiers were constructed using the first two principal components of the top ranked pathways as explanatory variables. A very high degree of overlap, 82 out of 134 KEGG pathways to be exact (Figure 4.5 (a)), was observed when comparing the three lists of KEGG pathways (one for each dataset) used in the classification models which were constructed using the first two principal components of the highly ranked gene-sets as explanatory variables. Further investigation of this method, using the GSE2034 dataset, showed that the addition of a pre-filter before Step 1 of the proposed methodology (i.e., Gene-set definition, Figure 4.3) had no effect on the median classification rate (Table 4.11).

When GO Terms that were associated with between 10 to 100 probes were used as gene-sets in the proposed methodology, the analysis of all three datasets showed similar performance statistics, with median classification rates of 0.65, 0.59, and 0.64 for the GSE2034, GSE4922, and the GSE6532 datasets respectively (Table 4.13), and the three lists of GO Terms used in the classifiers for the three datasets had 23 out of 644 GO Terms in common (Figure 4.6). Increasing the number of probes associated with a GO Term to between 10 and 1,000 produced no significant change to the median performance statistics for the GSE4922 dataset, when 1,000 random splits of this dataset into training and test sets were used (Table 4.15). Restricting the GO Terms

used as gene-sets to those only associated with Cell Cycle also produced no significant improvement in the median performance statistics (Table 4.16). A further modification to the methodology was investigated, this consisted of the addition of a pre-filter before Step 1 of the proposed methodology, similar to that used for the KEGG-based method, this also showed no improvement in the median performance statistics.

Comparing these three sets of results, the advantage of using the KEGG-based method over the probe-by-probe method was seen in the higher degree of overlap between the KEGG-pathways used in the classification models for the three datasets (82/134, Figure 4.5 (a)) compared to the very small degree of overlap seen between the probes used in the classification models of the probe-by-probe method for the three datasets (7/3568, Figure 4.4 (a)). This advantage was similarly seen, but to a lesser extent, when comparing the GO Term-based method to the probe-by-probe method, where there was a higher degree of overlap between the GO Terms used in the classification models for the three datasets (23/644, Figure 4.6) than the very small degree of overlap seen between the probes used in the classification models of the probe-by-probe method for the three datasets (7/3568, Figure 4.4 (a)). The proposed methodology, defining probe-sets as those probes associated with either KEGG pathways or GO Terms, yielded very similar performance statistics as those obtained using the probe-by-probe approach described in Section 4.5.1.

These results imply that when functional information, in the form of KEGG pathways or GO Terms, was incorporated into the classification process of the probe based data the predictive performance of the classifier was not hindered, and that by using the proposed gene-set methodology the degree of overlap between the gene-sets retained in the classifiers for each of the datasets analysed was substantially higher than that observed in the baseline approach, suggesting that the results obtained using the gene-set approach had a more biologically meaningful interpretation.

Gene-Based Data

To analyse each of the three datasets using expression levels based on genes the probe-based expression levels for probes associated with the same gene were averaged giving an estimated expression level for each gene represented on the microarray. When the baseline gene-by-gene based classification method described in Section 4.5.1 was applied to the three datasets (GSE2034, GSE4922 and GSE6532) the median performance statistics, based on 1,000 random splits of each dataset into training and test sets, were seen to be very similar across the three datasets (Table 4.5). However the genes used in the classification models for each dataset showed only a very small degree of overlap with only 9 out of the 2859 genes in common between the three datasets (Figure 4.4 (b)).

Applying the proposed methodology with gene-sets defined as KEGG pathways that were associated with at least 10 of the genes represented in each of the datasets produced consistent median performance statistics across all three datasets (Tables 4.9 and 4.10), when both the first principal component and the first two principal components were used to construct the classifiers. A slight improvement to the survival p-values for the GSE2034 and GSE6532 datasets was observed when the classifiers were constructed using the first two principal components of the top ranked pathways as explanatory variables. Comparing the KEGG pathways used in the classifiers which were built using the first two principal components of each pathway, a total of 149 KEGG pathways were used, of these 80 were used in the analyses for all three of the datasets (Figure 4.5 (b)).

The final set of analyses created gene-sets based on clusters from a hierarchical analysis of the correlation matrix of genes seen to be highly correlated with one another. After an appropriate linkage method and cutting point/height for the resulting dendrogram were decided upon. Classifiers were built using only the first principal component of each of the highly ranked clusters as explanatory variables, with the incorporation of a threshold at the classification point. The median performance statistics were based on 100 random splits of each dataset into training and test sets and were, again, seen to be consistent across the three datasets with the median classification rates being 0.61, 0.59 and 0.66 for the GSE2034, GSE4922 and GSE6532 datasets respectively. Using this method of defining gene-sets the median survival p-value, testing for a difference in survival times of the predicted test data, for the GSE6532 dataset was significant at the 5% level (0.0265, Table 4.25). As a way to improve the interpretability and comparability of the clusters obtained using the correlation based method, the genes contained in each of the clusters used in the classifiers for the first 20 random splits of each dataset were subjected to an annotation analysis (using GATHER (Chang and Nevins, 2006)) to find all the GO Terms that were associated with each cluster for each of the three datasets. This analysis produced a list of GO Terms for each dataset that were associated with at least one of the clusters used in the classifiers for the first 20 random splits of each dataset into training and test sets. There was a total of 561 GO Terms in the union of these three lists, of which 218 were common to all three datasets (Figure 4.7 (a)). Comparing only the first 10 most highly associated GO Terms for each of the clusters used in the classifiers for each datasets a total of 186 GO Terms were identified, of these 39 were common to all three datasets (Figure 4.7 (b)). The GO Terms that appeared most frequently as being highly associated with the majority of the clusters used in the classifiers were seen as being connected with immune response, defense response and cell cycle (Figure 4.8).

Comparing these three sets of analyses, the advantage of using the KEGG-based method over the gene-by-gene method can be seen in the higher degree of overlap

between the KEGG-pathways used in the classification models for the three datasets (80/149, Figure 4.5 (b)) compared to the very small degree of overlap seen between the genes used in the classification models of the gene-by-gene method for the three datasets (9/2859, Figure 4.4 (b)). The large degree of overlap between the GO Terms that were highly associated with the clusters used in the first 20 classification models for the three datasets (39/186, Figure 4.7 (b)) again shows how the interpretability of the results can be improved by using the gene-set based classification method over the standard gene-by-gene approach used in this comparison.

The proposed methodology, defining gene-sets as those genes associated with KEGG pathways or genes that are highly correlated with one another, has produced results based on the three datasets (Tables 4.9 and 4.25 respectively) that are similar to the results obtained using the gene-by-gene approach described in Section 4.5.1 (Table 4.5). The notable exception to this was for the GSE6532 dataset where the median survival p-value, testing for a significant difference between the survival times for the predicted response classes, was significant at the 5% level for the correlation based gene-set analysis method (0.0265, Table 4.25) and non-significant for the other two analysis methods, that is, the gene-by-gene approach (0.1161, Table 4.5) and the KEGG-based method (0.2465, Table 4.9). This vast improvement in the median survival p-value, obtained using the correlation based method, for the GSE6532 dataset was also seen, but to a lesser extent, in the GSE4922 dataset where the median survival p-value obtained using the correlation based method (0.0949, Table 4.25), showed improvement over those obtained using the gene-by-gene approach (0.2708, Table 4.5) and the KEGG-based method (0.3099, Table 4.9). For the GSE2034 dataset both the KEGG-based method and the correlation based method produced median survival p-values that were only slightly lower than that obtained by the gene-by-gene baseline approach. When 11 fold cross validation was used instead of the two thirds training set, one third test set splitting method no significant change in the sensitivity, specificity and classification rates was observed. However using 11 fold cross validation did significantly improve the survival p-values for the three datasets when the correlation method was used.

