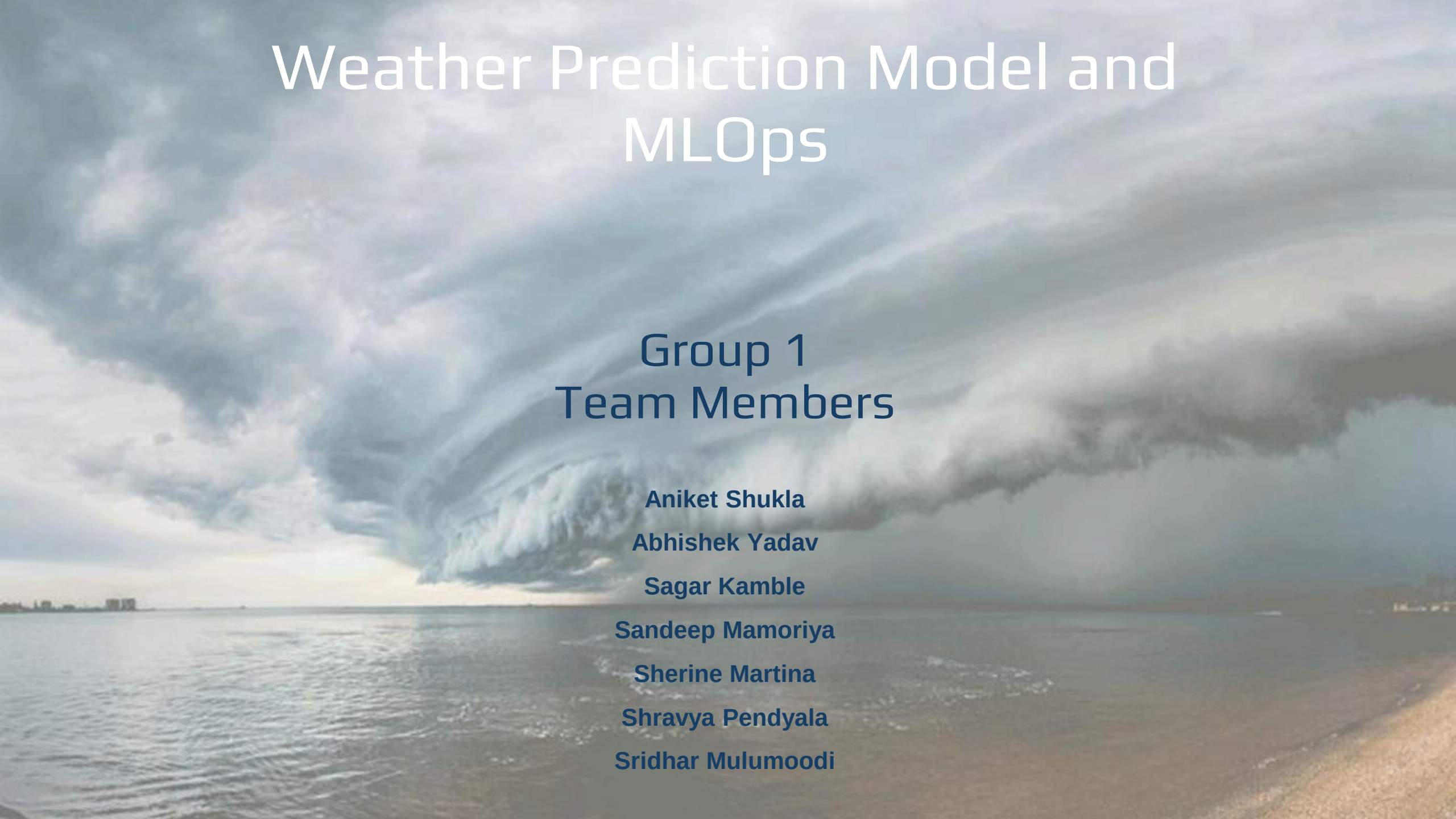


Weather Prediction Model and MLOps

Group 1
Team Members

Aniket Shukla

Abhishek Yadav

Sagar Kamble

Sandeep Mamoriya

Sherine Martina

Shravya Pendyala

Sridhar Mulumoodi

Problem Description

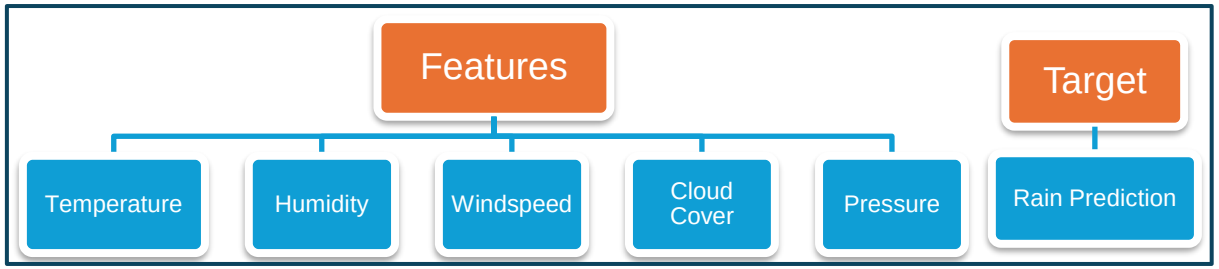
- The objective of this project is to develop a ML model that can forecast rainfall using a variety of input variables, such as temperature, humidity, wind speed, atmospheric pressure & cloud cover.
- Classification Problem

Dataset info & Cleaning

Dataset consists of 2,500 weather observations, it's a simple yet practical dataset for learning how **to predict rainfall** based on various weather conditions.

Ref: [Weather Forecast Dataset (kaggle.com)]

A	B	C	D	E	F
Temperature	Humidity	Wind_Speed	Cloud_Cover	Pressure	Rain
23.72034	89.59264	7.33560	50.50169	1032.37876	rain
27.87973	46.48970	5.95248	4.99005	992.61419	no rain
25.06908	83.07284	1.37199	14.85578	1007.23162	no rain
23.62208	74.36776	7.05055	67.25528	982.63201	rain
20.59137	96.85882	4.64392	47.67644	980.82514	no rain
26.14735	48.21726	15.25855	59.76628	1049.73875	no rain
