From Uncertainty to Precision: Enhancing Binary Classifier Performance through Calibration.

Agathe Fernandes Machado

Thursday, May 16th, 2024

UQÀM

Calibration: intuition (1/2)

"*There is a 30% chance of rain tomorrow.*" Dawid (1982)

Calibration: intuition (1/2)

"*There is a 30% chance of rain tomorrow.*" Dawid (1982)

Figure 1: Weather Forecasts on Tuesday, March 2024. Source: The Weather Channel.

Calibration: intuition (1/2)

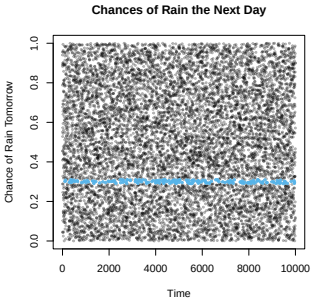
"*There is a 30% chance of rain tomorrow.*" Dawid (1982)

Figure 1: Weather Forecasts on Tuesday, March 2024. Source: The Weather Channel.

Consider a sequence of weather forecasts $\hat{s}(\mathbf{x}_t)$, where $t = 1, \ldots, T$ denotes the days of forecast and $\mathbf{x}$ represents characteristics used in forecasting.

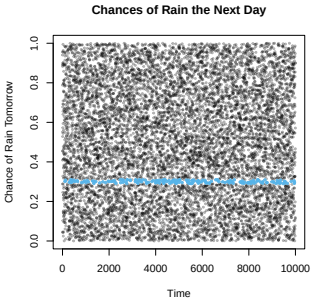
Calibration: intuition (2/2)

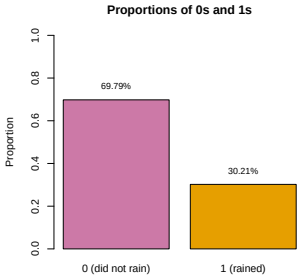
Within this sequence, we focus on days where $\hat{s}(\mathbf{x}_i)$ closely approximates 30%.
By assuming an infinite sequence, we can determine the long-term proportion p of days where the forecasted event actually occurred.



Calibration: intuition (2/2)

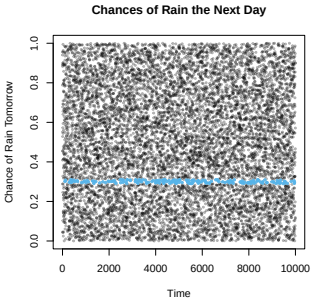
Within this sequence, we focus on days where $\hat{s}(\mathbf{x}_i)$ closely approximates 30%.
By assuming an infinite sequence, we can determine the long-term proportion p of days where the forecasted event actually occurred.

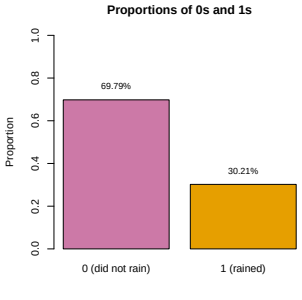


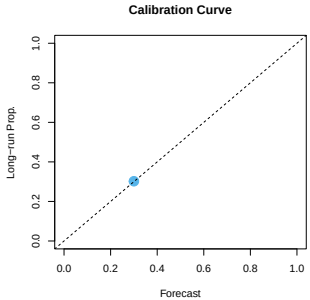


Calibration: intuition (2/2)

Within this sequence, we focus on days where $\hat{s}(\mathbf{x}_i)$ closely approximates 30%.
By assuming an infinite sequence, we can determine the long-term proportion p of days where the forecasted event actually occurred.

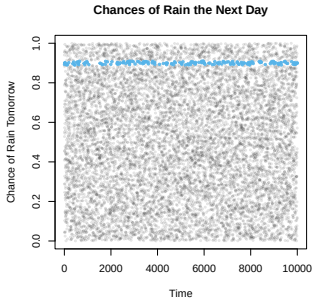


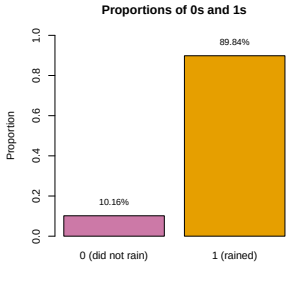


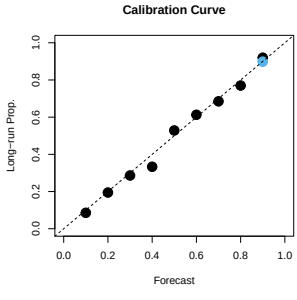


Calibration: intuition (2/2)

Within this sequence, we focus on days where $\hat{s}(\mathbf{x}_i)$ closely approximates 30%.
By assuming an infinite sequence, we can determine the long-term proportion p of days where the forecasted event actually occurred.







