Evaluation metrics and proper scoring rules

Classifier Calibration Tutorial
ECML PKDD 2020

Dr. Telmo Silva Filho
telmo@de.ufpb.br
classifier-calibration.github.io/

Departamento de ESTATÍSTICA
University of BRISTOL
UNIVERSITY of TARTU

Table of Contents

Expected/Maximum calibration error
- Binary-ECE/MCE
- Confidence-ECE/MCE
- Classwise-ECE/MCE
- What about multiclass-ECE?

Proper scoring rules
- Definition
- Brier score
- Log-loss
- Decomposition

Hypothesis test for calibration

Summary

Table of Contents

Expected/Maximum calibration error
- Binary-ECE/MCE
- Confidence-ECE/MCE
- Classwise-ECE/MCE
- What about multiclass-ECE?

Proper scoring rules
- Definition
- Brier score
- Log-loss
- Decomposition

Hypothesis test for calibration

Summary

Expected/Maximum calibration error

- As seen in the previous Section, each notion of calibration is related to a reliability diagram
  - This can be used to visualise miscalibration on binned scores
- We will now see how these bins can be used to measure miscalibration

Toy example

- We start by introducing a toy example:

	$\hat{p}_1$	$\hat{p}_2$	$\hat{p}_3$	y
1	1.0	0.0	0.0	1
2	0.9	0.1	0.0	1
3	0.8	0.1	0.1	1
4	0.7	0.1	0.2	1
5	0.6	0.3	0.1	1
6	0.4	0.1	0.5	1
7	1/3	1/3	1/3	1
8	1/3	1/3	1/3	1
9	0.2	0.4	0.4	1
10	0.1	0.5	0.4	1

	$\hat{p}_1$	$\hat{p}_2$	$\hat{p}_3$	y
11	0.8	0.2	0.0	2
12	0.7	0.0	0.3	2
13	0.5	0.2	0.3	2
14	0.4	0.4	0.2	2
15	0.4	0.2	0.4	2
16	0.3	0.4	0.3	2
17	0.2	0.3	0.5	2
18	0.1	0.6	0.3	2
19	0.1	0.3	0.6	2
20	0.0	0.2	0.8	2

	$\hat{p}_1$	$\hat{p}_2$	$\hat{p}_3$	y
21	0.8	0.2	0.0	3
22	0.8	0.1	0.1	3
23	0.8	0.0	0.2	3
24	0.6	0.0	0.4	3
25	0.3	0.0	0.7	3
26	0.2	0.6	0.2	3
27	0.2	0.4	0.4	3
28	0.0	0.4	0.6	3
29	0.0	0.3	0.7	3
30	0.0	0.3	0.7	3

Binary-ECE

- We define the expected binary calibration error binary−ECE (Naeini et al., 2015) as the average gap across all bins in a reliability diagram, weighted by the number of instances in each bin:

$$\text{binary}-\text{ECE} = \sum_{i=1}^{M} \frac{|B_i|}{N} |\bar{y}(B_i) - \bar{p}(B_i)|,$$

- Where M and N are the numbers of bins and instances, respectively, B_i is the i-th probability bin, $|B_i|$ denotes the size of the bin, and $\bar{p}(B_i)$ and $\bar{y}(B_i)$ denote the average predicted probability and the proportion of positives in bin B_i

Binary-MCE

- We can similarly define the maximum binary calibration error binary$-$MCE as the maximum gap across all bins in a reliability diagram:

$$\text{binary}-\text{MCE} = \max_{i \in \{1,\ldots,M\}} |\bar{y}(B_i) - \bar{p}(B_i)|.$$

Binary-ECE using our example

- Let us pretend our example is binary by taking class 1 as positive

	$\hat{p}_1$	$\hat{p}_0$	y
1	1.0	0.0	1
2	0.9	0.1	1
3	0.8	0.2	1
4	0.7	0.3	1
5	0.6	0.4	1
6	0.4	0.6	1
7	1/3	2/3	1
8	1/3	2/3	1
9	0.2	0.8	1
10	0.1	0.9	1

	$\hat{p}_1$	$\hat{p}_0$	y
11	0.8	0.2	0
12	0.7	0.3	0
13	0.5	0.5	0
14	0.4	0.6	0
15	0.4	0.6	0
16	0.3	0.7	0
17	0.2	0.8	0
18	0.1	0.9	0
19	0.1	0.9	0
20	0.0	1.0	0

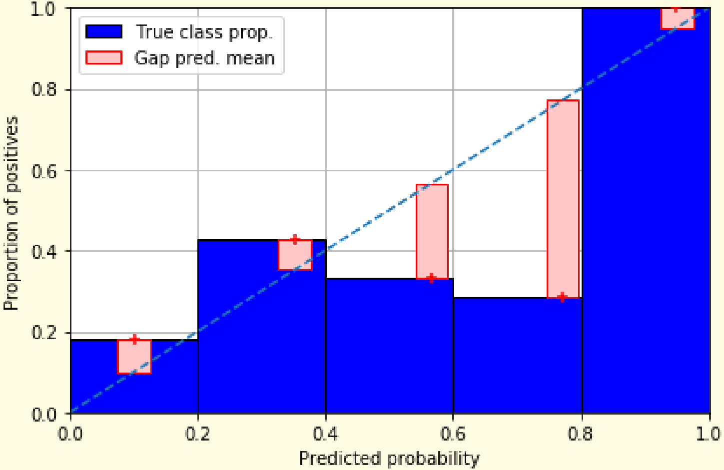
	$\hat{p}_1$	$\hat{p}_0$	y
21	0.8	0.2	0
22	0.8	0.2	0
23	0.8	0.2	0
24	0.6	0.4	0
25	0.3	0.7	0
26	0.2	0.8	0
27	0.2	0.8	0
28	0.0	1.0	0
29	0.0	1.0	0
30	0.0	1.0	0