These results imply that when functional information, in the form of KEGG pathways or clusters of highly correlated genes, were incorporated into the classification process of the gene-based data, the predictive performance of the classifier was not detrimentally impacted and in one of the datasets the separability of the predicted classes in terms of the survival times was shown to improve. These results also imply that by using the proposed gene-set methodology the degree of overlap between the gene-sets retained in the classifiers for each of the datasets analysed was greater than that observed in the baseline approach, which suggests that the results obtained using the gene-set approach had a more biologically meaningful interpretation.

5
Conclusions

The aim of this thesis was to investigate the feasibility of incorporating biological information and/or a constraint (based on biological and technological needs) into the binary classification process for gene expression data in an attempt to improve the predictive performance and biological interpretability of the resulting classifier. In this particular instance the gene expression data analysed pertained to the study of recurrence of melanoma and breast cancer.

This broad aim contained two main goals, the first of which was to assess the effect of using a pre-filter in combination with standard statistical ranking methods for gene selection on the predictive performance of a binary classifier. The second goal was to investigate the effect of using a gene-set based binary classification methodology on the predictive performance of the classifier employed.

Goal 1: Gene Selection for Class Prediction

This goal explored the use of a pre-filter that discarded genes which were seen to display a level of differential expression between two classes/phenotypes that was deemed to be of no practical difference, that is, the gene had not undergone a biologically relevant level of differential expression.

Both simulated and real gene expression data were used to assess if the classifiers, based on the genes that were retained by the pre-filter, showed any improvement in predictive performance compared to the baseline classifiers. Comparing the performance results from the baseline classifiers to the performance results from the classifiers based on the proposed gene selection method, no significant difference between these performance results was observed. However, by using the pre-filter, the genes retained in the

classifiers were not only able to predict the new/test data with a similar level of predictive ability as the baseline classifiers, but have also undergone a biologically relevant level of differential expression between the two classes/phenotypes. These genes could then be validated for use in a diagnostic test that requires such a constraint be placed on the expression levels of the genes involved.

One of the main objectives in array-based classification is to discover genes that can be used to discriminate between two or more different states, phenotypes or classes, so that the likelihood of a sample belonging to one of these classes can be calculated. Thus facilitating the development of a non-array based prognostic or diagnostic tool (e.g., using PCR-based techniques). For example, finding a set of genes that can accurately predict the likelihood of relapse in melanoma cancer patients. When gene expression data is obtained via 2 channel spotted microarrays the results from the research undertaken for goal one have shown that it is possible to place biological/technological constraints onto the gene expression data being analysed without detrimentally impacting the predictive performance of a classifier.

Goal 2: Incorporating Functional Information into the Classification Process

Although the pre-filter method, developed to investigate goal 1, could produce a list of genes that showed both statistical and practical significance in their expression level between two classes/phenotypes these genes may be unrelated in terms of biological function, which makes the biological interpretability of the classifier harder to ascertain. To explore this issue a gene-set based methodology, consisting of a seven step process (Figure 4.3), for incorporating such biological functional information was devised. To assess any improvement in the predictive performance of this methodology over a baseline gene-by-gene approach three publicly available breast cancer datasets were analysed.

When the performance results for the gene-set based classifiers were compared to those obtained for the baseline method, an improvement in the predictive performance was seen when the correlation based method for defining gene-sets was employed. Specifically this improvement was seen in the median survival p-values for the GSE4922 and GSE6532 datasets, which were significant at the 10% and 5% levels respectively, this was a major improvement over the baseline method. The gene-set based methodology also showed an improvement in the biological interpretability of the classifiers compared to the classifiers derived for the baseline method. The biological interpretability of the classifiers was, in this case, defined as the degree of overlap between the variables used in the classifiers for each of the three datasets analysed. For the gene-by-gene baseline approach the overlap between the genes/probes used in the classifiers was very small (9/2859 genes and 7/3568 probes, Figure 4.4) compared to the very high degree of

overlap observed for the gene-set approach (KEGG: 82/134 and 80/149 (Figure 4.5), GO: 23/644 (Figure 4.6), Correlation: 39/186 (Figure 4.7b)).

The results from the research undertaken in the exploration of goal two have shown that incorporating functional information into the classification process so that groups of functionally related genes are treated as explanatory variables in the classifier, can not only improve the biological interpretability of the classifier, but that the predictive ability of that classifier can also be improved. Improving the biological interpretability of a classifier could lead to a greater understanding of how groups of functionally related genes work together to affect the prognostic outcome of a given disease. For instance, this may provide the opportunity to develop drugs that can specifically target the groups of functionally related genes that lead to poor prognostic outcomes rather than a wide spectrum of functionally related genes that may include those that are beneficial to prognostic outcome. This may then lead to new ways of detecting and possibly preventing a number of diseases not only in humans but any living organism. For example, the identification of which functionally related genes are involved in bacterial diseases that affect plants.

Future Directions

A possible extension of the gene-set based methodology developed for goal two would be to incorporate clinicopathological variables into the classification process such as age at diagnosis, gender, ethnicity, tumour size and tumour grade. Incorporating this information into the classification process could not only improve the predictive performance of the classifier, it could also highlight unforeseen relationships between the clinicopathological variables and the genetic variability of the patients as well as providing a better understanding of how both of these work together to determine the prognostic outcome of a patient. One possible way to incorporate clinicopathological variables into the classification process would be to include them as explanatory inputs into the classifier alongside the highly ranked dimensionally reduced gene-sets.

Given the heterogeneous nature of cancer it is possible that the relapse category may be comprised of a number of subgroups each with its own biological reason for disease relapse. Investigation into these different biological reasons for disease relapse based on subgroups of the relapse category (defined using array data and/or clinicopathological variables) would not only broaden our understanding of which biological mechanisms play an important role in disease relapse for different subgroups of patients, it could also lead to the development of prognostic tools that are patient specific. The gene-set methodology developed for goal two could be extended to find gene-sets that can differentiate between each of the relapse subgroups, or a set number of these subgroups by using, for example, the multivariate equivalent to the F-test (i.e., MANOVA) in

the ranking stage of the analysis coupled with a multi-class classifier for building the classification models.

With the emergence of the next generation of sequencing techniques Mardis (2008) has implied that with the use of this new technology it will not only be possible to identify gene variants that can be used to predict disease susceptibility but that it will also be possible to determine functional annotations for these gene variants. It could therefore be possible that by incorporating such functional information into the classification process a similar increase in biological interpretability and predictive ability may be seen. Wold and Myers (2008) have pointed out (in their review of the possible uses of the next generation of sequencing techniques) that a subset of these techniques known as "sequence census" counting assays can be used as a way of obtaining the measure of a complex nucleic acid sample. Thus providing an alternative technological method to microarrays. They indicate that these assays can measure the global genome-wide abundance level of mRNA (to give but one example) and that these assays can bypass several of the longstanding technical issues that surround microarrays (e.g., cross-hybridization and quantification difficulties owing to the continuous nature of the hybridization signals). The count data obtained from such assays could be used to estimate differences in gene activity between different states/phenotypes by comparing the distribution of counts for each gene (or groups of genes) between the different states/phenotypes. By summarising the distributions of a group of functionally related genes it would be possible to obtain a one dimensional summary for each of the known functions in a biological system which could then be used in the analysis process.

It is hoped that the findings of this research and that of others in the field might provide advances in medical prognosis that can benefit not only those who suffer and treat cancer but a range of other conditions.

