Capstone Project

Bike Sharing Demand Prediction

(Prayag Raj Dubey)
Data science students
Cohort- Hudson, Alma Better

Introduction

In a span of a few decades, the sharing of bicycle systems has seen enamors growth. This system is a recently developed transportation system that provides people with bicycles for common use. The bicycle system provides the user to rent a bike from one docking station, where the user can ride and then return to another docking station. Bike-sharing systems are blooming across more than 1000 cities around the world, particularly in big or large cities like New York City, Paris, Washington DC, London, Beijing, and Barcelona.

One of the major bike-sharing systems is called **Ddareungi bike-sharing system in South Korea**, which started in the year **2015**, known as the **Seoul bike** in English. It was started to overcome issues like greater oil prices, congestion in traffic, and pollution in the environment and to develop a healthy environment for the citizen of Seoul to live in.

The data are downloaded from the South Korean website named SEOUL OPEN DATA PLAZA. **One-year data** are used in this research. The period of the dataset is **365 days (12 months) from December 2017 to November 2018**. From the data, the count of the rental **bikes rented at each hour** is calculated.

Objective

The main objective is to make a predictive model, which could help them in predicting the bike demands proactively. This will help them in a stable supply of bikes wherever needed.

1. Problem Statement

Currently, Rental bikes are introduced in many urban cities for the enhancement of mobility comfort. It is important to make the rental bike available and accessible to the public at the right time as it lessens the waiting time. Eventually, providing the city with a stable supply of rental bikes becomes a major concern. The crucial part is the prediction of the bike count required at each hour for the stable supply of rental bikes.

The given data is as follows:

- **SEOUL BIKE SHARING DEMAND DATASET**

ATTRIBUTE INFORMATION

• **Date:** year-month-day

• **Rented Bike count -** Count of bikes rented at each hour

• **Hour** - Hour of the day

• **Temperature**-Temperature in Celsius

• **Humidity** - %

• **Wind speed** - m/s

• **Visibility** - 10m

• **Dew point temperature** – Celsius

• **Solar radiation** - MJ/m2

• **Rainfall** – mm

• **Snowfall** – cm

• **Seasons** - Winter, Spring, Summer, Autumn

• **Holiday** - Holiday/No holiday

• **Functional Day** –Yes/No

2. Steps involved

- **Loading and discovering data**
  Now, we need to load our data from the external source, which in this case is uploaded to the drive. The data is in the format of the CSV (Comma Separated Values) file.
- **Data Cleaning**
  Data cleaning is an important step in the data analytics process in which you either remove or update information that is incomplete or improperly formatted.
- **Null values Treatment by different methods**
  We have our dataset in hand which is raw and unfiltered. As this step involves cleaning our data first by eliminating the columns which are not needed for our analysis. We have around **8760 rows × 14 columns** in our dataset.Since, there were no null values, we have left it untouched.

- **Exploratory Data Analysis**
  Exploratory Data Analysis is the approach of analyzing data, gathering and summarizing the important characteristics of the information, and using simple visualization that makes it easier to understand.

- **Importing necessary modules and libraries**

  We are importing the following libraries for their respective applications:

  Pandas:- Pandas is used to analyze data. It has functions for analyzing, cleaning, exploring, and manipulating data.

  Matplotlib:- Matplotlib is a graph plotting library in python that serves as a visualization utility. Most of the Matplotlib utilities lie under the pyplot submodule.

  Numpy:- NumPy is a Python library used for working with arrays. It also has functions for working in the domain of linear algebra, Fourier transform, and matrices.

  SciKit:- Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

Plotly:- The plotly is an interactive, open-source plotting library that supports over 40 unique chart types covering a wide range of statistical, financial, geographic, scientific, and 3-dimensional use-cases.

Seaborn:- Seaborn is a library that uses Matplotlib underneath to plot graphs. It will be used to visualize random distributions.

Datetime:- The Datetime module supplies classes for manipulating dates and times. While date and time arithmetic is supported, the focus of the implementation is on efficient attribute extraction for output formatting and manipulation.

Statsmodels:- Statsmodels is a Python module that provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests, and statistical data exploration.

- **Plotting various graphs for different parameters.**
- **Finding the key facts and relationships between various parameters.**
- **Observations according to the outputs of the graph visualizations.**

3. Data information

Bike Sharing Demand Prediction:

Statistical information contained in the Bike sharing demand prediction Database is based on reports from a variety of open media sources. Information is not added to the database unless and until we have determined the sources are credible.