Binary-ECE using our example

- We now separate class 1 probabilities and their corresponding instance labels into 5 bins: $[0, 0.2], (0.2, 0.4], (0.4, 0.6], (0.6, 0.8], (0.8, 1.0]$
- Then, we calculate the average probability and the frequency of positives at each bin

B_i	$\lvert B_i \rvert$		$\bar{p}(B_i)$		$\bar{y}(B_i)$
B_1	11	0.0, 0.0, 0.0, 0.0, 0.1, 0.1, 0.1, 0.2, 0.2, ...	1.1/11	0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1	2/11
B_2	7	0.3, 0.3, 1/3, 1/3, 0.4, 0.4, 0.4	2.5/7	0, 0, 0, 0, 1, 1, 1	3/7
B_3	3	0.5, 0.6, 0.6	1.7/3	0, 0, 1	1/3
B_4	7	0.7, 0.7, 0.8, 0.8, 0.8, 0.8, 0.8	5.4/7	0, 0, 0, 0, 0, 1, 1	2/7
B_5	2	0.9, 1.0	1.9/2	1, 1	2/2

These same bins can be used to build a reliability diagram



Finally, we calculate the binary-ECE

B_i	$\bar{p}(B_i)$	$\bar{y}(B_i)$	$\|B_i\|$
B_1	0.10	0.18	11
B_2	0.35	0.43	7
B_3	0.57	0.33	3
B_4	0.77	0.29	7
B_5	0.95	1.00	2

$$\text{binary}-\text{ECE} = \sum_{i=1}^{M} \frac{|B_i|}{N} |\bar{y}(B_i) - \bar{p}(B_i)|$$

$$\text{binary}-\text{ECE} = \frac{11 \cdot 0.08 + 7 \cdot 0.08 + 3 \cdot 0.24 + 7 \cdot 0.48 + 2 \cdot 0.05}{30}$$

$$\text{binary}-\text{ECE} = 0.1873$$

Binary-MCE

- For the binary-MCE, we take the maximum gap between $\bar{p}(B_i)$ and $\bar{y}(B_i)$:

B_i	$\bar{p}(B_i)$	$\bar{y}(B_i)$	$\lvert B_i \rvert$
B_1	0.10	0.18	11
B_2	0.35	0.43	7
B_3	0.57	0.33	3
B_4	**0.77**	**0.29**	**7**
B_5	0.95	1.00	2

$$\text{binary}-\text{MCE} = \max_{i \in \{1,\ldots,M\}} \lvert \bar{y}(B_i) - \bar{p}(B_i) \rvert$$

$$\text{binary}-\text{MCE} = 0.48$$

Confidence-ECE

- Confidence-ECE (Guo et al., 2017) was the first attempt at an ECE measure for multiclass problems
- Here, confidence means the probability given to the winning class, i.e. the highest value in the predicted probability vector
- We calculate the expected confidence calibration error $\text{confidence}-\text{ECE}$ as the binary-ECE of the binned confidence values

Confidence-MCE

- We can similarly define the maximum confidence calibration error confidence$-$MCE as the maximum gap across all bins in a reliability diagram:

$$\text{confidence}-\text{MCE} = \max_{i \in \{1,\dots,M\}} |\bar{y}(B_i) - \bar{p}(B_i)|.$$

Confidence-ECE using our example

- First, let us determine the confidence values:

	$\hat{p}_1$	$\hat{p}_2$	$\hat{p}_3$	y
1	**1.0**	0.0	0.0	1
2	**0.9**	0.1	0.0	1
3	**0.8**	0.1	0.1	1
4	**0.7**	0.1	0.2	1
5	**0.6**	0.3	0.1	1
6	0.4	0.1	**0.5**	1
7	**1/3**	1/3	1/3	1
8	**1/3**	1/3	1/3	1
9	0.2	**0.4**	0.4	1
10	0.1	**0.5**	0.4	1

	$\hat{p}_1$	$\hat{p}_2$	$\hat{p}_3$	y
11	**0.8**	0.2	0.0	2
12	**0.7**	0.0	0.3	2
13	**0.5**	0.2	0.3	2
14	**0.4**	0.4	0.2	2
15	**0.4**	0.2	0.4	2
16	0.3	**0.4**	0.3	2
17	0.2	0.3	**0.5**	2
18	0.1	**0.6**	0.3	2
19	0.1	0.3	**0.6**	2
20	0.0	0.2	**0.8**	2

