Deliverable 2: Modeling & Evaluation Report

Predicting Dengue Fever Cases Using Environmental and Climate Data

Peidi Dong (peidid) & Mohammad Khan (mhk2)

1. Introduction

This report builds directly on the Data Analysis Report (Deliverable 1), in which we explored the DengAI dataset covering weekly dengue case counts and environmental measurements for San Juan, Puerto Rico (1990–2008) and Iquitos, Peru (2000–2010). Following the cleaning, feature inspection, and exploratory analysis completed in that phase, the goal of this deliverable is to train, evaluate, and compare several regression models that predict weekly dengue case counts from climate and environmental variables.

The focus of this report is not only on predictive performance, but also on justifying modelling decisions, avoiding data leakage in a time-series setting, and interpreting model behaviour and errors in a meaningful way. Four regression models were trained alongside a simple baseline, evaluated using time-aware cross-validation and a held-out test set, and compared using RMSE, MAE, and R^2.

2. Data Preparation for Training

2.1 Feature Engineering

Dengue transmission is influenced not only by the current week's environmental conditions, but also by seasonal cycles and delayed biological effects. In Deliverable 1 we noted that mosquito development takes roughly one to two weeks, and virus incubation adds a further four to seven days. Therefore, the conditions driving cases in a given week likely occurred several weeks earlier. To make this temporal structure visible to the models, three families of features were engineered.

Cyclical encoding of week of year

The weekofyear variable is inherently cyclical: week 52 and week 1 are neighbouring weeks in time, but their raw numeric values make them appear maximally far apart. To represent this properly, weekofyear was transformed into two features, $\text{week\_sin} = \sin(2\pi \cdot w / 52)$ and $\text{week\_cos} = \cos(2\pi \cdot w / 52)$. This gives the model a smooth, continuous representation of seasonality in which adjacent weeks remain close regardless of where they fall in the calendar.

Lag features

To reflect delayed effects, 1-week, 2-week, and 3-week lags were created, within each city, for the target variable (total_cases) and for three key environmental variables identified during EDA:

reanalysis_avg_temp_k, reanalysis_specific_humidity_g_per_kg, and precipitation_amt_mm. Grouping the shift operation by city ensures that lag values from Iquitos never leak into San Juan's history.

Rolling averages

Weekly measurements can be noisy. For the same three climate variables, a 3-week rolling mean was computed using only prior weeks within each city (the window is shifted by one before rolling, so the current week is excluded). This captures the broader environmental state rather than single-week spikes and, critically, avoids leakage from the current observation into its own feature.

Handling imbalanced data

This is a regression task, so classification-oriented rebalancing techniques such as SMOTE, oversampling, and undersampling do not directly apply. However, the target distribution is strongly right-skewed: most weeks have low case counts and a small number of weeks record very high outbreak levels. These high-case observations were there because they represent genuine outbreak events rather than noise; so removing them would remove exactly the pattern the project is designed to study. Instead, the skew is addressed at evaluation time through the use of multiple metrics (RMSE, MAE, R^2) and a detailed error analysis by case level.

Resulting dataset

Lag and rolling features naturally produce missing values in the first few weeks of each city's series. These rows were removed before modelling, reducing the dataset from 1,456 rows to 1,450 rows with 42 columns in total (25 original plus 17 engineered).

Encoding Categorical Variables

The dataset includes one categorical variable, city, which indicates whether the observation belongs to San Juan or Iquitos.

This variable is encoded using one-hot encoding, allowing the model to learn separate effects for each city without introducing an artificial ordinal relationship between categories.

This approach is appropriate because the EDA showed that the two cities exhibit different dengue patterns. Encoding them as separate features enables the model to capture these city-specific differences effectively.

2.2 Data Splitting Strategy

Because the target depends on time-ordered environmental inputs, a random train/test split would allow future observations to influence model training through lagged features and rolling windows. To avoid this, a chronological split was used: all observations from 1990–2007 form the training set (1,303

rows) and observations from 2008–2010 form the held-out test set (147 rows). This mirrors how the model would be used in practice (trained on history, evaluated on the future).

This yields roughly a 90/10 split. Stratified splitting was not used because this is a regression problem with time-dependent data. Preserving temporal order is more important than maintaining a balanced distribution across splits.

A larger test share was not used because it would have required cutting further into the seasonal cycles present in the training data and reducing the model's exposure to full years of both cities. The test set was kept untouched until final evaluation; all hyperparameter selection was performed via cross-validation on the training data only. Reproducibility is ensured by fixing random_state=42 on all stochastic estimators (Decision Tree, Random Forest) and by the deterministic nature of the chronological split itself.

One consequence of this split is worth flagging: because San Juan's record ends in 2008 while Iquitos continues through 2010, the test set is predominantly Iquitos (approximately 130 IQ weeks vs. 17 SJ weeks). This will be revisited in the error analysis, where per-city performance is reported separately.

2.3 Feature Scaling

All numeric features were standardised using z-score scaling (StandardScaler), and the categorical city variable was one-hot encoded. Both transformations were wrapped in a ColumnTransformer inside a scikit-learn Pipeline, which means the scaler is fit on the training fold only during cross-validation and on the full training set before final evaluation. the test set never contributes to the scaling parameters. This prevents the common data-leakage error of scaling on the combined dataset before splitting.

Standardisation matters for Ridge Regression and for SVR with an RBF kernel, both of which are sensitive to feature magnitude: Ridge's L2 penalty would disproportionately shrink small-scale features, and SVR's distance-based kernel assumes comparable feature ranges. Tree-based models (Decision Tree, Random Forest) are invariant to monotonic rescaling of features, so scaling has no effect on them but is harmless inside the shared pipeline.

