Almost Bayesian conclusions without Bayesian assumptions

Volodya Vovk (joint work)

Department of Computer Science
Royal Holloway, University of London

Overconfidence Workshop
Amsterdam, 27–28 November 2015

Plan

1. The problem of calibration
2. Venn predictors
3. Venn–Abers predictors

The problem of overconfidence

In this talk I will give an example of a situation mentioned by Erik in his first email about this workshop:

> *Bayesian approaches lead to overconfidence in decision making and non-Bayesian approaches to remedying this.*

One of the non-Bayesian approaches is just to calibrate the probabilities coming from a Bayesian approach.

Overconfident predictors

- Examples of overconfident predictors:
  - people (can be trained)
  - Bayesian algorithms when their assumptions are violated
  - “almost Bayesian” algorithms (e.g., based on a narrow statistical model), again when their assumptions are violated
- It is an empirical fact that overconfidence is much more common than underconfidence.

A toy experiment: naive Bayes (1)

We consider naive Bayes with binary labels, a training set of size l, and a test set of size n, with $p = 5$ attributes.

- Generate the training and test labels $y_1, \ldots, y_{l+n}$ from the Bernoulli distribution with parameter $1/2$ (independently, here and in what follows).
- Generate the noise random variables $\eta_1, \ldots, \eta_{l+n}$ from the p-dimensional Gaussian distribution with

$$\mathbb{E}(\eta) = 0, \quad \mathrm{cov}(\eta^q, \eta^{q'}) = \begin{cases} 1 & \text{if } q = q' \\ \rho & \text{if not.} \end{cases}$$

For each $i = 1, \ldots, l + n$ set

$$x_i := \begin{cases} (1, 1) + \eta_i & \text{if } y_i = 1 \\ (-1, -1) + \eta_i & \text{if } y_i = 0. \end{cases}$$

A toy experiment: naive Bayes (2)

- For all $q = 1, \dots, p$ and for $y = 0, 1$, estimate

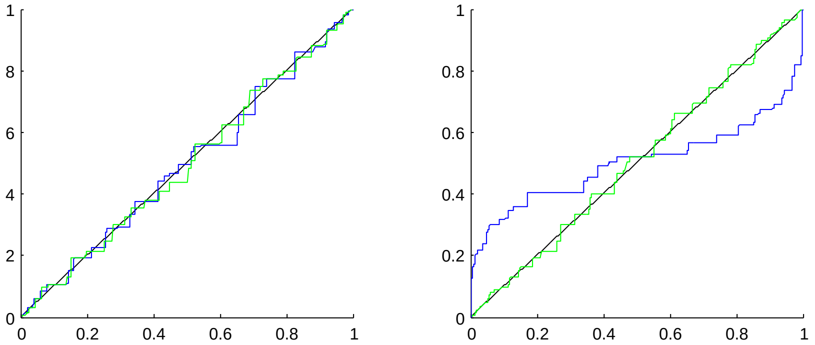
$$\hat{\theta}_y^q := \frac{\sum_{i : y_i = y} x_i^q}{|\{i \mid y_i = y\}|}.$$

- For each test object x_j, $j = l+1, \dots, l+n$, predict

$$p_j := \frac{\exp\left(-\frac{1}{2}\sum_{q=1}^{p}(x_j^q - \hat{\theta}_1^q)^2\right)}{\sum_{y=0}^{1} \exp\left(-\frac{1}{2}\sum_{q=1}^{p}(x_j^q - \hat{\theta}_y^q)^2\right)}.$$

- Run isotonic regression on (p_j, y_j), $j = l+1, \dots, l+n$. The resulting calibration curve $(p_j, \bar{y}_j)$, $j = l+1, \dots, l+n$, where $\bar{y}_{l+1} \le \dots \le \bar{y}_{l+n}$, should be close to the diagonal if $\rho = 0$.

Calibration curve (blue) for $\rho = 0$ (left) and $\rho = 0.5$ (right); $l = n = 10^6$ and $p = 5$



This talk

- For simplicity: only binary classification in this talk (two labels, 0 and 1; $P \mapsto p := P(\{1\})$).
- Most binary classification algorithms are "scoring algorithms" (output not only a prediction but also a score, the algorithm's "confidence" that the label is 1).
- We will be interested in "calibrating" the scores into probabilities.
- I will describe the traditional methods and a new method with validity guarantees.