	$\hat{p}_1$	$\hat{p}_2$	$\hat{p}_3$	y
21	**0.8**	0.2	0.0	3
22	**0.8**	0.1	0.1	3
23	**0.8**	0.0	0.2	3
24	**0.6**	0.0	0.4	3
25	0.3	0.0	**0.7**	3
26	0.2	**0.6**	0.2	3
27	0.2	**0.4**	0.4	3
28	0.0	0.4	**0.6**	3
29	0.0	0.3	**0.7**	3
30	0.0	0.3	**0.7**	3

Confidence-ECE using our example

- We binarise the labels by checking if the classifier predicted the right class:

confidence	correct
1.00	1
0.90	1
0.80	1
0.70	1
0.60	1
0.50	0
0.33	1
0.33	1
0.40	0
0.50	0

confidence	correct
0.8	0
0.7	0
0.5	0
0.4	0
0.4	0
0.4	1
0.5	0
0.6	1
0.6	0
0.8	0

confidence	correct
0.8	0
0.8	0
0.8	0
0.6	0
0.7	1
0.6	0
0.4	0
0.6	1
0.7	1
0.7	1

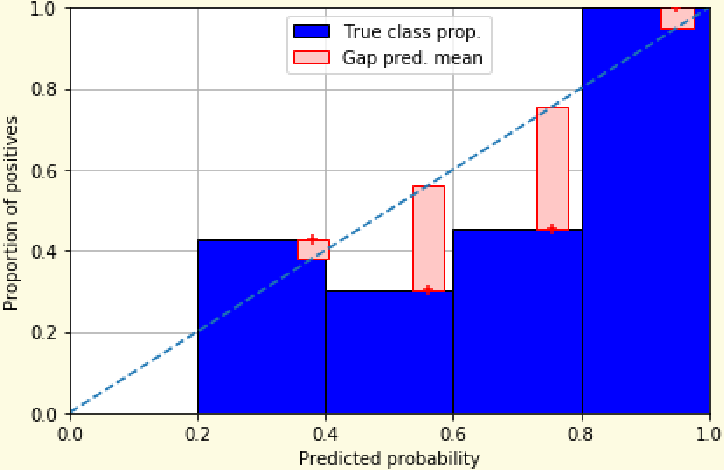
Confidence-ECE using our example

- We now separate the confidences into 5 bins:

B_i	$\lvert B_i \rvert$		$\bar{p}(B_i)$		$\bar{y}(B_i)$
B_1	0				
B_2	7	1/3, 1/3, 0.4, 0.4, 0.4, 0.4, 0.4	2.7/7	0, 0, 0, 0, 1, 1, 1	3/7
B_3	10	0.5, 0.5, 0.5, 0.5, 0.6, 0.6, 0.6, 0.6, 0.6, ...	5.6/10	0, 0, 0, 0, 0, 0, 0, 1, 1, 1	3/10
B_4	11	0.7, 0.7, 0.7, 0.7, 0.7, 0.8, 0.8, 0.8, 0.8, ...	8.3/11	0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1	5/11
B_5	2	0.9, 1.0	1.9/2	1, 1	2/2

- Note that bins that correspond to confidences **less than $1/K$** will always be **empty**

The corresponding reliability diagram



Finally, we calculate the confidence-ECE

B_i	$\bar{p}(B_i)$	$\bar{y}(B_i)$	$\lvert B_i \rvert$
B_1			0
B_2	0.38	0.43	7
B_3	0.56	0.30	10
B_4	0.75	0.45	11
B_5	0.95	1.00	2

$$\text{confidence}-\text{ECE} = \sum_{i=1}^{M} \frac{|B_i|}{N} |\bar{y}(B_i) - \bar{p}(B_i)|$$

$$\text{confidence}-\text{ECE} = \frac{0 + 7 \cdot 0.05 + 10 \cdot 0.26 + 11 \cdot 0.3 + 2 \cdot 0.05}{30}$$

$$\text{confidence}-\text{ECE} = 0.2117$$

Confidence-MCE

- For the confidence-MCE, we take the maximum gap between $\bar{p}(B_i)$ and $\bar{y}(B_i)$:

B_i	$\bar{p}(B_i)$	$\bar{y}(B_i)$	$\lvert B_i \rvert$
B_1			0
B_2	0.38	0.43	7
B_3	0.56	0.30	10
B_4	**0.75**	**0.45**	**11**
B_5	0.95	1.00	2

$$\text{confidence}-\text{MCE} = \max_{i \in \{1,\ldots,M\}} \lvert \bar{y}(B_i) - \bar{p}(B_i) \rvert$$

$$\text{confidence}-\text{MCE} = 0.3$$

Classwise-ECE

- Confidence calibration only cares about the winning class
- To measure miscalibration for all classes, we can take the average binary-ECE across all classes
- The contribution of a single class j to this expected classwise calibration error (classwise$-$ECE) is called **class-j-ECE**

Classwise-ECE

- Formally, classwise$-$ECE is defined as the average gap across all classwise-reliability diagrams, weighted by the number of instances in each bin:

$$\text{classwise}-\text{ECE} = \frac{1}{K}\sum_{j=1}^{K}\sum_{i=1}^{M}\frac{|B_{i,j}|}{N}|\bar{y}_j(B_{i,j}) - \bar{p}_j(B_{i,j})|,$$

