MODEL CALIBRATION

KARPOV.COURSES

DEFINITION

While most classifiers can produce a probability distribution for a given test instance, these probabilities are often not well calibrated, i.e., they may not be representative of the true probability of the instance belonging to a particular class

$$P(\mathrm{y} = \mathrm{j} \mid \mathrm{x}) = 0.8$$

For example, for those test instances x that are assigned a probability of belonging to class j, we should expect approximately 80% to actually belong to class j.

CALIBRATION

This is a useful technique for many applications, and is widely used in practice. For example, in a cost-sensitive classification setting, accurate probability estimates for each class are necessary to minimise the total cost.

It can also be important to have well calibrated class probability estimates if these estimates are used in conjunction with other data as input to another model.

Lastly, when data is highly unbalanced by class, probability estimates can be skewed towards the majority class, leading to poor scores for metrics such as F1 and poor threshold.

People tend to use output scores as probabilities. But they are not by default!

PROBABILITY CALIBRATION TREES

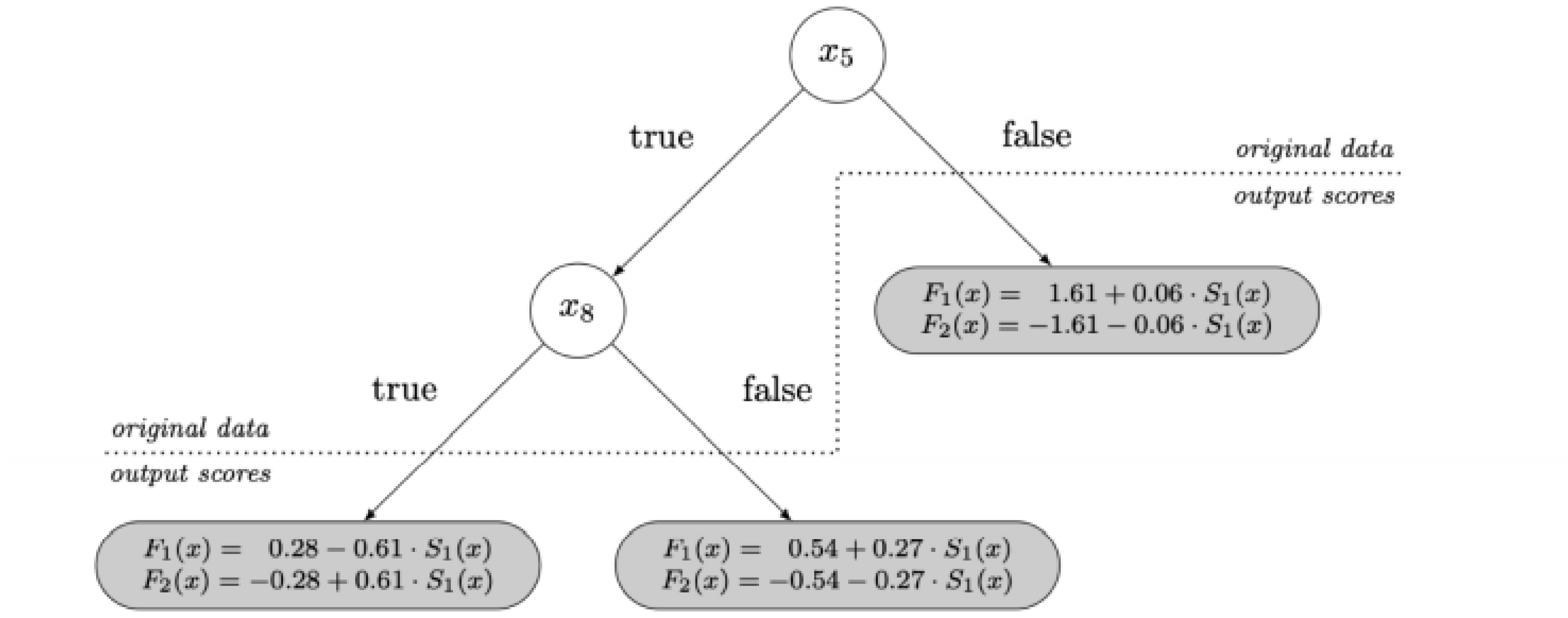


Figure 2: A probability calibration tree for the outputs of an SVM with an RBF kernel ($C = 10, \gamma = 0.01$) on the RDG1 dataset. RDG1 is a small two-class dataset with 10 binary attributes, and can be generated in the WEKA software using the eponymous data generator. x_5 and x_8 are attributes in the original data, while $S_1(x)$ is the output score of the SVM. The functions $F_i(x)$ compute the calibrated log-odds estimate of x belonging to class i, and must sum to zero. The final calibrated probabilities are computed with Equation 2.

TAXONOMY

Histogram Binning

Isotonic Regression

Platt calibration

Matrix and Vector Scaling

Scaling-binning calibrator

Probability calibration trees

Temperature Scaling

Maximum Mean Calibration Error

Label smoothing

Entropy penalty

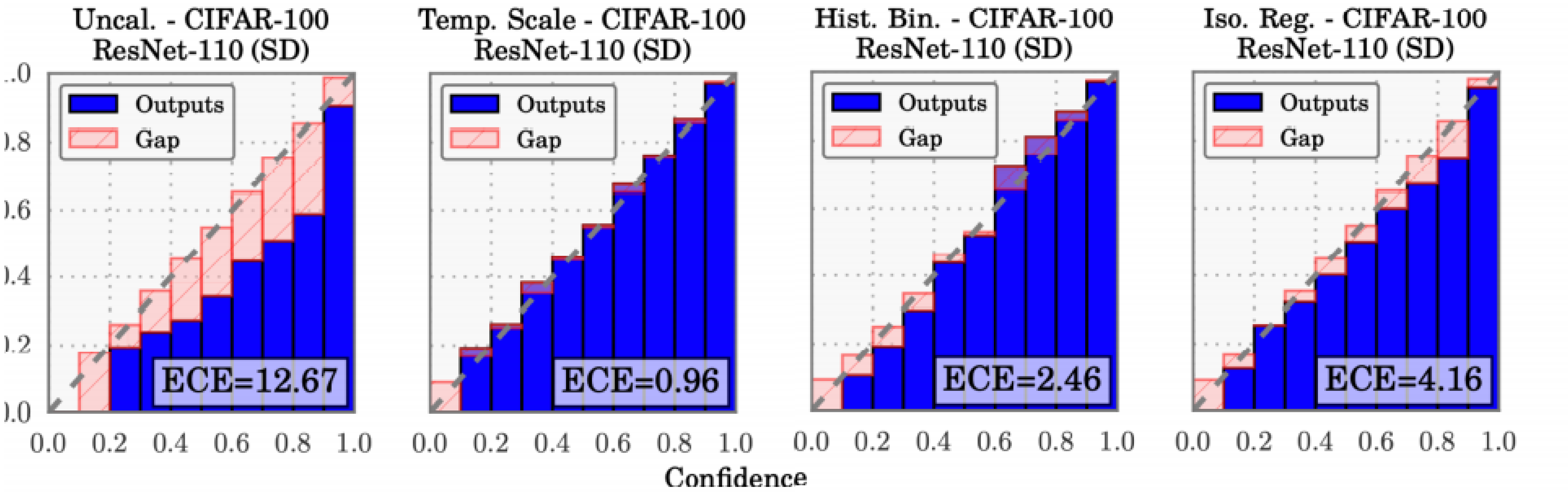
Focal Loss

DropOut

TEMPERATURE SCALING

Temperature scaling, the simplest extension of Platt scaling, uses a single scalar parameter $T > 0$ for all classes. Given the logit vector $\mathbf{z}_i$, the new confidence prediction is

