Sharing Bike Demand Prediction

Jiatao Yuan
Data Science Initiative
Brown University

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

December 9, 2021

Recap

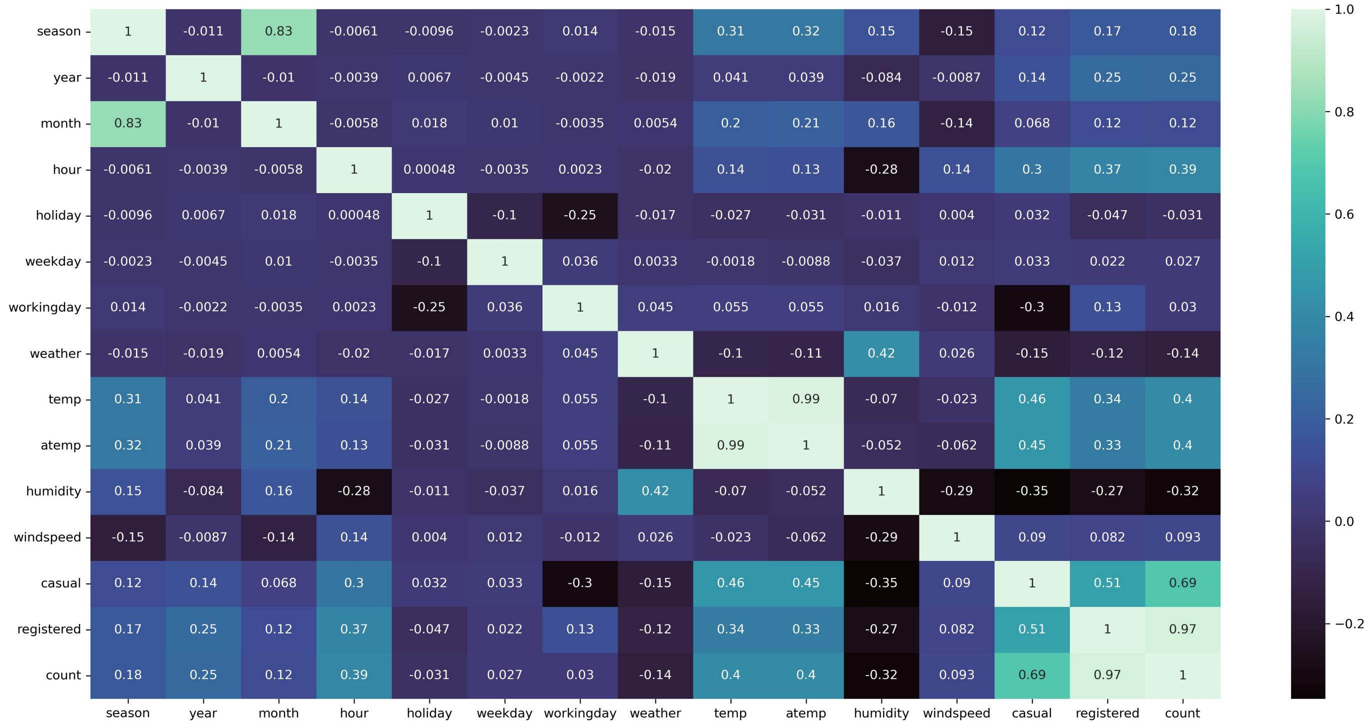
Intro

- The purpose of this project is to predict the demand for sharing bike per hour
- Regression methods were used
- Good sharing bike demand prediction can ease the traffic congestion and reduce the cost of the company
- Obtained from the UCI Machine Learning repository

Dataset Recap

	datetime	season	year	month	hour	holiday	weekday	workingday	weather	temp	atemp	humidity	windspeed	casual	registered	count
0	2011-01-01	1	2011	1	0	0	6	0	1	0.24	0.2879	0.81	0.0	3	13	16
1	2011-01-01	1	2011	1	1	0	6	0	1	0.22	0.2727	0.80	0.0	8	32	40
2	2011-01-01	1	2011	1	2	0	6	0	1	0.22	0.2727	0.80	0.0	5	27	32
3	2011-01-01	1	2011	1	3	0	6	0	1	0.24	0.2879	0.75	0.0	3	10	13
4	2011-01-01	1	2011	1	4	0	6	0	1	0.24	0.2879	0.75	0.0	0	1	1

