MACHINE LEARNING

1.A) GridSearchCV()

2.A) Random forest

3.B) The regularization will decrease

4.A) It regularizes the decision tree by limiting the maximum depth up to which a tree can be grown.

5.A) It's an ensemble of weak learners.

C) In case of classification problem, the prediction is made by taking mode of the class labels

6.C) Both of them

7.C)both bias and variance increase

8.B) model is overfitting

9.The Gini index of a dataset is calculated as 1 - (pA^2 + pB^2), where pA and pB are the proportions of class A and class B, respectively. For a dataset with 40% class A and 60% class B, the Gini index would be 1 - (0.4^2 + 0.6^2) = 0.48.

The entropy of a dataset is calculated as -pA * log2(pA) - pB * log2(pB), where pA and pB are the proportions of class A and class B, respectively. For a dataset with 40% class A and 60% class B, the entropy would be -0.4 * log2(0.4) - 0.6 * log2(0.6) = 0.97.

10.There are several advantages of Random Forests over Decision Trees:

i)Improved accuracy: Random Forests generally have higher accuracy than individual Decision Trees because they reduce overfitting by averaging the results of multiple Decision Trees.

ii)Reduced variance: Random Forests also have less variance than individual Decision Trees because they are built using a random subset of features at each split, which reduces the chances of the model being affected by noise in the data.

iii)Better handling of missing values: Random Forests are better equipped to handle missing values because they can still make predictions even if one feature is missing.

iv)Better handling of categorical variables: Random Forests are able to handle categorical variables better as it can convert them into numerical variables

v)Feature importance: Random forests can also be used to determine feature importance.

vi)Robustness: Random forests are robust to outliers and noise in the data.

vii)Good for large datasets: Random forests are good for large datasets as it can handle large number of variables and data instances.

11.Scaling numerical features in a dataset is important because some machine learning algorithms, such as those that use distance measures, are sensitive to the scale of the input features. Scaling the features can help ensure that all features are given equal consideration in the model.

Two techniques that are commonly used for scaling numerical features are:

Min-Max Scaling: It scales the data between 0 and 1.

Standardization: It scales the data so that it has a mean of 0 and a standard deviation of 1.

12.Scaling numerical features in a dataset can provide several advantages when using optimization algorithms such as gradient descent:

Faster convergence: Scaling the features can help the optimization algorithm converge faster by ensuring that the features are on a similar scale. This can prevent one feature from dominating the optimization process.

Stable convergence: Scaling the features can help ensure that the optimization algorithm is not affected by large variations in the scale of the features. This can make the optimization process more stable.

Better generalization: Scaling the features can help the optimization algorithm generalize better to new data by ensuring that the features are on a similar scale. This can make the model more robust to changes in the data.

Better performance: Scaling the features can help improve the performance of the model by ensuring that the features are on a similar scale. This can make the model more sensitive to important features, and less sensitive to noise in the data.

Better handling of missing values: Scaling the features can help to handle missing values in data by ensuring that the features are on a similar scale. This can make the model more robust to missing values in the data.

13.Accuracy is not a good metric to measure the performance of a model in case of a highly imbalanced dataset for a classification problem. This is because accuracy is calculated by dividing the number of correct predictions by the total number of predictions. In a highly imbalanced dataset, where one class has significantly more samples than the other, the model can achieve high accuracy by simply predicting the majority class for all samples, regardless of the actual class.

For example, if a dataset has 90% of samples belonging to class A and 10% of samples belonging to class B, a model that always predicts class A will achieve 90% accuracy, even though it is not making any correct predictions for class B.

In such cases, other metrics such as precision, recall, F1-score, AUC-ROC, G-mean, are more appropriate to evaluate the performance of the model. These metrics take into account the imbalance in the dataset and provide a more comprehensive evaluation of the model's performance.

14.F-score, also known as F1-score, is a metric that combines precision and recall to give a single score that balances both aspects of a model's performance. It is commonly used in binary classification problems where the imbalance is present.

