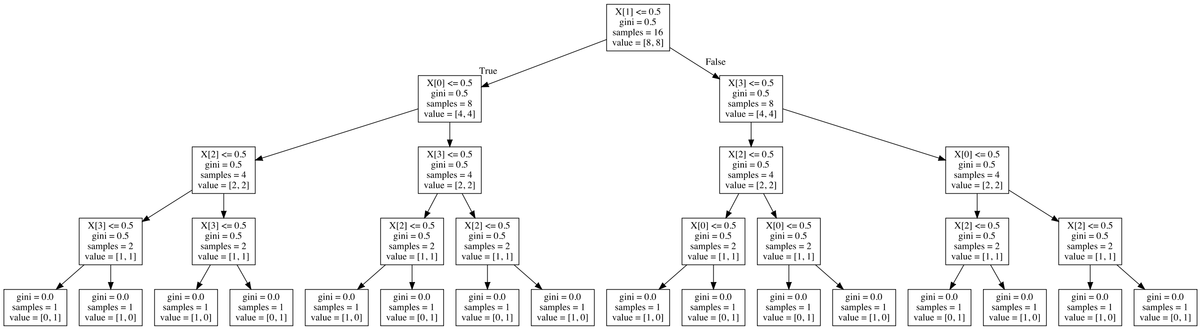
## Section 3.11 Question 1:

**Draw the full decision tree for the parity function of four Boolean attributes, A, B, C, and D. Is it possible to simplify the tree?**



The tree cannot be simplified.

## Section 3.11 Question 2:

**(a) Compute the Gini index for the overall collection of training examples.**

Class 0 has 10 counts and class 1 has 10 counts and total counts are 20.

- Gini index = 1- $(10/20)^2$- $(10/20)^2$

  = 0.5

**(b) Compute the Gini index for the Customer ID attribute.**

In this case, when we split the data by customer id, we see that there are 20 nodes/classes and each class has 0 Gini index. Therefore, the overall Gini index for customer id will be 0.

**(c) Compute the Gini index for the Gender attribute.**

In this case, we find that female with class C0 has 4 and C1 has 6 counts and male with class C0 has 6 and C1 has 4 counts.

- The Gini index for Male gender is 0.48.
- The Gini index for Female gender is 0.48.
- The overall Gini index for Gender is 0.48.

**(d) Compute the Gini index for the Car Type attribute using multiway split.**

In this case, there are three types of cars- Family, Luxury and Sports.

Family car type with class C0 has 1 and C1 has 3 counts.

Luxury car type with class C0 has 1 and C1 has 7 counts.

Sports car type with class C0 has 8 and C1 has 0 counts.

- The Gini index for Family car is 0.375.
- The Gini index for Sports car is 0.
- The Gini index for Luxury car is 0.2188.
- The overall Gini index for Car Type is 0.1625.

**(e) Compute the Gini index for the Shirt Size attribute using multiway split.**

There are four types of shirt size – Extra large, Large, Medium and small.

- The Gini index for Small shirt size is 0.48.
- The Gini index for Medium shirt size is 0.4898.
- The Gini index for Large shirt size is 0.5.
- The Gini index for Extra Large shirt size is 0.5.
- The overall Gini index for shirt size attribute is 0.4914.

**(f) Which attribute is better, Gender, Car Type, or Shirt Size?**

Car type is a better attribute because it has the lowest Gini Index that is 0.1625 as compared to Gender (0.5) and Shirt Size (0.4914).

**(g) Explain why Customer ID should not be used as the attribute test condition even though it has the lowest Gini.**

Customer id has a unique value. Whenever there is a new customer, will assign a new id. So this attribute can not be used as predictive attribute.

