## Predicting Precipitation with Machine Learning

**Team:** Kaitlyn Lane, Jessica Ling, Patrick Ledoit, Conor Luppnow.

**Project Mentor TA:** Harshwardhan Yadav

#### 1) Abstract

For this project, we study the problem of predicting rainfall in India. India has been devastated by large amounts of precipitation during monsoon season which affects many aspects of the land, from farmers growing crops to devastating landslides. Current weather predictions will predict around 10 days in advance, but predicting the rainfall farther in advance, on the scale of months would be invaluable to farmers and disaster prevention. To contribute to this issue we first improve a well used dataset. The initial dataset only tracks precipitation rates in India and is used to predict rainfall for future years. We have improved the dataset to include other weather factors such as temperature, dew points, sea level pressure etc. Secondly, we have improved existing model predictions through various techniques including hyper-parameter tuning and feature selection. With both of these contributions we have been able to significantly improve the error rates of the models.

#### 2) Introduction

Our system takes as input a month and the name of any of 37 meteorological subdivisions in India, and outputs the predicted total rainfall for that month and that subdivision. The data we are training on is raw measurable weather data on a monthly basis labeled by the total rainfall for that month and year. Our data spans January 2010 to January 2020 and has monthly granularity.

We will compare our performance to currently available open-source and academic models predicting rainfall on low-dimensional data; that is, current models that predict future rainfall solely on past rainfall. To do this, we will adhere to the methods used in the papers and projects described in [section (3)](#_z9qq7uxbhthg) with minor modifications to suit higher data dimensionality. Then, we will compare the performance of our model to the performance of these published models. We will test on the same year that prior models tested on and directly compare the predicted results, and we can also compare the loss of our predictions.

##### Motivation

Rainfall for tropical and sub-tropical regions is a vitally important prediction for daily life, including flood risk assessment and agricultural impacts from increased rainfall or drought. While current weather models are able to capture short-term (days or weeks in advance) predictions with reasonable accuracy, it is also important to be able to predict important rainfall events further in advance. If we are able to predict monthly rainfall based on aggregated monthly data rather than granular daily data, it may indicate the possibility to have more sophisticated ML-based weather forecasts based with more robust time-scaling on predictions.

#### 3) Background

Listed below are two papers, and two github sources. Both git-hub sources used the same dataset. One only predicted for a single sub-division using various models [Sudharsan, 2021], whereas the other [Gaurav, 2018] only used a linear regression model to predict monthly rainfall data. With regards to the papers, both sources do not have data for the past ~10 years. One paper, [Mohammed, M. *et al,* 2020], used three models to predict monthly rainfall, and the in the other paper, [Singh, P. *et al,* 2013] they used a neural network to predict monthly rainfall.

Thus, we first aim to address the data issue and improve the datasets used by all works and secondly we will improve the models.

1. Mohammed, M., Kolapalli, R., Golla, N., & Maturi, S. S. (2020). Prediction of rainfall using machine learning techniques. International Journal of Scientific and Technology Research, 9(01), 3236-3240.
2. Singh, P., & Borah, B. (2013). Indian summer monsoon rainfall prediction using artificial neural network. *Stochastic environmental research and risk assessment*, *27*, 1585-1599.
3. Sudharsan, D., Indhu S, I., Kumar, K. S., Menon, M. (2021). Rainfall Pattern Prediction. Github repository, <https://github.com/DeepthiSudharsan/Rainfall-Pattern-Prediction-using-ML>
4. Gaurav, V. (2018). Rainfall Patterns Analysis of India. GitHub repository, <https://github.com/vgaurav3011/Rainfall-Prediction>

#### 4) Summary of Our Contributions

The datasets that were used in both open source projects and the papers only used historical rainfall data for their predictions. Our first contribution involves improving this data. To augment this data, we collected data from <https://weatherandclimate.com/>, which had monthly, and yearly breakdowns of weather data for 37 different meteorological regions within India across 10 years, 2010 - 2020. This data was cleaned for use in exploratory data analysis. This work is complete for this checkpoint.

For our second contribution, first we experimented with different models and then we tuned hyperparameters to improve the performance of the model.

#### 5) Detailed Description of Contributions

##### 5.1 Methods

For the methods for the first contribution, we wanted to increase the feature space of data. Where all the sources previously mentioned used rainfall measurements only, we gathered data from <https://weatherandclimate.com/>. This website contains monthly and yearly breakdown of weather data for different regions in India. The website is structured such that the uri '[/region/month-year](https://colab.research.google.com/drive/1DuTTb6zHpkHlUXn0hgI5vLRo9N8MTErK#)' displays a table containing the weather information for the given region month and year. Thus, we could webscrape this data. A description of all features we collected can be found in [Table 5.1A.](#_x4syjqrghg0z) Once all the data from the site was gathered we were able to clean up the data.

The data from the website contained some data in metric and imperial. In order to only use one unit, we kept metric units. Furthermore, the data that both open-source projects used contained a good amount of missing data. The data from the website did not have any missing data in the years that it covered (2010-2020). Once the data was in a good format we were able to look at correlations between features. The correlations that we see are expected. All temperature metrics and dew point metrics (which relates to temperature by definition) are highly correlated. Same with wind and precipitation metrics. Sea level pressure and precipitation are also highly correlated, which also makes sense as sea level pressure is related to monsoon and rainfall. Unsurprisingly, snowfall and temperature are inversely correlated, as low temperatures would correlate with snow fall.

Finally, for use in models, all categorical data such as region and month were converted to numerical data so the models would work.

For our second contribution we experimented with various models, and then tuned the hyperparameters of those models to see if we could squeeze out some accuracy compared to the default parameters. This was done using scikit-learn’s ‘GridSearchCV’. This search over parameters space was the most compute intensive part of our project. We picked enough different parameters that we could sample a range we were happy with, while not choosing too many as to have to wait hours for the search to complete. We used the following models: random forest regressor, support vector regression, elastic net, a multilayer perceptron regressor, and linear model. To see hyperparameters used see the code.