Bibliography

Affymetrix (2005) Guide to Probe Logarithmic Intensity Error (PLIER) Estimation. *Tech. rep.*, Affymetrix I. Santa Clara, CA.

Agresti, A. and Coull, B. A. (1998) Approximate is better than "exact" for interval estimation of binomial proportions. *American Statistician*, **52(2)**, 119–126.

Akaike, H. (1974) New Look at Statistical-Model Identification. *IEEE Transactions on Automatic Control*, **AC19**, 716–723.

Aldenderfer, M. S. and Blasfield, R. K. (1984) *Cluster Analysis*. Beverly Hills: Sage Publications, Inc.

Alizadeh, A. A., Eisen, M. B., Davis, R. E., Ma, C., Lossos, I. S., Rosenwald, A., Boldrick, J. G., Sabet, H., Tran, T., Yu, X., Powell, J. I., Yang, L. M., Marti, G. E., Moore, T., Hudson, J., Lu, L. S., Lewis, D. B., Tibshirani, R., Sherlock, G., Chan, W. C., Greiner, T. C., Weisenburger, D. D., Armitage, J. O., Warnke, R., Levy, R., Wilson, W., Grever, M. R., Byrd, J. C., Botstein, D., Brown, P. O. and Staudt, L. M. (2000) Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature*, **403**, 503–511.

Allison, D. B., Cui, X., Page, G. P. and Sabripour, M. (2006) Microarray data anlysis: from disarray to consolidation and consensus. *Nature Reviews*, **7**, 55–65.

Ambroise, C. and McLachlan, G. J. (2002) Selection bias in gene extraction on the basis of microarray gene-expression data. *Proceedings of the National Academy of Sciences of the United States of America*, **99**, 6562–6566.

Anderberg, M. R. (1973) *Cluster Analysis for Applications*. Academic Press, New York.

Apweiler, R., Bairoch, A., Wu, C. H., Barker, W. C., Boeckmann, B., Ferro, S., Gasteiger, E., Huang, H. Z., Lopez, R., Magrane, M., Martin, M. J., Natale, D. A., O'Donovan, C., Redaschi, N. and Yeh, L. S. L. (2004) Uniprot: the universal protein knowledgebase. *Nucleic Acids Research*, 115–119.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., Sherlock, G. and Gene Ontology Consortium (2000) Gene ontology: tool for the unification of biology. *Nature Genetics*, **25**, 25–29. URL `http://www.ebi.ac.uk/faq/cgi-bin/go?file=14`.

Bair, E. and Tibshirani, R. (2004) Semi-supervised methods to predict patient survival from gene expression data. *PLoS Biology*, **2**, 0511–0522.

Barrett, T., Troup, D. B., Wilhite, S. E., Ledoux, P., Rudnev, D., Evangelista, C., Kim, I. F., Soboleva, A., Tomashevsky, M. and Edgar, R. (2007) NCBI GEO: mining tens of millions of expression profiles–database and tools update. *Nucleic Acids Research*, **35**, D760–765.

Barry, W. T., Nobel, A. B. and Wright, F. A. (2005) Significance analysis of functional categories in gene expression studies: a structured permutation approach. *Bioinformatics*, **21**, 1943–1949.

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The new S language: a programming environment for data analysis and graphics*. Pacific Grove, California: Wadsworth & Brooks/Cole Advanced Books & Software.

Benjamini, B. Y. and Yekutieli, D. (2001) The control of the false discovery rate in multiple testing under dependency. *The Annals of Statistics*, **29**, 1165–1188.

Berg, J. M., Tymoczko, J. L. and Stryer, L. (2002) *Biochemistry, Fifth Edition : International Version*. New York: W. H. Freeman, fifth edn.

Berkson, J. (1944) Application of the logistic function to bio-assay. *Journal of the American Statistical Association*, **39**, 357–365.

Bewick, V., Cheek, L. and Ball, J. (2004) Statistics review 12: Survival analysis. *Critical Care*, **8**, 389–394.

Black, M. A. (2002) *Statistical Issues in the Design and Analysis of Spotted Microarray Experiments*. Ph.D. thesis, Department of Statistics, Purdue University, USA.

Blanchard, A. P., Kaiser, R. J. and Hood, L. E. (1996) High-density oligonucleotide arrays. *Biosensors and Bioelectronics*, **11**, 687–690.

Bolstad, B. (2008) Preprocessing and Normalization for Affymetrix GeneChip Expression Microarrays. In *Methods in Microarray Normalization* (ed. P. Stafford), 41–59. Boca Raton: CRC Press.

Bolstad, B., Irizarry, R., Astrand, M. and Speed, T. (2003) A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, **19**, 185–193.

Bouchard, G. and Celeux, G. (2007) Model selection in supervised classification. URL `http://hal.inria.fr/inria-00070612/en/`.

Breiman, L., Friedman, J. H., olshen, R. and Stone, C. J. (1984) *Classification and Regression Trees*. Belmont, California: Wadsworth.

Brenner, S., Meselson, M. and Jacob, F. (1961) An unstable intermediate carrying information from genes to ribosomes for protein synthesis. *Nature*, **190**, 576.

Brown, M. P., Grundy, W. N., Lin, D., Cristianini, N., Sugnet, C. W., Furey, T. S., Ares, M. and Haussler, D. (2000) Knowledge-based analysis of microarray gene expression data using support vector machines. *Proceedings of the National Academy of Science. USA*, **97**, 262–267.

Burnham, K. P. and Anderson, D. R. (2004) Multimodel inference - understanding AIC and BIC in model selection. *Sociological Methods & Research*, **33**, 261–304.

Cawley, G. C. and Talbot, N. L. C. (2006) Gene selection in cancer classification using sparse logistic regression with Bayesian regularization. *Bioinformatics*, **22**, 2348–2355.

Celeux, G. (2005) AIC, BIC and other model selection criteria. In *Proceedings of the 11th monthly meeting of the MONOLIX group*. The MONOLIX group. URL `http://group.monolix.org/documents/expose_celeux.ps`.

Chang, J. T. and Nevins, J. R. (2006) GATHER: a systems approach to interpreting genomic signatures. *Bioinformatics*, **22**, 2926–2933.

Cheadle, C., Vawter, M. P., Freed, W. J. and Becker, K. G. (2003) Analysis of Microarray Data using Z Score Transformation. *Journal of Molecular Diagnostics*, **5**, 73–81.

Chen, J. J., Tsai, C.-A., Tzeng, S. and Chen, C.-H. (2007) Gene selection with multiple ordering criteria. *BMC Bioinformatics*, **8**, 74.

Chen, J. J., Wang, S.-J., Tsai, C.-A. and Lin, C.-J. (2007) Selection of differentially expressed genes in microarray data analysis. *Pharmacogenomics Journal*, **7**, 212–220.

Chilingaryan, A., Gevorgyan, N., Vardanyan, A., Jones, D. and Szabo, A. (2002) Multivariate approach for selecting sets of differentially expressed genes. *Mathematical Biosciences*, **176**, 59–69.

Cho, S.-B. (2002) Exploring features and classifiers to classify gene expression profiles of acute leukemia. *International Journal of Pattern Recognition & Artificial Intelligence*, **16**, 831–844.

Cho, S.-B. and Won, H.-H. (2003) Machine learning in DNA microarray analysis for cancer classification. In *First Asia-Pacific Bioinformatics Conference (APBC2003)* (ed. Y.-P. P. Chen), vol. 19 of *CRPIT*, 189–198. Adelaide, Australia: ACS.

Chu, F. and Wang, L. P. (2005) Applications of support vector machines to cancer classification with microarray data. *International Journal of Neural Systems*, **15**, 475–484.

