Data Mining for Business

Model Calibration

Dr. Shipra Maurya

Department of Management Studies

IIT (ISM) Dhanbad

Email: shipra@iitism.ac.in

Quick Exercise

	Model A	**Model B**
Accuracy	80%	80%
Confidence in each prediction	0.99	0.82

Which model is better?

- Model B is better as Model B considers itself precise 82% of the time and that's almost the case
- Model A is overconfident about its predictions

Model calibration helps you to understand whether your model is ready for deployment (real-world). How confident are you about the output of your model.

What is Model Calibration?

- A process in which we apply a post-processing operation on an already trained model in order **to improve the model prediction**

- A model is considered to be perfectly calibrated if its predicted values are in the order of the actual values or close to the actual values depending on the nature of the business problem

- Both classification and regression models can be calibrated

- Calibration is useful when the business problem is such in which predicting the accurate actual value (probability) is significantly important. For example, computing the probability of a person being heart patient

Perfectly Calibrated Classifier

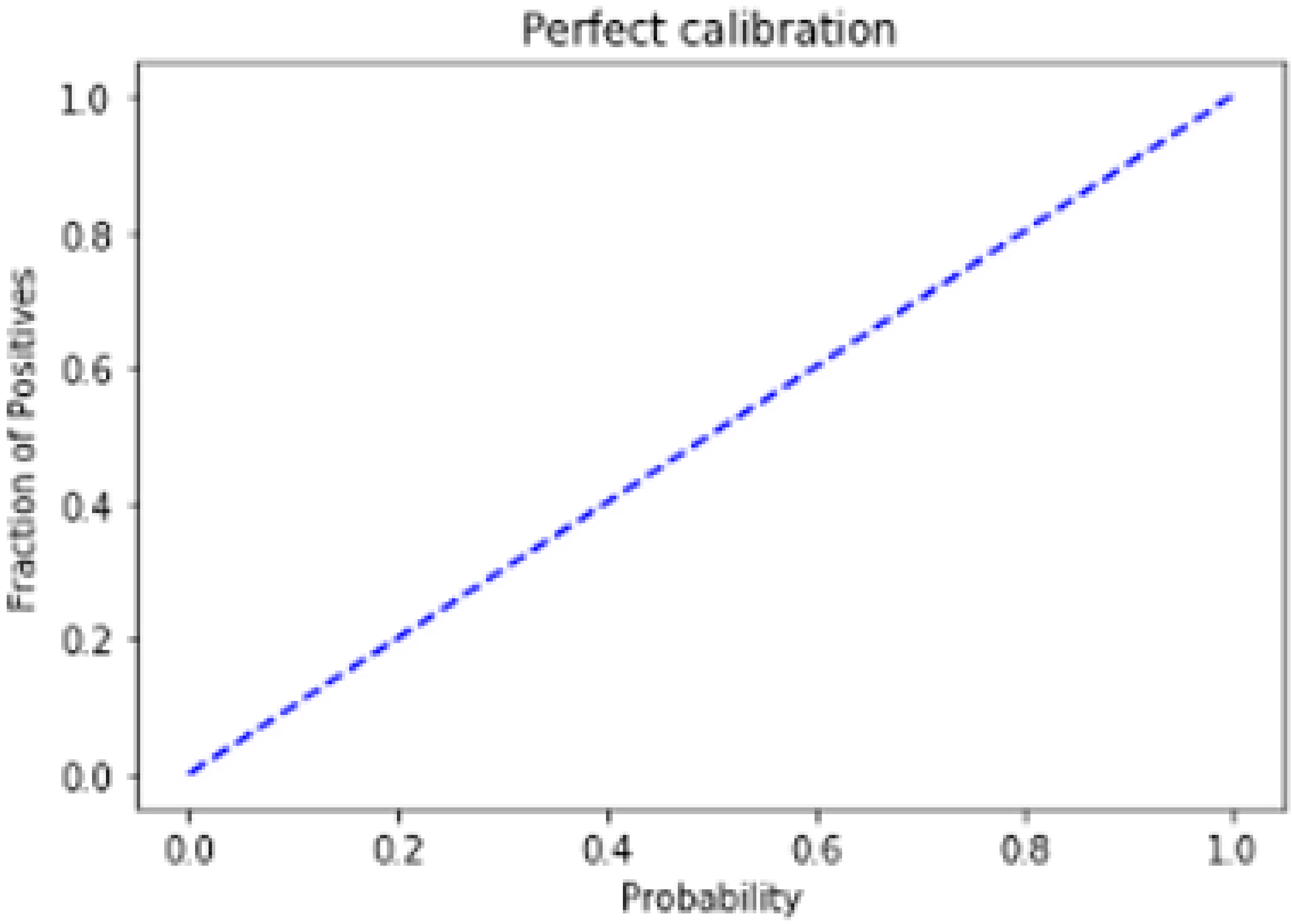


Image source - Internet

How to check if the Model is calibrated?