3. Baseline Model

Before any complex model was fitted, a simple baseline was established to serve as a reference point. The baseline is a mean predictor (scikit-learn's DummyRegressor with strategy="mean"), which always predicts the mean dengue case count observed in the training set, regardless of input features. A useful model must clearly outperform this baseline across RMSE, MAE, and R^2; anything less would indicate that the features and modelling effort are not contributing real predictive signals.

On the held-out test set, the baseline produced RMSE = 20.44, MAE = 19.20, and $R^2 = -1.91$. The strongly negative R^2 reflects the fact that the training mean is pulled upward by San Juan's large outbreak weeks, while the test set is dominated by Iquitos, which has much lower typical case counts. So predicting the

training mean is actively worse than predicting the test-set mean. In other words, the baseline is off by roughly 19–20 cases per week on average and fails to explain any variance in the test data. Every subsequent model is therefore required to clear a meaningful bar.

4. Model Training, Evaluation, and Justification

Four regression models were trained, each chosen for a specific reason. All models shared the same preprocessing pipeline, the same feature set, and the same time-aware cross-validation scheme so that comparisons would be fair.

For evaluation, three regression metrics were used: RMSE, MAE, and R^2. RMSE is useful because it penalizes larger errors more heavily, which is important in the presence of outbreak spikes. MAE measures the average absolute prediction error and is easier to interpret in case-count units. R^2 shows how much of the variance in dengue cases is explained by the model. Together, these metrics provide a balanced view of typical error, sensitivity to large mistakes, and overall explanatory power.

Cross-validation setup

Because the data is time-ordered, standard k-fold cross-validation is not appropriate, as it breaks the temporal ordering of observations and may allow the model to train on future weeks when predicting past ones, leading to temporal leakage. Instead, scikit-learn's TimeSeriesSplit with five folds was used. This builds expanding training windows where each fold trains on earlier observations and validates on the chronologically next block, exactly matching the structure of the final train/test split. Hyperparameters were tuned with GridSearchCV using negative RMSE as the scoring metric.

To ensure robust and fair evaluation, all models were trained on the same features, used the same preprocessing pipeline, and were tuned using the same time-aware cross-validation framework. Final performance was then assessed on the same untouched hold-out test set.

4.1 Ridge Regression

Justification. Ridge Regression was the first model trained, because the task is regression on a continuous target; because Deliverable 1's correlation heatmap revealed substantial multicollinearity among temperature variables and among humidity variables; and because the engineered lag and rolling features add further correlated predictors. Ordinary least squares is unstable under multicollinearity, whereas Ridge's L2 penalty shrinks correlated coefficients toward each other and improves generalisation. Its linear form also offers interpretability, which matters for communicating findings to public health stakeholders.

Assumptions. Ridge assumes an approximately linear relationship between predictors and the target after regularisation. This assumption is clearly not fully satisfied (as dengue dynamics are non-linear) but the engineered features (lags, rolling means, cyclical seasonal encoding) bring the problem much closer to a regime where a penalised linear model can perform well.

Hyperparameter tuning. The regularisation strength α was searched over {0.01, 0.1, 1.0, 10.0, 100.0} using 5-fold TimeSeriesSplit. The best value was α = 10.0, indicating that noticeable shrinkage improved validation performance.

Test-set performance. RMSE = 6.89, MAE = 4.09, R^2 = 0.67. This is a substantial improvement over the baseline (which had RMSE > 20 and negative R^2). An MAE of roughly 4 cases means that on a typical week the prediction is within 4 cases of the truth, and an R^2 of 0.67 means the model accounts for about two-thirds of the variance in dengue counts on the test set. The key caveat, explored in Section 6, is that this average performance masks larger errors during outbreak weeks.

4.2 Decision Tree Regressor

Justification. A Decision Tree was included as a non-linear, non-parametric counterpart to Ridge. Unlike a linear model, a tree can capture threshold effects. For example, dengue activity changes sharply once humidity or prior case counts cross certain levels. It could also capture interactions among features without being specified in advance. Decision trees are also highly interpretable due to their hierarchical split structure.

Assumptions. Trees make very few assumptions about the functional form. The main risk is overfitting, which grows with depth and is mitigated by restricting tree size.

Hyperparameter tuning. A grid over max_depth {3, 5, 7, 10, None}, min_samples_split {2, 5, 10, 20}, and min_samples_leaf {1, 2, 5, 10} was searched via 5-fold TimeSeriesSplit. The best configuration was max_depth = 5, min_samples_split = 2, min_samples_leaf = 5, which keeps the tree relatively shallow.

Test-set performance. RMSE = 7.51, MAE = 4.68, R^2 = 0.61. The tree clearly beats the baseline and confirms that non-linear structure is learnable, but it is slightly worse than Ridge on every metric. A plausible explanation is that once seasonal cyclical features and smooth lag/rolling features are available, the underlying signal is largely smooth, and sharp binary splits are a clumsier fit than a penalised linear combination.

4.3 Random Forest Regressor

Justification. A Random Forest was included to address the two main weaknesses of a single decision tree: high variance and sensitivity to specific splits. By averaging many trees trained on bootstrapped samples with feature randomisation, a forest typically produces smoother, more stable predictions and captures interactions without the overfitting risk of a deep single tree.