Chu, S., DeRisi, J., Eisen, M., Mulholland, J., Botstein, D., Brown, P. O. and Herskowitz, I. (1998) The Transcriptional Program of Sporulation in Budding Yeast. *Scince*, **282**, 699–705.

Cleveland, W. S. (1979) Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, **74**, 829–836.

Cleveland, W. S., Grosse, E. and Shyu, W. M. (1992) Local Regression Models. In *Statistical Models in S* (eds. J. M. Chambers and T. Hastie), 309–376. New York: Chapman and Hall.

Cooley, P., Hinson, D., Trost, H.-J., Antohe, B. and Wallace, D. (2001) Ink-Jet-Deposited Microspot Arrays of DNA and Other Bioactive Molecules. In *Methods in Molecular Biology Volume 170: DNA Arrays Methods and Protocols* (ed. J. B. Rampal), 117–130. Totowa, New Jersey: Human Press Inc.

Crick, F. (1966) On protein synthesis. *Symposium of the Society for Experimental Biology*, **31**, 3–9.

Cristianini, N. and Shawe-Taylor, J. (2000) *An introduction to support vector machines and other kernel-based learning methods*. Cambridge: Cambridge University Press.

Cui, X. Q. and Churchill, G. A. (2003) Statistical tests for differential expression in cDNA microarray experiments. *Genome Biology*, **4**, 210.1–210.10.

Curtis, R. K., Oresic, M. and Vidal-Puig, A. (2005) Pathways to the analysis of microarray data. *Trends Biotechnology*, **23**, 429–435.

Day, N. E. and Kerridge, D. F. (1967) A general maximum likelihood discriminant. *Biometrics*, **23**, 313–323.

Debouck, C. and Goodfellow, P. N. (1999) DNA microarrays in drug discovery and development. *Nature Genetics*, **21**, 48–50.

DeLisi, C. and Smith, D. A. (eds.) (1985) *The First Santa Fe Conference to Assess the Feasibility of a Human Genome Initiative*.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977) Maximum Likelihood from Incomplete Data via EM Algorithm. *Journal of the Royal Statistical Society Series B-Methodological*, **39**, 1–38.

Dudoit, S., Fridlyand, J. and Speed, T. P. (2002a) Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, **97**, 77–87.

Dudoit, S., Yang, Y. H., Callow, M. J. and Speed, T. P. (2002b) Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. *Statistica Sinica*, **12**, 111–140.

Duggan, D. J., Bittner, M., Chen, Y. D., Meltzer, P. and Trent, J. M. (1999) Expression profiling using cDNA microarrays. *Nature Genetics*, **21**, 10–14.

Edgar, R., Domrachev, M. and Lash, A. E. (2002) Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Research*, **30**, 207–210.

Edgington, E. S. (1980) *Randomization Tests*. New York: Marcel Dekker, Inc.

Efron, B. (1979) Bootstrap methods: another look at the jackknife. *Annals of Statistics*, **7**, 1–26.

— (1981a) nonparametric standard errors and confidence intervals. (with discussion). *Canadian Journal of Statistics*, **9**, 139–172.

— (1981b) Nonparametric estimates of standard error: the jackknife, the bootstrap, and other methods. *Biometrika*, **68**, 589–599.

— (1983) Estimating the error rate of a prediction rule - improvement on cross-validation. *Journal of the American Statistical Association*, **78**, 316–331.

Efron, B. and Tibshirani, R. J. (1993) *An Introduction to the Bootstrap*. New York, London: Chapman & Hall.

— (1997) Improvements on cross-validation: The .632+ bootstrap method. *Journal of the American Statistical Association*, **92**, 548–560.

Eickhoff, H., Schneider, U., Nordhoff, E., Nyarsik, L., Zenetner, G., Nietfeld, W. and Lehrach, H. (2002) Technology development for DNA chips. In *DNA Arrays Technologies and Experimental Strategies* (ed. E. Grigorenko), chap. 1. Boca Raton, Florida: CRC Press LLC.

Ein-Dor, L., Kela, I., Getz, G., Givol, D. and Domany, E. (2005) Outcome signature genes in breast cancer: is there a unique set? *Bioinformatics*, **21**, 171–178.

Eisen, M. B. and Brown, P. O. (1999) DNA arrays for analysis of gene expression. *CDNA preparation and characterization methods in enzymology*, **303**, 179–205.

Elston, C. W. and Ellis, I. O. (1991) Pathological Prognostic Factors in Breast-Cancer .1. The Value of Histological Grage in Breast-Cancer - Experience from a Large Study with Long-Term Follow-up. *Histopathology*, **19**, 403–410.

Everitt, B. S., Landau, S. and Leese, M. (2001) *Cluster Analysis*. London, Arnold, New York: Oxford University Press, fourth edn.

Fan, J. and Ren, Y. (2006) Statistical analysis of DNA microarray data in cancer research. *Clinical Cancer Research*, **12**, 4469–4473.

Fisher, R. A. (1935) *The Design of Experiments*. Edinburgh: Oliver & Boyd.

— (1936) The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, **7**, 179–188.

Fix, E. and Hodges, J. (1951) Discriminatory Analysis, Nonparametric Discrimination: Consistency Properties. *Tech. rep.*, Technical report, U.S. Air Force, School of Aviation Medicine.

Fodor, S. P. A., Read, J. L., Pirrung, M. C., Stryer, L., Lu, A. T. and Solas, D. (1991) Light-directed, spatially addressable parallel chemical synthesis. *Science*, **251**, 767–773.

Fraley, C. and Raftery, A. E. (2002) Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, **97**, 611–631.

— (2006) *mclust: Model-Based Clustering / Normal Mixture Modeling*. URL `http://www.stat.washington.edu/mclust`. R package version 3.0-0.

Furey, T. S., Cristianini, N., Duffy, N., Bednarski, D. W., Schummer, M. and Haussler, D. (2000) Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, **16**, 906–914.

Gautier, L., Cope, L., Bolstad, B. and Irizarry, R. (2004) affy - analysis of Affymetrix GeneChip data at the probe level. *Bioinformatics*, **20**, 307–315.

Gentleman, R. C., Carey, V. J., Bates, D. M., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge, Y., Gentry, J., Hornik, K., Hothorn, T., Huber, W., Iacus, S., Irizarry, R., Li, F. L. C., Maechler, M., Rossini, A. J., Sawitzki, G., Smith, C., Smyth, G., Tierney, L., Yang, J. Y. H. and Zhang, J. (2004) Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology*, **5**, R80. URL `http://genomebiology.com/2004/5/10/R80`.

Gerhold, D. and Caskey, C. T. (1996) It's the genes! EST access to human genome content. *BioEssays*, **18**, 973–981.

Goeman, J. J., van de Geer, S. A., de Kort, F. and van Houwelingen, H. C. (2004) A global test for groups of genes: testing association with a clinical outcome. *Bioinformatics*, **20**, 93–99.

Golub, G. H. and Van Loan, C. F. (1983) *Matrix-computations*. Oxford: North Oxford Academic.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., Bloomfield, C. D. and Lander, E. S. (1999) Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, **286**, 531–537.

Good, P. (2000) *Permutation Tests: A Practical Guide to Resampling Methods for Testing Hypotheses*. New York: Springer, 2 edn.

Gordon, A. D. (1981) *Classification: Methods for the Exploratory Analysis of Multivariate Data*. London, New York: Chapman and Hall.

Gosset, W. S. (1908) The probable error of a mean. *Biometrika*, **6**, 1–25.

Gower, A. D. (1966) Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika*, **53**, 325–338.

Grigorenko, E. V. (ed.) (2002) *DNA Arrays Technologies and Experimental Strategies*. Boca Raton, Florida: CRC Press LLC.

