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) 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.

A. 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.
B. Singh, P., & Borah, B. (2013). Indian summer monsoon rainfall prediction using artificial neural network. *Stochastic environmental research and risk assessment*, *27*, 1585-1599.
C. 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
D. 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' 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. 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 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. 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.)

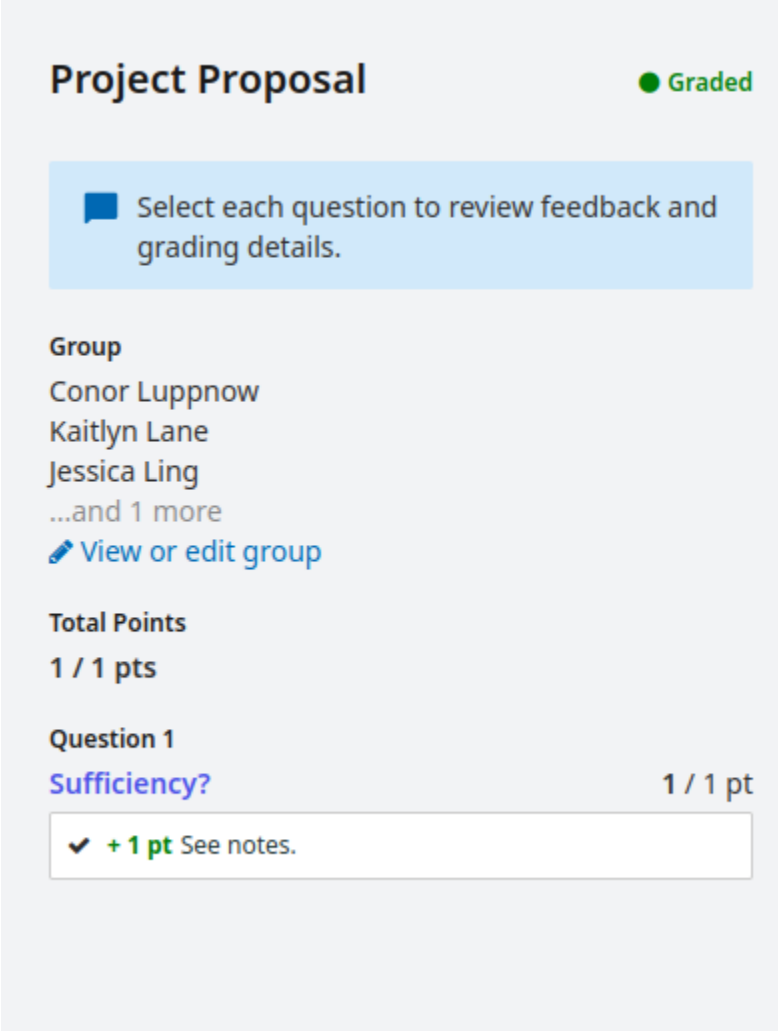
Project Timeline Planning Chart

PERSON (S)	TASK (S)	Wk0				Wk1				Wk2				Wk3				Wk4				Wk5				Wk6		Wk7				Wk8			
		March												APR														MAY								
		S 17	M 18	W 20	Th 21	S 24	M 25	W 27	Th 28	S 31	M 1	W 3	Th 4	S 7	M 8	W 10	Th 11	S 14	M 15	W 17	Th 18	S 21	M 22	W 24	Th 25	S 28	M 29	W 1	Th 2	S 5	M 6	W 8	Th 9	S 12	M 13	W 15	Th 16
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:



CIS 4190/5190 - Feedback on Project Proposal

Inbox ×

Harshwardhan Yadav <hyadav@seas.upenn.edu> Mar 27, 2024, 9:51 AM

to me, Jessica, Kaitlyn, ledoit26

Hi Team,

I am Harshwardhan Yadav, you can call me Hersh. So, I am your assigned TA for the project. I went through the proposal and believe it has great potential. Still, there are a few things I wanted to get a clear understanding of and some standard questions that we must answer, they are as mentioned below:

1. To what extent have they clearly articulated a problem (i.e. TEP definitions or the like)?
The goal seems to be clearly defined i.e. to find the precipitation by month. The problem seems to be a standard ML problem of determining output based on the input features. It would be great if an explanation about the line “Since the existing models were to help predict flood or crop yield, the absolute rainfall mattered more than the percent error” could be provided.

2. To what extent are the extensions they defined clearly articulated and feasible?
The contribution defined in the project is taking an open-source project and aiming at creating a model to provide better accuracy than them. This is a valid contribution but I believe if you could compare it with any standard research work then that would be nice. Please make clear to me if there are no such works done in the field. Again, beating their work is not what we are looking for, we want a genuine attempt at it. I am still very confused about the second contribution that the team has proposed. It would be nice to get more clarity on the second contribution based on which the team can be graded. The project seems feasible to me since there are no GPU requirements as the team is aiming at traditional ML algorithms for the project.
Since acquiring the data is hard, I want to know if the scope can be increased from India to some other country’s data as well.

