An introduction to Venn-ABERS Predictors

Paolo Toccaceli

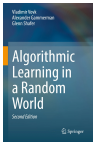
Centre for Reliable Machine Learning
Royal Holloway, University of London
currently at Graphcore

DSSV-ECDA 2023, Antwerp, July 6, 2023

Outline

- Informal presentation, with focus on concepts rather than on formalism.
  - Probabilistic prediction, calibration
  - Venn Predictors
  - Venn-ABERS Predictors
  - Example of application

- Venn-ABERS Predictors are one of several probabilistic "distribution-free" ML techniques developed at the Centre for Reliable Machine Learning, Royal Holloway, Univ. of London, by Prof. Gammerman and Prof. Vovk
  - Reference textbook:
    Vovk, Gammerman, Shafer,
    Algorithmic Learning in a Random World,
    Springer (2022, 2nd ed.)



  - Yearly symposium on the topic: COPA (2022, 11th edition)
  - Special issues on the topic in journals
- The techniques have theoretically proven guarantees, under minimal assumptions
- Not just of theoretical interest:
  - CP and VP are currently used in major companies in the pharmaceutical and agrochemical sectors

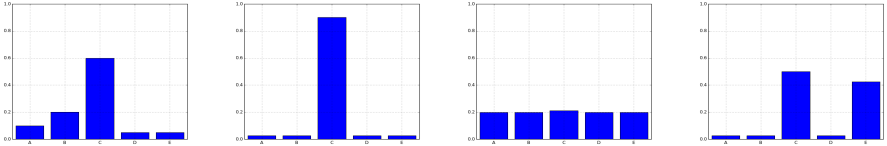
Calibrated Probabilistic Predictions: Venn Predictors

Why Probabilistic Prediction: a simple example

Suppose that, out of five possible labels A, B, C, D, and E, a classifier outputs C as prediction for an object x

Generally, this means that C is the most probable label for x, given the training data.

But this in itself does not tell us much!



In the 4 distributions above, C is the most probable label.

But the way we would act if we knew the distributions might be very different!

Probabilistic Prediction

- We want to predict the posterior conditional probability (density) of the label, given the training set and the test object.

$$\mathbb{P}[Y = y | X = \mathbf{x}]$$

- There are several different approaches:
  - Bayesian Machine Learning
    - Requires a “prior”
  - Kernel-based Density Estimation (Parzen-Rosenblatt)
    - Limitation: curse of dimensionality
  - “Classical Statistics” methods
    - e.g. From ECDF, as regularized solution of Fredholm integral equations (Vapnik)
  - ...the approach of this tutorial

Our setting and goal

- Let's restrict our attention to a classification setting (i.e. a discrete set of label values)
- Learning under unconstrained randomness:
  - we know the space of examples $(\mathbf{x}, y) \in \mathbf{X} \times \mathbf{Y}$
  - we know the examples are drawn independently from the same distribution Q (i.i.d. assumption)
  - at the outset we know nothing about the distribution Q.
- We want the predictor to be **valid**
  - Validity is the property of a predictor that outputs probability distributions that perform well against statistical tests based on subsequent observation of the labels.
  - In particular, we are interested in **calibration**:

$$\mathbb{P}[Y = y \mid P_y = p] = p$$

i.e. the relative frequency of the label y among the objects for which p is the predicted probability of $Y = y$ is indeed p

Venn Predictors

- Venn Predictors are a form of multi-probabilistic predictors for which we can prove **validity** properties
- But... it can be proved that validity cannot be achieved for probabilistic prediction in a general sense!
- This limitation is circumvented with the following two provisions:
  - For a given test object, we output multiple probabilities and we can guarantee that one of them is the valid one.
  - We restrict the statistical test for validity to calibration, i.e. the property that probabilities are matched by observed frequencies
    (for example, a particular label should occur in about 25% of the instances in which we give it a probability of 0.25)

Venn Predictors

- Let's start with an elementary approach
  - divide the training set objects into categories.
  - use some method to classify the test object into one of the categories.
  - use the frequencies of labels in the category of the test object as predicted probabilities for the object's label.
- We introduce some key differences
  - We divide *examples* rather than just objects into categories.
  - We create a test example from the test object by assigning a hypothetical label.
  - When we compute the frequencies of labels in the category containing the test example, we *include* the test example itself.
  - We repeat the category assignment and label frequency calculation, for each possible label value.

