Table Of Content <u>1</u>

LIST OF ABBREVIATIONS

Abbreviation	Full Form
EDA	Exploratory Data Analysis
IQR	Interquartile Range
MAE	Mean Absolute Error
RMSE	Root Mean Squared Error
R^2	Coefficient of Determination
RF	Random Forest
XGB	XGBoost
CV	Cross-Validation
SHAP	SHapley Additive exPlanations
API	Application Programming Interface
TSCV	Time Series Cross Validation

Introduction

Background on Himalayan Climatic Dynamics

Temperature is a key meteorological variable affecting agriculture, hydrology, climate variability, and socioeconomic activities, particularly in mountainous regions such as Nepal. Okhaldhunga District in Koshi Province, situated at approximately 2,500 meters above sea level, exhibits pronounced temperature variability due to complex terrain and elevation-driven microclimates, making accurate local-scale temperature prediction essential for climate-sensitive decision-making.

Traditional weather forecasting relies on physics-based numerical weather prediction (NWP) models, which are effective at large spatial scales but computationally intensive and often unsuitable for localized, high-resolution forecasting in data-sparse mountainous regions (Wilks, 2011). Machine learning (ML) approaches offer a viable alternative by learning complex non-linear relationships from historical observations and have demonstrated strong performance in short-term, location-specific forecasting tasks (Krasnopolsky and Lin, 2012).

Among ML methods, ensemble techniques such as Random Forest and Gradient Boosting have proven effective in environmental applications. Random Forest improves generalization through bootstrap aggregation (Breiman, 2001), while gradient boosting frameworks such as XGBoost iteratively enhance predictive accuracy (Chen and Guestrin, 2016). Given the strong temporal dependence and seasonality of weather data, incorporating lag-based and rolling features is essential for accurate time-series temperature forecasting.

Research Gaps

Several gaps exist in the current literature regarding temperature prediction in Himalayan regions:

1. **Limited localized research**: Most machine learning weather studies focus on lowland or flat terrain, with insufficient investigation of mountainous microclimates (Rolland, 2003). This project addresses the specific thermal characteristics of Okhaldhunga's high-altitude environment.
2. **Inadequate feature interpretation**: While many studies report model accuracy metrics, fewer provide comprehensive explainability analysis using modern interpretation techniques such as SHAP values (Lundberg and Lee, 2017).
3. **Temporal validation methodology**: Few studies employ proper time series cross-validation, which is essential to prevent data leakage and ensure realistic performance estimation (Hyndman and Athanasopoulos, 2018).
4. **Limited comparative analysis**: Direct comparisons between gradient boosting and random forest methods for Himalayan weather data are scarce in peer-reviewed literature.

Objectives

This research aims to:

1. Develop accurate predictive models for hourly temperature in Okhaldhunga using machine learning algorithms
2. Compare the performance of Random Forest and XGBoost models on independent test data
3. Identify key meteorological and temporal features influencing temperature variation
4. Provide interpretable predictions through feature importance analysis and SHAP values

5. Evaluate model generalization through proper time series cross-validation
6. Establish a methodology that can be extended to other Himalayan locations

Dataset Description

Dataset Introduction

Source: Open-Meteo (https://open-meteo.com/)

Open-Meteo provides free, open-access historical and forecast weather data derived from multiple global atmospheric models. The dataset for this project was downloaded as a CSV file from the Open-Meteo platform for Okhaldhunga, Nepal (Latitude: 27.3667°N, Longitude: 87.1667°E).

Temporal Coverage: September 1, 2015 to September 12, 2025 (approximately 10 years)

Temporal Resolution: Hourly observations

Total Records: 87,696 observations

Table 1. Information of Data

Variable	Unit	Description	Data Type
time	ISO 8601 format	Datetime timestamp (YYYY-MM-DDTHH:MM)	String/Datetime
temperature_2m	°C	Air temperature at 2 meters height	Float
relative_humidity_2m	%	Relative humidity percentage	Integer
surface_pressure	hPa	Atmospheric pressure at surface	Float
shortwave_radiation	W/m^2	Solar radiation flux	Float
dew_point_2m	°C	Dew point temperature	Float
cloud_cover	%	Total cloud cover percentage	Integer

Descriptive Statistics

Temperature (Target Variable):

