Sharing Bike Demand Prediction

Jiatao Yuan

Brown University

Introduction

Sharing bike brings much more convenience for people's life, and this also serves as a complement for the public transportation to ease the traffic. However, the demand for the sharing bike varies from time to time, if there are more sharing bikes on the street, it will not only increase the operation cost for the company but also have a reverse effect on easing the traffic pressure, which will block the road sometimes because the overwhelming amount of bikes. Therefore, it is essential to make a prediction about the demand for the sharing bike from time to time, which will lower the company's cost an also increase the efficiency. The data is from the UCI machine learning repository(https://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset). The original data is consisted of 17379 data points and 17 features.

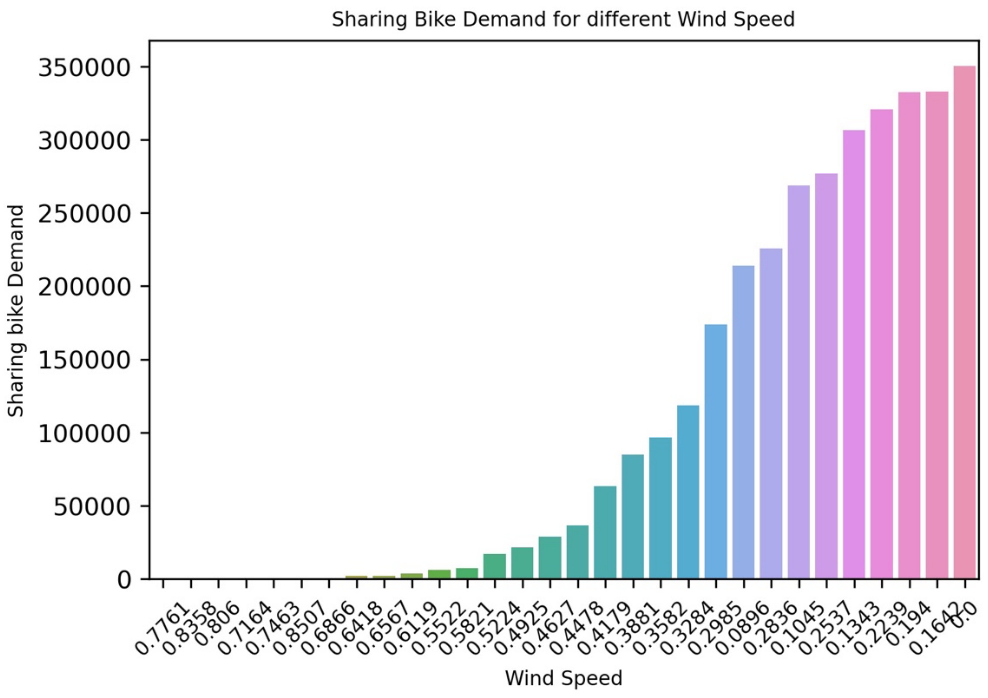
Overall, the features of this dataset are the environmental and seasonal settings that highly correlated with bike sharing, which is 'datetime', 'season', 'year', 'month', 'hour', 'holiday', 'weekday', 'working day', 'weather', 'temp', 'atemp', 'humidity', 'windspeed', 'casual', 'registered', all continuous numeric features are normalized or Min-Max Scaled. The causal and registered represents the number of the users of type causal and registered. The target variable is the number of bikes rented for that one-hour period. Since the target variable is continuous, then it is a regression problem.

There are plenty of interesting projects and papers that this dataset has been used. One project solves the problem of predicting the demand for the sharing bikes per hour and its RMSE is desirable, it found that there is there is a highly correlated feature pairs with the target variables, which are registered and causal users. And this project then found the sum of registered and causer users adds up to the target variables. And it makes sense to drop those two features because we can't know the exact number of those two features during that one-hour period, they're generated after we get the total number of users for that hour, which will reduce some collinearity problem in some basic linear regression model(Maskara, 2021). Another interesting project tries to use the L2- regularization to address some problem of this dataset-collinearity because this dataset is highly concentrated and highly correlated within some features, which gives me pretty good idea in designing my model, and his/her model turns out to be top 10% of the Kaggle competition(Vivekanandan, 2021).