Characteristics of the Dataset:

- Contains information on of Bike sharing system from nov. 2017 to Nov. 2018.
- IT has over 8000 records.
- Includes information on hourly basis i.e. 24 hours of data each day.
- Includes information on 14 variables which includes weather related columns.
- The data belongs to a bike rental system that is located in Seoul, South Korea.

Multicollinearity:

Multicollinearity is the occurrence of high intercorrelations among two or more independent variables in a multiple regression model. Multicollinearity can lead to skewed or misleading results when a researcher or analyst attempts to determine how well each independent variable can be used most effectively to predict or understand the dependent variable in a statistical model.

VIF(Variable Inflation Factors):-

VIF determines the strength of the correlation between the independent variables. It is predicted by taking a variable and regressing it against every other variable.

Outlier Detection:-

An outlier detection technique (ODT) is used to detect anomalous observations/samples that do not fit the typical/normal statistical distribution of a dataset. Simple methods for outlier detection use statistical tools, such as boxplot and Z-score, on each individual feature of the dataset.

Feature Engineering:-

Feature engineering is a machine learning technique that leverages data to create new variables that aren't in the training set. It can produce new features for both supervised and unsupervised learning, with the goal of simplifying and speeding up data transformations while also enhancing model accuracy.

Train and Test Split:-

The train-test split is used to estimate the performance of machine learning algorithms that are applicable for prediction-based Algorithms/Applications.

Optimization:-

Function optimization is the problem of finding the set of inputs to a target objective function that result in the minimum or maximum of the function.
It can be a challenging problem as the function may have tens, hundreds, thousands, or even millions of inputs, and the structure of the function is unknown, and often non-differentiable and noisy.

Regression:-

Regression is a supervised learning technique which helps in finding the correlation between variables and enables us to predict the continuous output variable based on the one or more predictor variables.

Linear Regression:-

Linear Regression is a supervised machine learning algorithm where the predicted output is continuous and has a constant slope. It’s used to predict values within a continuous range, rather than trying to classify them into categories. There are two main types:

Simple regression:

$$y=mx+b$$

Simple linear regression uses traditional slope-intercept form, where m and b are the variables our algorithm will try to “learn” to produce the most accurate predictions. x represents our input data and y represents our prediction.

Multivariable regression:

$$f(x,y,z)=w1x+w2y+w3z$$

A more complex, multi-variable linear equation might look like this, where w represents the coefficients, or weights, our model will try to learn.

Ridge Regression:-

In Ridge regression, we add a penalty term which is equal to the square of the coefficient. The L2 term is equal to the square of the magnitude of the coefficients. We also add a coefficient lambda to control that penalty term. In this case if lambda is zero then the equation is the basic OLS else if lambda > 0 then it will add a constraint to the coefficient. As we increase the value of lambda this constraint causes the value of the coefficient to tend towards zero. This leads to tradeoff of higher bias (dependencies on certain coefficients tend to be 0 and on certain coefficients tend to be very large, making the model less flexible) for lower variance.

Lasso Regression:-

Lasso regression stands for Least Absolute Shrinkage and Selection Operator. It adds penalty term to the cost function. This term is the absolute sum of the coefficients. As the value of coefficients increases from 0 this term penalizes, cause model, to decrease the value of coefficients in order to reduce loss. The difference between ridge and lasso regression is that it tends to make coefficients to absolute zero as compared to Ridge which never sets the value of coefficient to absolute zero.

Elastic Net Regressor:-

Sometimes, the lasso regression can cause a small bias in the model where the prediction is too dependent upon a particular variable. In these cases, elastic Net is proved to better it combines the regularization of both lasso and Ridge. The advantage of that it does not easily eliminate the high collinearity coefficient.

Decision Tree Regressor:-

Decision trees can be used for classification as well as regression problems. The name itself suggests that it uses a flowchart like a tree structure to show the predictions that result from a series of feature-based splits. It starts with a root node and ends with a decision made by leaves.

Decision trees are upside down which means the root is at the top and then this root is split into various several nodes. Decision trees are nothing but a bunch of if-else statements in layman terms. It checks if the condition is true and if it is then it goes to the next node attached to that decision.

Random Forest Regressor:-

Random Forest is a technique that uses ensemble learning, that combines many weak classifiers to provide solutions to complex problems.

As the name suggests random forest consists of many decision trees. Rather than depending on one tree it takes the prediction from each tree and based on the majority votes of predictions, predicts the final output.

