VBD-ModelBench: An evaluation method to benchmark predictive models for vector-borne diseases.

Adish Illikal[1], Prerna[1], Siva Athreya[2]

[1]ARTPARK, Bengaluru, Karnataka, India
[2]International Centre for Theoretical Sciences (ICTS), Bengaluru, Karnataka, India

July 2026

This document talks about evaluation methodology used to benchmark forecasting models for vector borne diseases. The emphasis is not only on whether a model minimises numeric error, but also on whether its forecasts remain useful for public-health decisions and enable interventions like vector control and resource allocation activities.

For each model, forecasts are assessed over different evaluation periods, lead times, and spatial units using a common set of prediction outputs. Results are then stratified to analyse performance differences that aggregate scores might not reveal, for example differences across regions, seasonal periods, case-load regimes, or forecast horizons. The main sections describe the recursive forecast generation setup,the operationally relevant aspects of the evaluation method in addition to standard error-based evaluation.

1 Recursive Forecast

In this section we describe various metrics that we shall use to evaluate the predictive models (post training and tuning). We assume that each model will be able to predict for a certain forecast horizon H and will require observed data from a lagging window L_z for dynamic features, (say case data and weather data). The models can aslo have static features that do not evolve over time.

We first set up notations for the input features (dynamic and static) and parameters in our models.

Variable	Description	Remarks
z	Absolute time index	$z \in \{0, \ldots, N\}$
O_z	Observed cases at time z	
P_z	Predicted cases at time z	
W_z	Dynamic Features like weather variables at time z	
I_z	Static Features	
H	Forecast horizon (number of total time steps predicted per run)	
h	Lead time (step within the forecast horizon)	$h \in \{0, \ldots, H\}$
s	Sliding window shift	$s \leq H$
L_x	Lag value for any feature x	
t_k	Origin of the $k-th$ forecast window	
K	Total number of possible sliding windows	
k	Index of the sliding window	$k \in \{0, \ldots, K-1\}$

Table 1: Notations describing dynamic and static variables.

Sliding Window We use a sliding window approach where the forecast window of length H (and hence the origin t_k) shifts by time step s:

$$t_k = t_0 + k.s$$

where, $t_0 = L$ and, $k = 0 : K - 1$.

The total number of possible windows is K:

$$K = \frac{T - L - H}{s} + 1 \tag{1}$$

For window k, at origin t_k, we will assume that the predicted values at $t_k + h$ for $h = 0, 1, 2, \ldots H-1$ are given by model f as follows

$$P^{(t_k)}_{t_k+h} = f\Big(Y^{(t_k)}_{t_k+h-1},\ Y^{(t_k)}_{t_k+h-2},\ \ldots, