Task 1

1) Explanation for feature selection: -
   a. The first feature I have taken is related to Age. Age column helps identifying the age of the traveler and if he/she is young then it increases the chance o0f survival.
   b. The second feature selected is the sex of the traveler. As it was observed that the number of female survivors was more than that of male survivor it is an important feature.
   c. The next feature to consider was the number of siblings and spouse. It was necessary as the sibling or spouse would be travelling together and chances of survival would be more.
   d. The next feature I considered is Parch, it is the number of parent and children on board. The reason was that since the group is travelling together then the chances of survival would be more.
   e. The next feature I considered was class of the ticket. In case of emergency the first-class passenger would have better chances of surviving than the other classes.

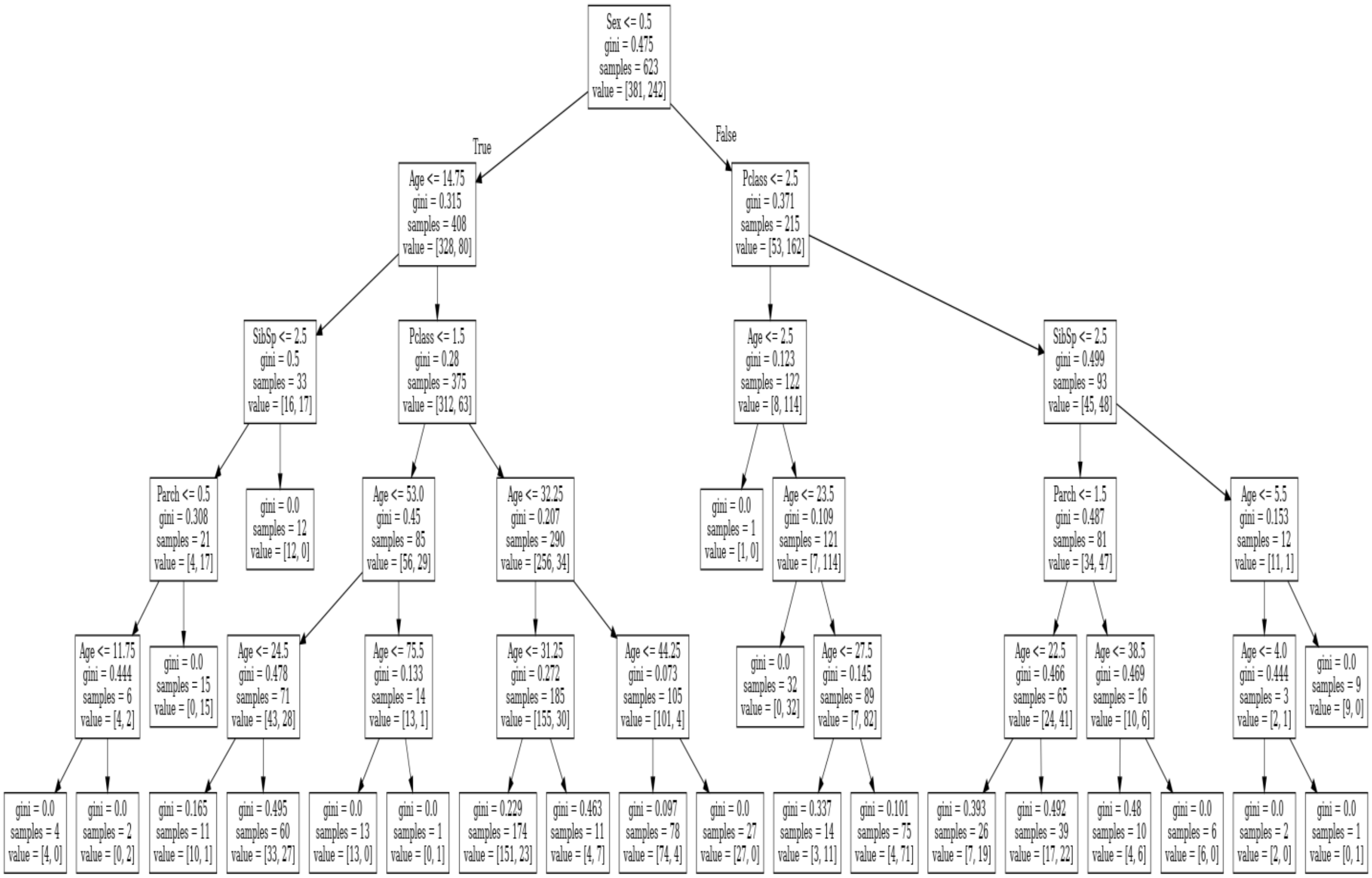
2) Accuracy of Decision tree → **76.57%**
3) Accuracy of Random Forest → **80.39%**
4) Which algorithm is better, Decision Tree or Random Forest?