An aside: Why Venn?

- John Venn [1834-1923] was a logician and philosopher.
- In 1866 he published *The Logic of Chance*, in which he laid the foundations for the frequentist notion of probability.
- He's credited to have been the first to formulate explicitly and study the **reference class problem**.

  "It is obvious that every individual thing or event have an indefinite number of properties or attributes observable in it, and might therefore be considered as belonging to an indefinite number of different classes of things"
- Which class do we take when calculating relative frequencies?
- Venn also thought that "the more special the statistics, the better". But this leads to a dilemma: the more specific the class is, the fewer the element in the class.
- In Venn Predictors, we use a ML method to create classes.

Venn Predictors in more formal terms

- Usual setting:
  - training examples $z_i = (x_i, y_i)$ forming a multiset $\langle z_1, \ldots, z_\ell \rangle$

    multiset: a collection that allows repeated elements (unlike a set)
  - test object $x_{\ell+1}$
  - We are seeking the probability of Y=y, given that X=$x_{\ell+1}$
- VPs do not give you exactly that: rather than one value, VPs output a **(multi)set** of probabilities.
- Venn Predictor output:
  - Given a test object $x_{\ell+1}$
    - it outputs $|\mathbf{Y}|$ probability distributions on $\mathbf{Y}$
- NOTE: Venn Predictors are multi-probabilistic predictors. They output several probabilities for each value that the label can take. One of these probabilities is the calibrated one, but one does not know which one.

An example of VP multi-probabilistic output

	0	1	2	3	4	5	6	7	8
0	0.6470	0.2941	0.0000	0.0588	0.0000	0.0000	0.0000	0.0000	0.0000
1	0.5625	0.3750	0.0000	0.0625	0.0000	0.0000	0.0000	0.0000	0.0000
2	0.5625	0.3125	0.0625	0.0625	0.0000	0.0000	0.0000	0.0000	0.0000
3	0.5625	0.3125	0.0000	0.1250	0.0000	0.0000	0.0000	0.0000	0.0000
4	0.5625	0.3125	0.0000	0.0625	0.0625	0.0000	0.0000	0.0000	0.0000
5	0.5625	0.3125	0.0000	0.0625	0.0000	0.0625	0.0000	0.0000	0.0000
6	0.5625	0.3125	0.0000	0.0625	0.0000	0.0000	0.0625	0.0000	0.0000
7	0.5625	0.3125	0.0000	0.0625	0.0000	0.0000	0.0000	0.0625	0.0000
8	0.5625	0.3125	0.0000	0.0625	0.0000	0.0000	0.0000	0.0000	0.0625

- Rows contain distributions (check that they sum to 1)
- Each row corresponds to a hypothetical assignment of a label to the test object. This will become clearer in the next slide.

- Given a test object x_i
  - For each possible label value y
    - we create an example (x_i, y)
    - Identify the category T to which the hypothetical example (x_i, y) belongs. We can use a ML method to do this.

      For instance, a nearest-neighbour taxonomy can be defined with this rule: "two examples are assigned to the same category if their nearest neighbours have the same label"; the category T would contains all the examples that have as label the label y_{NB} of the nearest neighbour of (x_i, y)
    - We compute the empirical probability distribution p_y of the labels over the examples in category T

      i.e. find all examples that have nearest neighbour with label y_{NB}; then for each possible label, count how many of those examples have that label; then normalize the counts to obtain relative frequencies.