Motivations

- We are interested in being able to **discriminate** between rainy/not rainy days.

Motivations

- We are interested in being able to **discriminate** between rainy/not rainy days.
- We are also interested in the **underlying risk**. Other examples include:

Motivations

- We are interested in being able to **discriminate** between rainy/not rainy days.
- We are also interested in the **underlying risk**. Other examples include:
  - does this patient have a disease or not (Calster et al. (2019))?

Motivations

- We are interested in being able to **discriminate** between rainy/not rainy days.
- We are also interested in the **underlying risk**. Other examples include:
  - does this patient have a disease or not (Calster et al. (2019))?
  - will this insured have an accident within the next year?

Motivations

- We are interested in being able to **discriminate** between rainy/not rainy days.
- We are also interested in the **underlying risk**. Other examples include:
  - does this patient have a disease or not (Calster et al. (2019))?
  - will this insured have an accident within the next year?
  - what is the probability for this individual to receive the treatment/control?

Motivations

- We are interested in being able to **discriminate** between rainy/not rainy days.
- We are also interested in the **underlying risk**. Other examples include:
  - does this patient have a disease or not (Calster et al. (2019))?
  - will this insured have an accident within the next year?
  - what is the probability for this individual to receive the treatment/control?
- In such cases, it is important that the **estimated scores** can be interpreted as **probabilities**.

Motivations

- We are interested in being able to **discriminate** between rainy/not rainy days.
- We are also interested in the **underlying risk**. Other examples include:
  - does this patient have a disease or not (Calster et al. (2019))?
  - will this insured have an accident within the next year?
  - what is the probability for this individual to receive the treatment/control?
- In such cases, it is important that the **estimated scores** can be interpreted as **probabilities**.
- This might become a problem when using **machine learning classifiers** based on **ensemble methods**.

This Talk

What the remainder of the talk is about:

- Reviewing of ways to **measure and visualise calibration** for a **binary classifier**.

This Talk

What the remainder of the talk is about:

- Reviewing of ways to **measure and visualise calibration** for a **binary classifier**.
- Proposing a new metric based on **local regression**: the Local Calibration Score.

This Talk

What the remainder of the talk is about:

- Reviewing of ways to **measure and visualise calibration** for a **binary classifier**.
- Proposing a new metric based on **local regression**: the Local Calibration Score.
- Observing the **impact of a poor calibration** on standard performance metrics.

This Talk

What the remainder of the talk is about:

- Reviewing of ways to **measure and visualise calibration** for a **binary classifier**.
- Proposing a new metric based on **local regression**: the Local Calibration Score.
- Observing the **impact of a poor calibration** on standard performance metrics.
- Examining calibration for **tree-based methods**.

Main results

- Our new metric, the **local calibration score** offers a more flexible way to visualise and measure calibration than methods based on empirical quantiles.
- **Calibration matters**: when training classifiers, looking at calibration of models should not be disregarded.

Roadmap

Setup

- Let us consider a **binary event** D whose observations are denoted $d_i = 1$ if the event occurs, and $d_i = 0$ otherwise, where i denotes the ith observations.

Setup

- Let us consider a **binary event** D whose observations are denoted $d_i = 1$ if the event occurs, and $d_i = 0$ otherwise, where i denotes the ith observations.
- Let us further assume that the (**unobserved**) probability of the event $d_i = 1$ depends on **individual characteristics**:

$$p_i = s(\mathbf{x}_i)$$

where, with sample size $n > 0$, $i = 1, \ldots, n$ represents individuals, and $\mathbf{x}_i$ the characteristics.

Setup

- Let us consider a **binary event** D whose observations are denoted $d_i = 1$ if the event occurs, and $d_i = 0$ otherwise, where i denotes the ith observations.
- Let us further assume that the (**unobserved**) probability of the event $d_i = 1$ depends on **individual characteristics**:

$$p_i = s(\mathbf{x}_i)$$

where, with sample size $n > 0$, $i = 1, \ldots, n$ represents individuals, and $\mathbf{x}_i$ the characteristics.