##### 5.2 Experiments and Results

Firstly, let us compare the two datasets that we have. The original dataset was 1 dimensional, only containing precipitation across multiple regions and time. However, quite a bit of the data was missing. As shown in [Table 5.2A](#_p52fq9ilrb07) over 60% of the time annual precipitation is missing. Precipitation for each month and season is missing roughly 10-30% of the time. Conversely, our dataset is complete. We do not have any null or missing values. However, the original dataset does contain data from 1901-2015 while our new dataset only contains data from 2010-2020. However, because of our higher dimensionality and more complete data, we believe that we have enough data to perform analysis and run various models. The last notable difference lies in the regions. The original dataset had 36 meteorological subdivisions while ours has 37. A number of the districts are the same, with a few differences such as West and East Madhya Pradesh being 2 separate subdivisions in original dataset vs only Madhya Pradesh being present in the new dataset.

As previously mentioned, we believe that our increased dimensionality will significantly improve the accuracy of the existing models. Features such as temperature, dew point, wind, sea level pressure will help make better predictions. This can be tested by comparing the same models on our cleaned dataset versus the dataset used in the opensource projects. We ran the same models with different datasets. When comparing against the project by [Gaurav, 2018], the results are summarized in [Table 5.2B](#_bei24fw6cshm). As both open source projects used the same dataset, we just showed results against one project.

From these results, it looks like using a dataset with a larger feature space helped a lot. The MAE for each model was significantly improved.

For our second contribution we tested various models and then hyperparameter tuned those models to see if any improvement could be made beyond having higher dimensional data. In order to test hyperparameter tuning we compared the default parameters against the tuned parameters. These results are shown in [Table 5.2C](#_8kki695wev4c). From these results it appears that using a dataset with a greater feature space allows for the models to significantly improve the prediction accuracy. For most of the models that were tuned, there was slight improvement to accuracy. The outlier being SVR which went from a MAE of 82 to 1.8.

#### 6) Compute/Other Resources Used

None of the models used were that intensive, so we were able to use google colab to run all of our code.

#### 7) Conclusions

With this project we have significantly improved the model for rainfall prediction specifically in India. Our most significant contribution is comes from improving the dataset. Our data is sourced from a relatively obscure website, that was difficult to find. The website has low rather low traffic, with an average of 38 thousand hits from December-May and with surprisingly only 40% of hits coming from India. For comparison the UPenn website (<https://www.upenn.edu/>) sees an average of 114.3 thousand hits and Walmart receives 84.4 million average hits. Thus, improving upon this dataset, and making it accessible to others, is a significant contribution. We were also very surprised by the performance of the linear model, it is one of the

Do concl for contrib 2.

##### For the Future:

During the initial phase of this project, we came across a Kaggle dataset maintained by Nidula Elgiryewithana,[Indian Weather Repository ( Daily Updating ) (kaggle.com)](https://www.kaggle.com/datasets/nelgiriyewithana/indian-weather-repository-daily-snapshot), which is storing the daily temperature across a few regions in India and was updating daily since August 2023. However, the updates stopped in early April 2024, meaning that the dataset has less than a years worth of data. A large public data collection like this that stores not only temperature, but other weather information as well, like humidity, precipitation, sea level pressure etc would be a great boon to predicting rainfall in India. Overall, the main pain point of this project is data collection so such a project, which would take many years, would be invaluable in the future. Then the data could be aggregated, transformed, and used to predict rainfall further in advance.

##### Ethical Considerations, and Broader Social and Environmental Impact:

India and other areas are at high risk of floods during different seasons, additionally, rainfall is key to agriculture. Knowing periods of drought or high rainfall is very important for farmers to determine the right crops to grow, the best time to harvest, and precautions to undertake on the field. Thus, predicting rainfall can help communities predict and prepare for floods and plan their agriculture ventures. This project will help farmers and other interested parties predict rainfall in advance. An even more complete dataset, as described in the above section would be even better and likely have a larger social impact.

#### (Exempted from page limit) Other Prior Work / References (apart from Sec 3) that are cited in the text:

1. Mohammed, M., Kolapalli, R., Golla, N., & Maturi, S. S. (2020). Prediction of rainfall using machine learning techniques. International Journal of Scientific and Technology Research, 9(01), 3236-3240.
2. Singh, P., & Borah, B. (2013). Indian summer monsoon rainfall prediction using artificial neural network. *Stochastic environmental research and risk assessment*, *27*, 1585-1599.
3. Sudharsan, D., Indhu S, I., Kumar, K. S., Menon, M. (2021). Rainfall Pattern Prediction. Github repository, <https://github.com/DeepthiSudharsan/Rainfall-Pattern-Prediction-using-ML>
4. Gaurav, V. (2018). Rainfall Patterns Analysis of India. GitHub repository, <https://github.com/vgaurav3011/Rainfall-Prediction>

<https://www.kaggle.com/datasets/rajanand/rainfall-in-india>

**Broader Dissemination Information:**

Your report title and the list of team members will be published on the class website. Would you also like your pdf report to be published?

YES

If your answer to the above question is yes, are there any other links to github / youtube / blog post / project website that you would like to publish alongside the report? If so, list them here.

* <description in a few words>: <URL>
* <description in a few words>: <URL>

...

