Beyond temperature scaling: Obtaining well-calibrated multiclass probabilities with Dirichlet calibration

Meelis Kull, **Miquel Perello Nieto**, Markus Kängsepp,
Telmo Silva Filho, Hao Song, Peter Flach

21st November 2019

Paper accepted at NeurIPS 2019

Current presentation for PyData Bristol

UNIVERSITY OF TARTU

University of BRISTOL

Universidade Federal da Paraíba ESTATÍSTICA

sphere

The Alan Turing Institute

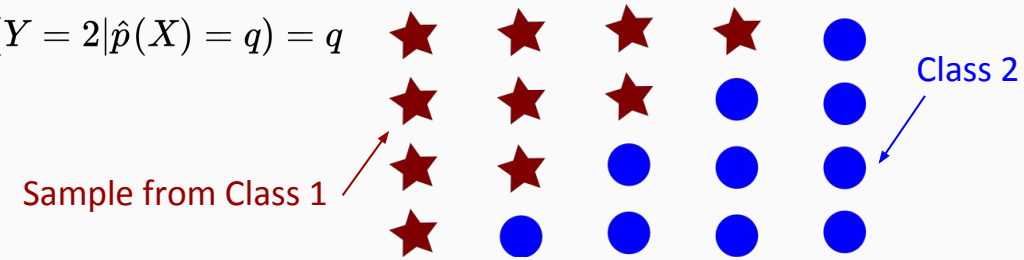
PyData Bristol

What is Classification Calibration?

A probabilistic classifier $\hat{\mathbf{p}}$ is:

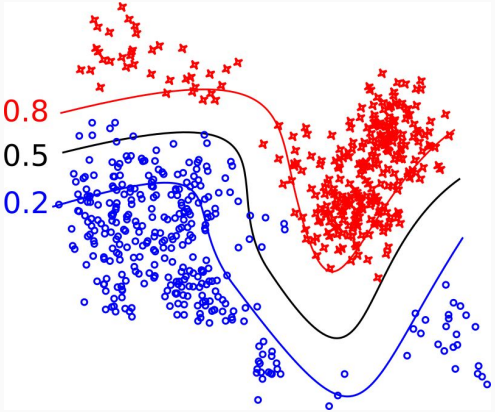
- Binary-calibrated if for any prediction q the proportion of positives among all instances x getting the same prediction $\hat{p}(x) = q$ are:

$$P(Y = 2|\hat{p}(X) = q) = q$$

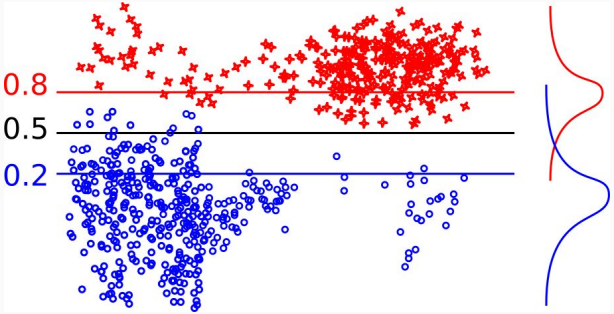


Empirical proportion Class 2:	0%	25%	50%	75%	100%
Calibrated classifier:	0%	25%	50%	75%	100%
Uncalibrated classifier:	9%	34%	70%	99%	100%

What is Classification Calibration?