Gunderson, K. L., Kruglyak, S., Graige, M. S., Garcia, F., Kermani, B. G., Zhao, C., Che, D., Dickinson, T., Wickham, E., Bierle, J., Doucet, D., Milewski, M., Yang, R., Siegmund, C., Hass, J., Zhou, L., Oliphant, A., Fan, J., Barnard, S. and Chee, M. S. (2004) Decoding randomly ordered DNA arrays. *Genome Research*, **14**, 870–877.

Hanczar, B., Courtine, M., Benis, A., Hennegar, C., Clement, K. and Zucker, J. D. (2003) Improving classification of microarray data using prototype-based feature selection. *SIGKDD Explorer Newsletter*, **5**, 23–30.

Hand, D. J. (1981) *Discrimination and Classification*. Chichester [Eng.], New York: John Wiley & Sons Ltd.

Harrell Jr, F. E., Dupont, C. and with contributions from many other users. (2009) *Hmisc: Harrell Miscellaneous*. URL `http://CRAN.R-project.org/package=Hmisc`. R package version 3.7-0.

Hartigan, J. A. (1975) *Clustering Algorithms*. New York: Wiley.

Hartigan, J. A. and Wong, M. A. (1979) A k-means clustering algorithm. *Applied Statistics*, **28**, 100–108.

Hastie, T. and Tibshirani, R. (2004) Efficient quadratic regularization for expression arrays. *Biostatistics*, **5**, 329–340.

Hastie, T., Tibshirani, R. and Friedman, J. (2001) *The Elements of Statistical Learning Data Mining, Inference, and Prediction*. New York: Springer.

Hedenfalk, I., Duggan, D., Chen, Y. D., Radmacher, M., Bittner, M., Simon, R., Meltzer, P., Gusterson, B., Esteller, M., Kallioniemi, O. P., Wilfond, B., Borg, A., Trent, J., Raffeld, M., Yakhini, Z., Ben-Dor, A., Dougherty, E., Kononen, J., Bubendorf, L., Fehrle, W., Pittaluga, S., Gruvberger, S., Loman, N., Johannsoson, O., Olsson, H. and Sauter, G. (2001) Gene-expression profiles in hereditary breast cancer. *New England Journal of Medicine*, **344**, 539–548.

Hotelling, H. (1933) Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, **24**, 417–441,498–520.

Hu, H., Li, J., Wang, H. and Daggard, G. (2006) Combined gene selection methods for microarray data analysis. *Knowledge-based intelligent information and engineering systems, PT 1, proceedings Lecture notes in Artificial Intelligence*, **4251**, 976–983.

Hubbell, E., Liu, W.-M. and Mei, R. (2002) Robust estimators for expression analysis. *Bioinformatics*, **18**, 1585–1592.

Ihaka, R. and Gentleman, R. (1996) R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, **5**, 299–314.

Irizarry, R. A., Bolstad, B. M., Collin, F., Cope, L. M., Hobbs, B. and Speed, T. P. (2003a) Summaries of Affymetrix GeneChip probe level data. *Nucleic Acids Research*, **31**, e15.

Irizarry, R. A., Hobbs, B., Collin, F., Beazer-Barclay, Y. D., Antonellis, K. J., Scherf, U. and Speed, T. P. (2003b) Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, **4**, 249–264.

Ivshina, A. V., George, J., Senko, O., Mow, B., Putti, T. C., Smeds, J., Lindahl, T., Pawitan, Y., Hall, P., Nordgren, H., Wong, J. E., Liu, E. T., Bergh, J., Kuznetsov, V. A. and Miller, L. D. (2006) Genetic reclassification of histologic grade delineates new clinical subtypes of breast cancer. *Cancer Research*, **66**, 10292–10301.

Jacobson, J. W., Oliver, K. G., Weiss, C. and Kettman, J. (2006) Analysis of individual data from bead-based assays ("bead arrays"). *Cytometry Part A*, **69A**, 384–390.

Jaeger, J., Sengupta, R. and Ruzzo, W. L. (2003) Improved gene selection for classification of microarrays. *Pacific Symposium on Biocomputing*, **8**, 53–64.

Jiang, Z. and Gentleman, R. (2007) Extensions to gene set enricment. *Bioinformatics*, **23**, 2028–2036.

Jirapech-Umpai, T. and Aitken, S. (2005) Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes. *BMC Bioinformatics*, **6**, 148.

John, T., Black, M. A., Toro, T. T., Leader, D., Gedye, C. A., Davis, I. D., Guilford, P. J. and Cebon, J. S. (2008) Predicting clinical outcome through molecular profiling in stage III melanoma. *Clinical Cancer Research*, **14(16)**, 5173–5180.

Jolliffe, I. T. (1972) Discarding variables in a principal component analysis I: artificial data. *Applied Statistics*, **21**, 373–374.

— (1986) *Principal Component Analysis*. New York: Springer-Verlag.

Kanehisa, M. (1997) A database for post-genome analysis. *Trends Genet.*, **13**, 375–376.

Kanehisa, M., Araki, M., Goto, S., Hattori, M., Hirakawa, M., Itoh, M., Katayama, T., Kawashima, S., Okuda, S., Tokimatsu, T. and Yamanishi, Y. (2008) KEGG for linking genomes to life and the environment. *Nucleic Acids Research*, **36**, D480–484.

Kanehisa, M. and Goto, S. (2000) KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res.*, **28**, 27–30.

Kanehisa, M., Goto, S., Hattori, M., Aoki-Kinoshita, K. F., Itoh, M., Kawashima, S., Katayama, T., Araki, M. and Hirakawa, M. (2006) From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Research*, **34**, D354–357.

Kaplan, E. L. and Meier, P. (1958) Nonparametric-Estimation fron Incomplete Observations. *Journal of the American Statistical Association*, **53**, 457–481.

Kaufman, L. and Rousseeuw, P. J. (1990) *Finding Groups in Data. An Introduction to Cluster Analysis*. New York: Wiley-Interscience.

Kendziorski, C., Newton, M., Lan, H. and Gould, M. (2003) On parametric empirical Bayes methods for comparing multiple groups using replicated gene expression profiles. *Statistics in Medicine*, **22**, 3899–3914.

Kendziorski, C., Newton, M. and Sarkar, D. (2006) *EBarrays: Empirical Bayes for Microarrays*. R package version 1.6.0.

Kerr, M. K., Martin, M. and Churchill, G. A. (2000) Analysis of variance for gene expression microarray data. *Journal of Computational Biology*, **7**, 819–837.

Kieseppa, I. A. (2001) Statistical model selection criteria and bayesianism. *Philosophy of Science*, **68**, S141–S152.

Kim, S. Y. and Volsky, D. J. (2005) PAGE: parametric analysis of gene set enrichment. *BMC Bioinformatics*, **6**, 144.

Kong, S. W., Pu, W. T. and Park, P. J. (2006) A multivariate approach for integrating genome-wide expression data and biological knowledge. *Bioinformatics*, **22**, 2373–2380.

Lai, C., Reinders, M. J. T., van't Veer, L. J. and Wessels, L. F. A. (2006) A comparison of univariate and multivariate gene selection techniques for classification of cancer datasets. *BMC Bioinformatics*, **7**, 235.

Lamb, J., Ramaswamy, S., Ford, H. L., Contreras, B., Martinez, R. V., Kittrell, F. S., Zahnow, C. A., Patterson, N., Golub, T. R. and Ewen, M. E. (2003) A mechanism of cyclin D1 action encoded in the patterns of gene expression in human cancer. *Cell*, **114**, 323–334.