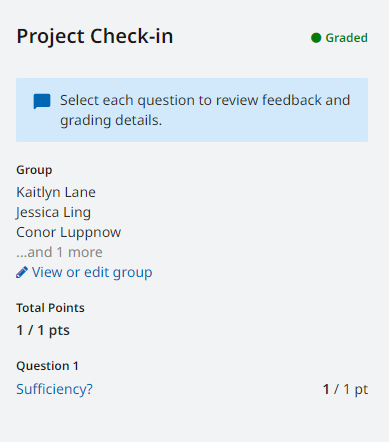
##### 

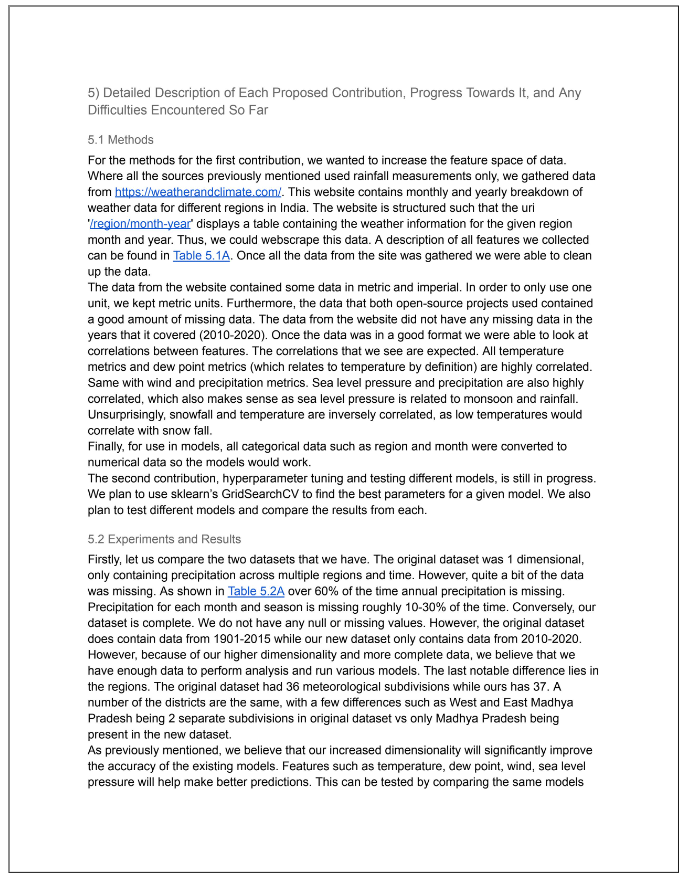
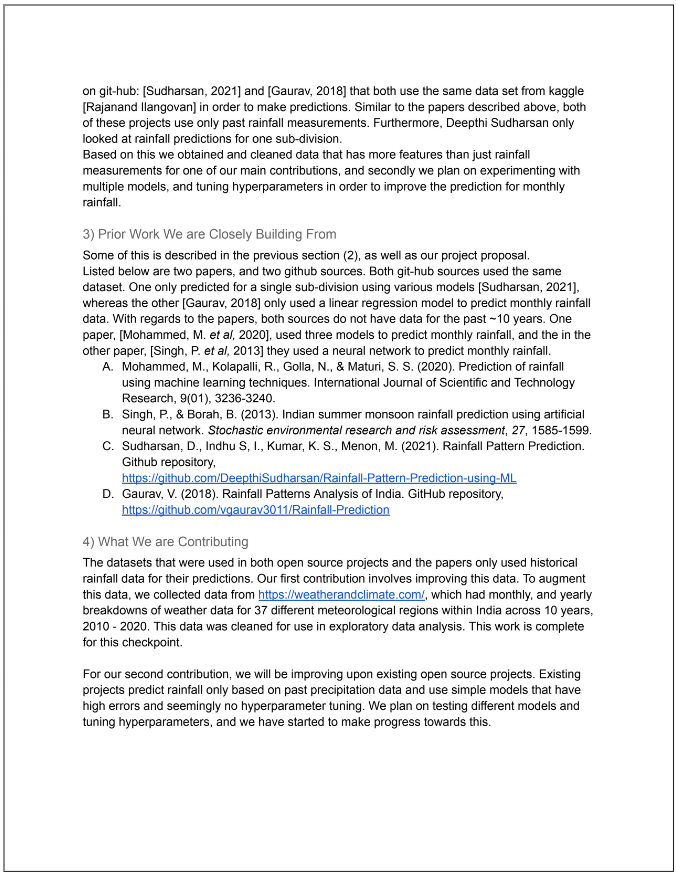
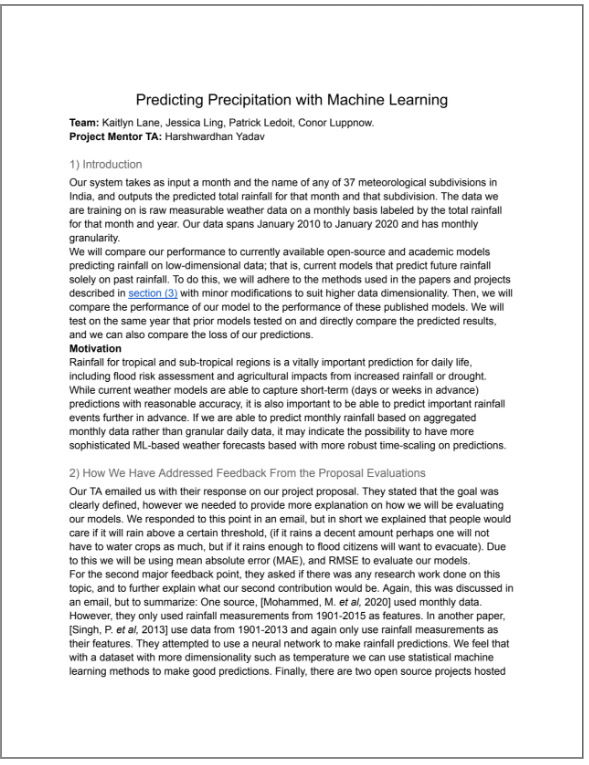
#### (Exempted from page limit) **Work Report: This may look like your GANTT chart from the midway report, with more completed steps now. Okay to modify.** (Mark completed steps in green, as shown here. For convenience, you may split into two charts, one till Nov 8, and another for after Nov 8, placed one below the other.)

