Ensemble Learning by Data Resampling

Michael Goebel

A thesis submitted in partial fulfilment of the requirements for
the degree of Doctor of Philosophy in Computer Science,
The University of Auckland, 2004

Abstract

We investigate ensemble learning methods that construct a classifier ensemble by repeatedly sampling the original training data and building a member classifier from each subsample. We find that the performance of standard Bagging can frequently be improved upon by simple variations of the sampling scheme, such as varying the sample size, or sampling without replacement instead of sampling with replacement.

For all methods tested, the ensemble performance is greatly dependent on properties of the problem domain and the data sample. We try to explain the observed performances of the various ensemble methods qualitatively and quantitatively, and find that current ensemble analysis methods such as margin distributions, κ-Error Diagrams, bias-variance decomposition etc. are not well suited for this task.

We postulate that the primary explanation for the performance of ensemble methods is to be found in their effects on accuracy of the ensemble members on one side and diversity among them on the other side, two contradictory goals which necessitate a compromise, or trade-off. This motivates the presentation of a precise yet general definition of what diversity is and how it is to be measured. This definition has the desirable property that it is applicable to all single-stage voting ensembles, under any given loss function.

We then study the mathematical relationships between ensemble loss, mean member loss, and diversity. For squared loss, we show that our definitions lead to the well known ensemble loss decomposition, and extend this decomposition to the case where the ensemble members, instead of a real number, return a probability distribution over $\mathcal{R}$.

For the case of 0-1 loss, we derive the exact mathematical relations between ensemble loss, mean member loss, and diversity. Studying those relations provides some valuable insights into ensembles behavior, and produces some unexpected hence interesting results. These results are also confirmed by the experimental observations.

Turning our attention back to the performance of Bagging variants, we show how the loss decomposition can be used to reduce the number of parameter settings which have to be tried out experimentally in order to find a well-performing ensemble method for a given particular problem.

Acknowledgments

I am greatly indebted to my supervisory committee, Pat Riddle, Mike Barley, and Hans Guesgen, for innumerable helpful comments, as well as for their general guidance and support.

Another big 'Thank you' to all those who provided valuable comments on earlier versions of this thesis, especially to Remco Bouckaert.

Special thanks also goes to the Department of Computer Science at the University of Auckland for their financial, technical, and administrative support.

Appendix A lists those who provided the datasets used in the experiments.

Credits are also due to all those who contribute their time and energy to produce all these wonderful free ('free' as in free speech) software packages used for the conduct of the research as well as for the production of this thesis – you are way too many to mention individually, but way too important not to mention at all.

Lastly, to my parents, Rotraut and Manfred, and to my fiancee, María Cristina, as well as to her parents, María del Carmen and José Carlos, for their extraordinary patience, love, and support.

This would not have been possible without you.

Contents

List of Tables

List of Figures

Abbreviations

BVD	Bias-Variance Decomposition
Bagging	Bootstrap Aggregating
Cragging	Cross-Validation Aggregating
ECOC	Error-Correcting Output Coding
i.i.d.	independently and identically distributed
iff	if and only if
MCMC	Markov Chain Monte Carlo
vs.	versus
w.r.t.	with respect to

