Boston Weather Prediction using MLP and Time Series

Siraj Akmal and Adam Ma

Northeastern University

Introduction/Abstract

This project aims to predict rainfall patterns in Boston using two different modeling approaches on historical weather data scraped from Weather Underground. First, we built a manually coded Multi-Layer Perceptron (MLP) to classify whether it will rain on a given hour, using features selected through a combination of correlation analysis and domain knowledge. The model achieved 83% accuracy on a balanced test set, confirming its ability to capture meaningful weather patterns. Second, we applied an AutoRegressive (AR) model to forecast total monthly precipitation. Time series decomposition revealed clear trend and seasonal components, and the AR(2) model showed significant dependence on the two previous months. Together, these models provide both daily-level and monthly-level insights into precipitation behavior in Boston.

Motivation and Data

As students living in Boston, we often found ourselves frustrated by the unpredictability of rain during our daily routines. This inspired us to explore whether machine learning models could help forecast rainfall more accurately and meaningfully on both a daily and monthly basis. We noticed that rainy days in Boston tend to come in clumps, suggesting underlying temporal patterns worth modeling.

To investigate this, we scraped over 40,000 hourly weather observations spanning four years from Weather Underground. The dataset includes a wide range of features such as temperature, dew point, humidity, heat index, wind speed and gusts, wind direction, pressure (inHg), pressure trend, UV index, visibility, cloud cover, and detailed precipitation metrics (e.g., hourly rainfall and snowfall).

For the classification task, we used this hourly dataset to predict whether precipitation occurred in a given hour. We engineered a binary label based on the "Hourly Precip (in)" column, set to 1 if any measurable rainfall was recorded. To prepare the inputs for modeling, we standardized continuous features and encoded cyclical variables such as wind direction into sine and cosine components. We also mapped pressure trends (e.g., rising, falling, steady) into numerical codes.

To address the severe class imbalance — with rain occurring in only 500 out of 40,000 hours — we created a balanced test set of 1,000 samples (500 rainy, 500 dry) for evaluation. This ensured that model performance reflected actual learning and not bias toward the majority class.

For the time series forecasting task, we aggregated the hourly precipitation values into monthly totals. This transformation enabled us to model broader rainfall patterns using temporal models. The cleaned and enriched dataset formed the foundation for both our MLP classifier and our AR(2) forecasting model.

Feature	Count	Mean	Std Dev	Min	Max
Temperature (°F)	41452	53.63	16.70	-9.0	99.0
Dew Point (°F)	41444	42.51	18.35	-31.0	77.0
Wind Speed (mph)	41440	10.85	5.53	0.0	110.0
Wind Gust (mph)	6618	27.56	6.42	16.0	68.0
Humidity (Hourly Precip (in)	41454	0.01	0.04	0.0	1.23
Visibility (mi)	41449	8.61	2.93	0.0	10.0
Pressure (inHg)	41452	29.97	0.25	28.63	30.76
UV Index	41440	0.72	1.37	0.0	9.0

Summary statistics for key weather features in the dataset.