Preprocessing And Exploratory Data Analysis

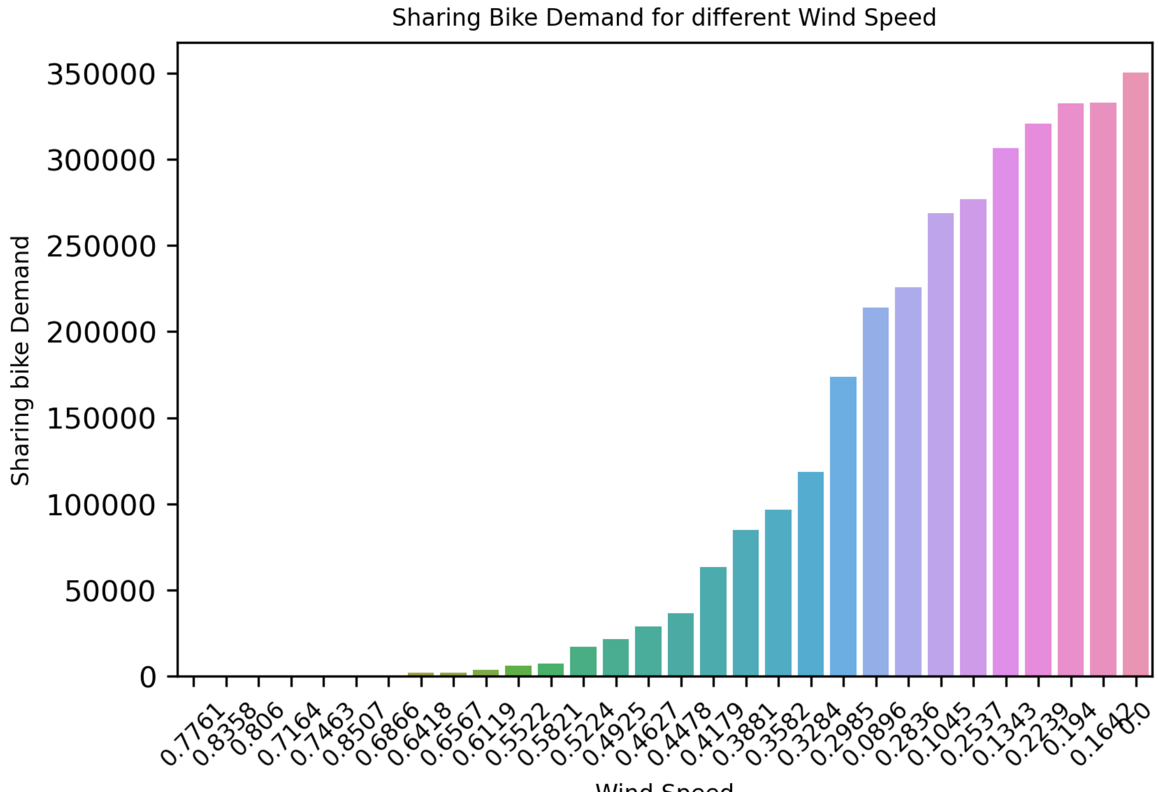


	season	year	month	hour	holiday	weekday	workingday	weather	temp	atemp	humidity	windspeed	casual	registered	count
season	1	-0.011	0.83	-0.0061	-0.0096	-0.0023	0.014	-0.015	0.31	0.32	0.15	-0.15	0.12	0.17	0.18
year	-0.011	1	-0.01	-0.0039	0.0067	-0.0045	-0.0022	-0.019	0.041	0.039	-0.084	-0.0087	0.14	0.25	0.25
month	0.83	-0.01	1	-0.0058	0.018	0.01	-0.0035	0.0054	0.2	0.21	0.16	-0.14	0.068	0.12	0.12
hour	-0.0061	-0.0039	-0.0058	1	0.00048	-0.0035	0.0023	-0.02	0.14	0.13	-0.28	0.14	0.3	0.37	0.39
holiday	-0.0096	0.0067	0.018	0.00048	1	-0.1	-0.25	-0.017	-0.027	-0.031	-0.011	0.004	0.032	-0.047	-0.031
weekday	-0.0023	-0.0045	0.01	-0.0035	-0.1	1	0.036	0.0033	-0.0018	-0.0088	-0.037	0.012	0.033	0.022	0.027
workingday	0.014	-0.0022	-0.0035	0.0023	-0.25	0.036	1	0.045	0.055	0.055	0.016	-0.012	-0.3	0.13	0.03
weather	-0.015	-0.019	0.0054	-0.02	-0.017	0.0033	0.045	1	-0.1	-0.11	0.42	0.026	-0.15	-0.12	-0.14
temp	0.31	0.041	0.2	0.14	-0.027	-0.0018	0.055	-0.1	1	0.99	-0.07	-0.023	0.46	0.34	0.4
atemp	0.32	0.039	0.21	0.13	-0.031	-0.0088	0.055	-0.11	0.99	1	-0.052	-0.062	0.45	0.33	0.4
humidity	0.15	-0.084	0.16	-0.28	-0.011	-0.037	0.016	0.42	-0.07	-0.052	1	-0.29	-0.35	-0.27	-0.32
windspeed	-0.15	-0.0087	-0.14	0.14	0.004	0.012	-0.012	0.026	-0.023	-0.062	-0.29	1	0.09	0.082	0.093
casual	0.12	0.14	0.068	0.3	0.032	0.033	-0.3	-0.15	0.46	0.45	-0.35	0.09	1	0.51	0.69
registered	0.17	0.25	0.12	0.37	-0.047	0.022	0.13	-0.12	0.34	0.33	-0.27	0.082	0.51	1	0.97
count	0.18	0.25	0.12	0.39	-0.031	0.027	0.03	-0.14	0.4	0.4	-0.32	0.093	0.69	0.97	1

(Figure of Correlation Matrix)

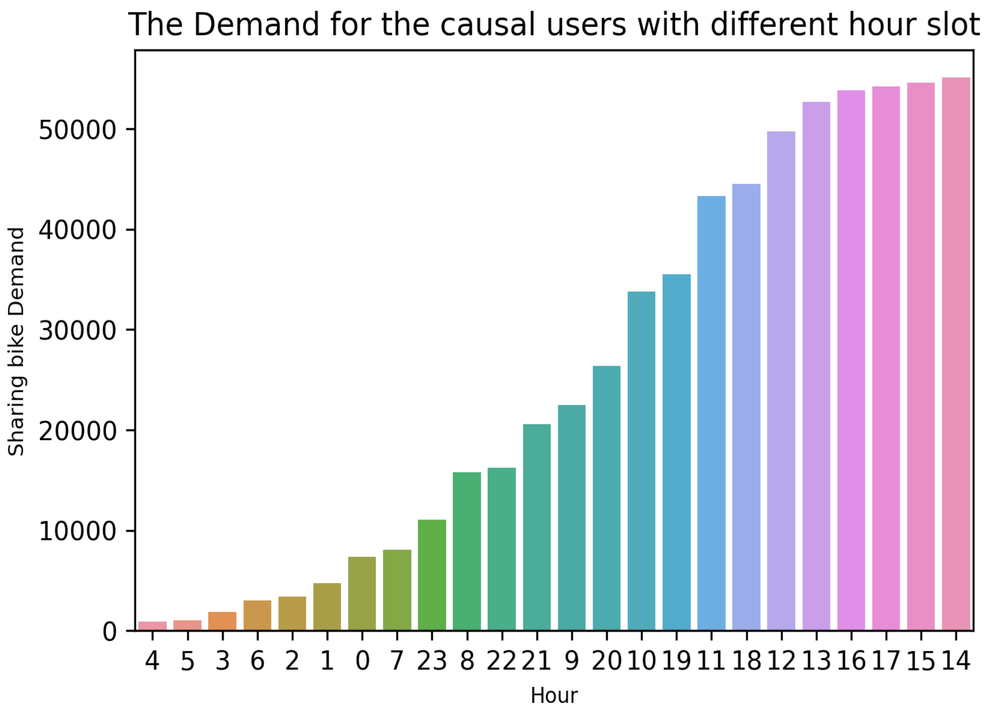
- Preprocessing: Casual, Registered and target variable(High correlations good or not?)
- EDA: Wind Speed and target variable(Low correlation bad or not?
- Business insights: Causal and Registered users with different hour slot