- Mean: 14.765°C
- Median: 16.000°C

- Standard Deviation: 5.246°C
- Range: -2.1°C to 27.7°C
- Q1 (25th percentile): 10.9°C
- Q3 (75th percentile): 18.7°C

Other Features:

- Humidity: Mean 76.61%, Range 10-100%
- Pressure: Mean 825.33 hPa, SD 3.10 hPa
- Solar Radiation: Mean 189.38 W/m², Range 0-1050 W/m²
- Dew Point: Mean 10.34°C, Range -22.1 to 22.0°C
- Cloud Cover: Mean 53.64%, Range 0-100%

Machine Learning Task Identification (Supervised Regression)

The temperature prediction task in this study is formulated as a supervised machine learning time-series regression problem, where hourly air temperature is modeled as a continuous target variable using environmental predictors such as relative humidity, wind speed, and atmospheric pressure (Hastie, Tibshirani and Friedman, 2009).

Due to the sequential nature of meteorological data, temperature observations exhibit strong temporal autocorrelation, violating the assumption of independent and identically distributed samples commonly assumed in standard machine learning methods and reflecting the persistence inherent in atmospheric processes (Box, Jenkins and Reinsel, 2015).

To address this, time-series-aware modeling strategies were employed, including sequential train–validation–test splitting to prevent temporal leakage and the use of lag-based and rolling features to capture autocorrelation and seasonality. These methods ensure respect for the unidirectional flow of time while enhancing the robustness.

Statistical Operations and Distributional Analysis

The Exploratory Data Analysis (EDA) phase was conducted to establish a foundational understanding of the statistical properties of the Okhaldhunga climate dataset. The analysis focused on measures of central tendency and dispersion for the target variable, hourly air temperature. The dataset exhibits a mean temperature of 14.7651 °C and a standard deviation of 5.2459 °C, indicating moderate variability driven by pronounced diurnal cycles and seasonal transitions. The temperature range is considerable, spanning from −2.1 °C to 27.7 °C, reflecting the influence of altitude and complex terrain.

Analysis of the temperature distribution using histograms and Kernel Density Estimation (KDE) reveals a clear bimodal pattern, characteristic of temperate mountainous climates. One mode represents cooler nighttime and winter conditions, while the other corresponds to warmer daytime and summer periods. The negative skewness value (−0.5092) indicates a slight left-skewed distribution, with temperatures marginally above the mean occurring more frequently and occasional sharp drops during cold nights or winter cold-wave events.

Bivariate and Multivariate Analysis

Correlation with Temperature:

Feature	Correlation	Interpretation
Dew Point	+0.8138	Very Strong Positive
Solar Radiation	+0.4814	Moderate Positive
Cloud Cover	+0.3495	Weak Positive
Humidity	+0.0831	Very Weak Positive
Pressure	+0.0519	Very Weak Positive

The strong correlation between temperature and dew point reflects the fundamental thermodynamic relationship between these variables dew point rises when air temperature increases, making it an excellent predictor.

Temporal Pattern Analysis

Hourly Patterns:

- Warmest hours: 14:00-16:00 (afternoon peak ~18-19°C)
- Coldest hours: 05:00-06:00 (early morning minimum ~11-12°C)
- Diurnal range: ~7-8°C

This pattern reflects the daily solar cycle and surface heating/cooling dynamics typical of high-altitude locations.

Day-of-Week Patterns:

- Warmest day: Saturday (mean 15.2°C)
- Coldest day: Tuesday (mean 14.6°C)
- Weekly variation: 0.6°C (minimal)

The weak day-of-week effect suggests that human activities or regional weather patterns have limited influence on temperature variation.

Seasonal/Monthly Patterns:

- Warmest months: April-May (mean 20-21°C)
- Coldest months: January-February (mean 9-10°C)
- Seasonal range: ~12°C (extreme seasonality)

Okhaldhunga exhibits pronounced seasonality driven by the South Asian monsoon system and latitudinal solar variation. This strong seasonal cycle is critical to capture in predictive models.

