nPlan

KDD2020
VIRTUAL CONFERENCE

1 2 3 4 5 6 7

How to Calibrate your Neural Network Classifier:

Getting True Probabilities from a Classification Model

Natalia Culakova
natalia@nplan.io

Dan Murphy
dan@nplan.io

Alan Mosca
alan@nplan.io

Tutorial Overview

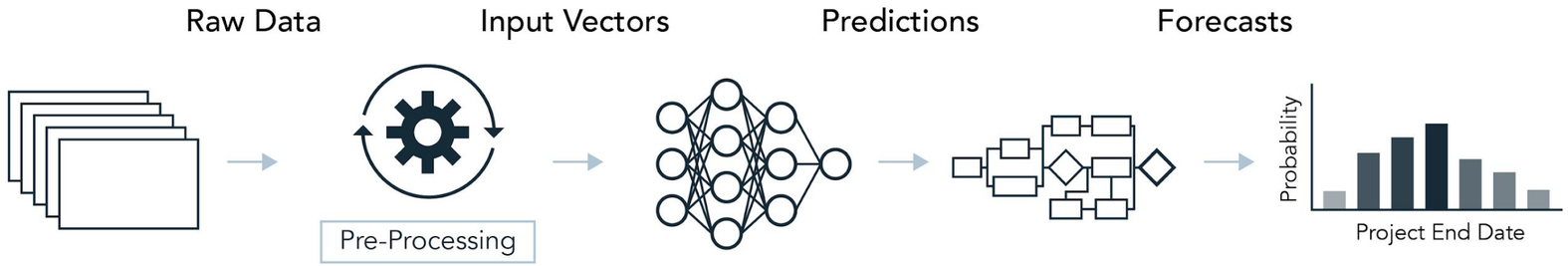
1. Overview of Calibration
2. Measuring Calibration
3. Coding Session
4. Literature Review
5. Methods for Calibrating Classifiers
6. Coding Session
7. Use Cases for Caalibration

nPlan

KDD2020

1. Overview of Calibration

- A measure of model's confidence in its predictions
- We want to
  - use the model as part of a larger system
  - the model's results as true probabilities

