AIR POLLUTION PREDICTION: TIME SERIES CLUSTERING & ARIMA

Aditya Upadhyay
North Carolina State University
adupadhy@ncsu.edu

Daxkumar Amin
North Carolina State University
dkamin@ncsu.edu

Nilay Kapadia
North Carolina State University
nkapadi@ncsu.edu

Priyance Mandlewala
North Carolina State University
pjmandle@ncsu.edu

Vaibhava Laxmi
North Carolina State University
vvaibha2@ncsu.edu

Vishrut Patel
North Carolina State University
vnpatel@ncsu.edu

***Abstract*—** Prediction of spatio-temporal data has been one of the major challenges in creating a good predictive model. There are many different approaches which have been used to create an accurate predictive model. Primitive predictive machine learning algorithms like simple linear regression have failed to produce accurate results primarily due to lack of computing power but also due to lack of optimization techniques. Recent developments in deep learning as well as improvements in computing resources has increased the accuracy of predicting time series data. However, with large spatio-temporal data sets spanning across various states, employing regression models on the entire data can cause per state and per date predictions to be corrupted. In this work, we look at dealing with pre-processing the times series. However, pre-processing involves a similarity measure, we explore the use of Dynamic Time Warping (DTW). K-means is then used to classify the spatio-temporal pollution data of 47 states in the United States over a period of 16 years from 2000 to 2016.

I. INTRODUCTION

Pollution has plagued every city worldwide and continues to impact daily life and functioning. This is exacerbated by concerns of temperature rise, global warming and rise in sea levels. Several international organizations such as the Paris Climate Agreement initiated by the United Nations aim to reduce the effect of pollution. Moreover, nations are dedicated to developing techniques to reduce pollution as well as methods to reduce the usage of fossil fuels. The United Nations primarily stress that it is important to predict the carbon footprint of each nation. This is necessary to initiate policy decisions and regulations which need to be implemented by the respective country to curb the ill effects. This consequently stresses on the development of reliable methods to predict the pollution levels in a country over a period of time as well as to predict the levels of different pollutants for targeted solutions. Countries deploy many sensors to record different pollutant levels in urban areas as well as near industrial zones, but the main index used by governments to depict the pollution levels is the Air Quality Index. This is an important measure as it helps to determine the overall quality of air which consequently is used to determine the adverse health and climate effects which are caused to the environment. In this work, we present a spatio-temporal prediction model which could be highly effective in determining the AQI as well as individual pollutant levels over a period of time. To overcome the limitations posed by large-scale data and high variability we adopted a unique combination of approaches to predict, accurately, the different pollution levels as well as the AQI. We conducted several experiments using different models and determined a low cost-complexity combination of models.

II. RELATED WORK

There has been extensive research on developing highly accurate spatio-temporal models using different machine learning approaches. This section emphasizes on 3 approaches we considered before choosing the appropriate model for our work. The 3 approaches we considered are: Autoregressive integrated moving average(ARIMA), Long-short term memory coupled with Recurrent Neural Networks, Support Vector Regression.

A. LSTM(RNN)

The primary purpose of LSTM is to learn long-term dependencies in the data and prevent the long-term dependency problem i.e. preventing the model from "remembering" the data over a long period of time. This method is coupled with Recurrent Neural Networks with a few changes in the Recurrent nodes of the RNN. The main change in the repeating node is that instead of one neural network layer, there are four layers which interact with each other. These four layers are themselves used in a "recurrent" way using gates to allow/disallow information through them.

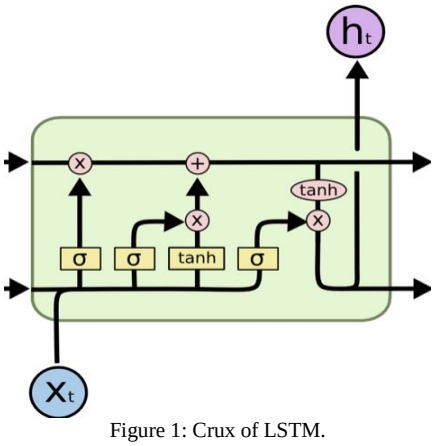

Figure 1: Crux of LSTM.

