IE School of Human Sciences and Technology
Master in Big Data and Business Analytics
Python
Professor Jorge Lopez Lazaro

Python: Final Project

Group O-1-5

Rafael Hernandez
Hatem Abdelmowgoud Hassan
Fei Dai
Theodore Stephane L. Willems
Mrad Azoury
Emily Fuller
Muhammad Furqan

Table of Contents

Executive Summary

The report details steps taken to predict number of total bikers on hourly basis for the last quarter of 2012 for Washington D.C. bike sharing system. The dataset comprises of hourly data and daily data from 2011 to 2012, daily data is merely an aggregation of hourly data. For the purpose of analysis, only hourly data was used. Two-step approach was followed to compute predictions, first, level 0 models were used to predict for third quarter of 2012 and then stacked ensemble was used to predict for fourth quarter of 2012. Linear forest, XGBoost, Random forest regressor, and K nearest neighbor were used to compute Level 0 predictions, which were added to explanatory data. Further, XGBoost was trained on quarter 3 explanatory variables, which contained predictions of level 0 models, to predict for the test set. The final accuracy score is **0.897.** As the model is able to predict number of bike users with almost 90% accuracy, it can be used by Washington D.C. bike sharing system to forecast demand and manage inventory levels.

Exploratory Data Analysis

Data understanding is the key component of carrying out a prediction task. Therefore, naturally, interactive and statics plots are utilized to understand the dataset. Dataset is divided into two files, hourly data and daily data, the latter is merely an aggregation of hourly data. Hence, exploratory data analysis was carried out using hourly data.

Time series

Interactive time series plot (fig 1 in appendix) shows that there are seasonal variations in data. Number of total bikers increase Despite seasonal variations, there is a positive year-on- year trend in the number of total bikers.

Box plots

The distributions of casual, registered, and total bikers (fig 2 in appendix) show outliers. This finding proves that in the span of dataset, there were times when number of bikers were much greater than the overall trend. In order to enable the model to capture unexpected increase in bikers, multiple features are created, which will be detailed in the feature engineering part.

The boxplots for seasonal variables (fig 3 in appendix), namely, temp, atemp, hum and wind speed, show outliers for hum and wind speed. This finding, also, fed into feature creation for seasonal variables in order for the model to be powerful enough to capture such outliers.

Line charts

The line charts (fig 4 in appendix) show that number of total bikers are highly affected by weather. The seasonal variations for number of total bikers and weather follow an identical trend. Therefore, it is reasonable to expect that weather will play an important part in predicting number of total bikers.

Working Day vs. Holiday

Another important finding is that the total bikers are much higher on weekends compared to weekdays (fig 5 in appendix). The difference in number of total bikers on weekends vs. working days justifies that the data should be split and modeled separately. However, graph for total bikers on working days vs. holidays (fig 6 in appendix) is a proof that number of total bikers do not only increase on weekends, but also on holidays that are during the weekdays. Therefore, data is split between working days and holidays. Predictions for the two datasets are computed separately.

Data Cleaning

Two data cleaning steps are taken before running a baseline model:

- Converting Date to datetime64 format
- One-hot encoding of categorical variables

Baseline

Splitting the Data

The dataset is split into train and test set. The train set comprises of data from the first quarter of 2011 till the end of second quarter of 2012 and the test set comprises of third quarter of 2012. Predictions are made for the third quarter of 2012 in the baseline step. The explanatory variables neither include casual users not registered users. The target variable is total users.

Linear Regression

A simple linear regression is used to predict hourly bikers for the third quarter of 2012. Baseline model predicts the number of total bikers with an accuracy of 60%.

Feature Engineering

Flag for Daylight and Noon time

Astral module is used to calculate flags for daylight and noon time.

- A customized function is defined to classify a row as daylight. If the hour of a record is less than the hour of sunset in Washington DC and more than the time of sunrise, it is flagged as daylight, otherwise it is flagged as not daylight.
- Noon time flag is also created using a customized function. If the hour of a record is equal to the hour of noon in Washington DC, it is flagged as noon, otherwise it is flagged as not noon.