EDA

Plot 1

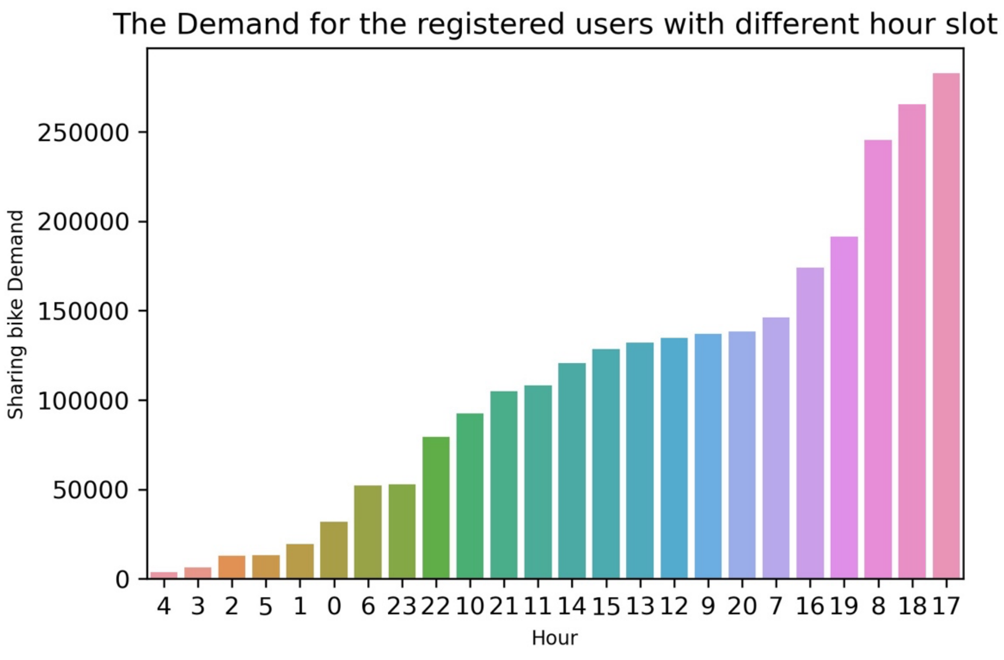
Sharing-Bike Demand with different Wind Speed



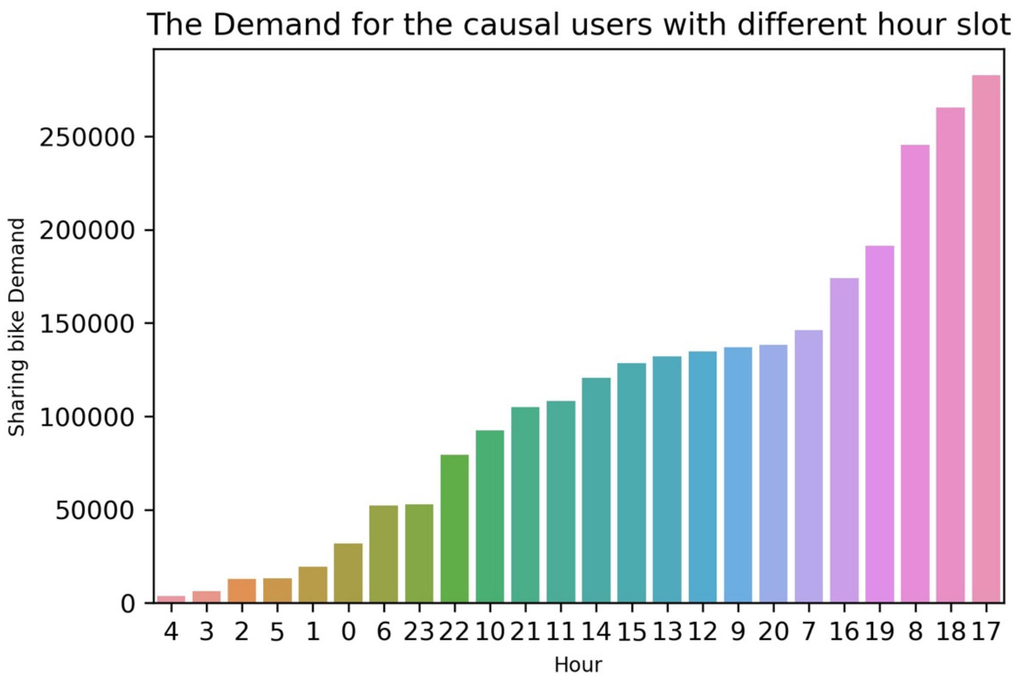
This graph is interesting because as you can see there is trend that as wind speed decreases, then the demand for renting the bike increases.