Hyperparameter tuning. The grid covered n_estimators {100, 200}, max_depth {5, 10, None}, and min_samples_leaf {1, 2, 5}, evaluated with 5-fold TimeSeriesSplit. The best configuration was n_estimators = 200, max_depth = None, min_samples_leaf = 1.

Test-set performance. RMSE = 7.60, MAE = 5.01, R^2 = 0.60. The forest performs very similarly to the single tree and, perhaps surprisingly, slightly worse than Ridge. The most likely reason is that the feature

engineering has already captured the dominant non-linear structure in a form that a linear model can exploit; the additional flexibility of the ensemble is not earning its keep here, and tree-based methods also struggle to extrapolate beyond observed ranges. This is relevant because the test period contains weeks that may fall outside the training distribution.

4.4 Support Vector Regressor

Justification. An SVR with an RBF kernel was included as a fundamentally different non-linear model. Unlike a tree, which partitions the feature space with axis-aligned splits, SVR learns a smooth non-linear decision surface and uses only a subset of training points (support vectors) to define it. This makes it a good stress test for whether gradual non-linear relationships, rather than threshold effects, drive dengue counts.

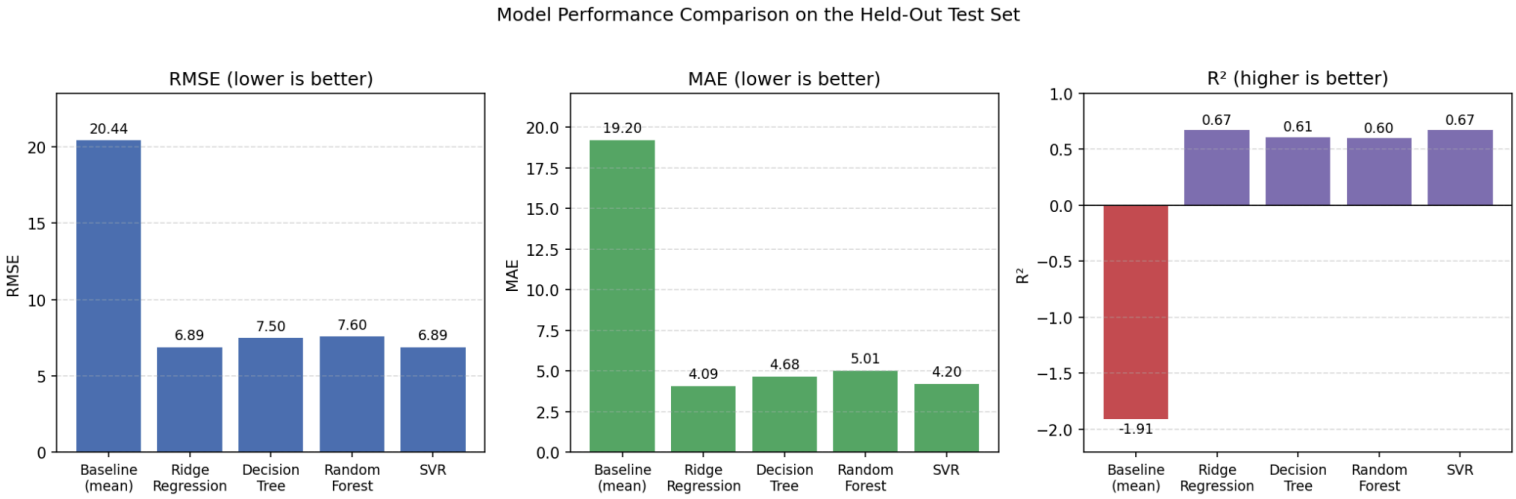
Assumptions. SVR assumes features are on comparable scales, which is ensured by the standardisation step in the pipeline. It also assumes the target surface is reasonably smooth, which matches the smoothing induced by lags and rolling means.

Hyperparameter tuning. A grid over C {0.1, 1, 10, 100}, epsilon {0.1, 0.5, 1.0}, and gamma {'scale', 0.01, 0.1, 1} was searched with 5-fold TimeSeriesSplit. The best configuration was C = 100, epsilon = 1.0, gamma = 0.01.

Test-set performance. RMSE = 6.89, MAE = 4.20, R^2 = 0.67. SVR essentially ties Ridge on RMSE and R^2, with a very slightly higher MAE. The near-identical performance of a penalised linear model and a flexible kernel model is informative: it suggests that after feature engineering, the signal that remains is close to linear in the engineered representation, and additional modelling capacity does not translate into improved predictive performance .

5. Model Comparison

Table 1 summarises the performance of all five models on the held-out test set. All four trained models outperform the baseline by a wide margin, confirming that the engineered features carry real predictive signal. Among the trained models, Ridge Regression and SVR are nearly indistinguishable and are both clearly ahead of the two tree-based models.



[Figure 1: Bar chart comparing RMSE, MAE, and R^2 across all five models]

Model	RMSE	MAE	R^2
Baseline (mean predictor)	20.44	19.20	−1.91
Ridge Regression	6.89	4.09	0.670
Decision Tree Regressor	7.51	4.68	0.608
Random Forest Regressor	7.60	5.01	0.598
Support Vector Regressor	6.89	4.20	0.670

Table 1. Test-set performance across all models.

5.1 Trade-offs between models

- **Ridge Regression** — strong predictive performance with the lowest MAE, coefficients that are directly interpretable after standardisation, and the lowest computational cost. The main limitation is its linear functional form.
- **Support Vector Regressor** — matches Ridge on RMSE and R^2, but is less interpretable (predictions depend on a kernel expansion over support vectors) and more expensive to train and tune.
- **Decision Tree** — easiest to inspect visually and captures threshold effects, but less accurate here and prone to instability.
- **Random Forest** — more stable than a single tree, but adds complexity and training cost without improving on Ridge in this dataset.