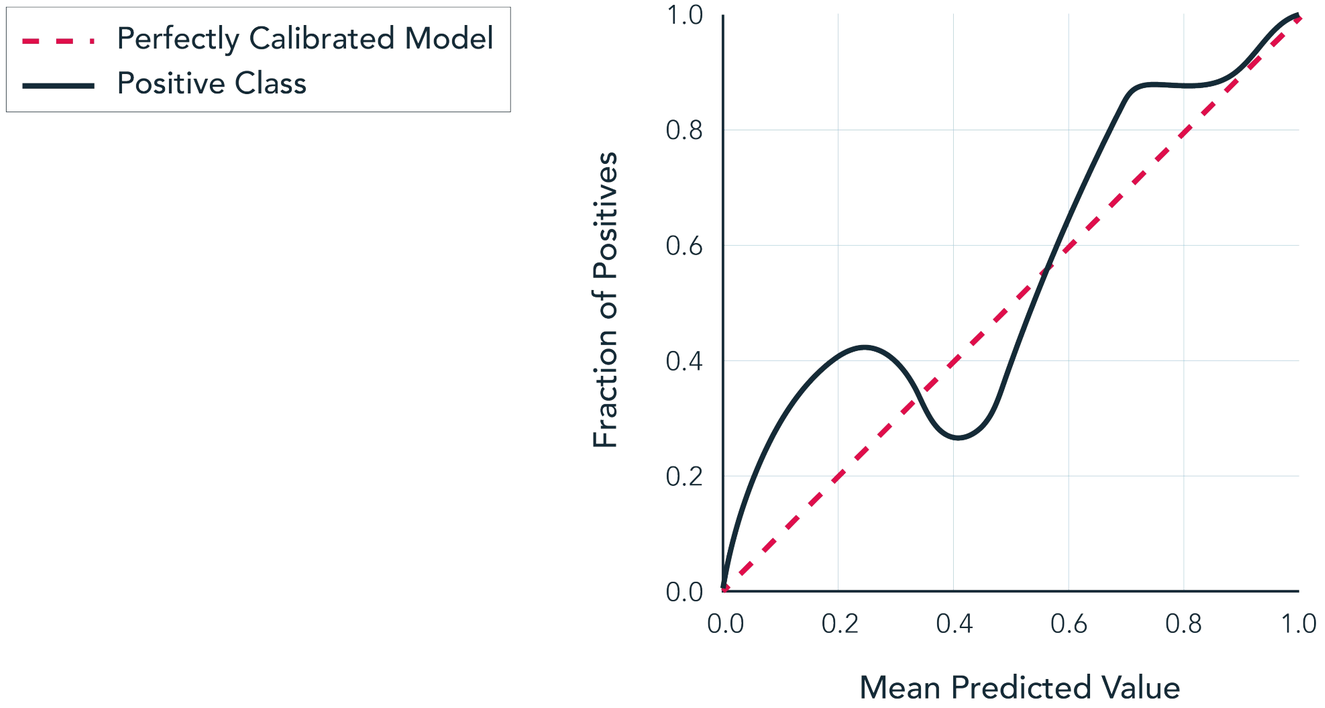


nPlan

KDD2020

2.1 Reliability Diagrams

- Visual comparison between true and predicted probabilities

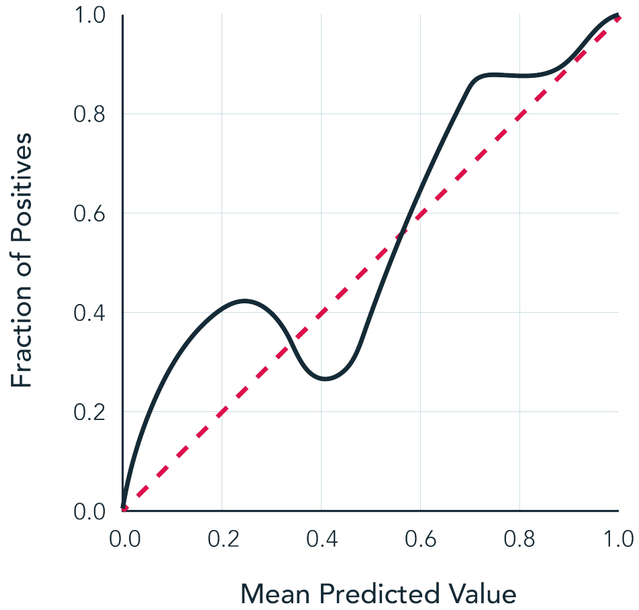


nPlan

KDD2020

2.2 Expected Calibration Error (ECE)

- Average difference between the accuracy and the confidence of a model for each bucket