Lee, E. T. and Go, O. T. (1997) Survival analysis in public health research. *Annual Review of Public Health*, **18**, 105–134.

Leung, Y. F. and Cavalieri, D. (2003) Fundamentals of cDNA microarray data analysis. *Trends Genetic*, **19**, 649–659.

Lewin, B. (1997) *Genes VI*. Oxford, New York, Tokyo: Oxford University Press.

Li, C. and Wong, W. H. (2001) Model-based analysis of oligonucleotide arrays: Expression index computation and outlier detection. *Proceedings of the National Academy of Sciences*, **98**, 31–36.

Li, W. T. and Nyholt, D. R. (2001) Marker selection by Akaike information criterion and Bayesian information criterion. *Genetic Epidemiology*, **21**, S272–S277.

Lim, H. A. (2002) *Genetically Yours, Bioinforming, Biopharming, Biofarming.* River Edge, NJ: World Scientific Publishing Co. Pte. Ltd.

Liu, T.-Y., Lin, C., Falcon, S., Zhang, J. and MacDonald, J. W. (2006) *KEGG: A data package containing annotation data for KEGG.* R package version 1.14.1.

Loi, S., Haibe-Kains, B., Desmedt, C., Lallemand, F., Tutt, A. M., Gillet, C., Ellis, P., Harris, A., Bergh, J., Foekens, J. A., Klijn, J. G. M., Larsimont, D., Buyse, M., Bontempi, G., Delorenzi, M., Piccart, M. J. and Sotiriou, C. (2007) Definition of clinically distinct molecular subtypes in estrogen receptor-positive breast carcinomas through genomic grade. *Journal of Clinical Oncology*, **25**, 1239–1246.

Lu, Y., Liu, P. Y., Xiao, P. and Deng, H. W. (2005) Hotelling's T^2 multivariate profiling for detecting differential expression in microarrays. *Bioinformatics*, **21**, 3105–3113.

Luo, F., Yang, Y., Zhong, J., Gao, H., Khan, L., Thompson, D. K. and Zhou, J. (2007) Constructing gene co-expression networks and predicting functions of unknown genes by random matrix theory. *BMC Bioinformatics*, **8**, 299.

Ma, J. W., Li, F. H. and Liu, J. F. (2005) Non-parametric statistical tests for informative gene selection. *Advances in Neural Networks - ISNN 2005, PT 3, Proceedings Lecture Notes in Computer Science*, **3498**, 697–702.

MacNaughton-Smith, P. (1965) Some statistical and other numerical techniques for classifying individuals. *Report 6*, Home Office research Unit. HMSO London.

MacNaughton-Smith, P., Williams, W. T., Dale, M. B. and Mockett, L. G. (1964) Dissimilarity analysis. *Nature*, **202**, 1034–1035.

MacQueen, J. (1967) Some methods for classification and analysis of multivatiate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability* (eds. L. LeCam and J. Neyman), 281–297. Berkeley, CA, University of California Press. Volume 1.

Maglott, D., Ostell, J., D., P. K. and Tatusova, T. (2005) Entrez gene: gene-centered information at ncbi. *Nucleic Acids Research*, **33**, D54–58.

Mann, H. B. and Whitney, D. R. (1947) On a Test of Whether One of 2 Random Variables is Stochastically Larger than the Other. *Annals of Mathematical Statistics*, **18**, 50–60.

Mantel, N. and Haenszel, W. (1959) Statistical Aspects of the Analysis of Data from Retrospective Studies of Disease. *Journal of the National Cancer Institute*, **22**, 719–748.

Mardia, K. V., Kent, J. T. and Bibby, J. M. (1979) *Multivariate analysis*. London: Academic Press.

Mardis, E. R. (2008) The impact of next-generation sequencing technology on genetics. *Trends in Genetics*, **24**, 133–141.

Marra, M. A., Hiller, L. and Waterston, R. H. (1998) Expressed sequence tags - establishing bridges between genomes. *Trends in Genetics*, **14**, 4–7.

Massey, F. J. (1951) The Kolmogorov-Smirnov Test for Goodness of Fit. *Journal of the American Statistical Association*, **46**, 68–78.

McArdle, B. H. (2001) *Multivariate Analysis: A Practical Guide for Biologists*. unpublished.

McCullagh, P. and Nelder, J. A. (1989) *Generalized Linear Models*. London: Chapman & Hall, second edn.

McCulloch, W. and Pitts, W. (1943) A logical calculus of the ideas imment in nervous activity. *Bulletin of Mathematical Biophysiology*, **5**, 115–133.

McGall, G. H. and Fidanza, J. A. (2001) Photolithographic Synthesis of High-Density Oligonucleotide Arrays. In *Methods in Molecular Biology Volume 170: DNA Arrays Methods and Protocols* (ed. J. B. Rampal), 71–102. Totowa, New Jersey: Human Press Inc.

McLachlan, G. J., Do, K. A. and Ambroise, C. (2004) *Analyzing microarray gene expression data*. Hoboken, New Jersey: Wiley-Interscience.

McNemar, Q. (1947) Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, **12**, 153–157.

Miescher, J. F. (1871) Ueber die chemische zusammensetzung der eiterzellen. *Hoppe-Seyler's medicinisch-chemische Untersuchungen*, **4**, 441–460.

Molinaro, A. M., Simon, R. and Pfeiffer, R. M. (2005) Prediction error estimation: a comparison of resampling methods. *Bioinformatics*, **21**, 3301–3307.

Mootha, V. K., Lindgren, C. M., Eriksson, K. F., Subramanian, A., Sihag, S., Lehar, J., Puigserver, P., Carlsson, E., Ridderstrale, M., Laurila, E., Houstis, N., Daly, M. J., Patterson, N., Mesirov, J. P., Golub, T. R., Tamayo, P., Spiegelman, B., Lander, E. S., Hirschhorn, J. N., Altshuler, D. and Groop, L. C. (2003) PGC-1 alpha-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes. *Nature Genetics*, **34**, 267–273.

Naciff, J. M., Richardson, B. D., Oliver, K. G., Jump, M. L., Torontali, S. M., Juhlin, K. D., Carr, G. J., Paine, J. R., Tiesman, J. P. and Daston, G. P. (2005) Design of a microsphere-based high-throughput gene expression assay to determine estrogenic potential. *Environmental Health Perspectives*, **113**, 1164–1171.

NCBI (2002) *NCBI Online Handbook*. National Library of Medicine. URL `http://www.ncbi.nlm.nih.gov/books/bv.fcgi?rid=handbook.glossary.1237`.

Nguyen, D. V. and Rocke, D. M. (2002) Tumor classification by partial least squares using microarray gene expression data. *Bioinformatics*, **18**, 39–50.

Novak, B. A. and Jain, A. N. (2006) Pathway recognition and augmentation by computational analysis of microarray expression data. *Bioinformatics*, **22**, 233–241.

Nykter, M., Aho, T., Ahdesmäki, M., Ruusuvuori, P., Lehmussola, A. and Yli-Harja, O. (2006) Simulation of microarray data with realistic characteristics. *BMC Bioinformatics*, **7**, 349.

Oakes, D. (2001) Biometrika centenary: Survival analysis. *Biometrika*, **88**, 99–142.

Pearson, K. (1901) On lines and planes of closest fit to systems of points in space. *Philosophical Magazine, Sixth Series*, **2**, 559–572.

Pesarin, F. (2001) *Multivariate Permutation Tests With Applications in Biostatistics*. Chichester, New York: John Wiley & Sons Ltd.

Peto, R. and Peto, J. (1972) Asymptotically Efficient Rank Invariant Test Procedures. *Journal Of The Royal Statistical Society Series A-General*, **135**, 185–207.