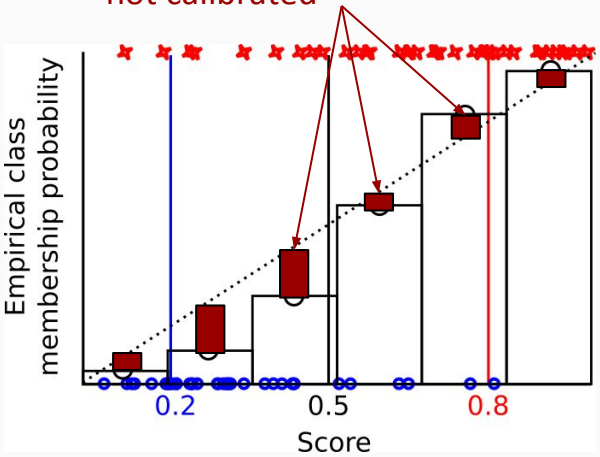


Decision boundary and parallel hyperplanes



Projection to an orthogonal vector

Deviation to diagonal indicates not calibrated



Model scores and empirical probabilities

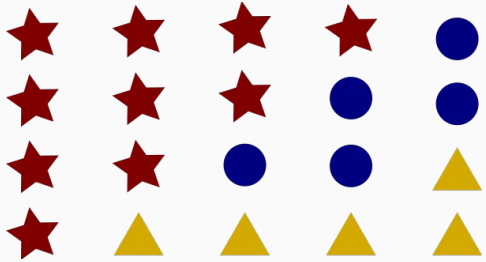
Multiclass Classification Calibration

A probabilistic classifier $\hat{\mathbf{p}}$ is:

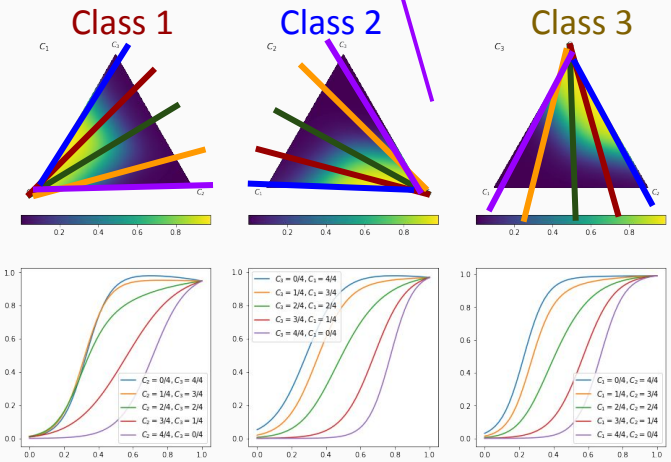
- Multiclass-calibrated if for any prediction vector $\mathbf{q} = (q_1, \ldots, q_k) \in \Delta_k$, the proportions of classes among all possible instances $\mathbf{x}$ getting the same prediction $\hat{\mathbf{p}}(\mathbf{x}) = \mathbf{q}$ are:

$$P(Y = i \mid \hat{\mathbf{p}}(X) = \mathbf{q}) = q_i \quad \text{for } i = 1, \ldots, k.$$

Multiclass example



Empirical proportions Class 1:	100%	75%	50%	25%	0%
Class 2:	0%	0%	25%	50%	50%
Class 3:	0%	25%	25%	25%	50%



UNIVERSITY OF TARTU · University of BRISTOL · Universidade Federal da Paraíba ESTATÍSTICA · sphere · The Alan Turing Institute · PyData Bristol

Why is calibration important? Optimal decision making

Cost Matrix	Predicted True	Predicted False
Actual True	-1£	100£
Actual False	50£	0£

Population proportion
50%
50%

Changes on costs

Changes on proportions

It also allows the decision to abstain

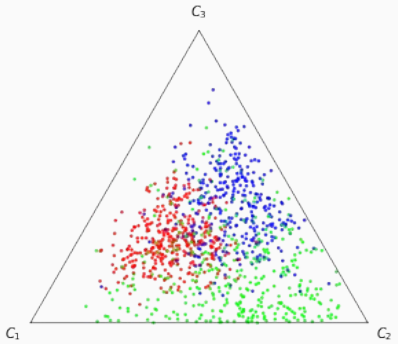
Cost Matrix	Predicted True	Predicted False
Actual True	-10£	90£
Actual False	40£	-5£