- Where $B_{i,j}$ is the i-th bin of the j-th class, $|B_{i,j}|$ denotes the size of the bin, and $\bar{p}_j(B_{i,j})$ and $\bar{y}_j(B_{i,j})$ denote the average prediction of class j probability and the actual proportion of class j in the bin $B_{i,j}$

Classwise-MCE

- Similarly the maximum classwise calibration error (classwise$-$MCE) is defined as the maximum gap across all bins and all classwise-reliability diagrams:

$$\text{classwise}-\text{MCE} = \max_{j \in \{1,\ldots,K\}\, i \in \{1,\ldots,M\}} |\bar{y}_j(B_{i,j}) - \bar{p}_j(B_{i,j})|.$$

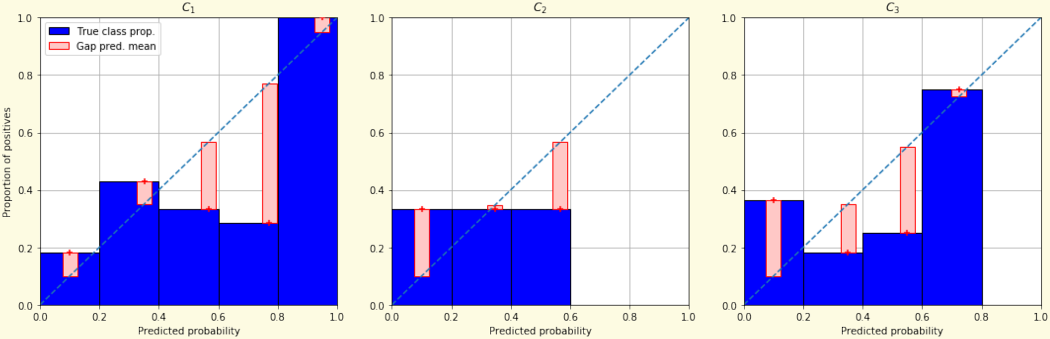
Classwise-ECE using our example

- We have already calculated class-1-ECE (0.1873) in our binary-ECE example
- Now we need to do the same for classes 2 and 3

$B_{i,2}$	$\|B_{i,2}\|$		$\bar{p}(B_{i,2})$		$\bar{y}(B_{i,2})$
$B_{1,2}$	15	0.0, 0.0, 0.0, 0.0, 0.0, 0.1, 0.1, 0.1, 0.1, ...	1.5/15	0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1	5/15
$B_{2,2}$	12	0.3, 0.3, 0.3, 0.3, 0.3, 1/3, 1/3, 0.4, 0.4...	4.2/12	0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1	4/12
$B_{3,2}$	3	0.5, 0.6, 0.6	1.7/3	0, 0, 1	1/3
$B_{4,2}$	0				
$B_{5,2}$	0				

$B_{i,3}$	$\|B_{i,3}\|$		$\bar{p}(B_{i,3})$		$\bar{y}(B_{i,3})$
$B_{1,3}$	11	0.0, 0.0, 0.0, 0.0, 0.1, 0.1, 0.1, 0.2, 0.2, ...	1.1/11	0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1	4/11
$B_{2,3}$	11	0.3, 0.3, 0.3, 0.3, 1/3, 1/3, 0.4, 0.4, 0.4...	3.9/11	0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1	2/11
$B_{3,3}$	4	0.5, 0.5, 0.6, 0.6	2.2/4	0, 0, 0, 1	1/4
$B_{4,3}$	4	0.7, 0.7, 0.7, 0.8	2.9/4	0, 1, 1, 1	3/4
$B_{5,3}$	0				

Each class has its own reliability diagram



Now we calculate class-2-ECE and class-3-ECE

$$\text{class}-2-\text{ECE} = \sum_{i=1}^{M} \frac{|B_{i,2}|}{N} |\bar{y}(B_{i,2}) - \bar{p}(B_{i,2})|$$

$$\text{class}-2-\text{ECE} = \frac{15 \cdot 0.23 + 12 \cdot 0.02 + 3\ \dot{0}.24 + 0 + 0}{30}$$

$$\text{class}-2-\text{ECE} = 0.147$$

$$\text{class}-3-\text{ECE} = \sum_{i=1}^{M} \frac{|B_{i,3}|}{N} |\bar{y}(B_{i,3}) - \bar{p}(B_{i,3})|$$

$$\text{class}-3-\text{ECE} = \frac{11 \cdot 0.26 + 11 \cdot 0.17 + 4 \cdot 0.3 + 4 \cdot 0.03 + 0}{30}$$

$$\text{class}-3-\text{ECE} = 0.2017$$

Finally, we take the mean of the 3 ECEs

$$\text{classwise}-\text{ECE} = \frac{1}{K}\sum_{j=1}^{K}\sum_{i=1}^{M}\frac{|B_{i,j}|}{N}|\bar{y}_j(B_{i,j}) - \bar{p}_j(B_{i,j})|$$

$$\text{classwise}-\text{ECE} = \frac{0.1873 + 0.147 + 0.2017}{3}$$

$$\text{classwise}-\text{ECE} = 0.1787$$

Classwise-MCE

- For the classwise-MCE, we take the maximum gap between $\bar{p}(B_{i,j})$ and $\bar{y}(B_{i,j})$ across all bins of all classes:

$B_{i,1}$	$\bar{p}(B_{i,1})$	$\bar{y}(B_{i,1})$	$\lvert B_{i,1}\rvert$
$B_{1,1}$	0.10	0.18	11
$B_{2,1}$	0.35	0.43	7
$B_{3,1}$	0.57	0.33	3
$B_{4,1}$	**0.77**	**0.29**	**7**
$B_{5,1}$	0.95	1.00	2

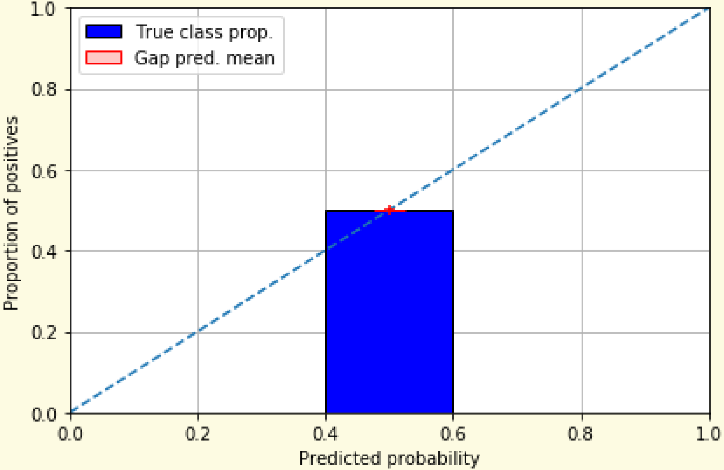
$B_{i,2}$	$\bar{p}(B_{i,2})$	$\bar{y}(B_{i,2})$	$\lvert B_{i,2}\rvert$
$B_{1,2}$	0.10	0.33	15
$B_{2,2}$	0.35	0.33	12
$B_{3,2}$	0.57	0.33	3
$B_{4,2}$			0
$B_{5,2}$			0

$B_{i,3}$	$\bar{p}(B_{i,3})$	$\bar{y}(B_{i,3})$	$\lvert B_{i,3}\rvert$
$B_{1,3}$	0.10	0.36	11
$B_{2,3}$	0.35	0.18	11
$B_{3,3}$	0.55	0.25	4
$B_{4,3}$	0.72	0.75	4
$B_{5,3}$			0

$$\text{classwise}-\text{MCE} = \max_{j\in\{1,\ldots,K\}\, i\in\{1,\ldots,M\}} |\bar{y}_j(B_{i,j}) - \bar{p}_j(B_{i,j})|$$

$$\text{classwise}-\text{MCE} = 0.48$$

Optimising ECE can be as simple as predicting the overall class distribution, regardless of the given instance



What about multiclass-ECE?

- True multiclass-ECE is still an open problem
- With large numbers of classes, the number of bins can be prohibitively high
  - Most bins would be empty
- Therefore, we turn to **proper scoring rules**

Table of Contents

Expected/Maximum calibration error

- Binary-ECE/MCE
- Confidence-ECE/MCE
- Classwise-ECE/MCE
- What about multiclass-ECE?

Proper scoring rules

- Definition
- Brier score
- Log-loss
- Decomposition

Hypothesis test for calibration

Summary

Proper scoring rules

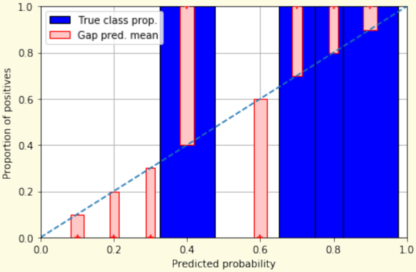
- We now talk about loss measures ($\breve{\phi}$) that prefer Bayes-optimal classifiers over other classifiers
- For any given $P(\mathbf{X}, Y)$, $\mathbf{x} \in \mathcal{X}$, the following is satisfied:

$$\mathbb{E}_{y \sim P(Y|\mathbf{X}=\mathbf{x})}\left[\breve{\phi}(\mathbf{q}, y)\right] \geq \mathbb{E}_{y \sim P(Y|\mathbf{X}=\mathbf{x})}\left[\breve{\phi}\left(P(Y \mid \mathbf{X} = \mathbf{x}), y\right)\right]$$

- And the left side is equal to right side if and only if $\mathbf{q} = P(Y \mid \mathbf{X} = \mathbf{x})$
- $P(Y \mid \mathbf{X} = \mathbf{x})$ is a vector with elements $P(Y = j \mid \mathbf{X} = \mathbf{x})$

Proper scoring rules

- Proper scoring rules are calculated at the item level, while ECE measures are averages across bins
- Think of them as putting **each item in its separate bin**, then computing the average of some **loss** for each **predicted probability** and its corresponding **observed label**
  - Instead of the absolute difference, as in ECE, this loss can be the quadratic error or the Kullback–Leibler divergence, which have better mathematical properties



Brier score/Quadratic error/Euclidean distance

$$\breve{\phi}_{\text{BS}}\left(\mathbf{Q}, \mathbf{y}\right) = \frac{1}{N}\sum_{n=1}^{N}\sum_{j=1}^{K}\left(\mathbb{I}(y_n = j) - q_{n,j}\right)^2$$