5.2 Final model selection

Ridge Regression is selected as the final model. Its RMSE and R^2 are within 0.01 of SVR's, and its MAE is the best of all models. Given equivalent predictive performance, Ridge is preferred because it is

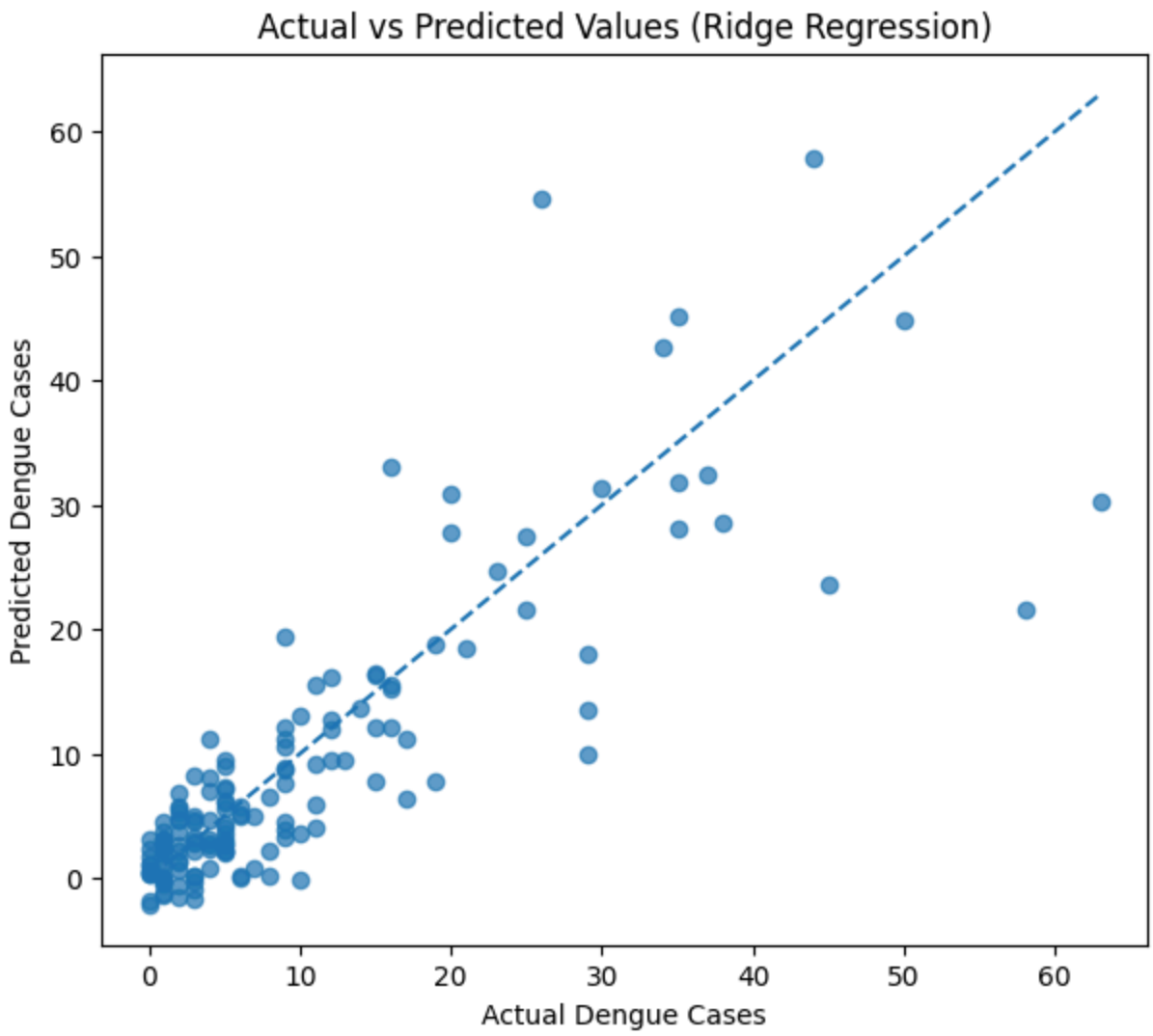
substantially more interpretable, substantially cheaper to train and retrain, and clearer to justify to non-technical stakeholders, all of which matter for the public-health use case motivating the project. This choice aligns with the guidance that the best model is not always the most flexible one, but the one whose accuracy, simplicity, and interpretability are best matched to the problem.

6. Error Analysis

Aggregate metrics hide where a model fails. This section examines the Ridge Regression predictions on the test set along four dimensions: overall fit, residual structure, performance by outbreak intensity, and feature influence.

