Calibrators

Classifier Calibration Tutorial

ECML PKDD 2020

Hao Song
hao.song@bristol.ac.uk
classifier-calibration.github.io/
Table of Contents

Start with a toy dataset

Calibrators

Regularisation and Bayesian Treatments

Implementation

Wrap Up
Table of Contents

Start with a toy dataset

Calibrators

Regularisation and Bayesian Treatments

Implementation

Wrap Up
For better illustration, we adopt a toy dataset and a set of visualisations.
The toy dataset (feature space, discriminative view)

As we know the data distribution, we can calculate the Bayes optimal scoring model, that is:

\[ s(x) = P(Y = 1 | X = x) \]
The distribution of predictions

We can further calculate the distribution of $s(x)$. 

![Graphs showing score distribution, class 1 distribution, and class 2 distribution.](image-url)
Reliability diagram

As well as \( P\left( Y = 1 \mid s(x) \right) \) (reliability diagram). Since the model is Bayes optimal, we have a perfect reliability diagram.
With a classifier

Now if we train a neural network with some samples.
Post-hoc calibrators

This is where we need post-hoc calibrators.
Post-hoc calibrators (scaling view)

- It is also common to re-scale a real vector output into calibrated probability vector space. (e.g. SVM margins, final layer of a neural network)
- Since a real vector and a probability vector can be transformed into each other through link function and inverse link function (e.g. soft-max and logit transform), for later slides we will use the probability vector view by default.
Table of Contents

Start with a toy dataset

Calibrators

Regularisation and Bayesian Treatments

Implementation

Wrap Up
Calibrators

Binary approaches:
  Empirical Binning
  Isotonic Regression
  Platt Scaling
  Beta calibration

Multi-class approaches:
  Temperature Scaling
  Vector Scaling
  Matrix Scaling
  Dirichlet calibration

Notable mentions
Binary approaches


Empirical Binning

While being simple and effective, binning approaches can only give discrete outputs.
Empirical Binning

A suitable number of bins / binning algorithm is important to get good results.
Empirical Binning

Parameters:

Bins: \( \{ B_1, \ldots, B_M \} \), \( B_j \subset [0, 1] \)

Bin averages: \( a = (a_1, \ldots, a_M) \), \( a_j \in \{ 0, 1 \} \)

Predictive Function:

\[
c(p; B_1, \ldots, B_M, a) = \sum_{j=1}^{M} \mathbb{I}(p \in B_j) \cdot a_j
\]

Objective Function:

\[
L(a) = \sum_{j=1}^{M} \left| \frac{\sum_{i=1}^{N} \mathbb{I}(p_i \in B_j) \cdot y_i}{|B_j|} - a_j \right|
\]
Isotonic Regression

With the ROC-convex hull method, isotonic regression can give good calibration performance with automatic binning and interpolation.
Isotonic Regression

More data points are beneficial for isotonic regression, but the monotonicity assumption might not be suitable for certain datasets / base models.
Isotonic Regression

Parameters:

Bin edges: \( b = (b_1, \ldots, b_M) \), \( b_j \in \{0, 1\} \), \( b_j < b_{j+1} \)

Edge values: \( v = (v_1, \ldots, v_M) \), \( v_j \in [0, 1] \), \( v_j \leq v_{j+1} \)

Predictive Function:

\[
c(p; b, v) = \frac{1}{\sum_{j=1}^{M-1} I(p \geq b_j) \cdot I(p \leq b_{j+1})} \sum_{j=1}^{M-1} I(p \geq b_j) \cdot I(p < b_{j+1}) \cdot \left( v_j + \frac{p - b_j}{b_{j+1} - b_j} (v_{j+1} - v_j) \right)
\]

Objective Function:

\[
\mathbb{I}(b, v) = \frac{1}{N} \sum_{i=1}^{N} \left( c(p_i; b, v) - y_i \right)^2
\]
Platt Scaling