- We can easily see that this value is not minimised by constantly predicting the class distribution, as in ECE

$$\mathbf{Q} = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}$$

$$\mathbf{y} = [1, 2]$$

$$\breve{\phi}_{\text{BS}}\left(\mathbf{Q}, \mathbf{y}\right) = \frac{(1-0.5)^2 + (0-0.5)^2 + (0-0.5)^2 + (1-0.5)^2}{2}$$

$$\breve{\phi}_{\text{BS}}\left(\mathbf{Q}, \mathbf{y}\right) = 0.5$$

Log-loss/Cross entropy

$$\breve{\phi}_{\text{LL}}\left(\mathbf{Q}, \mathbf{y}\right) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{j=1}^{K}\mathbb{I}(y_n = j) \cdot \log(q_{n,j})$$

- Frequently used to as the training loss of machine learning methods, such as neural networks
- Only penalises the probability given to the true class

$$\breve{\phi}_{\text{LL}}\left(\mathbf{Q}, \mathbf{y}\right) = -\frac{(1 \cdot \log(0.5) + 0 \cdot \log(0.5) + 0 \cdot \log(0.5) + 1 \cdot \log(0.5))}{2}$$

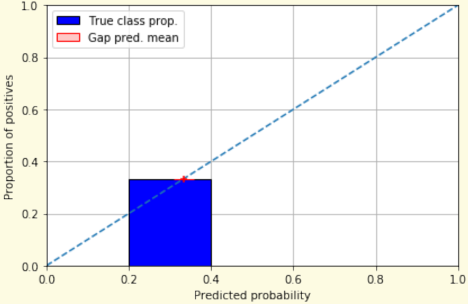
$$\breve{\phi}_{\text{LL}}\left(\mathbf{Q}, \mathbf{y}\right) = 0.6931$$

Let us rewind a bit

- As mentioned before, a model that always outputs the class proportion will have a perfect ECE of 0, but its log-loss is not 0 (in fact, it's 0.6365)

$$\mathbf{Q} = \begin{bmatrix} 2/3 & 1/3 \\ 2/3 & 1/3 \\ \vdots & \vdots \\ 2/3 & 1/3 \end{bmatrix}$$

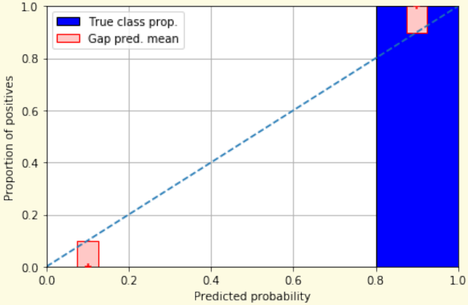
$$\begin{aligned}\mathbf{y} = [&1, 1, 1, 1, 1, 1, 1, 1, 1, 1,\\ &0, 0, 0, 0, 0, 0, 0, 0, 0, 0,\\ &0, 0, 0, 0, 0, 0, 0, 0, 0, 0]\end{aligned}$$



An evaluation trade-off

- What happens if our model gives 0.9 probability to the instances' true classes?

$$\text{accuracy} = 1$$
$$\text{ECE} = 0.1$$
$$\text{log-loss} = 0.1054$$



Proper scoring rule decomposition

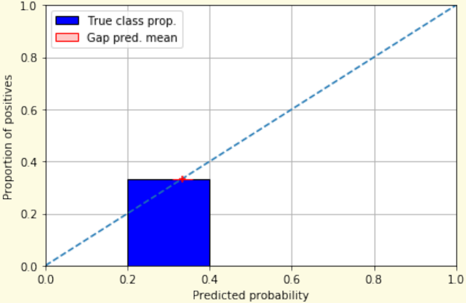
- ECE increased (0 to 0.1), but log-loss decreased (0.6365 to 0.1054)
- So why did log-loss decrease?
  - Because proper scoring rules do not measure only calibration
  - In fact, they can be decomposed into terms with different interpretations (Kull and Flach, 2015)

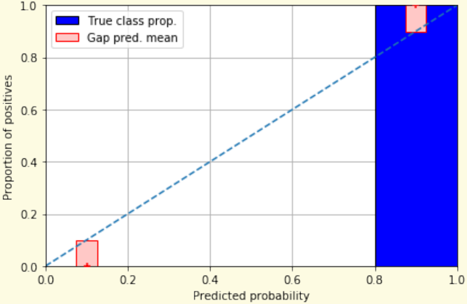
Proper scoring rule decomposition

- An intuitive way to decompose proper scoring rules is into refinement and calibration losses: $\mathbb{E}\left[\breve{\phi}\right] = \mathsf{RL} + \mathsf{CL}$
  - Refinement loss: is the loss due to producing the same probability for instances from different classes
  - Calibration loss: is the loss due to the difference between the probabilities predicted by the model and the proportion of positives among instances with the same output