3. What are the strengths of the proposal?
a. Good Ethical considerations of the project
b. Computational feasibility
c. A clear framework of what the team wants to achieve

4. What are the weaknesses or risks of the proposal?
a. Second contribution is vague or not given
b. Some clarity is required on their first contribution

There is no deadline to this, but the sooner you clarify these questions the better as I would also be clear about where the project is headed.

I'm hoping that you clarify these things about the project through the mail(replying to this thread), or if it seems easier, we can meet to discuss these points.

Best regards,
Harshwardhan

Title: Predicting Precipitation with Machine Learning
Team: Jessica Ling, Kaitlyn Lane, Patrick Ledoit, Conor Luppnow

Task T: We propose to predict the precipitation in major meteorological subdivisions in India by month m as a function of historical weather information (monthly rainfall, average daily high/low temperature,).

Experience E: To do this, we will pull data from the Ministry of Earth Sciences, the India Meteorological Department, and potentially other data sources from 1901 to 2017. We will also compare our result to the predictions made by others discussed in 'Prior Work'

Performance metrics P: For ultimate evaluation of our contributions, we will use the mean absolute error of rainfall mm prediction. This is the same metric that is used by the two models cited in Prior Work, so we can easily compare our performance to theirs using mean absolute error. Since the existing models were to help predict flood or crop yield, the absolute rainfall mattered more than the percent error.

Prior work:

1. Deepthi Sudharsan, Isha Indhu S, Kavya S Kumar, et. al, "Rainfall Pattern Prediction". Github implementation at: https://github.com/DeepthiSudharsan/Rainfall-Pattern-Prediction-using-ML. This is a student project that predicts monthly rainfall in one meteorological subdivision in India (Tamil nadu) based solely on monthly rainfall from 1905-2015 using three methods: linear regression, lasso regression, and random forest. All models were imported from the sklearn toolbox.
2. V. Gaurav, "Rainfall Prediction". Github implementation at: https://github.com/vgaurav3011/Rainfall-Prediction This is a rainfall analysis project using monthly rainfall data from 1901 - 2015 (using the same dataset as above). The analysis is run on all provinces and briefly investigates linear regression (as well as with L1 and L2 regularization), random forest, SVR, and SGD. All models were imported from the sklearn toolbox.
3. Rajand Ilangovan, "Rainfall in India", Dataset at: https://www.kaggle.com/datasets/rajanand/rainfall-in-india This is the dataset that both repositories above use for their analysis. It has monthly, regional rainfall data from 1901-2015 across 36 meteorological sub-divisions in India.

Nature of main proposed contribution(s):

- The dataset used above has a large time scale (1905-2015) but is limited by both granularity (monthly) and type (mm rainfall only). We will use various other data sources to obtain higher-dimensional data.
    - We will possibly augment that dataset by verifying across another dataset and averaging the resulting monthly mm rainfall.
    - We will also add additional pertinent data, such as monthly high/low temperatures.

- They used linear regression, lasso regression, and random forest machine learning methods. We will use similar methods, changing parameters.
    - Generally, the projects above did an 80:20 split of training to testing data. We will use k-fold cross-validation.
- We will initially perform some exploratory data analysis and visually describe our data.

Why we care: Knowing the precipitation helps you prepare in the morning so you do not get rained on. Currently, weather predictions only predict ~10 days ahead, our model will predict farther into the future and help individuals plan around the weather. For example, it could help inform farmers of how much water they need to use to water their crops.

Which parts of the curriculum from this class do you expect to apply?:
We expect to use the machine learning methods from the earlier portion of the class, such as linear regression, lasso regression, and possibly a random forest regressor. We will also need to perform some data cleaning and data wrangling as the datasets we are using are not perfect. We will also use k-fold cross-validation in order to obtain good train/test/validation splits. As it stands so far, we do not intend to use any neural networks, but if there is a good enough reason to do so we may try using one.

Compute Requirements: We will use the Google Colab instances as well as personal machines in order to run our programs.

Expected challenges and risk mitigation: One challenge we will have to deal with is potentially a lack of data. Initial research leads us to believe that we can find some additional measures such as temperature for some areas of India. However, finding data for all the subdivisions of India that are formatted in the same way as the existing dataset may be difficult. We may have to manually aggregate weather data as several datasets give daily weather information over time.

Another anticipated challenge is handling unexpected external factors in weather predictions, particularly in considering how a predictive weather model may be different from 1905 to 2015. For one, data availability is a concern. We may want to limit the time-scope of our model, particularly since finding the data to augment current data (adding temperature, for example) is exponentially harder for years farther in the past. For another, weather patterns may have significantly shifted during that century, which may lead to certain systemic difficulties in creating a model that predicts accurately for all years within that century. This is an area that will require further investigation to clarify.