The four layers are: Three sigmoid layers and a tanh layer. The figure above shows how the combination is used in a node of a RNN. We tested this model on our data and obtained a RMSE of 3.378 using 30 hidden layers and an epoch of 5. This initial estimate is a good indicator of the accuracy of the model. Increasing the number of epochs would further optimize the RMSE but due to lack of computing resources, the predictions associated with it have not been calculated. The graph shown below is a rough estimate of how accurate the model is over the data. The blue graph represents the true points of the data whereas the orange graph shows the predicted values of the model.

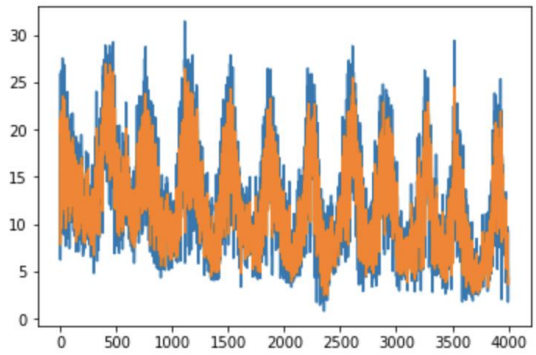

Figure 2: Results from LSTM

B. ARIMA

Autoregressive Integrated Moving Average model is a short-term (at least 40 data points) time-series prediction model which can be used primarily for predicting data which has a low variance or fewer outliers and tends to follow a stable trend. This model is most suited for data which shows a high-level of seasonality. In case a lack of seasonality, there is a high chance that the calculations associated with the model will not be computed due to certain constraints. We tested this model on our data with the NO2 mean, and time attributes from all states with lag value of 5 and obtained a RMSE of 2.006. While this value is small, there is a significant issue to be considered. For this, the data is the mean value of pollutant level for a particular date across all the states in US i.e. 5840 data points. For a particular date, every state will result in the same prediction. This provided a motivation to rethink our prediction methods.

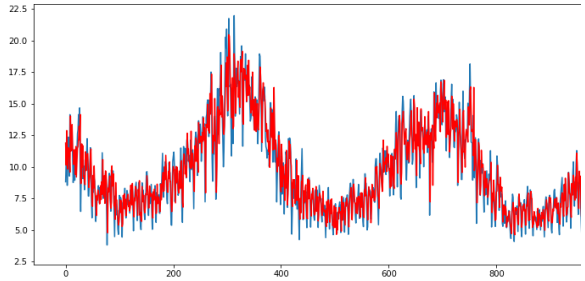

Figure 3: ARIMA results on the entire data (47 states, 17 years)

III. METHODOLOGY

The knowledge of similarity between time series is widely used for speech recognition and signature recognition. In our project, we make use of two pieces of knowledge - factors influencing pollution and seasonality observed in every year between 2000 - 2016. With respect to these concepts we determine the similarity between time series of multiple states and the similarity between time series of the 192 months in the years 2000 - 2016. Please note that we have worked largely with NO2 data as this has been seen to be the cause for lung diseases.

Through reviewing papers such as [1], we understand that air pollution in one state is affected by topology, climate and industrial activities in adjacent states. By merely fitting a regression model on the entire data across U.S, we destructively allow unrelated states to influence the regression of pollution in another state. Therefore, to overcome such a deterrent influence, we delve into clustering air pollution series of individual states. The resulting clusters combine those time series that are similar, i.e. they would be topologically, topographically and climatically similar. This

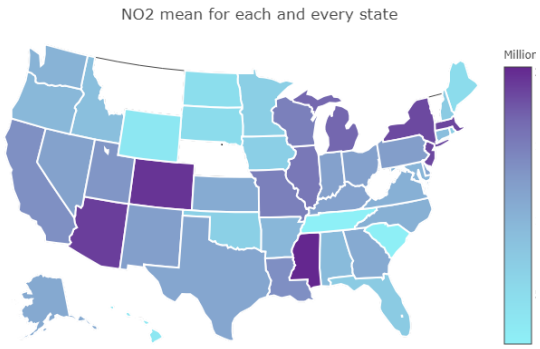

Figure 4: NO2 mean distribution generated from our dataset.

effectively deals with the spatio-temporal behaviour of the data.

While viewing the data collected, we had also noticed a seasonality in the data. On noticing the recurring "U's", the position of the peaks and falls, we hypothesize that this trend is similar to the effects of pollution due to the seasons in a year [2]. This was corroborated when classification carried out. This information can then be fit onto regression models such as ARIMA to predict climate based pollution statistics.