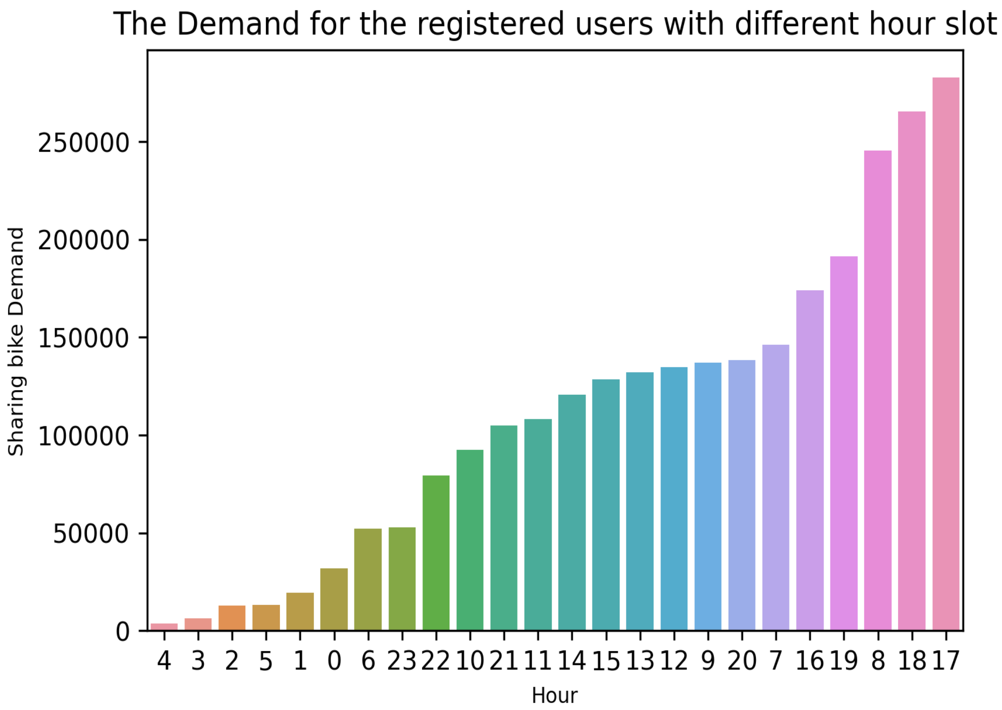
Preprocessing And Exploratory Data Analysis



(Figure of Hourly demand for different Wind Speed)

Preprocessing And Exploratory Data Analysis





Cross Validation

- Split the data:
  1. Since I add the 6-hour time lags for the dataset, now the data is i.i.d.
  2. I split the data into train, validation, test sets. Test set took up 20% of the whole dataset. Train and validation set took 80% and split into 5-Folds process.
- CV Pipeline
  - R^2 is the score
  - Ridge regression: Standard Scaler, One Hot Encoder, drop features of high collinearity, 5 folds, GridSearch and 5 random states
  - Random Forest, XGBoost, GradientBoost: One Hot Encoder, 5 folds, GridSearch and 5 random states
- Hyperparameters Tuning
  - Ridge Regression : L2 regularization term alpha is tuned
  - Random Forest: max_depth and max_features are tuned
  - XGBoost: alpha, max_depth, lambda, learning rate are tuned
  - GradientBoost: learning rate and max_depth are tuned