- To **estimate this probability**, we can use a statistical model (*e.g.*, a GLM) or a machine learning model (*e.g.*, a random forest).

Definition

Calibration of a Binary Classifier (Schervish (1989))

For a binary variable D, a model is well-calibrated when

$$\mathbb{E}[D \mid \hat{s}(\mathbf{X}) = p] = p, \quad \forall p \in [0, 1] \ . \tag{1}$$

Definition

Calibration of a Binary Classifier (Schervish (1989))

For a binary variable D, a model is well-calibrated when

$$\mathbb{E}[D \mid \hat{s}(\mathbf{X}) = p] = p, \quad \forall p \in [0, 1] \ . \tag{1}$$

Note: conditioning by $\{\hat{s}(\mathbf{x}) = p\}$ leads to the concept of (local) calibration; however, as discussed by Bai et al. (2021), $\{\hat{s}(\mathbf{x}) = p\}$ is *a.s.* a null mass event. Thus, calibration should be understood in the sense that

$$\mathbb{E}[D \mid \hat{s}(\mathbf{X}) = p] \overset{a.s.}{\to} p \text{ when } n \to \infty \ ,$$

meaning that, asymptotically, the model is well-calibrated, or locally well-calibrated in p, for any p.

Visual approach: calibration curve

- Estimation of $\mathrm{g}(\cdot)$ (which measures **miscalibration** on predicted scores $\hat{s}(\mathbf{x})$):

$$\mathrm{g} : \begin{cases} [0,1] \to [0,1] \\ p \mapsto \mathrm{g}(p) := \mathbb{E}[D \mid \hat{s}(\mathbf{x}) = p] \end{cases} . \tag{2}$$

- **Challenge**: having enough observations with identical scores is difficult.
- **Solution**: grouping obs. into B **bins**, defined by the **quantiles** of predicted scores:
  - The average of observed values ($\bar{d}_b$ with $b \in \{1, \ldots, B\}$), in each bin b can then be compared with the central value of the bin.
  - **Calibration curve** (reliability diagram (Wilks (1990)): middle of each bin on the x-axis, averages of corresponding observations on the y-axis.
  - When the model is **well-calibrated**, all B points lie on the **bisector**.

Metrics (1/2)

Expected Calibration Error or ECE (Pakdaman Naeini et al. (2015))

$$\text{ECE} = \sum_{b=1}^{B} \frac{n_b}{n} \mid \text{acc}(b) - \text{conf}(b) \mid$$

where n is the sample size, n_b is the number of observations in bin $b \in \{1, \ldots, B\}$.

Metrics (1/2)

Expected Calibration Error or ECE (Pakdaman Naeini et al. (2015))

$$\text{ECE} = \sum_{b=1}^{B} \frac{n_b}{n} \mid \text{acc}(b) - \text{conf}(b) \mid$$

where n is the sample size, n_b is the number of observations in bin $b \in \{1, \ldots, B\}$.

Accuracy $\text{acc}(b)$: The average of empirical probabilities or fractions of correctly predicted classes.

$$\text{acc}(b) = \frac{1}{n_b} \sum_{i \in \mathcal{I}_b} \mathbb{1}_{\hat{d}_i = d_i} \qquad (3)$$

The predicted class $\hat{d}_i$ for observation i is determined based on a classification threshold $\tau \in [0, 1]$ where $\hat{d}_i = 1$ if $\hat{s}(\mathbf{x}_i) \geq \tau$ and 0 otherwise.

Metrics (1/2)

Expected Calibration Error or ECE (Pakdaman Naeini et al. (2015))

$$\text{ECE} = \sum_{b=1}^{B} \frac{n_b}{n} \mid \text{acc}(b) - \text{conf}(b) \mid$$

where n is the sample size, n_b is the number of observations in bin $b \in \{1, \ldots, B\}$.

Accuracy $\text{acc}(b)$: The average of empirical probabilities or fractions of correctly predicted classes.

$$\text{acc}(b) = \frac{1}{n_b} \sum_{i \in \mathcal{I}_b} \mathbb{1}_{\hat{d}_i = d_i} \quad (3)$$

The predicted class $\hat{d}_i$ for observation i is determined based on a classification threshold $\tau \in [0, 1]$ where $\hat{d}_i = 1$ if $\hat{s}(\mathbf{x}_i) \geq \tau$ and 0 otherwise.