6.1 Actual vs. predicted values

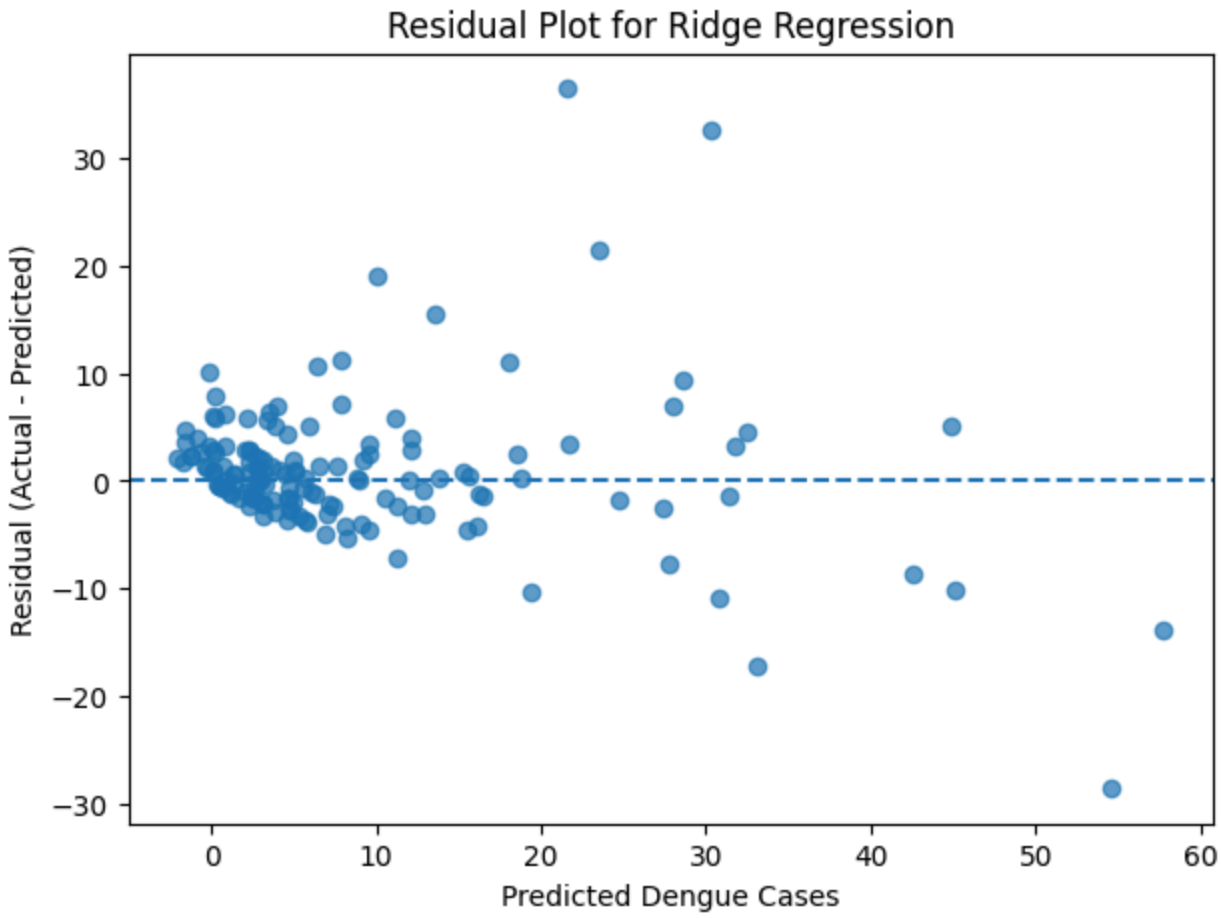
For weeks with low to moderate case counts the predictions cluster tightly around the diagonal, indicating that the model tracks typical dengue activity well. For higher case counts (roughly 40 cases and above) the scatter widens markedly and points fall below the diagonal, meaning the model systematically under-predicts the magnitude of outbreaks. This is an important limitation for any public-health use: the model is most useful for monitoring steady-state activity and least useful precisely when accuracy matters most.



[Figure 2: Ridge Regression predictions vs. actual dengue case counts on the test set.]

6.2 Residual analysis

Residuals are centred near zero for low predicted values but fan out as predicted values increase. This heteroscedasticity, which is variance that grows with the predicted mean, is a classical warning sign that the linear-Gaussian assumption underlying Ridge is not fully appropriate for count data with a long right tail. It also confirms that confidence in predictions should vary by regime: tight around quiet weeks, much looser around likely outbreaks.



[Figure 3: Residuals (actual − predicted) plotted against predicted dengue cases]

