2016 Fall CPS 571/STA 561: Homework 2

Duke University

September 8, 2016

1 ROC and AUC

Given the dataset `data.csv` with feature vector $\mathbf{x}$ including age, course and likeStats, and the binary label (or class) y. Consider the linear classifier $g(\mathbf{x}) = \boldsymbol{\theta}^\top \mathbf{x} + \theta_0$. Consider also a non-linear classifier defined as $f(\mathbf{x}) = \sigma(\boldsymbol{\theta}^\top \mathbf{x} + \theta_0)$, where $\sigma(z) = (1+e^{-z})^{-1}$ is a "sigmoid" function that forces the output to be between 0 and 1. You are not expected to estimate the model parameters $\boldsymbol{\theta}$ and θ_0, instead you are given the values $\boldsymbol{\theta} = (0.05, -3, 2.5)$ and $\theta_0 = 0.3$.

(a) First calculate the value of $g(\mathbf{x})$ for each data point. What choices of threshold would minimize misclassification error on the training set? Compute the confusion matrix for these thresholds.

(b) Calculate the value $f(\mathbf{x})$ for each data point. Assuming you want to minimize misclassification error, propose an appropriate threshold for this classifier and state your reason. Compute the confusion matrix, precision, recall, and F1 score for the threshold.

(c) For classifier $f(\mathbf{x})$, plot the ROC curve, and compute the AUC. (Please do not use package built-in functions, such as sklearn.metrics. You are expected to write functions that compute the (fpr, tpr) pairs for ROC plotting and AUC calculation.) An easy way to do this might be to use sorting functions. It is ok if the ROC curve is a scatter plot, you do not need to connect the points.

(d) Given any monotonic function $h(z) : R \to R$, what is the relationship between (i) the ROC curve of any function $f(\mathbf{x})$ on data $\{(\mathbf{x}_i, y_i)\}_i$ and (ii) the ROC curve $h(f(\mathbf{x}))$ on the same data?

2 Decision Tree

In class, we used the information gain (overall reduction in entropy) as the criterion for choosing which feature to split on. Another option would be to use the Gini index.

Remember that the splitting criteria is a measure of how "impure" the node is. If the nodes are "pure," it means that the fraction of positives in the node is either close to 1 or close to 0. (Either almost all the observations are positive or they are almost all negative.) If there are 2 classes, the Gini index for $(p,\ 1-p)$ is the following, where p is the fraction of positives in the node and $1-p$ is the fraction of negatives:

$$\text{Gini index}(p, 1-p) = 1 - p^2 - (1-p)^2 = 2p(1-p).$$

You can see that if p is close to either 0 or 1 the Gini index will be very low. We will choose to split the node with the best reduction in Gini index, averaged across the leaves (children) of the possible split. Let us denote N as the number of observations in the node we are considering to split. Define p to be the fraction of positives in the node we are considering to split. Denote p_c as the fraction of positives in the c^{th} branch of the potential split, and $1-p_c$ as the fraction of negatives in the c^{th} branch of the potential split. Denote N_c as the number of observations falling into the c^{th} branch of the potential split. Then:

$$\text{Gini Reduction} = \text{Gini index}(p, 1-p) - \sum_{\text{children c}} \frac{N_c}{N} \text{ Gini index}(p_c, 1-p_c).$$

(a) Choose the best feature to split on first in the made-up dataset below. Show all work leading to your answer; you will not get credit for simply reporting the best feature and its induced decrease in Gini index, you need to show the reduction in Gini index for all features as well.

X_1	X_2	X_3	Y
1	0	0	0
1	1	0	0
1	0	1	0
0	1	1	1
0	0	0	0
0	0	1	1
1	1	1	0
0	1	0	1
0	1	1	1
0	0	1	1

(b) If you had split according to information gain, would you get the same result?

2.1 Decision Trees

One great thing about decision trees is that they can represent logical functions. How would you represent the following as a decision tree? You can choose how each node in the decision tree votes (positive equates to yes and negative equates to no).

- A AND $\sim B$
- A OR (B AND C)
- A XOR B
- (A AND B) OR (C AND D)

2.2 Decision Trees and Random Forests

- Create a probability distribution such that if you randomly draw a training set of size n_δ from it, then with probability at least $1 - \delta$, decision trees perform worse than random forests. Assume the test set is a very large sample drawn from the same distribution as the training set.

  In order to do this, you may need to specify a parameter value for at least one algorithm. You will need to specify n_δ as a function of δ, and the result should be true for any δ.

  Hint 1: Choose a distribution that is almost completely boring.
  Hint 2: Everything can be done in one dimension. Conceptually it is easier if you allow splits at any integer.

  Note that we will give almost full credit for this problem if you get the idea right, even if you don't quite get all the calculations to work out.

- Create a probability distribution such that if you randomly draw a training set from it, then with high probability, decision trees will perform better than random forests. Describe how the advantage of decision trees over random forests will change as n grows.

  Hint 1: This problem is not as difficult as you might think.
  Hint 2: Everything can be done in one dimension. Conceptually it is

easier if you allow splits at any integer.

Note: You don't need to work out any calculations for this problem, you just need to get the idea right.