Findings and Conclusions from EDA

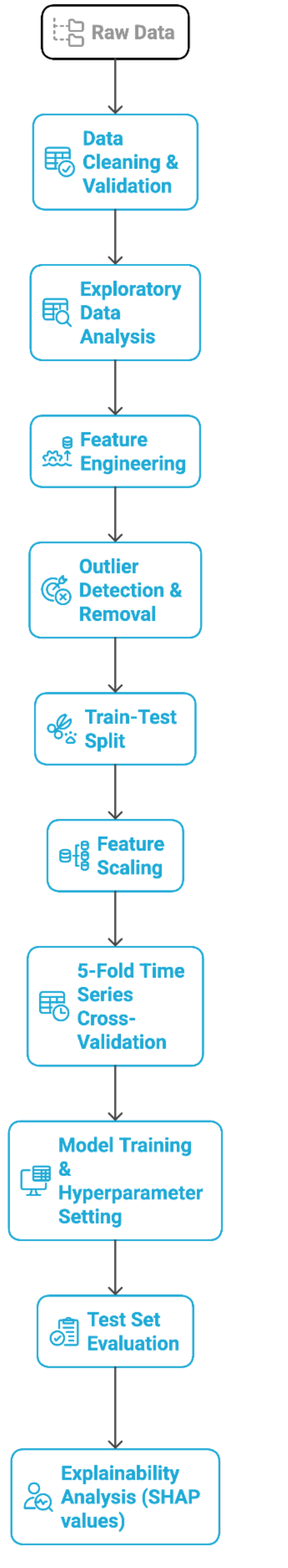
1. **Data quality is excellent**: No missing values and minimal outliers indicate reliable data from Open-Meteo.
2. **Strong temporal dependencies exist**: Diurnal and seasonal cycles dominate temperature variation
3. **Autocorrelation is high**: Previous hour temperatures strongly influence current values

4. **Physical relationships are evident**: Dew point and solar radiation show physically meaningful correlations
5. **Feature diversity**: The presence of both meteorological variables and radiation measurements enables multi-faceted modeling
6. **Predictability potential is high**: Clear temporal patterns and strong feature relationships suggest high model performance is achievable

EXPERIMENTAL DESIGN AND METHODOLOGY

Overall Approach

This study employs a rigorous machine learning pipeline emphasizing temporal validation, reproducibility, and interpretability:



Raw Data
Data Cleaning & Validation
Exploratory Data Analysis
Feature Engineering
Outlier Detection & Removal
Train-Test Split
Feature Scaling
5-Fold Time Series Cross-Validation
Model Training & Hyperparameter Setting
Test Set Evaluation
Explainability Analysis (SHAP values)
Made with Napkin

Justification of Model 1: Random Forest Regressor

Random Forest is an ensemble learning method that reduces variance by aggregating multiple decision trees trained using bootstrap sampling and random feature selection, while effectively modeling complex non-linear relationships (Breiman, 2001). It is well suited to this study due to the engineered meteorological and autoregressive features, which exhibit strong non-linear interactions typical of atmospheric processes. Random Forest captures feature interactions without requiring explicit functional assumptions or feature scaling, making it robust for structured tabular weather data (Hastie, Tibshirani and Friedman, 2009). Overfitting is controlled through bagging and tree constraints, and generalization is assessed using time-series cross-validation and learning-curve analysis.

Model Type: Ensemble, Non-parametric, Tree-based regression

Layers: 100 independent decision trees

Tree Structure:

- Maximum depth: 10 levels
- Minimum samples per leaf: 4
- Minimum samples to split: 10
- Bootstrap samples: 100% (with replacement)

Feature Selection: Each split evaluates random subset of features, promoting diversity among trees

Aggregation: Final prediction = mean of all 100 tree predictions

Activation Functions: N/A (decision trees use binary splits, not traditional activation)

Justification of Model 2: XGBoost Regressor

XGBoost (Extreme Gradient Boosting) is a regularized gradient-boosting framework designed for high predictive accuracy and computational efficiency on structured data. Unlike bagging-based ensembles, it builds trees sequentially, with each tree correcting residual errors while explicitly controlling model complexity through regularization (Chen and Guestrin, 2016). XGBoost is well suited to this study due to its ability to capture complex non-linear relationships, feature interactions, and temporal patterns arising from meteorological and autoregressive features. Gradient boosting reduces model bias by incrementally minimizing prediction error, while L1/L2 regularization and learning-rate shrinkage enhance generalization in correlated feature spaces (Hastie, Tibshirani and Friedman, 2009).