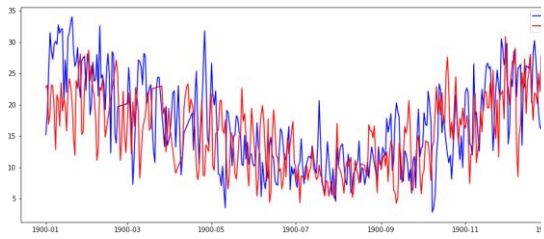

Figure 5: Seasonality on Arizona County 19 (NO2) for the years 2000 and 2001.

As per [3], merely using Euclidean distance results in no weight for phase shifted time series. For instance: if two-time series are T0: 1213110 and T1: 8121311, then with Euclidean distance, the distance between the two is calculated piecewise. (1 and 8), (2 and 1), (1 and 2) etc. However, as it is noticeable, the two-time series differ only by one position. This doesn't make the series as distant as Euclidean distance concludes it to be.

Dynamic Time Warping, a matrix N*N is created with the squared distance from one in the first, time series to every point in the other time series. With the above example, the matrix below is received. Every element is (t0 - t1)^2. With the help of this matrix, those distance elements are chosen such that the sum of the alignments is the minimum sum. If that is the case, the highlighted elements would be chosen. This way, the phase difference between the two series does not contribute to the distance. However, it is biased towards reducing the aforementioned effect and compares the last data point of a time series to the first of another. Hence rules such as Boundary conditions - restricting the alignment derived from the matrix to begin and end at the diagonal ends of the matrix, continuity conditions - restricting the number of elements compared with to find the shortest distance, monotonicity condition - making sure the points compared are spaced in time from the last iteration.

t_i/t_0	1	2	1	3	1	1	0
1	0	1	0	4	0	0	1
1	0	1	0	4	0	0	1
3	4	1	1	0	4	4	9
1	0	1	0	4	0	0	1
2	1	0	1	1	1	1	4
1	0	1	0	4	0	0	1
8	49	36	49	25	49	49	64

Figure 6: DTW Matrix

IV. IMPLEMENTATION

The first step for implementation is to pre-process the data available for training the model. The dataset available from Kaggle contains a total of 1.4 million data tuples with over 28 attributes. The high dimensionality had an adverse impact on the model building. The attributes included pollution level prediction for 4 different pollutants. Thus we trimmed down to one pollutant - NO2. The same model can be applied to different pollutant attributes as per requirement. Also, the data provided is for 16 years with hourly values for all the days. We trimmed this down to one value per day which will be mean of all values recorded for the day. This reduced the dimensionality of the time-series data and made it possible to train the model given our hardware and time constraint. For geographic attributes, the dataset contained information for 47 different states with multiple counties within each state. We generalized the data to the state level to ensure that we can identify similarity in data patterns for nearby states, hence reducing geographic dimensionality.

Our initial approach involved using Support Vector Regression (SVR). We implemented the Support Vector Machine algorithm and tried to extend it into SVR. The algorithm to generate possible hyperplanes, we used the approach same as SVM, but with the change that the hyperplane with maximum data points in given margin was selected. This hyperplane hence is the SVR for the data. Due to mathematical complexities that we could not get through with, we switched to the implemented method. Nonetheless, implementing SVR using python libraries, we were able to successfully validate that SVR is one of the correct ways to carry out the objective of the project.

Post-pre-processing, the next logical step was to establish correlation among different levels of data in context of time-series' present in the data. Initial experiments were focused on discovering time-series' at the '*month*' level of the data, for each day of the given month. The graph shown below is for a sample of the data. The graph clearly indicates the extreme variance and lack of correlation between different values. This consequently shows a definitive lack of seasonality for a given month.

To resolve this issue, we analysed yearly time-series' of different states and established that there is noticeable correlation between time-series' of different states. Thus, to increase intra-cluster correlation, the time-series' of different states which are similar to each other are merged.