→ We can observe that random forest is better algorithm compared to decision tree. The reason being that the random forest has better accuracy of the two. Random forest also was able to provide with better solution by considering different features instead of just considering a single feature as decision tree. Random forest combines multiple outputs together to generate the final accuracy.

5) What are your observations and conclusions from the algorithm comparison and analysis?

→ Following are the observations and conclusion after the analysis: -

- Whenever we make a change in feature set the output accuracy and the size of the decision tree changed.
- While preprocessing the data it is important for handling the null data or missing data.
- While selecting the features, it is necessary to understand the output of the required dataset. The reason being that few columns while important don’t make any difference in keeping in the dataset.
- While fitting the decision tree, it is very important for selecting the splitting criteria. It was observed that on changing the splitting criteria the accuracy of the decision tree was changing.
- The change in the depth of the tree would slightly affect the accuracy of the tree.
- In case of random forest upon increasing the number of random states the accuracy of the forest increased but after a certain threshold it remained the same.
- The value of estimators and depth help in determining the accuracy and status of the tree.
- It was also concluded that the output of random forest is better than that of the decision tree.


Sex <= 0.5
gini = 0.475
samples = 623
value = [381, 242]
True
False
Age <= 14.75
gini = 0.315
samples = 408
value = [328, 80]
Pclass <= 2.5
gini = 0.371
samples = 215
value = [53, 162]
SibSp <= 2.5
gini = 0.5
samples = 33
value = [16, 17]
Pclass <= 1.5
gini = 0.28
samples = 375
value = [312, 63]
Age <= 2.5
gini = 0.123
samples = 122
value = [8, 114]
SibSp <= 2.5
gini = 0.499
samples = 93
value = [45, 48]
Parch <= 0.5
gini = 0.308
samples = 21
value = [4, 17]
gini = 0.0
samples = 12
value = [12, 0]
Age <= 53.0
gini = 0.45
samples = 85
value = [56, 29]
Age <= 32.25
gini = 0.207
samples = 290
value = [256, 34]
gini = 0.0
samples = 1
value = [1, 0]
Age <= 23.5
gini = 0.109
samples = 121
value = [7, 114]
Parch <= 1.5
gini = 0.487
samples = 81
value = [34, 47]
Age <= 5.5
gini = 0.153
samples = 12
value = [11, 1]
Age <= 11.75
gini = 0.444
samples = 6
value = [4, 2]
gini = 0.0
samples = 15
value = [0, 15]
Age <= 24.5
gini = 0.478
samples = 71
value = [43, 28]
Age <= 75.5
gini = 0.133
samples = 14
value = [13, 1]
Age <= 31.25
gini = 0.272
samples = 185
value = [155, 30]
Age <= 44.25
gini = 0.073
samples = 105
value = [101, 4]
gini = 0.0
samples = 32
value = [0, 32]
Age <= 27.5
gini = 0.145
samples = 89
value = [7, 82]
Age <= 22.5
gini = 0.466
samples = 65
value = [24, 41]
Age <= 38.5
gini = 0.469
samples = 16
value = [10, 6]
Age <= 4.0
gini = 0.444
samples = 3
value = [2, 1]
gini = 0.0
samples = 9
value = [9, 0]
gini = 0.0
samples = 4
value = [4, 0]
gini = 0.0
samples = 2
value = [0, 2]
gini = 0.165
samples = 11
value = [10, 1]
gini = 0.495
samples = 60
value = [33, 27]
gini = 0.0
samples = 13
value = [13, 0]
gini = 0.0
samples = 1
value = [0, 1]
gini = 0.229
samples = 174
value = [151, 23]
gini = 0.463
samples = 11
value = [4, 7]
gini = 0.097
samples = 78
value = [74, 4]
gini = 0.0
samples = 27
value = [27, 0]
gini = 0.337
samples = 14
value = [3, 11]
gini = 0.101
samples = 75
value = [4, 71]
gini = 0.393
samples = 26
value = [7, 19]
gini = 0.492
samples = 39
value = [17, 22]
gini = 0.48
samples = 10
value = [4, 6]
gini = 0.0
samples = 6
value = [6, 0]
gini = 0.0
samples = 2
value = [2, 0]
gini = 0.0
samples = 1
value = [0, 1]