Ethical considerations and broader social 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.

(exempted from 2-pg limit) Work Plan over the next ~5 weeks:

PERSON (S)	TASK (S)	Wk0				Wk1				Wk2				Wk3				Wk4				Wk5			
		March												APR											
		S 17	M 18	W 20	Th 21	S 24	M 25	W 27	Th 28	S 31	M 1	W 3	Th 4	S 7	M 8	W 10	Th 11	S 14	M 15	W 17	Th 18	S 21	M 22	W 24	Th 25
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																						■		

Each task over the next 5 weeks will involve everyone. Each task will be broken down further and delegated.

(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

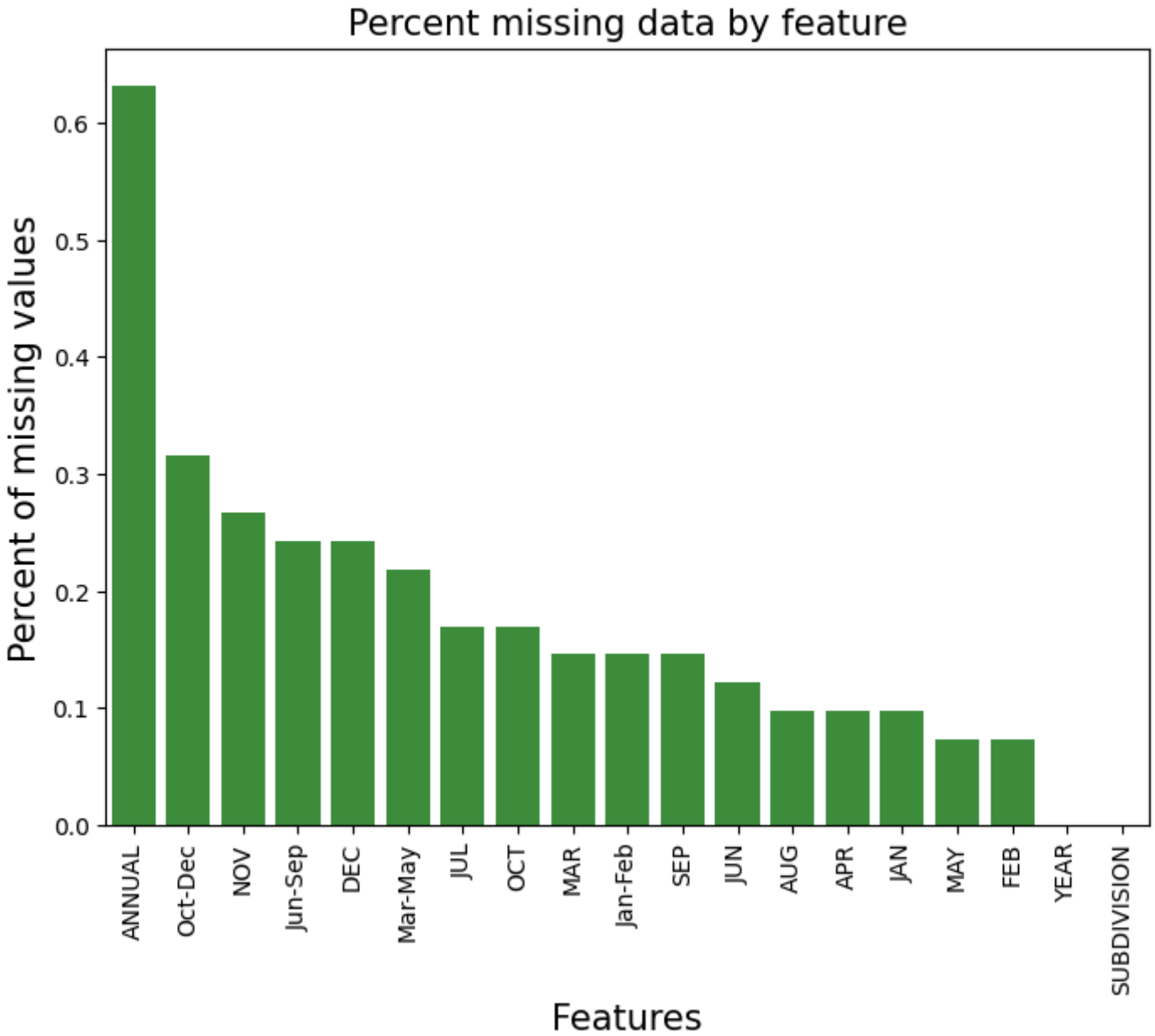


Table 5.2B - MAE For Original and New Dataset

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