Day 1:

1)Machine Learning is a subset of Artificial Intelligence which allows applications to predict accurate results after learning from the data without human interventions.

2)Application of ML: Image Recognition, Self-driving cars, Product Recommendations, etc.

3)Types of machine learning :

a)Supervised Learning: Here the algorithms are trained using data that are well labelled.Eg: Linear Regression,SVM,etc.

b)Unsupervised learning: In this, the algorithms are trained against unlabelled data. Two categories of unsupervised learning are Clustering and Association.

c)Semi-supervised learning:The algorithms are trained on both labelled and unlabelled data. Approaches used are graph-based methods, low-density separation, Heuristic approaches, etc.

d)Reinforcement Learning: In this, the decisions are made using trial and error methods to obtain the highest accuracy i.e output depends on the current state of input. Important approaches are Monte Carlo Methods, Temporal Difference methods, etc.

4)Scikit-Learn: It is the most useful library for machine learning which contains a lot of efficient tools for machine learning and statistical modelling.It is built on numpy,scipy and matplotlib.

Day 2: NumPy i.e. Numerical Python is a python library useful for working on arrays faster than lists. These are the concepts I learned today:

1) Install numpy and import under alias np.

2) Create an ndarray object using array() function. We can create 0-D,1-D,2-D,etc arrays depending on values passed. Eg: a0=np.array(25) is a 0-D array while a1=np.array([1,3,4]) is a 1-D array. The number of dimensions can be mentioned using the ndmin argument.

3) Accessing the elements: Elements can be accessed using their indexes, in case of multidimensional arrays we can use comma separated integers representing the dimensions and the index of the element to access the elements. Negative indexing helps to access an array from the end.

4) Data Types :NumPy has extra data types which are referred with 1 character.
i.e. i - integer, b - boolean,u - unsigned integer, f - float, o - object,etc.

5) Worked on matrix:

a)Create a matrix

b) Count number of rows & columns of matrix

c) Return the minimum & maximum elements of matrix, also min & max elements of each row & each column.

d)Reshape: add / remove dimensions / change no. of elements in each dimension.

e) find transpose, diagonal values, dot product.

f) Addition, subtraction, average, multiplication(element-wise and row-column wise) on elements of matrix.

g) Make all elements of the matrix 0 using np.zeros() / 1 using np.ones().

h) Generate a matrix of random values using the random module.

Day3:
Pandas is generally used for data analytics. It allows importing data from various file formats such as: JSON, CSV, Excel, etc.

1)Features: Data manipulation, reading and writing data,reshaping of data, data filtration, data cleaning.

2) Install panda using pip and import as pd

3)Data Structures:

a)Series: It's a 1-D labelled array that can hold data of any type. The various inputs can be of types array, dictionary or scalar.

b) Dataframe: It's a 2-D data structure i.e data is aligned in tabular fashion in rows and columns. It can be created using various inputs like lists, series, dict,another dataframe.

c)Panel: It's a 3-D container of data.

4) Used the titanic.csv dataset and practised the following methods & attributes:

a)head()- It returns the first n rows. The default no. of rows is 5. The tail() returns last n rows.

b)Shape- Returns a tuple having no.of rows and columns.

c)info()- It gives a summary of DataFrames including non-null values, index dtype, column dtype,etc.

d)DataFrames.columns- It returns the column labels

e)Index.value_counts()- It counts the number of distinct values in the given index.

f) Index.unique() - Gives the unique values in the count.

g) Access columns :DataFrame [column_name]
h) Access rows :
1. DataFrame.iloc[ ] : index mentioned should be in the integer form.
2. DataFrame.loc[ ] : index can be in numeric or even the name of column

i) Filtering of data and sorting of data.

j) Grouping and aggregation : DataFrame.groupby() - It splits the data into groups according to the condition mentioned. Also performed mean(), count() and size() methods on the groups obtained.

Day 4: Today's topic was Linear Regression

****

Day 4: Today's topic was an overview of Linear Regression.
Linear Regression is a Machine Learning model based on Supervised Learning. It is mainly used for establishing a relationship between a dependent variable and one or more independent variables and then predicting values of the dependent variable.
2 types of Linear Regression Model :
Simple Regression Model and Multiple Regression Model.