Task 2

1) What is the training error rate for the tree? Explain how you get the answer?

- To calculate the training rate of the entire tree we first find the outliers in the tree nodes. Then we take the sum of their weights attached and divide the sum by the total sum of the weights of all the features.
  - The outlier weight at left node of D ➔ 5
  - The outlier weight at right node of D ➔ 6
  - The outlier weight at left node of E ➔ 2
  - The outlier weight at right node of E ➔ 6
  - The outlier weight at left node of C ➔ 5
  - The outlier weight at right node of C ➔ 5
- Then we take the sum of all these weights.
  - Sum of outlier weight = (5+6+2+6+5+5) = **29**
- Then we take sum of all the weights.
  - Sum of all weights = (5+14+6+7+2+10+8+6+5+17+15+5) = **100**
- Error rate would be then the average of both sums
  - Error rate = 29/100

    = **<u>0.29</u>**

- Another method to calculate would be using the misclassification error of all the nodes and then taking an average of all the error values.
- The steps would be as follows: -
  - Take the misclassification error at left node of D → 1- max (14/19,5/19) = **0.2631**
  - Take the misclassification error at right node of D → 1- max (6/13,7/13) = **0.4615**
  - Take the misclassification error at left node of E → 1- max (2/12,10/12) = **0.167**
  - Take the misclassification error at right node of E → 1- max (8/14,6/14) = **0.4286**
  - Take the misclassification error at left node of C → 1- max (5/22,17/22) = **0.2272**
  - Take the misclassification error at right node of C → 1- max (15/20,5/20) = **0.25**
- The next step would be to take the average of all the error rates.
  - Average error rate = (0.2631 + 0.4615 + 0.167 + 0.4286 + 0.2272 + 0.25) / 6 = **<u>0.299</u>**

The reason we choose the classification error method to calculate the training error is because we want to calculate the rate of the outliers present in the nodes. These outliers are the error present in the nodes. We therefore can determine the error rate from this step.

2) Given a test instance T= {A=0, B=1, C=1, D=1, E=0}, what class would the decision tree above assign to T? Explain how you get the answer?

➔)

- The class assigned would the dominant class of the left node of **‘E’** which is **‘+’**.
- The reason is that according to the test data T= {A=0, B=1, C=1, D=1, E=0}, the path followed from A would be ‘0’.
- The next node would ‘B’ where the value is ‘1’.
- The next node would be ‘E’ where the value is ‘0’.
- This bring to the leaf node where the dominant class value is ‘+’.
- We can also consider the error rates of all the leaf nodes. The least error rate is for left node of the leaf node ‘E’ instead of the right node. The error calculated in the previous answer can be considered. (Left node = **0.167** && Right node **= 0.4286)**.
- The reason it would not be going to other leaf nodes is because there is no connecting path between the stated path derived from the test data and the given decision tree.

Task 3

Task 3 screenshots have been attached below.