20.93968	40.79944	2.23257	45.82751	1014.17377	no rain
32.29433	51.84847	2.87362	92.55150	1006.04173	no rain
34.09157	48.05711	5.57021	82.52487	993.73205	no rain
19.58604	82.97829	5.76054	98.01445	1036.50346	rain
29.79313	81.31765	16.92610	93.92329	1029.40269	no rain



Data importing

```
url = "https://raw.githubusercontent.com/shravyapendyala/CCE_Assignment_1/refs/heads/main/weather_forecast_data.csv"
dataset=pd.read_csv(url)
data=pd.read_csv(url)
dataset.head(10)
```

	Temperature	Humidity	Wind_Speed	Cloud_Cover	Pressure	Rain
0	23.720338	89.592641	7.335604	50.501694	1032.378759	rain
1	27.879734	46.489704	5.952484	4.990053	992.614190	no rain
2	25.069084	83.072843	1.371992	14.855784	1007.231620	no rain
3	23.622080	74.367758	7.050551	67.255282	982.632013	rain
4	20.591370	96.858822	4.643921	47.676444	980.825142	no rain
5	26.147353	48.217260	15.258547	59.766279	1049.738751	no rain
6	20.939680	40.799444	2.232566	45.827508	1014.173766	no rain
7	32.294325	51.848471	2.873621	92.551497	1006.041733	no rain
8	34.091569	48.057114	5.570206	82.524873	993.732047	no rain

