Northeastern University, Khoury College of Computer Science

CS 6220 Data Mining | Assignment 6

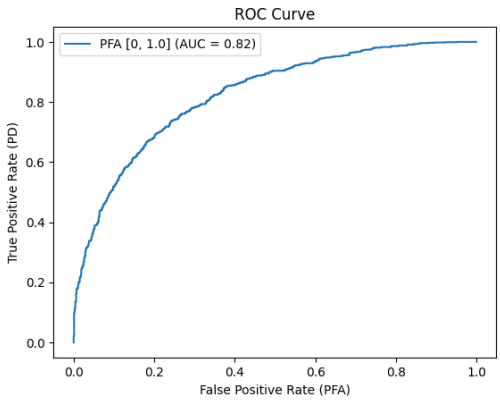
Due: November 14, 2023(100 points)

Chun Wei Tseng

https://github.com/Chun-Wei-Tseng/CS6220-Assignment-6

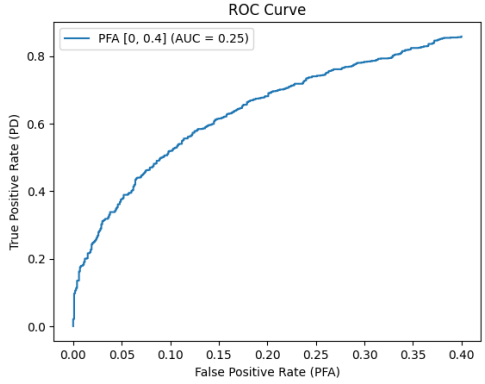
1. Plot the ROC curve and calculate the AUC for the following ranges:
   a. $P_{FA} \in [0, 1.0], the\ full\ range\ of\ the\ thresholds$

→ AUC: 0.82



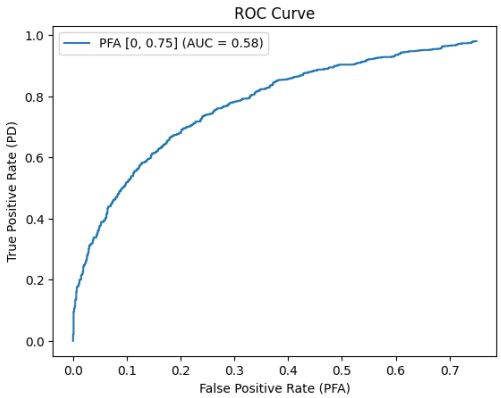
   b. $P_{FA} \in [0, 0.4]$

→ AUC: 0.25



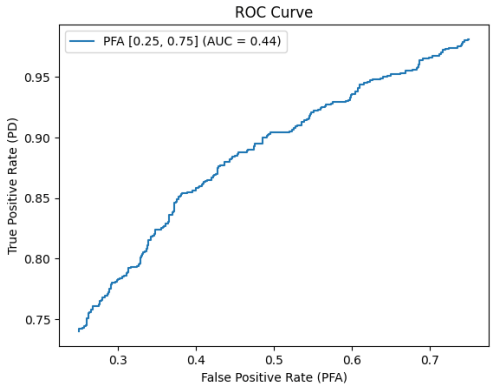
   c. $P_{FA} \in [0, 0.75]$

→ AUC: 0.58



d. $P_{FA} \in [0.25, 0.75]$

→ AUC: 0.44



2. Your implementation notes:
    a. Describe your implementation. How would you sweep your thresholds? For each threshold, how would you calculate the PFA and PD? What is the runtime in big-O notation?

→ I iterated through N elements in scores and use each of them as threshold to generate predicted labels (N elements). If elements in the scores has a value smaller than or equal to the threshold, I set the predicted value to it as 0, else 1. Then I initiate true positive (tp), false positive (fp), true negative (tn), false negative (fn) as 0s. Then I iterated through the predicted labels and compared each of them with the corresponding actual labels. After updating all the values of tp, fp, tn, fn, I used them to calculate the PFA and PD at each threshold. PFA is calculated by dividing number of false positives by total number of negative labels (false positives + true negatives). PD is calculated by dividing number of true positives by total number of positive labels (true positives + false negatives). After getting all the PFAs and PDs regarding each threshold, I then iterated through the total PFAs (N items) and selected the ones within the given range of values and their corresponding PDs. Finally, I iterated through the selected PFAs (less than or equal to N elements)and PDs and calculate the AUC accordingly. The runtime is O(N^2).

    b. Determine the runtime of your implementation in big-O.

→ In the outer loop, I iterated through N elements and use them as thresholds to generate predicted labels (N elements). For each of the threshold, I iterated

through N elements in the predicted labels and compared them to the actual labels to find total numbers of true positive, true negative, false positive, false negative. Which takes O(N^2). Constructing pfa_in_range, pd_in_range, auc_in_range all takes O(N) since I only have to iterate through N elements for a constant times, and the operations inside those loops takes constant time. Therefore, the runtime of my implementation is in O(N^2).

c. Can you make your implementation run in O(N log N)?

→ Sorting scores and labels can be done in O(N log N). Instead of using nested for loop (O(N^2)) to generate PFAs and PDs for each threshold (N thresholds in total). We can use binary search to find the maximum and minimum thresholds (starting with minimum score and maximum score as initial values), which takes O(log N). In each search, we still need to iterate through all N elements to generate predicted labels list and tp, fp, tn, fn, and calculated PFAs and PDs. Those calculations inside the loop take constant time to run. The above process has O(N log N). The time to find PFAs and corresponding PDs and calculating AUC stays the same, which is in O(N). Therefore, the implementation in this case run in O(N log N).

3. What thresholds provide a precision of 0.9?

→ `Thresholds provides a precision of 0.9: [1.106543268101517, 1.0997159612732068, 1.1059731003652715, 1.1159880060210832, 1.1066845370765228], Average thresholds: 1.10698097456752`

`Relative precisions: [0.8994413407821229, 0.8991825613079019, 0.8997214484679665, 0.9008498583569405, 0.8991596638655462], Average Precision: 0.8996709745560956`

4. At this threshold, what is the accuracy of the classifier?

→ `Accuracies of the classifier at this threshold: [0.643, 0.6465, 0.6435, 0.6415, 0.6425], which is about 0.6434`