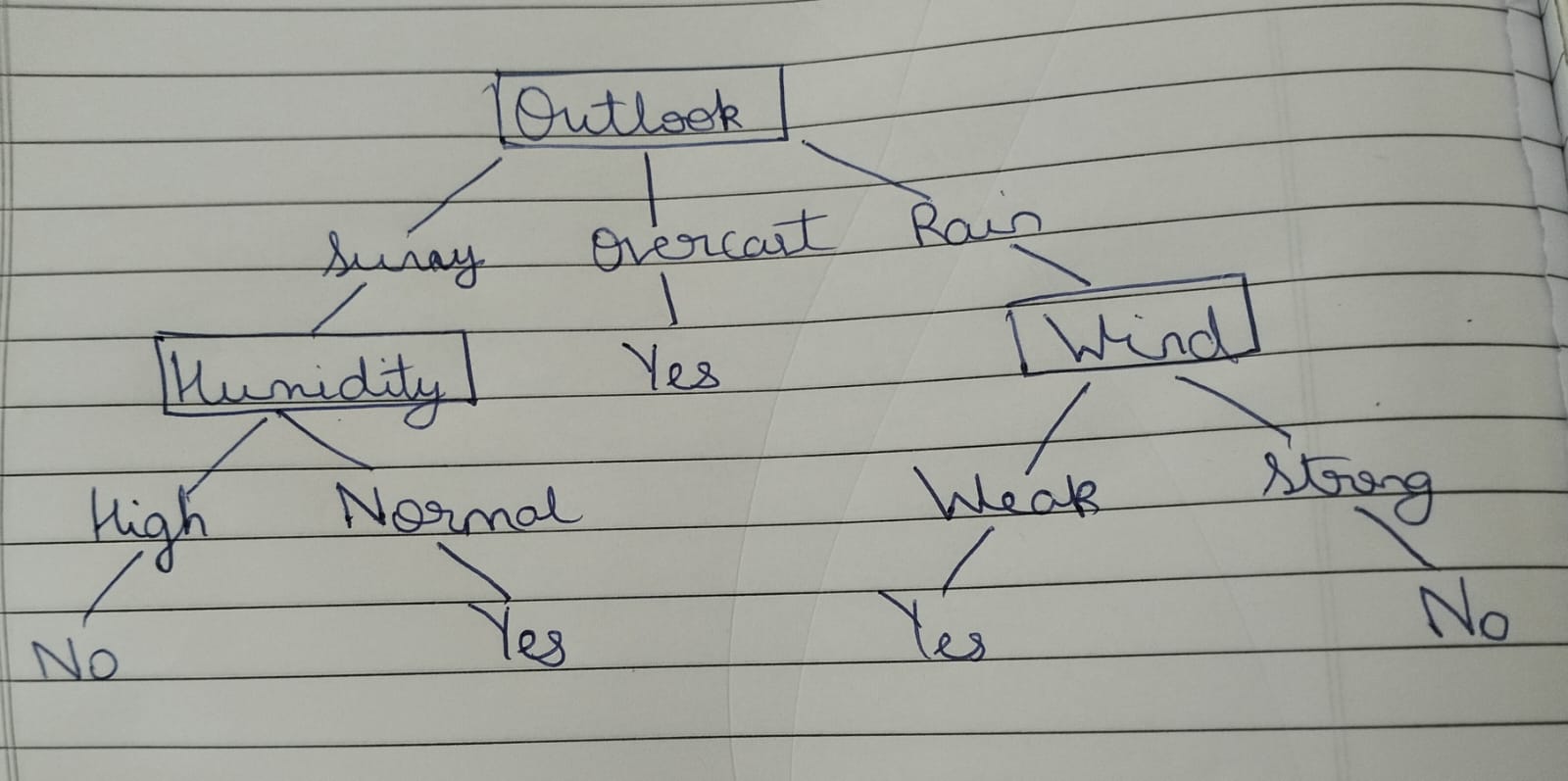
**Data Science Assignment**

**1)** **Perform Exploratory Data Analysis (EDA) on Iris dataset.**

[iris.ipynb - Colaboratory (google.com)](https://colab.research.google.com/drive/1UP7NfcciTxz-XkAbQ2oGMxbWE3qgVenh)

**2) What is Decision Tree? Draw decision tree by taking the example of Play Tennis**

Decision Tree is a flowchart like tree structure where each internal node represents a decision making process and each branch represents an outcome of the test and each terminal node is an outcome/class.