Data pre-processing

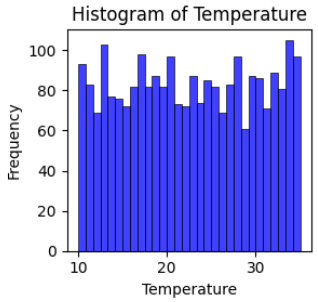
Using $LabelEncoder()$ from $sklearn$, converted the categorical data into numerical data so that it can be used for fitting the model

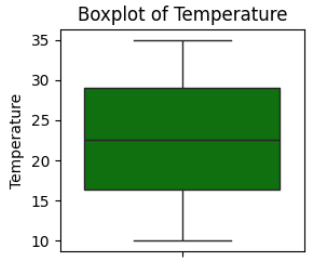
	Temperature	Humidity	Wind_Speed	Cloud_Cover	Pressure	Rain
0	23.720338	89.592641	7.335604	50.501694	1032.378759	rain
1	27.879734	46.489704	5.952484	4.990053	992.614190	no rain
2	25.069084	83.072843	1.371992	14.855784	1007.231620	no rain
3	23.622080	74.367758	7.050551	67.255282	982.632013	rain
4	20.591370	96.858822	4.643921	47.676444	980.825142	no rain

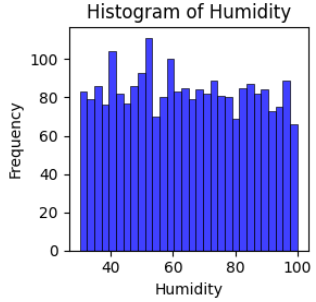
	Temperature	Humidity	Wind_Speed	Cloud_Cover	Pressure	Rain
0	23.720338	89.592641	7.335604	50.501694	1032.378759	1
1	27.879734	46.489704	5.952484	4.990053	992.614190	0
2	25.069084	83.072843	1.371992	14.855784	1007.231620	0
3	23.622080	74.367758	7.050551	67.255282	982.632013	1
4	20.591370	96.858822	4.643921	47.676444	980.825142	0

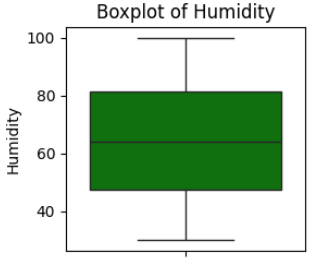
ML Methodologies

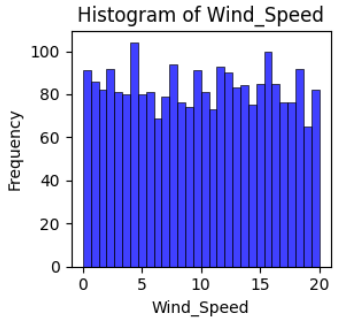
Histogram & Boxplot of Features Distribution