Plot 2

The Demand for the registered users with different hour slot

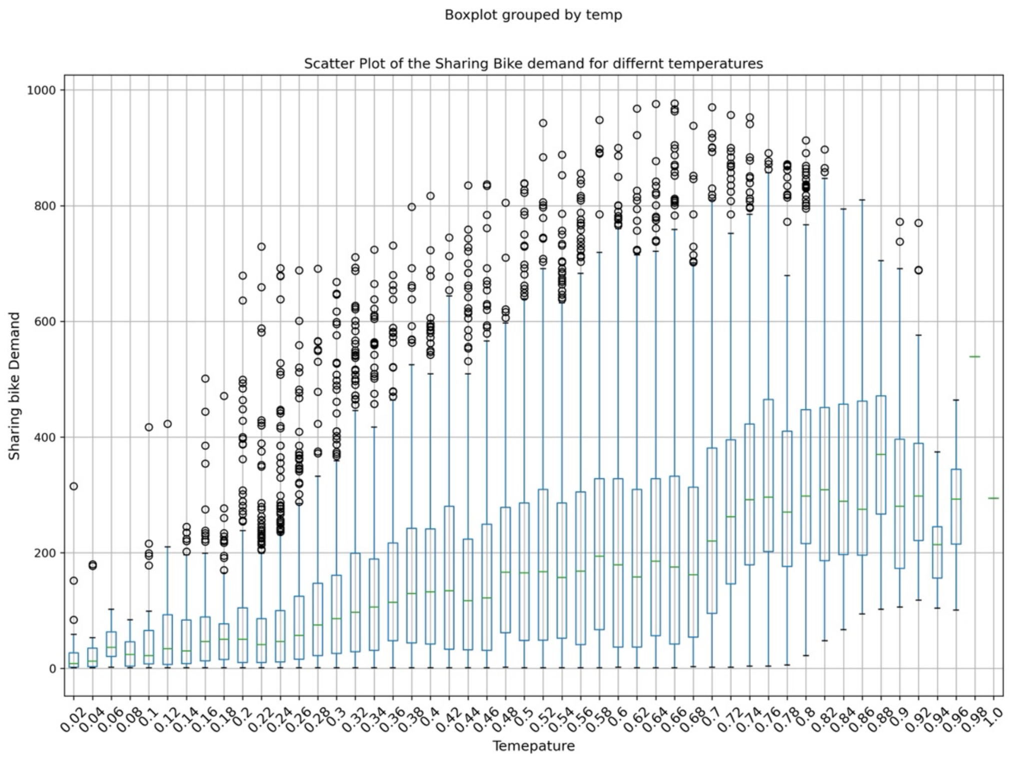


The Demand for the causal users with different hour slot



It is interesting that the registered users' demand for the sharing bike is high at 8AM and 5 and 6 PM. From that, we probably can infer that the registered users are mostly using sharing bike to work or so when we compared to another figure of causal users.

Plot 3
Scatter Plot of the Sharing Bike demand for differnt temperatures



As you can see, the average demand for the sharing bikes is increase almost when the temperature increases (except the temperature is too high), so there is a correlation between temperature and the demand for the sharing bikes.

