**Seoul Bike Sharing Demand Prediction**

**Puneet Suthar**

**Data Science Enthusiast**

**AlmaBetter, Bangalore**

**Abstract:**

The scarcity of available bikes is a significant challenge in bike-sharing systems, prompting researchers to develop methods to accurately predict demand and enable effective redistribution of bikes. However, the task of predicting bike counts can be difficult, particularly when the available data is imbalanced. Despite numerous efforts to develop models that accurately predict demand, there is currently no consensus on which machine learning techniques provide the best performance, due to variations in the features used. Furthermore, there are no standardized features that have been proven to improve model performance. Feature engineering is heavily emphasized in Kaggle kernels, but not as much in published journal articles. This report will examine the best-performing machine learning techniques, feature engineering methods, and features that can significantly enhance bike-sharing demand prediction.

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 bike count required at each hour for the stable supply of rental bikes.

1. **Introduction**

Bike rentals have experienced a surge in popularity in recent years, with people using the service more frequently due to its relatively affordable rates and convenience of pick-up and drop-off at their own discretion. Ensuring the availability and accessibility of rental bikes to the public at the appropriate times reduces waiting time and ultimately provides a steady supply of bikes to the city. The objective of this project is to develop a machine learning model capable of forecasting the demand for rental bikes in Seoul.

1. **Seoul Bike Sharing Dataset Insight**

This dataset has around 8,760 observations in it with 14 columns and it is a mix of categorical and numeric values The dataset contains weather information (Temperature, Humidity, Windspeed, Visibility, Dewpoint, Solar radiation, Snowfall, Rainfall), the number of bikes rented per hour and date information. Exploring them will definitely help in understanding of the booking trends.

**Column Information**

We are given the following columns in our data:

1. Date : year-month-day

2. Rented Bike count - Count of bikes rented at each hour

3. Hour - Hour of the day

4. Temperature-Temperature in Celsius

5. Humidity - %

6. Wind Speed - m/s

7. Visibility - 10m

8. Dew point temperature - Celsius

9. Solar radiation - MJ/m2

10. Rainfall - mm

11. Snowfall - cm

12. Seasons - Winter, Spring, Summer, Autumn

13. Holiday - Holiday/No holiday

14. Functional Day - No(Non Functional Hours), Yes(Functional hours)

1. **Steps involved**
   1. Performing EDA (exploratory data analysis).
   2. Observation and conclusions from the data .
   3. Getting the data ready for Model training.
   4. Model Selection by Evaluating metrics
   5. Training the model.
   6. Deployment of model as a web app.
2. **Performing EDA (exploratory data analysis)**
3. Exploring head and tail of the data to get insights on the given data.
4. Looking for null values, No. of zeros, duplicates, no. of unique in every column, it help us to make a guideline for preliminary feature engineering section before major EDA.
5. In Preliminary feature engineering section we performed following tasks:
   1. Rename the complex columns name.
   2. Converting Date to datetime object and extracting Date, month, year, week days
   3. creating new column for weekend (yes or no).
   4. Dropping unwanted Date column.
6. In Major EDA we performed below experimemts:
   1. Ploted a distribution plot of numeric data to get the idea of the skewness.
   2. Box plot to get a look on outliers.
   3. Regression plot to see the effect of feature on target column.
   4. Some bar plots between categorical columns and target column
   5. A correlation heat map to understand the correlation and multicollinearity.
   6. A pair plot to have a broad outlook to the data feature dependency.
7. **Observations and conclusions from the data** 
   1. We observed that count of rows of Non functioning days are 295 but this count is including hours, because the data is recorded every hour, and holidays are 432 which are also including count of hours a day.
   2. So, Actual Non Functioning days are 295/24 = 12.291 (Which should be a int not float if 24 hrs of Non functioning days are non functioning ) and Actual holidays = 432/24 = 18.
   3. Some column contain high no. of zeros (Solar Radiation, Rainfall, Snowfall).
   4. Found no null values no duplicates.
   5. Some column are higly skewed (Rented bike count, wind speed, visibility, solar radiation, rainfall, snowfall.)
   6. low bike count in 2017, high bike count in 2018.
   7. 2 peaks are observed in 24 hours ( 8:00 and 18:00)
   8. Bike count increase as temp increased.
   9. Low humidty high bike count , high humidity low bike count.
   10. low bike count in high wind speed, (a suprising peak at 7.2 and 7.4 m/s)
   11. Bike count increases with solar radiation.
   12. People tend to rent bikes when there is no or less rainfall and snowfall
   13. Demand is high in summer > Autumn > Spring > Winter.
   14. High demand in No Holiday compared to holidays.
   15. People tend to rent bikes when the temperature is between -5 to 25 degrees.
   16. People tend to rent bikes when the visibility is
   17. between 300 to 1700.
   18. Highest peaks in 5,6,7 months
   19. Count least on sunday.
   20. There is drop in demand on 1,2,34,10,11,12 specific dates every month.
   21. The rentals were more in the morning and evening times. This is because people not having personal vehicle, commuting to offices and schools tend to rent bikes
8. **Getting the data ready for training**
   1. Performed lable encoding in to Seasons, Holiday, Functioning\_Day, WeekDay, Weekend.
   2. Applied square root transformation to the Rented bike count column, to reduce the skewness, skewness reduced from 0.98 to 0.15
   3. Applied min max scaler on all the features.
9. **Model Selection by Evaluating metrics**
   1. Some models are chosen based on the data and the scatter plots.
   2. **Linear regression:**

Linear regression is a statistical method used to model the relationship between a dependent variable and one or more independent variables. The goal of linear regression is to find the best linear equation that describes the relationship between the variables, and to use this equation to make predictions.

