BIKE SHARING DEMAND PREDICTION

Hark Pun,
Data Science Trainee,
AlmaBetter, Bangalore.

ABSTRACT:

As a convenient, economical, and eco-friendly travel mode, bike-sharing greatly improved urban mobility. However, it is often very difficult to achieve a balanced utilization of shared bikes due to the asymmetric user demand distribution and the insufficient numbers of shared bikes, docks, or parking areas. If we can predictthe short-run bike-sharing demand, it will help operating agencies rebalance bike- sharing systems in a timely and efficient way.

INTRODUCTION:

Some countries around the world are practicing righteous scenarios, rendering mobility at a fair cost and reduced carbon discharge. On the contrary other cities are far behind in the track. Urban mobility usually fills 64% of the entire kilometers travelled in the world. It ought to be modelled and taken over by inter-modality and networked self-driving vehicles which also provides a sustainable means of mobility. Systems called Mobility on Demand have a vital part in raising the vehicles' supply, increasing its idle time and numbers.

PROBLEM STATEMENT:

- Maximize: The availability of bikes to the customer.
- Minimize: Minimize the time ofwaiting to get a bike on rent.

PROJECT GOAL:

To find factors and causes which influence shortages of bikes and time delay of availing bikes on rent. Using the data provided, this paper aims to analyze the data to determine what variables are correlated with bike demand prediction. Hourly count of bikes for rent will also be predicted.

FEATURE DESCRIPTION:

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 –
  NoFunc (Non-Functional Hours),
  Fun (Functional hours)

MISSING VALUES:

One of the ways to handle missing values is to simply remove them from our dataset. We have known that we can use the null() and not null() functions from the pandas library to determine null values. Since there are no missing values in this data set.

DUPLICATED DATA:

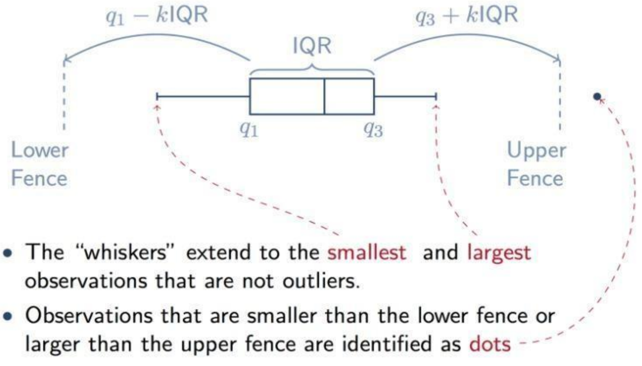
It is very likely that your dataset contains duplicate rows. Removing them is essential to enhance the quality of the dataset. Since there is no duplicate value in these datasets.

EXPLORATORY DATA ANALYSIS:

After loading the dataset, we performed this method by comparing our target variable that is "Rented Bike Count" with other independent variables. This process helped us figuring out various aspects and relationships among the target and the independent variables. It gave us a better idea of which feature behaves in which manner compared to the target variable.

OUTLIER TREATMENT:

Interquartile Range (IQR)



- The "whiskers" extend to the smallest and largest observations that are not outliers.
- Observations that are smaller than the lower fence or larger than the upper fence are identified as dots

The interquartile range rule is important for spotting outliers. Interquartile Range score or middle 50% or H-spread is a measure of statistical dispersion, being equal to the difference between the75th percentile and 25th percentile i.e.,third quartile(Q3) and first quartile(Q1). Weidentify the outliers as values less than **Q1 -(1.5 * IQR)** or greater than **Q3+(1.5 * IQR). IQR=Q3-Q1**

QUANTILE CAPPING AND FLOORING:

In this technique, we will do the flooring (e.g., the 5th percentile) for the lower values and capping (e.g., the 95th percentile) for the higher values. The lines of code below print value in between 5th and 95th percentiles, respectively. These values will be used for quantile-based flooring and capping.

FEATURE TRANSFORMATION:

Transformation of the skewed variables may also help correct the distribution of the variables. These could be logarithmic, square root, or square transformations. In our dataset Dependent variable i.e., "Rented Bike Count" having a moderate right skewed, to apply linear regression dependent features must follow thenormal distribution. Therefore, we use square root transformation on top of it.

CLEANING AND MANIPULATING THE DATASET:

DATA PREPROCESSING:

It is the process of transforming raw data into a useful, understandable format. Real-world or raw data usually has inconsistent formatting, human errors, and can also be incomplete. Data pre-processing resolves such issues and makes datasets completer and more efficient.