**3)** **In k-means or KNN, we use Euclidean distance to calculate the distance between nearest neighbours. Why not Manhattan distance**

Manhattan distance is defined as the sum of the vertical and horizontal distances between two points. It is given by the value (|x1-x2| + |y1-y2|).

We use Euclidean distance as it can span the entire dimension.

The Euclidean distance has been set as standard to avoid ambiguity as it is possible that the closest point can be different in both the systems which can lead to different outputs.

**4)** **How to test and know whether or not we have overfitting problem?**

Overfitting is the scenario when a model is able to perform well at the time of training and does not perform well at the time of testing.

It happens when the training dataset is huge and model learns to fit the data very closely i.e. the model becomes very specific. Due to this, it is unable to perform well on new test data.

**5) How is KNN different from k-means**

K - means is an unsupervised learning algorithm whereas KNN is classified as a supervised learning algorithm.

K – means is a clustering algorithm in which a set of points is partitioned into K sets such that the points in each set tend to be close to each other. These points have no external classification.

KNN or K – nearest neighbors is a classification or regression algorithm in which the class or value of the point in consideration is predicted by combining the classification or taking average of K nearest points respectively.

**6) Can you explain the difference between a Test Set and a Validation Set?**

Training Set – This dataset is used to train the model. It adjusts the weights on the neural network. The model tries to find patterns in this dataset and uses them to predict unknown inputs which are not in the training set.

Validation Set – This dataset is used to avoid overfitting. If we increase the accuracy in the training dataset but the accuracy in the validation set decreases or remains same, it implies that the neural network is overfitted and we should stop training anymore.

Testing Set – This dataset is used to predict the outputs of unknown input variables by observing the pattern of the Training Set. It is used to check the overall effectiveness of the network

**7)** **How can you avoid overfitting in KNN?**

KNN is one of the machine learning algorithms that is prone to overfitting. In order to avoid overfitting in KNN, it is a good practice to use a validation set to estimate the performance of a model on unseen data. By analyzing the model’s performance on unseen data, we can avoid overfitting.

We can also use cross-validation in which we will partition the data into subsets and each subset undergoes training and validation cycles.

In KNN, we can use a method called distance weighting, in which the closer neighbors have more weightage than the farther ones.

We can remove features that do not contribute to the accuracy of the model. This method is called feature selection.

We should add a penalty term to the objective function that measures the complexity of a model. This can help prevent overfitting of data. This is called regularization.

**8) What is Precision?**

Of all the points that the model predicted to be positive, the fraction of points that are actually positive is termed as Precision. It is used in Performance Measurement of models.

Precision = TP/(TP + FP)

**9) Explain How a ROC Curve works.**

ROC curve is a graph of TPR vs FPR for different values of threshold. All the probability scores which are greater than or equal to the threshold are predicted to have output 1 and the rest 0. We do this for threshold values corresponding to scores of all points i.e. if n points are there in set then we will have n threshold values. The TPR and FPR for each model are calculated and then plotted to obtain the ROC Curve. We then find Area under this curve to obtain the AUC. If the AUC is closer to 0, then it is a bad model and if it is closer to 1, then it is a good model.

**10) What is Accuracy?**

Accuracy is defined as the fraction of points that are correctly classified in a dataset. It is easy to understand but does not work well with imbalanced data.

Accuracy = Number of correctly classified points / Total number of points.

**11) What is F1 Score?**

F1 Score is defined as the harmonic mean of Precision and Recall.

Of all the points that the model predicted to be positive, the fraction of points that are actually positive is termed as Precision. It is used in Performance Measurement of models.

Precision = TP/(TP + FP)

Of all the points that are actually positive, the fraction of points that the model predicted to be positive is termed as Recall. It is used in Performance Measurement of models.

Recall = TP/TP+FN

**12) What is Recall?**

Of all the points that are actually positive, the fraction of points that the model predicted to be positive is termed as Recall. It is used in Performance Measurement of models.

Recall = TP/TP+FN

**13)What is a Confusion Matrix, and Why do we Need it?**

A confusion matrix is a table that shows how well a machine learning model is doing in terms of classification. It compares the predicted labels from the model with the actual labels from a test dataset. The table has four different sections that represent different types of predictions made by the model, and by looking at these different types of predictions, we can get a better sense of how well the model is doing overall

**14)What do you mean by AUC curve?**

The Area under the ROC curve is the AUC. If the AUC is closer to 0, then it is a bad model and if it is closer to 1, then it is a good model.