## Section 3.11 Question 3:

**(a) What is the entropy of this collection of training examples with respect to the class attributes?**

- The entropy for target class is .99107

**(b) What are the information gains of a1 and a2 relative to these training examples?**

Attribute a1:

- Entropy of False for a1: 0.7219280948873623
- Entropy of True for a1: 0.8112781244591328
- The overall Entropy for a1: 0.7616392191414825
- Information Gain for a1: 0.22943684069673975

Attribute a2:

- Entropy of False for a2: 1.0
- Entropy of True for a2: 0.9709505944546686
- The overall Entropy for a2: 0.9838614413637048
- Information Gain for a2: 0.007214618474517431

**(c) For a3, which is a continuous attribute, compute the information gain for every possible split.**

| a3 | Class label | Split point | Entropy | Information Gain |
|---|---|---|---|---|
| 1.0 | + | 2.0 | 0.8484 | 0.1427 |
| 3.0 | - | 3.5 | 0.9885 | 0.0026 |
| 4.0 | + | 4.5 | 0.9183 | 0.0728 |
| 5.0<br>5.0 | -<br>- | 5.5 | 0.9839 | 0.0072 |
| 6.0 | + | 6.5 | 0.9728 | 0.0183 |
| 7.0<br>7.0 | +<br>- | 7.5 | 0.8889 | 0.1022 |

After reviewing the above table, we can see that best split for a3 occurs at split point equals to 2.

**(d) What is the best split (among a1, a2, and a3) according to the information gain?**

- Information Gain for a1 = 0.2294
- Information Gain for a2 = 0.0072
- Information Gain for a3 at split point 2 = 0.1427

So, according to the information gain a1 is the best split.

**(e) What is the best split (between a1 and a2) according to the classification error rate?**

For attribute a1:

- false error rate: 0.19999999999999996
- true error rate: 0.25
- error rate: 0.2222222222222222

For attribute a2:

- false error rate: 0.5
- true error rate: 0.4
- error rate: 0.4444444444444444

According to error rate a1 produces the best split.

(f) **What is the best split (between a1 and a2) according to the Gini index?**

**For attribute a1:**

- Gini Index of true for a1: 0.375
- Gini Index of false for a1: 0.31999999999999984
- The overall Gini Index of a1:  0.34444444444444433

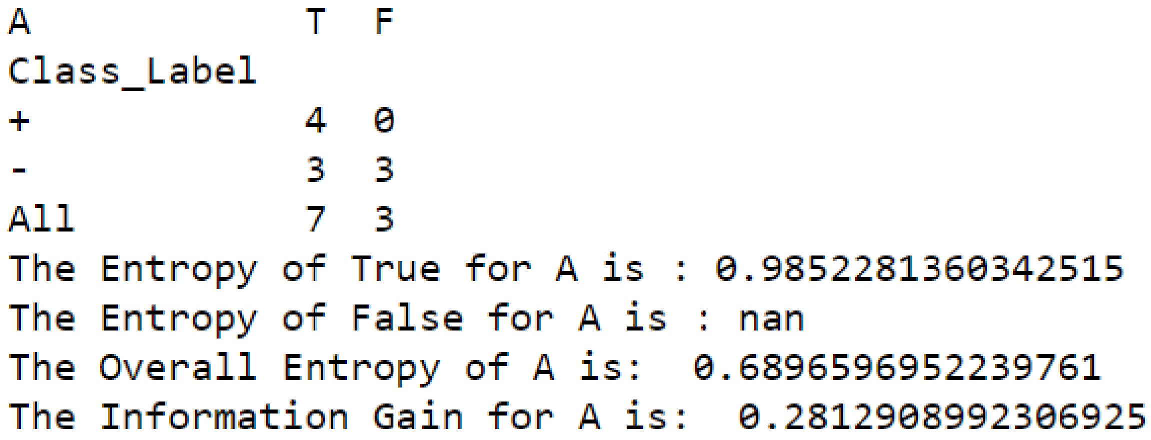
For attribute a2:

- Gini Index of true for a2: 0.48
- Gini Index of false for a2:  0.5
- The overall Gini Index of a2:  0.4888888888888889