Relative Values

In exploratory data analysis, it was found that there are outliers in seasonal variables. In order to make a robust model that is able to predict outliers, new variables are created for **temp, atemp, hum, and wind speed.** In case of temp, mean of temp for the last seven days is deducted from current temp value and the resulting value is divided by standard deviation of temp for the last seven days.

Rush hour flag

The interactive time series shows that there are variations in casual, registered, and total bikers during the span of a day. This realization led to creation of a rush hour flag. The logic for this flag is as follows:

- Working Day:
  10:00 AM to 6:00 PM is flagged as **high rush hour.** 7:00 PM to Midnight and 8:00 AM and 9:00 AM are flagged as **medium rush hour.** Whereas, all other hours are flagged as **low rush hour.**
- Holiday:
  7:00 AM to 9:00 AM and 4:00 to 8:00 PM is flagged as **high rush hour.** 6:00 AM, 10:00 AM till 1:00 PM, 3:00 PM, and 9:00 PM till 11:00 PM are flagged as **medium rush hour.** Whereas, all other hours are flagged as **low rush hour.**

Mean of Total Bikers

Exploratory data analysis highlighted outliers in total bikers. In order to make a robust model that is able to predict outliers, new variable is created for total bikers. Mean of total bikers in the last three weeks for the same hour as the current row's hour is computed and added as a new variable to the dataset. This variable was created separately for working days and holidays as they depict different patterns.

Genetic Programming

A supervised algorithm that uses simple mathematical equations such as summation, multiplication, square root, etc. in order to find a relationship between the existing features and the target. It tries multiple combination of these equations and has a learning process which gets better with the number of generations it is set to have. This function added 15 features each to working days and holidays datasets.

Modelling

Holidays and Working days were modelled separately and later their predictions were combined. Initially it was decided to go with the same approach for both datasets, which is that a simple model is applied with a gridsearch to get the best possible parameters and see which parameters behave better. However, a further step was taken to improve the score. Stacking ensemble was used to understand errors of the first models and build on them. Expected results were obtained for the working days dataset but not for the holidays.

In order to stack ensemble, third quarter of year 2 was left aside and level 0 models were trained on the one year and a half. Predictions for third quarter were obtained from level 0 models and joined with explanatory variables. Thereafter, an ensemble was trained on third quarter in order to predict for the fourth quarter (test set).

Level 0 Models

Four regressors were tried on holidays and working days datasets for the first year and a half:

Models	Working Days	Holidays
Linear Regression	Q3: 0.9414 | Q4: 0.8838	Q3: 0.8333 | Q4: 0.8785
Random Forest Regressor	Q3: 0.9289 | Q4: 0.8715	Q3: 0.857 | Q4: 0.8496
XGBoost	Q3: 0.9344 | Q4: 0.898	Q3: 0.8879 | Q4: 0.8905
K Nearest Neighbors	Q3: 0.9211 | Q4: 0.8340	Q3: 0.8169 | Q4: 0.7577

N.B.: Support vector machine was tried for working days but it was very time consuming and computationally expensive, also the results were not impressive. Therefore, it was not tried for non working days.

Inspection of each model's result, led to the following insights:

- Every model yielded better scores for working days. The reason could that working days are represented by registered users, which means that there is a more predictable pattern
- Level 0 models' Predictions for Q3 had a higher accuracy than the accuracy for Q4. Again, this can be explained from the difference in behavior that was seen in winter compared to other seasons.
- Another unusual behavior observed is that linear regression was giving better results than Random Forest. This means that there is a linear relationship between the data at hand and the target, and that the assumption that a random forest would always give a better result is not true specially in a regression problem.