DATA CLEANING:

Cleansing is the processof cleaning datasets by accounting for missing values, removing outliers,correcting inconsistent data points, andsmoothing noisy data. In essence, the motive behind data cleaning is to offer complete and accurate samples formachine learning models.

NOISE DATA:

A large amount of meaningless data is called noise. More precisely, it's the random variance in a measured variable or data having incorrect attribute values. Noise includes duplicate orsemi-duplicates of data points, data segments of no value for a specific research process, or unwanted information fields.

DETECTING MULTICOLLINEARITY BY VARIANCE INFLATION FACTOR(VIF)

$$VIF_i = \frac{1}{1 - R_i^2}$$

Variance inflation factor (VIF) is a measure amount of multicollinearity in a set of multiple regression variables.

Mathematically, the VIF for a regression model variable is equal to the ratio of the overall model variance to the variance of a model that includes only that single independent variable. This ratio is calculated for each independent variable**.**

Here we have taken the VIF for consideration value is 10 for having some important features to accord with the model which we will be using in this dataset.

MODEL BUILDING: PREREQUISITES

FEATURE SCALING: Scaling data is the process of increasing or decreasing the magnitude according to a fixed ratio, in simpler words you change the size but not the shape of the data.

Different types of feature scaling:

- **Standardization**: In this method we centralize the data, then we divide by the standard deviation to enforce that the standard deviation of the variable is one.

$$X_{std} = \frac{X - \bar{X}}{s_X}$$

- **Normalization:** Normalization most often refers to the process of "normalizing" a variable to be between 0 and 1. Think of this as squishing the variable to be constrained to a specific range. This isalso called min-max scaling.

$$X_{norm} = \frac{X - min(X)}{max(X) - min(X)}$$

EVALUATION MATRICS:

Evaluation metrics are a measure of how good a model performs and how well it approximates the relationship. Let us look at **MSE, RMSE and R-squared, Adjusted R-squared**

MEAN SQUARED ERROR(MSE)

The most common metric for regression tasks is MSE. It has a convex shape. It is the average of the squared difference between the predicted and actual value.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$$

ROOT MEAN SQUARED ERROR(RMSE)

This is the square root of the average of the squared difference of the predicted and actual value.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}$$

R-SQUARED

R-square is a comparison of residual sum of squares *(SS_{res})* with total sum of squares*(SS_{tot})*.

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \overline{y})^2}$$

ADJUSTED R-SQUARED:

The main difference between adjusted R-squared and R-square is that R-squared describes the amount of variance of the dependent variable represented by every single independent variable, while adjusted R-squared measuresvariation explained by only theindependent variables that affect dependent variable.

$$R^2_{adjusted} = [\frac{(1-R^2)(n-1)}{n-k-1}]$$

HYPERPARAMETER TUNING:

Hyperparameters are the variables that the user specifies usually while building the Machine Learning model.

GRIDSEARCH CV()

Using a different combination of all the specified hyperparameters and their values and calculates the performance for each combination and selects the best value for the hyperparameters. This makes the processing time-consuming and expensive based on the number of hyperparameters involved.

RANDOMIZEDSEARCHCV():

RandomizedSearchCV the model selects the combinations randomly.

For example, if there are 500 values in the distribution and if we input **n_iter=50** then random search will randomly sample 50 values to test.

GridSearchCV and RandomizedSearchCV method isavailable in the scikit-learnclass model_selection**.** It can be initiated by creating an object of GridSearchCV() or RandomizedSearchCV.

These are main arguments-

For GridSearchCV - **estimator**, **param_grid**, **cv**, **scoring.**

For RandomizedSearchCV - **estimator**, **param_distribution**, **cv**, **scoring, n_iter**

- **estimator**: In which model you want to perform GridSearchCV.
- **param_grid/param_distribution:** Dictionary with parameters names (str) as keys and lists of parameter settings to try as values, or a list of such dictionaries, in which case the grids spanned by each dictionary in the list are explored.
- **scoring**: Strategy to evaluate the performance of the cross-validated model on the test set.
- **cv:** Determines the cross-validation splitting strategy.
- **n_iter:** Number of parameter settings that are sampled. n_iter trades off runtime vs quality of the solution.

We perform RandomizedSearchCV on Random Forest model and GridSearchCV on XGBoost and LGBoost model.

ALGORITHMS:

LINEAR REGRESSION:

Linear regression is a supervised machine learning model majorly used in forecasting. Supervised machine learning models are those where we use the training data to build the model and then test the accuracy of the model using the loss function.