**15)What is Precision-Recall Trade-Off?**

A model with high precision means that when it predicts a positive result, it is likely to be correct. On the other hand, a model with high recall means that it can identify most of the positive cases in the dataset. Generally, when we try to increase one the other decreases. This phenomenon is called as Precision-Recall Trade-Off.

**16)What are *Decision Trees*?**

Decision Tree is a flowchart like tree structure where each internal node represents a decision-making process and each branch represents an outcome of the test and each terminal node is an outcome/class

**17)Explain the *structure* of a Decision Tree**

It starts with a root node which is the most general classification in the set. It branches off a finite number of times based on the answer of certain decisions. These decisions are represented by the branches in the tree. Each node corresponds to an output depending on the decision’s answer. The terminal/leaf nodes are the most specific classification. They represent the different possible outputs in the dataset.

**18)What are some *advantages* of using Decision Trees?**

Decision trees are easy to understand. They are simple.

They can handle large data similarly to small data which makes them more easier to understand.

It can also handle missing values if any and make decisions based on the available data. It can be used very effectively in terms of feature selection.

They can be easily pruned to avoid overfitting.

**19)How is a *Random Forest* related to *Decision Trees*?**

The algorithm Random Forest uses multiple decision trees to make predictions. Each decision tree is trained based on a particular subset of the entire dataset. The combined prediction of all the decision trees is the prediction of the random forest.

**20)How are the different nodes of decision trees *represented*?**

In a decision tree, the nodes are typically represented as circles or squares, with arrows connecting the nodes to show the flow of decisions. There are two main types of nodes in a decision tree: decision nodes and leaf nodes.

1)Rectangle or Square - Represent decision

2)Circles or Ovals – Represent Chance nodes for random or probabilistic events

3)Points or Dots – Represent Leaf nodes.

**21)What type of node is considered *Pure*?**

A pure node is a leaf node in which all the instances of datapoints belong to only one class or type. They represent an unambiguous prediction.

**22)How would you deal with an *Overfitted Decision Tree*?**

We would analyze the different features on which the decision tree depends. We can then deduce the importance of each feature and we can remove the less important features i.e. we would remove the nodes/branches that do not contribute much to the accuracy of the model. This would help decrease the complexity of the decision tree and thereby prevent overfitting. This method is called pruning

**23)What are some *disadvantages* of using Decision Trees and how would you solve them?**

A big disadvantage of decision trees is that they tend to easily become quite complicated. They are easily overfitted.

Another major disadvantage of decision tree is that it is biased towards the majority class.

**24)What is *Gini Index* and how is it used in Decision Trees?**

The Gini index, also known as Gini impurity. In other words, it measures the degree of impurity or randomness in a set of categorical values.

The Gini Index can be calculated by the formula

Gini Index = 1 – sigma(P(i)^2)

In decision trees, the algorithm chooses the feature that produces the smallest Gini index as the best split.

**25)How would you define the *Stopping Criteria* for decision trees?**

Stopping criteria for decision trees refer to the conditions that determine when the tree-building process should stop. These criteria help prevent overfitting. Some common stopping Criteria are:

1)Minimum impurity decrease

2)Limit on number of leaf nodes

3)Setting maximum depth of tree

**26)What is *Entropy*?**

Entropy is a measure of randomness of a system.

It is defined as H(S) = - Σ (pi \* log2(pi)), pi is the probability of an item belonging to a specific class.

For pure data(only one class), entropy is zero while it is maximum when equal number of items are present in each class.

**27)How do we *measure* the Information?**

The number of possible outcomes or choices that exist for that message will decide the amount of information conveyed by a message.

The formula for information gain is

IG = H(S) - Σ (|Sv| / |S| \* H(Sv))

where IG is the information gain, H(S) is the entropy of the parent node, |Sv| is the number of samples in the subset Sv resulting from splitting the data on the feature, and H(Sv) is the entropy of the subset Sv.

The higher the information gain, the more information that feature provides for the clarification task.

**28)What is the difference between *Post-pruning* and *Pre-pruning*?**

**Pruning** is of two types:

Post-pruning (Backward pruning): Full tree is generated and then the non-significant branches are pruned/removed.

Pre-pruning (Forward pruning): This approach stops the non-significant branches from generating.

