## Predicting Precipitation with Machine Learning

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

**Project Mentor TA:** Harshwardhan Yadav

#### 1) 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)](#_4qdql9m95ush) 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.

#### 2) How We Have Addressed Feedback From the Proposal Evaluations

Our TA emailed us with their response on our project proposal. They stated that the goal was clearly defined, however we needed to provide more explanation on how we will be evaluating our models. We responded to this point in an email, but in short we explained that people would care if it will rain above a certain threshold, (if it rains a decent amount perhaps one will not have to water crops as much, but if it rains enough to flood citizens will want to evacuate). Due to this we will be using mean absolute error (MAE), and RMSE to evaluate our models.

For the second major feedback point, they asked if there was any research work done on this topic, and to further explain what our second contribution would be. Again, this was discussed in an email, but to summarize: One source, [Mohammed, M. *et al,* 2020] used monthly data. However, they only used rainfall measurements from 1901-2015 as features. In another paper, [Singh, P. *et al,* 2013] use data from 1901-2013 and again only use rainfall measurements as their features. They attempted to use a neural network to make rainfall predictions. We feel that with a dataset with more dimensionality such as temperature we can use statistical machine learning methods to make good predictions. Finally, there are two open source projects hosted on git-hub: [Sudharsan, 2021] and [Gaurav, 2018] that both use the same data set from kaggle [Rajanand Ilangovan] in order to make predictions. Similar to the papers described above, both of these projects use only past rainfall measurements. Furthermore, Deepthi Sudharsan only looked at rainfall predictions for one sub-division.

Based on this we obtained and cleaned data that has more features than just rainfall measurements for one of our main contributions, and secondly we plan on experimenting with multiple models, and tuning hyperparameters in order to improve the prediction for monthly rainfall.

#### 3) Prior Work We are Closely Building From

Some of this is described in the previous section (2), as well as our project proposal.

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.

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) What We are Contributing

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, we will be improving upon existing open source projects. Existing projects predict rainfall only based on past precipitation data and use simple models that have high errors and seemingly no hyperparameter tuning. We plan on testing different models and tuning hyperparameters, and we have started to make progress towards this.

#### 5) Detailed Description of Each Proposed Contribution, Progress Towards It, and Any Difficulties Encountered So Far

##### 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.

The second contribution, hyperparameter tuning and testing different models, is still in progress. We plan to use sklearn’s GridSearchCV to find the best parameters for a given model. We also plan to test different models and compare the results from each.

##### 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.

In terms of our second contribution we plan to experiment with various models, and then hyperparameter tune. Currently we are using sklearn’s gridsearchCV to tune our hyperparameters. While this work is still in progress it seems that the majority of improvement came from using a dataset with more dimensions, and now we have to squeeze out the last parts by hyperparameter tuning and testing different models. We will test the models by running ones with the best hyperparameters against the ones from the open source project on the same data to see how much of an improvement there is.

##### 5.3 Notable Difficulties

We decided to gather data from India because it is a large English-speaking country with economically significant consequences from unpredictable rainfall patterns thus we were hoping to find data readily available in some downloadable format. Indeed, historical data for temperature and rainfall dating back to 1901 were readily available on Kaggle. However, other vitally important information for modern weather forecasting was extremely difficult to find. Most weather data was either inaccessible because of paywalls from private weather organizations or government agencies, and publicly available data was rudimentary, including just temperature or rainfall with low spatial and time granularity. Ultimately, we had to compromise with obtaining more detailed data (as described earlier) for a shorter, more recent time period.

#### 6) Risk Mitigation Plan

Currently, it seems that using a dataset with more dimensions has helped a lot when compared to work previously done. Using the better dataset greatly reduced the MAE scores when compared to other work that used a dataset with only rainfall measurements. At the moment, it seems that using different models and hyperparameters tuning does not change the MAE by the same amount as using a better dataset. However, we are still working on this, but we do not expect changing parameters to lead to a drastic improvement.

In terms of compute, the most intensive operation is hyperparameter tuning. However, for the models that we have investigated, doing this does not take that long. So, we are not worried about waiting long periods of time for any computation.