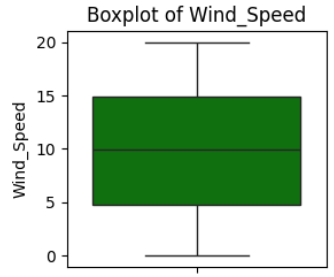


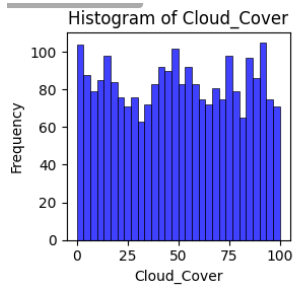


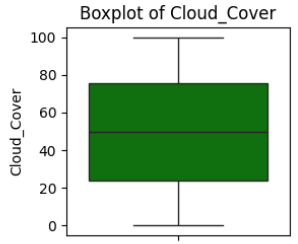


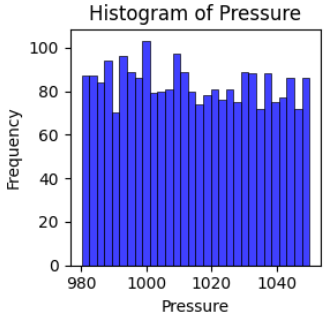


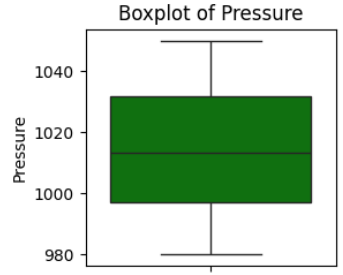












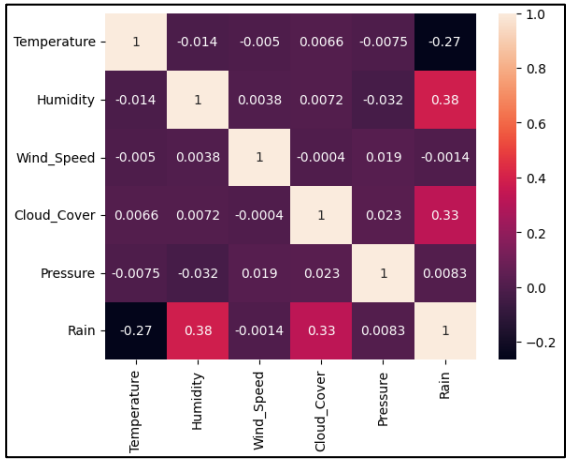
Wind Speed feature is dropped

Feature Selection

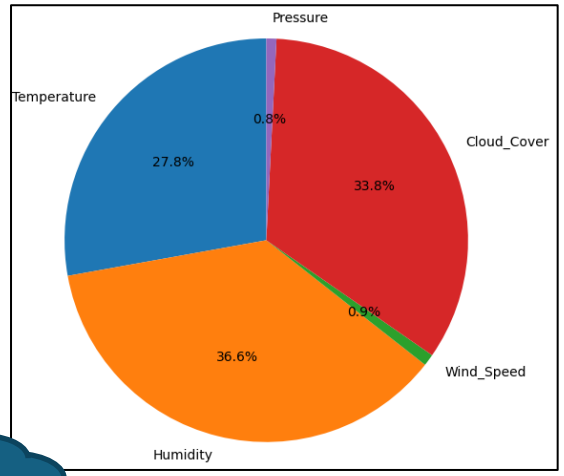
To improve model performance, reduce over-fitting, and simplify the model

Methods

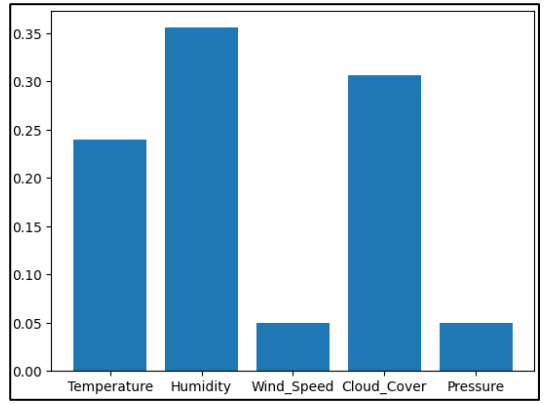
- Correlation Matrix
- Extra Tree Classifier
- Random Forest