1. Calibration or Reliability Plot
2. Brier Loss

```
from sklearn.calibration import calibration_curve
from sklearn.metrics import brier_score_loss
```

Calibration or Reliability Plot

- A line plot of the actuals and predicted values processed using below steps:
  - **Step 1** – first take a set of data samples (preferably other than train and test set) and get the classifier predictions for it. Then take all predictions and group them into bins
  - **Step 2** – calculate mean of actual values in each bin for regression problem and fraction of actual positives (event) per bin for classification problem
  - **Step 3** – calculate confidence per bin by taking mean of probability (predicted values) in each bin
  - **Step 4** - Now plot step 2 variable (y-axis) with step 3 variable (x-axis)

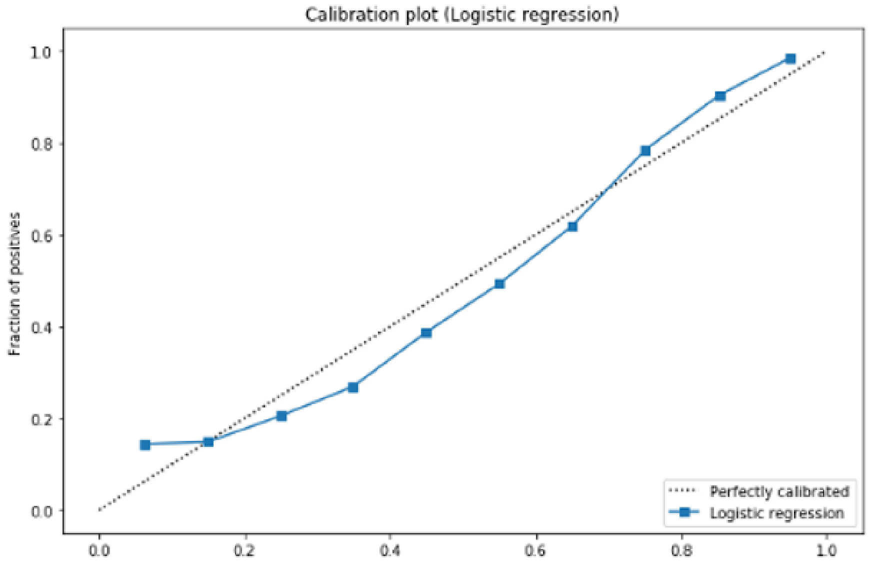


Image source - Internet

Confidence Per Bin

$$\mathrm{conf}(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \hat{p}_i.$$

Brier Loss

- Quantitative measure of calibration fit
- Lower the better

$$Brier_{loss} = \frac{1}{n} * \sum_{i=0}^{n} (y_{true}^{(i)} - y_{proba}^{(i)})^2$$

Without Class Weight

With Class Weight (wj)

$$Brier_{loss} = \frac{1}{n} * \sum_{j=0}^{1} w_j * \sum_{i=0}^{n} (y_{true}^{(i)} - y_{proba}^{(i)})^2$$

- Python package - from sklearn.metrics import brier_score_loss

Image source - Internet

How to calibrate the model?

- Isotonic Regression – does not make the sigmoid assumption for the calibration curve and hence can be applied in more number of cases. Suitable for large datasets

- Platt scaling – does make sigmoid assumption for the calibration curve. Suitable for small datasets and for specific problems only

Note – empirically proven that boosted trees, random forests and SVMs perform best after calibration and logistic regression mostly does not need calibration

Isotonic Regression (1/3)

- It is a non-parametric regression technique that does not assume linearity among variables and constant error variance

- Isotonic function is a monotonic function. It continuously increases/decreases. It tries to minimize:

minimize $\sum_i w_i(y_i - \hat{y}_i)^2$

subject to $\hat{y}_i \leq \hat{y}_j$ whenever $X_i \leq X_j$,

where the weights w_i are strictly positive, and both `X` and `y` are arbitrary real quantities.