Notation

$a = b$	Equality of a and b
$a := b$	Definition of a as b; or assignment of a to value of b
$\forall$	For all (universal quantifier)
$f : A \to B$	Function f from A to B
$I()$	Indicator function
$\{a_1, \ldots, a_n\}$	Finite set or multi-set consisting of n elements $a_1, \ldots, a_n$
$\{a \vert \mathcal{P}(a)\}$	Set containing all elements a for which $\mathcal{P}(a)$ is true
$\emptyset$	Empty set
$\vert A \vert$	Cardinality (number of elements in set or tuple A)
$\vert a \vert$	Absolute value of numeric variable a
$A \times B$	Cartesian product of two sets A and B
A^n	$A \times A \times \cdots \times A$ (n times) iff A is a set,
	$A * A * \cdots * A$ (n times) iff A is a number
$\mathcal{R}$	The set of all real numbers
$\mathcal{N}$	The set of all natural numbers, including 0
$\mathcal{N}^+$	The set of all positive natural numbers
∞	Infinity
$\mathbf{a} = \langle a_1, \ldots, a_n \rangle$	Ordered tuple $\mathbf{a}$ consisting of n elements $a_1, \ldots, a_n$
$p(a)$	Probability that a is true iff a is a predicate
	Probability that $A = a$ iff A is a random variable
$P(A)$	Probability distribution of random variable A
$P(A, B)$	Joint probability distribution of random variables A and B
$P(A \vert B)$	Probability distribution of A conditioned B
$P(A \vert b)$	Probability distribution of A, given that $B = b$
$\{P(A)\}$	Set containing all probability distributions $P(A)$
$\{P(\mathcal{R})\}$	Set containing all probability distributions over $\mathcal{R}$
$E_{a \in A}[V_a]$	Expectation of random variable V taken over set A
$E_{P(c)}[V_c]$	Expectation of random variable V according to $P(c)$
$\mathbf{X}$	Input space
Y	Outcome space
$\widehat{Y}$	Ensemble prediction space
$\widehat{Y}_c$	Member prediction space
$\mathbf{s} = \langle \mathbf{x}, y \rangle$	A given input/outcome - pair (instance)
$\mathbf{S} = \langle \mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_m \rangle$	Multi-set (sample) of instances
$P(\mathbf{X}, Y)$	Probability distribution over instances

$p(\mathbf{x}, y)$	Probability of encountering instance $\langle \mathbf{x}, y \rangle$
C	Classifier or ensemble $C : \mathbf{X} \rightarrow \widehat{Y}$
J	Classifier inducer (learner)
l	Loss function $l : \widehat{Y} \times Y \rightarrow \mathcal{R}$
$l(\widehat{y}, y)$	Loss of prediction $\widehat{y}$ relative to outcome y
$l_2(\widehat{y}, y)$	Squared loss of prediction $\widehat{y}$ relative to outcome y
$l_{01}(\widehat{y}, y)$	0-1 loss of prediction $\widehat{y}$ relative to outcome y
$l_{\|\|}(\widehat{y}, y)$	absolute loss of prediction $\widehat{y}$ relative to outcome y
k	number of classes in discrete outcome space
m	sample size for sample of instances
n	number of classifiers in ensemble
c, C_i	Member classifiers
V	Voting function
$\mathbf{c} = \langle c_1, \ldots, c_n \rangle$	Tuple of member classifiers
$\mathbf{w} = \langle w_1, \ldots, w_n \rangle$	Tuple of voting weights
$\widehat{y}_C(\mathbf{x})$	Classifier prediction
$\widehat{y}(\mathbf{x})$	Ensemble prediction
$\widehat{\mathbf{y}}(\mathbf{x})$	Tuple of ensemble members' predictions
$\widehat{y}_c(\mathbf{x})$	Ensemble member prediction
$\widehat{P}(Y\|x)$	Belief distribution of probabilistic classifier for input x
$R(\widehat{y}\|x)$	Conditional risk of predicting $\widehat{y}$ for input x
$L(\mathbf{x}, y)$	Loss of ensemble for instance $\langle \mathbf{x}, y \rangle$
$\overline{L}(\mathbf{x}, y)$	Mean member loss of ensemble for instance $\langle \mathbf{x}, y \rangle$
$\overline{D}(\mathbf{x})$	Diversity of ensemble members for input x
L	Expected loss of ensemble for domain
$\overline{L}$	Expected mean member loss of ensemble for domain
$\overline{D}$	Expected diversity of ensemble for domain
$\mathcal{B}(s; n)$	*Bagging*$(s; n)$ (using n runs and relative sample size s)
$C(f; n)$	*Cragging*$(f; n)$ (using n runs and f folds)