Random forests use the bagging method. It creates a subset of the original dataset, and the final output is based on majority ranking and hence the problem of overfitting is taken care of.

XGBoost Regressor:-

XGBoost is a powerful approach for building supervised regression models. The validity of this statement can be inferred by knowing about its (XGBoost) objective function and base learners. The objective function contains loss function and a regularization term. It tells about the difference between actual values and predicted values, i.e how far the model results are from the real values. The most common loss functions in XGBoost for regression problems is reg:linear, and that for binary classification is reg:logistics. Ensemble learning involves training and combining individual models (known as base learners) to get a single prediction, and XGBoost is one of the ensemble learning methods. XGBoost expects to have the base learners which are uniformly bad at the remainder so that when all the predictions are combined, bad predictions cancels out and better one sums up to form final good predictions.

Gradient Boosting Regressor:-

Gradient boosting gives a prediction model in the form of an ensemble of weak prediction models, which are typically decision trees.When a decision tree is the weak learner, the resulting algorithm is called gradient-boosted trees; it usually outperforms random forest.A gradient-boosted trees model is built in a stage-wise fashion as in other boosting methods, but it generalizes the other methods by allowing optimization of an arbitrary differentiable loss function.

Data visualization Analysis:

Data Visualization is the process of analyzing data in the form of graphs or maps, making it a lot easier to understand the trends or patterns in the data.

Correlation Heat map:

Analysis of the relation between the various columns of the cleaned data through the Correlation Heat map Matrix.

A correlation heat map is a graphical representation of a correlation matrix representing the correlation between different variables.

Box Plot:-

A boxplot is a standardized way of displaying the distribution of data based on a five number summary (“minimum”, first quartile [Q1], median, third quartile [Q3] and “maximum”). It can tell you about your outliers and what their values are.

Line Chart:

A line chart is a graphical representation of an asset's historical action that connects a series of data points with a continuous line.

Horizontal Bar Chart:

A horizontal bar chart is a graph in the form of rectangular bars. The length of these bars is proportional to the values they represent.

Count Plot:-

The countplot is used to represent the occurrence(counts) of the observation present in the categorical variable. It uses the concept of a bar chart for the visual depiction.

Distplot:-

distplot() function is used to plot the distplot. The distplot represents the univariate distribution of data i.e. data distribution of a variable against the density distribution. The seaborn. distplot() function accepts the data variable as an argument and returns the plot with the density distribution.

Subplot:-

A subplot is a narrative thread that is woven through a book to support the elements of the main plot.

Bar Chart:

A bar chart is a chart that presents categorical data with rectangular bars with heights or lengths proportional to the values that they represent. A bar chart is sometimes called a column chart.

Challenges:-

•Large Dataset to handle.

•Needs to plot lot of Graphs to analyse.

•Carefully handled Feature selection part as it affects the R2 score.

•Carefully tuned Hyperparameters as it affects the R2 score.

•Handled the positive skewness of the target variable.

•Handled the high correlation between various features.

•Need to convert categorical features into numerical features using feature engineering.

Conclusion:

Comparison of RMSE values for Test Data as lower the RMSE better the model performance:

LinearRegressor RMSE: **7.487**

Ridge Regression RMSE: **7.485**

Lasso Regression RMSE: **7.741**

Elastic Net Regressor RMSE: **8.788**

DecisionTreeRegressor RMSE: **5.233**

RandomForestRegressor RMSE: **4.027**

XGBoostRegressor RMSE: **4.824**

GradientBoostingRegressor RMSE: **4.824**

• XGBoost Regressor and Gradient Boost Regressor gives the highest R2 score of 98% and

96% respectively for Train dataset and 91% for both regressor's Test dataset.So, We can deploy these models.

• The Temperature, Hour & Functioning Day are the most important features that positively drive the total rented bikes count.

• In conclusion, the demand prediction for the given Seoul bike sharing dataset can be accurately predicted using XGBoost Regressor and Gradient Boost Regressor.

References-

- **Seoul Bike Rental Analysis public Dataset:**

  https://data.seoul.go.kr/index.do

- **Python Pandas Documentation:**

  https://pandas.pydata.org/pandas-docs/stable

- **Python Numpy Documentation:**

  https://numpy.org/doc/

- **Python MatPlotLib Documentation:**

  https://matplotlib.org/stable/index.html

- **SciKit Documentation:**

  https://scikit-learn.org/stable/