Methods

Data Preprocessing
After preprocessing the data, there are 17372 data points and 31 features. I used the one hot encoder to the feature ‘working day', 'weather',' holiday' because they are categorical and unordered. I used the standard scaler to the new time-lag features ‘count1’,’count2’,’count3’,’count4’,’count5’,’count6’ because they are numerical and unbounded. This data is a time-series data so it is not i.i.d . It doesn’t have group structure.

ML_pipeline
I made an assumption after I see result of the autoregression: the current demand for the sharing bike is only dependent on the previous at most 6 hours. For example, if it is midnight (i.e 4AM), then the previous at most six hours is correlated with this hour because the previous hours are also midnight-hours, which is low-demanding. Also, you could see it in the result of the autoregression. I try to add more lags, this will not be an issue because I can put a penalty when I do the model tuning. After I add the time lags in my dataset, then my preprocessed dataset is i.i.d. Then I split the data as follows: 60% for the training dataset and 20% separately for the validation and testing dataset. I also used basic-K fold with 5 folds to make the cross-validation and tune the hyperparameter.

ML algorithms_1: Ridge
I tunedthe parameters of L2 regularization term alpha of the values of np.logspace(-8, -2, 200) under 10 random states, which a list of 200 numbers equally spaced between -10,-2 on the log-scale.

ML algorithms_2: Random Forest
I tuned parameters of max_depth and max_features. The max_depth parameters is ranging from 1 to 20, and the max_features is between 0 and 1. There are 20 combinations for 5 different random states

ML algorithms_3: GraidentBoost Regressor
I tuned parameters of learning rate and max_depth. The learning rate parameters’ value is a list of 3 numbers of 0.1, 0.2,0.3. The max_depth parameters’ value is a list of 3 numbers of 3,5,6. Therefore, it’s a combination of 9 numbers for 5 random state of 5 folds.

ML algorithms_4: XGBOOST

I tuned the parameters of "reg_alpha", "reg_lamda" and "max_depth". The max_depth is tuned of a list of 1,3,7,8,9,10. The lambda values are a list of numbers of [0e0, 1e-2, 1e-1, 1e0, 1e1, 1e2] The alpha values are a list of numbers of [0e0, 1e-1, 1e0, 1e1, 50] for 5 random states with 5 folds.

Evaluation

Since my model is a regression model, so R^2 measures how good my model will fit the dataset. Therefore, I choose R^2 as my evaluation metric then I could tell my model's goodness of fitting.

Uncertainty

For non-deterministic ML methods including XGBOOST and Random Forest, I try to make 5 random states with different random numbers for each method and I could measure that using the standard deviation of the test scores for each model.

Results

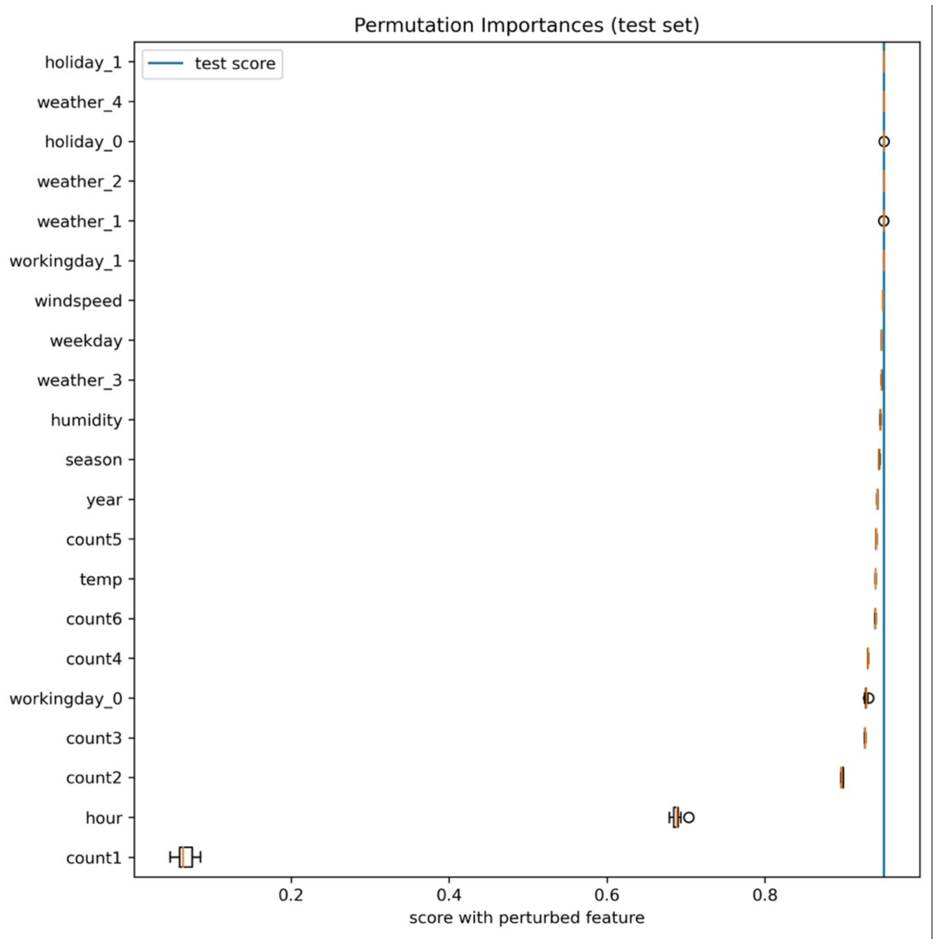
Model	Test Score
Ridge Regression	0.882+ - 0.005
Random Forest	0.958+- 0.001
GradientBoost	0.959+-0.001
XGBOOST	0.940+ - 0.002

Baseline of R^2 is 0. The most predictive model is GradientBoost Regressor with best parameters of max_depth=6, random_state=84,learning rate of 0.1

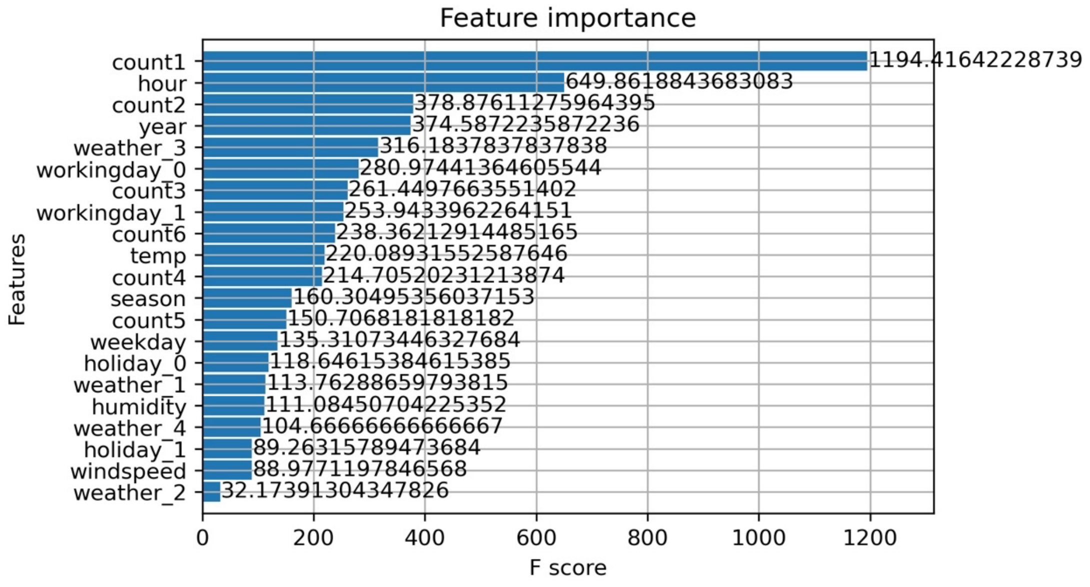
Global Feature Importance

For the XGBOOST model, I calculate 3 different types of the global feature importance.