The ONE that allows SVMs to output probabilities.
Platt Scaling

We can also use it to calibrate from the final layer of the MLP classifier.
Platt Scaling

Parameters:
- Slope: \( w \in \mathbb{R} \)
- Intercept: \( b \in \mathbb{R} \)

Predictive Function:
\[
c(p; w, b) = \frac{1}{1 + \exp(-w \cdot p - b)}
\]

Objective Function:
\[
\mathbb{L}(w, b) = \frac{1}{N} \sum_{i=1}^{N} \ln \left( \sum_{j=1}^{2} \mathbb{I}(y_i = j) \cdot c(p; w, b) \right)
\]
While Platt scaling can be derived with conditional Gaussian distribution with shared variance, probabilities are on a finite support hence Gaussian is less suitable.
Beta Calibration

With a Beta assumption, we can get calibration maps that get beyond sigmoid.

![Diagram showing Beta calibration with different parameters](image-url)
Beta Calibration

As well as a identity map if the original model is already calibrated, or no better calibration map can be modelled.
Beta Calibration

Parameters:
- Slope 1: \( a \in \mathbb{R} \)
- Slope 2: \( b \in \mathbb{R} \)
- Intercept: \( c \in \mathbb{R} \)

Predictive Function:
\[
c(p; a, b, c) = \frac{1}{1 + \exp\left(-a \ln p - b \ln(1-p) - c\right)}
\]

Objective Function:
\[
\mathbb{L}(a, b, c) = \frac{1}{N} \sum_{i=1}^{N} \ln\left(\sum_{j=1}^{2} -\mathbb{I}(y_i = j) \cdot c(p; a, b, c)\right)
\]
Multi-class approaches

  In *34th International Conference on Machine Learning*, pages 1321–1330, Sydney, Australia, 2017

Start with adding one more class

Prediction for class 1

Prediction for class 2

Prediction for class 3
Temperature Scaling

Parameters:
Temperature: $t \in \mathbb{R}$

Predictive Function:
$$c_j(p; t) = \frac{\exp(-t \cdot \text{logit}_j(p))}{\sum_{j=1}^{K} \exp(-t \cdot \text{logit}_j(p))}$$

Objective Function:
$$\mathbb{L}(t) = \frac{1}{N} \sum_{i=1}^{N} \ln \left( \sum_{j=1}^{K} -\mathbb{I}(y_i = j) \cdot c_j(p_i; t) \right)$$
Temperature Scaling
Vector Scaling

Parameters:
Vector: \( w \in \mathbb{R}^K \)
Intercept: \( b \in \mathbb{R}^K \)

Predictive Function:
\[
c_j(p; w) = \frac{\exp(-w_j \cdot \text{logit}_j(p) - b_j)}{\sum_{j=1}^{K} \exp(-w_j \cdot \text{logit}_j(p) - b_j)}
\]

Objective Function:
\[
\mathbb{L}(w, b) = \frac{1}{N} \sum_{i=1}^{N} \ln \left( \sum_{j=1}^{K} -\mathbb{I}(y_i = j) \cdot c_j(p_i; w, b) \right)
\]
Vector Scaling

MLP VS (class 1)

MLP VS (class 2)

MLP VS (class 3)
Matrix Scaling

Parameters:
Matrix: $(w_1, \ldots, w_K), w_j \in \mathbb{R}^K$
Intercept: $b \in \mathbb{R}^K$

Predictive Function:
$$c_j(p; w_1, \ldots, w_K, b) = \frac{\exp(-w_j^T \text{logit}(p) - b_j)}{\sum_{j=1}^K \exp(-w_j^T \text{logit}(p) - b_j)}$$

Objective Function:
$$\mathbb{L}(w_1, \ldots, w_K, b) = \frac{1}{N} \sum_{i=1}^N \ln \left( \sum_{j=1}^K -\mathbb{I}(y_i = j) \cdot c_j(p_i; w_1, \ldots, w_K, b) \right)$$
Matrix Scaling