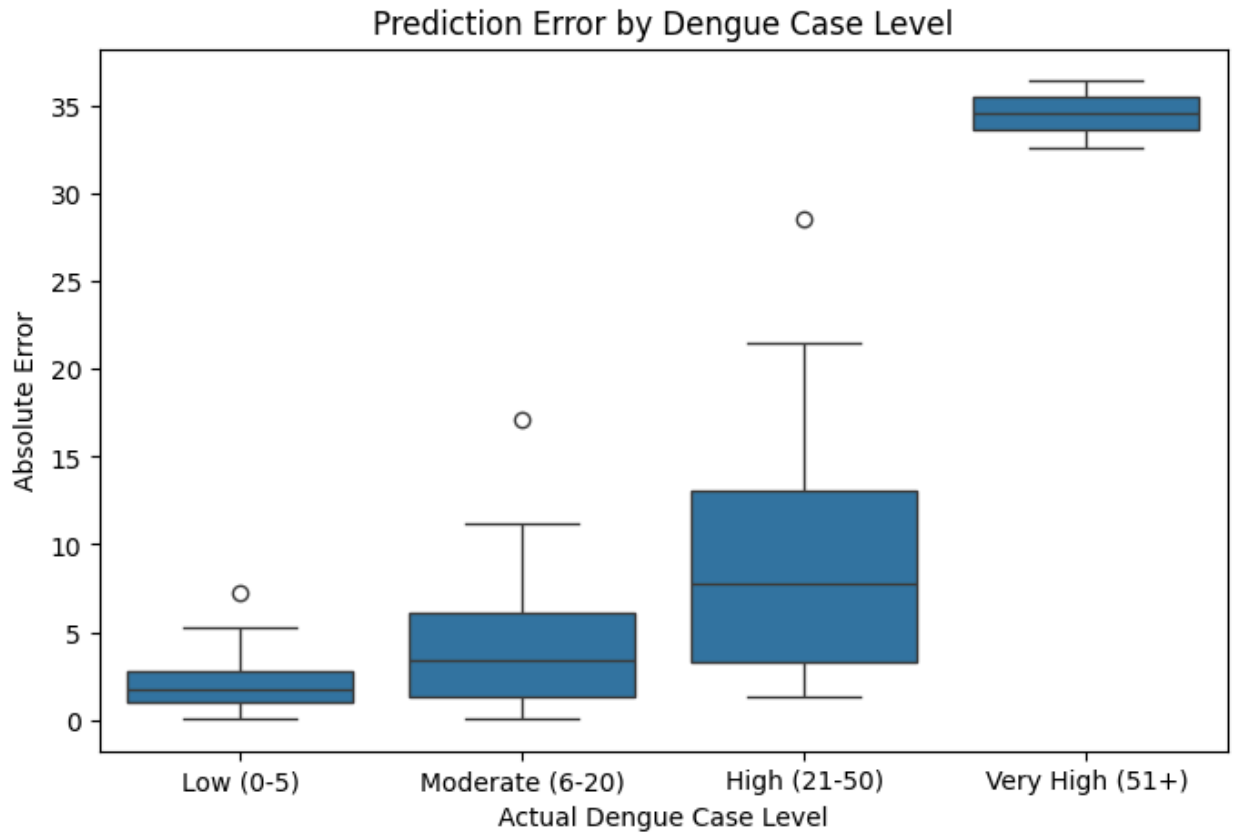
6.3 Largest prediction errors

The ten largest absolute errors on the test set are concentrated almost entirely in Iquitos during 2008. Several exceed 30 cases, and the pattern alternates between under-prediction during rapid rises and over-prediction immediately after peaks, consistent with a model that leans heavily on recent-week lags and therefore reacts to changes. When dengue activity shifts sharply from week to week, the lag-based signal arrives too late for a calibrated prediction.

6.4 Error by case level

Grouping the test set into four case-level bands makes the pattern unambiguous. In the Low band (0–5 cases) the mean absolute error is about 2 cases; in Moderate (6–20) it rises to roughly 4.3; in High (21–50) it climbs further; and in the Very High band (51+) the mean absolute error is roughly 34.6 cases. The model is therefore well-calibrated for background activity and progressively worse as outbreaks

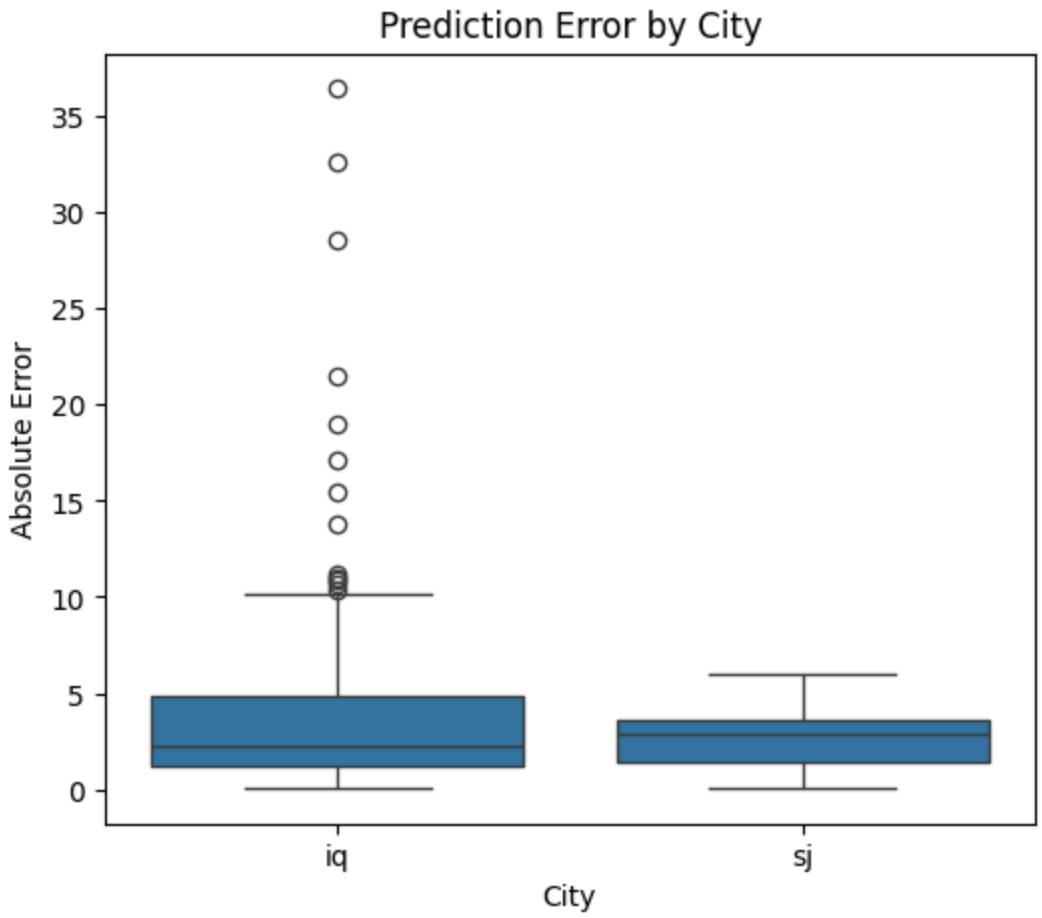
intensify. Because outbreak weeks are comparatively rare, these large errors barely move the overall RMSE and MAE, which is exactly why aggregate metrics alone would be misleading.