The overall entropy of the tree is → **0.971**

The information gain after splitting A is → **0.28148**

The information gain after splitting B is → **0.2565**

After calculating the both the information gain, we can conclude that decision tree would choose the “A” attribute for splitting and then after the first split it will choose “B” attribute for splitting the tree.

Task 3

1)

A	B	CL
T	F	+
T	T	+
T	T	+
T	F	−
T	T	+
F	F	−
F	F	−
F	F	−
T	T	−
T	F	−

entropy of the entire tree will be calculated as below:-

⟹ 4 positive instance, 6 neg instance

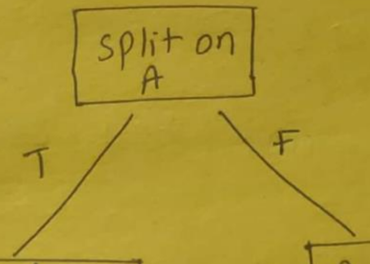
$$\text{entropy} = \frac{-4}{10}\left(\log\frac{4}{10}\right) - \frac{6}{10}\left(\log_2\frac{6}{10}\right)$$

$$= -\frac{4}{10}\left(-1.32198\right) - \frac{6}{10}\left(-0.7369\right)$$

$$= 0.970968$$

$$\approx \underline{\underline{0.971}}$$

2) splitting on A



A	B	CL
T	F	+
T	T	+
T	T	+
T	F	−
T	T	+
T	T	−
T	F	−

A	B	CL
F	F	−
F	F	−
F	F	−

∴ entropy of left node ⟹ 4 +ve instance
3 −ve instance.

$$= -\frac{4}{7}\left(\log_2 \frac{4}{7}\right) - \frac{3}{7}\left(\log_2 \frac{3}{7}\right).$$

$$= -\frac{4}{7}(-0.8073) - \frac{3}{7}(-1.2223)$$

= 0.985~~[illegible]~~028

≈ ~~0.986~~ 0.9850

∴ entropy of right node ⟹ 3 −ve instance
0 +ve ~~pos~~ instance.

$$= -\frac{3}{3}\left(\log_2 \frac{3}{3}\right) - 0.$$

$= 0.$

∴ Information gain when splitted at A will be

$$= \frac{7}{10}\left[0.9850\right] + 0.$$

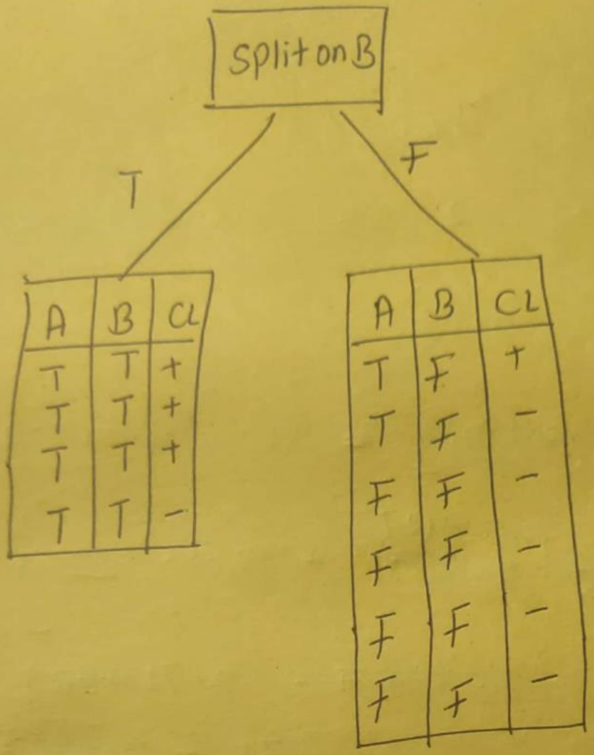
= ~~0.6902.~~ 0.68952

Information gain = 0.971 − ~~0.6902~~ 0.68952

= ~~0.2808.~~ 0.28148

Information gain at point A = 0.28148

3) Splitting on B.



