How Much Did It Rain? II

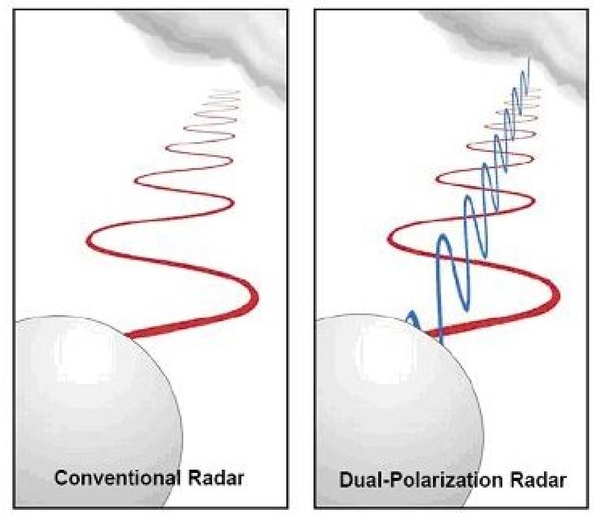
Forero Guaitero Kevin Andres [1], Lancheros Sanchez Christian Camilo [2], Delgado Jimenez Miguel Esteban [3], [*]equal contribution

[1]Universidad Distrital Francisco José de Caldas, COL; [2]Facultad de Ingeniería; [3]Bogotá D.C

Correspondence to: kaforerog@udistrital.edu.co, cclancheross@udistrital.edu.co, medelgadoj@udistrital.edu.co

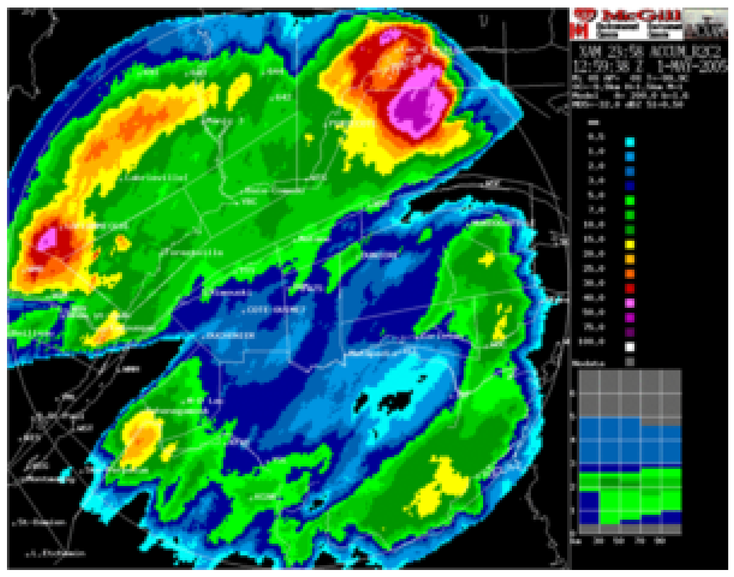
Introduction to Data Science

github.com/MiGuElEsTeBaNUD



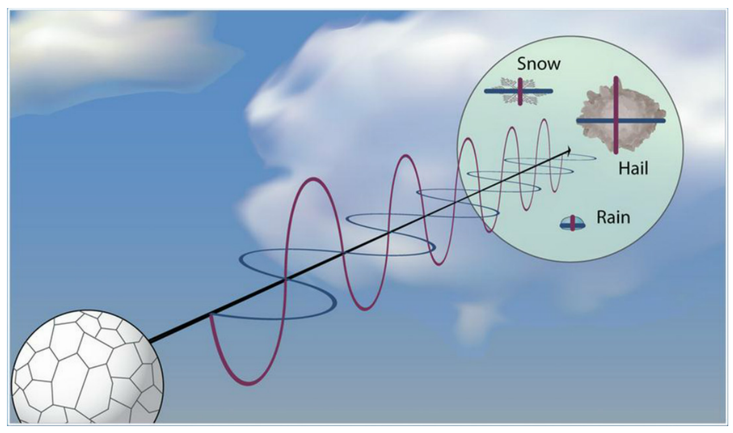
Introduction

This data analysis project in Python focuses on addressing the challenge of rainfall measurement prediction, using a new and improved data set and evaluation metric. Accurate measurement of precipitation is a complex problem due to its spatial and temporal variability. Rain gauges, although effective tools for measurement at specific locations, cannot be installed everywhere. To achieve broad coverage, data provided by weather radars are used, which estimate precipitation at the national level. However, these estimates rarely coincide exactly with direct measurements from rain gauges.