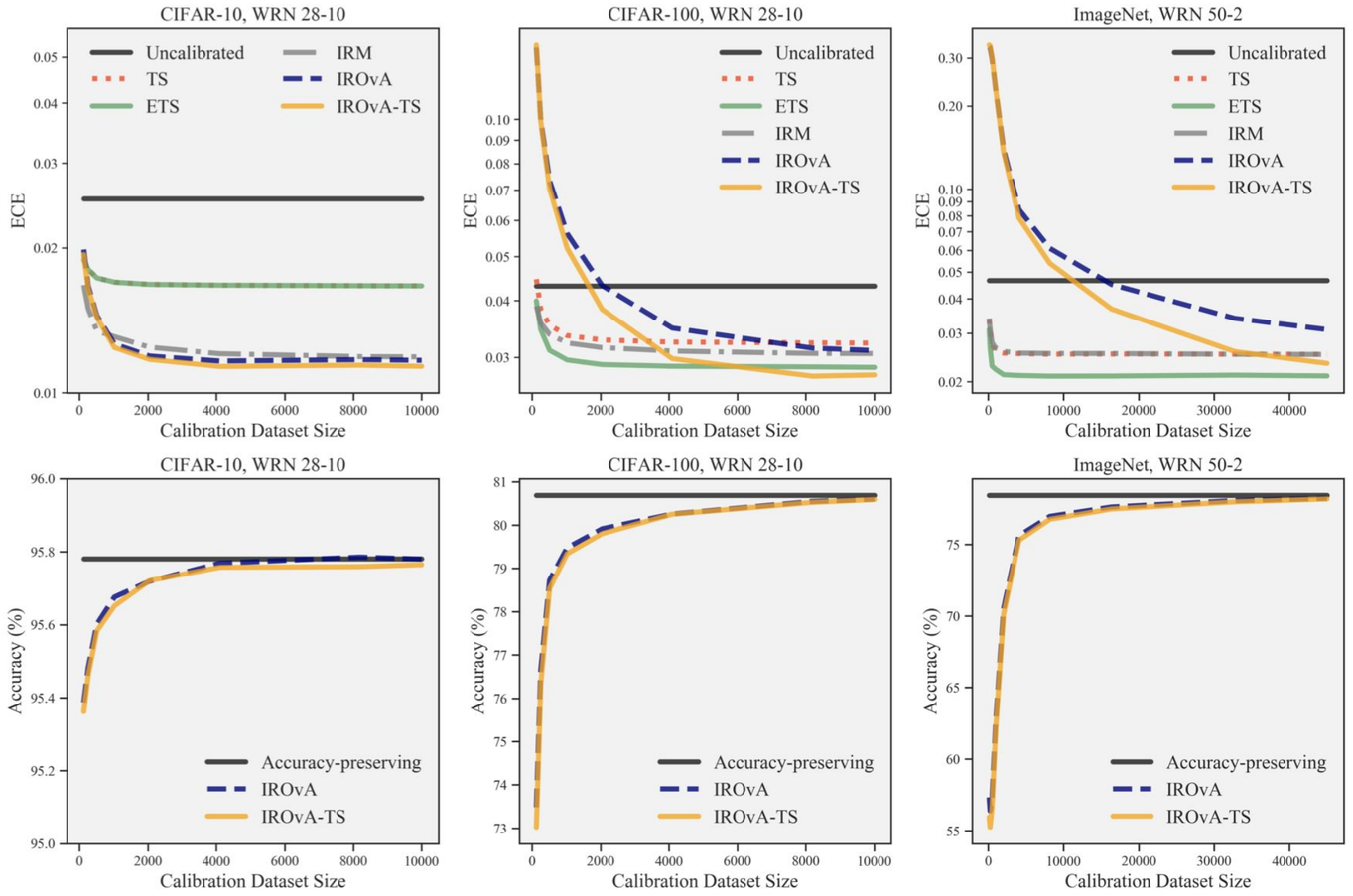


$$ECE = \sum_{m=1}^{M} \frac{|B_m|}{M} |acc(B_m) - conf(B_m)|$$

nPlan

KDD2020

2.2 Expected Calibration Error (ECE)

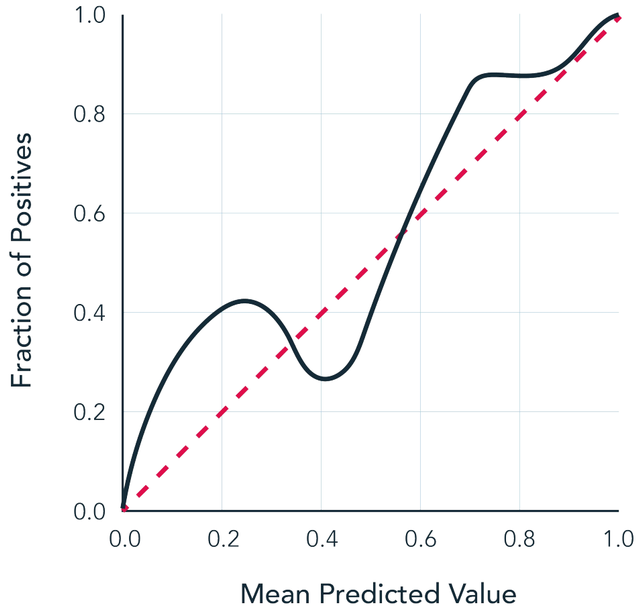


nPlan

KDD2020

2.3 Maximum Calibration Error (MCE)

- Maximum difference between the accuracy and the confidence of a model for each bucket
- Useful when trying to minimise the impact of the worst case scenario