Proper scoring rule decomposition

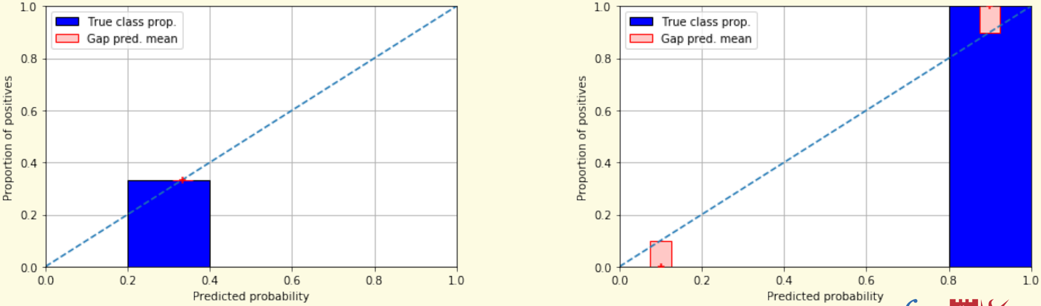
- An intuitive way to decompose proper scoring rules is into refinement and calibration losses: $\mathbb{E}\left[\breve{\phi}\right] = \text{RL} + \text{CL}$
  - Refinement loss: is the loss due to producing the same probability for instances from different classes (**the second model reduces this loss**)





Proper scoring rule decomposition

- An intuitive way to decompose proper scoring rules is into refinement and calibration losses: $\mathbb{E}\left[\breve{\phi}\right] = \text{RL} + \text{CL}$
  - Calibration loss: is the loss due to the difference between the probabilities predicted by the model and the proportion of positives among instances with the same output (**the second model increases this loss**)



Proper scoring rule decomposition

- Since we don't usually know the real score distribution, we would need to once again rely on binning if we wanted to actually estimate refinement and calibration losses
- Additionally, the terms are calculated (estimated) differently, depending on the proper scoring rule
- **Fun fact:** the loss of the optimal classifier is not necessarily 0
  - This is due to irreducible loss, which is only 0 if the attributes provide enough information to uniquely determine the instances' right label Y, with probability 1 (Kull and Flach, 2015)

Table of Contents

Expected/Maximum calibration error
- Binary-ECE/MCE
- Confidence-ECE/MCE
- Classwise-ECE/MCE
- What about multiclass-ECE?

Proper scoring rules
- Definition
- Brier score
- Log-loss
- Decomposition

Hypothesis test for calibration

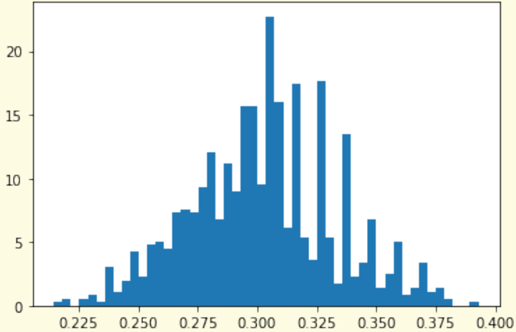
Summary

Hypothesis test for calibration

- Given a classifier $\hat{\boldsymbol{p}}$, we can check if its predictions for a test set $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)\}$ are calibrated according to an arbitrary loss measure $\phi(\hat{\boldsymbol{p}}(X_{\text{test}}), \boldsymbol{y}_{\text{test}})$, such as ECE, log-loss or Brier score

Calculating the p-value

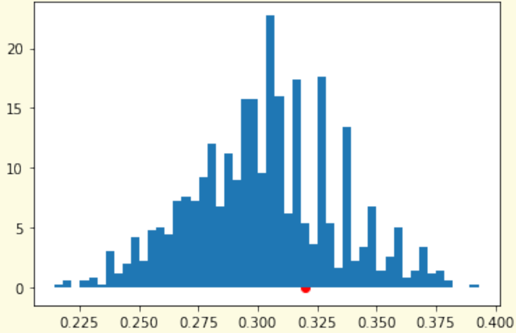
- We use a simple resampling-based hypothesis test under the null hypothesis that the classifier's outputs are calibrated (Vaicenavicius et al., 2019)
- First, we generate S bootstrapped label sets $\boldsymbol{y}_s$, $s \in \{1, \ldots, S\}$, such that each $y_{s,i}$ is sampled from $\hat{\boldsymbol{p}}(\hat{\boldsymbol{x}}_i)$
- Then we calculate $\phi(\hat{\boldsymbol{p}}(X_{\text{test}}), \boldsymbol{y}_s)$ for each label set s



Calculating the p-value

- We then calculate the p-value as:

$$P\Big(\phi(\hat{\boldsymbol{p}}(X_{\text{test}}), \boldsymbol{y}_s) > \phi(\hat{\boldsymbol{p}}(X_{\text{test}}), \boldsymbol{y}_{\text{test}})\Big) = P\Big(\phi(\hat{\boldsymbol{p}}(X_{\text{test}}), \boldsymbol{y}_s) > 0.32\Big) \tag{1}$$

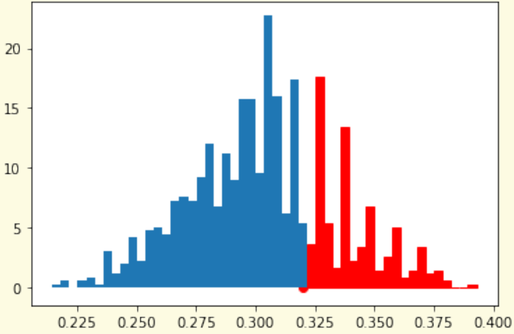


Calculating the p-value

- We then calculate the p-value as:

$$P\Big(\phi(\hat{\boldsymbol{p}}(X_{\text{test}}), \boldsymbol{y}_s) > 0.32\Big) \approx 0.26 \tag{2}$$