[Figure 4: Absolute error grouped by actual case level (Low 0–5, Moderate 6–20, High 21–50, Very High 51+)]

6.5 Error by city

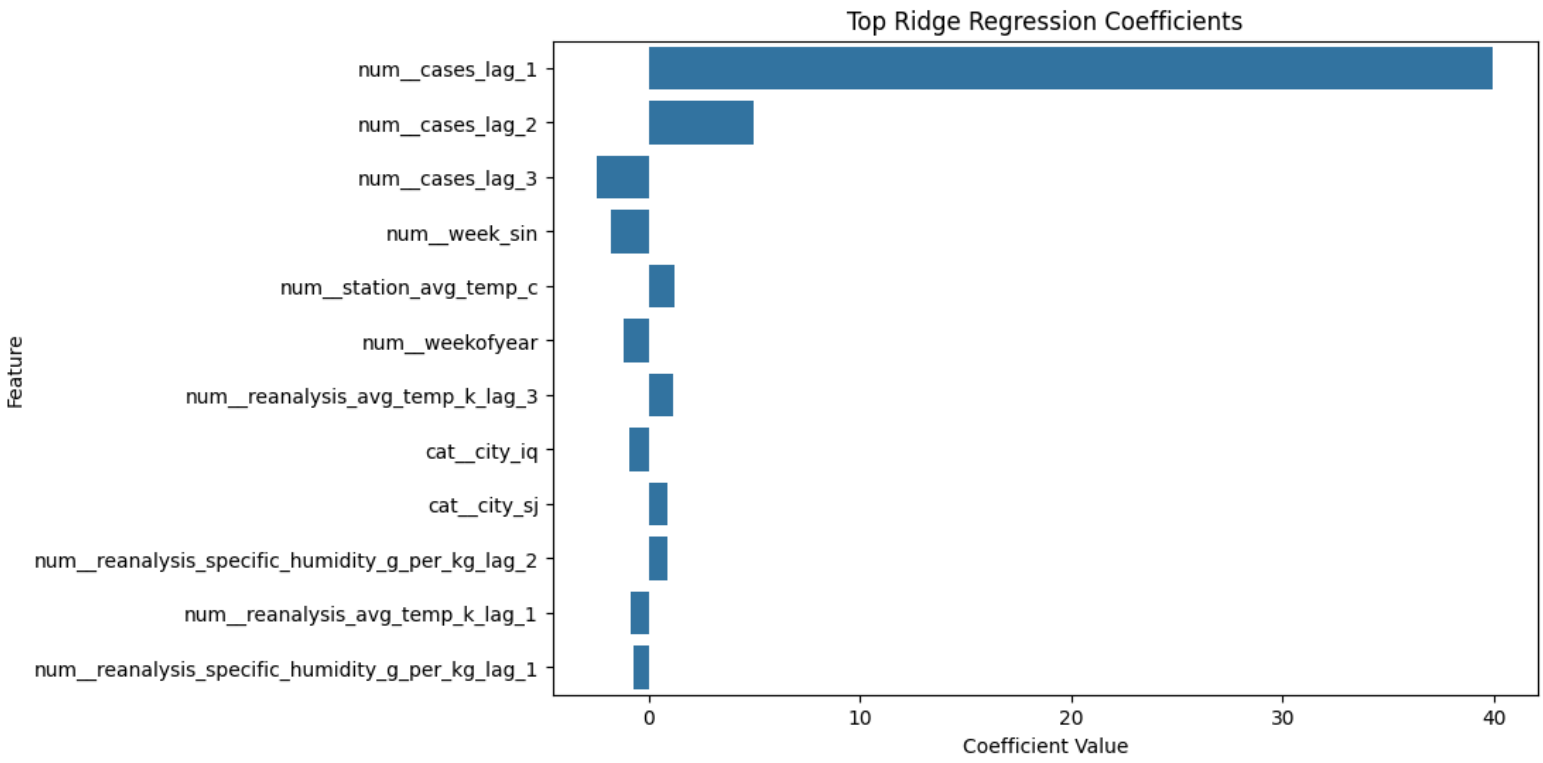
Errors in San Juan are tightly clustered and consistently small, whereas Iquitos shows greater variability and all of the most extreme errors. Two factors likely contribute. First, Iquitos accounts for the large majority of test observations (approximately 130 of 147), so it carries almost all of the risk of an extreme week surfacing in the test window. Second, the exploratory analysis in Deliverable 1 showed that Iquitos has a shorter history, a different seasonal profile, and greater dependence on humidity variables — all of which a single pooled model handles less cleanly than a city-specific model would.



[Figure 5: Boxplot of absolute error by city (Iquitos vs. San Juan)]

6.6 Feature influence

Because Ridge is linear and features are standardised, coefficient magnitudes are directly comparable. The dominant feature by a wide margin is cases_lag_1 (coefficient ≈ 40), an order of magnitude larger than any other. cases_lag_2 and cases_lag_3 are the next most influential, followed by seasonal features (week_sin, weekofyear), temperature lags, and humidity lags. The practical implication is that the model is, to a large extent, a persistence forecaster: its strongest signal is "next week will look a lot like this week". This explains both its strong average accuracy (dengue activity is autocorrelated week to week) and its failure mode around outbreaks (persistence cannot anticipate a regime change). It also moderates one of the project's original research questions: environmental variables do contribute, but their marginal contribution on top of recent case history is smaller than we expected at the start of modelling.