1)Simple Regression Model:-

2 main objectives of simple regression model :
a)Establish a relationship between two variables(Specifically establish if there is a statistically significant relation between the two.)
b)Forecast new observations.

2 roles of variables:

1. Dependent variable: It's the variable whose value we want to forecast. We can denote it as y
2. Independent variable: variable that explains the other one. These values are independent. Can be denoted as x.

To implement simple linear regression in python, we have to do the following steps:
1.Get actual X and Y data and create a dataframe.
2.Import linear model from sklearn.
3.Use the function linear_model.LinearRegression() and then predict y by using the model build.
4.Compare the actual values and the predicted values.
5.Input new value of X and check the predicted value.

2)Multiple Linear Regression : In this regression model, the dependent values can be predicted based on two or more values of independent or predictor variables.

Day 5 -Summary :

Multiple Linear Regression is used to find the relationship between a dependent variable and several independent variables. It is a broader class of regressions which encompases both linear and nonlinear regressions.

Advantages :

1.It can determine the relative effect of one or more predictors.

2.It can identify the anomalies and outliers .

Disadvantage: It will fit the data with linear relation even if the data is not linearly distributed.

Steps to implement multiple regression :-

a)Use the pandas module to load the dataset.

b)Create a list of independent values and denote it as X and pass the dependent value into variable y.

c)Import linear_model from sklearn. Create an object using the LinearRegression() method from sklearn module.

d)This object has the method fit() which takes the dependent and independent values as parameters and fills objects with the data which describes the relationship.

e) Now the object can predict the dependent value using predict() method.

The difference between simple and multiple is that in multiple regression models we need to pass the x_train and x-test model in the form of a 2-D array in which the rows represent the record and the column will represent the independent variables. In case of 3 independent variables the column of x_train and x_test will have 3 columns in each.

Day 6 : Overview of Polynomial Regression

Polynomial Regression is a special case of linear regression where we can fit a polynomial equation on the data with a curvilinear relationship between target variable and independent variable. It uses the relationship between variables x and y to find the best way to draw a line through the data points.

Advantages :

1)The model can fit nonlinear data better than the linear regression.

2)It can basically fit a wide range of curvature.

Disadvantages :

1)Too sensitive to outliers i.e. even presence of one or two outliers can seriously affect the result.

2)Increase in the number of degree increases the chance of overfitting.

Steps to implement polynomial regression :

1)Load the dataset using pd.read_csv() method and import the linear regression model from sklearn.

2)Import Polynomial features from sklearn. Create an object of PolynomialFeatures() and pass the hyperparameter i.e the degree.

3)Then the fit_transform() method will return the polynomial form of the input provided.

4)Create an object of LinearRegression() and again fit the result obtained above using the linear regression model.

5)Find out the value of the dependent variable using the predict() method.

In polynomial regression, we need to know how to hyper tune its parameters.

Day 7:

Support Vector Regression uses the SVM, a classification algorithm. It tries to best fit the line within the threshold error values. The best model should fit maximum data with minimum error rate and minimum tolerance.

Advantages :

1.It works well with data with higher dimension and clear margin of separation.

2.It supports both linear and nonlinear regressions and is memory efficient.

Disadvantages :

Not suitable for data without clear margin separation, for large datasets and noisy datasets.

Some essential terms :

1.Kernel :

Its function is to take data as input and transform into required form.There are different types of kernel functions such as linear, non-linear, polynomial,RBF,etc.

2.Hyperplane: It is the line that is used to predict continuous values and fit the line accordingly. In SVM, it is a separating line between two data classes.

3.Boundary Lines: These are two parallel lines drawn to both sides of the Support vector with the error threshold value.

4.Support Vector: These are the data points closest to the boundary.

Hyperparameters:

1.Error rate(e) : It is also referred to as epsilon. The distance between model and boundary line is called error rate.

2.Tolerance(ei) : It indicates the tolerable rate for error.

Implementation:

1.From sklearn import SVR.

