April 12, 2018

Partner Report - Milestone 3
Keyan Halperin, Christopher Hase, Justin S. Lee, Casey Meehan
Harvard IACS

Ben and Christine:

We are reaching out to deliver some updates on our pollution prediction capstone project. We have made some key additions since our last milestone report, and want you to have a sense of where we are headed in the last month of the semester.

In our last report delivered on March 9, 2018, we expressed our intention to investigate effective imputation methods, as well as further our usage of high-performance computing resources, such as Harvard's Odyssey cluster. To respond to the need for a more effective imputation scheme, we have experimented with a variety of imputation routines. The 'missForest' imputation algorithm has proven most promising, and so we have assembled the scripts needed to impute the roughly two decades of $PM_{2.5}$ data. Our final revision of our imputation routine has nearly completed on the full dataset, currently configured to run on Odyssey.

In addition to more advanced imputation, our group has developed a sophisticated prediction model. Having seen strong temporal correlations in the $PM_{2.5}$ data, we've built a convolutional neural network that utilizes these temporal dependencies for all dynamic features.

Looking forward, Pavlos and David have turned our attention to the prospect of quantifying uncertainty using Gaussian processes (GP). We are currently looking into applying GP techniques to increase our prediction accuracy, reduce model overfitting, and estimate the uncertainty of our predictions.

## I: Current Stage

The two areas we have made the most progress in since our last milestone report are (1) missing data imputation and (2) modelling $PM_{2.5}$ via a convolutional neural network (CNN). We describe our progress in each area below:

(1) Missing Data Imputation

As mentioned briefly in the introduction, we have been using the 'missForest' method for missing data imputation, which involves an iterative process of constructing a random forest for each column of data to then fill in missing values. More information on this method can be found here.

Previously, when we tried utilizing the original R implementation of missForest, we determined that this version used too much memory and was too slow, even when given significant resources on Harvard's Odyssey computing cluster. Thus, we decided to use a Python implementation (`predictive_imputer`) of missForest which used models from Python's scikit-learn package. We believed that the scikit-learn models would be better able to handle large data sizes. Although lacking in documentation, it seems from our testing that `predictive_imputer` has the same functionality as the R missForest package.

One decisive advantage in using the Python implementation is that we can save a trained imputation model as a binary file using Python's `pickle` module, then load it again at a later time to impute new data. Because of this, the trained model can be used for online imputation as new data is received, so long as the dataset used to train the model is representative of the new data. In contrast, the R implementation of missForest does not create a model that can be used for further imputation; it simply returns an imputed dataset.

(2) Convolutional Neural Network (CNN)

We have built a novel CNN architecture using PyTorch that allows us to use information from days before and after a day we are trying to predict $PM_{2.5}$ for. In particular, the architecture involves applying convolutions only to variables that change over time within a $PM_{2.5}$ sensor sequence while still using variables that are constant within the same sequence. We are planning on improving input features and tuning hyper-parameters via cross-validation in order to improve the accuracy of the model.

## II. Planned Next Steps

In the coming weeks, we will train our models on the fully imputed data set and compare the prediction performance with the results we obtained on the 1% subset. We also plan on using Gaussian process regression since it may be effective for geographic and temporal interpolation, and it can allow us to quantify the uncertainty of our predictions. In addition, we are also considering using the output from a Gaussian process as an input to our CNN.

Please let us know if you have any suggestion or concerns regarding our project.

Sincerely,

Keyan, Chris, Justin, and Casey