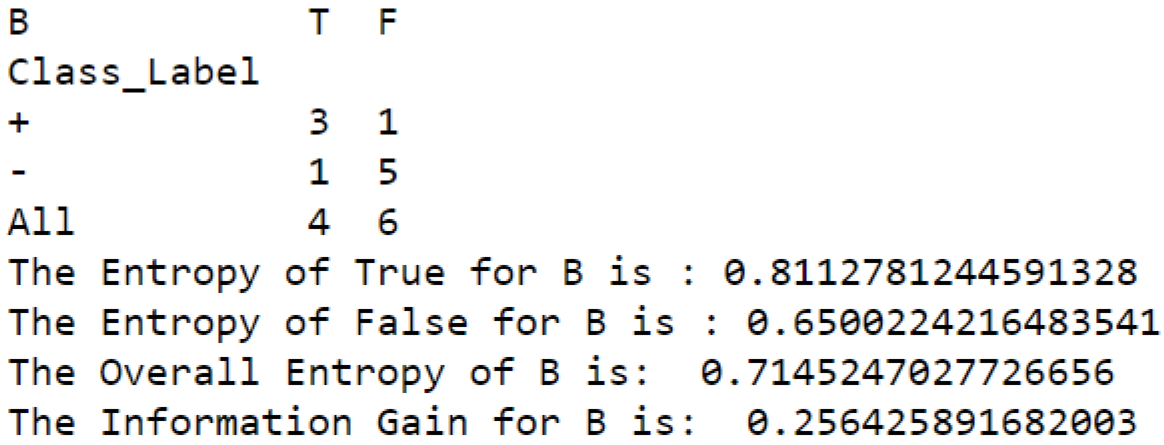
Since the Gini index for a1 is smaller, it produces the best split.

## Section 3.11 Question 5:

(a) **Calculate the information gain when splitting on ,4 and B. Which at- tribute would the decision tree induction algorithm choose?**



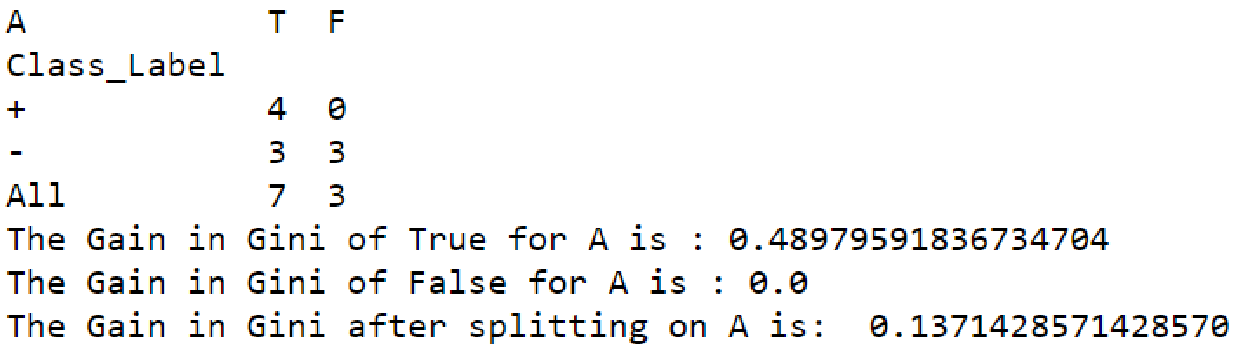
```
A               T  F
Class_Label
+               4  0
-               3  3
All             7  3
The Entropy of True for A is : 0.9852281360342515
The Entropy of False for A is : nan
The Overall Entropy of A is:  0.689659695223976l
The Information Gain for A is:  0.2812908992306925
```



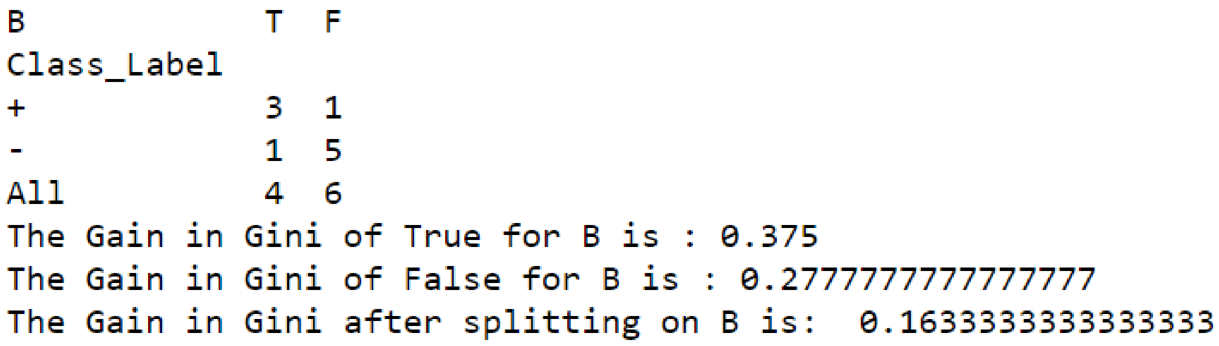
```
B               T  F
Class_Label
+               3  1
-               1  5
All             4  6
The Entropy of True for B is : 0.8112781244591328
The Entropy of False for B is : 0.6500224216483541
The Overall Entropy of B is:  0.7145247027726656
The Information Gain for B is:  0.2564258916820003
```

