Research presentation

Paolo Toccaceli

Centre for Reliable Machine Learning
Royal Holloway, University of London

https://cml.rhul.ac.uk/people/ptocca/HomePage/

March 2021

Area of research

- Non-parametric probabilistic methods
  - The only assumption is: training and test data are i.i.d.
  - Unconstrained randomness: distribution fixed but unknown.
- Conformal Predictors, Venn Predictors, Conformal Predictive Distributions
- My research
  - Applications: drug discovery and development
  - Methods: Combination of CP

Conformal Prediction

- Framework for predictions with **guaranteed error rate** under i.i.d. assumption.
- Uses a different approach for expressing uncertainty: **multi-valued** predictions.
  - it hedges predictions so that they do not exceed a chosen error rate.
- Conformal Prediction is a framework: any scoring ML methods can be used.

- It is based on the notion of Non Conformity Measure.
  - A function of a “bag” of observations and of a test observation that expresses how dissimilar the test observation is w.r.t. the bag of observations.
  - The NCM can be computed with the help of a ML method. e.g. as $\mid y_i - \hat{y}_i \mid$

- Given training set $Z = \Lbag z_1, \ldots, z_\ell \Rbag$, where $z_i = (x_i, y_i)$ and a test object $x_{\ell+1}$, for each possible label value $\bar{y}$
  - create hypothetical observation $z_{\ell+1} = (x_{\ell+1}, \bar{y})$
  - compute $\ell + 1$ NCMs: $\alpha_i := \mathcal{A}(Z \cup \{z_{\ell+1}\} \backslash \{z_i\}, z_i) \qquad i = 1, \ldots, \ell + 1$
  - compute a p-value: $p_Y := \frac{\left|\{i=1,\ldots,\ell+1 : \alpha_i \geq \alpha_{\ell+1}\}\right|}{\ell+1}$,
  - prediction for significance level $\epsilon \in [0, 1]$:

$$\Gamma^\epsilon \left(x_1, y_1, \ldots, x_\ell, y_\ell, x_{\ell+1}\right) := \{y \in Y : p_y > \epsilon\}$$

- The prediction is considered correct if Γ^ϵ contains the actual label, otherwise it is an error.

- NOTE: Several technicalities were omitted.

- **Validity** property: Errors occur with frequency $< \epsilon$, barring statistical fluctuation.
- CP predictions are sets. They can also be empty (all labels are rejected at the significance level ϵ)
- Validity can be banally achieved by always predicting the entire set of labels.
- We seek prediction sets that are as small as possible (**efficiency**).
- The more accurate the NCM, the more efficient the CP is.
  Validity is guaranteed regardless of the accuracy of the NCM.
- In its most general formulation, the method is computationally heavy.
  A simpler form exists (inductive or 'split' CP) with the same guarantees.
- The validity guarantee can be made **label-conditional**.
  This is important in the case of imbalanced data sets.

Venn Predictors

- Many classification methods claim to output ‘probabilities’, but do they?
- It seems reasonable to require **calibration**:

  $$\mathbb{P}\left[Y = y \,|\, P_y = p\right] = p$$

  i.e. observed relative frequencies correspond to predicted probabilities
- When trying to predict a probability, we are faced with the **problem of the ‘reference class’**, i.e. how to define the equivalence class grouping the examples that we consider sufficiently similar for the purpose of estimating a probability.
- Venn Predictors rely on an underlying ML method to determine the ‘reference class’ of an example.

Venn Predictors

- Venn Predictors provide a calibration guarantee, but their predictions are hedged.
  If the possible label values are k, VPs output k probability distributions (each specifying the probability for each of the k possible values).
  - Given the test object $x_{\ell+1}$
    For every possible value y of the label:
    - We form the bag $\wr z_1, \dots, z_{\ell+1} \int$, with the hypothetical example $z_{\ell+1} = (x_{\ell+1}, y)$
    - (Using an underlying ML) Identify the category T to which the example $(x_{\ell+1}, y)$ belongs.
    - The empirical probability distribution p_y of the labels in category T is obtained as:
$$p_y(y') := \frac{|\{(x^*, y^*) \in T : y^* = y'\}|}{|T|}$$
    - In words: for every possible value y' of the label, we calculate the fraction of examples in category T that have label y'
- The calibration guarantee applies to one of the k probability distributions; which one varies from test object to test object.
- The discrepancies across probability distributions can be taken as an indication of the sensitivity of the probability estimate.