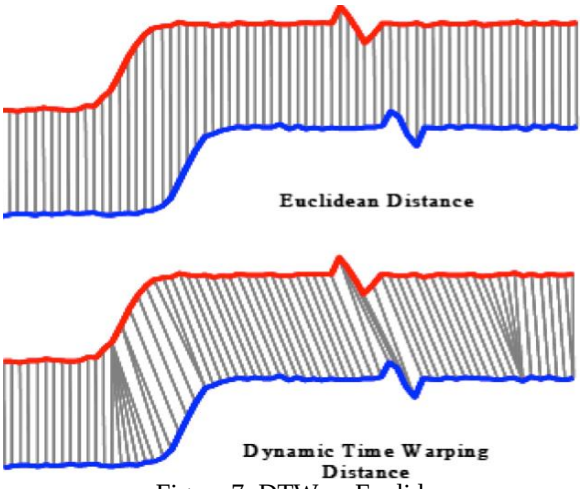

Figure 7: DTW vs Euclidean

After obtaining the time series data, we need to find alignment to determine the distance between two, time series to form clusters. Euclidean is one of the most common methods used to determine the distance, but it is not effective for time series data. Hence, we use weighted Dynamic Time Warping to calculate the alignment between any two given time series. Dynamic time warping finds the optimal non-linear alignment between two, time series. To quantify this result, we calculated the alignment between the time series' of 2000 and 2001. After applying both the methods, the results are:

Euclidean distance = 125

Dynamic Time Warping (window size = 10) = 73

The results were then confirmed with multiple iterations and the change in distance was observed. This led to the conclusion of using DTW over Euclidean distance.

To improve the performance of DTW, LB-Keogh method is used. For large datasets, as in our case, LB-Keogh makes retrieval of time-warped time series feasible. It provides a lower bound to the DTW method. With DTW as a suitable method to measure similarity between time-series data, we implement the k-means clustering. Each cluster contained the time-series which follow common behaviour and clustered by states. Thus, all the time series in a given cluster we have similar time series from different states. So we have a common behaviour of time series for multiple states.

TABLE I. PARAMETER CHOICES

K-Means	Clusters: 4 Iterations: 10
DTW	Alignment: 5 LB_Keogh Reach: 5
ARIMA	Lag order: 5 Degree of differencing: 1

We now try to fit a regression line over all the time series' in a given cluster. This regression line can be used to predict a future time-series for the states that area associated with the cluster. For our case, we have used Autoregressive Integrated Moving Average (ARIMA). Given our input location, we retrieve the cluster to which the location belongs and use the regression that was fit for the cluster to predict the pollutant level.

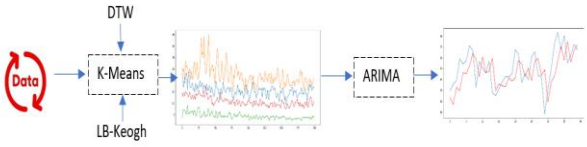

Figure 8: Implementation Flow

V. RESULTS

After successful execution of the project, the results indicated that k-means clustering combined with ARIMA can be useful in prediction of time-series data. K-means helped cluster together states with similar behaviour and then used ARIMA for regression fitting.

The k-means clustering algorithm clustered states based on similarity of time series of the states.

TABLE II. STATE-WISE CLUSTERING

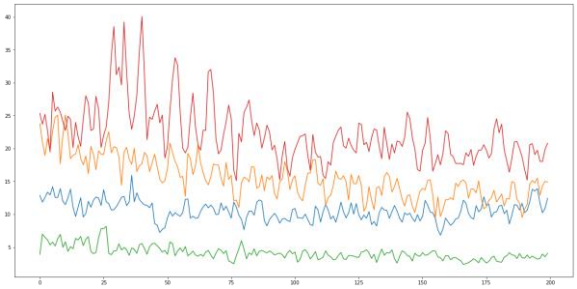
Figure 9: Cluster centroids of state-wise clustering

Cluster	States	RMSE
1	IN, LA, MS, TX, NC, CT, WI, GA, UT, NM	2.227
2	DC, IL, MI, NJ, NY, PA, VA, MA	2.349
3	NV, TN, SC, IA, ME, WY, ND, SD, OH, HI, RI	0.158
4	FL, KS, KY, NH, MD, AR, ID, OR, DE, AL, WA, MN	1.209

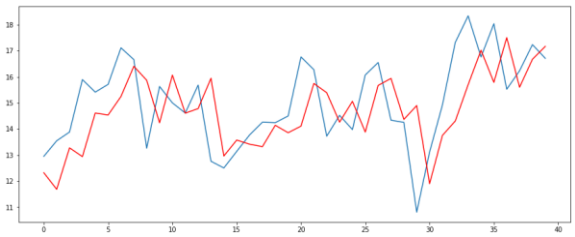

Figure 10: ARIMA for cluster 1

- **Intra-cluster distance:** 7615.39
- **Inter-cluster distance:** 834.99
- **Total Sum Square (Variance):** 8450.384
- **Total MSE:** 2.657

Additionally, we had performed the above k-means with DTW method on Arizona County 19 NO2 data. The time series were each a month's data for all the years between 2000-2016, i.e. 192-time series. K-means classified these time series, forming clusters of those months that showed the same level of NO2 pollution.