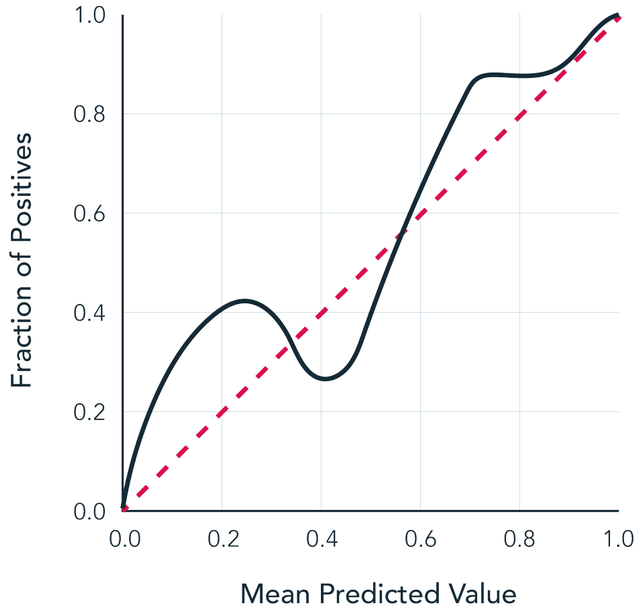
$$MCE = max_{m \in \{1,\ldots,M\}} |acc(B_m) - conf(B_m)|$$

nPlan

KDD2020

2.4 Root Mean Squared Calibration Error (RMSCE)

- Average difference between the accuracy and the confidence of a model for each bucket weighted by the support for each class



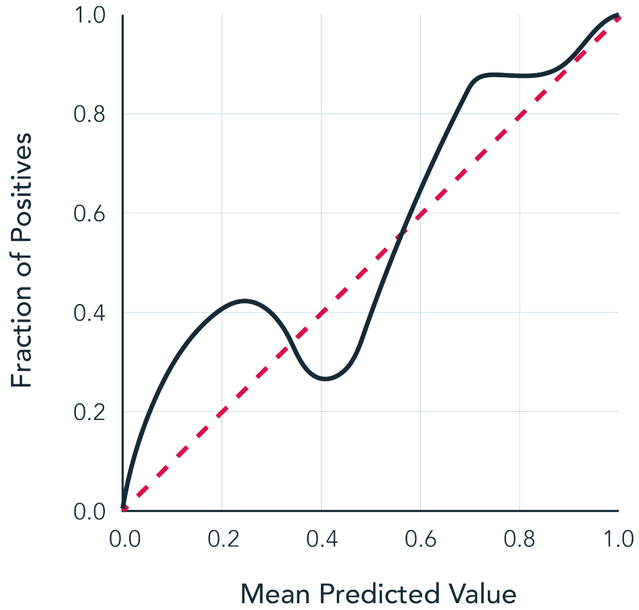
$$RMSCE = \sqrt{\frac{1}{N}\sum_{m=1}^{M} \mid B_m \mid (acc(B_m) - conf(B_m))^2}$$

nPlan

KDD2020

2.5 Multiclass Case

- Each of the metrics can be extended to the multiclass case
- One paper (Guo 2017) suggests only considering the argmax of your classifier - useful in e.g. an image classification setting
- Each class can be considered separately
- All classes can be considered together