Confidence $\text{conf}(b)$: Indicates the model's average confidence within bin b by averaging predicted scores.

$$\text{conf}(b) = \frac{1}{n_b} \sum_{i \in \mathcal{I}_b} \hat{s}(\mathbf{x}_i)$$

Metrics (2/2)

Brier Score (Brier (1950))

The **Brier Score** does not depend on bins and is defined as:

$$BS = \frac{1}{n}\sum_{i=1}^{n}(d_i - \hat{s}(\mathbf{x}_i))^2 \tag{4}$$

where d_i is the observed event and $\hat{s}(\mathbf{x}_i)$ the estimated score.

Metrics (2/2)

Brier Score (Brier (1950))

The **Brier Score** does not depend on bins and is defined as:

$$BS = \frac{1}{n}\sum_{i=1}^{n}(d_i - \hat{s}(\mathbf{x}_i))^2 \tag{4}$$

where d_i is the observed event and $\hat{s}(\mathbf{x}_i)$ the estimated score.

Mean Squared Error (MSE)

By substituting the observed event d_i by the true probability p_i (which can only be observed in an experimental setup), the metric becomes the MSE:

$$\text{True MSE} = \frac{1}{n}\sum_{i=1}^{n}(p_i - \hat{s}(\mathbf{x}_i))^2 \tag{5}$$

Our Approach: Smoother Visualization Technique

We propose an alternative approach to visualize model calibration, aiming for a **smoother representation**: **local regression** (Loader (1999)).

- Measuring calibration consists in estimating a **conditional expectation**: a local regression seems appropriate.

Our Approach: Smoother Visualization Technique

We propose an alternative approach to visualize model calibration, aiming for a **smoother representation**: **local regression** (Loader (1999)).

- Measuring calibration consists in estimating a **conditional expectation**: a local regression seems appropriate.
- Local Regression have been disregarded in high dimensions due to poor properties, but it is **highly efficient in small dimensions**, as in this case with only one predictive feature, $\hat{s}(\boldsymbol{x}) \in [0, 1]$.

Our Approach: Smoother Visualization Technique

We propose an alternative approach to visualize model calibration, aiming for a **smoother representation**: **local regression** (Loader (1999)).

- Measuring calibration consists in estimating a **conditional expectation**: a local regression seems appropriate.
- Local Regression have been disregarded in high dimensions due to poor properties, but it is **highly efficient in small dimensions**, as in this case with only one predictive feature, $\hat{s}(\boldsymbol{x}) \in [0, 1]$.
- Given the number of data points, the precision of quantile binning can be suboptimal when determining the appropriate bin count.

Our Approach: Smoother Visualization Technique

We propose an alternative approach to visualize model calibration, aiming for a **smoother representation**: **local regression** (Loader (1999)).

- Measuring calibration consists in estimating a **conditional expectation**: a local regression seems appropriate.
- Local Regression have been disregarded in high dimensions due to poor properties, but it is **highly efficient in small dimensions**, as in this case with only one predictive feature, $\hat{s}(\boldsymbol{x}) \in [0, 1]$.
- Given the number of data points, the precision of quantile binning can be suboptimal when determining the appropriate bin count.
- By contrast, with local regression, one can specify the percentage of nearest neighbors, providing greater flexibility.

Our new metric: LCS

Local Calibration Score (LCS)

A local regression of degree 0, denoted as $\hat{g}$, is fitted to the predicted scores $\hat{s}(\mathbf{x})$.
This fit is then applied to a vector of **linearly spaced values** within the interval $[0, 1]$.
Each of these points is denoted by l_j, where $j \in \{1, \ldots, J\}$, with J being the target number of points on the visualization curve.
The LCS is defined as:

$$\text{LCS} = \sum_{j=1}^{J} w_j \left(\hat{g}(l_j) - l_j\right)^2, \tag{6}$$

where w_j is a weight defined as the density of the *score* at l_j.