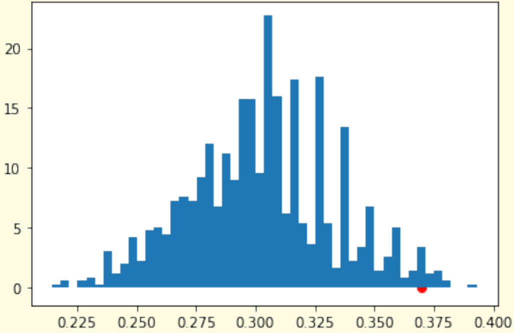
- We cannot reject the null hypothesis here



Calculating the p-value

- Now suppose the original labels were such that our classifier's classwise-ECE had a value of 0.37

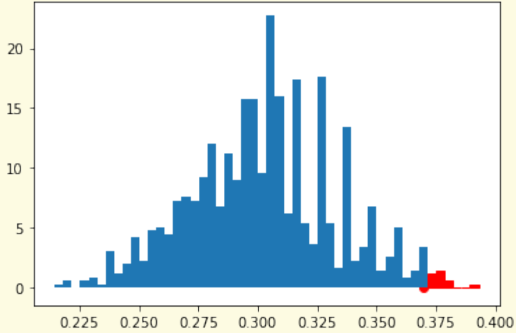
$$P\left(\phi(\hat{\boldsymbol{p}}(X_{\text{test}}), \boldsymbol{y}_s) > 0.37\right). \tag{3}$$



Calculating the p-value

- Now suppose that our classifier's classwise-ece had a value of 0.37

$$P\left(\phi(\hat{\boldsymbol{p}}(X_{\text{test}}), \boldsymbol{y}_s) > 0.37\right) \approx 0.01 \tag{4}$$



Calculating the p-value

- Now suppose the original labels were such that our classifier's classwise-ece had a value of 0.37

$$P\left(\phi(\hat{\boldsymbol{p}}(X_{\text{test}}), \boldsymbol{y}_s) > 0.37\right) \approx 0.01 \tag{5}$$

- We reject the null hypothesis: the model is **miscalibrated**

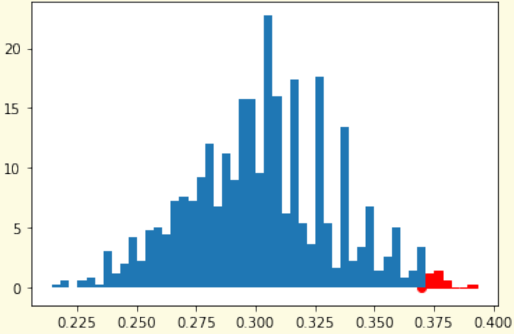


Table of Contents

Expected/Maximum calibration error
- Binary-ECE/MCE
- Confidence-ECE/MCE
- Classwise-ECE/MCE
- What about multiclass-ECE?

Proper scoring rules
- Definition
- Brier score
- Log-loss
- Decomposition

Hypothesis test for calibration

Summary

Summary

- There are various ways to visualise and quantify calibration
- ECE measures aim at producing an aggregate measure of the visual information provided in reliability diagrams
  - Thus, their optimisation is **not** guaranteed to produce desirable classifiers
- Proper scoring rules measure different aspects of probability correctness
  - They have been used as training losses in classifier training for a while
  - But they cannot tell **“where”** the model is more miscalibrated
- Finally, the hypothesis test for calibration can help determine if a particular loss value means that the classifier is calibrated or not

What happens next

15.30 - Break and preparation for hands-on session

15.50 - Hao Song: Calibrators

Binary approaches; multi-class approaches;
regularisation and Bayesian treatments; implementation

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.

References

C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On Calibration of Modern Neural Networks. In *34th International Conference on Machine Learning*, pages 1321–1330, Sydney, Australia, 2017. URL `https://dl.acm.org/citation.cfm?id=3305518`.

M. Kull and P. Flach. Novel decompositions of proper scoring rules for classification: Score adjustment as precursor to calibration. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD'15)*, volume 9284, pages 68–85. Springer Verlag, 2015.

P. Naeini, G. F. Cooper, and M. Hauskrecht. Obtaining Well Calibrated Probabilities Using Bayesian Binning. In *29th AAAI Conference on Artificial Intelligence*, feb 2015. URL `www.aaai.org`.

J. Vaicenavicius, D. Widmann, C. Andersson, F. Lindsten, J. Roll, and T. B. Schön. Evaluating model calibration in classification. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3459–3467, 2019. URL `https://github.com/uu-sml/`.

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.

Evaluation metrics and proper scoring rules

Classifier Calibration Tutorial
ECML PKDD 2020

Dr. Telmo Silva Filho
telmo@de.ufpb.br
classifier-calibration.github.io/

Departamento de ESTATÍSTICA
University of BRISTOL
UNIVERSITY OF TARTU