[Figure 6: Top Ridge Regression coefficients, ranked by absolute magnitude]

Overall, the error analysis shows that while the model performs well under stable conditions, it struggles with rapid changes, highlighting the need for features or models that can anticipate regime shifts rather than react to them.

7. Improvements and Next Steps

7.1 Improving predictions during outbreak periods

The clearest weakness is performance during outbreak weeks, driven by the model's reliance on recent case counts. Several targeted feature additions could help: week-to-week rate-of-change features in both case counts and environmental variables; rolling variance as an instability indicator; threshold-crossing flags (e.g., a binary feature that fires once specific humidity sustains a value known to favour transmission); and interaction features between temperature and humidity lags, since the biological literature suggests these act jointly rather than independently.

7.2 More advanced modelling approaches

Models explicitly designed for time-series could better handle the regime-change behaviour seen here. Options include classical approaches such as SARIMA or SARIMAX (which would use the climate variables as exogenous regressors), gradient-boosted trees with tuned monotonic constraints, and a recurrent

model with a larger training window such as an LSTM. Any of these should be evaluated against the current Ridge baseline with the same time-aware cross-validation protocol to ensure fair comparison.

7.3 Feature-engineering refinements

The current lag window of three weeks is short relative to the biological lag of two to three weeks plus additional reporting delay. Extending lags to four–eight weeks, experimenting with longer rolling windows (6- and 12-week) and richer rolling statistics (variance, min, max, trend slope), and testing Box–Cox or log(1+x) transformations on the skewed target are all reasonable next steps. A small ablation that removes cases_lag_* would also clarify how much the environmental variables contribute on their own, which may be a useful answer to the project's underlying research question.

7.4 City-specific modelling

Error analysis shows that Iquitos is the harder city. Deliverable 1 already documented that the two cities have different seasonal shapes and different dominant correlates (humidity-led in Iquitos, week-of-year-led in San Juan). Training two separate models (one per city) would let each learn its own coefficient profile and would likely reduce the large Iquitos errors. A compromise between a single pooled model and two entirely separate ones would be to include city-by-feature interaction terms.

7.5 Deployment considerations

In a realistic deployment the model would produce a weekly case-count estimate and a simple trend signal for public-health staff. Given the measured underestimation during outbreaks, it should not be presented as a standalone early-warning system. A more honest framing is that it supports trend monitoring and should be combined with surveillance data and expert judgement, with a conservative interpretation of predictions in weeks where recent activity has been rising or environmental conditions are favourable.

7.6 Monitoring and maintenance

If deployed, the model's RMSE and MAE in recent weeks should be tracked continuously, together with city-stratified versions of the same metrics. A sustained rise in error, especially in one city, is the clearest signal of model drift, which can plausibly be driven by climate variation, vector-control programmes, or changes in reporting practice. A retraining cadence of at least once per dengue season, with additional triggered retrains if drift metrics breach a threshold, would be a sensible starting policy.

7.7 Stronger Baseline (Persistence Model)

A stronger baseline for time-series data would be a persistence model, where the prediction is simply the number of dengue cases from the previous week. Given that the current model relies heavily on lag features (particularly recent case counts), such a baseline would provide a more meaningful benchmark.

Comparing against this would help quantify how much additional value the model provides beyond simple temporal persistence.

7.8 Evaluating Dependence on Lag Features

Since the model relies heavily on recent case counts, future work could evaluate performance without lag features to better understand the independent contribution of environmental variables. This would help determine whether climate and seasonal features alone provide meaningful predictive power, or whether most of the signal comes from short-term temporal persistence.

8. Conclusion

This deliverable trained and compared four regression models (Ridge Regression, Decision Tree, Random Forest, and Support Vector Regressor) against a mean-predictor baseline on the DengAI dataset, using engineered features that capture seasonality (cyclical week encoding), biological lag (1–3 week lags on cases, temperature, humidity, and precipitation), and short-term smoothing (3-week rolling means). Data leakage was avoided through a strictly chronological train/test split and TimeSeriesSplit cross-validation for hyperparameter tuning.

Ridge Regression was selected as the final model. Its test-set performance (RMSE = 6.89, MAE = 4.09, R^2 = 0.67) is statistically indistinguishable from SVR's on RMSE and R^2, better on MAE, and substantially more interpretable and cheaper to train. Both clearly outperform the tree-based models (RMSE ≈ 7.5–7.6) and the baseline (RMSE = 20.44).

These numbers are sufficient for weekly trend monitoring in a public-health support role but are not sufficient to predict outbreak peaks accurately. The error analysis made this gap precise: mean absolute error grows from about 2 cases in quiet weeks to roughly 34.6 cases in very-high outbreak weeks, and the largest errors are concentrated in Iquitos 2008 during rapid transitions. Coefficient inspection showed that cases_lag_1 dominates the model (coefficient ≈ 40, an order of magnitude above any other feature), which explains both the model's strong average accuracy and its weakness around regime changes. In this case, persistence-style forecasters are fundamentally reactive.

The most important next step, given these findings, is to address outbreak prediction directly: either by training separate city-specific models (Iquitos's dynamics are not well served by a pooled fit), or by adding features designed to signal impending regime change (rate-of-change indicators, longer lags, and temperature-humidity interaction terms) ideally in combination with an explicitly time-series-aware model such as SARIMAX. The current Ridge model provides a well-understood, well-validated baseline against which any of these extensions should be measured.