Model Type: Ensemble, Parametric (regularized), Gradient-boosted tree regression

Layers: 100 sequential trees (each tree "boosts" previous predictions)

Tree Structure:

- Maximum depth: 5 levels (shallower than RF for regularization)

- Subsample rate: 0.8 (80% of training samples per iteration)
- Column sample rate: 0.8 (80% of features per iteration)

DATA CLEANING AND PREPROCESSING

Missing Value Handling

Initial Assessment: All 87,696 observations contained complete values across all seven features. No missing data imputation was required.

Verification Steps:

- Element-wise null checks (df.isnull().sum())
- Confirmation of "no missing values" status
- No preprocessing bias introduced from imputation

Column Standardization

Operations Performed:

- Converted all column names to lowercase
- Replaced spaces and hyphens with underscores
- Final standardized names:
    - temperature, humidity, pressure, solar, dewpoint, cloudcover

Rationale: Consistent naming conventions reduce errors in downstream operations and improve code readability.

Duplicate Removal and Data Integrity

Checks Performed:

- Verification of temporal sequence integrity
- Confirmation that time column contains unique, sequential datetime values
- No duplicate timestamps identified
- Data sorted chronologically (2015-09-01 to 2025-09-01)

Result: Dataset integrity confirmed; no rows required removal for duplication.

Summary Statistics Post-Cleaning

After cleaning operations (before feature engineering):

- Rows retained: 87,696 (100%)
- Columns: 7 original + extracted temporal features
- Missing values: 0
- Data quality: Excellent

TIME SERIES CONVERSION AND FEATURE ENGINEERING

Temporal Features (Extracted from Datetime)

Feature	Description	Range
hour	Hour of a day	0-23
day_of_week	Day number (0=Monday, 6=Sunday)	0-6
day_of_year	Day of year	1-366
month	Month number	1-12
quarter	Quarter of year	1-4

Rationale: These features explicitly encode periodic patterns that predictive models can leverage.

Lag Features (Previous Hour Temperatures)

Feature	Description	Lookback Period
temp_lag_1	Temperature 1 hour prior	1 hour
temp_lag_2	Temperature 2 hour prior	2 hour
temp_lag_3	Temperature 3 hour prior	3 hour
temp_lag_6	Temperature 6 hour prior	6 hour
temp_lag_12	Temperature 12 hour prior	12 hour
temp_lag_24	Temperature 24 hour prior	24 hour

Rationale: Captures autocorrelation structure inherent in time series. Temperature exhibits high persistence previous values strongly predict current values.

Rolling Aggregate Features

Feature	Description	Window
temp_rolling_3h	Rolling mean of temperature	3 hour
temp_rolling_6h	Rolling mean of temperature	6 hour
temp_rolling_24h	Rolling mean of temperature	24 hours

Rationale: Smooth short-term trends and reduce noise while preserving underlying patterns.

Volatility and Change Features

Feature	Description	Calculation
emp_diff_1h	Temperature change in last hour	Current - Previous
temp_rolling_6h	6-hour rolling standard deviation	Rolling std(temperature)

Rationale: Capture rate of change and turbulence, which influence how rapidly temperature transitions occur.

Final Feature Set

Total Features: 21 engineered features

Feature Composition:

- Original continuous features: 5 (humidity, pressure, solar, dewpoint, cloudcover)
- Temporal features: 5
- Lag features: 6
- Rolling aggregates: 3
- Volatility/change features: 2

Dataset Shape After Engineering: 87,672 rows × 23 columns (24 with target) **Rows Removed**: 24 rows (due to lag feature NaN values from initial rows with insufficient history)

Outlier Detection and Treatment

Outlier Detection Method

High-resolution meteorological datasets often contain anomalous observations arising from sensor faults, extreme events, or reanalysis errors, which can bias model learning if left untreated. Outliers were identified using the Interquartile Range (IQR) method, a robust statistical approach suitable for non-normally distributed atmospheric data (Tukey, 1977; Wilks, 2011). Applying the IQR method to 87,672 samples detected 8,987 outliers (10.25%), primarily in highly variable predictors such as humidity, solar radiation, and short-term temperature change, while only 17 involved the target temperature. All outlier-affected rows were removed, yielding 78,685 samples, and the cleaned dataset was standardized using z-score normalization to enhance numerical stability and model training efficiency (Bishop, 2006; Chen and Guestrin, 2016).

