

EN.553.753  
Junzhou Lin

## Project 2[Detail is in the coding provided]

### Part One:

I have encountered two problems about the data during this process.

The first problem is when converting “Date(HourEnding)” to a data frame, the time variable will not be at an exact hour because it might have some milliseconds off. I created a function to round the time to an exact hour.

	2000-01-01				
Prior	04:59:59.712	After	2000-01-01 05	2000-01-01	05

The second problem is that I found even the length of two data frames are the same, but one is ending on 2018/08/08 7pm and another is ending on 2018/08/10 0am. With this been said, there are at least 29 rows of error data.

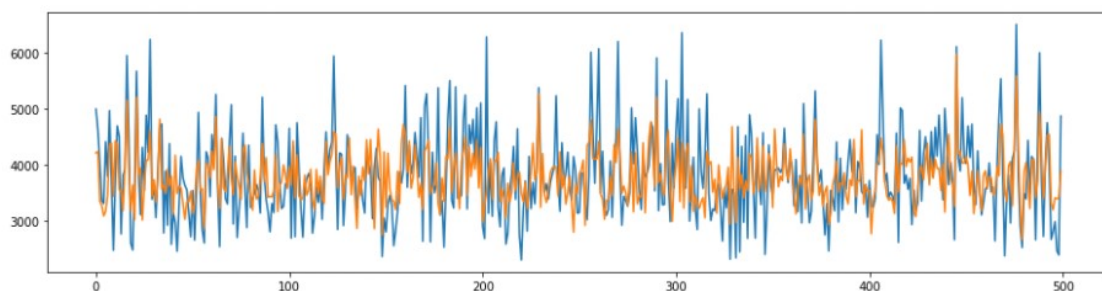
For the ‘Load’ data frame, there are 18 rows have repeated ‘Date(HourEnding)’ but with a different ‘Load’ variable, I just delete the repeated one.

	7247	2000-10-29 01	2000-10-29	01	2633.0
Prior	7248	2000-10-29 01	2000-10-29	01	2501.0
	7247	2000-10-29 01	2000-10-29	01	2633.0
After	7248	2000-10-29 02	2000-10-29	02	2481.0

For the ‘Temperature’ data frame, not every hourly data is provided. I add one hour for ‘Date(HourBeginning)’ to match with the ‘Date(HourEnding)’ then Merge two data frame on “Date(Hour)” so the whole row will be dropped if any data with a same ‘Date(Hour)’ is not provided in both data frame .

### Part Two:

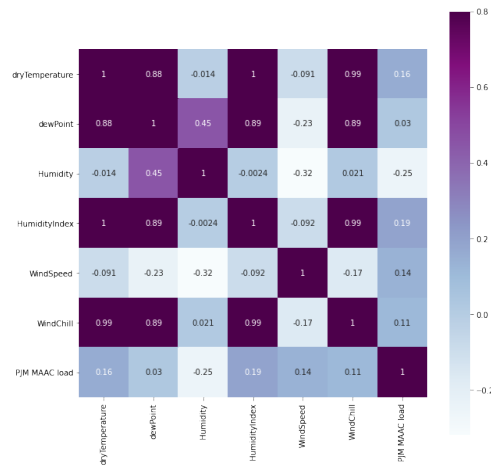
At the first try, I was trying to build the model through a linear regression without thinking too much. But only ended up with a 0.4047  $R^2$ .



After plotting the Load vs. Temperature graph, I realized that nonlinear regression model might be the correct approach to this project. The graph shows an quadratic trend since warm temperature in the middle usually have a small amount of load and the more extreme temperature on either side have a higher amount of load.

Ploting a heat map is also helpful to see the correlation between each parameters

and the load, since I want to consider the possibility of taking out useless (or not that useful) parameters.



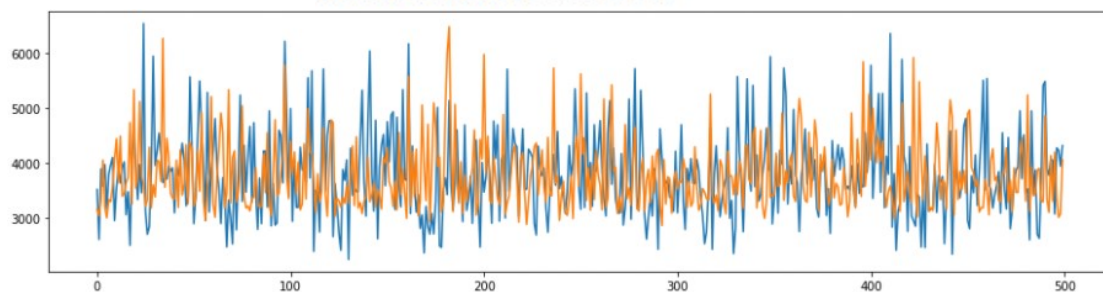
Part three:

With y as the “Load”, I tried different combination of parameters for x. In conclusion, the model with all parameters included will give the best result (highest R<sup>2</sup>)

Our model is built through pipeline with LASSO as a type of Regularized regression. Most importantly, I used Scikit-learn PolynomialFeatures for automatically generate polynomial features from a set of linear features, with degree range from 2 to 4. (Note: Quadratic function is obviously the most fitted function in this case by looking at the graph, I’m not sure why degree of 3 performs better than degree of 2 in our model) Higher the degree will ends up with a better fitted regression model but will consume enormously more amount of time. Degree of 4 is our best choice time and efficient wise.

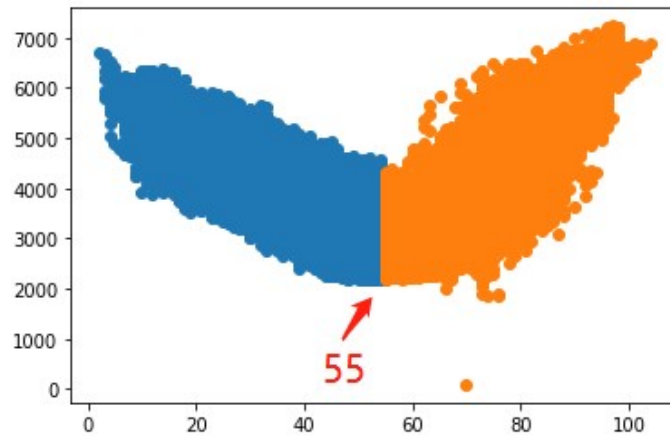
Following is our best fitted result: R<sup>2</sup>=0.6673 [With all parameters included]

```
explained_variance: 0.6673
mean_squared_log_error: 0.0162
r2: 0.6673
MAE: 378.6299
MSE: 215629.0633
RMSE: 464.3588
None
score0.6689827348047802
```

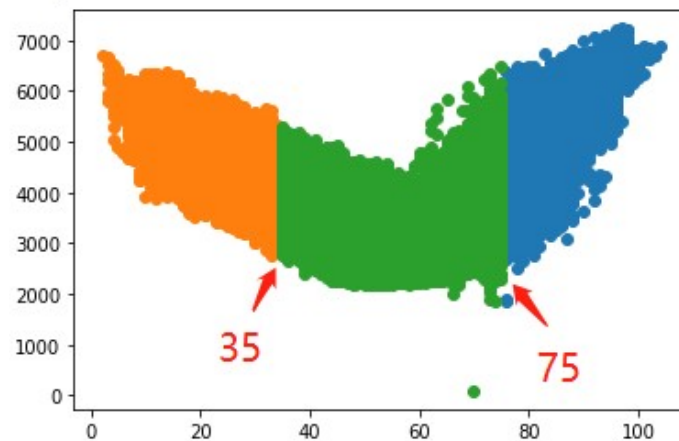
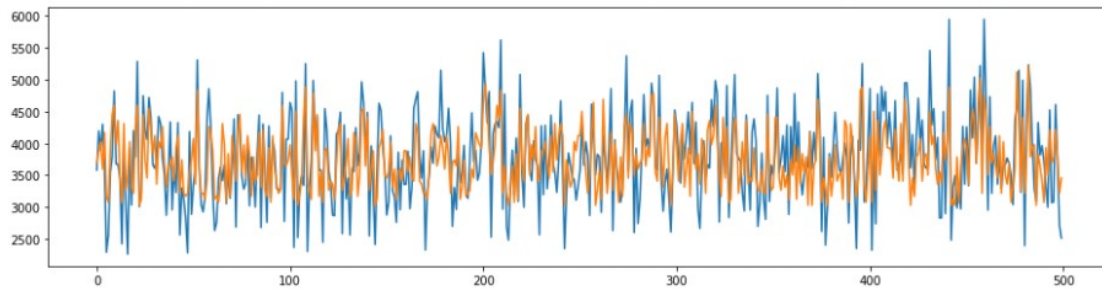


Extra:

Another way I have tried is to build models after split the dataframe on “Temperature”, so ‘splitted’ data frames would exhibit a different type of trend.



Splitting the data frame from the middle where drytemperature=55, so both data frames will exhibit an linear trend.  $R^2$  I got from this model is 0.5948 which lower than our best fitted model.



Splitting the data frame into three parts, a nonlinear trend with data that satisfy  $35 \leq \text{drytemperature} \leq 75$  and two linear trend with data that either drytemperature > 35 or < 75. Unfortunately, the result  $R^2 = 0.642$  which is lower than our best model.