#### 

#### (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>
5. https://www.kaggle.com/datasets/rajanand/rainfall-in-india

##### 

#### (Exempted from page limit) **Full Work Plan, including the previous work plan with completed/incomplete steps (okay to modify from the proposal), and the remaining steps:** (create additional columns with deadlines for steps towards the final report, assigning responsibilities to individual team members to the extent possible. The GANTT chart you used in the proposal will be a good starting point. Mark completed steps in green, as shown here. For convenience, you can split into two charts, one till Nov 8, and another for after Nov 8, placed one below the other.)

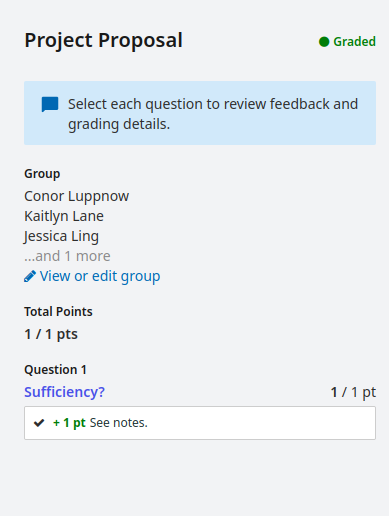
[Project Timeline Planning Chart](https://docs.google.com/spreadsheets/d/10n5HWlQUaEOv36W8EvNIGTw6CIFya-VA1PCiH_7-YJA/edit#gid=522870941)

| **PERSON (S)** | **TASK (S)** | **Wk0** | | | | **Wk1** | | | | **Wk2** | | | | **Wk3** | | | | **Wk4** | | | | **Wk5** | | | | **Wk6** | | | | **Wk7** | | | | **Wk8** | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **March** | | | | | | | | | **APR** | | | | | | | | | | | | | | | | | **MAY** | | | | | | | | | |
| S17 | M18 | W20 | Th21 | S24 | M25 | W27 | Th28 | S31 | M1 | W3 | Th4 | S7 | M8 | W10 | Th11 | S14 | M15 | W17 | Th18 | S21 | M22 | W24 | Th25 | S28 | M29 | W1 | Th2 | S5 | M6 | W8 | Th9 | S12 | M13 | W15 | Th16 |
| **All** | Project Proposal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **All** | Data Set Search |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **All** | Data Preprocessing, Merge, and EDA |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **All** | Feature Engineering |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **All** | Run Existing Models on New Data |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **All** | Project Check In |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **All** | Hyperparameter tuning for existing models |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **All** | New Model Creation |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **All** | Model Comparison |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **All** | Comparison to Open Source Project |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **All** | Project final report and video |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