Population proportion
70%
30%

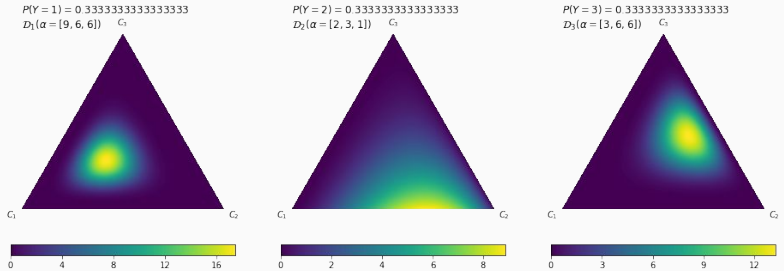
Dirichlet Calibration

We consider the distribution of prediction vectors $\hat{p}(\mathbf{x})$ separately on instances of each class, and assume these are Dirichlet distributions with different parameters:

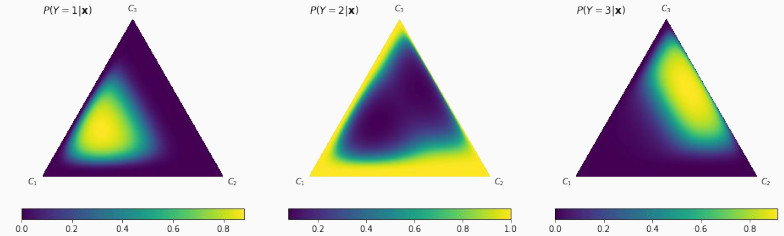
$$\hat{\mathbf{p}}(X) \mid Y = j \sim \text{Dir}(\alpha^{(j)})$$



True density



True posterior probabilities



UNIVERSITY OF TARTU 1632 · University of BRISTOL · Universidade Federal da Paraíba ESTATÍSTICA · sphere · The Alan Turing Institute · PyData Bristol

Evaluation of multiclass calibration is critical

Confidence-calibrated if

$$P\Big(Y = argmax\big(\hat{\mathbf{p}}(X)\big) \,\Big|\, max\big(\hat{\mathbf{p}}(X)\big) = c\Big) = c.$$

Empirically measured as

$$\text{confidence-ECE} = \sum_{i=1}^{m} \frac{|B_i|}{n} |y_j(B_i) - \hat{p}_j(B_i)|$$

Classwise-calibrated if

$$P(Y = i \mid \hat{p}_i(X) = q_i) = q_i.$$

Empirically measured as

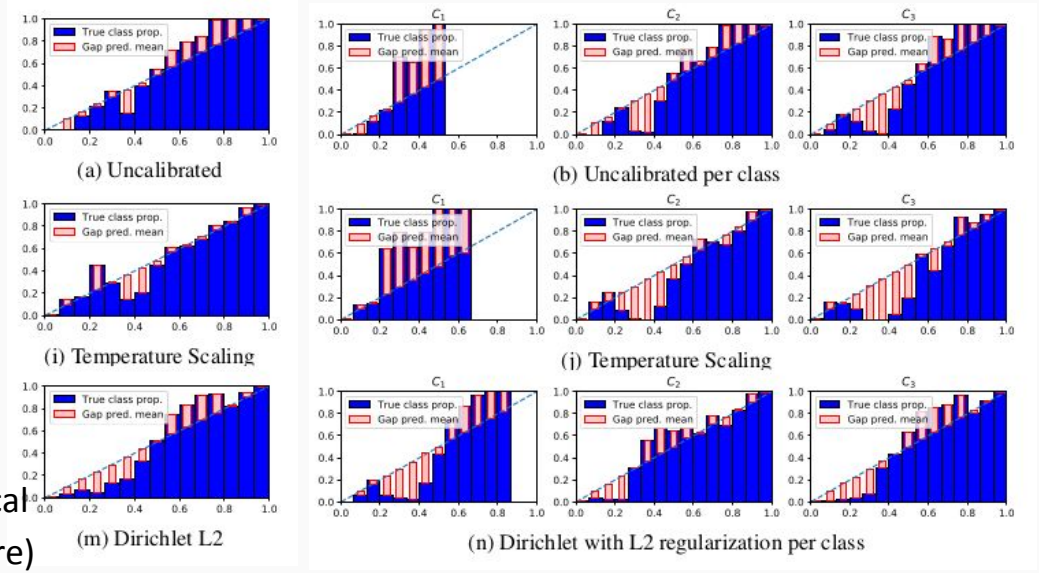
$$\text{classwise}-\text{ECE} = \frac{1}{k} \sum_{j=1}^{k} \sum_{i=1}^{m} \frac{|B_{i,j}|}{n} |y_j(B_{i,j}) - \hat{p}_j(B_{i,j})|$$