* 1. **Elastic Net:**

Elastic Net is a regularization technique used in linear regression and other models to prevent overfitting. It combines the penalties of L1 and L2 regularization, resulting in a model that can handle correlated features and select relevant features for the model.

* 1. **Random Forest Regressor:**

Random Forest Regressor is a popular machine learning algorithm that uses an ensemble of decision trees to make predictions. It combines the results of multiple decision trees to produce a more accurate and robust model that can handle noisy data and avoid overfitting.

* 1. **Decision Tree Regressor:**

Decision Tree Regressor is a machine learning algorithm that builds a tree-like model of decisions and their possible consequences. It partitions the data into smaller subsets based on the values of the input variables and uses these partitions to predict the target variable.

* 1. **Gradient Boost Regressor:**

Gradient Boost Regressor is a machine learning algorithm that combines multiple weak decision trees to make predictions. It iteratively trains decision trees to correct the errors of the previous trees, resulting in a powerful model that can handle complex data and nonlinear relationships between variables.

* 1. To find the best suitable model , I defined the a function ModelSelection() which takes a input of x and y data , list of models to be checked, a dictionary contains the hyperparameter, no. of splits, and a scoring method. It gives us the best hyperparameters for the giveen models and neg mean absolute error score for a quick comparison.
  2. But single metric is not enough to get a best model, so a new function is defined CrossValidation\_model\_comparision() it takes input as a list of models, feature and target data, no. of splits, and a scaler to scale the data. And gives data frame containing Min, Max and Avg of the Mean absolute error, Mean Squared error, Root mean squared error, r2\_score for every model by performing cross validation.
  3. Evaluating metrics :
     1. **Mean Absolute Error (MAE):-**

MAE is a metric used to evaluate the accuracy of a model by measuring the average absolute difference between the predicted values and the actual values. The lower the MAE, the better the model's performance.

* + 1. **Mean Squared Error (MSE):-**

MSE is a metric used to evaluate the accuracy of a model by measuring the average of the squared differences between the predicted values and the actual values. The lower the MSE, the better the model's performance in terms of accuracy.

The main difference is that MAE measures the average absolute difference, while MSE measures the average squared difference. Which help to magnify the errors.

* + 1. **Root Mean Squared Error (RMSE):-**

Root Mean Squared Error (RMSE) is a metric used to evaluate the accuracy of a model by measuring the square root of the average of the squared differences between the predicted values and the actual values. The lower the RMSE, the better the model's performance in terms of accuracy. It is similar to MSE, but RMSE is expressed in the same units as the data being measured, making it more interpretable.

* + 1. **r2\_score :-**

The R-squared (R2) score is a statistical measure that represents the proportion of the variance in the dependent variable that is explained by the independent variables in a regression model. It is a value between 0 and 1, where 1 indicates a perfect fit and 0 indicates no fit. A higher R2 score indicates a better fit of the model to the data.

1. **Training the best selected model**

The Gradient Boost regressor is selected as our best model for the web app. Because it have least MAE, MSE, RMSE, and highest R2 score.

The 80% of the data is used in training the model. And a model is saved as a pickle file to be used in Deployment.

1. **Deployment of Selected Model as a web app.**

This code inserts a streamlit app that predicts the count of rental bikes based on weather data, date, and other variables. It uses an API to collect weather data and preprocesses it. The code has a function get\_season() to convert the date into a season. It collects data from the user such as holiday or not, date of prediction, and function day or not. The user enters the date for which they want to predict the bike count. Then, the app collects the weather data for that date using the open-meteo API. It stores data of every hour as a dictionary and assigns the values to the DataFrame created for each hour. It encodes the labels using LabelEncoder and loads the trained model from a pickle file. Finally, it predicts the bike count and displays the result.

**5. Conclusion**

* We have made significant progress in our machine learning project by training 7 different models on the training dataset. Each of these models has been refined through hyperparameter tuning to enhance its performance and achieve the best possible results. Our objective is to build accurate predictive models that can forecast the demand for rental bikes based on a variety of weather conditions and other factors.
* After training and testing each of the models, we found that the Gradient Boost prediction model delivered the lowest Root Mean Squared Error (RMSE). This model combines multiple decision trees and trains them iteratively to produce a highly accurate and robust model that can handle complex data and nonlinear relationships between variables. The Gradient Boost model is therefore an excellent choice for businesses that prioritize accuracy in their predictions.
* However, we recognize that the final choice of model for deployment ultimately depends on the specific needs and preferences of the business stakeholders. For example, if stakeholders place a high value on model interpretability, we can deploy the decision tree model. This model uses a tree-like structure to identify key factors that influence bike rentals, making it easier to understand the underlying logic of the predictions.
* Overall, our project has succeeded in developing and refining machine learning models that can make accurate predictions about bike rentals under varying weather conditions. These models can help businesses optimize their rental operations by providing insights into the factors that drive customer demand, and ultimately lead to better decision-making and improved business outcomes.

**References**

1. Towards Data Science
2. Stack overflow
3. Medium