Probability Threshold

Overview of Probability Threshold

- Classifiers usually provide a probability score for each prediction they yield
  - The probability score is usually between 0.5 and 1.0, inclusive
- The higher the probability score, the more certain the classifier is of its prediction
  - In a binary classification problem the probability score can be reversed
    - i.e. if a classifier is 60% confident of its prediction being class A, it is going to be 40% confident of its prediction being class B

Selecting a probability threshold

In a binary classification problem, you can define a threshold of this probability score, to be the decision point for the classification process

- Prob. threshold = 0.5 by default
- A higher probability threshold would mean less predictions of class B and more of class A

Tweaking the probability threshold can change the classification results substantially

- For many binary classification problems, it is essential to do that, in order to optimize the classifier's performance

Sensitivity analysis of a probability threshold

By tweaking the probability threshold you can perform some sensitivity analysis on the classifier

- i.e. check how stable the results are

Usually, it is very important to have not just high accuracy but also **stable accuracy**, for a given problem

- There are other ways to perform sensitivity analysis, but for binary classification problems, this is a quite common one

ROC Curve

ROC CURVE

A way of evaluating the performance of a classifier, for different threshold possibilities using a chart (i.e. an intuitive way to do sensitivity analysis for a binary classifier)

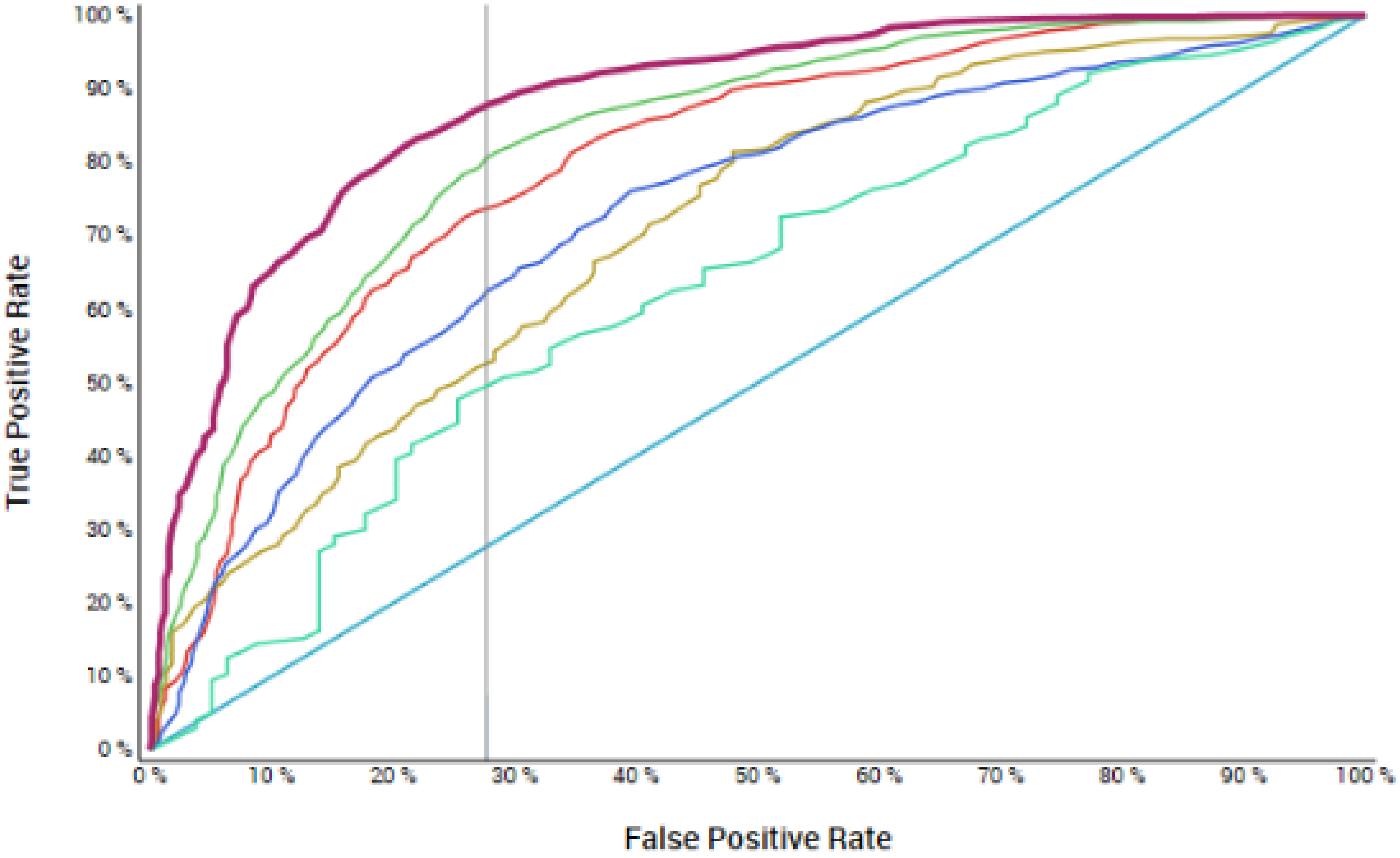
- ROC = Receiver Operating Characteristic

Depicts the relationship between TP rate and FP rate

- Usually takes the form of a zig-zag line, when plotted (ROC curve)

Often there is a straight line between (0, 0) and (1, 1), denoting the performance of a random classifier

Example of several ROC curves



Generating an ROC curve for a binary classifier

1. Calculate the CM of the classifier for a given probability threshold
2. Calculate the TP and the FP rates
3. Plot TP rate and FP rate on a chart
4. Repeat steps 1-3 for various probability thresholds
5. Connect the plotted points to generate a curve

Evaluating ROC curves

Intuitively, the curve that is closer to the top-left corner is the best

> (mnemonic rule: most interesting route from LA to NYC, via Seattle)

– Analytically, the curve with the largest area under it is the best

Area Under Curve (AUC) is a common metric related to ROC analysis

– AUC takes values between 0 and 1. Anything from 0.5 and less is considered bad. Typical values range between 0.7 and 0.9

Comparing classifier performance using an ROC curve

A classifier that is "higher" on the ROC chart than another classifier, is generally better

Two classifiers may be both better than one another for different parts of the curve spectrum

Comparing the AUC values of the two classifiers gives a general view of which one is better **overall**

Some considerations for the ROC curve – 1

The ROC curve can help us pick a probability threshold that makes sense to us

- Which probability threshold is optimal depends on our own view of the problem
- In general, we opt for the threshold that is based on the point in an ROC curve that is closest to the top-left corner

Some considerations for the ROC curve – 2

The ROC curve is the industry standard for Classification Accuracy. A confusion matrix is only valuable when you can specify and justify its probability threshold.

It's quite useful to include an ROC chart in your report for a data science project

Python functions for Accuracy Measures

sklearn.metrics class in sklearn package

Confusion Matrix and friends

- CM: *confusion_matrix* function
- Separate TP, TN, etc. from CM: *ravel* function
- accuracy rate: *accuracy_score* function
- error rate: 1 - accuracy_score
- precision: *precision_score* function
- recall: *recall_score* function
- f1 score: *f1_score* function

ROC curve family

- Actual ROC curve: *roc_curve* function
- AUC: *roc_auc_score* and *auc* functions

Summary

- The comparison of the probability threshold with the probability score determines the predicted class.
- ROC curve may help determine an appropriate probability threshold.
  - Better curve toward the top left
  - Use Area Under Curve to compare classifiers