After comparing the information gain of attribute, A and B. Attribute a will be chosen to split the node.

**(b) Calculate the gain in the Gini index when splitting on A and B. Which attribute would the decision tree induction algorithm choose?**



```
A              T  F
Class_Label
+              4  0
-              3  3
All            7  3
The Gain in Gini of True for A is : 0.48979591836734704
The Gain in Gini of False for A is : 0.0
The Gain in Gini after splitting on A is:  0.1371428571428570
```



```
B              T  F
Class_Label
+              3  1
-              1  5
All            4  6
The Gain in Gini of True for B is : 0.375
The Gain in Gini of False for B is : 0.2777777777777777
The Gain in Gini after splitting on B is:  0.1633333333333333
```

After comparing the Gini index of attribute, A and B. B will be chosen to split the node.

**(c) Figure 4.13 shows that entropy and the Gini index are both monotonously increasing on the range [0, 0.5] and they are both monotonously decreasing on the range [0.5, 1]. Is it possible that information gain and the gain in the Gini index favor different attributes? Explain.**

**Answer:**

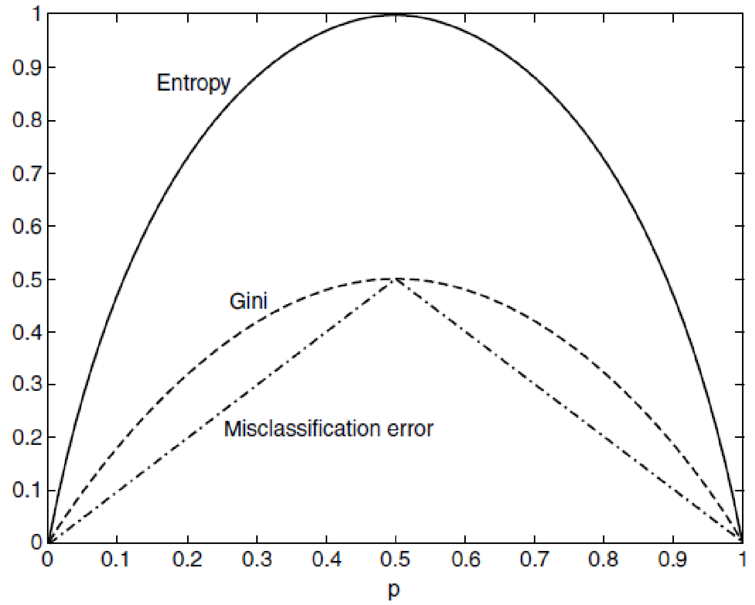


**Figure 4.13.** Comparison among the impurity measures for binary classification problems.

Yes, entropy and Gini index have similar range and monotonous behavior, their respective gains which are scaled differences of the measures, do not necessarily behave in the same way.

## Section 3.11 Question 7:

a) **Compute a two-level decision tree using the greedy approach described in this chapter. Use the classification error rate as the criterion for splitting. What is the overall error rate of the induced tree?**

**Answer:**

Splitting attribute at Level 1:

- The classification error for attribute X is: 0.5
- The classification error for attribute Y is: 0.4
- The classification error for attribute Z is: 0.3

The lowest classification error is Z. therefore the next step is to split Z.