**29)Compare *Linear Regression* and *Decision Trees***

Linear regression assumes a linear relationship between the input features and the output variable, whereas decision trees can model nonlinear relationships. Linear regressions are relatively simple. Decision trees can also provide feature importance rankings.

**30)What is the relationship between *Information Gain* and *Information Gain Ratio?***

Information gain and information gain ratio are two metrics used in decision trees to evaluate the quality of a split

The formula for information gain is

IG = H(S) - Σ (|Sv| / |S| \* H(Sv))

where IG is the information gain, H(S) is the entropy of the parent node, |Sv| is the number of samples in the subset Sv resulting from splitting the data on the feature, and H(Sv) is the entropy of the subset Sv.

Information gain measures the reduction in entropy achieved by splitting the data on a particular feature, while information gain ratio adjusts the information gain by taking into account the intrinsic information of the feature itself.

**31)Compare *Decision Trees* and *k-Nearest Neighbours***

Decision trees are a type of tree-based model that partitions the feature space into regions based on the values of the input features. k-NN, on the other hand, is a distance-based model that classifies a new sample based on the class labels of the k-nearest neighbors in the training data.

There is a lot of computational work in KNN so it can take more time to train than decision trees and KNN requires longer prediction times

Decision trees can handle noisy data efficiently than Knn.

**32)While building Decision Tree how do you choose which attribute to *split* at each node?**

The performance of a decision tree model depends on the choice of the attribute that is split at each node. There are several methods to select the splitting attribute:

1)Information Gain - The formula for information gain is

IG = H(S) - Σ (|Sv| / |S| \* H(Sv))

where IG is the information gain, H(S) is the entropy of the parent node, |Sv| is the number of samples in the subset Sv resulting from splitting the data on the feature, and H(Sv) is the entropy of the subset Sv

It measures the reduction in entropy achieved by splitting the data based ont hat attribute

2)Gain ratio – It is a variation o. The attribute that has the highest information gain is selected as the splitting criterion.f the Information Gain method. Information gain measures the reduction in entropy achieved by splitting the data on a particular feature, while information gain ratio adjusts the information gain by taking into account the intrinsic information of the feature itself. . The attribute that has the highest information gain ratio is selected as the splitting criterion.

3)Gini Index - The Gini index, also known as Gini impurity. In other words, it measures the degree of impurity or randomness in a set of categorical values.

The Gini Index can be calculated by the formula

Gini Index = 1 – sigma(P(i)^2)

In decision trees, the algorithm chooses the feature that produces the smallest Gini index as the best split.

4)Chi-square - The independence of the attribute and class are put to the test with the Chi-square algorithm. This works by comparing the expected frequency under the assumption of independence with the observed frequency of each class. The chosen attribute is determined by the splitting criterion of the one with the highest Chi-square value.

**33)How would you compare different *Algorithms* to build *Decision Trees*?**

The different algorithms used to build decision trees are:

1)CART(Classification and Regression Trees) – It uses the Gini Index to select the splitting attribute. It can create binary trees

2)ID3(Iterative Dichotomiser 3) – It will select the attribute that provides the highest information gain at each node without considering future nodes. This algorithm tends to overfit data

3)C4.5 – C4.5 uses information gain ratio to select the splitting attribute, in place of information gain in ID3. C4.5 prunes tree to prevent overfitting.

4) CHAID (Chi-Squared Automatic Interaction Detection) – CHAID uses the Chi-Squared test to select the splitting attribute, which measures association between attribute and the target variables. It works only for categorical values and cannot handle continuous variables

**34)How do you *Gradient Boost* decision trees?**

Gradient Boost is an ensemble method. Ensemble methods are methods in which multiple decision trees are combined to obtain a more accurate model. Decision trees are sequentially added to the model, and each decision tree is trained to compensate for the mistakes of the previous trees.

The steps involved in the Gradient Boosting process are:

1)Start with a decision tree.

2)Find errors in the decision tree

3)Train another decision tree which will predict the errors in the first decision tree

4)Update the model

5)Repeat the process to improve performance of model

By following these steps, we will obtain a gradient boost decision tree.

**35)What are the differences between *Decision Trees* and *Neural Networks*?**

Decision trees partition the feature space based on certain decisions.

Neural networks are a deep learning model which are trained to extract features and identify patterns in the data. They consist of a layer of interconnected nodes. Neural networks require large amounts of training data.

On the other hand, decision trees can be trained quickly. Decision trees are easy to understand while neural networks are complex.

Decision trees can easily avoid overfitting while neural networks are prone to overfitting.