Venn-ABERS predictors

- For binary classification, there is a particular form of VP called Venn-ABERS predictor.

- It calibrates a score into a pair of probabilities.

- 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: assume that $g()$ be an non-decreasing function.
  - Platt's scaling: fit a sigmoid

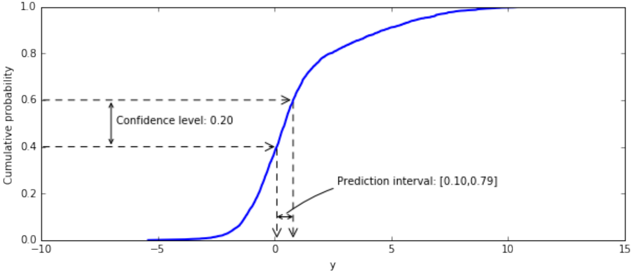
- Intuitively, Venn-ABERS gives the "Venn Predictor treatment" to Isotonic Regression.

Conformal Predictive Distributions

- Regression setting.
- Predictive Distribution: given a test object, the prediction is a probability distribution over the continuous label.
- Generally, PDs are the preserve of Bayesian Methods.
- Conformal Predictive Distributions offer a non-parametric method for estimating PDs.
  - No prior required
  - Not constrained to a distribution family
- CPDs are expressed in the form of cumulative distribution functions, rather than probability densities.

Conformal Predictive Distributions

- Guaranteed coverage is the key property of CPD
  - We can choose a confidence level α and we can read, off the predictive distribution, intervals of y.
  - The coverage property guarantees the actual value is in the chosen intervals with relative frequency α (barring statistical fluctuation) over the test examples.



Conformal Predictive Distributions

- CPD framework is formulated following an approach analogous to that of Conformal Predictors (i.e. using a (Non-)Conformity Measure), but with added complications (not covered here).
- One instance of CPDs uses Kernel Ridge Regression (KRRPM) to compute the CM.
- It is possible to derive an explicit form that can be implemented in an efficient way in terms of linear algebra operations.

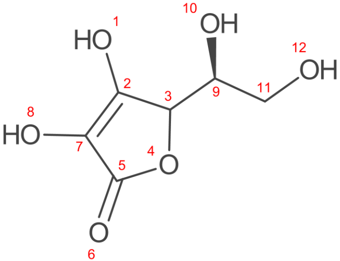
Application to Drug Discovery and Development

- The reliable prediction of the biological properties of an arbitrary compound can reduce the costs and the duration of drug discovery and development.
- ExCAPE: Exascale Compound Activity Prediction Engines, EU Horizon 2020 project
  - Design, develop, and implement CAP methods that fully exploit Exascale HPC platforms
  - Partners from academia, pharmaceutical industry, government research outfits, IT company, consultancies
  - Data set: $\approx 800k$ compounds, ≈ 900 targets
- AstraZeneca: PK and PhysChem property prediction
  - Collaboration with Quantitative Biology group

QSAR as a Machine Learning problem

- Quantitative Structure-Activity Relationship (QSAR)
  Let's assume that the specific biological property of a molecule is determined by the presence of particular chemical groups in certain spatial arrangements
- **Training Example**: (Object,Label)
  Label: biological activity, $y \in \{\text{Active}, \text{Inactive}\}$
  Object: (sparse) vector, $x \in \mathbb{N}^{|K|}$, where K is the set of "molecular descriptors"
- **Test Example**: Object
  molecular descriptors of a compound of which we want to predict the activity