Data

We consider observations $z = (x, y)$ consisting of two components:

- an object $x \in \mathbf{X}$
- and its label $y \in \mathbf{Y} := \{0, 1\}$ (i.e., we consider the binary case);

$\mathbf{X}$ is a measurable space, and so is $\mathbf{Z} := \mathbf{X} \times \mathbf{Y} = \mathbf{X} \times \{0, 1\}$.

Scoring classifiers

Many binary classification algorithms are in fact scoring classifiers:

- When trained on a training set of observations and fed with a test object x, they output a prediction score $s(x)$.
- We will call $s : \mathbf{X} \to \mathbb{R}$ the scoring function for that training set.
- The actual classification algorithm is obtained by fixing a threshold c and predicting the label of x to be 1 if and only if $s(x) \geq c$ (or if and only if $s(x) > c$).
- Alternatively, one could apply an increasing function g to $s(x)$ in an attempt to "calibrate" the prediction scores, so that $g(s(x))$ can be used as the predicted probability that the label of x is 1.

Platt's method (1)

- Platt's (1999) method uses sigmoids

$$g(s) := \frac{1}{1 + \exp(As + B)},$$

  where $A < 0$ and B are parameters.
- Platt discusses two approaches:
  - run the scoring algorithm and fit the parameters A and B on the full training set
  - or run the scoring algorithm on a subset (I will call it the proper training set) and fit A and B on the rest (the calibration set).
- Platt recommends the second approach (especially that he is interested in SVM, and so s tend to cluster around ± 1).

Platt's method (2)

- Platt's recommended method of fitting *A* and *B*:

$$-\sum_{i=1}^{k}\Bigl(t_i\log p_i+(1-t_i)\log(1-p_i)\Bigr)\to\min$$

where, in the simplest case, $t_i := y_i$ are the labels (minimizing the log loss on the calibration set).

- Platt recommends regularization:

$$t_i := \frac{N_1+1}{N_1+2} \text{ if } y_i = 1 \qquad t_i := \frac{1}{N_0+2} \text{ if } y_i = 0,$$

where N_1 (resp. N_0) is the number of observations labelled 1 (resp. 0).

Platt's method (3)

- Disadvantage of Platt's method: the optimal calibration function is quite often far from being a sigmoid.
- And if the training set is very big, we will suffer, since we can learn the best shape of the calibration function g.
- On the positive side, Platt's method involves regularization, and e.g., it never suffers infinite loss when using the log loss function.

Isotonic regression (1)

- The method of isotonic regression (ABERS, 1955) was applied to calibrating scoring algorithms by Zadrozny and Elkan (2002).
- Train the scoring classifier on the proper training set and compute the prediction score $s(x_i)$ for each calibration object x_i, $i = 1, \ldots, k$.
- Let g be the increasing (=non-decreasing) function on the set $\{s(x_1), \ldots, s(x_k)\}$ that maximizes the likelihood

$$\prod_{i=1}^{k} p_i, \quad \text{where } p_i := \begin{cases} g(s(x_i)) & \text{if } y_i = 1 \\ 1 - g(s(x_i)) & \text{if } y_i = 0. \end{cases}$$

Isotonic regression (2)

- Such a function g is unique and can be easily found using the "pair-adjacent violators algorithm" (PAVA).
- The function g is called the isotonic regression for $((s(x_1), y_1), \dots, (s(x_k), y_k))$.
- To predict the label of a test object x, the method of isotonic regression (IR) finds the closest $s(x_i)$ to $s(x)$ and outputs $g(s(x_i))$ as its prediction.
- Variant: we can do both training and calibration on the full training set.

Isotonic regression (3)

- The method can be implemented efficiently, in time $O(k)$.
- First construct CSD (cumulative sum diagram).
- The IR function corresponds to its GCM (greatest convex minorant).
- Apply "Graham's scan".
- It suffices to use one stack.

Disadvantages of isotonic regression

- There is no regularization (which leads, in particular, to an infinite log loss for a large test set); we have to add an *ad hoc* one (if we want to avoid this).
- The isotonic regression function is defined only on the training objects, and we have to take an *ad hoc* decision about its values on test objects.
- Isotonic regression (applied naively) does not have any validity guarantees.

Plan

1. The problem of calibration
2. Venn predictors
3. Venn–Abers predictors

Introduction

- Venn–Abers predictors belong to a class of algorithms with guaranteed “validity” (under the IID assumption).
- Another such class: “conformal predictors” (output prediction sets with a guaranteed coverage probability).
- A disadvantage of prediction sets (or p-values) is that they are not easy to combine with losses/utilities to obtain optimal decisions.
- Venn predictors, on the other hand, output well-calibrated probabilities (this property, however, is somewhat less intuitive than the validity property for conformal predictors).