Our new metric: LCS

Local Calibration Score (LCS)

A local regression of degree 0, denoted as $\hat{g}$, is fitted to the predicted scores $\hat{s}(\mathbf{x})$.
This fit is then applied to a vector of **linearly spaced values** within the interval $[0, 1]$.
Each of these points is denoted by l_j, where $j \in \{1, \ldots, J\}$, with J being the target number of points on the visualization curve.
The LCS is defined as:

$$\text{LCS} = \sum_{j=1}^{J} w_j \left(\hat{g}(l_j) - l_j\right)^2, \tag{6}$$

where w_j is a weight defined as the density of the *score* at l_j.

Note: Austin and Steyerberg (2019) defined a similar metric using a L1 norm.

Data Generating Process

We **simulate** binary observations as in Gutman et al. (2022):

$$D_i \sim \mathcal{B}(p_i),$$

where individual probabilities are obtained using a logistic sigmoid function:

$$p_i = \frac{1}{1 + \exp(-\eta_i)},$$
$$\eta_i = \mathbf{a}\mathbf{x}_i + \varepsilon_i$$

with $\mathbf{a} = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \end{bmatrix} = \begin{bmatrix} 0.1 & 0.05 & 0.2 & -0.05 \end{bmatrix}$ and $\mathbf{x}_i = \begin{bmatrix} x_{1,i} & x_{2,i} & x_{3,i} & x_{4,i} \end{bmatrix}^\top$.
The observations $\mathbf{x}_i$ are drawn from a $\mathcal{U}(0,1)$ and $\varepsilon_i \sim \mathcal{N}(0, 0.5^2)$.

Forcing Poor Calibration

To simulate **uncalibration**, we generate samples of $2,000$ observations and we apply (monotonous) transformations to the true probabilities, either on:

1. the latent probability p_i:

$$p_i^u = \left(\frac{1}{1 + \exp(-\eta_i)} \right)^{\alpha} \quad . \tag{7}$$

2. the linear predictor η_i:

$$\eta_i^u = \gamma \times ((-0.1)x_1 + 0.05x_2 + 0.2x_3 - 0.05x_4 + \varepsilon_i) \quad . \tag{8}$$

Forcing Poor Calibration

To simulate **uncalibration**, we generate samples of $2,000$ observations and we apply (monotonous) transformations to the true probabilities, either on:

1. the latent probability p_i:

$$p_i^u = \left(\frac{1}{1 + \exp(-\eta_i)} \right)^{\alpha} \quad . \tag{7}$$

2. the linear predictor η_i:

$$\eta_i^u = \gamma \times ((-0.1)x_1 + 0.05x_2 + 0.2x_3 - 0.05x_4 + \varepsilon_i) \quad . \tag{8}$$

The resulting transformed probabilities are considered as the scores: $\hat{s}(\mathbf{x}) := p_i^u$

Distortions

- We examine variations in $\{1/3, 1, 3\}$ for α and γ
- For each of the 6 scenarios, we generate 200 samples of $2,000$ obs.

Distortions

- We examine variations in $\{1/3, 1, 3\}$ for α and γ
- For each of the 6 scenarios, we generate 200 samples of $2,000$ obs.

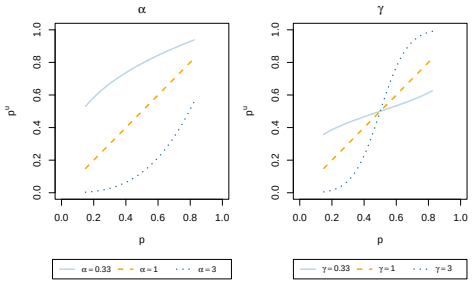


Figure 2: Distorted Prob. as a Function of True Prob., Depending on the Value of α (left) or γ (right)

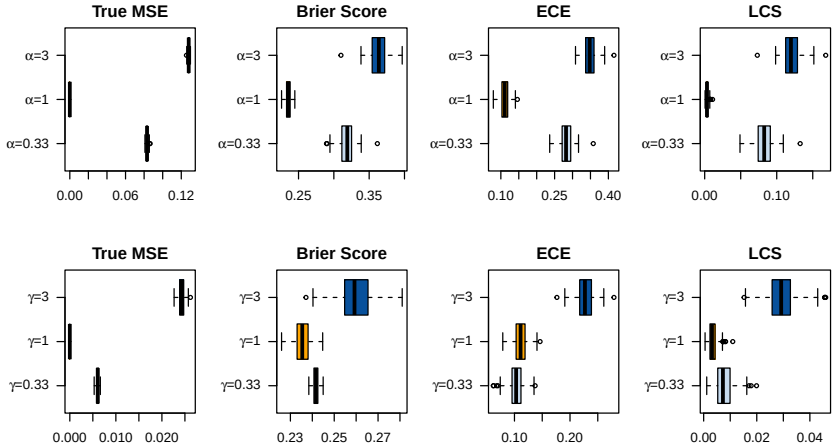


Figure 3: Calibration Metrics on 200 Simulations for each Value of α (top) or γ (bottom).