2.Create an object of SVR and define the value of the kernel.

3.Fit and train the data.

4.Predict the output value for given input.

****

Day 8 :-
A decision tree is a flowchart like structure where each node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label. It can handle both categorical and numerical data.
The tree can be learned by splitting the source set into subsets based on an attribute value test, the derived subsets are again split in a recursive manner. The recursion does not stop until it reaches the node that has the value of the target variable.
The Classification And Regression Tree (CART) is a term used to refer to classification and regression procedures of a decision tree.

Advantages :
Decision trees require less effort for data preparation during preprocessing and do not require scaling of data. Missing values in data does not much affect the process of building a decision tree.

Disadvantages:
They can be computationally expensive and are prone to errors in classification problems with many class and small number of training examples.

Implementation:
1.From sklearn.tree DecisionTreeRegressor class.
2.Create an object of the class and pass random_state(any integer value) as its parameter
3.Train the model and fit the decision tree regressor to the dataset.
4.Predict new value.
****

Day 9:-
A Random Forest is an ensemble technique that performs both regression and classification using multiple decision trees and bagging method.

Ensemble learning: It's a technique which combines the decisions from multiple models to improve the performance i.e model becomes more flexible, less biased and less data sensitive.
Types of ensemble learning:
1.Bagging: It refers to training a bunch of individual models in a parallel way.
2.Boosting: Training a bunch of individual models in a sequential manner.

Random Forests Regression uses bagging technique in which the entire dataset is divided into small datasets which are trained on individual decision trees. Their outcomes are combined using mean,median,mode or any other technique to get the final output.

Advantages:

Random Forests work well with both categorical and continuous variables. It can handle missing values automatically and is robust to outliers.

Disadvantages:
This algorithm requires more computational power and resources and more time for training.

Implementation:
1.Import RandomForestRegressor class from sklearn.ensemble module
2.Create an object of the class with n_estimators and random_state as parameters.
  Here, n_estimators is the number of trees to be used in the forest.
3.Use .fit() method to train the model and predict the value by passing the testing data.

****

Day 10:
Today's topics involved creating a machine learning model for Car price prediction and then deploying it on the web using Flask.
Implementation:
1.Import the numpy and pandas library.
2.Load the dataset which consists of variables name,year,selling price,km_driven,fuel,seller_type,transmission,owner
3.Check info,use iloc to split into dependent & independent variables, use LabelEncoder from sklearn.preprocessing module to normalize
4.Split dataset into training & testing set.
5.Create RandomForestRegressor model & fit training data
6.Check accuracy & encode label of each column, predict output value against testing value.
7.Save the created model using pickle library which enables to save trained models in a file and then restore them for reusing.
8.For creating a flask app. Create a new python file and import flask libraries.
9.Import pickle & read model using pickle.load()
10.Setup routes, use render_template to render html file, build the flask model.
11.Execute the python file and check the prediction price for given inputs on the webpage hosted on the local browser.
****
Day 11:
Logistic Regression is a supervised classification algorithm. Classification problems are those in which the independent variable takes only discrete values for a set of input values. Logistic Regression is a statistical method for predicting binary classes. In this algorithm, a threshold value needs to be set depending on the classification problem. It can be used for binary and multi-class classification problems.
In binary class problems, the sigmoid function is used that converts the outcome of linear regression model into probability & then limits its value between 0 and 1.

The softmax function is used for multiclass logistic problems that convert the outcomes of linear regression models into probability and then convert the class with maximum probability as its outcome.
Advantages:
Easy to implement, training the model doesn't require high computational power and is efficient in case of datasets with features that are linearly separable.
Disadvantages:
It cannot solve non-linear problems and sensitive to outlier because of which it can give incorrect result in presence of data values that deviate from expected range.

For implementation, we need to import the Logistic Regression class from sklearn.linear_model.

Day 12:
The K- Nearest Neighbour is a type of supervised machine learning algorithms. It does not have a specialized training phase and uses all the data for training while classification. It is also a non-parametric learning algorithm as it doesn't assume anything about the underlying training data.