Splitting attribute at Level 2:

- For Z=0, the error rate for X and Y is 0.3
- For Z=1, the error rate for X and Y is 0.3
- The overall error rate for the induced tree is: 0.3

b) **Repeat part (a) using X as the first splitting attribute and then choose the best remaining attribute for splitting at each of the two successornodes. What is the error rate of the induced tree?**

**Answer:**

For X=0, the error rate for Y is 0.0833

For X=0, the error rate for Z is 0.25

As the attribute Y leads to a smaller error rate, the next step is to split Y.

For X=1, the error rate for Y is 0.125
For X=1, the error rate for Z is 0.375
As the attribute Y leads to a smaller error rate, the next step is to split Y.

The overall Error Rate of the induced tree is 0.1.

c) **Compare the results of parts (a) and (b). Comment on the suitability of the greedy heuristic used for splitting attribute selection.**

**Answer:** When comparing the results of parts a and b, the suitability of the greedy heuristic does not produce optimum outcomes.

## Section 4.14 Question 1:

**Consider a binary classification problem with the following set of attributes and attribute values:**

- **Air Conditioner= {Working, Broken}**
- **Engine = {Good, Bad}**
- **Mileage = {High, Medium, Low}**
- **Rust = {Yes, No}**

**Suppose a rule-based classifier produces the following rule set:**

**Mileage = High → Mileage = High Mileage = Low→ Value = High Air Conditioner**

**(a) Are the rules mutually exclusive?**

Answer: No

(**b) Is the rule set exhaustive?**

Answer: Yes

**(d) Is ordering needed for this set of rules?**

Answer: Yes, because a test instance may trigger more than one rule.

**(d) Do you need a default class for the rule set?**

Answer: No because every instance is guaranteed to trigger at least one rule.

## Section 4.14 Question 12:

**Consider the one-dimensional data set shown in Table 4.12.**

1. **Classify the data point x=5.0 according to its 1-, 3-, 5-, and 9- nearest neighbors (using majority vote).**

Euclidean distance (one dimension) = squareroot $(x-x_i)^2$

| X | y | Euclidean distance(x=5) | 1-nearest neighbor | 3-nearest neighbor | 5-nearest neighbor | 9-nearest neighbor |
|---|---|---|---|---|---|---|
| 0.5 | - | 4.5 | | | | - |
| 3.0 | - | 2 | | | | - |
| 4.5 | + | 0.5 | | | | + |
| 4.6 | + | 0.4 | | | +(5th) | + |
| 4.9 | + | 0.1 | + | + | + | + |
| 5.2 | - | 0.2 | | - | - | - |
| 5.3 | - | 0.3 | | - | - | - |
| 5.5 | + | 0.5 | | | +(5th) | + |
| 7.0 | - | 2 | | | | - |
| 9.5 | - | 4.5 | | | | |
| Classification | | | + | - | + | - |

2. Repeat the previous analysis using the distance-weighted voting approach

(distance-weight) wi=1/d(x′ , xi)2

| X | y | Euclidean distance(x=5) | Distance weight | 1-nearest neighbor | 3-nearest neighbor | 5-nearest neighbor | 9-nearest neighbor |
|---|---|---|---|---|---|---|---|
| 0.5 | - | 4.5 | 0.049 | | | | - |
| 3.0 | - | 2 | .25 | | | | - |
| 4.5 | + | 0.5 | 4 | | | + | + |
| 4.6 | + | 0.4 | 6.25 | | + | + | + |
| 4.9 | + | 0.1 | 100 | + | + | + | + |
| 5.2 | - | 0.2 | 25 | | - | - | - |
| 5.3 | - | 0.3 | 11.11 | | | - | - |
| 5.5 | + | 0.5 | 4 | | | + | + |
| 7.0 | - | 2 | .25 | | | | - |
| 9.5 | - | 4.5 | .05 | | | | |

1-nearest neighbor = (0.1) *100=10
= +

3-nearest neighbor= ((0.4 * 6.25) + (0.1*100)) or (0.2*25)
=12 or 5
= +

5-nearest neighbor= ((0.5*4) + (0.4 * 6.25) + (0.1*100) +(0.5*4)) or ((0.2*25) +(0.3*11.11))
= (16.5) or (8.33)
= +

**9-nearest neighbor = 16.5 or (4.5*.049) + (2*.25) + (2*.25) + 8.33**
**=16.5 or 9.55**
**= +**

## Section 4.14 Question 15:

1. **Demonstrate how the perceptron model can be used to represent the AND and OR functions between a pair of Boolean variables.**

If x1 and x2 be a pair of Boolean variables and y is the output.