nPlan

KDD2020

3. Coding Session

Switch to your jupyter notebooks for the coding session

nPlan

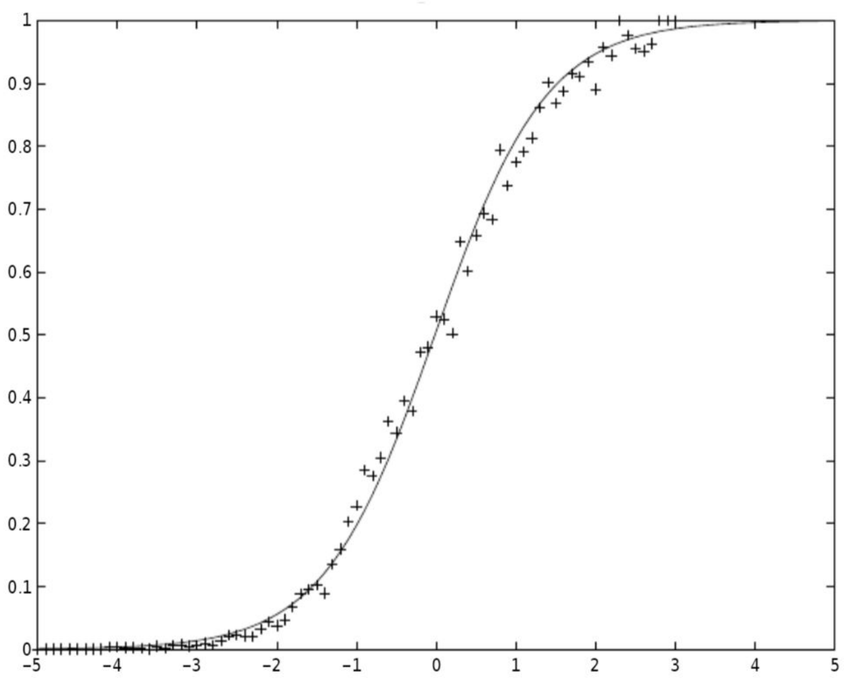
KDD2020

4. Brief Literature Review

- Focussed on several relevant papers
  - Platt (1999)
  - Caruana (2005)
  - Guo (2018)
- More that are not covered here are in the resources section

4.1 Platt (1999)

- Seminal paper in which scaling for calibration is discussed
- Seeking to translate SVM outputs (-inf, inf) to posterior probabilities
- Fitted a sigmoid to the outputs in posterior probability space



nPlan

KDD2020

4.2 Niculescu-Mizil (2005)

- A comparison of different models and calibration methods - including platt scaling and isotonic regression
- Conclusion made that calibrating neural nets of the time was not as important as with other models