How the KNN algorithm works:
Step-1: Choose a value for K. K is the number of nearest neighbours which is the core deciding factor.
Step-2: Find the closest K data points using methods such as Euclidean distance, Hamming distance, Manhattan distance and Minkowski distance.
Step-3: Among these K neighbours, count the number of data points in each category.
Step-4: Assign the new data point to the category with most neighbours.

Advantages:
Easy to use, quick calculation time, no training required before making predictions.
Disadvantages:
Value of K needs to be chosen carefully as lower value of K results in higher influence of noise & higher value of K might result in overfitting, has a high prediction cost for larger datasets.

Implementation of the KNN algorithm is done using the KNeighboursClassifier class from sklearn.neighbours.

Movie & product recommendation system, stock prediction, text mining, etc are some applications of KNN algorithm.

Day 13:
A Support Vector Machine is a discriminative classifier formally defined by a separable hyperplane. In SVM we take the output of linear function & if that output is greater than 1, we

identify it with one class and if the output is -1, we identify it with another class since the threshold values are changed to 1 and -1.

Steps that explain how it works are:
1) Split the categories using various hyperplanes as much as possible.
2)It takes hyperplane one by one and checks which one is the optimal hyperplane.
3)Tries to find out the hyperplane with a large margin.
4)It works by maximising its margin as much as possible.

Types of kernels used by SVM:
1)Linear Kernel
2)Polynomial Kernel
3)Radial Basis Function(RBF) Kernel

Advantages:
SVM works well when there is a clear margin of separation between classes & is more effective in high dimensional spaces.
Disadvantages:
SVM algorithm is not suitable for large datasets. It does not perform well with noisy dataset & is sensitive to outliers.

Implementation:
1)Import SVC from sklearn.SVM and create an object of the class with the type of kernel mentioned as its parameter.
2)Train the model with training dataset and predict its outcome for testing dataset.

Common Applications of SVM are face detection, image classification, text and hypertext categorization, etc.

****

Day 14:
Learned about different Kernels and about hyperplane in SVM.

Kernels are mathematical functions that take data as input and transform it into the required form. There are several types of SVM kernels each having different functionality.

The hyperplane is an n-1 dimensional subspace for n-dimensional space. The learning of hyperplane in linear SVM is done by transforming the problem using some linear algebra. This is where the kernel plays a role.

In brief, the model implements the kernel function in such a way that it will increase the dimensionality of space first & then try to find the optimal hyperplane which classifies all the data points and then converts back to its original state.

Learned the equation of RBF kernel. It's a function which changes with distances from a location. The function depends on the distance between the landmark & the datapoint & also depends on the parameter that controls the thickness of the landmark. Datapoints which are near to the landmark will be projected on higher elevation while those far from it will be projected lower. If the thickness is increased, more data points are projected on the elevation.

****

Day 15:
Naive Bayes algorithm is a supervised machine learning algorithm for classification based on Baye's theorem with an assumption of assumption among the predictors.
3 Types of Naive Bayes model under the scikit-learn library are:
1)Multi-variate Bernoulli Naive Bayes
2)Multinomial Naive Bayes.
3)Gaussian Naive Bayes.

Advantages:
It is easy and fast to predict a class of test data set. It performs well in multi-class prediction and in case of categorical input variables.

Disadvantage:
1)A limitation is the assumption of independent predictors.
2)If a categorical variable has a category (in the test data set), which was not observed in training data set, then the model will assign a zero probability and will be unable to make a prediction. This is known as "Zero Frequency".

Implementation of the algorithm can be using the GaussianNB class from sklearn.naive_bayes.

Common Applications are real-time prediction, spam filtering, multi-class prediction, etc.

****
****

Day 16:
Today's topic Decision Tree Classification is a continuation of regression with decision tree learnt on day-8.

Decision tree classification is commonly used in the fields of biomedical engineering, medicine, financial analysis, etc.

Implementation:
1)From sklearn.tree DecisionTreeRegressor class.
2)Create an object of the class and pass random_state(any integer value) and criterion = entropy as its parameters.
Here, the criterion is a function to measure the quality of split and entropy will calculate the homogeneity of the sample.
3)Fit the model using the training dataset and predict the outcome against testing data.