- Does not make sigmoid assumption for calibration curve

Isotonic Regression (2/3)

- It is Piecewise linear model
- The slope of the solution is always non-negative as it is a monotonic function
- In Python
  - from sklearn.isotonic import IsotonicRegression
- It suffers from overfitting problem and hence is not fit to be used for small datasets

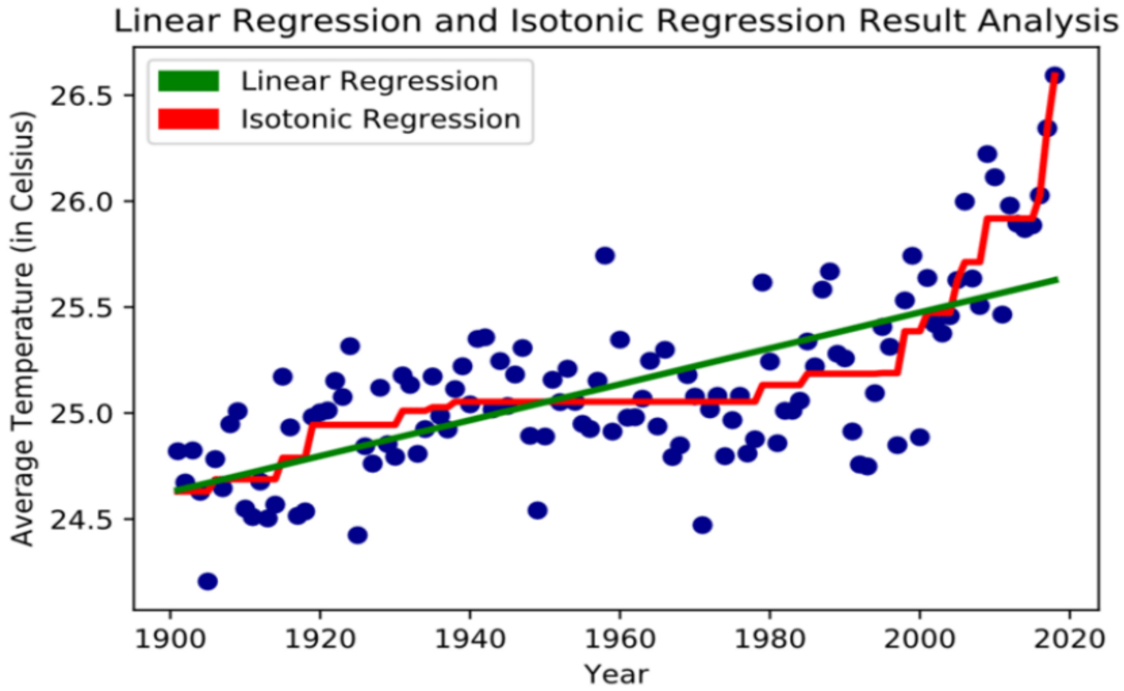


Image source - Internet

Isotonic Regression (3/3)

- **Steps:**
  1. First fit an analytical model on the training data
  2. Make predictions on the test/calibration data
  3. Develop calibration plot and check whether the model is satisfactorily calibrated or not, if not then follow below steps
  4. Take the actual target and predicted target from step 2
  5. Fit an isotonic regression model on step 4 data where y=actual target and x= predicted target

Platt Scaling

- Platt scaling is tantamount to fitting a logistic regression line to the calibration plot
- Applied on classification problems only
- Works well for small datasets
- Sigmoid assumption for the calibration curve (the only shape that the logistic regression line can fit to)