Results

- Model Performance

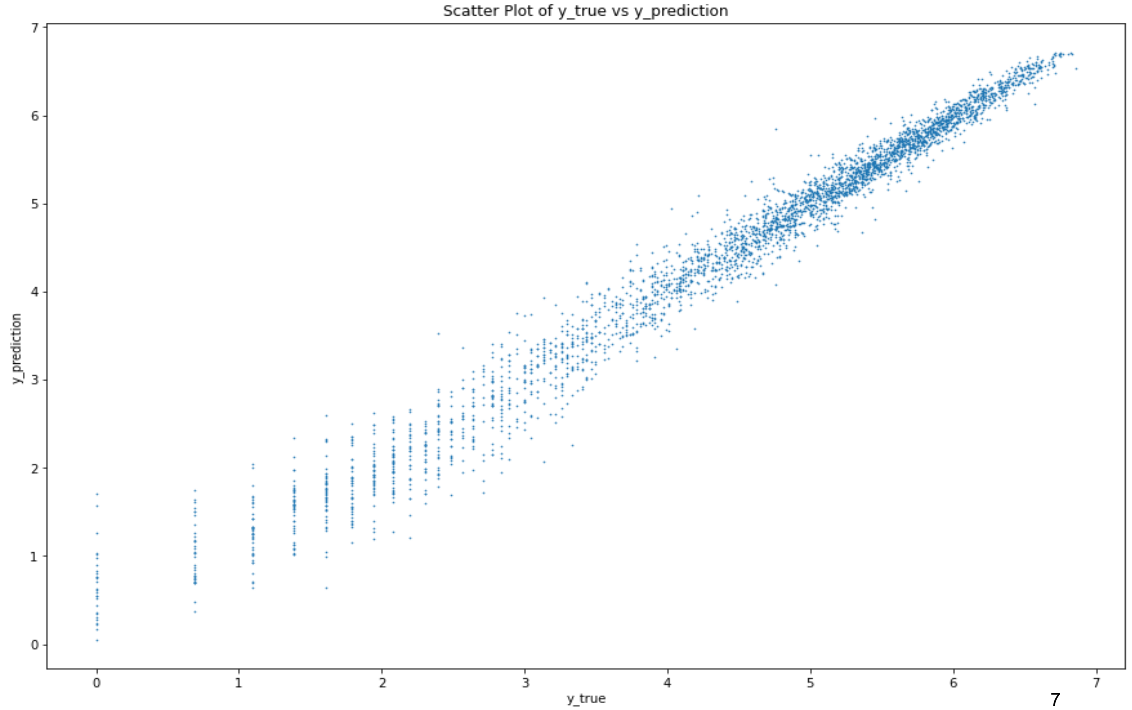
Model	Test Score
Ridge Regression	0.882+ - 0.005
Random Forest	0.958+- 0.001
GradientBoost	0.959+-0.001
XGBOOST	0.940+ - 0.002