Advantages of Classification with Decision Tree:
Easy to interpret for small-sized trees, excludes unimportant features and fast at classifying unknown records.

Disadvantages of Classification with Decision Tree:
Large trees can be difficult to interpret and the models are often biased towards splits on features having large number of levels.

****

Day 17:
Random Forest Classification is a supervised machine learning algorithm based on ensemble learning.

Applications of the algorithms are commonly seen in the fields of banking sector, medicines, stock market,etc.

Working of random forest algorithm:
1)First, start with the selection of random samples from a given dataset.
2)Next, this algorithm will construct a decision tree for every sample. Then it will get the prediction result from every decision tree.
3)Voting will be performed for every predicted result.
4)Select the most voted prediction result as the final prediction result.

Implementation:
1)Import RandomForestClassifier from sklearn.ensemble
2)Create an object and mention the value of n_estimator.
3)Fit the model and predict the outcome against testing data.
****
Day 18:
Mini project on classification algorithms.
Worked on the iris dataset which contains about 3 species of iris.
Steps followed:
1)Import the data and important libraries.
2)Divide the data into dependent and independent variables.

In this case, the independent variables considered were SepalLengthCm, SepalWidthCm, PetalLengthCm, PetalWidthCm and the dependent variable was species.
3)The dependent variable contains categorical values hence they need to be converted into numeric data using LabelBinarizer.
4)Split the dataset into train and test.
5)The RandomForestClassifier is the classification algorithm used here to train the model.
6)Predict the output.
7)Evaluate the model i.e compare the predicted output with the test data using confusion_matrix class and find out the accuracy using accuracy_score

****

Day 19:
Topics: Underfitting and Overfitting.
The goal of a good machine learning model is to generalize well from the training data to any data from the problem domain. To achieve a good model, we need to overcome the two main deficiencies of ML algorithms which are underfitting and overfitting.

1)Underfitting:
An ML algorithm is said to be underfitting when it cannot capture the underlying trend of data. If the training accuracy and validation accuracy is less then the model is understood to be underfitting.
To reduce underfitting:
1.Increase model complexity
2.Increase the number of features.
3. Remove noise from the data

2)Overfitting:
Overfitting occurs when a model even learns the details and even noise in the training data to such an extent that it negatively impacts the performance of model on new data. In this case, the training accuracy is high but the testing accuracy is less.

To reduce overfitting:
1.Increase the training data.
2.Reduce model complexity.
3.Use of regularization techniques such as Ridge regularization and Lasso regularization.

****

Day 20:

In a machine learning problem, normally when we divide the dataset into training data and testing data for training the model and evaluating its performance respectively, variance problem can occur i.e a situation when the accuracy obtained on one test set is much different from the accuracy obtained on another test. A solution to such problems is using the K-Fold Cross Validation.

In K-Fold Cross Validation, the data is divided into K folds. Out of the K folds, K-1 sets are used for training while the remaining set is used for testing. The algorithm is trained and tested K times i.e. each time a new set is used as a testing set the remaining sets are used for training. Finally, the result of the K-Fold Cross-Validation is the average of the results obtained on each set. It can be implemented using cross_val_score from sklearn.model_selection.

Grid Search is the process of scanning the data to configure optimal parameters for a given model. It can be implemented using scikit-learn's GridsearchCV. We need to mention the hyperparameters and the values that we want to try out & then it will evaluate all possible combinations of hyperparameter values using cross-validation.

****

Day 21:
1) Regression vs Classification:
The most significant difference between regression and classification is that regression helps predict a continuous quantity while classification predicts discrete labels.

2)Unsupervised Learning:
i) Clustering:
It is the task of dividing data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups.

ii) Association:
It is a rule-based ML technique that finds important relations between variables or features in a dataset.

3)Reinforcement Learning:
It is the training of ML models to make sequence of decisions. The decisions are made using trial and error methods to obtain the highest accuracy i.e output depends on the current state of the input.

4)Learned the deployment of ML web applications on Heroku. Heroku is a platform that enables developers to build, run and operate applications entirely in the cloud.