Taxonomies

A Venn taxonomy A is a measurable function that assigns to each $n \in \{2, 3, \ldots\}$ and each sequence $(z_1, \ldots, z_n) \in \mathbf{Z}^n$ an equivalence relation $\sim$ on $\{1, \ldots, n\}$ which is equivariant in the sense that, for each n and each permutation π of $\{1, \ldots, n\}$,

$$(i \sim j \mid z_1, \ldots, z_n) \Longrightarrow (\pi(i) \sim \pi(j) \mid z_{\pi(1)}, \ldots, z_{\pi(n)}),$$

where the notation $(i \sim j \mid z_1, \ldots, z_n)$ means that i is equivalent to j under the relation assigned by A to $(z_1, \ldots, z_n)$. The measurability of A means that for all n, i, and j the set $\{(z_1, \ldots, z_n) : (i \sim j \mid z_1, \ldots, z_n)\}$ is measurable. Define

$$A(j \mid z_1, \ldots, z_n) := \{i \in \{1, \ldots, n\} \mid (i \sim j \mid z_1, \ldots, z_n)\}$$

to be the equivalence class of j.

Venn predictors

Let $(z_1, \ldots, z_l)$ be a training sequence of observations $z_i = (x_i, y_i)$, $i = 1, \ldots, l$, and x be a test object.

The Venn predictor associated with a given Venn taxonomy A outputs the pair (p_0, p_1) as its prediction for x's label, where

$$p_y := \frac{|A(l+1 \mid z_1, \ldots, z_l, (x, y)) \cap \{i \mid y_i = 1\}|}{|A(l+1 \mid z_1, \ldots, z_l, (x, y))|}$$

for both $y \in \{0, 1\}$.

- Intuitively: p_0 and p_1 are the predicted probabilities that the label of x is 1 (useful only when $p_0 \approx p_1$).
- The probability interval output by a Venn predictor is the convex hull $\mathrm{conv}(p_0, p_1)$ of the set $\{p_0, p_1\}$.

Unbiasedness in the small

- A random variable P taking values in $[0,1]$ is perfectly calibrated for a random variable Y taking values in $\{0,1\}$ if $\mathbb{E}(Y \mid P) = P$ a.s.
- Intuitively: P is the prediction made by a probabilistic predictor for Y; perfect calibration means that the probabilistic predictor gets the probabilities right, at least on average, for each value of the prediction.
- When we have an approximate equality: P is “valid”, “well calibrated”, or “unbiased in the small”.

Validity theorem

A selector is a random variable taking values 0 or 1.

Theorem: Let $(X_1, Y_1), \dots, (X_l, Y_l), (X, Y)$ be IID random observations. Fix a Venn predictor V and an $l \in \{1, 2, \dots\}$. Let (P_0, P_1) be the output of V given $(X_1, Y_1, \dots, X_l, Y_l)$ as the training set and X as the test object. There exists a selector S such that P_S is perfectly calibrated for Y.

- Intuitively: at least one of the two probabilities output by the Venn predictor is perfectly calibrated.
- Hopefully, they are not far apart.

Unbiasedness in the large

Corollary: For any Venn predictor V and any $l = 1, 2, \ldots$,

$$\mathbb{P}(Y = 1) \in \left[\mathbb{E}\left(\underline{V}(X; X_1, Y_1, \ldots, X_l, Y_l)\right), \right.$$
$$\left. \mathbb{E}\left(\overline{V}(X; X_1, Y_1, \ldots, X_l, Y_l)\right)\right],$$

where $(X_1, Y_1), \ldots, (X_l, Y_l), (X, Y)$ are IID observations and $[\underline{V}(\ldots), \overline{V}(\ldots)]$ is the probability interval produced by V for the test object X based on the training sequence $(X_1, Y_1, \ldots, X_l, Y_l)$.

Universality of Venn predictors (1)

- A multiprobabilistic predictor is a function that maps each training sequence $(z_1, \dots, z_l) \in \mathbf{Z}^l$ to a subset of $[0, 1]$.
- A multiprobabilistic predictor is invariant if it is independent of the ordering of the training sequence $(z_1, \dots, z_l)$.
- An invariant selector for an invariant multiprobabilistic predictor F is a measurable function $f : \mathbf{Z}^{l+1} \to [0, 1]$ such that $f(z_1, \dots, z_{l+1})$ does not change when $z_1, \dots, z_l$ are permuted and such that $f(z_1, \dots, z_{l+1}) \in F(z_1, \dots, z_l)$ for all $(z_1, \dots, z_{l+1})$.
- An invariant multiprobabilistic predictor F is invariantly perfectly calibrated if it has an invariant selector f such that

$$\mathbb{E}\bigl(Y \mid f(Z_1, \dots, Z_l, (X, Y))\bigr) = f(Z_1, \dots, Z_l, (X, Y)) \text{ a.s.}$$

whenever $Z_1, \dots, Z_l, (X, Y)$ are IID observations.

Universality of Venn predictors (2)

Theorem: If an invariant multiprobabilistic predictor F is invariantly perfectly calibrated, then it contains a Venn predictor V in the sense that both elements of $V(Z_1, \ldots, Z_l)$ belong to $F(Z_1, \ldots, Z_l)$ almost surely provided $Z_1, \ldots, Z_l$ are IID.

The invariance assumptions in this theorem are essential:

Proposition: One can construct a perfectly calibrated non-invariant multiprobabilistic predictor that does not contain a Venn predictor.

Plan

1. The problem of calibration
2. Venn predictors
3. Venn–Abers predictors

Disadvantage of the Venn method

- It is not easy to come up with a suitable Venn taxonomy (has to be hand-crafted for each problem and application area).
- Venn–Abers predictors is a class of Venn predictors that can be applied in an automatic manner on top of any scoring algorithm.
- Venn–Abers predictors are based on Isotonic Regression, and can be regarded its regularized version.

Venn–Abers predictors

Try the two different labels, 0 and 1, for the test object x. Let s_0 be the scoring function for $(z_1, \dots, z_l, (x, 0))$, s_1 be the scoring function for $(z_1, \dots, z_l, (x, 1))$, g_0 be the isotonic regression for

$$\left((s_0(x_1), y_1), \dots, (s_0(x_l), y_l), (s_0(x), 0)\right),$$

and g_1 be the isotonic regression for

$$\left((s_1(x_1), y_1), \dots, (s_1(x_l), y_l), (s_1(x), 1)\right).$$

The multiprobabilistic prediction output by the Venn–Abers predictor (VAP) is (p_0, p_1), where $p_0 := g_0(s_0(x))$ and $p_1 := g_1(s_1(x))$. (And we can expect p_0 and p_1 to be close to each other unless IR overfits grossly.)

Validity of Venn–Abers predictors

The following proposition says that Venn–Abers predictors are Venn predictors and, therefore, inherit all properties of validity of the latter.

Proposition: Venn–Abers predictors are Venn predictors.

- This is also true if scores take values in a partially ordered set (a standard setting for isotonic regression).
- The reason is that the isotonic regression is always of the following form: the given set of scores is split into disjoint blocks; the isotonic regression is constant on each block and equal to the average of its labels.

Why are partially ordered scores important?

An example (Vladimir Vapnik's idea of synergy):

- Suppose we have several (say two) good scoring algorithms (such as SVMs with different kernels).
- Define the composite score of an object as the pair $(\text{score}_1, \text{score}_2)$ (a point in the plane). The order:

$$(x_1, y_1) \preceq (x_2, y_2) \quad \text{means} \quad (x_1 \le x_2) \;\&\; (y_1 \le y_2).$$

- The corresponding isotonic regression potentially makes use of a valuable synergy between the two algorithms.

Probabilistic predictors out of Venn–Abers predictors

We can't compare Venn–Abers predictors with known probabilistic predictors using standard loss functions; we need to fit the former (somewhat artificially) to the standard framework by extracting one probability p from p_0 and p_1.

We use two loss functions: the log loss and the square loss

$$\lambda_{\log}(p,y) := \begin{cases} -\log(1-p) & \text{if } y=0 \\ -\log p & \text{if } y=1 \end{cases} \qquad \lambda_{\mathrm{sq}}(p,y) := 4(y-p)^2,$$

log being binary log. To apply them to VAPs, we replace a probability interval by an average of its end-points (to be discussed).

Loss functions

On a given test sequence of length n we calculate the mean log error and the mean square error

$$\text{MLE} := \frac{1}{n} \sum_{i=1}^{n} \lambda_{\text{log}}(p_i, y_i) \in [0, 1],$$

$$\text{MSE} := \frac{1}{n} \sum_{i=1}^{n} \lambda_{\text{sq}}(p_i, y_i) \in [0, 1],$$

where p_i is the probabilistic prediction for the label y_i of the ith observation in the test sequence.