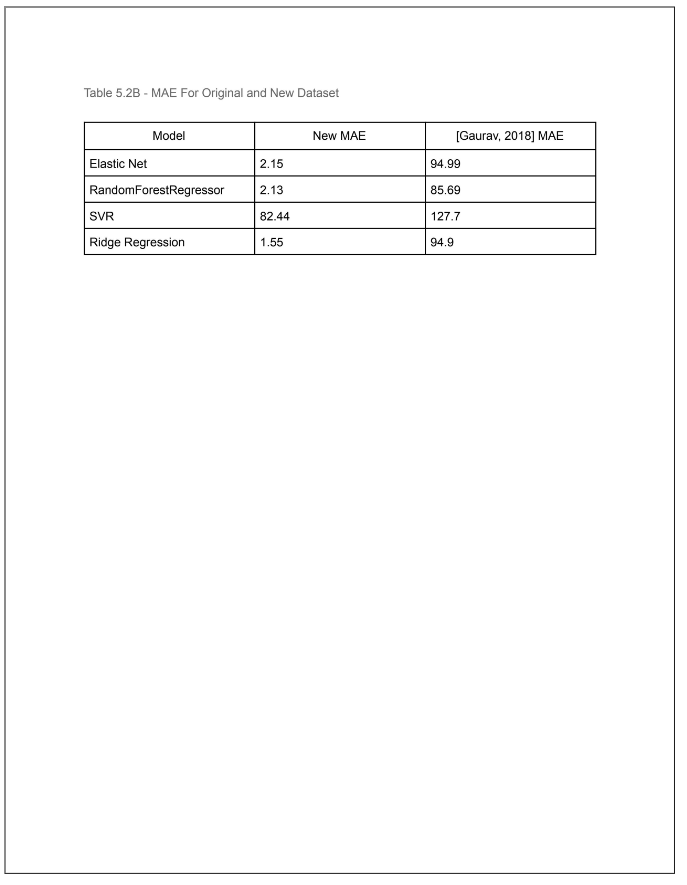
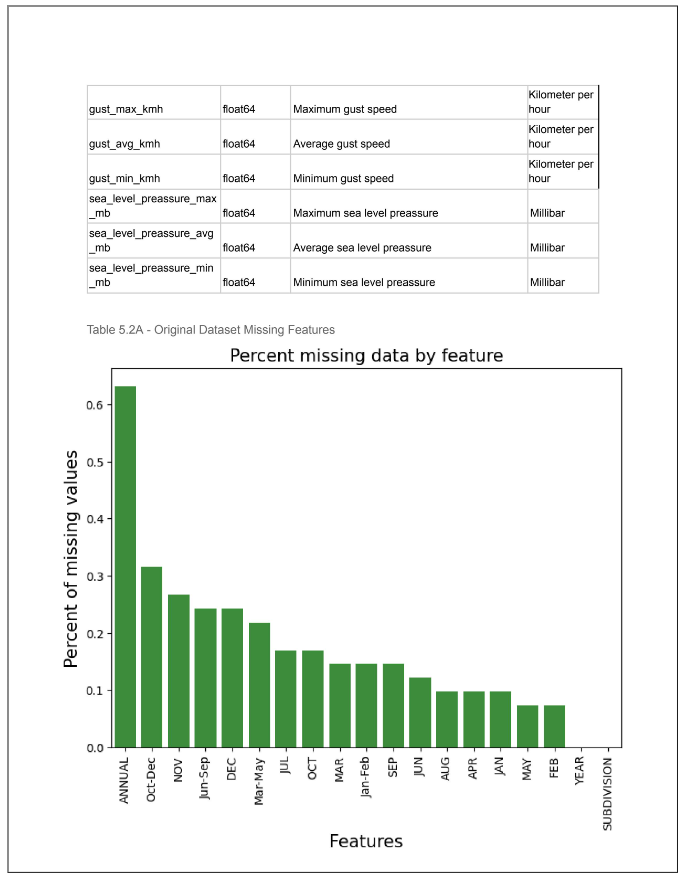
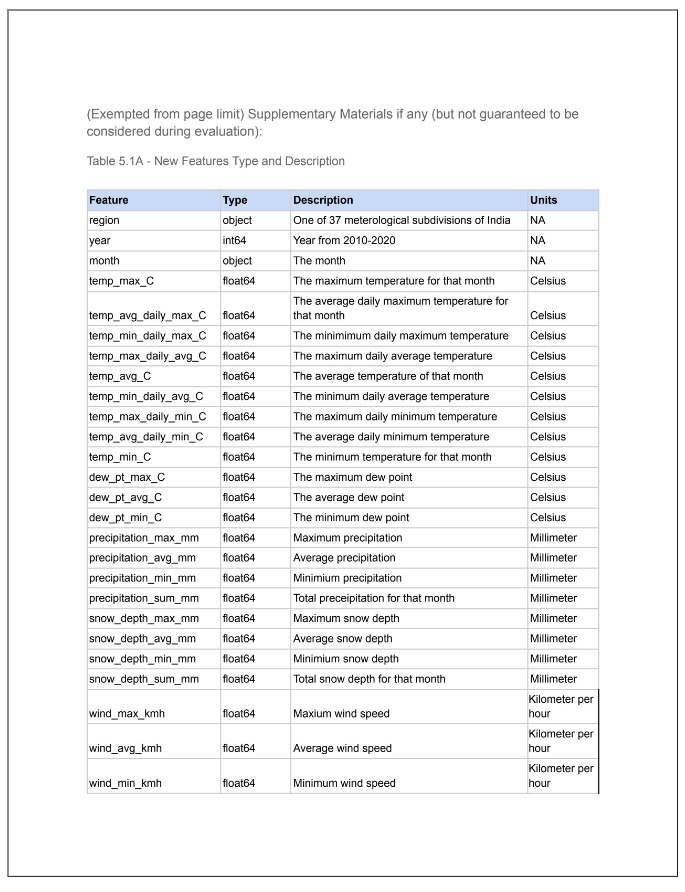
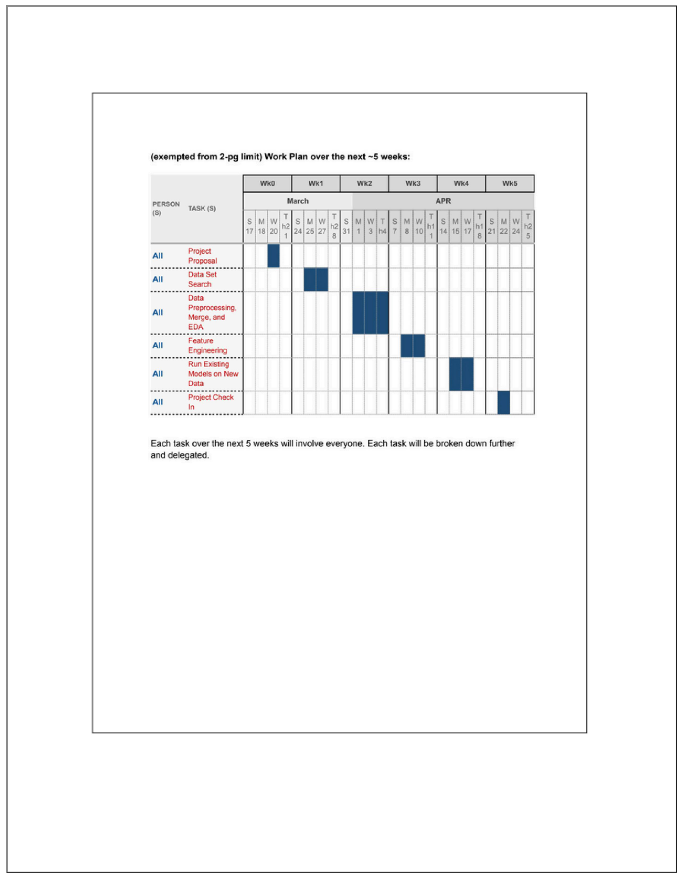
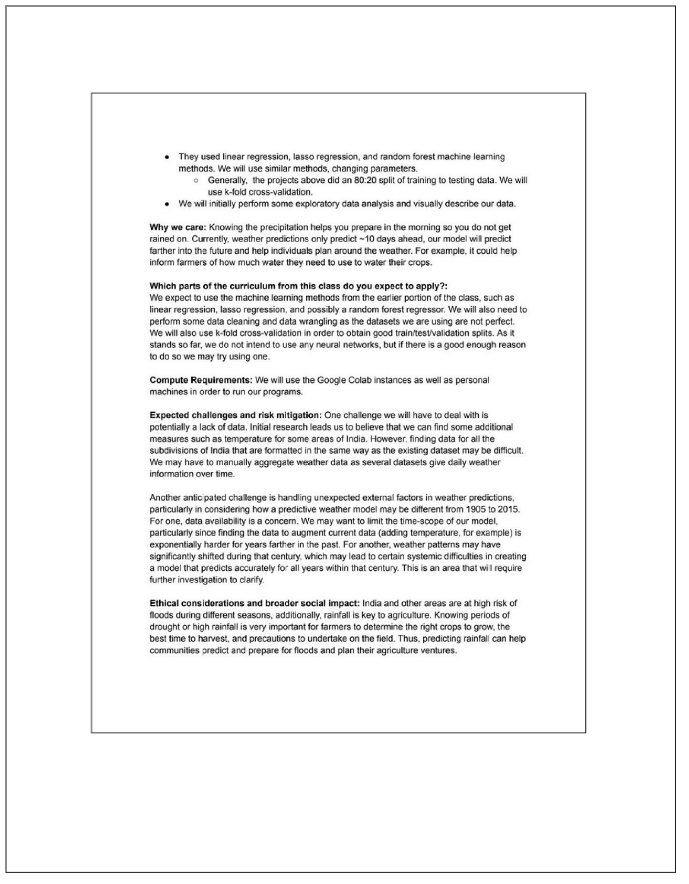
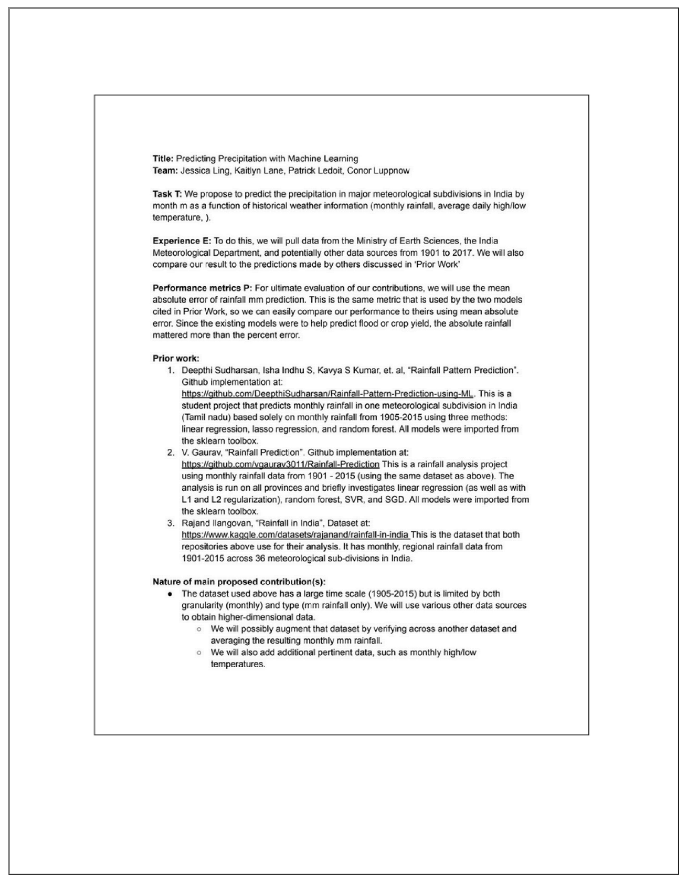
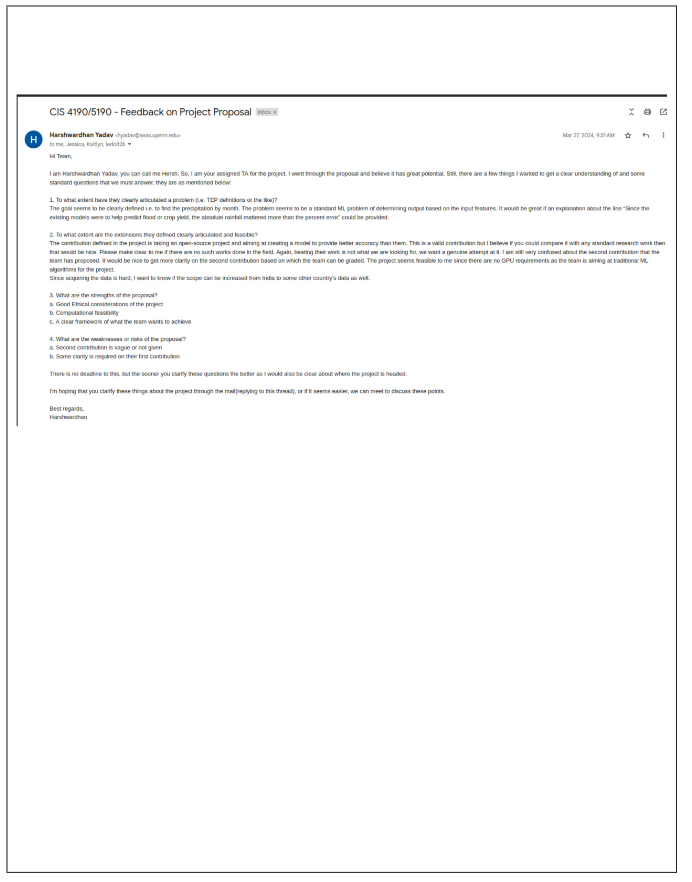
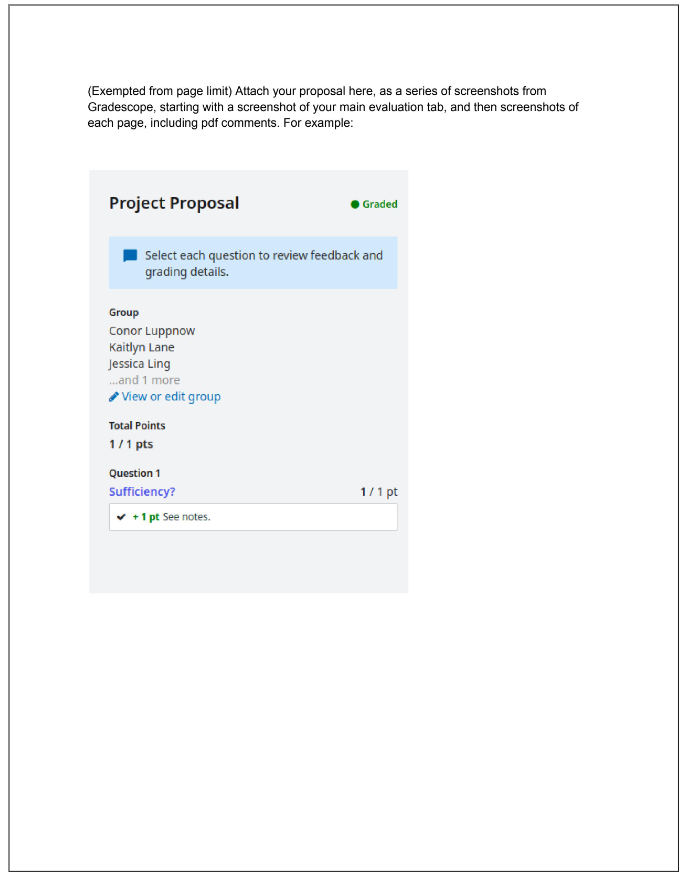
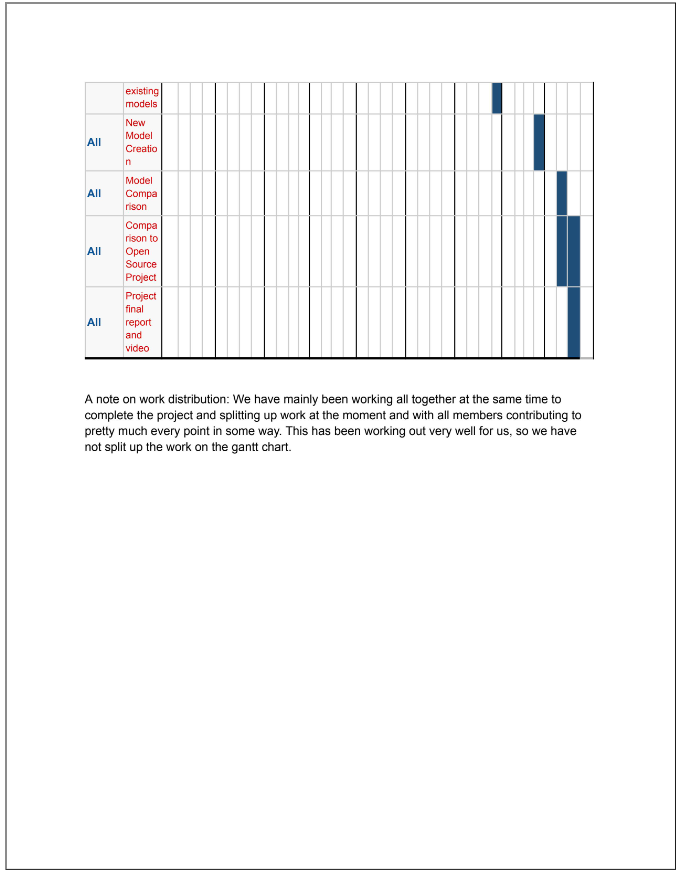