MLP-MS (class 1)

MLP-MS (class 2)

MLP-MS (class 3)

class 1

class 1

class 1

class 3

class 3

class 3

before calibration
after calibration
true model
Dirichlet Calibration

Parameters:
Coefficients: \((\mathbf{w}_1, \cdots, \mathbf{w}_K), \mathbf{w}_j \in \mathbb{R}^K\)
Intercept: \(\mathbf{b} \in \mathbb{R}^K\)

Predictive Function:
\[
c_j(p; \mathbf{w}_1, \ldots, \mathbf{w}_K, \mathbf{b}) = \frac{\exp(-b_j - \mathbf{w}_j^T \ln p)}{\sum_{j=1}^K \exp(-b_j - \mathbf{w}_j^T \ln p)}
\]

Objective Function:
\[
\mathbb{L}(\mathbf{w}_1, \ldots, \mathbf{w}_K, \mathbf{b}) = \frac{1}{N} \sum_{i=1}^N \ln \left( \sum_{j=1}^K -\mathbb{I}(y_i = j) \cdot c_j(p_i; \mathbf{w}_1, \ldots, \mathbf{w}_K, \mathbf{b}) \right)
\]
Dirichlet Calibration

MLP DIR (class 1)

MLP DIR (class 2)

MLP DIR (class 3)

class 1

class 2

class 3
Notable Mentions


Notable Mentions


Table of Contents

Start with a toy dataset

Calibrators

Regularisation and Bayesian Treatments

Implementation

Wrap Up
Regularisation and Bayesian Treatments

- Typical approaches are generally easy to adopt: $L_0$, $L_1$, $L_2$, lasso, ridge, as well as common Bayesian inference with certain priors on the parameters.
- In Dirichlet calibration, the authors proposed an off-diagonal $L_2$ regularisation approach for Dirichlet calibration, which improves the generalisation of calibration for deep nets by limiting the pair-wise interaction among different classes.
## Table of Contents

- Start with a toy dataset
- Calibrators
- Regularisation and Bayesian Treatments
- Implementation
- Wrap Up
Implementation

There are also some common practices when implementing a calibration approach:

▶ Have multiple inner folds to train the base model and calibrators separately
▶ Approaches including Beta, Dirichlet, and matrix scaling can be easily trained with existing logistic regression implementations.
▶ For calibrators with a convex loss, when the number of data points and classes is manageable, explicit Newton approaches is generally better than stochastic optimisation.
Table of Contents

Start with a toy dataset

Calibrators

Regularisation and Bayesian Treatments

Implementation

Wrap Up
Lessons learned

To select a suitable calibrator, consider the following:

▶ Do you care about the entire probability vector or just about a single class? (the latter → binary approaches)
▶ Do you have a large calibration set? (yes → non-parametric approaches)
▶ Do you have a small calibration set? (yes → consider regularisation)
▶ Are you only interested in certain probability values? (yes → binning approaches)
▶ Any other questions?
What happens next

16.50 - Miquel Perello-Nieto: Hands-on session
17.30 - Peter Flach, Hao Song: Advanced topics and conclusion
   Cost curves; calibrating for F-score; regressor calibration

All times in CEST.
Acknowledgements

► The work of MPN was supported by the SPHERE Next Steps Project funded by the UK Engineering and Physical Sciences Research Council (EPSRC), Grant EP/R005273/1.

► The work of PF and HS was supported by The Alan Turing Institute under EPSRC Grant EP/N510129/1.

► The work of MK was supported by the Estonian Research Council under grant PUT1458.

► The background used in the title slide has been modified by MPN from an original picture by Ed Webster with license CC BY 2.0.
Calibrators

Classifier Calibration Tutorial
ECML PKDD 2020

Hao Song
hao.song@bristol.ac.uk
classifier-calibration.github.io/