A	B	Cl
T	T	+
T	T	+
T	T	+
T	T	−

A	B	Cl
T	F	+
T	F	−
F	F	−
F	F	−
F	F	−
F	F	−

∴ entropy on left node ⇒ 3 +ve instances.
1 −ve instances.

$$= -\frac{3}{4}\left[\log_2\left(\frac{3}{4}\right)\right] - \frac{1}{4}\left[\log_2\left(1/4\right)\right]$$

$$= -\frac{3}{4}\left[-0.41503\right] - \frac{1}{4}\left[-2\right]$$

= ~~0.1887~~ 0.81127

entropy on right node => 1 +ve instance
5 -ve instance.

$$= -\frac{1}{6}\left[\log_2\left(\tfrac{1}{6}\right)\right] - \tfrac{5}{6}\left[\log_2\left(\tfrac{5}{6}\right)\right].$$

$$= -\frac{1}{6}\left[-2.5849\right] - \tfrac{5}{6}\left\{-0.2630\right].$$

$$= 0.65.$$

∴ Combined entropy for information gain

$$= \left[\frac{4}{10}\left(\overset{0.81127}{\cancel{0.1887}}\right) + (0.65)\,\frac{6}{10}\right].$$

= ~~0.4654~~ 0.714509

∴ Information gain = 0.971 − ~~0.4654~~ 0.714509

= − ~~0.5055~~.

