TIME SERIES ANALYSIS AND FORECASTING

Project Overview

****Project:** Time Series Analysis and Forecasting of Weather Data**

****Overview:****

In this project, I developed a machine learning model to forecast weather data using historical data from Wyoming. The project involved various stages including data cleaning, feature engineering, exploratory data analysis, model selection, evaluation, and comparison of different time series forecasting techniques. The primary goal was to build accurate and efficient predictive models to provide insights into future temperature trends.

****Problem Statement:****

The objective of this project was to predict daily temperature values based on historical weather data. Accurate weather forecasting is crucial for numerous applications including agriculture, energy management, and daily planning.

****Tools & Technologies:****

- **Python:** The primary programming language used.
- **Pandas & NumPy:** For data manipulation and analysis.
- **Matplotlib & Seaborn:** For data visualization and exploratory data analysis.
- **Statsmodels & Prophet:** For building and evaluating time series forecasting models.
- **Jupyter Notebooks:** For interactive development and documentation.

Data Preparation

****Data Cleaning:****

- Handled missing values and inconsistencies in the dataset.
- Resampled the hourly data to daily frequency to reduce noise and computational load.

Feature Engineering:

- Extracted relevant time-based features to enhance model accuracy.
- Created additional time series features such as trends and seasonality.

Exploratory Data Analysis (EDA)

Visualized distributions and relationships between features using histograms, box plots, and heatmaps:

- Identified key patterns and trends in temperature data over time.
- Analyzed seasonal variations and long-term trends.

Algorithm Selection:

Evaluated multiple time series forecasting algorithms including:

- ARIMA (AutoRegressive Integrated Moving Average)
- Prophet

Model Tuning:

Used cross-validation and grid search to optimize hyperparameters and improve model performance.

Evaluation Metrics:

Assessed models using MAE (Mean Absolute Error), MSE (Mean Squared Error), and RMSE (Root Mean Squared Error) to ensure balanced and robust performance.

Results

Performance:

- **ARIMA Model:**

ARIMA MAE: 6.465299053087701
ARIMA MSE: 59.510877822677834
ARIMA RMSE: 7.714329382563195

- **Prophet Model:**

Prophet MAE: 3.3079407179234535
Prophet MSE: 16.20309769380967
Prophet RMSE: 4.025307155213087

Key Insights:

- **Seasonal Patterns:** Identified clear seasonal patterns in temperature data.

- **Trend Analysis:** Long-term trend analysis provided insights into climate patterns.

Deployment

Interactive Web Application:

Deployed the model using Flask, allowing users to input data and get real-time temperature forecasts.

User Interface:

Designed a simple and intuitive interface for users to enter details and obtain predictions.

Key Learnings

- **Data Preprocessing:** The importance of handling missing data and creating meaningful features to improve model accuracy.

- **Model Evaluation:** Understanding the trade-offs between different performance metrics and selecting the best model based on balanced performance.

- **Deployment:** Gaining practical experience in deploying a machine learning model and creating an interactive user interface for real-world applications.

GitHub Repository

Explore the complete project and code here:

https://github.com/sravanthi224/Quanta-4.git