***Flight Price Prediction using sagemaker***

* 1. Sagemaker :- Amazone sagemaker is a fully managed machine learning service. with sagemaker , data scientist and developers can quickly and confidently build and train, deploy ML Models into a production-ready hosted environment. It provides a UI Experience for running ml workflows that makes Sagemaker ML tools available across multiple integrated development environments(IDEs).
* Advantages:-
  + It is fully managed service
  + it provides wide range of libraries for ML workflows
  + it provide wide range of algorithms for various problems
    - Regression
    - classification
    - object detection
    - etc
  + One-click Deployment
  + it integrate with other AWS services
  + It provides notebooks and IDE for development
    - Classic Jupiter
    - Jupiter lab
    - Sagemaker Studio
  + Detailed documentation
  + It has huge community and online support.
* AWS Sagemaker Services:-
  + IAM:- It stands for Identity & access management.
    - The AWS IAM is used for managing our account in AWS though which we can create multiple users , groups and allocate the AWS resources separately so that any misuse of our account is not able to do by users and groups
    - Sagemaker IAM provide us 3 level security:-
      * IAM GROUP
      * IAM USER
      * IAM Role
  + EC2:- it stands for elastic cloud computer. it provide us cloud based computer with having the configuration that we need to get. like we want a system having 16 rams, with any Operating system and some storage then we can get it on EC2.
  + S3:- It stands for simple storage service. In this S3 bucket we store our large or any size of files we can store these files here.
* SDK:- A Software development kit is a set of platform-specific building tools for developers. you require components like debuggers, compilers, and libraries to create code that runs on a specific platform, operating system, or programming language . SDK’s put everything you need to develop and run software in one place. Additionally, they contain resources like documentation , tutorials, and guides as well as API and frameworks for faster application development.
  + Advantages:-
    - Efficient development
    - Reduce time complexity
    - Seamless Integration
    - Reduce Costs
  + AWS provide a separate SKD for Python specially called BOTO3 .
* Data cleaning & Github:-
  + Read about:- Dictionary comprehension and Dictionary unpacking.
* EDA:- It’s the practice of exploring your dataset by utilizing visualization tools and statistical measurements to understand and extract underlying patterns and information within your dataset.
  + This gives you clear picture of the data, helps us make data-informed decisions and solve crucial problems with much fewer assumptions and more facts.
  + This step is the backbone of any Data Science project and takes up a major chunk of the project timeline.
  + Statistic is used for summarization and making inference from our data by using concept like Central Tendency , Dispersion , spread, Hypothesis testing , Plots, and graphs.
  + Strength of Association:-
    - Pearson’s Correlation:- The Pearson correlation is used to know about the relationship between two numeric values, but it assumes our values are linearly separable
    - Spearman’s Rank Correlation:- The Spearman’s Rank Correlation is used to know about the relationship between two numeric variables but it assumes that our values have non-linear separation. In Today’s world our most of data have non-linear separation, so it is good to know about this Spearman’s Rank Correlation.
    - Cramer’s V :- The Cramer’s V is used to know correlation between two categorical variables.it will take all unique categories form both variables and then it create a frequency table and it run chi-square test on this table and it used the test stats value and then we calculate the Cramer’s V values. The values is in between (0 to 1), Higher the value higher the relationship between two categorical variables.
  + Hypothesis Testing:-
    - Test for Normality:- This test is used to get information about our numeric variables are following Normal Distribution or not. Here some tests are:-
      * Shapiro-Wilk Test:-
      * Anderson-Darling Test:- This test is quite more powerful than other tests for Normality Test.
    - Test For Association:- This test is used to get the information about the correlation that we have is genuine or not. Here are the test:-
      * Pearson’s Test
      * Spearman’s Test
      * Categorical Variables:-
        + Chi-Square Test
      * Numeric - Categorical Variables:-
        + One-way ANOVA Test
        + Kruskal-Wallis Test
    - Steps Involved:-
      * Sate the Hypothesis
        + Null Hypothesis
        + Alternate Hypothesis
      * Determine significance level (alpha ~5%)
      * Determine which test to perform
      * Collect necessary data (sample)
      * Obtain Critical values
      * Compute Test Statistic (and p-value)
      * Compare:-
        + Significance level vs p-value ; or
        + Critical value vs Test Statistic
      * State conclusions:- Wheater we need to accept the null hypothesis or fail to reject the null hypothesis.
  + For High level information about outliers:-
    - Isolation Forest:- Basically, it is ensemble algorithm like RandomForest . In this , at each first model numeric features and it will randomly pick one feature then randomly pick one value in that feature and then make a split on this value and same thing work with all trees
      * if a datapoint is an outlier then that point it will be capture at the top node.
      * now these steps are performed at each node
* Missingno library :- there are some built-in functions for getting information about missing value and it generates plots , heatmap, dendrogram etc
* Feature Engineering:-
  + Airline column:-In this column we already observed that there are more than categories which has occurred in this column very low frequency , so it good to combine all these into one called other by RareLabelEncoder
  + In RareLabelEncoder there are few parameters we use:-
    - tol:- it mean after which percentage we can combine the categories into one. for example, it we have 5 categories who occurred less than this it will group them into one.
    - replace\_with:- this parameter is used to define what named we should provide for grouped data
    - n\_categorical:- it represent after how many columns it will start grouping it together.
  + In this column there are categorical values, so we need to encode this into number by using OneHotEncoder and we use few parameters are:-
    - sparse\_output=False:- it means it can’t give spare value
    - handle\_unknown=’ignore’:- it will ignore that values which is new for the OneHotEncode in validation, testing set and by default it will generate an error but by using this it will create a new column as replace all unknown with 0.
  + In this column we might impute some datapoints, so we use SimpleImputer.
* date\_of\_journey column:- In this column there are information of date, year, week etc. so we need to extract all this and make a separate column . so, for this purpose we are going to use DateTimeFeatures form feature\_engine library.
  + DateTimeFeatures parameters are:-
    - features\_to\_extract:- in this we can pass the list of features that we want to extract form this date time column
    - yearfirst=True :- it mean it place the year first and does not extract this.
    - format=’mixed’:- it mean our datetime column have mixed values.
  + Now we need to scale the column in similar scale for better training of the data by using MinMaxScaler().
* Source&deatination:-
  + we are performing few operations on these columns together and operations are:-
    - First group all the rare items
    - Perform Mean Encoder:- it will calculate the mean for each category with target variable and encode them and for this we are using MeanEncoder form feature\_engine library.
    - now we need to scale down this value and we are using
      * Power Transformer:- It work like it transform the values in such a way with that our transformed variables are symmetric as possible.
      * for this we are using this from sklearn
    - Now we create a new feature called is\_north or not , it means that the city is belong to north or south.
* Departure Time/ Arrival time:- In this column we are going to perform same operations as source and destination, but we fetch different things. we are going to perform operations like :-
  + Date time features extraction
  + MinMax scaling
  + Part of Day
  + count encoding
  + MinMax scaling
* Duration:
  + Capping by Quantiles( Winsorizer)
  + Imputation
    - Duration categories
    - Ordinal encoding(ordinal column; specify categories)
    - RBF Percentile Similarity (RBF Kernel):- For this purpose we create our custom class and convert into the transformer
    - Power Transformer
    - compute the rbf(gaussian) kernel between x and y
    - K(x,y)=exp(-gamma ||x-y||^2)
    - for each pair of rows x in X and y in Y
* Total\_stops:-
  + we are creating a new feature with this operation. we are converting flight into direct flight if it have 0 total-stops and else we consider it as non-direct flight.
  + after this we handle the missing value and use Function Transformer
* additional\_info:-
  + we are grouping the rare categories into one category
  + we are performing OneHotEncoder for encoding the value into numbers
  + we are making a new column from this by operation in which if it have any information make it 1 and if it have no information make it as 0.
  + Now imputing the values by SimpleImputer
* Feature Selection:-
  + for feature selection we are using the RandomForestRegressor algorithm, and we pass our all feature in this .
  + after that we are using SelectBySingleFeaturePerfromance from feature-engine library for selecting the feature by calculating the r-square vale for each feature and the feature which have r-squared vale larger than 0.1 threshold then we select that feature else we reject it . The threshold value and the scoring value is dependent on us .
* Model Development and Deployment:-