Seoul Bike Sharing Demand Prediction

We were asked to forecast bike rental demand of Bike sharing based on historical usage patterns in relation with weather, time and other data. The dataset contains weather information (Temperature, Humidity, Windspeed, Visibility, Dewpoint, Solar radiation, Snowfall, Rainfall), the number of bikes rented per hour and date information. 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. Using these Bike Sharing systems, people rent a bike from one location and return it to a different or same place on need basis. People can rent a bike through membership or on demand basis.

We followed the following sequence of steps to solve this problem statement:

- Understanding the dataset, doing some basic inspection on the raw data to check the number of columns, understanding distribution of data and checking statistics of the data in each variable. Checking for missing values, Visualizing the distributions and boxplots of each variable to handle the outliers, Cleaning the data.
- Feature engineering, created some new features, dropped unnecessary features and encoding the data into numeric form. Tried making the dependent variable normally distributed by some transformations.
- Bi-variate analysis to check whether there's any linear relationship between independent and the dependent variable. Correlation analysis to visualize the severity of multicollinearity. Removing multicollinearity based on VIF factor.
- Finally scaling the data and experimenting different algorithms. First, we started with simple models like Linear Regressor and Decision Tree then to enhance the accuracy we tried some complex algorithms like Tree ensemble.
- Since there was not much linear relation between the independent and dependent variables, the linear regressor model did not perform well, so we moved to Tree based algorithms and performance was drastically improved. We kept on improving the model performance by using some boosting and ensemble algorithms and tuning the hyperparameters. The best performance was given by XGBoost model.

- We observed different evaluation metrics with best set of hyperparameters for the experimented models to overcome underfitting or overfitting and also had a rough idea of feature importance for each model.

We observed following results after completing the task:

- Functioning day is the most influencing feature and temperature is at the second place for Linear Regressor.
- Temperature is the most important feature for DecisionTree, RandomForest and Gradient Boosting Regressor.
- Functioning day is the most important feature and Winter is the second most for XGBoost Regressor.
- RMSE Comparisons:
    - LinearRegressor RMSE: 370.46
    - DecissionTreeRegressor RMSE : 302.53
    - RandomForestRegressor RMSE: 290.02
    - XGBoostRegressor RMSE : 242.72
    - GradientBoostingRegressor RMSE : 248.18
- The feature temperature is on the top list for all the regressors except XGBoost.
- XGBoost is acting different from all the regressors as it is considering whether it is winter or not. And is it a working day or not. Though winter is also a function of temperature only but it seems this trick of XGBoost is giving better results.
- XGBoostRegressor has the Least Root Mean Squared Error. So, it can be considered as the best model for given problem.

XGBoost is giving the best accuracy of around 82%