TABLE III. PARAMETER CHOICES

Cluster	States
1	July, August
2	January, December
3	February, March, October, November
4	April, May, June, September

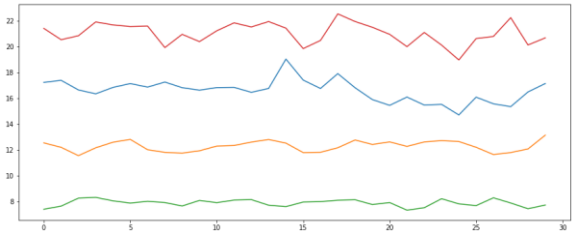

Figure 11: Centroids formed on the month-wise data.

VI. CONCLUSION AND FUTURE DIRECTIONS

For predicting air pollutant level, we considered multiple models. The most suitable method as per our evaluation is to cluster states with similar behaviour of pollutant levels. K-Means clustering followed by ARIMA can be used for time series prediction. We observe that pollutant levels follow seasonal behaviour. Using K-Means clustering, states that follow similar pollutant level behaviour were clustered together. For calculating distance for clustering, we conclude that Euclidean distance is not the correct approach. Dynamic Time Warping is one of the possible measures to calculate the alignment between two time-series'. Implementing DTW along with LB-Keogh (lower bound DTW) helps fasten the DTW for the given large dataset. Finally, ARIMA can be used to create time-series regression over the clusters for prediction. This provides with one time series regression line for each cluster. Thus, we have k clusters, each representing a group of similar behaviour patterns, with each cluster fitted to one regression line each.

The model is currently fit only for a single pollutant for convenient model training. It can, in the future, be extended to cover multiple pollutants whose data is available in our dataset. The time series data, with more computation power, can be carried out at hourly basis. This provides a more drilled/scrutinized analysis of the time-series data. With a proper understanding of the mathematics behind Support Vector Regression, we can implement it using the Support Vector Machine implementation done for this project.

VII. REFERENCES

[1] NATIONAL PATTERNS IN ENVIRONMENTAL INJUSTICE AND INEQUALITY: OUTDOOR NO2 AIR POLLUTION IN THE UNITED STATES. LARA P. CLARK, DYLAN B. MILLET, JULIAN D. MARSHALL

[2] JHUN I, COULL BA, SCHWARTZ J, HUBBELL B AND KOUTRAKIS P 2015 THE IMPACT OF WEATHER CHANGES ON AIR QUALITY AND HEALTH IN THE UNITED STATES IN 1994–2012 ENVIRON. RES. LETT. 10 084009

[3] NIENNATRAKULV, RANTANAMAHATAMA, ON CLUSTERING MULTIMEDIA TIME SERIES DATA USING K-MEANS AND DYNAMIC TIME WARPING. 2007 IEEE SEOUL.

[4] BENKABOU, SEIF-EDDINE ET AL. "UNSUPERVISED OUTLIER DETECTION FOR TIME SERIES BY ENTROPY AND DYNAMIC TIME WARPING." KNOWLEDGE AND INFORMATION SYSTEMS 54 (2017): 463-486.

VIII. IMPORTANT LINKS

Link to GitHub code:
https://github.com/daxamin/Air-Pollution-Prediction

IX. SELF – ASSESSMENT

Team Member: Aditya Upadhyay

Tasks I worked on –
Initial analysis of support vector regression and implementing it using SVM code, Data preprocessing tasks, Contributed to steps regarding Dynamic Time Warping

Code Contribution –
Contributed to the code for SVM and KNN
Number of lines of code: ~250-300

Contribution to poster -
Created the image depicting the workflow of the system. Also wrote all the conclusion part.