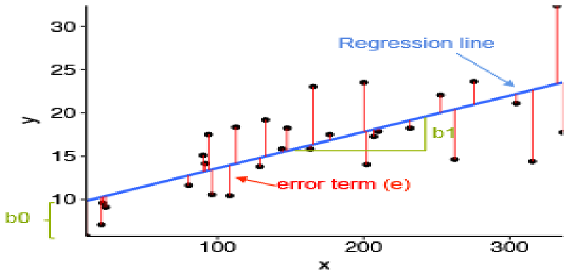
As the name suggests, it assumes a linear relationship between a set of independent variables to that of the dependent variable (the variable of interest).

We're going to fit a line

$$y = \beta0 + \beta1x$$

to our data. Here, x is called the independent variable or predictor variable, and y is called the dependent variable or response variable. Before we talk about how to do the fit, let's take a closer look at the important quantities from the fit:

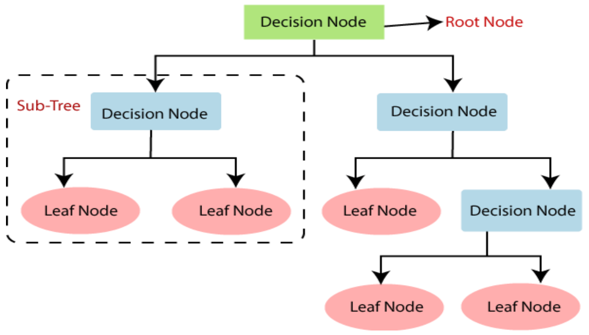
- β1 is the slope of the line: this is one of the most important quantities in any linear regression analysis
- β0 is the intercept of the line



DECISION TREE:

Decision Tree is a supervised learning technique that can be used for both classification and Regression problems. It is a tree-structured classifier, where internal nodesrepresent the features of a dataset, branches representthe decision rules, and each leaf noderepresents the outcome. In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. To build a tree, we use the CART algorithm, which stands for Classification and Regression Tree algorithm. A decision tree can contain categorical data (YES/NO) as well asnumeric data.

- **Root Node:** Root node is from where the decision tree starts.
- **Leaf Node:** Leaf nodes are the final output node
- **Splitting:** Splitting is the process of dividing the decision node/root node into sub-nodes
- **Branch/Sub Tree:** A tree formed by splitting the tree.
- **Pruning:** Pruning is the process of removing the unwanted branches from the tree.
- **Parent/Child node:** The root node of the tree is called the parent node, and other nodes are called the child nodes.

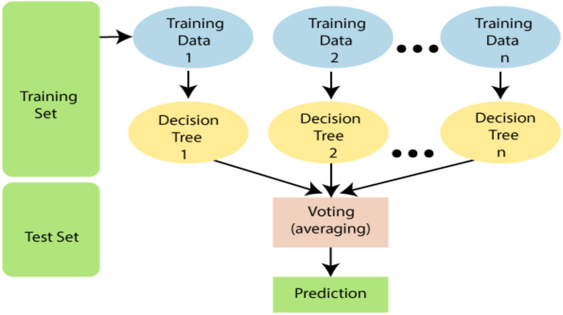


Ensemble uses two types of methods:

- **Bagging**– It creates a different training subset from sample training data with replacement & the final output is based on majority voting. For example – Random Forest
- **Boosting**– It combines weak learners into strong learners by creating sequential models such that the final model has the highest accuracy. For example – AdaBoost, XGBoost, CatBoost etc.

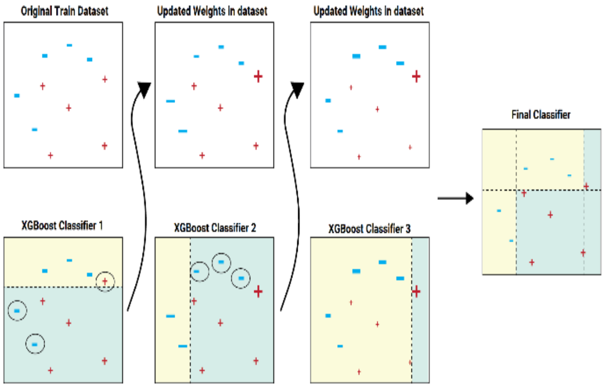
RANDOM FOREST:

Random Forest is aclassifier that contains multiple number of Decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and itpredicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.