Pham, T. D., Wells, C. and Crane, D. I. (2006) Analysis of microarray gene expression data. *Current Bioinformatics*, **1**, 37–53.

Pitman, E. J. G. (1937a) Significance tests which may be applied to samples from any population. *Journal of the Royal Statistical Society Supplement*, **4**, 119–130.

— (1937b) Significance tests which may be applied to samples from any population. Part II. *Journal of the Royal Statistical Society Supplement*, **4**, 225–232.

— (1938) Significance tests which may be applied to samples from any population. Part III The analysis of variance test. *Biometrika*, **29**, 322–335.

Quackenbush, J. (2001) Computational genetics: computational analysis of microarray data. *Nature Reviews Genetics*, **2**, 418–427.

Quenouille, M. (1949) Approximate tests of correlation in time series. *Journal of the Royal Statistial Society B*, **11**, 18–44.

R Development Core Team (2005) *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. URL `http://www.R-project.org`. ISBN 3-900051-07-0.

Ramaswamy, S., Tamayo, P., Rifkin, R., Mukherjee, S., Yeang, C. H., Angelo, M., Ladd, C., Reich, M., Latulippe, E. Mesirov, J. P., Poggio, T., Gerald, W., Loda, M., Lander, E. S. and Golub, T. R. (2001) Multiclass cancer diagnosis using tumor gene expression signatures. *Proceedings of the National Academy of Science. USA*, **98**, 15149–15154.

Ransohoff, D. (2004) Opinion - Rules of evidence for cancer molecular-marker discovery and validation. *Nantur Reviews Cancer*, **4**, 309–314.

Ross, D. T., Scherf, U., Eisen, M. B., Perou, C. M., Spellman, P., Iyer, V., Jeffrey, S. S., de Rijn, M. V., Waltham, M., Pergamenschikov, A., Lee, J. C. F., Lashkari, D., Shalon, D., Myers, T. G., Weinstein, J. N., Bostein, D. and Brown, P. O. (2000) Systematic variation in gene expression patterns in human cancer cell lines. *Nature Genetics*, **24**, 227–234.

Ruan, Y., Gilmore, J. and Conner, T. (1998) Towards Arabidopsis genome analysis: monitoring expression profiles of 1400 genes using cDNA microarrays. *Plant Journal*, **15**, 821–833.

Sasaki, T., Song, J. Y., Kogaban, Y., Matsui, E., Fang, F., Higo, H., Nagasaki, H., Hori, M., Miya, M., Murayamakayano, E., Takiguchi, T., Takasuga, A., Niki, T., Ishimaru, K., Ikeda, H., Yamamoto, Y., Mukai, Y., Ohta, I., Miyadera, N., Havukkala, I. and Minobe, Y. (1994) Toward cataloging all rice genes - large-scale sequencing of randomly chosen rice cDNAs from a callus cDNA library. *Plant Journal*, **6**, 615–624.

Saxena, V., Orgill, D. and Kohane, I. (2006) Absolute enrichment: gene set enrichment analysis for homeostatic systems. *Nucleic Acids Research*, **34**, e151.

Schema, M., Shalon, D., Davis, R. W. and Brown, P. O. (1995) Quantitative monitoring of gene-expression patterns with a complementary-DNA microarray. *Science*, **270**, 467–470.

Scherf, W., Ross, D. T., Waltham, M., Smith, L. H., Lee, J. K., Tanabe, L., Kohn, K. W., Reinhold, W. C., Myers, T. G., Andrews, D. T., Scudiero, D. A., Eisen, M. B., Sausville, E. A., Pommier, Y., Bostein, D., Brown, P. O. and Weinstein, J. N. (2000) A gene expression data base for the molecular pharmacology of cancer. *Nature Genetics*, **24**, 236–244.

Schermer, M. J. (1999) Confocal scanning in microscopy in microarray detection. In *DNA Microarrays: A Practical Approach* (ed. M. Schena), 17–42. New York: Oxford University Press.

Schoenberg, I. L. (1935) Remarks to maurice fréchet's article 'sur la définition axiomatique d'une classe d'espace distanciés vectoriellement applicable sur l'espace de hilbert'. *Annals of Mathematics, 2nd Series*, **36**, 724–732.

Schwarz, G. (1978) Estimating Dimension of a Model. *Annals of Statistics*, **6**, 461–464.

Seber, G. A. F. (1984) *Multivariate Observations*. New York: Wiley.

Segal, E., Yelensky, R. and Koller, D. (2003) Genome-wide discovery of transcriptional modules from DNA sequence and gene expression. *Bioinformatics*, **19**, i273–282.

Shalon, D., Smith, S. J. and Brown, P. O. (1996) A DNA microarray system for analyzing complex DNA samples using two-color fluorescent probe hybridization. *Genome Research*, **6**, 639–645.

Shen, R., Ghosh, D., Chinnaiyan, A. and Meng, Z. (2006) Eigengene-based linear discriminant model for tumor classification using gene expression microarray data. *Bioinformatics*, **22**, 2635–2642.

Sheskin, D. J. (1997) *Handbook of parametric and non-parametric statistical procedures*. Boca Raton: CRC Press.

Shi, J. and Walker, M. G. (2007) Gene set enrichment analysis (GSEA) for interpreting gene expression profiles. *Current Bioinformatics*, **2**, 133–137.

Simpson, J. (ed.) (2005) *Oxford English Dictionary Online*. Oxford University Press. URL `http://www.oed.com/`.

Singhal, S., Kyvernitis, C. G., Johnson, S. W., Kaiser, L. R., Liebman, M. N. and Albelda, S. M. (2003) MicroArray Data Simulator For Improved Selection of Differentially Expressed Genes. *Cancer Biology & Therapy*, **2**, 383–391.

Smyth, G. K. (2004) Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, **3**, 3. URL `http://www.bepress.com/sagmb/vol3/iss1/art3`.

— (2005) Limma: linear models for microarray data. In *Bioinformatics and Computational Biology Solutions using R and Bioconductor* (eds. R. Gentleman, V. Carey, S. Dudoit, R. Irizarry and W. Huber), 397–420. New York: Springer.

Smyth, G. K., Michaud, J. and Scott, H. (2005) The use of within-array replicate spots for assessing differential expression in microarray experiments. *Bioinformatics*, **21**, 2067–2078. URL `http://www.bepress.com/sagmb/vol3/iss1/art3`.

Smyth, G. K. and Speed, T. (2003) Normalization of cDNA microarray data. *Methods*, **31**, 265–273.

Sneath, P. H. A. (1957) The application of computers to taxonomy. *Journal of General Microbiology*, **17**, 201–226.

Sokal, R. R. and Michener, C. D. (1958) A statistical method for evaluating systematic relationships. *University of Kansas Scientific Bulleten*, **38**, 1409–1438.

Song, Q. (2009) *Microarray-based gene set analysis in cancer studies*. Ph.D. thesis, Department of Statistics, The University of Auckland, NZ.

Song, Q. and Black, M. A. (2007) *pcot2: Principle Coordinates and Hotelling's T-Square method*. URL `http://bioconductor.org/packages/2.1/bioc/vignettes/pcot2/inst/doc/pcot2.pdf`. R package version 1.2.0.

Sonquist, J. A. and Morgan, J. N. (1963) Problems in the analysis of survey data and a proposal. *Journal of the American Statistical Association*, **58**, 415–435.

Sorensen, T. (1948) A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analysis of the vegetation on danish commons. *Biologiske Skrifter*, **5**, 1–34.