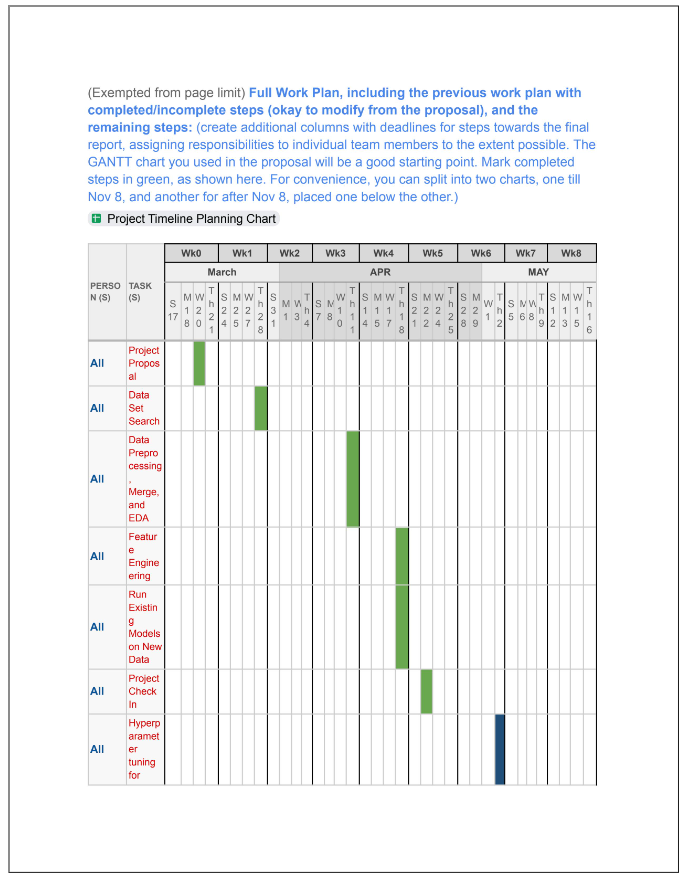
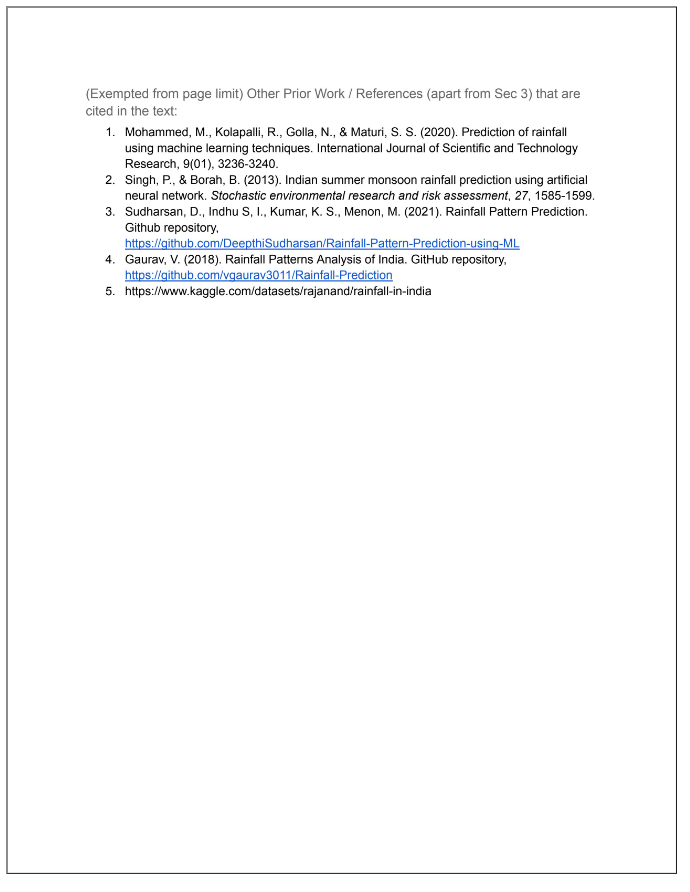
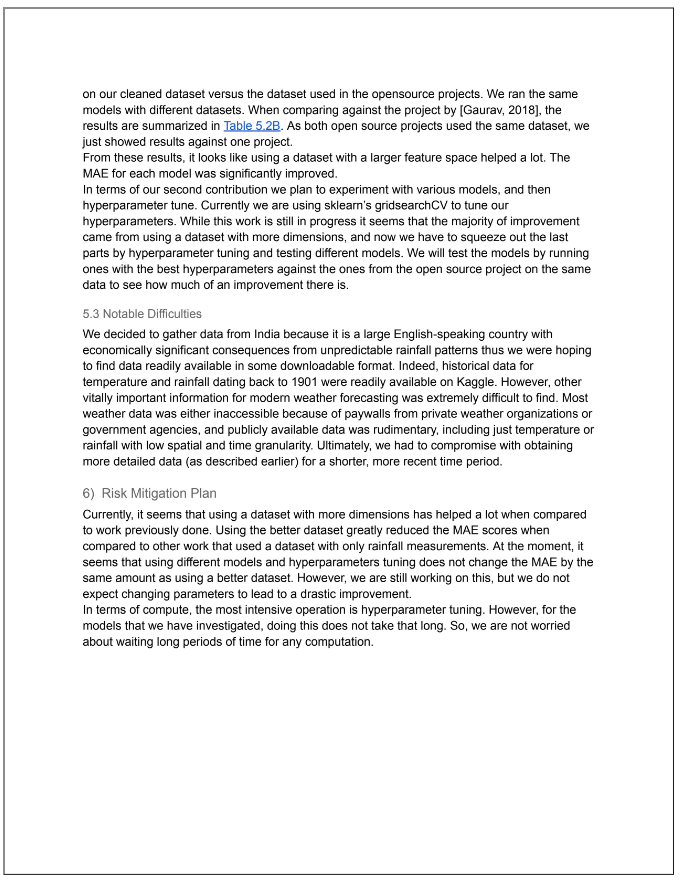
| **PERSON (S)** | **TASK (S)** | **Wk5** | | | | **Wk6** | | | | **Wk7** | | | | **Wk8** | | | | **Wk9** | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **OCT** | | | | | | | | | | | | | | | | | **NOV** | | |
| S3 | M4 | W6 | Th7 | S10 | M11 | W13 | Th14 | S17 | M18 | W20 | Th21 | S24 | M25 | W27 | Th28 | S31 | M1 | W3 | Th4 |
| **Name 1** | Task 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Name 2, Name 1** | Task 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Name 3** | Task 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Name 3, Name 2** | Task 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Name 2, Name 3** | Task 5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **...** | Task 6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **...** | Task 7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **...** | Task 8 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **...** | Task 9 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **...** | Task 10 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