Stacking

Based off the result shown above, only the top 3 models were selected, namely, Linear Regression, Random Forest and XGBoost. Level 0 models' predictions for Q3 and Q4 were stacked with their respective explanatory variables datasets. A level 1 model, XGBoost, was trained on third quarter and predictions were computed for fourth quarter (test set). Following results were obtained:

Models	Working Days	Holidays
XGBoost	Q4: 0.901	Q4:0.858

Stacking ensemble was useful for working days as R2 score improved by 0.01, but surprisingly there was a drop in R2 for Holidays. Research showed that Stochastic Gradient Descent could be an appropriate level 1 model as it also acts on errors using lasso and ridge and elastic net penalty, also, it has learning rate. Infact, it gave a better score (0.88) than XGBoost but still not more then the level 0 XGBoost.

This means that the new predictions that were added to the initial dataset were not useful for Level 1 model and maybe led to overfitting in the training. The final decision was to keep the stacking ensemble for Working Days, which produced an R2 score 0f 0.901 but not for the Holidays. For holidays, only level 0 XGBoost was used that yielded a score of 0.8905. After combining the two predictions, the final R2 score was **0.897.**

Appendix

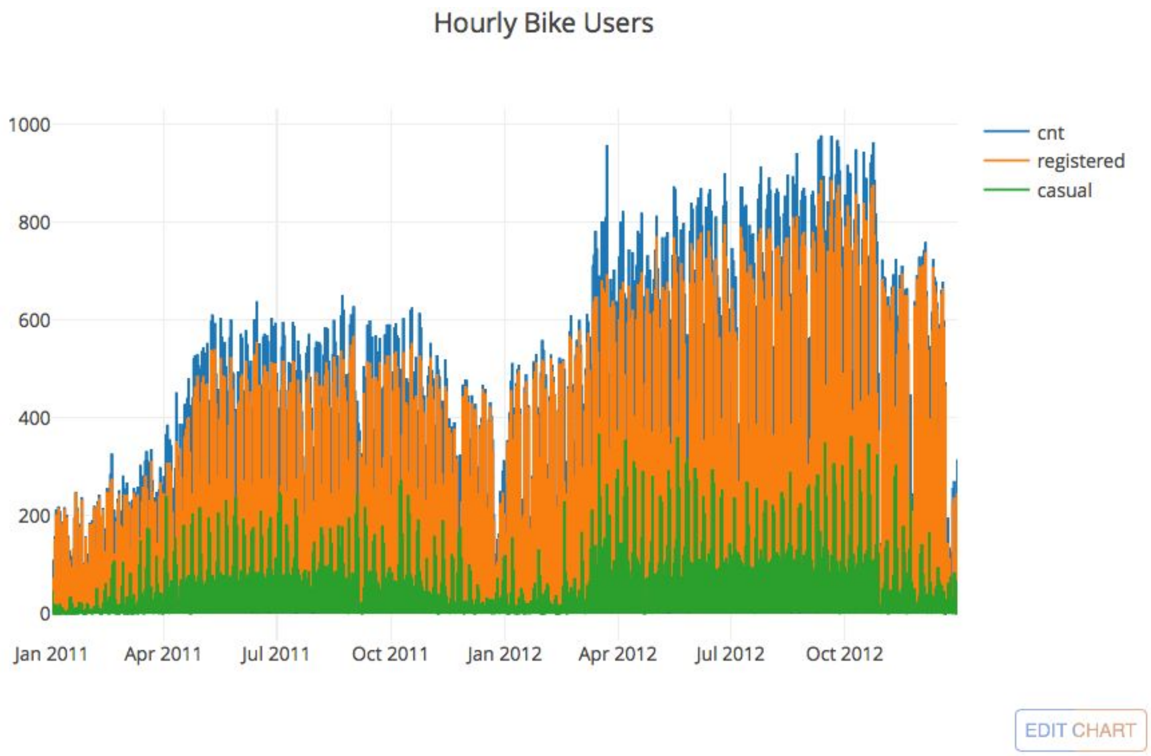

Fig 1

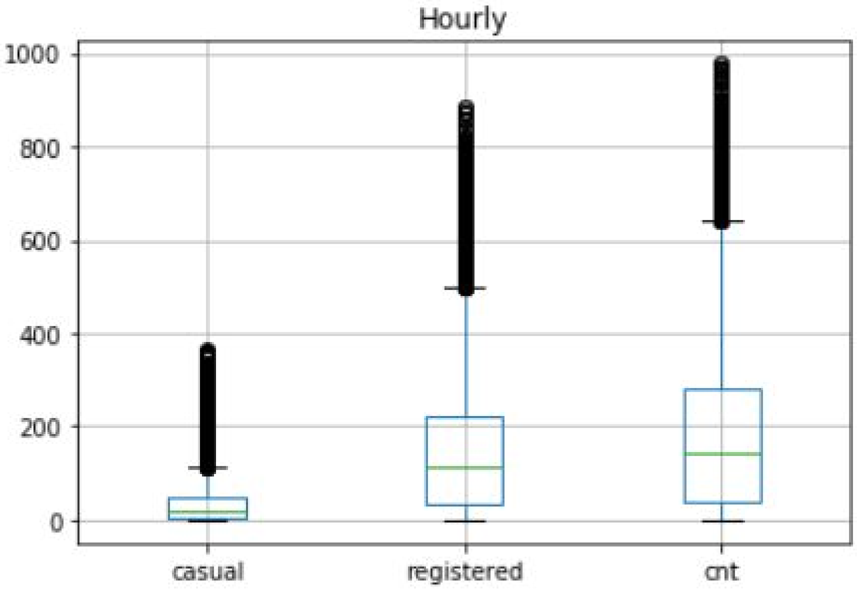

Fig 2

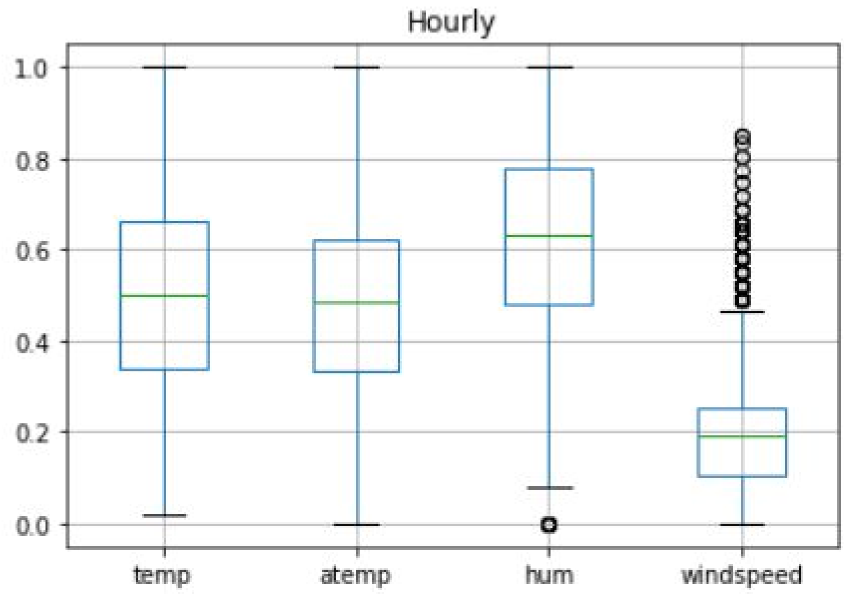

Fig 3

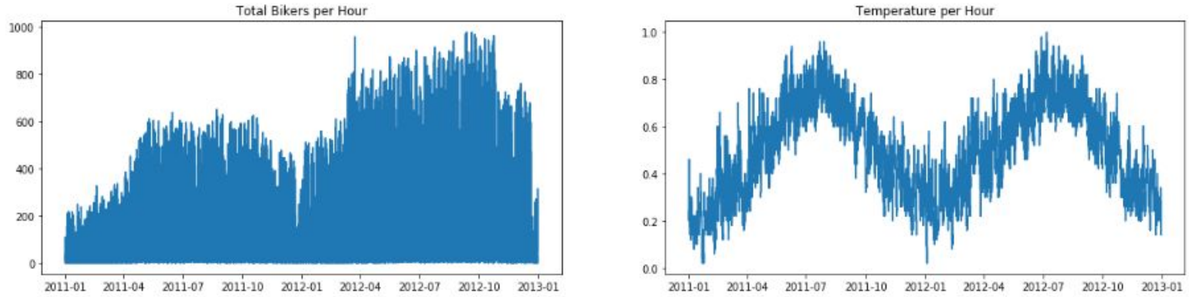

Fig 4

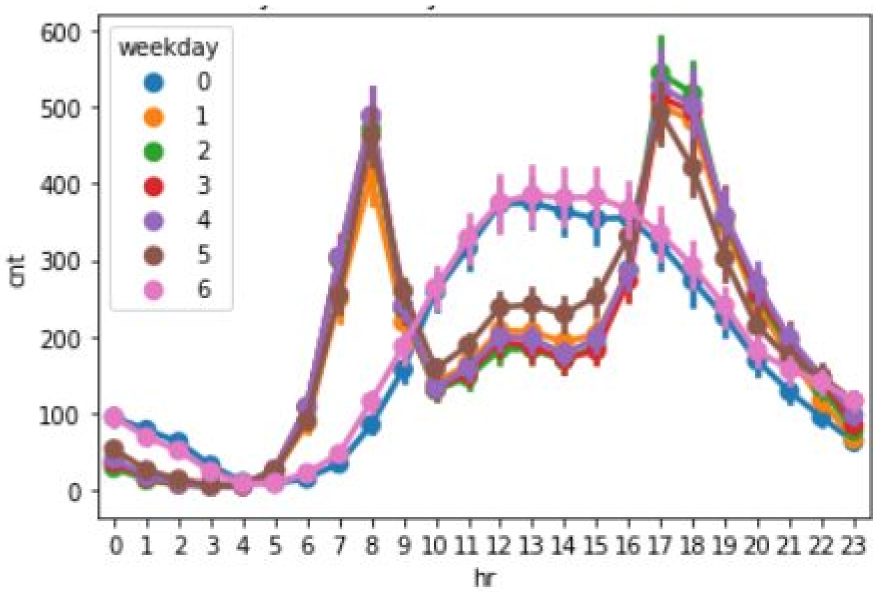

Fig 5

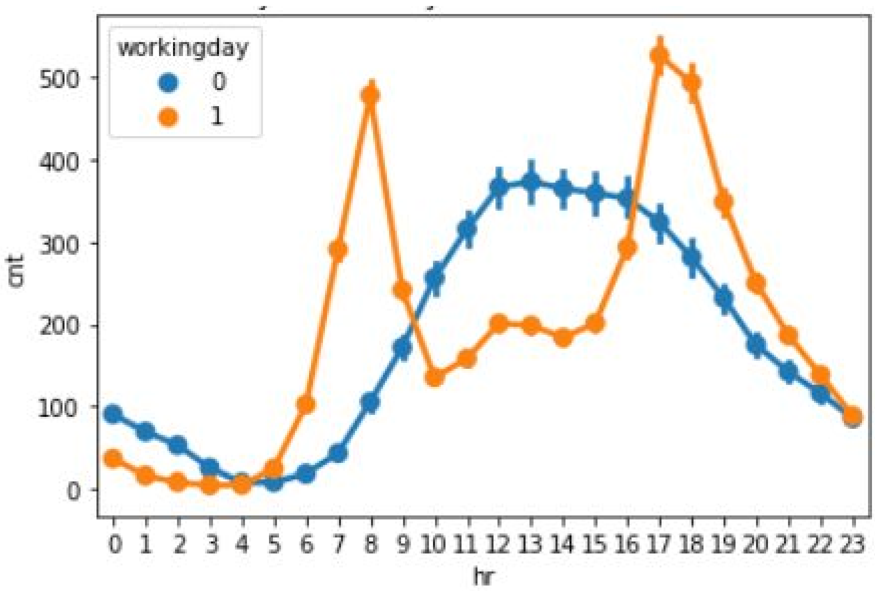
workingday
0
1
cnt
500
400
300
200
100
0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
hr

Fig 6