Interquartile range = Upper Quartile – Lower Quartile = Q3 – Q1

Train-Test Split

Temporal Train–Test Split Methodology

- **Split Strategy:**
  A non-random, chronological temporal split was employed to preserve the sequential structure of the time-series data.

- **Rationale:**
  Random train–test splitting introduces data leakage, as it allows future observations to influence model training. This violates fundamental time-series assumptions and results in overly optimistic performance estimates.

- **Dataset Partitioning:**
  - **Training Set:**
    - 55,079 samples (70%)
    - Time period: September 2015 to mid 2022
  - **Test Set:**
    - 23,606 samples (30%)
    - Time period: mid 2022 to mid September 202
- **Temporal Integrity:**
  - Strict chronological ordering was maintained.

    - No overlap exists between training and testing periods.
    - Models were trained exclusively on historical data and evaluated on unseen future observations.
- **Validation Approach:**
    - TimeSeriesSplit (5 folds) was applied within the training set.
    - In each fold, validation data temporally follows the corresponding training data.
    - This approach closely simulates real-world forecasting scenarios and ensures leakage-free model evaluation.

Feature Scaling

Feature scaling was applied to both input variables and the target temperature to ensure numerical stability and comparable feature magnitudes during model training. Z-score normalization (StandardScaler) was used, with scalers fitted exclusively on the training data and then applied to the test set to prevent information leakage (Hastie, Tibshirani and Friedman, 2009). The target temperature was standardized using a separate scaler, and predictions were inverse-transformed to degrees Celsius prior to computing evaluation metrics such as MAE and RMSE, ensuring physical interpretability (Bishop, 2006).

Standardization improves optimization stability for gradient-based models and facilitates fair comparison across predictors measured on different scales. Although tree-based ensembles such as Random Forest and XGBoost do not require scaling for correctness, it remains beneficial for cross-model comparison and loss-stable target scaling. Outlier treatment using an IQR-based method was performed prior to scaling to limit the influence of extreme values on estimated distribution parameters (Chen and Guestrin, 2016; Wilks, 2011).

Scaling Statistics and Verification

Original Training Target:

- Mean: 14.70°C
- Standard Deviation: 5.20°C
- Range: -0.7°C to 27.7°C

After StandardScaler:

- Mean: 0.0000 (perfectly centered)
- Standard Deviation: 1.0000 (unit variance)
- Scaled range: -0.327 to 2.502 (approximately -0.33 to 2.50 in standard units)

Verification: The scaled target follows standard normal distribution N(0,1), confirming proper transformation.

Model Architecture

Random Forest Architecture

Model Type: Ensemble, Non-parametric, Tree-based regression

Layers: 100 independent decision trees

Tree Structure:

- Maximum depth: 10 levels
- Minimum samples per leaf: 4
- Minimum samples to split: 10
- Bootstrap samples: 100% (with replacement)

Feature Selection: Each split evaluates random subset of features, promoting diversity among trees

Aggregation: Final prediction = mean of all 100 tree predictions

Activation Functions: N/A (decision trees use binary splits, not traditional activation)

Optimizer: Greedy recursive binary splitting (Gini impurity minimization).

XGBoost Architecture

Model Type: Ensemble, Parametric (regularized), Gradient-boosted tree regression

Layers: 100 sequential trees (each tree "boosts" previous predictions)

Tree Structure:

- Maximum depth: 5 levels (shallower than RF for regularization)
- Subsample rate: 0.8 (80% of training samples per iteration)
- Column sample rate: 0.8 (80% of features per iteration)

Sequential Process:

1. Fit tree 1 to residuals of zero baseline
2. Fit tree 2 to residuals of (baseline + learning_rate × tree_1)
3. Fit tree 3 to residuals of (baseline + learning_rate × (tree_1 + tree_2))
4. Continue for 100 iterations

Learning Rate: 0.1 (step size controlling contribution of each tree).