One comment on calibration

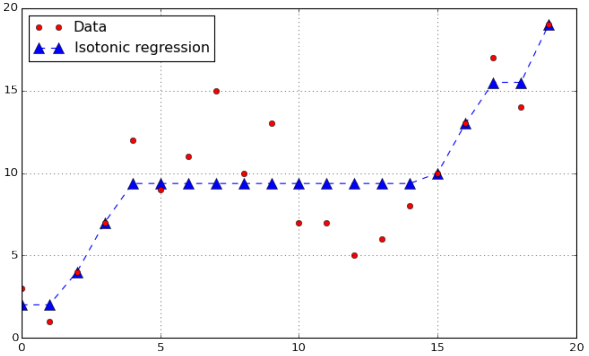
- Calibration is desirable, but it is not the only property we seek in predictions
- Example: weather prediction
  - If we were asked to predict with what probability it will rain tomorrow, we can simply always respond with the long-term average probability of rain.
  - It would be a calibrated prediction, but it is hardly useful.
  - What we want to have a more **specific** prediction.
- Venn Predictors offer a theoretically-backed framework in which we no longer have to worry about calibration; we can focus only on making predictions more specific.

Venn-ABERS predictors

Calibration of scores into probabilities

- Let's restrict our attention to binary classification
- Many machine learning algorithms for classification are in fact *scoring classifiers*: they output a prediction score $s(x)$ and the prediction is obtained by comparing the score to a threshold.
- One could apply a function g to $s(x)$ to calibrate the scores so that $g(s(x))$ can be used as predicted probability.
  - Isotonic Regression: let's assume that $g()$ be an non-decreasing function.
  - Platt's scaling: let's fit a sigmoid

Isotonic Regression[1] example



Non-decreasing function that minimises sum of square residues

[1] Monotonic: "one ordering", either Isotonic ("order-preserving") or Antitonic ("against the order")

Isotonic calibrator

- The *isotonic calibrator* g for $((s(x_1), y_1), (s(x_2), y_2), \ldots, (s(x_\ell), y_\ell))$ is the non-decreasing function on $s(x_1), s(x_2), \ldots, s(x_\ell)$ that maximises the likelihood

$$\prod_{i=1,2,\ldots,\ell} p_i$$

where:

$$p_i = \begin{cases} g(s(x_i)) & \text{if } y_i = 1 \\ 1 - g(s(x_i)) & \text{if } y_i = 0 \end{cases}$$

- The isotonic calibrator can be found as isotonic regression on $(s(x_1), y_1), (s(x_2), y_2), \ldots, (s(x_\ell), y_\ell))$.

Isotonic regression and Venn prediction

- Isotonic Regression is piecewise-constant.
- In each interval of s in which $g(s)$ is constant, the IR takes the average of the values of the training points in that interval[2]
- So, when the labels are encoded as 0 and 1, the value of the IR is the relative frequency of label 1 in the interval.
- It's what we need to use it as Venn predictor!
  - The categories of the Venn taxonomy are the intervals over which the IR is constant.

[2] theorem by Ayer, Brunk, Ewing, Reid, Silverman (1954)

Venn-ABERS predictors

- Let $s_0(x)$ be the scoring function for $(z_1, z_2, \ldots, z_\ell, (x, 0))$,
  $s_1(x)$ be the scoring function for $(z_1, z_2, \ldots, z_\ell, (x, 1))$,
  $g_0(x)$ be the isotonic calibrator for

$$((s_0(x_1), y_1), (s_0(x_2), y_2), \ldots, (s_0(x_\ell), y_\ell), (s_0(x), 0))$$

  and $g_1(x)$ be the isotonic calibrator for

$$((s_1(x_1), y_1), (s_1(x_2), y_2), \ldots, (s_1(x_\ell), y_\ell), (s_1(x), 1))$$