Correlation matrix



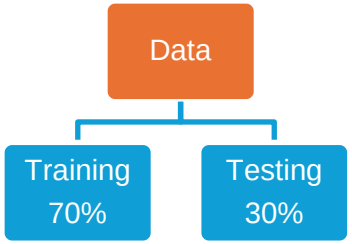
Random Forest



Extra Tree Classifier

ML Methodologies

Training and Testing the model

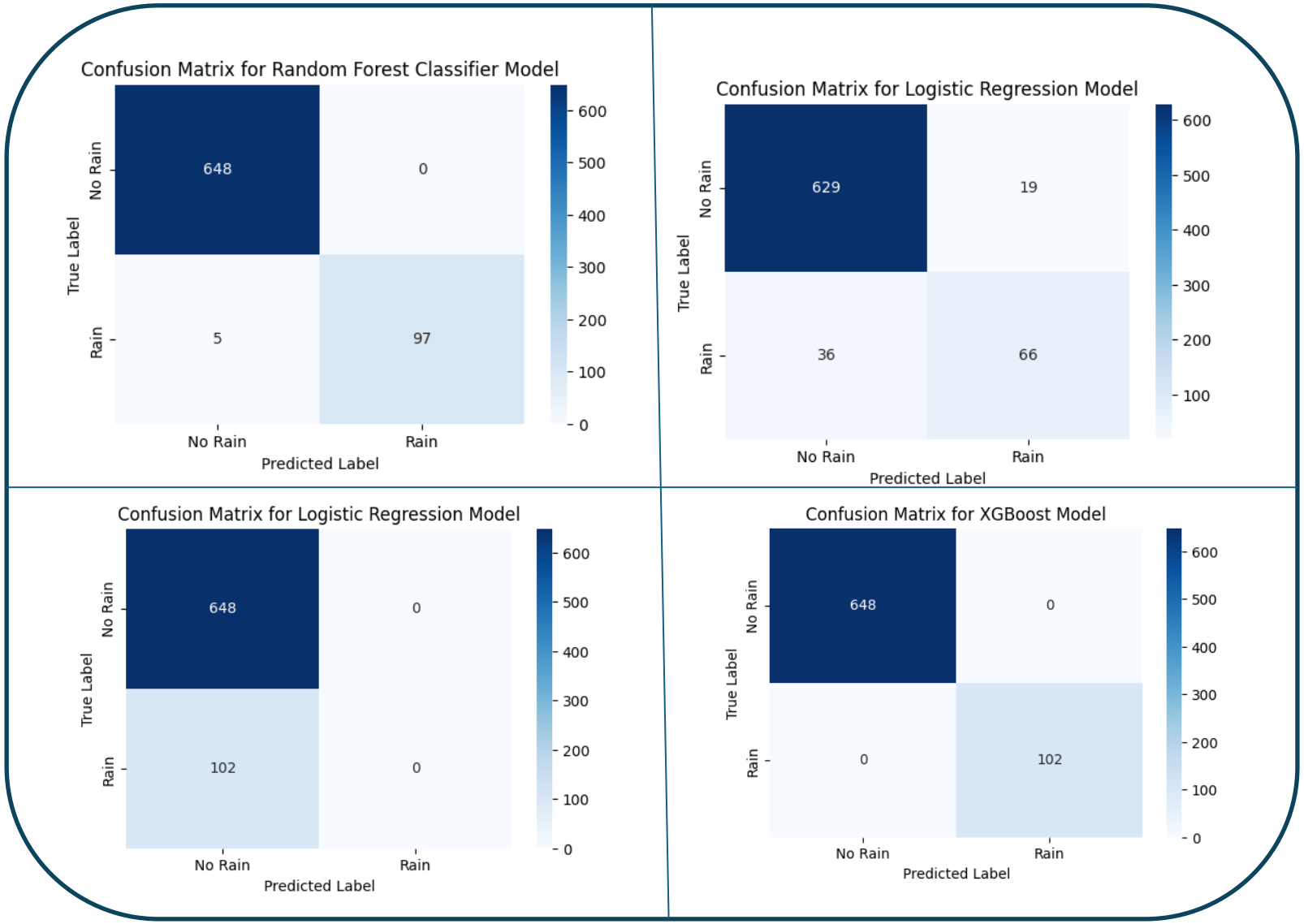


	Random Forest Classifier	Logistic Regression	SVM	XGBoost
Pros	• High accuracy • Versatile in dealing with complex datasets containing outliers • Effective classification and regression	• Good accuracy for simply datasets • Easy to implement and interpret • Interprets model coefficients as indicators of feature importance	• Models high-dimensional data • Possess good generalization performance (classifies new and unseen data)	• High accuracy and high precision • Regularization techniques avoids over-fitting