- Baseline Model

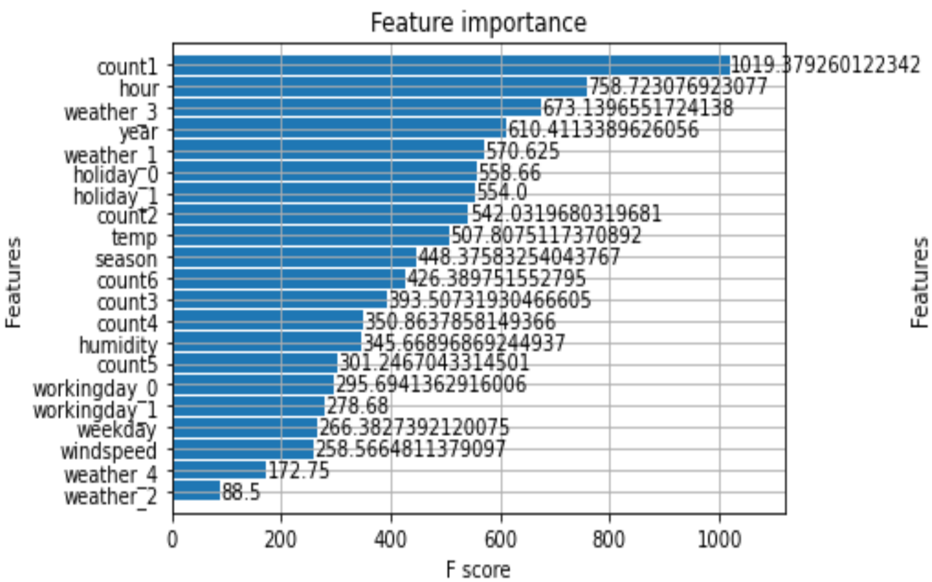
$$R^2 = 1 - \frac{RSS}{TSS}$$

Become Zero when the RSS equal to TSS

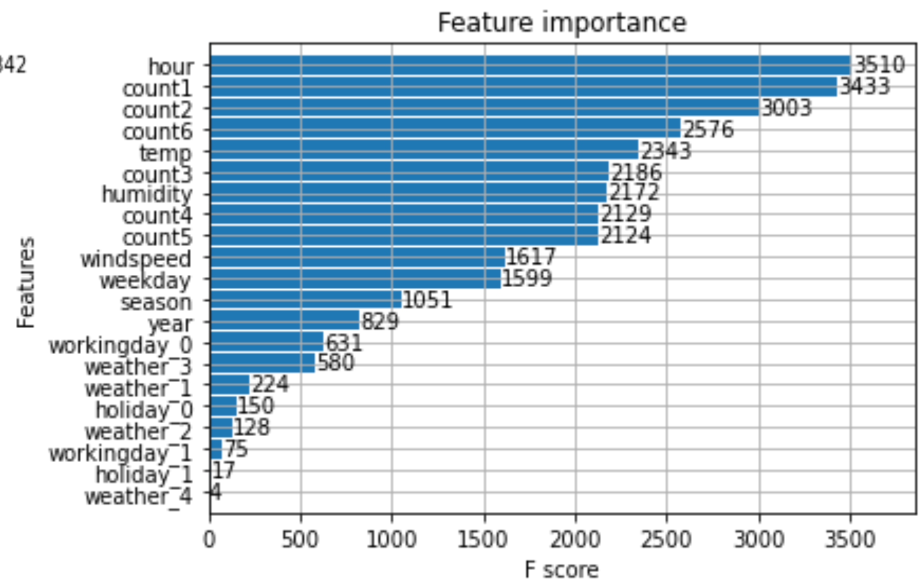
- Scatter Plot of y_true vs y_prediction
  - XGBoost



Global Feature Importance for XGBoost



Global Feature Importance of Weights

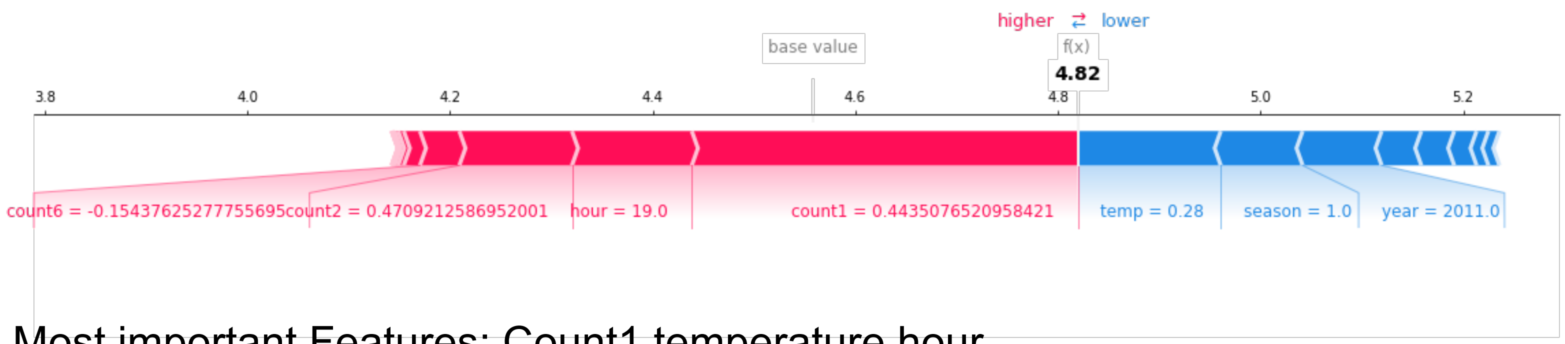


Global Feature Importance of Cover

Most Important Features: Hour, time-lag of 1,2,6 hour(Count1,2,6), weather3(Light Snow, Rain), Holiday, temperatures

Local Feature Importance for XGBoost

- Data Point 900



Most important Features: Count1,temperature,hour
Least important feature: count6

Outlook

- For models:
  - Tune parameters more precisely and have a better range
  - Collect more recent data points to make the predictions

- For features:
  - Consider more interactions between features
  - Collect more features including the volume of rainfalls, etc and see their feature importance.

Questions

Any Questions?