Every proper loss is minimised by the canonical calibration function (eg. log-loss and Brier score)

Confidence Calibration Error

Calibration Error Per class

Class 1 Class 2 Class 3



UNIVERSITY OF TARTU

University of BRISTOL

Universidade Federal da Paraíba ESTATÍSTICA

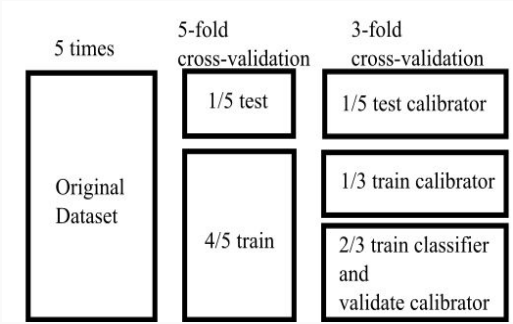
sphere

The Alan Turing Institute

PyData Bristol

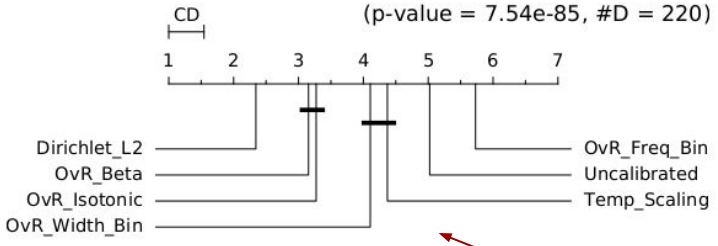
Non-neural Experiments: Settings

- 21 datasets and 11 classifiers = 231 settings
  - Logistic, nbayes, forest, adas, lda, qda, tree, knn, mlp, svc-linear, svc-rbf
- 8 calibration methods (for each of the 231 settings):
  - OvR_Isotonic, OvR_Width_Bin, OvR_Freq_Bin, OvR_Beta, Temp_Scaling, Vect_Scaling, Dirichlet_L2, Dirichlet_ODIR
- 8 evaluation measures
- 5 times 5-fold-crossval.
  - Inner 3-fold-crossval.

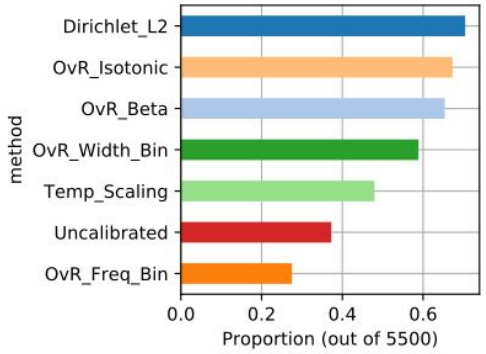


dataset	n_samples	n_features	n_classes
abalone	4177	8	3
balance-scale	625	4	3
car	1728	6	4
cleveland	297	13	5
dermatology	358	34	6
glass	214	9	6
iris	150	4	3
landsat-satellite	6435	36	6
libras-movement	360	90	15
mfeat-karhunen	2000	64	10
mfeat-morphological	2000	6	10
mfeat-zernike	2000	47	10
optdigits	5620	64	10
page-blocks	5473	10	5
pendigits	10992	16	10
segment	2310	19	7
shuttle	101500	9	7
vehicle	846	18	4
vowel	990	10	11
waveform-5000	5000	40	3
yeast	1484	8	10