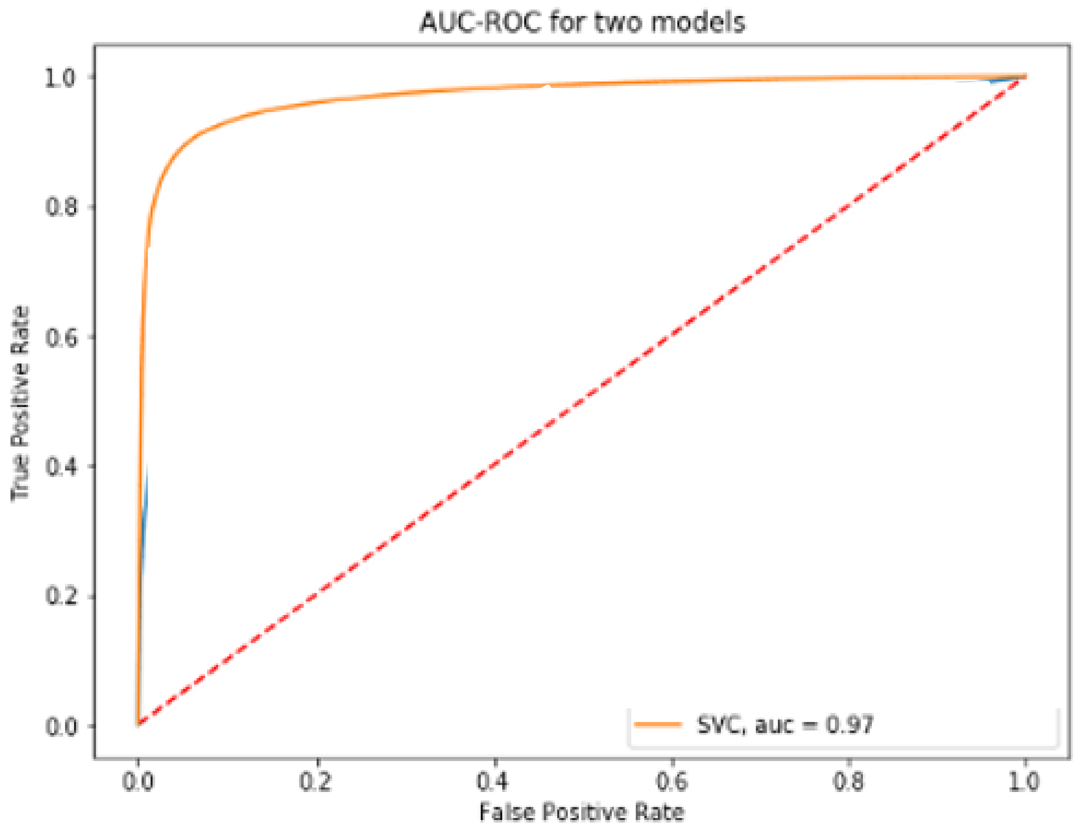
$$P_i = \frac{1}{1 + e^{-(\beta_1 + \beta_2 x_{2i} + \cdots + \beta_k x_{ki} + u_i)}}$$

