CS 174 Project Milestone - Air Quality Prediction

Hanli Li, Yue Song, Ziyang Wang

Problem Recap In this project, we are trying to predict the levels of air pollutants on an hourly basis. There are 39 dependent variables, each represents the level of a certain kind of air pollutant at a certain location. There are 5 predictor variables: time chunk of a year, month, weekday, hour, position within each time chunk. We build our predictive model based on supervised neural network. One thing worth mentioning is that, although we are provided with the information of weather condition, air pressure, wind speed, etc., we do not include these into our feature set. There are two reasons for this, the first is that these information is missing for a majority of our training examples and testing examples, the second and more important reason is that we are trying to build a predictive model. At a future time point, we do not know the natural conditions, but we still want to predict the levels of air pollutants

Current Progress Within each time chunk, air pollutants levels were recorded for 11 consecutive days on an hourly basis. However, we only have access to the first 8 days. So we decided to split those 8 days into two parts: the first 6 days as our training set, and the following 2 days as our testing set.

The first challenge is to deal with missing values. Among those 39 response variables, only 14 have their missing value proportion under 40%, 19 of the remaining respectively has a missing value proportion of at least 75%. Referring to the correlation between response variables, we find that a lot of those with a high missing value proportion are correlated with the 14 low proportion ones. We think the 14 set contains enough information of the complete 39 set, so we train our model with the 14 set as dependent variables and use linear regression to impute missing values before training.

The speed of neural network training is directly related to the number of hidden layers and number of units within each layer. With 14 outputs, we first tried one hidden layer with different number of hidden units. Our training set has approximately 30,000 examples. The training speed was slow and we failed to get successful training results. After multiple unsuccessful trials, we further split the 14 set into two where one has 8 of the 14 and the other has the remaining 6. The split was based on the correlation matrix. Variables in the 8 subset are relatively correlated with each other than with any of those in the 6 subset, and vice versa. Then we trained one neural network on each of these subsets. With

one hidden layer, each produced successful training result. With two hidden layers, the training was also successful, with smaller errors, but more steps for the algorithm to converge.

Upcoming Work First of all, we need to find a better method to impute missing values other than linear regression. Second, instead of choosing dependent variables based on correlations, we need to use dimension reduction algorithms such as principle component analysis, k-nearest neighbors, and so on. Furthermore, we need to tune the parameters of our neural network model, this includes the number of hidden layers, number of units in hidden layers, etc. Finally, we need to evaluate the performance of neural network model on testing set.