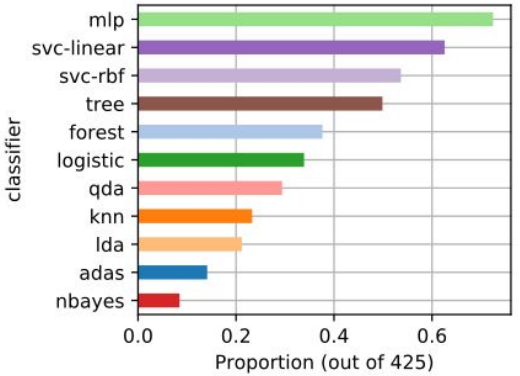
Non-neural Experiments: Results



(a) p-cw-ECE critical difference



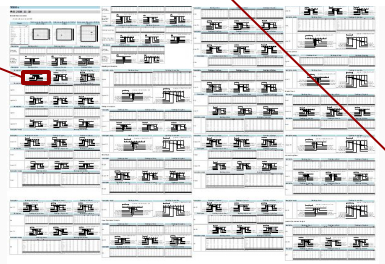
(b) p-cw-ECE for calibrators

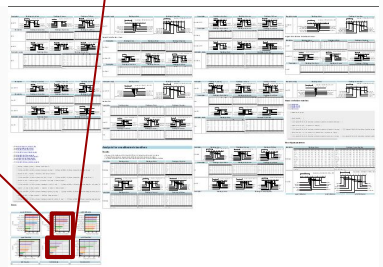


(c) p-cw-ECE for classifiers

These are just 3 figures out of 30 pdf pages of results.

And the 30 pages are only a subset of the generated results.





UNIVERSITY OF TARTU

University of BRISTOL

Universidade Federal da Paraíba ESTATÍSTICA

sphere

Alan Turing Institute

PyData Bristol

Conclusion

1. Dirichlet calibration: New parametric general-purpose multiclass calibration method
    a. Natural extension of the two-class Beta calibration
    b. Easy to implement with multinomial logistic regression on log-transformed class probabilities
2. Our experiments show
    a. Best or tied best performance with 21 datasets x 11 classifiers
    b. Advances state-of-the-art on Neural Networks by introducing ODIR regularisation

For more details check https://dirichletcal.github.io/

UNIVERSITY OF TARTU 1632 University of BRISTOL Universidade Federal da Paraíba ESTATÍSTICA sphere The Alan Turing Institute PyData Bristol

Extra 1: Experimental considerations

- Use random seeds everywhere (models, data partitions, data shuffling)
  - These can be the iteration number
- If comparing two models, train and test in the exact same data partitions
- Use arguments in a user-friendly command-line interface
- Parallelize everything that can be parallelizable (with common sense)
- Store raw results for all executions **before plotting**
- Create a code to summarize all results from raw (csv) files
- Use statistical tests to compare the proposed method

Extra 2: Statistical comparison

- Choose your objective metric based on your problem requirements
  - accuracy, log-loss, mean squared error, expected calibration error…
- Test the compared methods in several scenarios:
  - 21 datasets each one with 11 classifiers = 231 combinations
- Repeat experiments changing dataset partitions and model initializations
  - In my case 5 times 5-fold-cross-validation and inner 3-fold-cross-validation
  - This results in 25 metrics per method that we will use to estimate the expected performance
  - Use the resulting expected values to rank the methods to be compared
- Use the rankings of the compared methods (8) in all combinations (231)
- Use Friedman test statistic to see significance levels between all methods
- Use Bonferroni-Dunn one-tailed statistical test to compute minimum ranking distance to see the Critical Difference Diagram