Data Mining & Machine Learning

Yong Zheng

Illinois Institute of Technology
Chicago, IL, 60616, USA

ILLINOIS TECH
College of Computing

HW1

- Smoothing By Bin Means
  - Students created bins only without replacing values by using mean value in each bin
- Min-max normalization
  - Students normalized the scale to [0, 1] or [0,5], rather than [1, 5]

HW1

- Nominal variable to binary variables
  - If there are N values in the nominal variable, we only need N-1 new binary variables. Students used N columns
  - Wrong transformation for “Genre”. Some students used 0, 1, 2, 3, 4… to encode it

HW1

- Python for data preprocessing
  - Fill in missing values without checking whether a variable has missing value or not
  - Correlation among variables
    - Do not know which method to be used
    - Be able to run coding, like ANOVA, but do not know how to interpret it
  - Normalize columns to [1, 5], where students forgot the normalization applied to binary columns

Schedule

- Classification Evaluation Metrics
- Naïve Bayes by Python

Schedule

- Classification Evaluation Metrics
- Naïve Bayes by Python

Classification Evaluation Metrics

- Accuracy is not the only metric
- Take binary classification for example Confusion Matrix

Predicted Labels

Actual Labels	+ (Yes)	- (No)
+ (Yes)	**True Positives (TP)**	**False Negatives (FN)**
- (No)	**False Positives (FP)**	**True Negatives (TN)**

Accuracy = (TP + TN)/All

Error rate = (FP + FN)/All

They are just overall metrics

It is possible that a model works well on overall, but very bad on a single label

Overall Acc = 90%, Acc on Positive label = 40%

Classification Evaluation Metrics

$$precision = \frac{TP}{TP + FP}$$

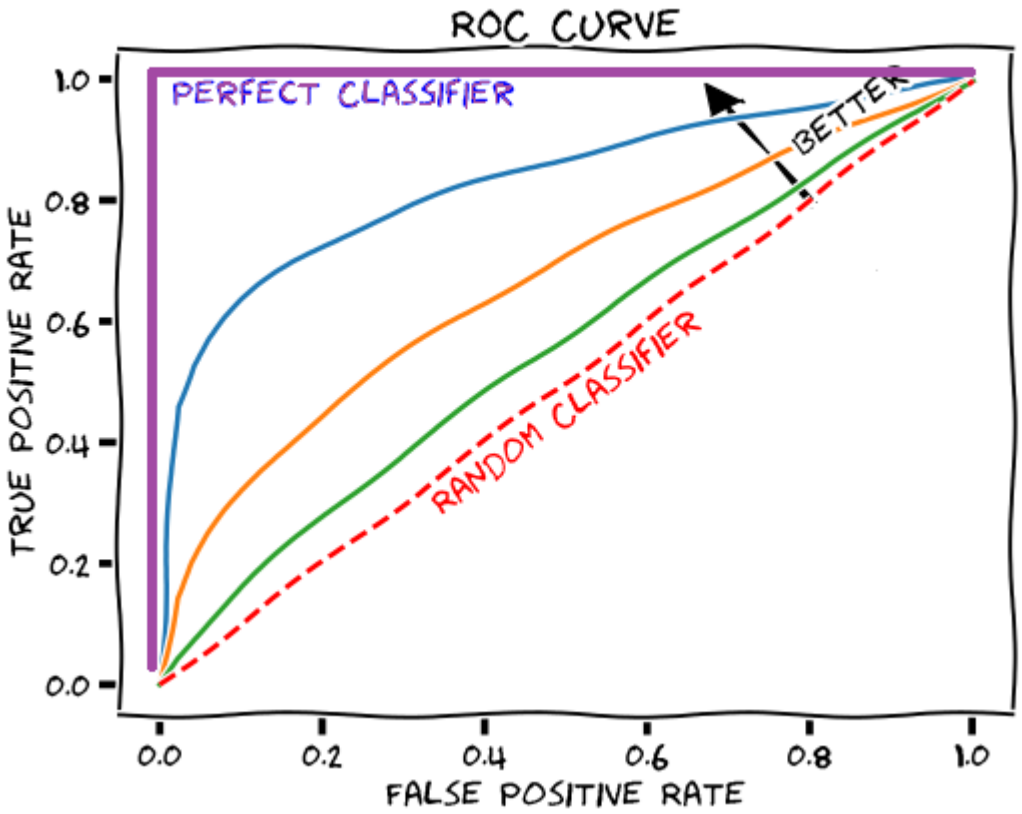
$$recall = \frac{TP}{TP + FN}$$

$$F = \frac{2 \times precision \times recall}{precision + recall}$$

Specificity = TN/(FP+TN)

Classification Evaluation Metrics

- ROC Curve: false positive vs true positive rate
  false positive rate = 1 - specificity



You can observe the area under the curve. If this area is larger, a model is better.

Classification Evaluation Metrics

- In multi-class and multi-label classification, precision and recall can be calculated by two ways: Macro Average & Micro Average
- In Binary Classification

$$precision = \frac{TP}{TP + FP}$$

- Macro Precision

$$\frac{Pre_A + Pre_B + Pre_C}{3}$$

- Micro Precision

$$\frac{TP_A + TP_B + TP_C}{TP_A + TP_B + TP_C + FP_A + FP_B + FP_C}$$

Assume that we have 3 values in label

Classification Evaluation Metrics

- The metrics based on Micro Average can better represent the performance when labels are imbalanced
- In Binary Classification

$$recall = \frac{TP}{TP + FN}$$

- Macro Recall

$$\frac{Recall_A + Recall_B + Recall_C}{3}$$

- Micro Recall

$$\frac{TP_A + TP_B + TP_C}{TP_A + TP_B + TP_C + FN_A + FN_B + FN_C}$$

Assume that we have 3 values in label

Schedule

- Classification Evaluation Metrics
- Naïve Bayes by Python

Naïve Bayes in Python

- *Categorical Naive Bayes*
- Bernoulli Naive Bayes
- Gaussian Naive Bayes
- Multinomial Naive Bayes
- Complement Naive Bayes

https://scikit-learn.org/stable/modules/naive_bayes.html

Next Week

- I will attend two conferences next week
- Course materials and videos have been recorded and uploaded to Blackboard system
  - Decision Trees
  - Python for Decision Trees
  - Materials will be available on next Tuesday
    - Videos were uploaded to Google Drive
    - You need to login Google Drive by using your hawk email, in order to get access to these videos