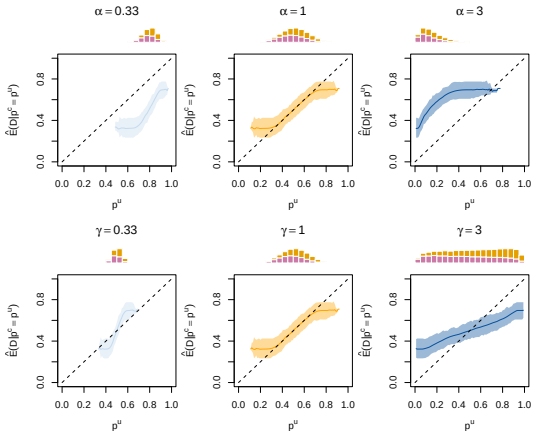


Figure 4: Calibration Curve Obtained with Local Regression, on 200 simulations for each Value of α (top) or γ (bottom). Distribution of the true probabilities are shown in the histograms (gold for $d = 1$, purple for $d = 0$).

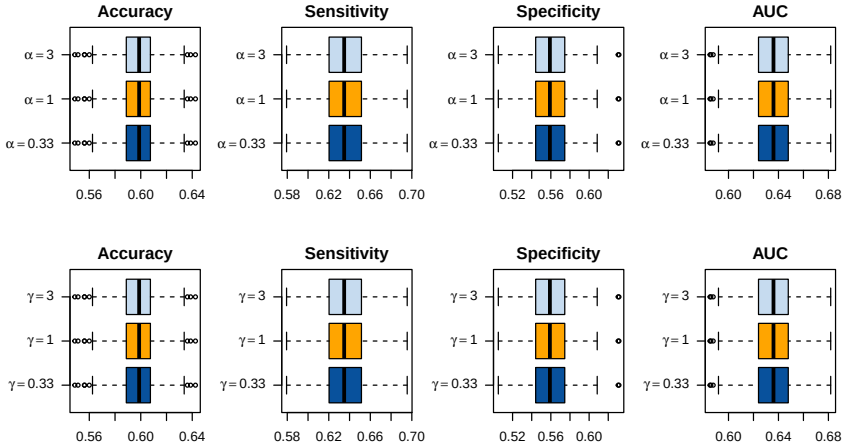


Figure 5: Standard Goodness of Fit Metrics on 200 Simulations for each Value of α (top) or γ (bottom). The probability threshold is set to $\tau = 0.5$.

Calibration for Machine Learning Algorithms

- With the promise of better performance from machine learning models, it can be tempting to rely on these types of models, such as random forests or derivatives, to estimate binary events.

Calibration for Machine Learning Algorithms

- With the promise of better performance from machine learning models, it can be tempting to rely on these types of models, such as random forests or derivatives, to estimate binary events.
- However, the distribution of scores returned by these models can be far from the distribution of the underlying probabilities.

Calibration for Machine Learning Algorithms

- With the promise of better performance from machine learning models, it can be tempting to rely on these types of models, such as random forests or derivatives, to estimate binary events.
- However, the distribution of scores returned by these models can be far from the distribution of the underlying probabilities.
- Here we present an overview of the preliminary results we have obtained with regression trees.

Calibration for Machine Learning Algorithms

- With the promise of better performance from machine learning models, it can be tempting to rely on these types of models, such as random forests or derivatives, to estimate binary events.
- However, the distribution of scores returned by these models can be far from the distribution of the underlying probabilities.
- Here we present an overview of the preliminary results we have obtained with regression trees.
- More results in the next version of the paper...

Trees

Are trees well calibrated?

Trees

Are trees well calibrated?