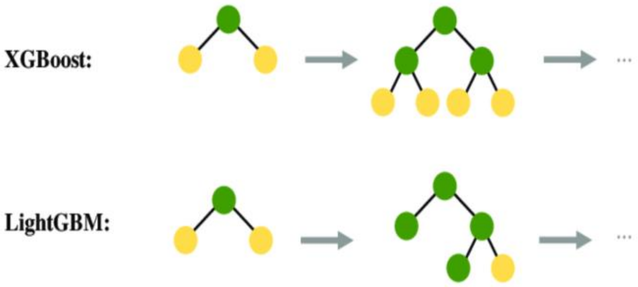
XTREME GRADIENT BOOSTING:

In this algorithm,decision trees are created in sequential form. Weights play an important role in XGBoost. Weights are assigned to all independent variables which are then fed into the decision tree which predicts results. The weight of variables predicted wrong by the tree is increased and these variables are then fed to the second decision tree. These individual predictors then ensemble to give a strong and more precise model. XGBoost comes under the boosting ensemble techniques which combines the weaknessof primary learners to the next strong and compatible learners.



LIGHT GRADIENT BOOSTING:

LightGBM is a gradient boosting framework based on decision trees to increases the efficiency of the model and reduces memory usage. LightGBM splits the tree leaf-wise as opposed to other boosting algorithms that grow tree level-wise. It chooses the leaf with maximum delta loss to grow. Since the leaf is fixed, the leaf-wise algorithm has lower loss compared to the level-wise algorithm. Leaf-wise tree growth might increase the complexity of the model and may lead to overfitting in small datasets. Below is a diagrammatic representation of Leaf-Wise Tree Growth.



FEATURE IMPORTANCE:

Feature Importance refers to techniques that calculate a score for all the input features for a given model; the scores simply represent the “importance” of each feature.
A higher score means that the specific feature will have a larger effect on the model that is being used to predict a certain variable – Feature technique isassociated with the tree-based algorithms like random forest, XGBoost and so on.

In linear regression we use coefficient as a type of feature importance – Linear learning algorithms fit a modelwhere the prediction is the weighted sum of the input values. Examples include linear regression, logistic regression, and extensions that add regularization, such as ridge regression and the elastic net. All these algorithms find a set ofcoefficients to use in the weighted sum in order to make a prediction. directly as a crude type of feature importance score.

SUMMARY:

The project goal is to minimize the waiting time or make rental bike available at the right time, so company want to predict the bike count required at each hour for the stable supply.

- Performing exploratory data analysis on data to understand the how independent variable behave with dependent variable.
- Checking multicollinearity because after plotting heatmap correlation we understood there are two independent variables strongly corelate with each other, so calculate the **VIF** then dropping the one of the features.
- For data preparation the categorical features convert by using one hot encoding or label encoding.
- Methodology for solving this problem is needed to understand the data, prepare the data and visuals the data, important features selection and building a models finalize based on r2 score and root mean square error after that tune the hyperparameter.
- Most Important feature results have shown that temperature and hour of the day are most influence variable in hourly rented bike demand prediction.
- Budling the model using different-different ML algorithm like Ridge, Lasso, Decision tree, KNN, XGBoost, Random Forest, LGBM, support vector machine. Comparing these models with r2 and RMSE evaluation metrics.
- We get best r2 score and low root mean square error from LGBM model.
- To understand how affecting the feature for best model results we use **Shaply** for model explainability.

CONCLUSION:

Exploratory data analysis-

- First for a Working Day where the rental count high at peak office hours (8am and 5pm).
- Second for a Non-working day where rental count is more or less uniform across the day with a peak at around noon.
- Temperature: People generally prefer to bike at moderate to high temperatures. We see highest rental counts between 30 to 35 degrees Celsius.
- Hour: Demand of rental bike is high at Evening time in between 4PM to 8PM.
- Weather: As one would expect, we see highest number of bike rentals on a clear day and the lowest on a snowy or rainy day.
- Season: Demand of rental bikes is high during Summer and Autumn season.
- Humidity: With increasing humidity, we see decrease in the number of bike rental count.
- So from the above operations we are coming to the conclusion that for LGBM(Light Gradient Boosting Algorithm) model performing very well than the other models.

So, in future if we want to do some prediction with this data then the LGBM model will fit perfectly to do the prediction with the highest R2 score and lowest RMSE value.

REFRENCES:

- https://www.kaggle.com/
- https://medium.com/search?q=linear+regression
- https://www.analyticsvidhya.com/blog/