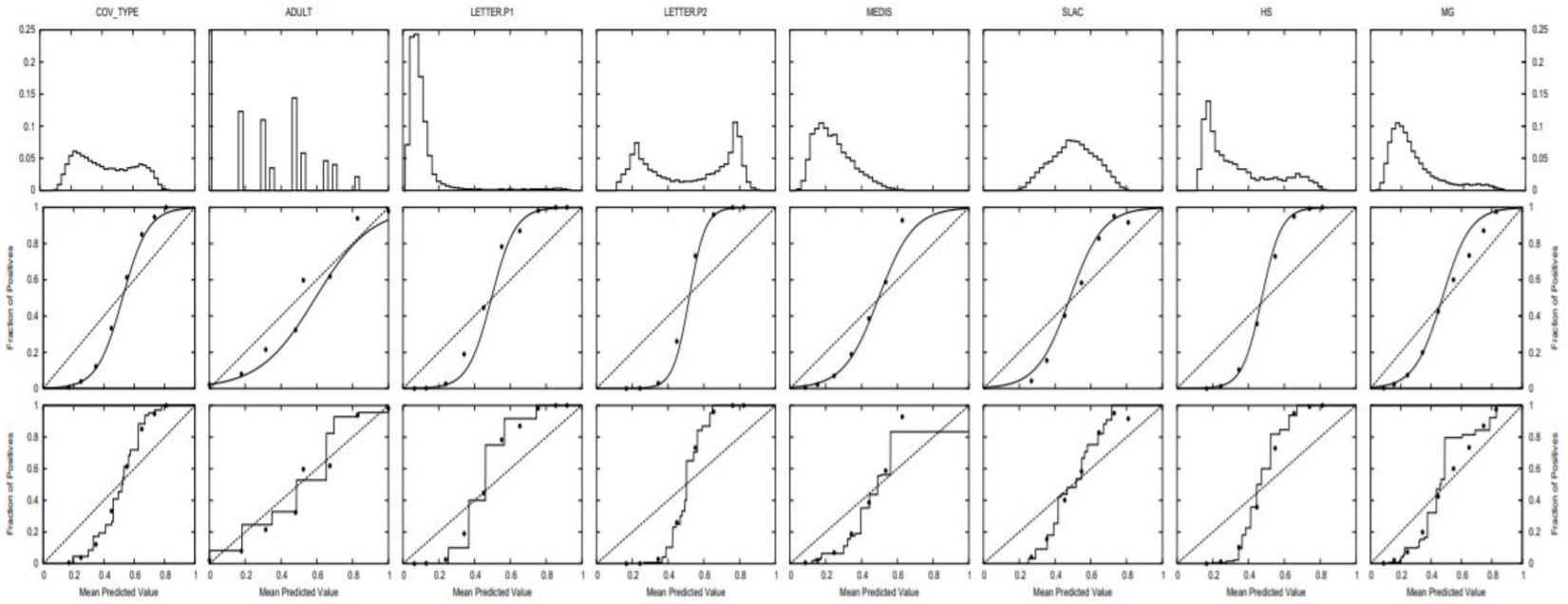
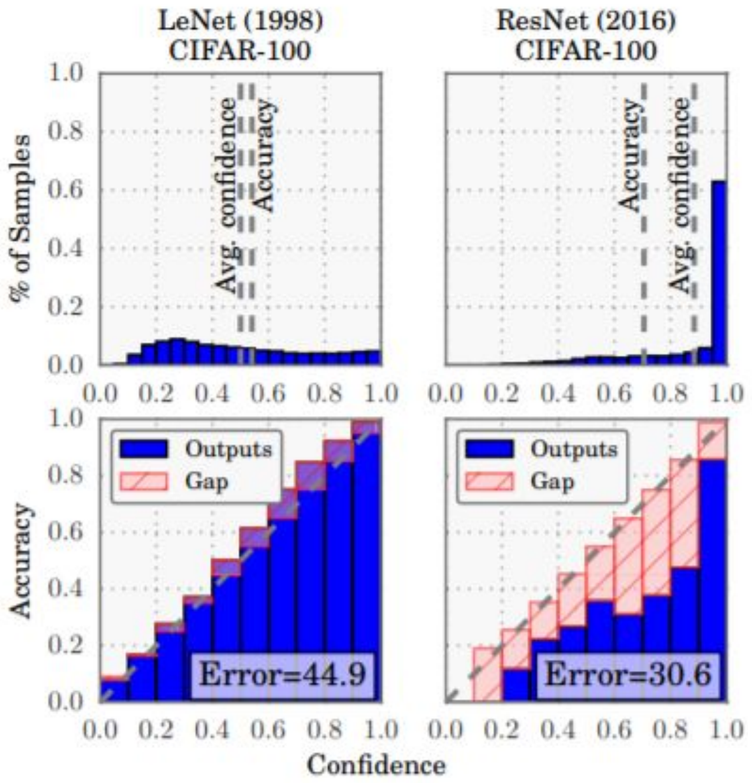
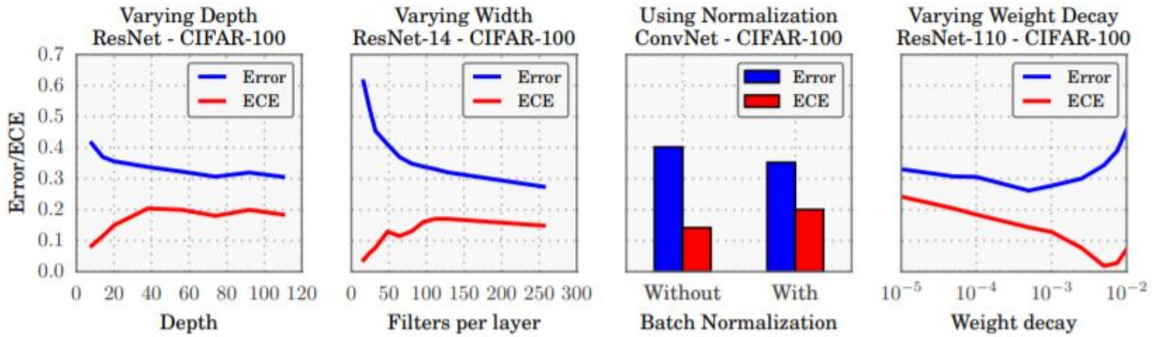


Figure 1. Histograms of predicted values and reliability diagrams for boosted decision trees.