- Some learning algorithms are designed to yield well-calibrated probabilities. These include **decision trees**, whose leaf probabilities are optimal on the training set (Kull et al. (2017))
- Earlier studies show that also classical methods such as **decision trees**, boosting, SVMs and naive Bayes classifiers tend to be miscalibrated (Wenger et al. (2020))

Preliminary Results

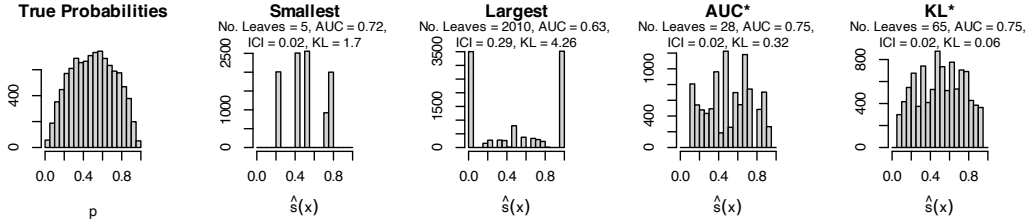


Figure 6: Distribution of true probabilities and estimated scores on validation set for trees of interest

Wrap up

- Our new metric, the **local calibration score** offers a more flexible way to visualise and measure calibration than methods based on empirical quantiles.
- **Calibration matters**: when training classifiers, looking at calibration of models should not be disregarded.
- Next steps: complete calibration with KL divergence for tree-based methods

Comments are welcome: `fernandes_machado.agathe@courrier.uqam.ca`

5 Appendix

References I

Austin, P. C. and Steyerberg, E. W. (2019). The integrated calibration index (ici) and related metrics for quantifying the calibration of logistic regression models. *Statistics in Medicine* 38: 4051–4065, doi:10.1002/sim.8281.

Bai, Y., Mei, S., Wang, H. and Xiong, C. (2021). Don' t just blame over-parametrization for over-confidence: Theoretical analysis of calibration in binary classification. In *International Conference on Machine Learning*. PMLR, 566–576.

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly Weather Review* 78: 1–3.

Calster, B. V., , McLernon, D. J., Smeden, M. van, Wynants, L. and Steyerberg, E. W. (2019). Calibration: the achilles heel of predictive analytics. *BMC Medicine* 17, doi:10.1186/s12916-019-1466-7.

Dawid, A. P. (1982). The well-calibrated bayesian. *Journal of the American Statistical Association* 77: 605–610.

Gutman, R., Karavani, E. and Shimoni, Y. (2022). Propensity score models are better when post-calibrated.

Kull, M., Filho, T. M. S. and Flach, P. (2017). Beyond sigmoids: How to obtain well-calibrated probabilities from binary classifiers with beta calibration. *Electronic Journal of Statistics* 11: 5052 – 5080, doi:10.1214/17-EJS1338SI.

Loader, C. (1999). *Fitting with LOCFIT*. New York, NY: Springer New York, chap. 3. 45–58.

References II

Pakdaman Naeini, M., Cooper, G. and Hauskrecht, M. (2015). Obtaining well calibrated probabilities using bayesian binning. *Proceedings of the AAAI Conference on Artificial Intelligence* 29: 2901–2907, doi:10.1609/aaai.v29i1.9602.

Schervish, M. J. (1989). A General Method for Comparing Probability Assessors. *The Annals of Statistics* 17: 1856–1879, doi:10.1214/aos/1176347398.

Wenger, J., Kjellström, H. and Triebel, R. (2020). Non-Parametric Calibration for Classification. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, Proceedings of Machine Learning Research.

Wilks, D. S. (1990). On the combination of forecast probabilities for consecutive precipitation periods. *Weather and Forecasting* 5: 640–650, doi:10.1175/1520-0434(1990)005<0640:OTCOFP>2.0.CO;2.

(Mis-)Calibration and standard metrics

What are the impacts of miscalibration on standard metrics?
We will consider metrics based on the predictive performances calculated using a confusion table:

Table 1: Confusion Table

Actual/Predicted	**Positive**	**Negative**
Positive	TP	FN
Negative	FP	TN

where

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}; \quad \text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

(Mis-)Calibration and standard metrics

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{N}}$$

Overall correctness of the model

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

Ability to correctly identify positive class

$$\text{Specificity} = \textit{TPR} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

Ability to correctly identify negative class

AUC (Area Under Curve)

TPR and TFP for various prob. threshold τ