An example: Signature descriptors



Ascorbic acid

Counts	Signature
6	[C]
6	[O]
4	[O]([C])
2	[C]([C]=[C][O])
2	[C]([C][C][O])
2	[O]([C]([C]=[C]))
1	[C](=[C]([C](=[O][O,0])[O])[C]([C]([C][O])[O,0])[O])
1	[C](=[C]([C][O])[C]([O]=[O])[O])
1	[C]([C](=[C]([C,0][O])[O])=[O][O]([C,0]([C])))
1	[C]([C](=[C][O])=[O][O]([C]))
1	[C]([C](=[O][O]([C,0]))=[C]([C,0]([C])[O])[O])
1	[C]([C]([C]([C][O])[O])[O])
1	[C]([C]([C]([O])[O])[C](=[C]([C,0][O])[O])[O]([C,0](=[O])))
1	[C]([C]([C][O])=[C]([C][O])[O])
1	[C]([C]([C][O])[C](=[C][O])[O]([C]))
1	[C]([C]([C][O])[O])
1	[C]([C]([O])[C]([C](=[C][O])[O]([C]))[O])
1	[C]([C]([O])[C]([C][O])[O])
1	[C]([C][O])
1	[C]([C][O]=[O])
1	[O](=[C]([C](=[C][O])[O]([C])))
1	[O](=[C]([C][O]))
1	[O](=[C])
1	[O]([C](=[C]([C][O])[C]([O]=[O])))
1	[O]([C]([C](=[C,0][O])=[O])[C]([C]([C][O])[C,0]([O])))
1	[O]([C]([C]([C][O])))
1	[O]([C]([C]([C][O])=[C]([C][O])))
1	[O]([C]([C]([O])[C]([C][O])))
1	[O]([C]([C]))
1	[O]([C]([C][C]))
1	[O]([C]([C][C])[C]([C]=[O]))
1	[O]([C][C])

A signature[1] is a sub-graph of the labelled molecular graph.

[1] J-L. Faulon, et al. The signature molecular descriptor. *Journal of Chemical Information and Computer Sciences*, 2003.

ExCAPE Challenges and solutions

- **Data Volume:** one node not sufficient
  - Distributed iterative approach: a variant of CascadeSVM
  - Inductive Conformal Prediction
- **Imbalance:** the Active class is often $\approx$ 1% of the total
  - Weighting of minority class
  - Mondrian Conformal Prediction
- **High dimensionality:** the number of features is in the order of 10^5
  - Specialized ML methods (e.g. Kernel methods)

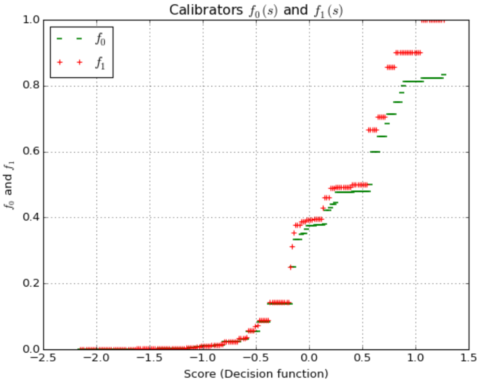
  Sparseness: non-zero feature values are $\approx$ 0.03%
  - Specialized data structures and kernels (e.g. Sparse Tanimoto)
- Open source Venn-ABERS implementation on `https://github.com/ptocca/VennABERS`

Application of CP

- Data set: AID827 from public-domain repository PubChem
- Binary classification problem: Active / Inactive
- High imbalance: only 1.2% Active
- Underlying ML method: SVM with Tanimoto+RBF kernel
- Test set with 10,000 compounds
- Prediction: Uncertain, Active, Inactive, Empty.
- In the table below, each line corresponds to a confusion matrix for the given error rate

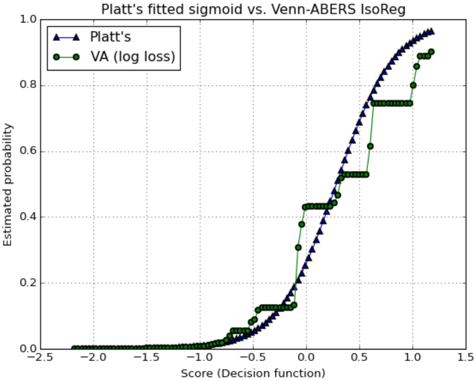
Target error rate	Active pred Active	Inactive pred Active	Inactive pred Inactive	Active pred Inactive	Empty pred	Uncertain	Active Error Rate	Inactive Error Rate
1%	47.65	94.10	1044.90	0.95	0.0	8812.40	0.82%	0.95%
5%	67.20	490.40	3091.75	5.20	0.0	6345.45	4.52%	4.96%
10%	76.15	999.25	4703.75	10.60	0.0	4210.25	9.22%	10.11%
15%	82.10	1484.85	6021.80	17.30	0.0	2393.95	15.04%	15.02%
20%	86.55	1982.25	6928.95	22.80	0.0	979.45	19.83%	20.05%