Polarimetric System

Polarimetric radars offer higher quality data compared to conventional Doppler radars as they transmit pulses of radio waves in horizontal and vertical orientations. This advance allows for more precise and detailed estimates of precipitation.



Inprove Rain Predictions

Throughout this project, we will use advanced data analysis techniques to explore and improve rainfall predictions, leveraging data from polarimetric radars and evaluating their performance with a new metric. The goal is to reduce the discrepancy between radar predictions and rain gauge measurements, thereby improving the accuracy and reliability of precipitation forecasts.

Variables to consider

- **Id**: A unique number for the set of observations over an hour at a gauge.
- **minutes_past**: Minutes past the top of the hour that the radar observations were carried out. Radar observations are snapshots at that point in time.
- **radardist_km**: Distance of the gauge from the radar whose observations are being reported.
- **Ref**: Radar reflectivity in km.
- **Ref_5x5_10th**: 10th percentile of reflectivity values in a 5x5 neighborhood around the gauge.
- **Ref_5x5_50th**: 50th percentile.
- **Ref_5x5_90th**: 90th percentile.
- **RefComposite**: Maximum reflectivity in the vertical column above the gauge, measured in dBZ.
- **RefComposite_5x5_10th**, **RefComposite_5x5_50th**, **RefComposite_5x5_90th**: Percentiles of composite reflectivity in a 5x5 neighborhood.
- **RhoHV**: Correlation coefficient (unitless).
- **RhoHV_5x5_10th**, **RhoHV_5x5_50th**, **RhoHV_5x5_90th**: Percentiles of RhoHV in a 5x5 neighborhood.
- **Zdr**: Differential reflectivity in dB.
- **Zdr_5x5_10th**, **Zdr_5x5_50th**, **Zdr_5x5_90th**: Percentiles of Zdr in a 5x5 neighborhood.
- **Kdp**: Specific differential phase (deg/km).
- **Kdp_5x5_10th**, **Kdp_5x5_50th**, **Kdp_5x5_90th**: Percentiles of Kdp in a 5x5 neighborhood.
- **Expected**: Actual gauge observation in mm at the end of the hour.

Data preprocesing

Data transform

We describe the data transformation, like this:

```
# Data Transform
transformed_df = raw_df.groupby('Id').mean()
transformed_df = transformed_df.fillna(0)


#View transformed data and stats
transformed_df.describe()
transformed_df['Expected'].describe()
transformed_df.head()
```

Feature Engineering

Feature creation

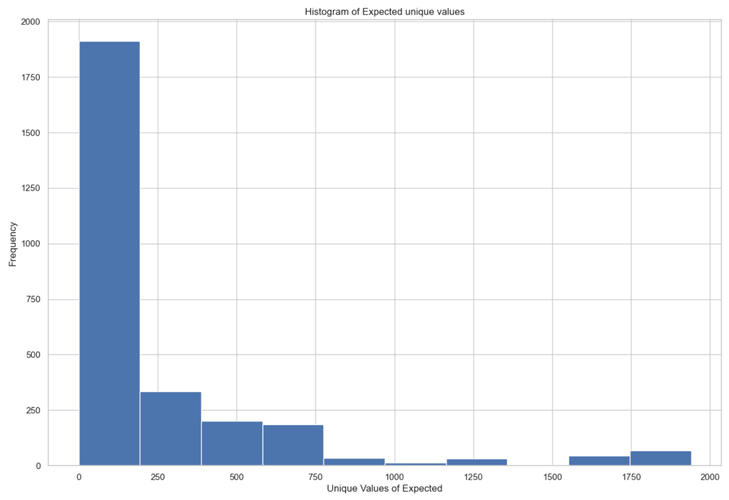
We have created a new feature called `Ref_radardist_product`, which is the product of the `Ref` (Radar Reflectivity) and `radardist_km` (Radar Distance) columns. This feature can provide useful information by combining the intensity of radar reflectivity with the distance to the radar, which could help improve prediction models.

```
import pandas as pd


file_path = 'assets/cleaned_train.csv'
df = pd.read_csv(file_path)

# Create a new feature: product of 'Ref' and 'radardist_km'
df['Ref_radardist_product'] = df['Ref'] * df['radardist_km']


output_path = 'assets/cleaned_train_with_features.csv'
df.to_csv(output_path, index=False)
```