In our first experiment we randomly permute the dataset and use the first $2/3$ observations for training and the remaining $1/3$ for testing.

Simplified Venn–Abers predictors

The simplified Venn–Abers predictor for a given scoring classifier is defined as follows. Let $(z_1, \dots, z_l)$ be a training sequence and x be a test object. Define s to be the scoring function for $(z_1, \dots, z_l)$, g_0 to be the isotonic regression for

$$\bigl((s(x_1), y_1), \dots, (s(x_l), y_l), (s(x), 0)\bigr),$$

and g_1 to be the isotonic regression for

$$\bigl((s(x_1), y_1), \dots, (s(x_l), y_l), (s(x), 1)\bigr).$$

The multiprobabilistic prediction output for the label of x by the simplified Venn–Abers predictor (SVAP) is (p_0, p_1), where $p_0 := g_0(s(x))$ and $p_1 := g_1(s(x))$.

Ranking of the best three methods (out of W/VA/SVA/IR) (1)

	log loss
Australian	W (JB), VAP (LR), SVAP (LR)
Breast	SVAP (NB), VAP (NB), W (JB)
Diabetes	VAP (LR), SVAP (SVM), W (SVM)
Echo	VAP (SVM), SVAP (NB), W (JB)
Hepatitis	VAP (SVM), SVAP (NB), W (JB)
Ionosphere	SVAP (NB), VAP (SVM), W (SVM)
Labor	SVAP (SVM), W (NN), VAP (SVM)
Liver	VAP (NN), W (JB), SVAP (LR)
Vote	SVAP (SVM), W (SVM), VAP (J)

For IR, the full training set is used both for training the scoring classifier and for calibration. W=Weka.

Ranking of the best three methods (out of W/VA/SVA/IR) (2)

	square loss
Australian	W (JB), SVAP (JB), VAP (LR)
Breast	SVAP (NB), VAP (SVM), IR (NB)
Diabetes	VAP (SVM), W (LR), SVAP (SVM)
Echo	VAP (NB), SVAP (NB), W (SVM)
Hepatitis	SVAP (NB), VAP (NB), IR (NB)
Ionosphere	SVAP (NB), IR (NB), W (JB)
Labor	W (NB), SVAP (SVM), IR (NB)
Liver	VAP (NN), SVAP (NN), W (JB)
Vote	W (SVM), SVAP (SVM), VAP (J)

IVAPs (1)

As functions, IVAPs are defined as follows:

- Divide the training set of size l into two subsets, the proper training set of size m and the calibration set of size k, $l = m + k$.
- Train a scoring algorithm on the proper training set.
- Find the scores $s_1, \ldots, s_k$ of the calibration objects $x_1, \ldots, x_k$.
- When a new object x arrives, compute its score s.

IVAPs (2)

- Fit isotonic regression to $(s_1, y_1), \ldots, (s_k, y_k), (s, 0)$ obtaining a function f_0.
- Fit isotonic regression to $(s_1, y_1), \ldots, (s_k, y_k), (s, 1)$ obtaining a function f_1.
- The multiprobabilistic prediction for the label y of x is the pair $(f_0(s), f_1(s))$.

Intuitively, the prediction is that the probability that $y = 1$ is either $f_0(s)$ or $f_1(s)$.

Validity of IVAPs

A random variable P taking values in $[0, 1]$ is perfectly calibrated (as predictor) for a random variable Y taking values in $\{0, 1\}$ if $\mathbb{E}(Y \mid P) = P$ a.s. A selector is a random variable taking values in $\{0, 1\}$.

Proposition: Let (P_0, P_1) be an IVAP's prediction for X output based on a training sequence $(X_1, Y_1), \dots, (X_l, Y_l)$. There is a selector S such that P_S is perfectly calibrated for Y provided the random observations $(X_1, Y_1), \dots, (X_l, Y_l), (X, Y)$ are IID.

Computational efficiency of IVAPs

Proposition:

- Given the scores $s_1, \dots, s_k$ of the calibration objects, the prediction rule for computing the IVAP's predictions can be computed in time $O(k \log k)$ and space $O(k)$.
- Its application to each test object takes time $O(\log k)$.
- Given the sorted scores, the prediction rule can be computed in time and space $O(k)$.

CVAPs (1)

A CVAP is just a combination of K IVAPs, where K is the parameter of the algorithm.