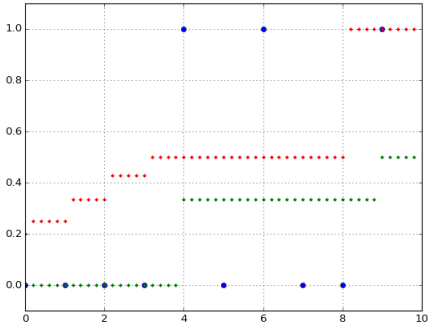
- The multiprobabilistic prediction output by the Venn-ABERS predictor is:
  (p_0, p_1), where $p_0 := g_0(s_0(x))$ and $p_1 := g_1(s_1(x))$

- The calibration guarantee is:

$$\mathbb{E}(Y|P) = P \qquad a.s.$$

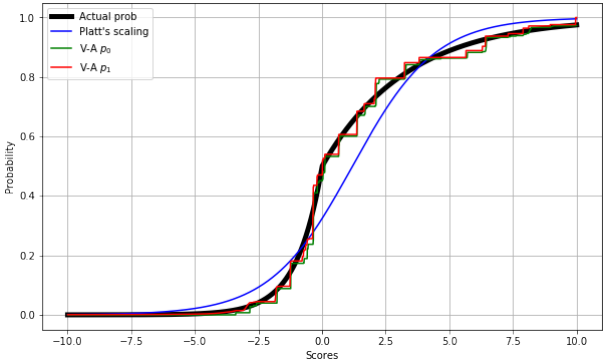
  for an appropriate choice of P out of p_0 and p_1

Toy example



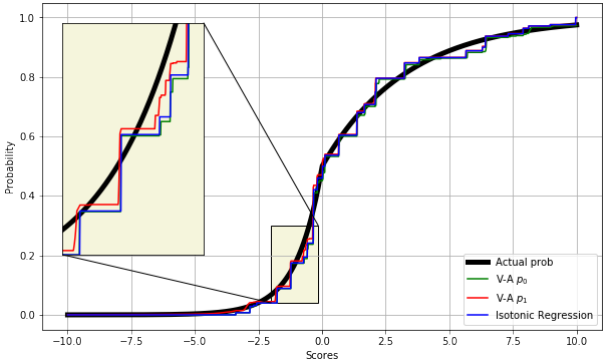
- Example of the two Isotonic Regressions in Venn Prediction
  - Blue dots: data (+1 or 0)
  - green dots: $g_0(s)$, red dots: $g_1(s)$

Synthetic data: V-A vs. Platt's



- Data set with deliberate departure from sigmoid
  - Venn-ABERS calibrator manages to recover actual score-probability relationship
  - Platt's scaling is not as accurate

Synthetic data: V-A vs. Isotonic Regression



- IR too recovers the score-probability relationship
  - However, the probability estimates are not as fine-grained
  - IR: 31 different probability levels
  - Venn-ABERS: 211 for p_0, 1823 for p_1

Further observations

- Compared with bare Isotonic Regression, the multi-probabilistic output also provides an indication of the reliability of the probability estimates.
  - If the probabilities differ, this can be taken as an indication of the uncertainty on the probability estimate itself.
- Compared with Platt's Scaling (fitting a sigmoid as calibrator), Venn-ABERS predictors do not make any assumption on the shape (functional form) of the calibrator.

Single value prediction

- Can we deal with a single-valued probability instead?
- One approach is the following:
  - Assume a loss function $L(y, p)$ and minimize the expected loss with respect to it
  - E.g. log loss:

$$\begin{cases} -\log p & \text{if } y = 1 \\ -\log(1-p) & \text{if } y = 0 \end{cases}$$

  - The optimal p for log loss is

$$p = \frac{p_1}{1 - p_0 + p_1}$$

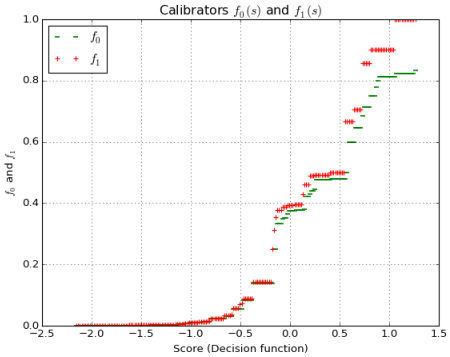
  - There is also an optimal p for Brier score (aka RMSE).