$$\hat{q}_i = \max_k \sigma_{\mathrm{SM}}(\mathbf{z}_i/T)^{(k)}. \qquad (9)$$



LABEL SMOOTHING

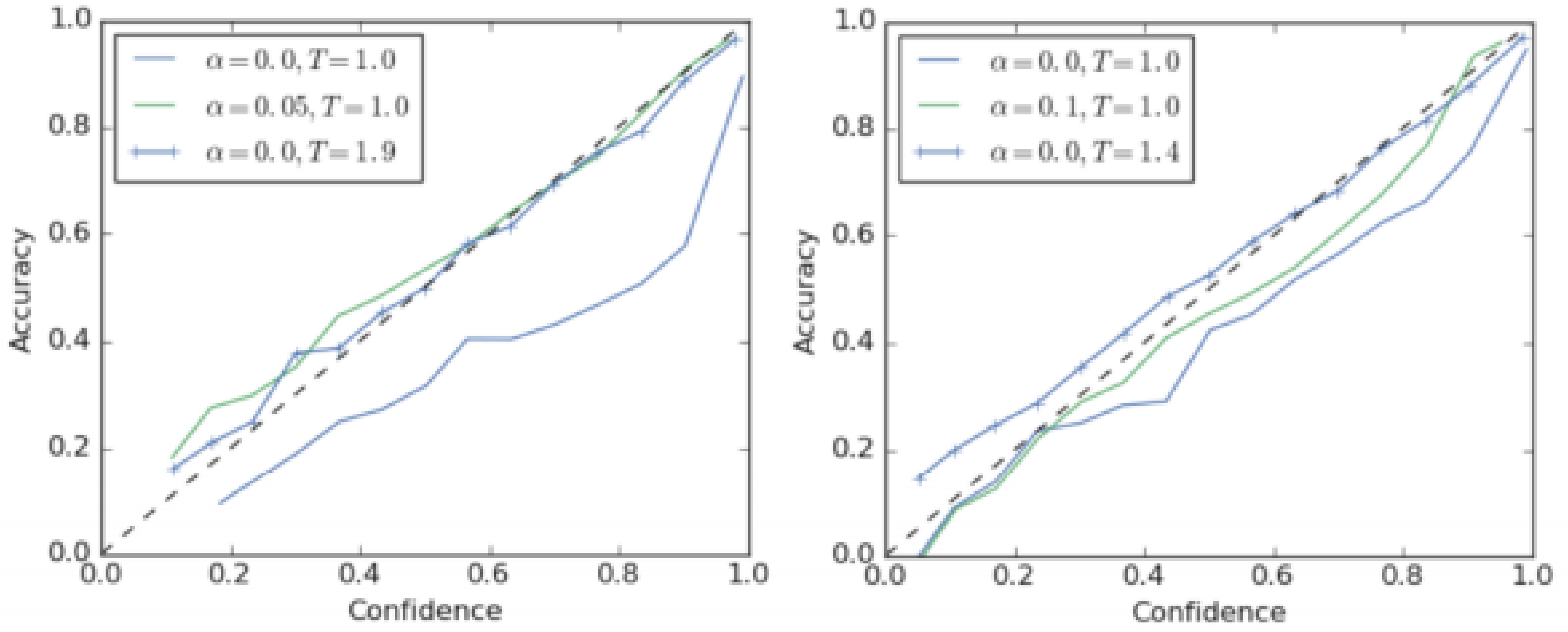


Figure 2: Reliability diagram of ResNet-56/CIFAR-100 (left) and Inception-v4/ImageNet (right).

Table 3: Expected calibration error (ECE) on different architectures/datasets.

DATA SET	ARCHITECTURE	BASELINE ECE (T=1.0, $\alpha = 0.0$)	TEMP. SCALING ECE / T ($\alpha = 0.0$)	LABEL SMOOTHING ECE / α (T=1.0)
CIFAR-100	RESNET-56	0.150	0.021 / 1.9	0.024 / 0.05
IMAGENET	INCEPTION-V4	0.071	0.022 / 1.4	0.035 / 0.1
EN-DE	TRANSFORMER	0.056	0.018 / 1.13	0.019 / 0.1

- It is easy to calibrate the model
- Calibration makes your output probabilistic
- Calibrated model has a natural threshold
- Calibrated outputs are ready to use for stacking