- Split the training multiset T randomly into K folds $T_1, \dots, T_K$.
- For $k \in \{1, \dots, K\}$,

$$(p_0^k, p_1^k) := \text{IVAP}(T \setminus T_k, T_k, x).$$

- Return $\text{GM}(p_1)/(\text{GM}(1-p_0) + \text{GM}(p_1))$ (where GM stands for geometric mean, so that $\text{GM}(p_1)$ is the geometric mean of $p_1^1, \dots, p_1^K$ and $\text{GM}(1-p_0)$ is the geometric mean of $1-p_0^1, \dots, 1-p_0^K$).

CVAPs (2)

- The folds should be of approximately equal size.
- A justification of the expression $\mathrm{GM}(p_1)/(\mathrm{GM}(1-p_0)+\mathrm{GM}(p_1))$ used for merging the IVAPs' outputs: minimax (details on the next slide).
- We have no theoretical guarantees of validity for CVAPs.

Merging multiprobabilistic predictions (1)

Suppose the predictions are $(p_0^1, p_1^1), \ldots, (p_0^K, p_1^K)$. The extra cumulative loss of $(1-p, p)$ when the true label is 1 is

$$\log \frac{p_1^1}{p} + \cdots + \log \frac{p_1^K}{p},$$

and when the true label is 0 it is

$$\log \frac{1-p_0^1}{1-p} + \cdots + \log \frac{1-p_0^K}{1-p}.$$

Equalizing the two expressions:

$$\frac{p_1^1 \cdots p_1^K}{p^K} = \frac{(1-p_0^1)\cdots(1-p_0^K)}{(1-p)^K},$$

which gives the required expression for p.

Merging multiprobabilistic predictions (2)

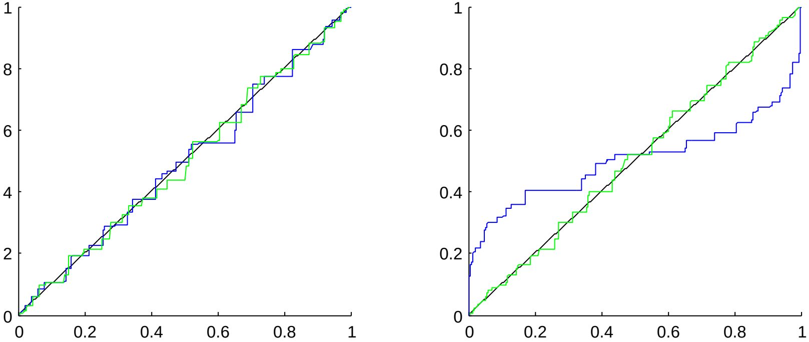
In the case of the square loss function, we solve the linear equation

$$\begin{aligned}(1-p)^2 - (1-p_1^1)^2 + \cdots + (1-p)^2 - (1-p_1^K)^2 \\ = p^2 - (p_0^1)^2 + \cdots + p^2 - (p_0^K)^2\end{aligned}$$

in p; the result is

$$p = \frac{1}{K}\sum_{k=1}^{K}\left(p_1^k + \frac{1}{2}(p_0^k)^2 - \frac{1}{2}(p_1^k)^2\right).$$

Calibration curves for NB and NB CVAP; $\rho = 0$ (left) and $\rho = 0.5$ (right); $l = n = 10^5$ and $p = 5$



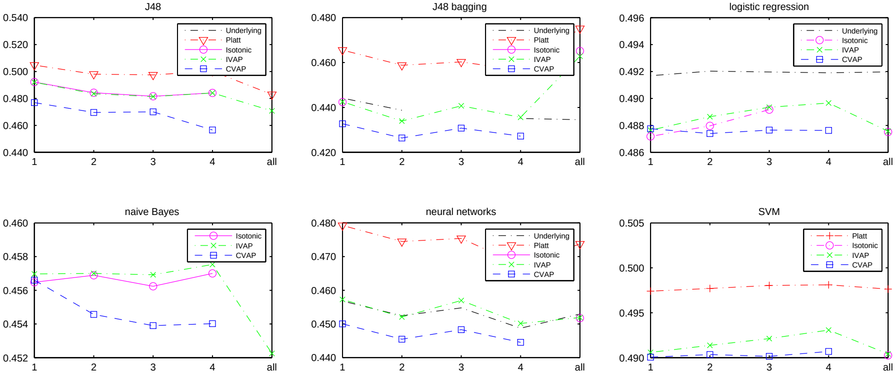
Experiment 1

- The `adult` data set (used in all papers on this topic; results are typical).
- We use the given training (32,561 observations) and test (16,281) sets; the training set is split into a proper training set and a calibration set (with no randomization, for reproducibility).
- The horizontal axis is labelled by