For AND function, a possible perceptron model between a pair of Boolean variables would be:

Y = sgn [x1 + x2 – 1.5]

For OR function, a possible perceptron model between a pair of Boolean variables would be:

Y = sgn [x1 + x2 – 0.5]

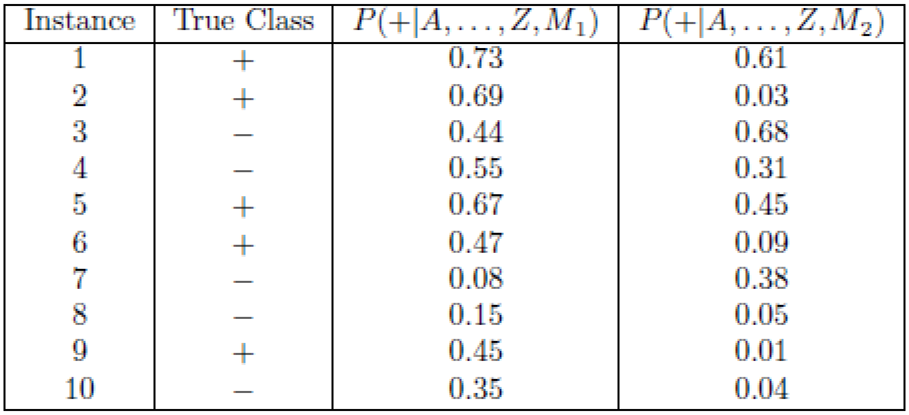
2. **Comment on the disadvantage of using linear functions as activation functions for multi-layer neural networks.**

With linear activation functions, all layers of the neural network collapse into one. No matter how many layers in the neural network, the last layer will be a linear function of the first layers. So, a linear activation function turns the multi-layers neural network into just one layers. A multilayer neural network with a linear activation function is simply a linear regression model. It has limited power and ability to handle complexity varying parameters of input data. Such a network is just as expressive as a perceptron.

## Section 4.14 Question 16:

**You are asked to evaluate the performance of two classification models, M1 and M2. The test set you have chosen contains 26 binary attributes, labeled as A through Z. Table 4.13 shows the posterior probabilities obtained by applying the models to the test set. (Only the posterior probabilities for the positive class are shown). As this is a two-class problem, P (−) = 1−P (+) and P (−|A, ···, Z) =1−P (+|A, ···, Z). Assume that we are mostly interested in detecting instances from the positive class.**

1. **Plot the ROC curve for both M1 and M2. (You should plot them on the same graph.) Which model do you think is better? Explain your reasons.**



| Instance | True Class | $P(+\|A,\ldots,Z,M_1)$ | $P(+\|A,\ldots,Z,M_2)$ |
|---|---|---|---|
| 1 | + | 0.73 | 0.61 |
| 2 | + | 0.69 | 0.03 |
| 3 | − | 0.44 | 0.68 |
| 4 | − | 0.55 | 0.31 |
| 5 | + | 0.67 | 0.45 |
| 6 | + | 0.47 | 0.09 |
| 7 | − | 0.08 | 0.38 |
| 8 | − | 0.15 | 0.05 |
| 9 | + | 0.45 | 0.01 |
| 10 | − | 0.35 | 0.04 |

a) sort the test scores in increasing order of their output values.

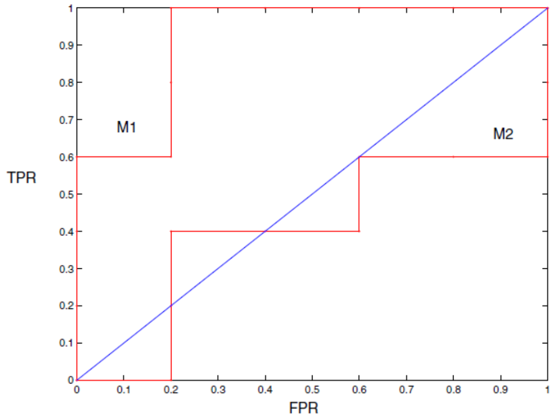
| Instance | True class | P ($M_1$) |
|---|---|---|
| 1 | + | 0.73 |
| 2 | + | 0.69 |
| 5 | + | 0.67 |
| 4 | - | 0.55 |
| 6 | + | 0.47 |
| 9 | + | 0.45 |
| 3 | - | 0.44 |
| 10 | - | 0.35 |
| 8 | - | 0.15 |
| 7 | - | 0.08 |