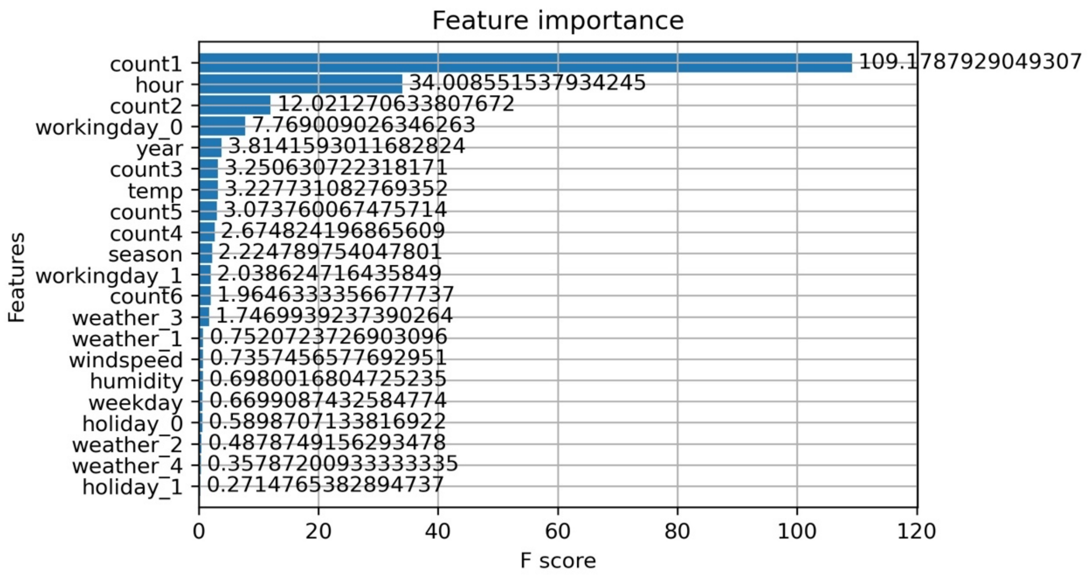
XGB-Permutation:



<u>XGB – Cover</u>



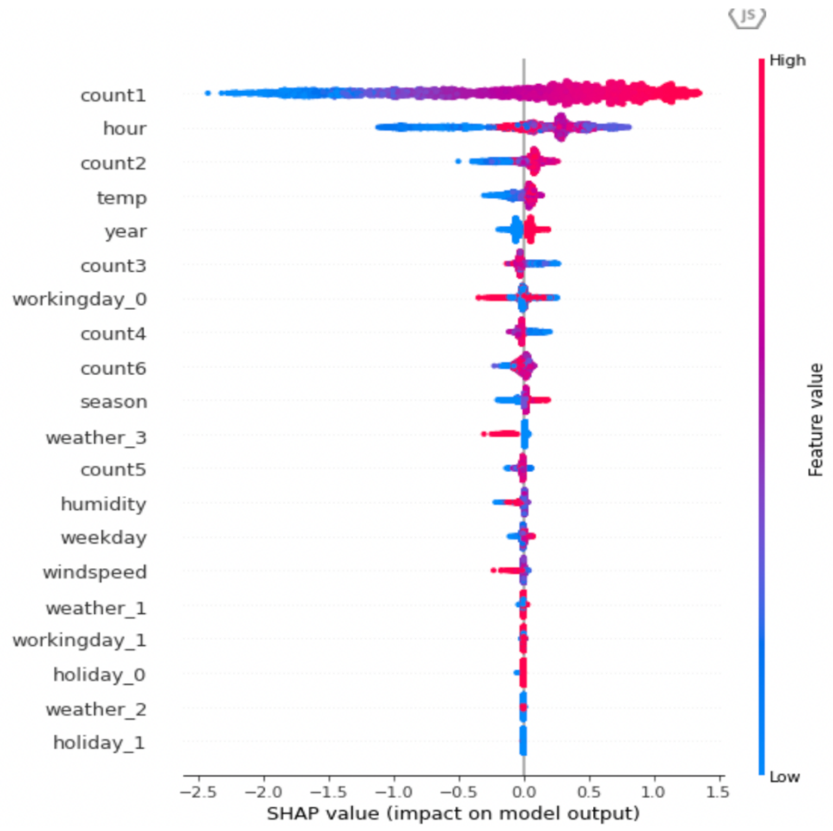
XGB-Gain:



Among all of these three types of plots, I found the count1(which is time lag of 1 hour) is the feature of the highest importance, which make senses to me because for example, if the previous is a busy hour, for example, people going to work or get back home, then it is very indictive of the trend for the next hour's demand for the sharing bike. Also, if it is a midnight, then the previous hours should be important in evaluating next hour's demand. This can be seen in XGB-Gain and XGB-Cover. Gain score implies the importance of the feature for generating the prediction, while the coverage score implies the relative number of observations related to this feature. The hour and other count_numbers also indicate that the trend of previous demand for the sharing bikes both for the permutating, XGB-Gain of the prediction importance, XGB-Cover of the coverage importance. There are also some important features like weather and temperature, which can be found in three of the graphs. That means people takes the weather and temperature into consideration when they want to rent the bike. If there is a huge storm or rain, then I think people will not consider to rent a bicycle back home. Also, if the temperature is super low, people will less tend to ride a bike. The least important feature is holiday1 for the XGB-Gain and the permutation test, I guess probably the holiday is relatively small to the total dataset, so it is not enough to make a huge impact for the prediction. The least important feature for the XGB_Cover is weather_2, which is Mist + Cloudy.

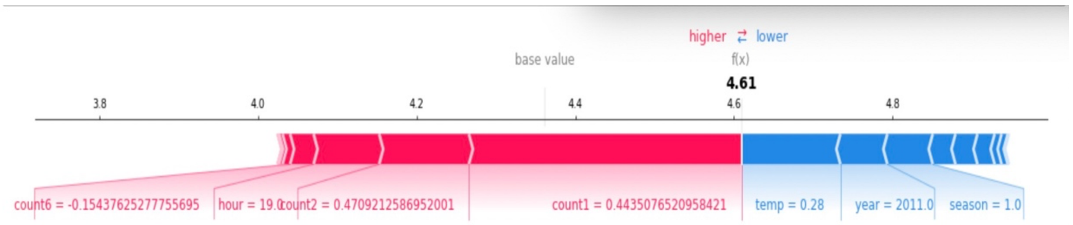
Local Feature Importance

SHAP-Value-Summary Plot



SHAP-Value-SummaryPlot

Point600



For the data point 600, this explainer says the most of the important features contributes to a higher demand for the sharing bike, while the year and temperature, season push this points to a lower demand for the sharing bike. To put in the context, this plot says when temperature is relatively low(normalize 0.28), this one-hour-time period will have less demand for sharing bikes. Also for the season and year. On the other hand, it says this hour period’s previous two-hour periods are strong indicators of increasing demand for the this one-hour-period.

Outlook

In order to improve the model, I will consider interactions between each features. Also, I should try to find a better way to make this dataset i.i.d because there will probably some correlations between each day, each season, etc. Besides this, I should try more hyperparameters for tuning the model and try to get more recent data points for different years to test the feature importance.

References

Dataset: Bike Sharing Dataset Data Set is from UCI machine learning repository: https://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset

[1].Maskara, V. (2021). *Analyzing Bike Sharing Demand | Vivek Maskara*. Vivek Maskara. Retrieved 7 December 2021, from https://www.maskaravivek.com/post/analyzing-bikeshare-demand/.

[2]. Srinivasan, V. (2017, May 3). *How to finish in the top 10 percentile in Bike Sharing Demand Competition In Kaggle? (part -1)*.
Medium. https://medium.com/analytics-vidhya/how-to-finish-top-10-percentile-in-bike-sharing-demand-competition-in-kaggle-part-1-c816ea9c51e1

GitHub repository
https://github.com/Francis958/Data1030-Final-Project