Cons	• Handling huge datasets can be time consuming • Not preferred when the model must be highly interpretable.	• Non-linear problems can't be dealt with this model • Not preferred when the number of features is more than the observations	• Handling huge datasets is not possible • Limited to two-class problems (multi-class is dealt using other strategies)	• Tuning the hyper-parameters of this algorithm can be time consuming • "Black Box" algorithm as it is difficult to interpret and understand the predictions.

Results

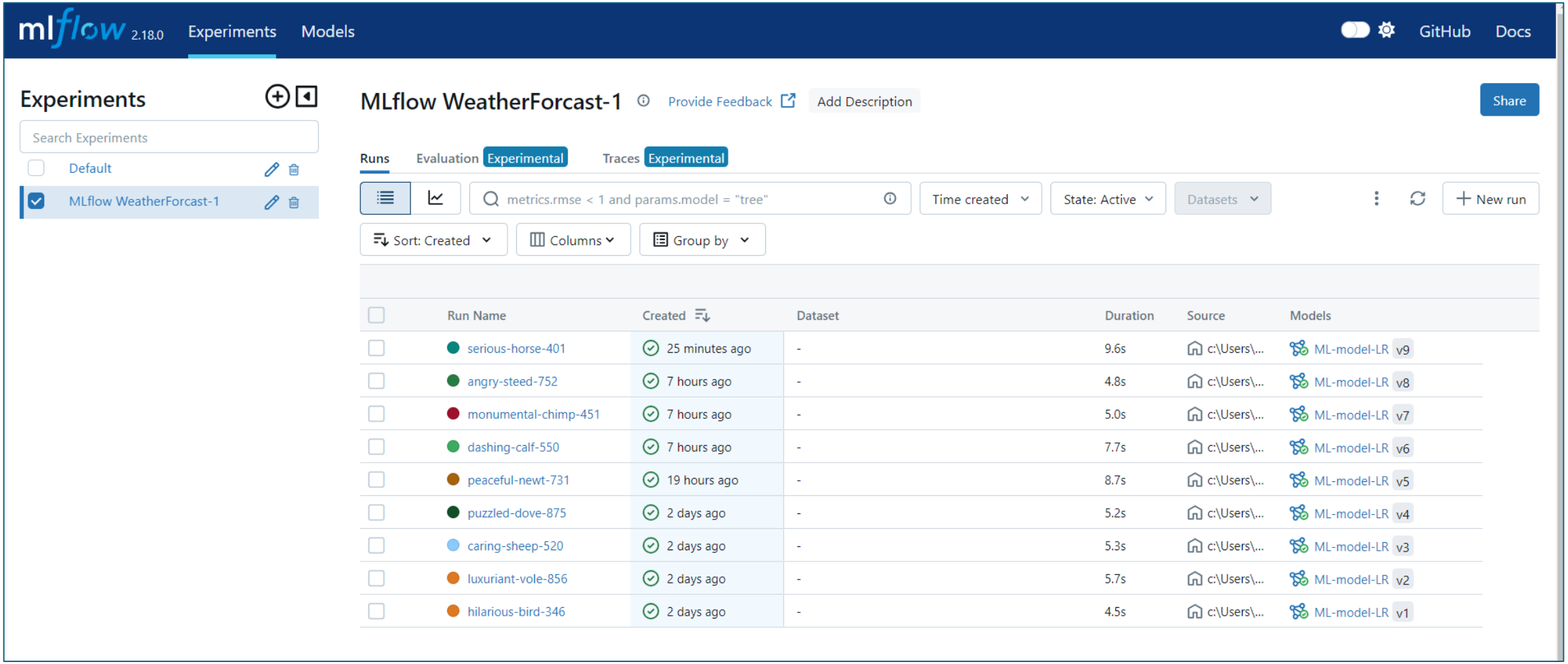
Accuracy and Confusion Matrix

Training Model	Accuracy
Random Forest Classifier	99.3
Logistic Regression	93.2
SVM	86.4
XGBoost	100