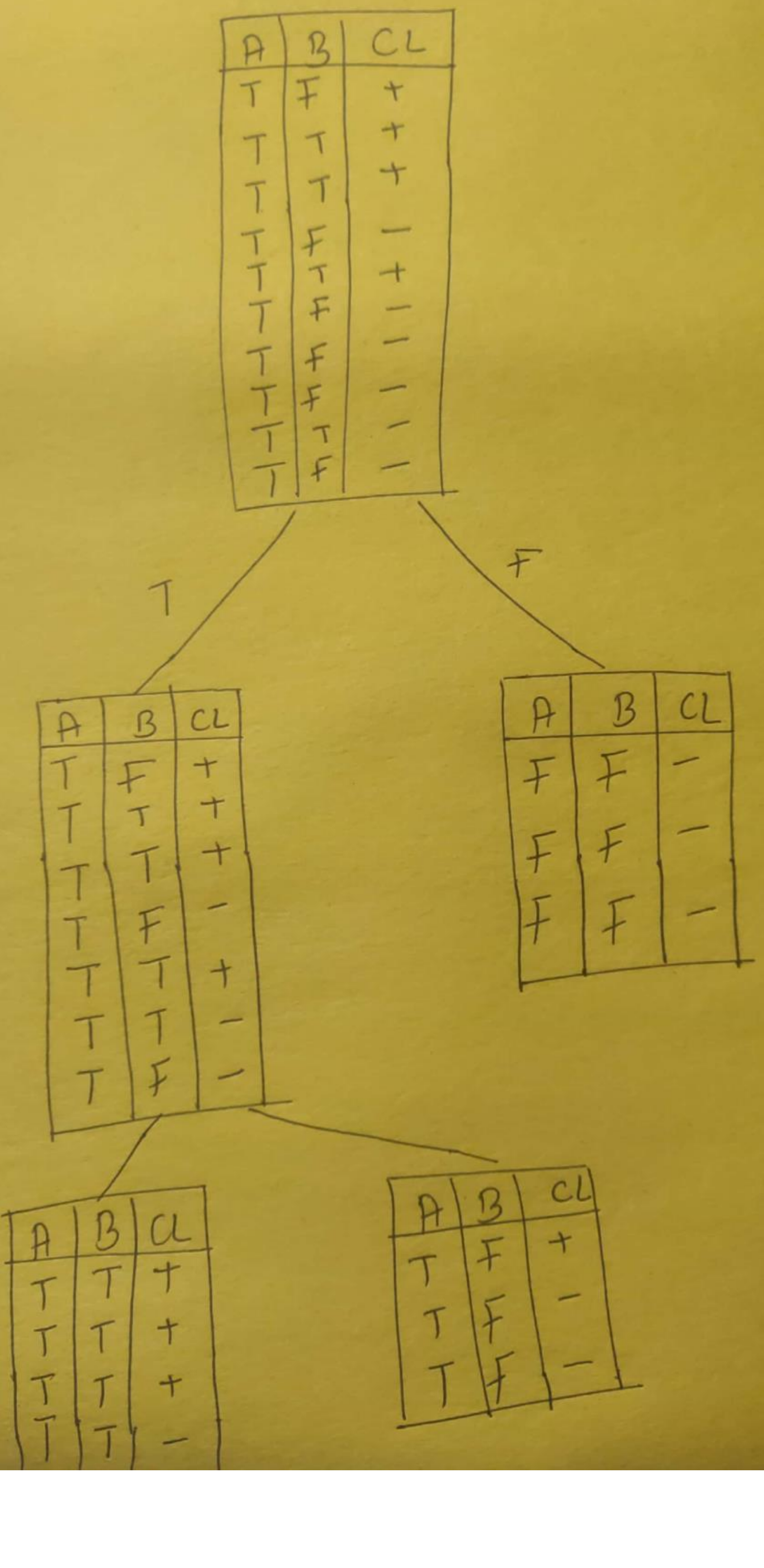
= 0.256491

Information ~~gal~~ gain at point B = 0.2565

4)

-> The decision Tree would choose attribute 'A' for splitting. As we have calculated the gain at both features, the gain is more at 'A' compared to 'B'. Hence we will use node 'A'.

5) Decision Tree.



A	B	CL
T	F	+
T	T	+
T	T	+
T	F	—
T	T	+
T	F	—
T	F	—
T	F	—
T	T	—
T	F	—

A	B	CL
T	F	+
T	T	+
T	T	+
T	F	−
T	T	+
T	T	−
T	F	−

A	B	CL
F	F	—
F	F	—
F	F	—

A	B	CL
T	T	+
T	T	+
T	T	+
T	T	—

A	B	CL
T	F	+
T	F	—
T	F	—

Task 4

4.1) Are decision trees a linear classifier?

- Decision tree are not a linear classifier. It is used for fitting linearly inseparable datasets. Inseparable datasets can be defined as the data points between classes that cannot be separated by a single line. A linear classifier will classify based on a value of a linear combination of different features. Decision tree splits the data based on a feature value instead of its distance. The feature of a dataset can be tightly bound, which in case of linear classification only single line cannot be used to distinguished. This is one of the distinguishing factors that the decision tree is nonlinear classifier.

4.2) What are the weaknesses of decision trees?

- Weakness of decision tree are as follows: -
  - A small change in the dataset will cause the decision tree to become unstable.
  - Decision tree tends to split a node into many partitions, each partition is small and unique. This is a problem because the decision tree since decision tree algorithm uses greedy approach the tree will be unbalanced and overfitted.
  - Lack of generalization. This may arise when the tree gets split into many partitions then each leaf node will be unique. This will make it harder find any generalized term for any conclusion.
  - Not able to work better in case of regression or having continuous input. Decision trees work better when they are trained to assign a data point to a select class but when they have a continuous output such as stock price they do not work as well.

4.3) Is Misclassification errors better than Gini index as the splitting criteria for decision trees?

- In certain cases, misclassification error can be better than Gini index as a splitting criterion for decision tree. Usually whenever the user wants to maximize or increase the classification accuracy then misclassification error can be used. The problem with this can be since decision tree uses greedy algorithm the noise level will also increase at each level. Therefore, in case of the noisy data instead of it getting removed it will increase. While Gini is better in this scenario, but Gini index is more sensitive to changes for node changes or probabilities than misclassification error. The advantage of Gini is that since Gini index is differentiable the y can handle more numerical optimization. Therefore, in conclusion even though all three measures are similar we can say that misclassification error can be used in specific instances where we want to maximize the accuracy otherwise, we will have to use Gini index.

Code Link

https://drive.google.com/drive/folders/1TZjL1a0lD_eI2RpoqPba302cwERS05-0?usp=sharing