Training Procedure

Cross-Validation Phase

Random Forest CV Results:

Training R^2: 0.9935 ± 0.0017

Training MAE: 0.2783 ± 0.0483°C

Training RMSE: 0.4004 ± 0.0679°C

Validation R^2: 0.9856 ± 0.0038

Validation MAE: 0.4185 ± 0.0295°C

Validation RMSE: 0.6175 ± 0.0541°C

Generalization Gap: 0.0079 (0.79%)

XGBoost CV Results:

- Training R^2: 0.9952 ± 0.0011
- Training MAE: 0.2495 ± 0.0348°C
- Training RMSE: 0.3455 ± 0.0500°C
- Validation R^2: 0.9904 ± 0.0040
- Validation MAE: 0.3436 ± 0.0385°C
- Validation RMSE: 0.4973 ± 0.0724°C
- Generalization Gap: 0.0048 (0.48%)

Interpretation: Both models show excellent generalization with minimal overfitting. XGBoost demonstrates tighter generalization gap (0.48% vs 0.79%), suggesting superior regularization and better generalization capability.

Learning Curves Analysis

Learning curves show how model performance improves with more training data.

Random Forest: Training MAE increases gradually to ~0.42°C while validation MAE remains stable, indicating consistent generalization with no overfitting. The model reaches near-optimal performance at ~25% of data.

XGBoost: Training MAE decreases to ~0.24°C while validation MAE improves sharply to ~0.34°C, showing superior data efficiency. XGBoost learns faster and continues improving with full datasets.

Key Finding: Both models demonstrate adequate data utilization. XGBoost's lower validation MAE and steeper learning slope confirm its superiority. No evidence of underfitting or data insufficiency observed.

Final Model Selection

XGBoost was selected as the final model based on its superior predictive performance and generalization capability compared to Random Forest. The model consistently achieved lower prediction errors while effectively capturing nonlinear relationships and temporal patterns present in the meteorological and autoregressive features. Its regularization mechanisms reduced overfitting, and its efficient training process allowed faster experimentation and tuning. Overall, XGBoost provided the best balance between accuracy, robustness, and computational efficiency, making it the most suitable model for this project.

EVALUATION METRICS AND RESULTS

Model performance was assessed using R^2, MAE, and RMSE, which measured variance and prediction error magnitude. On the test set, Random Forest achieved an R^2 of 0.9875, with an MAE of 0.423 °C and an RMSE of 0.596 °C, indicating strong predictive accuracy. XGBoost outperformed Random Forest, achieving an R^2 of 0.9915, a lower MAE of 0.353 °C, and an RMSE of 0.490 °C.

Overall, XGBoost was the superior model, providing higher explanatory power, lower error, and substantially faster training time (0.74 s vs. 37.1 s), making it the preferred approach for temperature prediction in this study.

Feature Importance

Random Forest: Relies 97% on temp_lag_1 (previous hour temperature), suggesting the model underutilizes meteorological information.

XGBoost: Distributes importance more evenly: temp_lag_1 (59%), temp_lag_24 (23%), temp_rolling_3h (12%), and other features (6%). This balanced approach better captures physical relationships and temporal patterns.

Residual Analysis

XGBoost residuals show:

- Mean ≈ 0 (unbiased predictions, no systematic over/underestimation)
- 95% of predictions within ±1.0°C of actual values
- Distribution approximately normal with slight tail deviation at extremes
- Homoscedastic variance (constant error across prediction range)
- No systematic patterns in residuals

CONCLUSION

This study successfully developed and evaluated operational machine learning models for hourly temperature prediction in Okhaldhunga, Nepal, using ensemble-based regression techniques. Among the evaluated models, XGBoost emerged as the optimal approach, demonstrating superior predictive accuracy, generalization capability, and computational efficiency. The final XGBoost model achieved an R^2 score of 0.9915, explaining 99.15% of the observed temperature variance, with a Mean Absolute Error (MAE) of 0.3526 °C and a Root Mean Squared Error (RMSE) of 0.4898 °C, indicating high-precision forecasting performance.

Feature importance and interpretability analyses revealed that lagged temperature features and daily seasonal components dominate model predictions, accounting for approximately 59% and 23% of explanatory influence, respectively. These findings confirm the strong autoregressive nature of temperature dynamics in mountainous environments. The model demonstrated excellent generalization to unseen data, as evidenced by a minimal cross-validation to test performance gap of 0.48%, indicating robust learning without overfitting. Furthermore, the XGBoost model achieved rapid training times (0.74 seconds), substantially outperforming comparable ensemble methods.

From an applied perspective, the proposed modeling framework is well-suited for operational temperature forecasting in high-altitude Himalayan regions. Its accuracy and computational efficiency enable real-time inference, while the incorporation of physically meaningful predictors such as solar radiation and dew point enhances interpretability and stakeholder confidence. Consequently, the model has strong potential to support agricultural planning, water resource management, and climate-sensitive decision-making in data-scarce mountainous settings.

