Northeastern University, Khoury College of Computer Science

CS 6220 Data Mining — Assignment 6

Due: 28th March, 2024(100 points)

Dharun Suryaa Nagarajan
HOMEWORK GIT REPOSITORY

Homework Questions

2. Plot the ROC curve and calculate the AUC for the following ranges:

   1. $P_{FA} \in [0, 1.0]$, the full range of thresholds
      Answer: AUC= 0.824524

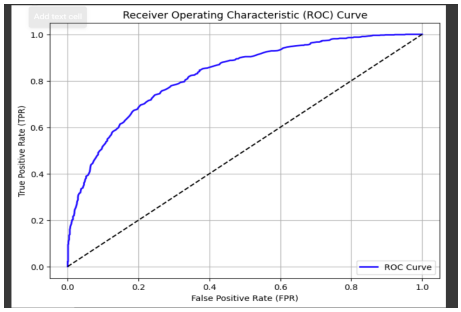

Figure 0.1: File 1

   2. $P_{FA} \in [0, 0.4]$
      Answer: AUC: 0.2523285

   3. $P_{FA} \in [0, 0.75]$
      Answer: AUC: 0.5722555

   4. $P_{FA} \in [0.25, 0.75]$
      Answer: AUC: 0.437881

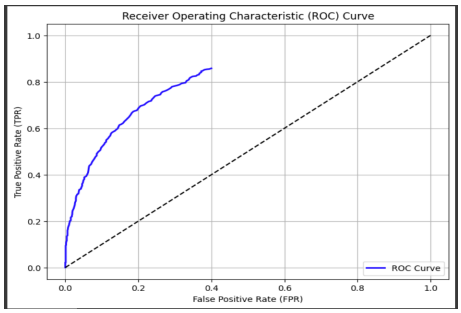

Figure 0.2: File 2

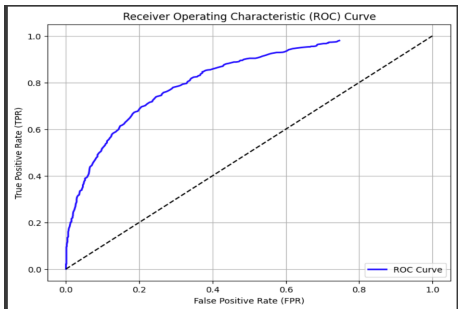

Figure 0.3: File 3

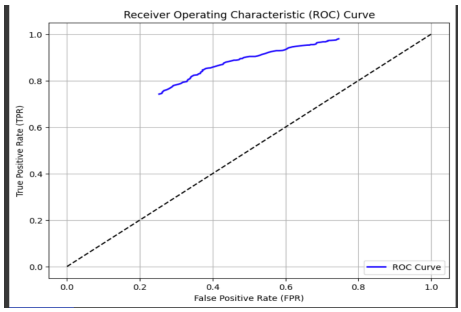

Figure 0.4: File 4

```
def roc_range(scores, labels, maxfpr, minfpr = 0):
    fpr_in_range, tpr_in_range, auc_in_range = np.array([]), np.array([]), []
    threshold = 0.01
    min_val = min(scores) - threshold
    max_value = max(scores) + threshold
    for i in np.arange(max_value,min_val,-threshold):
      TP = TN = FP = FN = 0
      for pred, actual in zip(scores, labels):
          if pred >=i and actual == 1:
              TP += 1
          elif pred < i and actual == 0:
              TN += 1
          elif pred >= i and actual == 0:
              FP += 1
          elif pred < i and actual == 1:
              FN += 1
      temp = TP/(TP+FP) if TP+FP != 0 else 1
      if temp>= 0.9 and temp<0.91:
        print(temp,i)
      tpr_in_range = np.append(tpr_in_range, TP/(TP+FN))
      fpr_in_range = np.append(fpr_in_range, FP/(FP+TN))
    indices = np.where((fpr_in_range >= minfpr) & (fpr_in_range <= maxfpr))
    # Extract the FPR, TPR, and thresholds within the specified range
    fpr_in_range = fpr_in_range[indices]
    tpr_in_range = tpr_in_range[indices]
    auc_in_range = np.trapz(tpr_in_range,fpr_in_range)
    return fpr_in_range, tpr_in_range, auc_in_range
```

Figure 0.5: Code written

3. Your implementation notes:

    1. 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?

       Implementation and threshold calculation: The roc range function is designed to compute the Receiver Operating Characteristic (ROC) curve within a specified false positive rate (FPR) range, determined by minfpr and maxfpr, using predicted scores (scores) and actual labels (labels). It initializes empty arrays for FPR, TPR, and AUC, sets a threshold for thresholding iterations, calculates the minimum and maximum threshold values based on the scores, and then iterates through thresholds to calculate true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). The function computes the true positive rate (TPR) and false positive rate (FPR) for each threshold, filters them based on the specified FPR range, and calculates the area under the curve (AUC) within that range using trapezoidal integration. The final result includes the filtered FPR, TPR, and AUC values within the specified FPR range.

    2. Determine the runtime of your implementation in $big - O$.
       Answer : $O(n^2)$

    3. Can you make your implementation run in $O(NlogN)$?
       Answer: By sorting the the scores and labels and maintaining a previous store of the variables (TP, FP, TN, FN) we can get it under nlogn cause of the sorting and O(nlogn) becomes the overall runtime.

4. What thresholds provide a *precision* of 0.9?
Answer: 1.12
5. At this threshold, what is the accuracy of the classifier?
Answer : 0.6415