4.3 Guo (2017)



- Considers only the more complex NN models and training methods of recent times
- Shows over-confidence in their predictions, and suggests calibration methods:
  - Multidimensional Platt scaling
  - Isotonic regression



nPlan

5. Methods for Calibrating Classifiers

- After-training post-processing calibration methods
- Requirements
  - Validation set
- Assumptions
  - Train, test and validation sets are drawn from the same distribution
- Wins
  - Simple implementation
  - No need to train a new model

nPlan

KDD2020

5.1 Isotonic Regression

- Non-parametric method for calibration of binary classifiers
- Can be extended to multiclass problem using the one-vs-all approach
- Learns a piecewise function that minimizes square loss

$$\hat{q}_i = \sum_{i=1}^{n} (f((\hat{p}_i) - y_i)^2$$

nPlan

KDD2020

5.2 Platt Scaling

- Predictions of the classifier are used as an input into a logistic regression model

$$z - logit,\ or\ the\ pre\text{-}activation\ values\ of\ the\ last\ layer$$
$$\hat{p}_i = \sigma(z_i)$$
$$\hat{q}_i = \sigma(az_i + b),\ a, b \in \mathbb{R}$$

5.3 Temperature Scaling

- Extension of Platt scaling for multi-class models
- Single parameter T > 0 for all classes
- Doesn't change the maximum of softmax
    - Prediction class and accuracy stay the same

$$\hat{q}_i = max_k \sigma_{SM}(z_i/T)^{(k)}$$

5.4 Matrix and Vector Scaling

- Extension of Platt scaling for multi-class models
- Matrix scaling
    - Linear transformation is applied where the two parameters are optimised

$$\hat{q}_i = max_k \sigma(\boldsymbol{W}\boldsymbol{z}_i + \boldsymbol{b})^{(k)}$$

$$\hat{y}_i = argmax_k \sigma(\boldsymbol{W}\boldsymbol{z}_i + \boldsymbol{b})^{(k)}$$

- Vector scaling
    - Restricts the parameters of matrix to the diagonal axis

nPlan

KDD2020

1
2
3
4
5
6

6. Coding Session

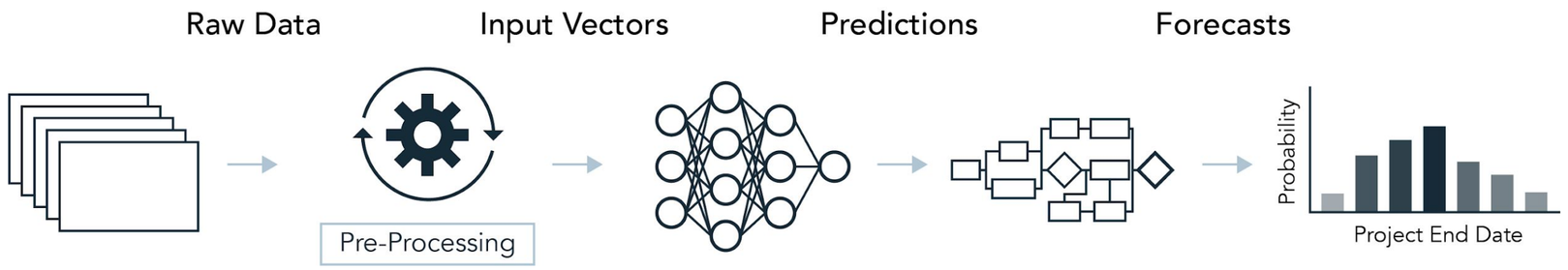
Switch to your jupyter notebooks for the coding session

nPlan

KDD2020

7. Use Cases for Calibration

- Weather forecasting
- Medical diagnostics
- More complex systems where neural net is just part of a bigger machine
- When you want to use your outputs as probabilities



nPlan

KDD2020

1
2
3
4
5
6
7

Resources

- Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. - On calibration of modern neural networks
- Platt, J. - Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods
- Zhang J., Kailkhura B., Yong-Jin Han T. - *Mix-n-Match*: Ensemble and Compositional Methods for Uncertainty Calibration in Deep Learning
- Zadrozny, B., & Elkan, C. - Transforming classifier scores into accurate multiclass probability estimates
- Murphy, Allan H., and Robert L. Winkler - Reliability of Subjective Probability Forecasts of Precipitation and Temperature
- Niculescu-Mizil, Alexandru & Caruana, Rich. - Predicting good probabilities with supervised learning
- Scikit-learn documentation and examples about calibration

nPlan

KDD2020