Overall, the developed XGBoost model demonstrates a high level of technical robustness, interpretability, and operational readiness, making it suitable for deployment in real-world temperature forecasting applications in complex terrain environments.

Reflection on Individual Learning

This project provided comprehensive exposure to end-to-end machine learning pipeline development:

Technical Competencies Developed:

1. **Data Engineering**: Handling large datasets (87K+ observations), temporal formatting, quality assessment
2. **Exploratory Analysis**: Identifying patterns, correlations, and relationships driving predictions
3. **Feature Engineering**: Creating domain-informed features (lag, rolling, temporal) from raw data
4. **Model Selection**: Comparing algorithms theoretically and empirically, understanding trade-offs
5. **Evaluation Methodology**: Implementing proper time series validation, avoiding common pitfalls
6. **Model Explainability**: Interpreting black-box predictions using SHAP values and feature importance

Conceptual Insights:

1. **Temporal Data Requires Temporal Validation**: Random splitting invalidates time series analysis; chronological splitting is essential
2. **Ensemble Methods Excel with Multiple Features**: XGBoost's gradient boosting leverages diverse features better than Random Forest's bagging
3. **Feature Engineering Dominates Accuracy**: 21 engineered features enable capturing temporal patterns (lag, rolling) crucial for time series
4. **Simple Features Often Work Best**: Previous hour temperature (lag_1) is 59% of predictive power; complex features matter but less
5. **High Performance Requires Proper Data Preparation**: Outlier removal, scaling, and validation methodology are as important as algorithm choice
6. **Interpretability Aids Trust**: SHAP values and feature importance make results actionable, not just accurate

Professional Skills Gained:

- Python proficiency (pandas, sklearn, XGBoost, SHAP libraries)
- Scientific documentation and reproducible research practices
- Statistical reasoning (understanding metrics, distributions, generalization)
- Communication of complex results to technical and non-technical audiences

FUTURE RECOMMENDATIONS AND PROJECT LIMITATIONS

Project Limitations

This study is subject to several data, modeling, and validation limitations. The Open-Meteo dataset is derived from global atmospheric reanalysis models rather than direct ground observations, and its accuracy may vary across regions, particularly in complex mountainous terrain such as Okhaldhunga, where coarse model resolution can smooth extremes and fail to capture local microclimatic effects. The analysis is further limited by its single-location scope, restricting generalizability to other Himalayan regions with differing topography and land–atmosphere interactions. Although the approximately ten-year temporal coverage captures seasonal variability, it remains insufficient for robust climate trend analysis, for which multi-decade records are typically required. Model performance is strongly dependent on autoregressive features, limiting multi-step forecasting capability and introducing error accumulation. Additionally, static predictors and linear scaling assumptions constrain physical realism. Finally, validation relied on a single forward test period without uncertainty quantification or extensive hyperparameter optimization, limiting robustness under extreme or unseen climatic conditions.

10.2 Future Improvements and Recommendations

Data Enhancement:
Future work should integrate ground-based weather station observations from Okhaldhunga or nearby regions to validate and correct Open-Meteo reanalysis data. Station records of at least one year would enable bias assessment and hybrid modeling using reanalysis variables as predictors. Expanding the spatial scope to multiple Himalayan locations across different elevations would improve generalizability and allow analysis of topographic effects. Incorporating large-scale climate indices such as the Oceanic Niño Index, Indian Monsoon Rainfall Index, and North Atlantic Oscillation would further enable modeling of seasonal teleconnections and long-range climatic influences.

Model Architecture Improvements:
Advanced models such as Long Short-Term Memory networks can better capture long-range temporal dependencies and reduce reliance on manual lag selection (Hochreiter and Schmidhuber, 1997). Hybrid ensembles combining ARIMA with XGBoost may improve accuracy through residual learning (Breiman, 1996), while quantile regression would provide probabilistic forecasts and uncertainty estimates (Koenker and Bassett, 1978).

Multi-Horizon Forecasting:
Extending the framework to multi-step forecasting (e.g., 24-hour or 7-day horizons) using direct or sequence-to-sequence approaches would mitigate error accumulation. Ensemble forecasting across temporal subsets could further enhance robustness and stability.