| Instance | True class | P (M2) |
|---|---|---|
| 3 | - | 0.68 |
| 1 | + | 0.61 |
| 5 | + | 0.45 |
| 7 | - | 0.38 |
| 4 | - | 0.31 |
| 6 | + | 0.09 |
| 8 | - | 0.05 |
| 10 | - | 0.04 |
| 2 | + | 0.03 |
| 9 | + | 0.01 |

b) classify TRP and FPR

| Instance | True class | P ($M_1$) | TRP | FPR |
|---|---|---|---|---|
| 1 | + | 0.73 | 1 | 0 |
| 2 | + | 0.69 | 2 | 0 |
| 5 | + | 0.67 | 3 | 0 |
| 4 | - | 0.55 | 3 | 1 |
| 6 | + | 0.47 | 4 | 1 |
| 9 | + | 0.45 | 5 | 1 |
| 3 | - | 0.44 | 5 | 2 |
| 10 | - | 0.35 | 5 | 3 |
| 8 | - | 0.15 | 5 | 4 |
| 7 | - | 0.08 | 5 | 5 |

| Instance | True class | P (M2) | TRP | FPR |
|---|---|---|---|---|
| 3 | - | 0.68 | 0 | 1 |
| 1 | + | 0.61 | 1 | 1 |
| 5 | + | 0.45 | 2 | 1 |
| 7 | - | 0.38 | 2 | 2 |
| 4 | - | 0.31 | 2 | 3 |
| 6 | + | 0.09 | 3 | 3 |
| 8 | - | 0.05 | 3 | 4 |
| 10 | - | 0.04 | 3 | 5 |
| 2 | + | 0.03 | 4 | 5 |
| 9 | + | 0.01 | 5 | 5 |



M1 is better because area under the ROC curve is larger than the area under ROC curve for M2.

2. **For model M1, suppose you choose the cutoff threshold to be t=0.5. In other words, any test instances whose posterior probability is greater than t will be classified as a positive**

**example. Compute the precision, recall, and F-measure for the model at this threshold value.**

Cut off threshold is t =0.5

TP = (1,2,5)

TN = (3,7,8,10)

FN = (6,9)

Precision (p) = TP / (TP+FP) = 3/3+1 = 0.75

Recall (r) = TP/(TP+FN) = 3/3+2 = 0.6

The harmonic mean between recall and precision is $F_1$: 2/(1/r + 1/p) = 0.667

3. **Repeat the analysis for part (b) using the same cutoff threshold on model M2. Compare the F-measure results for both models. Which model is better? Are the results consistent with what you expect from the ROC curve?**

TP = (1)

TN = (7,4,8,10)

FN = (5,6,2,9)

FP = (3)

Precision (P) = 1 / (1+1) = 0.5

Recall (r)= 1 / (1+4) =0.2

F1= 2 / (1/0.2 + 1/0.5) = 0.2857

When comparing M1 and M2. F1 is larger for M1 which indicating a better model.

4. **Repeat part (b) for model M1 using the threshold t=0.1. Which threshold do you prefer, t=0.5 or t=0.1? Are the results consistent with what you expect from the ROC curve?**

Threshold is t = 0.1

TP = (1,2,5,6,9)

TN = (7)

FN = (0)

FP = (4,1,0,8)

Precision (P) = 5 / (5+4) = 0.5556

Recall (r)= 5 / (5+0) =1

F1= 2 / (1/1 + 1/0.55556) = 0.7143

When comparing the threshold values of 0.1 and 0.5, 0.1 indicates a larger F1 which concludes that it is a preferred threshold. This also matches what we found in part a when plotting the ROC curve.