Contribution to report -
Maintained the implementation, parts of results and conclusion. Contributed the non-graph images for the report.

Meetings - The team had roughly 15 meetings and I missed one meeting due to other subject deadlines. Of the 4 times the team met the professor, I was availale for 3 times.

Contribution Rating - Equal

Team Member: Daxkumar Amin

Tasks I worked on - implementing K-means clustering on different time-series' state-wise & month-wise. Used DTW for the same and implemented ARIMA on cluster centroids

Code Contribution - Initially pre-processed the data using python scripts - fill NA & group by Date and County for trimming. Developed script in python for Dynamic Time Warping & its use in K-Means. Also implemented ARIMA on the cluster centroids. Evaluated the K-Means clustering by calculating intra-cluster & inter-cluster distance and total variance. Number of lines ~ 500.

Contribution to poster - Created table for K-Means clustering results that includes the information of 4 clusters formed i.e. the states that belong to each cluster & Total MSE of each cluster.

Contribution to report - Some portions of implementation. Generated the required graphs for clustering and ARIMA. Also included the clustering result table.

Meetings - Our group has met roughly 15 times throughout the semester. I have been present in all of the meetings. I have met Dr. Raju 4 times across the semester for the project.

Contribution Rating - Equal.

Team Member: Nilay Kapadia

Tasks I worked on - Analyzing different algorithms and comparing their results to choose an optimal one. Such as comparing LSTM/RNN, vanilla ARIMA and Support Vector Regression.

Contribution to code - Contributed for the development of analysis methods LSTM, ARIMA and SVR. Contributed approximately 350-400 lines of code.

Contribution to Poster - The entire formatting and printing of the poster as well a a few points in the Alternative Methods section.

Contribution to Report - The Related Work section, Introduction and abstract of the report. In this section I analyzed the different algorithms as mentioned in the Tasks I worked on and produced a superficial comparison among the different algorithms which were analyzed.

Meetings - Our group has met roughly 15 times throughout the semester. I have attended all meetings. I have met Dr. Raju 3 times.

Contribution Rating - I rate my contribution to the project **equal**.

Team Member: Priyance Mandlewala

Tasks I worked on - Understanding Lower Bound Keogh for expediting calculation of DTW distance. Also, I contributed in optimizing the K-means clustering on different time-series' state-wise & month-wise.

Contribution to code - contributed in plotting the graph using Plotly for initial data exploring. Secondly, I did pair programming in monthly and state wise clustering of time series and contributed in developing functions for DTW distance and Lower Bound Keogh.
Number of lines contributed - ~ 500 lines

Contribution to Poster - Contributed to explore initial data and forming graphs for initial phase on map of USA.

Contribution to Report - I worked on some part of implementation as I worked on the code for optimizing the code.

Meetings - Met roughly for around 15 times during the whole span of spring semester. I met Dr. Raju for roughly 2 times.

Contribution Rating - I rate my contribution to the project equal.

Team Member: Vaibhava Lakshmi

Tasks I worked on - Ideating, understanding methods to deal with our data for prediction, assisted in pre-processing, conducted literature review for DTW - K-means, generating time series, explored the possibility of implementing regression models, debugging and implementing the K-means and DTW method for month specific and state specific data.

Contribution to the code - contributed in data pre-processing for the state-wise and monthly clustering, in coding for the same with DTW and using the centroid results to better understand the data. For example: cleaning the clustering after K means as with the monthly clustering. ~350 lines.

Contribution to the Poster - Described the problem, technical section and additional methods.

Contribution to the Report - Described the methodology, assisted with the description of the abstract and introduction.
.
Meetings - As mentioned, group has met ~15 times and I have been a part of all the meetings. I have met Dr. Raju 3 times.

Contribution Ratings - My contribution to the project is equal.

Team Member: Vishrut Patel

Tasks I Worked On - Understanding SVR**,** finding alternatives to SVR, data preprocessing, analyzing data to find best combination of methods to predict the air pollution factors.

Contribution to Poster - References and formatting, printing the poster

Contribution to Code - Worked on development of LSTM, SVR and ARIMA analysis methods.
Number of lines of code: ~300-350

Contribution to Report - Assisted in methodology and implementation part.

Meetings - We met roughly 15 times over the period of spring semester of which I was present in 12 of them. I have met Dr Raju 3 times across the semester for project related discussion.

Contribution Rating - Equal