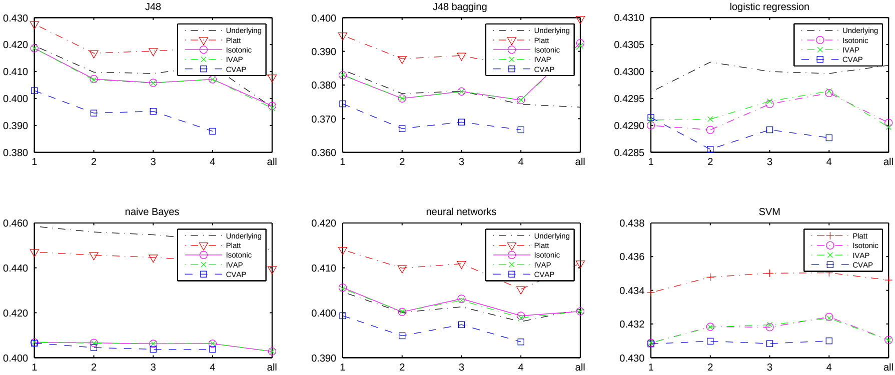
$$\frac{\text{the size of the proper training set}}{\text{the size of the calibration set}}.$$

(Except for "all", in which all training set is used for both training and calibration; unstable.)

Results for the log loss



Results for the square loss



Experiment 2

- A natural question: are CVAP results better because of the cross-over or extra regularization?
- The same data set.
- We use the first 5,000 observations as the training and the rest as the test set. IVAP and IR: the first 4,000 are used as the proper training set, and the following 1,000 as the calibration set.

Results for the log loss

algorithm	Platt	IR	IVAP	CVAP
J48	0.532	0.519	0.514	0.481
J48 bagging	0.489	∞	0.470	0.456
logistic	0.520	∞	0.504	0.497
naive Bayes	0.553	∞	0.484	0.475
neural networks	0.534	∞	0.514	0.481
SVM	0.537	∞	0.521	0.512

Results for the square loss

algorithm	Platt	IR	IVAP	CVAP
J48	0.455	0.449	0.448	0.417
J48 bagging	0.418	0.416	0.416	0.401
logistic	0.452	0.453	0.446	0.438
naive Bayes	0.467	0.433	0.431	0.423
neural networks	0.461	0.463	0.457	0.424
SVM	0.468	0.471	0.462	0.452

Experiments on tiny data sets

- The following tables report results used for our tables given earlier.
- SVAP gives a better result than IR in 52 cases out of 54 for the square loss (IR is hopeless for the log loss).
- For these tiny data sets it is difficult to improve the performance of bagging by calibration (but bagging rarely produces best results).

Results for the log loss [natural] (1)

		J48			J48 Bagging			logistic regression		
	size	W	SVAP	IR	W	SVAP	IR	W	SVAP	IR
Australian	690	∞	**0.469**	∞	**0.328**	0.344	∞	0.342	**0.340**	∞
Breast	286	∞	**0.642**	∞	**0.581**	0.636	∞	**0.584**	0.586	∞
Diabetes	768	∞	**0.635**	∞	**0.504**	0.561	∞	0.492	**0.491**	∞
Echo	132	∞	**0.670**	∞	**0.556**	0.563	∞	∞	**0.606**	∞
Hepatitis	155	∞	**0.528**	∞	**0.420**	0.434	∞	∞	**0.504**	∞
Ionosphere	351	∞	**0.410**	∞	∞	**0.251**	∞	∞	**0.524**	∞
Labor	57	∞	**0.537**	∞	0.427	**0.385**	∞	1.927	**0.297**	∞
Liver	345	∞	**0.866**	∞	**0.609**	0.707	∞	0.619	**0.611**	∞
Vote	435	∞	**0.145**	∞	0.135	**0.131**	∞	1.059	**0.148**	∞

