Data Mining - hw1 report

Q1. How do you select features for your model input, and what preprocessing did you perform?

Preprocessing:

While reading the raw data, originally, a process was required for every 18 lines read. To facilitate access, I stored the raw data in a 2D array of size 18 X 5760, where each row represents various different features, and each column represents data for each hour. Additionally, since some fields contain the symbols '#', '*', 'x', 'A', these values need to be converted to the floating-point number '0.0'.

Feature Selection:

Calculate the correlation coefficient between each feature and the PM2.5 feature, and filter out the features that have a higher impact on PM2.5.

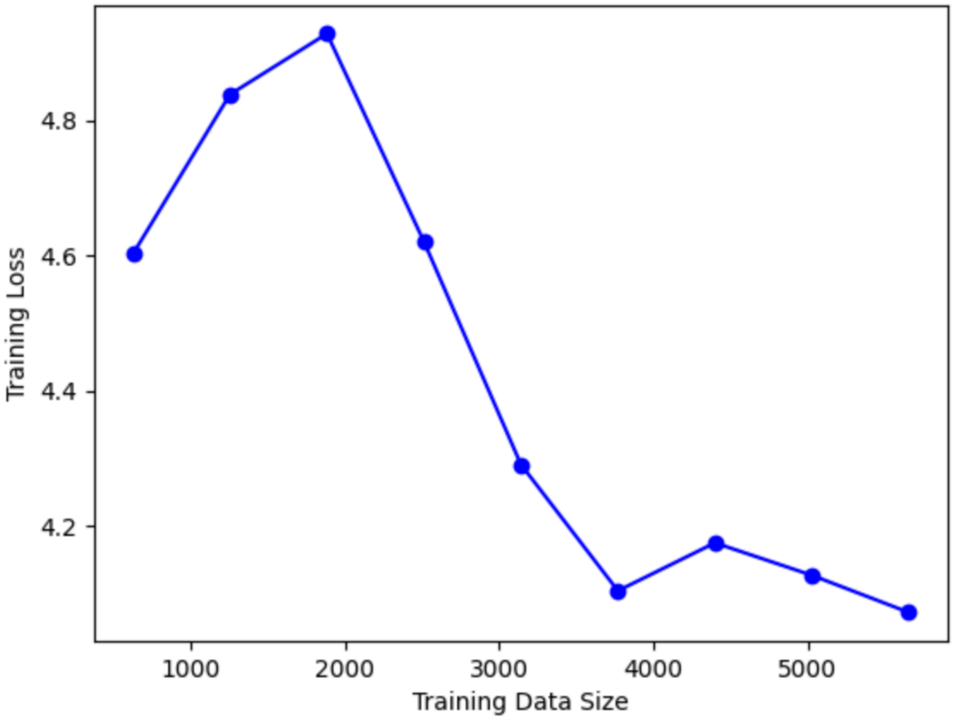
Feature	Correlation	Feature	Correlation	Feature	Correlation
AMB_TEMP	0.29124	NOx	0.46323	SO2	0.29789
CH4	0.21811	O3	0.10199	THC	0.30337
CO	0.60092	PM10	0.81630	WD_HR	0.01713
NMHC	0.51113	PM2.5	1.0	WIND_DIREC	0.01194
NO	0.23385	RAINFALL	0.07785	WIND_SPEED	0.14693
NO2	0.49946	RH	0.08467	WS_HR	0.16215

The table above shows the correlation coefficients between each feature and the PM2.5 feature. Orange represents the highest correlation coefficient, followed by blue and green.

In this assignment, I discarded all features with a correlation coefficient less than 0.3. Additionally, for features with a correlation coefficient greater than 0.5, a new feature of the square of the original feature will be added, and for those with a correlation coefficient greater than 0.8, a new feature of the cube of the original feature will also be added.

Q2. Compare the impact of different amounts of training data on the PM2.5 prediction accuracy. Visualize the results and explain them.

In this experiment, the learning rate was set to 0.5, and I let the model go through 50,000 iterations. When we look at the line graph below, the horizontal axis shows us how much training data I'm using, and the vertical axis tells us about the Training Loss, which is basically how well the model is doing — lower numbers are better.



From the line graph, it can be observed that, fundamentally, the larger the volume of data, the lower the Training Loss should typically be. This is expected as a model with more data often generalizes better, avoiding the pitfalls of overfitting.

Moreover, an interesting observation can be made: in the beginning, as the data size gets bigger, the training loss gets higher. This scenario could be indicative of overfitting to a very small dataset. Initially, the model might show a low loss, performing well on a dataset of limited size by learning its specifics rather than general patterns. However, as the volume of data increases, the model's ability to generalize could decrease momentarily, resulting in a higher training loss.

Q3. Discuss the impact of regularization on PM2.5 prediction accuracy.

In this experiment, I kept everything the same as I did in experiment Q2 — I didn't change the model or any of the settings, except for one thing: the *lambda_value*. I experimented with a total of four different *lambda_values*.

The table below shows the Training Loss resulting from experiments with different *lambda_values*.

Lambda Value	Training Loss
0	4.065024
1e-8	4.065024
1e-6	4.065036
1e-4	4.066253
1e-2	4.186076

From the table above, it clearly indicates that the implementation of regularization techniques did not yield the anticipated improvements in model performance.

This could be attributed to the nature of the features selected for the model. It's quite plausible that these features were already quite general or perhaps too simple, leading to a scenario where overfitting was not present in the first place.