Steps in Platt Scaling

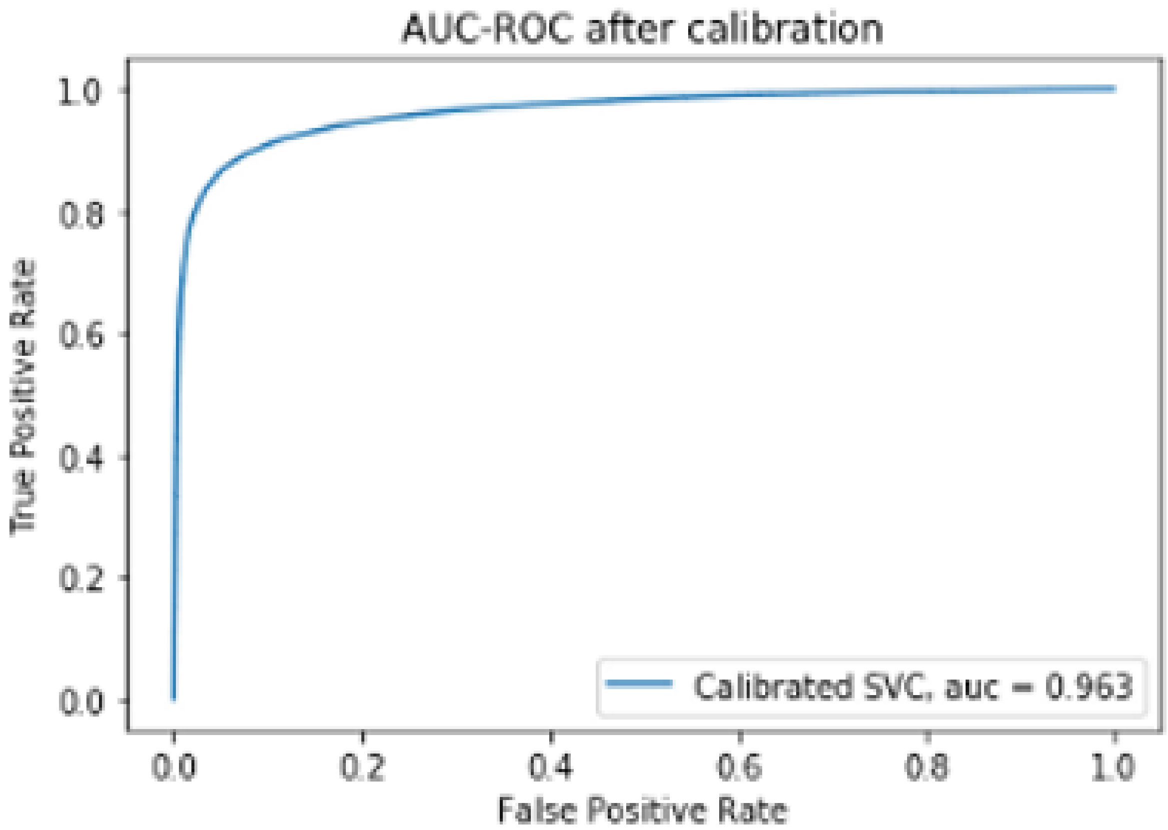
1. Split the train data set into training, test and Validation set
2. Train the model on the training data set
3. Score test data set and Validation data set; using the output of validation set, test if calibration is required or not
4. Run a logistic model on the Validation data set output using the actual target variable and the predicted values.
5. Score the test data set using the model created in step 4 with feature as the output of scoring on test data set in step 3.

Impact of Model calibration (1/2)