The mathematical formula for F1-score is:

F1-score = 2 * (Precision * Recall) / (Precision + Recall)

where:

Precision = (True Positives) / (True Positives + False Positives)

Recall = (True Positives) / (True Positives + False Negatives)

The value of F1-score ranges from 0 to 1, where 1 represents a perfect score and 0 represents the worst possible score.

F1-score is useful when the cost of false negatives and false positives are high, this means that the metric is sensitive to misclassification. If you want to put more weight on precision, the F-beta score can be used, where the beta parameter can be adjusted to the desired weight.

15.In the context of machine learning, "fit", "transform", and "fit_transform" are methods that are typically used on an instance of a transformer or an estimator.

"fit" method is used to estimate the parameters of a model, such as the mean and standard deviation for standardization or the min and max values for min-max scaling. It is used on the training data to learn the parameters, which can then be used to transform new data.

"transform" method is used to apply the transformation on a dataset, using the estimated parameters learned by the "fit" method. It is used on both the training and test data.

"fit_transform" method is a convenience method that combines the fit and transform methods together. It first estimates the parameters of a model on the training data and then applies the transformation on the same training data.

For example, when applying StandardScaler on a dataset, we can use the following:

```
scaler = StandardScaler()

scaler.fit(X_train)

X_train = scaler.transform(X_train)

X_test = scaler.transform(X_test)
```

It's important to note that when using the fit_transform method, the same parameters learned from the training data should be applied to the test data by using the transform method, otherwise the model will be overfitted.

WORKSHEET 7 SQL

1.B. Candidate keys

2.B. Primary keys cannot contain NULL values…

C. A table can have only one primary key with single or multiple fields….

3.C. Insert

4.C. ORDERBY

5.C. SELECT

6.C. 3NF

7.C. All of the above can be done by SQL

8.B. DML

9.B. Table

10.B. 2 NF

11.SQL join statements allow us to access information from two or more tables at once. They also keep our database normalized. Normalization allows us to keep data redundancy low so that we can decrease the amount of data anomalies in our application when we delete or update a record.

12.There are four different types of join operations:

(INNER) JOIN: Returns dataset that have matching values in both tables

LEFT (OUTER) JOIN: Returns all records from the left table and matched records from the right

RIGHT (OUTER) JOIN: Returns all records from the right table and the matched records from the left

FULL (OUTER) JOIN: Returns all records when there is a match in either the left table or right table

13.SQL Server is a relational database management system (RDBMS) developed by Microsoft. It is a software product that is used to manage and store data in a structured way, allowing users to easily retrieve, update, and manipulate the data. SQL Server supports the use of the SQL (Structured Query Language) language, which is used to interact with the database and perform operations such as inserting, updating, and retrieving data.

SQL Server is commonly used in enterprise environments for a variety of tasks, such as data warehousing, business intelligence, and online transaction processing. It can run on a variety of platforms, including Windows, Linux and Docker. It also offers a variety of features such as data encryption, high availability, and disaster recovery options.SQL Server also provides a variety of tools and technologies for working with data, including the SQL Server

Management Studio for managing the database, and SQL Server Reporting Services for generating reports.

SQL Server comes in different editions, such as Express, Web, Standard and Enterprise, each edition is designed to meet different needs and requirements, depending on the size of the organization, the number of users and the complexity of the environment.

14.A primary key is a special relational database table column (or combination of columns) designated to uniquely identify each table record.

A primary key is used as a unique identifier to quickly parse data within the table. A table cannot have more than one primary key.

A primary key's main features are:

It must contain a unique value for each row of data.

It cannot contain null values.

Every row must have a primary key value.

A primary key might use one or more fields already present in the underlying data model, or a specific extra field can be created to be the primary key.

15.ETL is a process that extracts the data from different source systems, then transforms the data (like applying calculations, concatenations, etc.) and finally loads the data into the Data Warehouse system. Full form of ETL is Extract, Transform and Load.

STATISTICS WORKSHEET-7

1.b) 0.135

2.a) 0.67

3.c) 0.745

4.b) 0.577

5.c) 0.6

6.a) 0.33

7.a) 0.45

8.b) 0.22

9.d) 0.56

10.a) 0.33

11.a) 0.33

12.c) 0.78

13.d) 0.25

14.c) 0.24

15.c) 1/2