Sotiriou, C., Wirapati, P., Loi, S., Harris, A., Fox, S., Smeds, J., Nordgren, H., Farmer, P., Praz, V., Haibe-Kains, B., Desmedt, C., Larsimont, D., Cardoso, F., Peterse, H., Nuyten, D., Buyse, M., Van de Vijver, M. J., Bergh, J., Piccart, M. T. and Delorenzi, M. (2006) Gene expression profiling in breast cancer: Understanding the molecular basis of histologic grade to improve prognosis. *Journal of the National Cancer Institute*, **98**, 262–272.

Southern, E. M. (2001) DNA Microarrays. In *Methods in Molecular Biology Volume 170: DNA Arrays Methods and Protocols* (ed. J. B. Rampal), 1–16. Totowa, New Jersey: Human Press Inc.

Speed, T. (ed.) (2003) *Statistical Analysis of Gene Expression Microarray Data*. Chapman & Hall/CRC CRC Press LLC.

Statnikov, A., Aliferis, C. F., Tsamardinos, I., Hardin, D. and Levy, S. (2005) A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. *Bioinformatics*, **21**, 631–643.

Steemers, F. J. and Gunderson, K. L. (2005) Illumina, Inc. *Pharnacogenomics*, **6**, 777–782.

Stekel, D. (2003) *Microarray Bioinformatics*. New York: Cambridge University Press.

Su, Y., Murali, T. M., Pavlovic, V., Schaffer, M. and Kasif, S. (2003) RankGene: identification of diagnostic genes based on expression data. *Bioinformatics Application Notes*, **19**, 1578–1579.

Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., Paulovich, A., Pomeroy, S. L., Golub, T. R., Lander, E. S. and Mesirov, J. P. (2005) Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceeding of the National Academy of Sciences of the United States of Anerica*, **102**, 15545–15550.

Svensson, J. P., Stalpers, L. J. A. Esveldt-van Lange, R. E. E., Franken, N. A. P., Haveman, J., Klein, B., Turesson, I., Vrieling, H. and Giphart-Gassler, M. (2006) Analysis of gene expression using gene sets discriminates cancer patients with and without late radiation toxicity. *Public Libraray of Science Medicine*, **3**, 1904–1914.

Szabo, A., Bouncher, K., Carroll, W. L., Klebanov, L. B., Tsodikov, A. D. and Yakovlev, A. Y. (2002) Variable selection and pattern recognition with gene expression data generated by the microarray technology. *Mathematical Biosciences*, **176**, 71–98.

Therneau, T., Grambsch, P. and Fleming, T. (1990) Martingale based residuals for survival models. *Biometrika*, **77**, 147–160.

Therneau, T. and Lumley, T. (2007) *survival: Survival analysis, including penalised likelihood.* R package version 2.31.

Tibshirani, R., Hastie, T., Narasimhan, B. and Chu, G. (2002) Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proceedings of the National Academy of Sciences of the United States of America*, **99**, 6567–6572.

Torgerson, W. S. (1952) Multidimensional scaling: I. theory and method. *Psychometrika*, **17**, 401–419.

Tukey, J. W. (1958) Bias and confidence in not quite large samples. (abstract). *Annals of Mathematics and Statistics*, **29**, 614.

Tusher, V. G., Tibshirani, R. and Chu, G. (2001) Significance analysis of microarrays applied to the ionizing radiation response. *Proceedings of the National Academy of Sciences*, **98**, 5116–5121.

Vapnik, V. N. (1995) *The nature of statistical learning theory*. New York: Springer.

— (1998) *Statistical learning theory*. New York: Wiley.

Varma, S. and Simon, R. (2006) Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics*, **7**, 91.

Venables, W. N. and Ripley, B. D. (1999) *Modern applied statistics with S-PLUS*. New York: Springer-Verlag Inc, third edn.

— (2002) *Modern Applied Statistics with S*. New York: Springer, fourth edn. URL `http://www.stats.ox.ac.uk/pub/MASS4`. ISBN 0-387-95457-0.

Wang, A. and Gehan, E. D. (2005) Gene selection for microarray data analysis using principal component analysis. *Statistics In Medicine*, **24**, 2069–2087.

Wang, L., Zhang, B., Wolfinger, R. D. and Chen, X. (2008) An Integrated Approach for the Analysis of Biological Pathways using Mixed Models. *PLoS Genetics*, **4**.

Wang, Y., Jatkoe, T., Zhang, Y., Mutch, M. G., Talantov, D., Jiang, J., McLeod, H. L. and Atkins, D. (2004) Gene Expression Profiles and Molecular Markers To Predict Recurrence of Dukes' B Colon Cancer. *Journal of Clinical Oncology*, **22**, 1564–1570.

Wang, Y., Klijn, J. G. M., Zhang, Y., Sieuwerts, A. M., Look, M. P., Yang, F., Talantov, D., Timmermans, M., Meijer-van Gelder, M. E., Yu, J., Jatkoe, T., Berns, E. M. J. J., Atkins, D. and Foekens, J. A. (2005a) Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. *The Lancet*, **365**, 671–679.

Wang, Y., Makedon, F. S., Ford, J. C. and Pearlman, J. (2005b) HykGene: a hybrid approach for selecting marker genes for phenotype classification using microarray gene expression data. *Bioinformatics*, **21**, 1530–1537.

Ward, J. H. (1963) Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, **58**, 236–244.

Weakliem, D. L. (1999) A critique of the Bayesian information criterion for model selection. *Sociological Methods & Research*, **27**, 359–397.

White, K. P., Rifkin, S. A., Hurban, P. and Hogness, D. S. (1999) Microarray analysis of Drosophila development during metamorphosis. *Science*, **286**, 2179–2184.

Widrow, B. and Hoff, M. (1960) Adaptive switching curcuits. *IRE WESCON Convention record*, **4**, 96–104.

Wilcoxon, F. (1945) Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, **1**, 80–83.

Wild, C. J. and Seber, G. A. F. (1993) Comparing 2 proportions from the same survey. *American Statistician*, **47**, 178–181.

Wilson, E. B. (1927) Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, **22**, 209–212.

Wold, B. and Myers, R. M. (2008) Sequence census methods for functional genomics. *Nature Methods*, **5**, 19–21.

Wolfinger, R. D., Gibson, G., Wolfinger, E. D., Bennett, L., Hamadeh, H., Bushel, P., Afshari, C. and Paules, R. S. (2001) Assessing gene significance from cdna microarray expression data via mixed models. *Journal of Computational Biology*, **8**, 625–637.

Wolfsberg, T. G. and Landsman, D. (2001) Expressed sequence tags (ESTs). In *Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins* (eds. A. D. Baxevanis and B. F. F. Ouellette), 283–301. New York: Wiley-Liss.

Wu, Z., Irizarry, R. A., Gentleman, R., Martinez-Murillo, F. and Spencer, F. (2004) A Model-Based Background Adjustment for Oligonucleotide Expression Arrays. *Journal of the American Statistical Association*, **99**, 909–917.

Yang, L., Tran, D. K. and Wang, X. (2001) BADGE, BeadsArray for the detection of gene expression, a high-throughput diagnostic bioassay. *Genome Research*, **11**, 1888–1898.

Young, F. W. and Hamer, R. M. (1987) *Mulitdimensional Scaling: History, Theory, and Applications*. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Young, G. and Householder, A. S. (1938) Discussion of a set of points in terms of mutual distances. *Psychometrika*, **3**, 19–22.

Zhang, J., Wang, Y. J., Khan, J. and Clarke, R. (2004) Gene selection in class space for molecular classification of cancer. *Science in China Series F: Information Sciences*, **47**, 301–314.

Zhu, J. and Hastie, T. (2004) Classification of gene microarrays by penalized logistic regression. *Biostatistics*, **5**, 427–443.