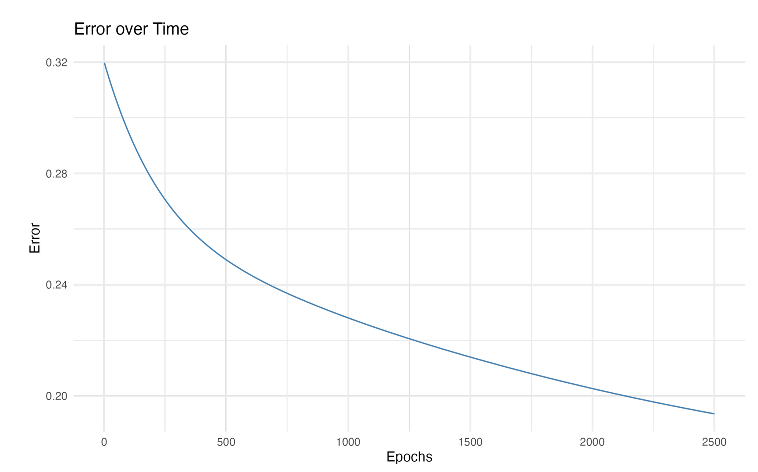
Methodology

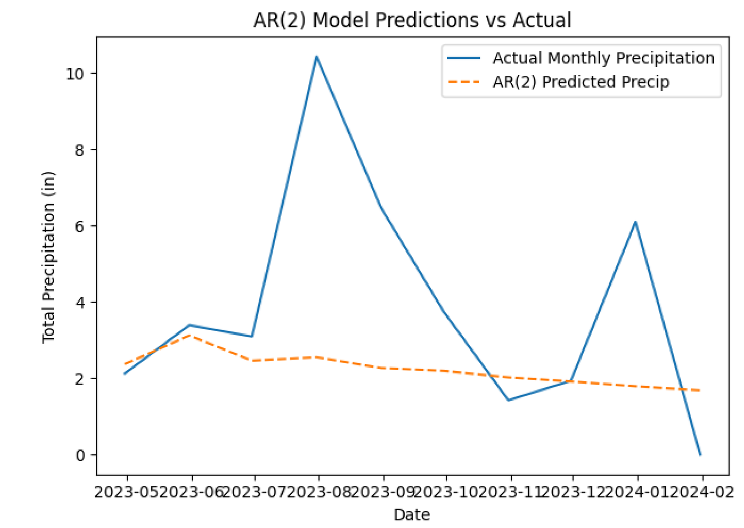
We developed two separate models to approach the problem from both short-term and long-term forecasting perspectives. For the classification task, we manually implemented a Multi-Layer Perceptron (MLP) from scratch using NumPy. The model architecture included a single hidden layer with a ReLU activation function and a sigmoid output layer. We trained the model using backpropagation with gradient descent, minimizing binary cross-entropy loss. Input features were selected through a combination of correlation analysis and domain knowledge, and included wind speed, pressure, humidity, UV index, hour of the day, and pressure trend direction.

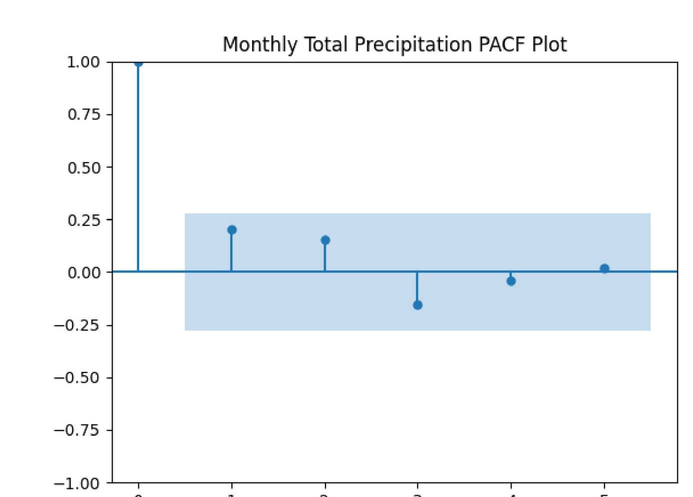
For the time series component, we aggregated the daily precipitation data into monthly totals and used an AutoRegressive (AR) model of order 2 to capture temporal dependencies. Model order was selected based on partial autocorrelation analysis, and the AR(2) model was trained using conditional maximum likelihood estimation. We evaluated model performance through time series decomposition, which revealed a clear seasonal structure and a small, random residual component, indicating a good model fit.

Results

We evaluated each model using relevant visualizations and metrics. The MLP model showed a steady decline in training loss, confirming the effectiveness of our backpropagation implementation. For the AR model, predictions tracked seasonal trends to some extent, and PACF analysis justified the AR(2) structure.







Key Results:

- MLP model reached 83% accuracy on a balanced test set after training on 4 years of daily weather data.
- Training loss decreased consistently over 2500 epochs, indicating convergence.
- AR(2) model captured seasonality but underpredicted extreme monthly precipitation values.
- PACF plot supported the use of two autoregressive lags.
- Both models offer complementary short-term (daily) and long-term (monthly) insights into Boston rainfall.

Discussion / Future Work / Web App

While our models produced promising results for Boston-specific rainfall prediction, there are clear opportunities for future improvement. First, expanding the dataset to include more years and additional geographic regions would allow us to detect broader climatic trends and improve model generalizability. We also see value in exploring more advanced time series models and hybrid approaches. In particular, incorporating external variables like temperature anomalies, climate indices, or satellite-derived data could enhance forecasting performance. Additionally, future iterations of this project could experiment with convolutional neural networks (CNNs) to extract spatial features from weather radar images. These spatial patterns could help the model distinguish between different types of precipitation events and produce more nuanced and accurate forecasts.