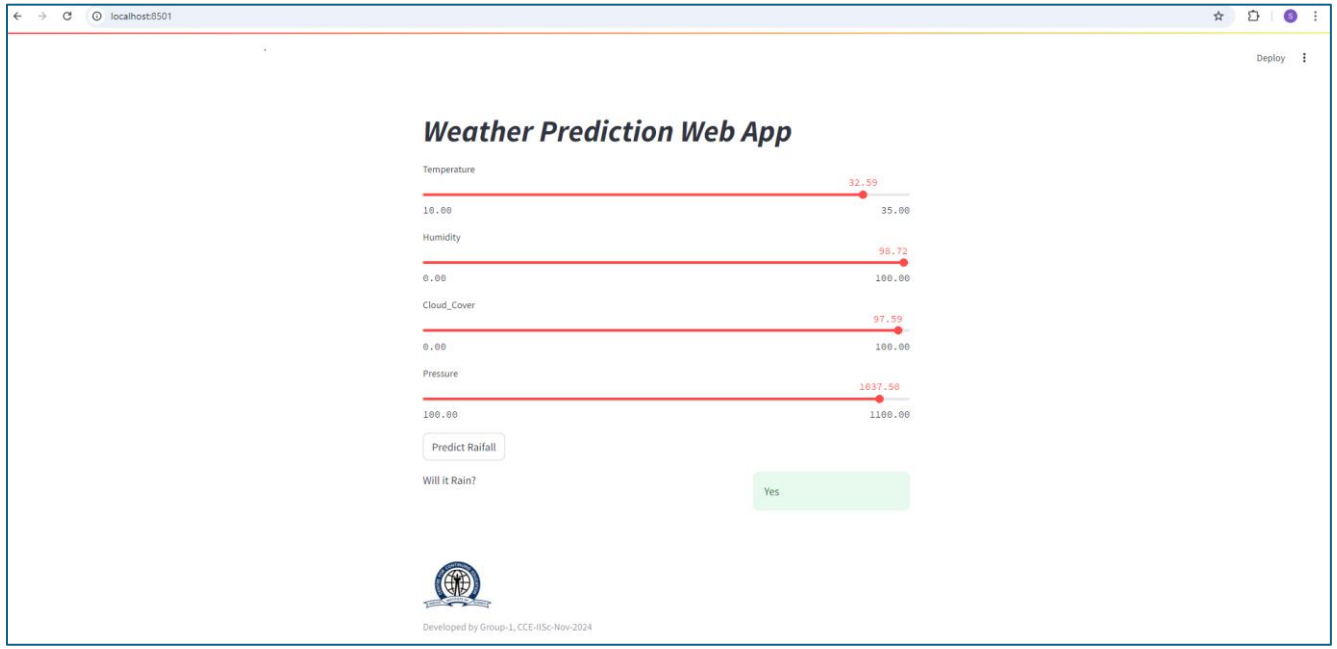
MLOps

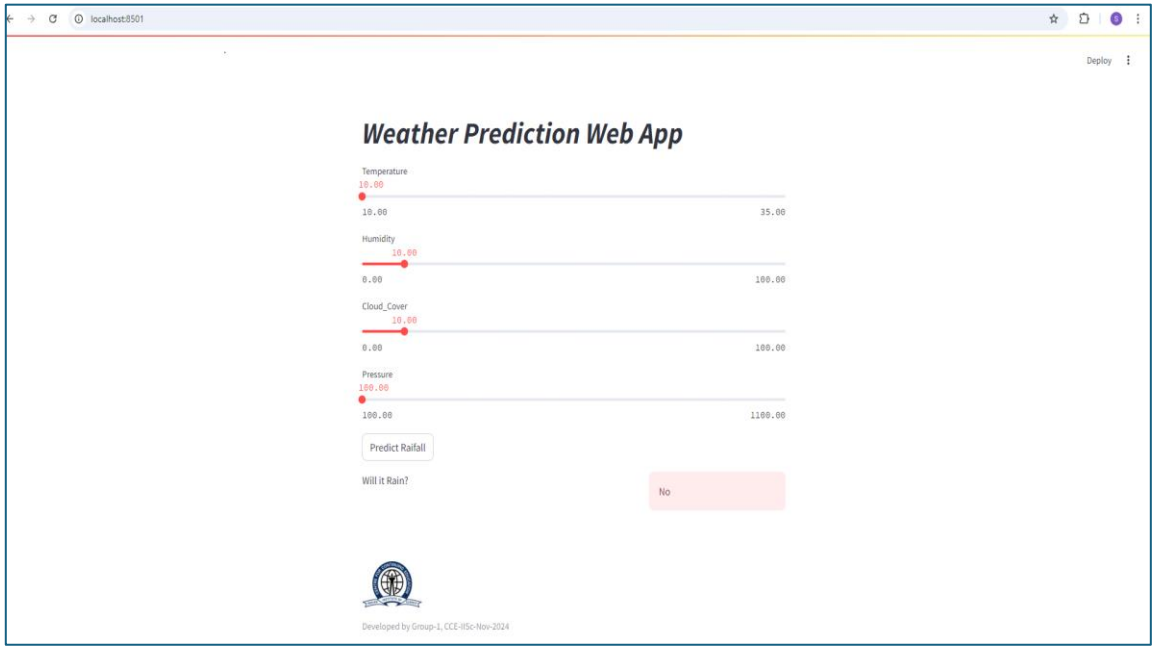
Train and store model in MLFlow and later fetch it to deploy on end user side



MLOps

Webapp using Streamlit to do live prediction using user inputs by fetching model from MLFlow registry





Contributions

Team Members	Data cleanup and acquisition	ML model selection / training	Hyper parameter tuning/MLops	Metrics	Presentation	Documentation
Aniket Shukla	5/ 100	5/ 100	5/ 100	10/ 100	5/ 100	5/ 100
Abhishek Yadav	30/ 100	30/ 100	5/ 100	20/ 100	10/ 100	30/ 100
Sagar Kamble	10/ 100	15/ 100	70/ 100	10/ 100	10/ 100	10/ 100
Sandeep Mamoriya	10/ 100	5/ 100	5/ 100	20/ 100	35/ 100	20/ 100
Sherine Martina	5/ 100	5/ 100	5/ 100	10/ 100	30/ 100	25/ 100
Shravya Pendyala	30/ 100	30/ 100	5/ 100	20/ 100	5/ 100	5/ 100
Sridhar Mulumoodi	10/ 100	10/ 100	5/ 100	10/ 100	5/ 100	5/ 100
Total	100 / 100	100 / 100	100 / 100	100 / 100	100 / 100	100 / 100

Conclusions

- Using EDA, we can nicely analyse input data/features and based on such analysis, it helps to choose/drop any feature based on their relevance
- It was helpful to check available classification models and compare them for our problem statement to get best fit. The maximum accuracy attained is using the XGBoost model which is 100%. The model looks overfitting so we chose RF model to conclude
- Using MLOps, it was very helpful to use trained ML model in reality. Streamlit helped to create webapp for user inputs and MLFlow helped to store trained models. With this, we achieved MLOps in low cost manner by hosting it on local PC servers. This approach helps in organizations who wants to keep sensitive data and models internal. Otherwise, cloud options can be used for further hosting of models.
- As a futuristic scope, long-term predictions can be estimated leading to the forecasting of rainfall conditions in a particular region