#### (Exempted from page limit) Attach your midway report here, as a series of screenshots from Gradescope, starting with a screenshot of your main evaluation tab, and then screenshots of each page, including pdf comments. This is similar to how you were required to attach screenshots of the proposal in your midway report.

#### 





  
  
(Exempted from page limit) Supplementary Materials if any (but not guaranteed to be considered during evaluation):

#### (Exempted from page limit) Supplementary Materials if any (but not guaranteed to be considered during evaluation):

##### Table 5.1A - New Features Type and Description

| **Feature** | **Type** | **Description** | **Units** |
| --- | --- | --- | --- |
| region | object | One of 37 meterological subdivisions of India | NA |
| year | int64 | Year from 2010-2020 | NA |
| month | object | The month | NA |
| temp\_max\_C | float64 | The maximum temperature for that month | Celsius |
| temp\_avg\_daily\_max\_C | float64 | The average daily maximum temperature for that month | Celsius |
| temp\_min\_daily\_max\_C | float64 | The minimimum daily maximum temperature | Celsius |
| temp\_max\_daily\_avg\_C | float64 | The maximum daily average temperature | Celsius |
| temp\_avg\_C | float64 | The average temperature of that month | Celsius |
| temp\_min\_daily\_avg\_C | float64 | The minimum daily average temperature | Celsius |
| temp\_max\_daily\_min\_C | float64 | The maximum daily minimum temperature | Celsius |
| temp\_avg\_daily\_min\_C | float64 | The average daily minimum temperature | Celsius |
| temp\_min\_C | float64 | The minimum temperature for that month | Celsius |
| dew\_pt\_max\_C | float64 | The maximum dew point | Celsius |
| dew\_pt\_avg\_C | float64 | The average dew point | Celsius |
| dew\_pt\_min\_C | float64 | The minimum dew point | Celsius |
| precipitation\_max\_mm | float64 | Maximum precipitation | Millimeter |
| precipitation\_avg\_mm | float64 | Average precipitation | Millimeter |
| precipitation\_min\_mm | float64 | Minimium precipitation | Millimeter |
| precipitation\_sum\_mm | float64 | Total preceipitation for that month | Millimeter |
| snow\_depth\_max\_mm | float64 | Maximum snow depth | Millimeter |
| snow\_depth\_avg\_mm | float64 | Average snow depth | Millimeter |
| snow\_depth\_min\_mm | float64 | Minimium snow depth | Millimeter |
| snow\_depth\_sum\_mm | float64 | Total snow depth for that month | Millimeter |
| wind\_max\_kmh | float64 | Maxium wind speed | Kilometer per hour |
| wind\_avg\_kmh | float64 | Average wind speed | Kilometer per hour |
| wind\_min\_kmh | float64 | Minimum wind speed | Kilometer per hour |
| gust\_max\_kmh | float64 | Maximum gust speed | Kilometer per hour |
| gust\_avg\_kmh | float64 | Average gust speed | Kilometer per hour |
| gust\_min\_kmh | float64 | Minimum gust speed | Kilometer per hour |
| sea\_level\_preassure\_max\_mb | float64 | Maximum sea level preassure | Millibar |
| sea\_level\_preassure\_avg\_mb | float64 | Average sea level preassure | Millibar |
| sea\_level\_preassure\_min\_mb | float64 | Minimum sea level preassure | Millibar |

##### Table 5.2A - Original Dataset Missing Features

##### Table 5.2B - MAE For Original and New Dataset

| Model | New MAE (mm) | [Gaurav, 2018] MAE (mm) |
| --- | --- | --- |
| RandomForestRegressor | 2.252 | 85.69 |
| SVR | 81.96 | 127.70 |
| Elastic Net | 3.205 | 94.99 |
| Ridge Regression | 1.55 | 94.90 |

##### Table 5.2C - MAE for default and tuned parameters

| Model | MAE (default parameters) [mm] | MAE (tuned parameters) [mm] |
| --- | --- | --- |
| Random Forest Regressor | 2.252 | 2.091 |
| SVR | 81.96 | 1.778 |
| Elastic Net | 3.205 | 1.539 |
| Ridge Regression | 1.55 | 1.50 |
| MLP | 4.322 | 3.211 |
| Linear Regression | 1.595 | 1.468 |

##### 

##### Table 5.2D - Combined Table

| Model | MAE (Original Dataset) [mm] | MAE (new dataset) [mm] | MAE (tuned params) [mm] |
| --- | --- | --- | --- |
| Random Forest Regressor | 85.69 | 2.252 | 2.091 |
| SVR | 127.70 | 81.96 | 1.778 |
| Elastic Net | 94.99 | 3.205 | 1.539 |
| Ridge Regression | 94.90 | 1.55 | 1.50 |
| MLP (New) | 87.19 | 4.322 | 3.211 |
| Linear Regression (New) | 93.64 | 1.595 | 1.468 |