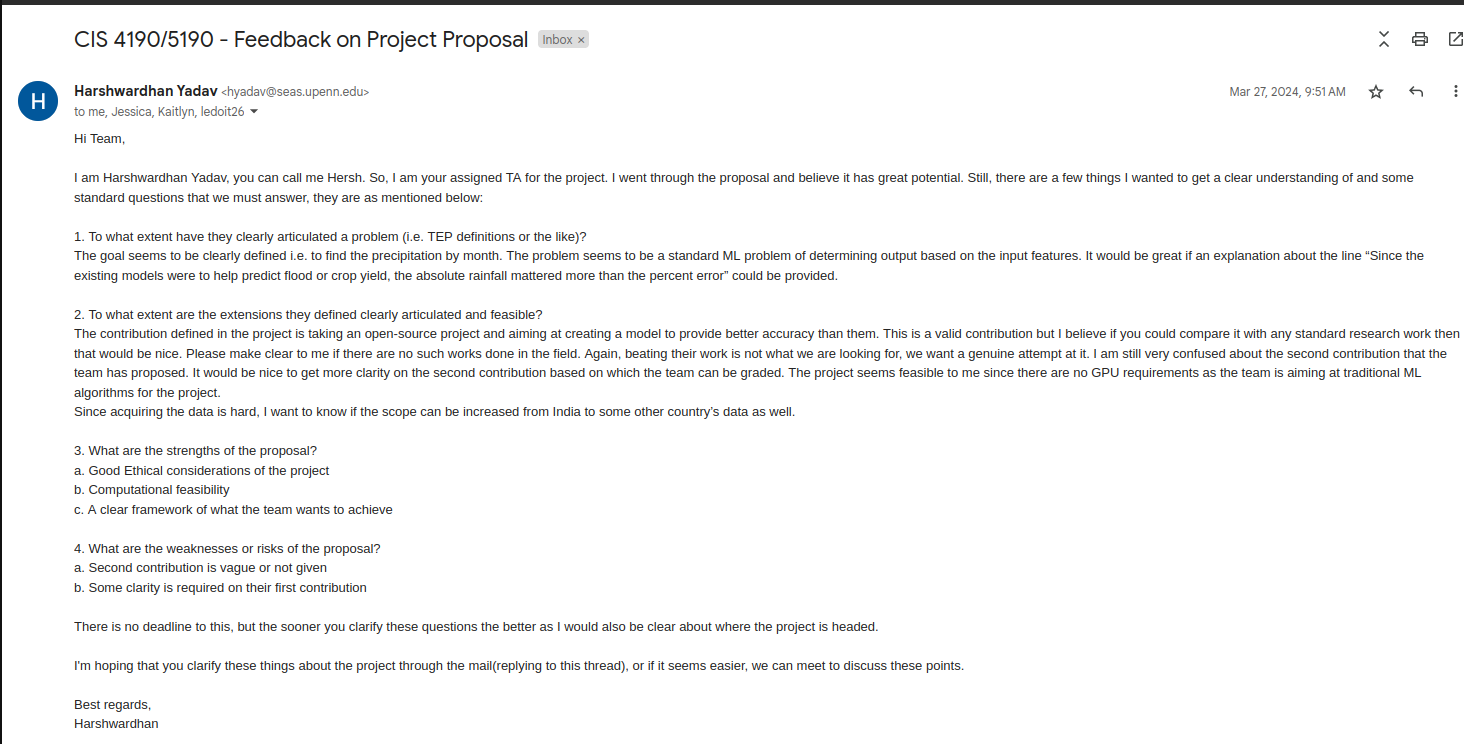
A note on work distribution: We have mainly been working all together at the same time to complete the project and splitting up work at the moment and with all members contributing to pretty much every point in some way. This has been working out very well for us, so we have not split up the work on the gantt chart.

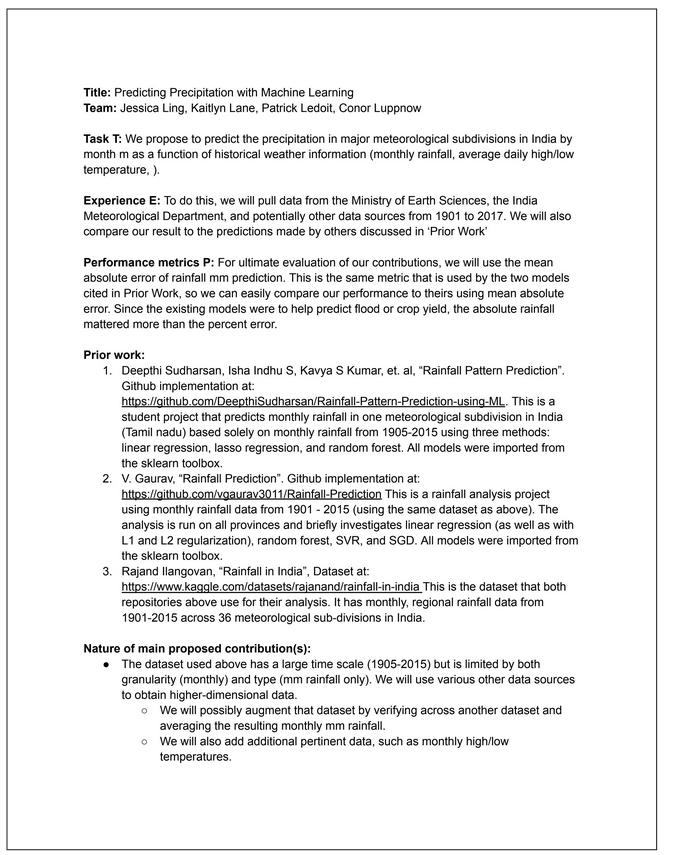
(Exempted from page limit) Attach your proposal here, as a series of screenshots from Gradescope, starting with a screenshot of your main evaluation tab, and then screenshots of each page, including pdf comments. For example:

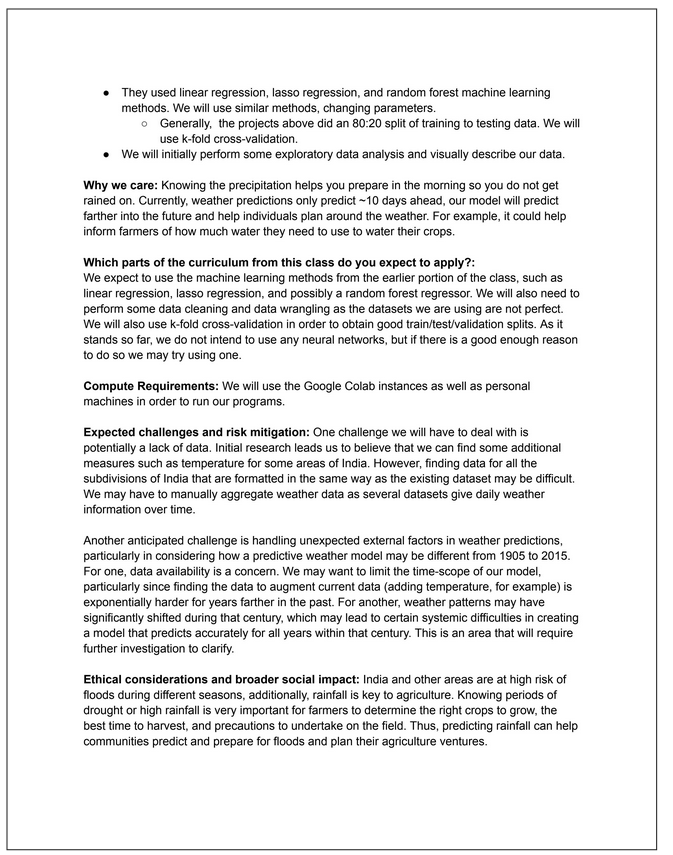
#### 

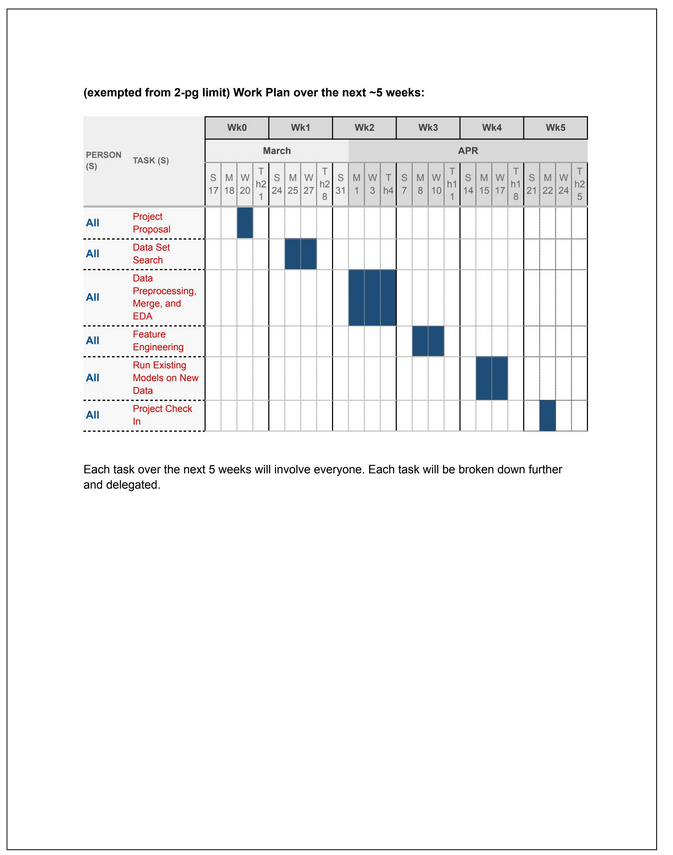


#### 









#### (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 | [Gaurav, 2018] MAE |
| --- | --- | --- |
| Elastic Net | 2.15 | 94.99 |
| RandomForestRegressor | 2.13 | 85.69 |
| SVR | 82.44 | 127.7 |
| Ridge Regression | 1.55 | 94.9 |

##### 