Synthetic data example: Loss

- Losses on test data (average over 20 sets of 5000 test examples)

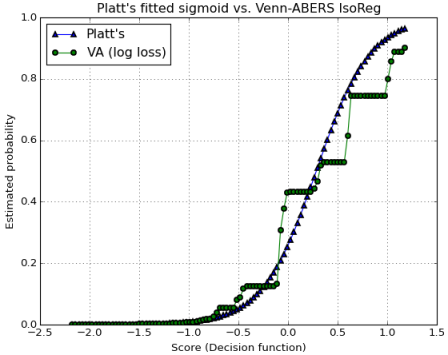
Probability estimate	Log loss	Brier
Isotonic Regression	0.380691	0.123435
Platt scaling	0.400399	0.131967
Venn-ABERS	0.368595	0.123379
VA p_0	0.375235	0.123389
VA p_1	0.375019	0.123461

Real-life example



- Venn-ABERS Calibrators for Compound Activity Prediction
  - Applied to SVM decision function
  - green dots: $g_0(s)$, red dots: $g_1(s)$
- Imbalanced data set (class 1 was $\approx 1\%$)

Platt scaling vs. Venn-ABERS



- Platt scaling vs. (log-loss) Venn-ABERS
  - Platt's scaling is possibly less accurate for high probs

Large-scale Venn-ABERS predictors

- Venn-ABERS appear much more computationally demanding than Isotonic Regression or Platt's Scaling.
  - For every evaluation, we have to retrain the underlying machine learning algorithm and recompute Isotonic Regressions
  - This would not scale to large data sets.
- Inductive Venn-ABERS Predictors
  - The training set is split into a proper training set and a calibration set.
  - The proper training set is used to train the underlying ML algorithm once.
  - The Isotonic Regression is calculated only on the calibration set.
  - This method retains the theoretical validity guarantee.
- In actual fact, it can be computed very efficiently
  - It is possible to exploit the fact that only one data point is added to an otherwise fixed calibration set
  - Most computation occurs once for $g_0()$ and once for $g_1()$.
  - The evaluation requires only a binary search in a pre-computed data structure

Conclusions

- Complementing a prediction with its probability can enable better decision making
- Venn Predictors are (multi)probabilistic predictors with validity guarantee
- Venn-ABERS Predictors are Venn Predictors that can be applied on top of a Scoring Classifier
- VAP do not assume a functional form (e.g. sigmoid) for the relationship between score and probability

Acknowledgements

- Many thanks to Prof. Alexander Gammerman, Prof. Vladimir Vovk, Prof. Zhiyuan Luo, and Dr. Ilia Nouretdinov for useful advice and insightful discussions.
- This work was made possible by the AstraZeneca grant "Machine Learning for Chemical Synthesis" (R10911) and by the ExCAPE H2020 project.
- I also acknowledge the generosity of Graphcore for supporting my attendance at this conference.

References

- Vladimir Vovk, Alex Gammerman, and Glenn Shafer.
  Algorithmic Learning in a Random World.
  Springer Cham, 2022.
  https://doi.org/10.1007/978-3-031-06649-8

- Vladimir Vovk, Ivan Petej.
  Venn-Abers Predictors
  In *Proceedings of 30th Conference on Uncertainty in Artificial Intelligence*, 2014
  http://alrw.net/articles/07.pdf

- Vladimir Vovk, Ivan Petej, and Valentina Fedorova.
  Large-scale probabilistic predictors with and without guarantees of validity.
  Technical Report arXiv:1511.00213 [cs.LG], arXiv.org e-Print archive, November 2015.

- Python implementation at: https://github.com/ptocca/VennABERS