Our web app is publicly hosted online and accessible to anyone interested in exploring the project further. It provides an interactive dashboard with deeper insights into the models and dataset. For the time series component, the app includes plots for the ACF and PACF, full seasonal-trend decomposition of monthly precipitation, MA Models,and AR(2) model predictions. It also contains detailed summaries and visualizations of model performance, offering a transparent view into our methodology and results.

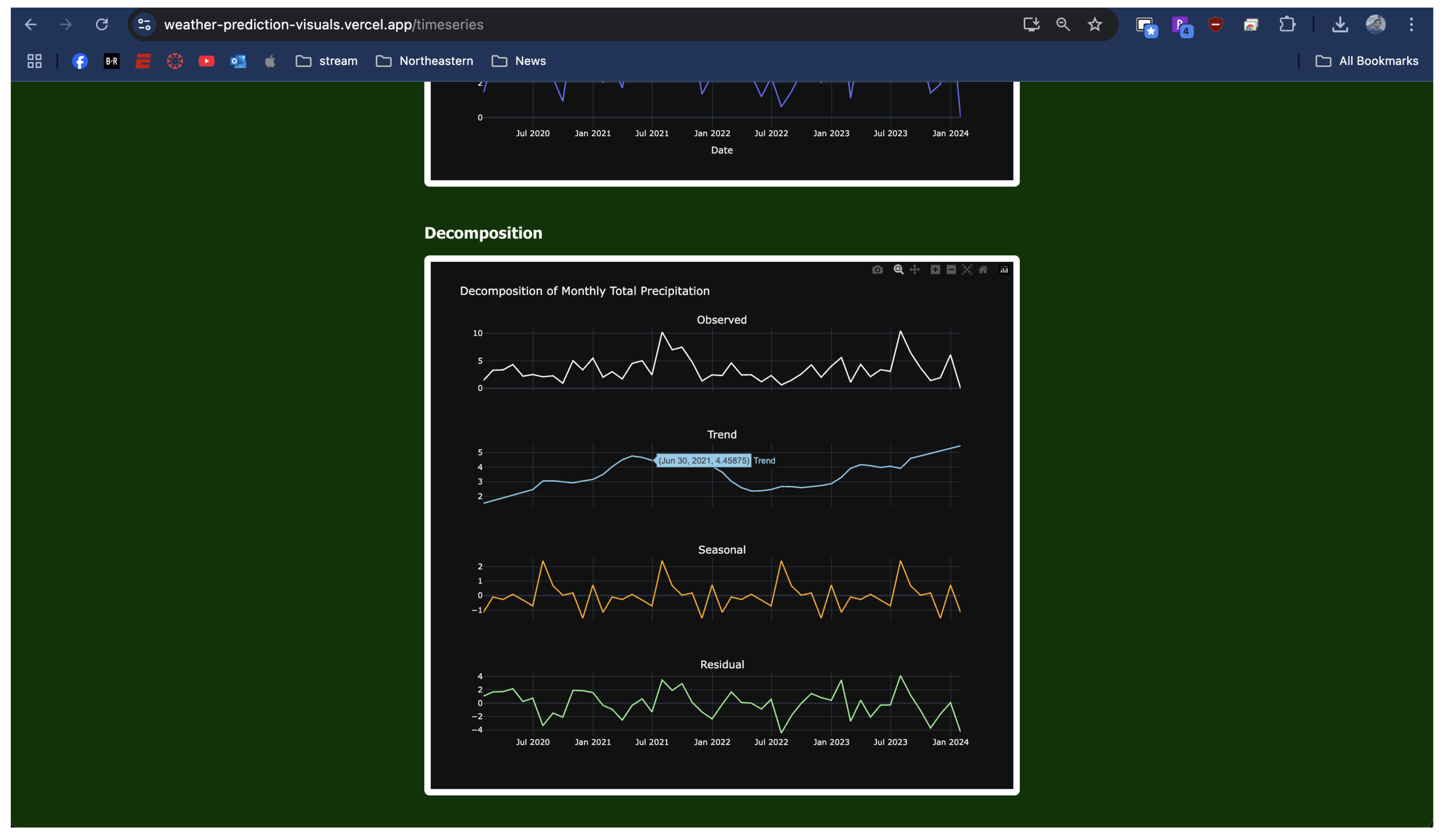


Fig. 1: Screenshot of our interactive weather prediction web app.

Scan to explore model outputs and visuals.