Results for the log loss [natural] (2)

	naive Bayes			neural networks			SVM Platt		
	W	SVAP	IR	W	SVAP	IR	W	SVAP	IR
Australian	0.839	**0.367**	∞	0.557	**0.450**	∞	0.391	**0.351**	∞
Breast	0.663	**0.551**	∞	0.774	**0.738**	∞	0.583	**0.582**	∞
Diabetes	0.753	**0.508**	∞	0.536	**0.519**	∞	0.491	**0.490**	∞
Echo	0.658	**0.522**	∞	0.770	**0.605**	∞	0.558	**0.538**	∞
Hepatitis	0.936	**0.372**	∞	0.753	**0.484**	∞	0.435	**0.404**	∞
Ionosphere	0.704	**0.227**	∞	0.625	**0.379**	∞	0.359	**0.333**	∞
Labor	1.854	**0.296**	∞	0.325	**0.298**	∞	3.643	**0.287**	∞
Liver	0.727	**0.661**	∞	0.642	**0.615**	∞	0.645	**0.639**	∞
Vote	0.594	**0.211**	∞	0.235	**0.158**	∞	0.125	**0.121**	∞

Results for the square loss [RMSE] (1)

	J48			J48 Bagging			logistic regression		
	W	SVAP	IR	W	SVAP	IR	W	SVAP	IR
Australian	0.366	**0.359**	0.366	**0.313**	0.318	0.323	**0.317**	0.319	0.321
Breast	0.472	**0.463**	0.473	**0.443**	0.460	0.474	**0.442**	0.444	0.450
Diabetes	0.449	**0.443**	0.449	**0.407**	0.420	0.427	**0.399**	0.401	0.402
Echo	0.478	**0.460**	0.482	0.427	**0.423**	0.444	0.457	**0.446**	0.475
Hepatitis	0.407	**0.401**	0.419	**0.362**	0.368	0.391	0.400	**0.384**	0.411
Ionosphere	0.318	**0.312**	0.318	0.267	**0.261**	0.267	0.357	**0.349**	0.361
Labor	0.407	**0.402**	0.413	0.361	**0.339**	0.341	0.294	**0.287**	0.303
Liver	0.528	**0.518**	0.528	**0.457**	0.478	0.493	0.460	**0.458**	0.461
Vote	0.187	**0.186**	0.187	0.187	**0.186**	0.188	0.198	**0.195**	0.203

Results for the square loss [RMSE] (2)

	naive Bayes			neural networks			SVM Platt		
	W	SVAP	IR	W	SVAP	IR	W	SVAP	IR
Australian	0.392	**0.333**	0.335	**0.360**	0.361	0.371	0.343	**0.325**	0.327
Breast	0.449	**0.427**	0.433	**0.485**	0.491	0.508	0.443	**0.442**	0.447
Diabetes	0.420	**0.410**	0.413	0.413	**0.413**	0.417	**0.399**	0.400	0.402
Echo	0.428	**0.412**	0.426	0.457	**0.443**	0.468	**0.416**	0.418	0.431
Hepatitis	0.357	**0.335**	0.342	0.396	**0.379**	0.427	**0.350**	0.353	0.364
Ionosphere	0.281	**0.250**	0.251	0.321	**0.316**	0.333	0.312	**0.312**	0.316
Labor	**0.256**	0.284	0.281	**0.279**	0.293	0.307	**0.274**	0.280	0.283
Liver	0.480	**0.478**	0.487	0.459	**0.456**	0.463	0.473	**0.472**	0.477
Vote	0.292	0.251	**0.250**	0.216	**0.206**	0.227	**0.183**	0.185	0.188

A role for imprecise probabilities?

- Venn–Abers predictors output imprecise probabilities (multiprobabilities), which we then merge into precise probabilities in order to evaluate their quality.
- Perhaps imprecise probabilities are useless for automatic (*en masse*) decision making.
- But they can be quite informative in less formal and more individual cases (such as quoting probabilities for the presence of an illness in medicine).

Conclusion (1)

Open problem:

- Suppose we know the data-generating mechanism but still apply IVAP to the optimal probabilistic predictions. How much do we lose? (For example, how far will p_0 and p_1 be from the optimal p asymptotically?)

A similar programme in the case of conformal predictors:

Evgeny Burnaev and Vladimir Vovk. Efficiency of conformalized ridge regression. Proceedings of COLT 2014.

And if we lose a lot: doesn't it mean that the Bayesian algorithm is extremely fragile? (Larry Wasserman, in the context of conformal prediction)

Conclusion (2)

Further details:

- Vladimir Vovk, Alex Gammerman, and Glenn Shafer. Algorithmic Learning in a Random World. New York: Springer, 2005.
- Vladimir Vovk and Ivan Petej. Venn–Abers predictors. Proceedings of UAI 2014. Available on arXiv.
- Vladimir Vovk, Ivan Petej, and Valentina Fedorova. Large-scale probabilistic predictors with and without guarantees of validity. Proceedings of NIPS 2015 (to appear). Available on arXiv.

Thank you for your attention!