Application of Venn-ABERS



- 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

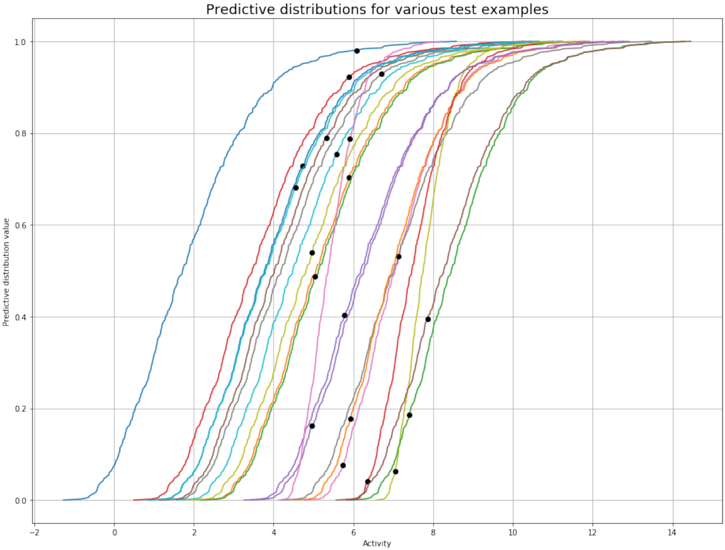


- Platt scaling vs. (log-loss) Venn-ABERS
    - Platt's scaling is possibly less accurate for high probs

AstraZeneca collaboration

- We applied KRRPM to prediction of PharmacoKinetic and PhysChem properties using AstraZeneca internal assay data
  - 4 biological endpoints: HLM, hERG, LogD, hPPB
  - dataset sizes: from $\approx 40k$ to $\approx 180k$
  - number of features: from $\approx 100k$ to $\approx 200k$
  - Linear and RBF Kernel
- Computations run on AZ Scientific Computing Platform, a HPC platform
  - 132GB of RAM, 32-core CPUs, 1000+ nodes
  - Parallelization over cores and over nodes
- Implemented in Python, with Cython for performance-critical parts
- Scaled KRRPM up to training set size of 80k by using directly BLAS matrix library and optimizing use of temporaries
- Contributed code to `scikit-learn` v0.22
  ENH Faster manhattan_distances() for sparse matrices (PR#15049)

Conformal Predictive Distributions



Combination of CP

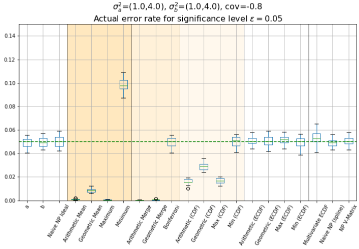
- Ensembling is a well-established strategy for improving predictive performance.
- In Statistical Hypothesis Testing the combination of p-values has been received a lot of attention.
- Can we combine CP p-values so that:
  - validity is preserved
  - efficiency is improved
- Rationale: by operating at the p-value level, we do away with the problem of incommensurate scores

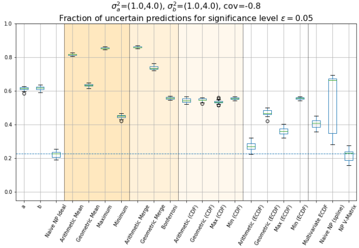
Original work on p-value combination

- Conventional combination methods do not preserve validity or require independence. Also, they are not adaptive.

- I considered the case of binary classification and proposed:
  - ECDF Calibration: a simple technique to recover validity (sacrificing part of the training set)
  - Learning to Combine: an adaptive combination scheme based on multinomial LR
  - Efficient combination using the Neyman-Pearson Lemma

A comparison of combination methods


σ_a^2=(1.0,4.0), σ_b^2=(1.0,4.0), cov=-0.8
Actual error rate for significance level $\varepsilon = 0.05$


σ_a^2=(1.0,4.0), σ_b^2=(1.0,4.0), cov=-0.8
Fraction of uncertain predictions for significance level $\varepsilon = 0.05$

- Top diagram: error rates. Ideally the error rate should be 0.05.
- Bottom diagram: fraction of predictions with both labels (the smaller, the better).