Graphics with Matplotlib

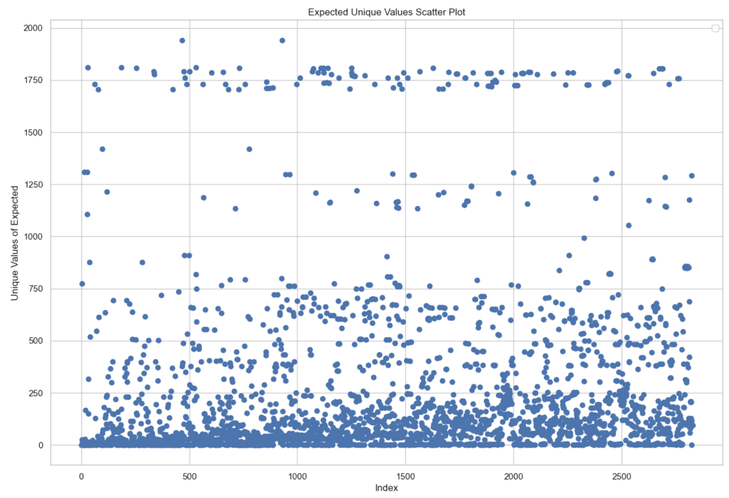
Histogram:

The histogram is useful for visualizing the distribution of numerical data or discrete categories. Shows the frequency of occurrence of different values or ranges of values in a data set.



Dispersion diagram:

The scatter plot visualizes the relationship between two numerical variables. Each point on the graph represents an individual observation on a Cartesian plane.



Model selection and evaluation:

Random Forest and NN Kerast:

The choice between Random Forest (RF) and neural networks in Keras (NN Keras) depends on the characteristics of the problem and the data. RF is preferable when direct interpretability of model decisions, robust handling of missing data and outliers, computational efficiency on moderately large data sets, and stability against overfitting are required. In contrast, NN Keras is better suited for capturing complex, non-linear relationships in large data sets, improving with more data and allowing hierarchical representation of features. The choice is based on whether to prioritize interpretability and processing efficiency (RF) or the ability to model complex relationships and leverage large amounts of data (NN Keras).

Model	MSE	MAE	R-squared	Training Time
Random Forest	14802.15	32.49	0.5259	15m 75s
NN (Keras)	25709.02	55.92	0.1765	46m 52.1s

Final conclusion

Model selection and evaluation:

Random Forest and NN Kerast:

This study aimed to contribute to the advancement of rainfall prediction by leveraging polarimetric radar data and machine learning techniques within the framework of the ”How Much Did It Rain?” competition. By applying data preprocessing, feature engineering, and model development, we sought to improve the accuracy of hourly rainfall predictions. Our findings demonstrate the potential of machine learning models in predicting rainfall using polarimetric radar data. While the specific performance metrics achieved in this study require further analysis and comparison with other approaches, the overall methodology provides a solid foun- dation for future research. The use of data visualization techniques throughout the analysis process proved invaluable in understanding the complex relationships within the data and interpreting the model’s results. Future research could explore the incorporation of ad- ditional meteorological variables, such as temperature, hu- midity, and wind speed, to enhance model performance. Additionally, investigating the impact of different machine learning algorithms and hyperparameter tuning strategies is warranted. By continuing to refine rainfall prediction models, we can contribute to improved weather forecasting, disaster pre- paredness, and water resource management.

Referencias

References

[1] Breiman, L. (2001). Random forests. *Machine learning*, 45(1), 5-32.

[2] Chollet, F. (2015). Keras: Deep learning library for Theano and TensorFlow. URL: `https://keras.io`

[3] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.