- The accuracy of the model may (may not) go down (up) post model calibration



Before Model Calibration

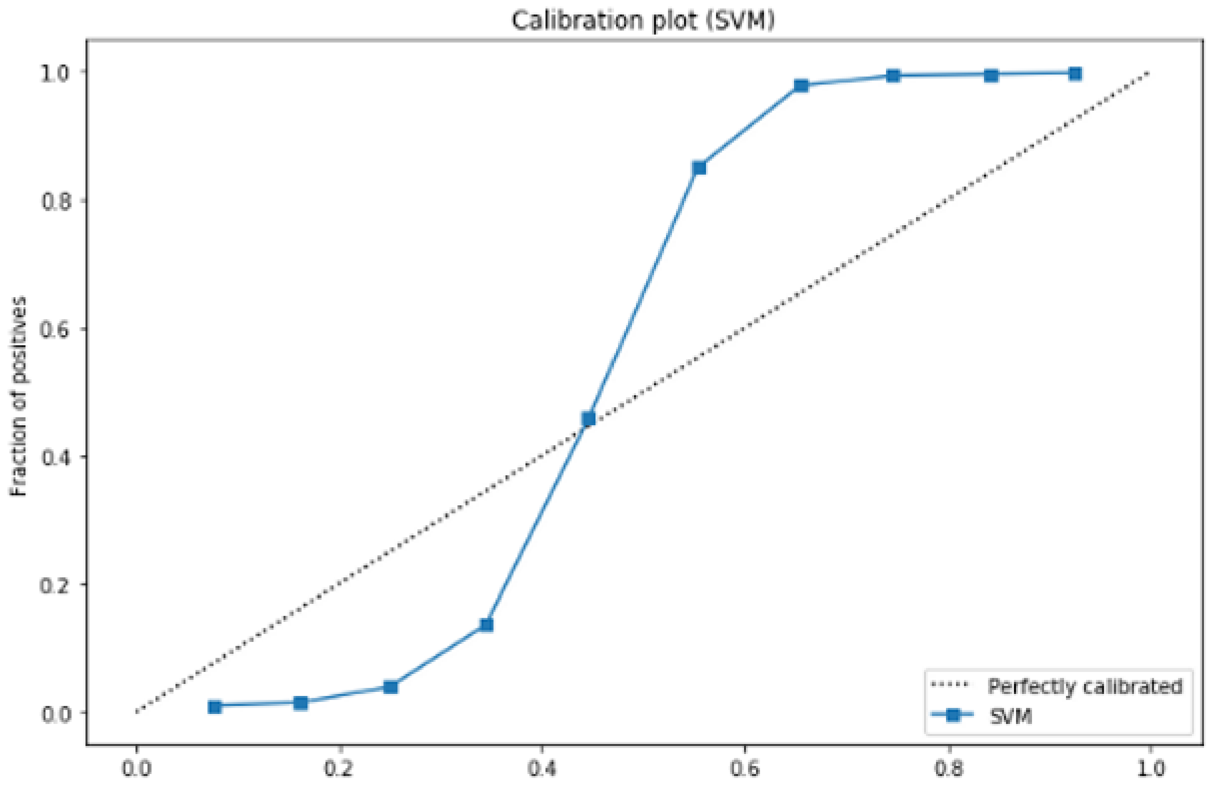


After Model Calibration

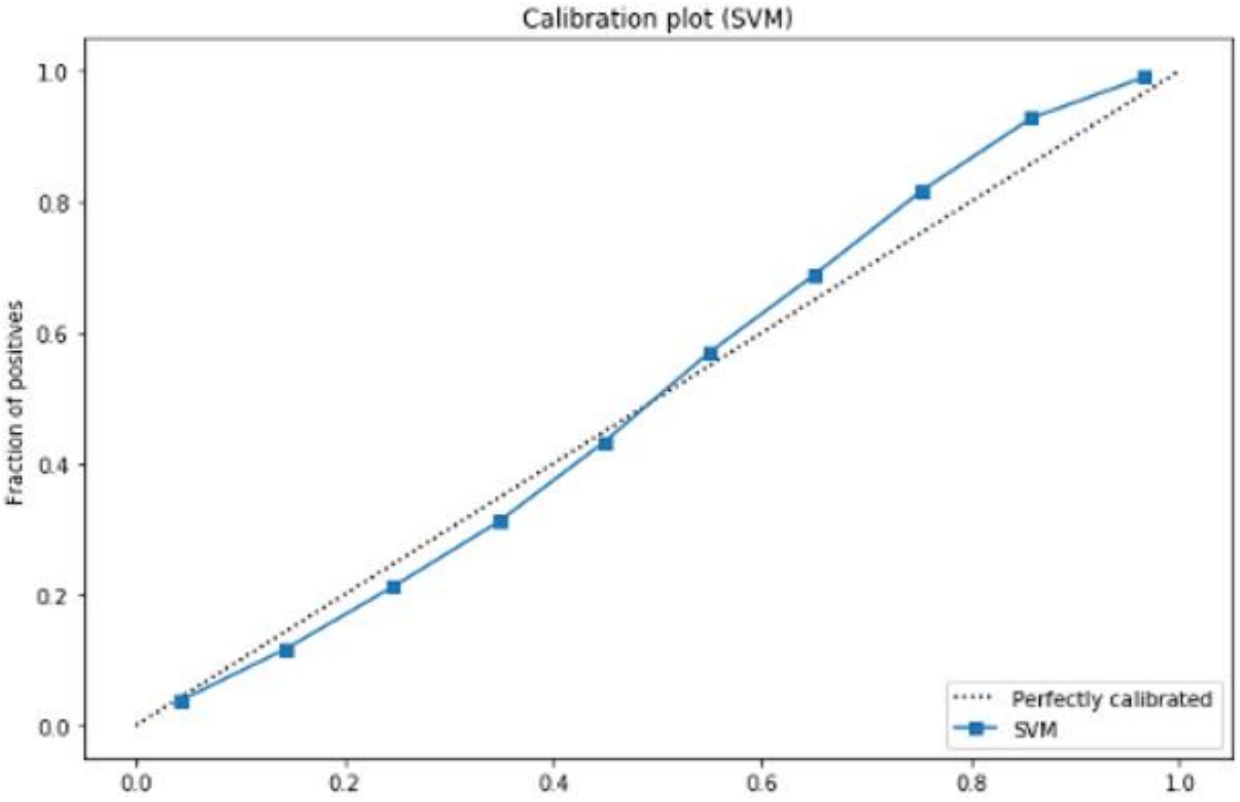
Image source - Internet

Impact of Model calibration (2/2)

- The accuracy of the calibrated model has slightly reduced



Before Model Calibration



After Model Calibration

Image source - Internet

Thank you!