**Project Report**

**Weather Data Analysis and Prediction**

# 1. Introduction

**Weather prediction is a key application of data science and machine learning, providing insights into future trends by analyzing historical patterns. This project involves collecting, processing, and modeling historical temperature data to forecast future temperatures using regression techniques.**

# 2. Objective

**The main objective of this project is to:  
- Analyze daily temperature data over one year.  
- Build a regression model to predict future temperatures.  
- Visualize actual and predicted results.  
- Evaluate model performance using appropriate metrics.**

# 3. Tools and Technologies Used

**- Programming Language: Python  
- Libraries:  
- Pandas  
- NumPy  
- scikit-learn  
- Matplotlib  
- Seaborn  
- Jupyter Notebook  
- Development Environment: Visual Studio Code**

# 4. System Requirements

**- Python 3.x  
- pip package manager  
- Operating System: Windows, Linux, or macOS  
- Recommended RAM: Minimum 4GB**

# 5. Methodology

**The project workflow includes:  
  
1. Data Collection:  
 Generated synthetic daily temperature data covering 365 days.  
  
2. Data Preprocessing:  
 - Cleaning missing or invalid data.  
 - Adding features such as day of the year.  
  
3. Model Building:  
 - Splitting data into training (80%) and testing (20%).  
 - Using Linear Regression to fit the model.  
  
4. Evaluation:  
 - Calculating RMSE (Root Mean Squared Error) and MAE (Mean Absolute Error) on the test set.  
  
5. Prediction:  
 - Forecasting temperatures for the next 30 days.  
  
6. Visualization:  
 - Plotting actual vs. predicted values for clear understanding.**

# 6. Project Structure

**The project files are organized as follows:  
  
weather-prediction-pro/  
├── data/  
│ └── historical\_weather.csv  
├── notebooks/  
│ └── EDA\_and\_Modeling.ipynb  
├── src/  
│ ├── data\_preprocessing.py  
│ ├── model.py  
│ └── utils.py  
├── .vscode/  
│ ├── settings.json  
│ └── launch.json  
├── .gitignore  
├── README.md  
├── requirements.txt  
└── main.py  
  
Description of key files:  
- data/historical\_weather.csv: Contains historical daily temperature data.  
- main.py: Orchestrates the data loading, model training, prediction, and evaluation.  
- src/: Holds all modular scripts for data preprocessing, modeling, and utilities.  
- notebooks/: Jupyter notebook for exploratory data analysis.  
- requirements.txt: List of dependencies to install.  
- .vscode/: VS Code configuration for a ready-to-run setup.**

# 7. Results and Discussion

**Model Evaluation:  
  
The model was evaluated on the test data using two metrics:  
  
- RMSE (Root Mean Squared Error): Indicates the standard deviation of prediction errors.  
- MAE (Mean Absolute Error): Measures average magnitude of errors.  
  
Sample Output:  
  
Test RMSE: 2.10  
Test MAE: 1.70  
  
Visualization:  
The following graph shows the comparison between actual and predicted temperatures:  
  
[Insert your screenshot here]  
  
Future Predictions:  
The model successfully forecasted temperatures for the next 30 days.  
  
[Insert table or screenshot of predictions here]**

# 8. Conclusion

**This project demonstrates how machine learning techniques can be applied to time series data for temperature prediction. The model achieved satisfactory accuracy and provided clear visualization of trends. Future improvements may include:  
  
- Incorporating more advanced models (e.g., Random Forest, ARIMA).  
- Using real-world weather datasets.  
- Adding additional features such as humidity or wind speed.**

**Prepared by:  
Abhishek Singh  
Codec Technologies**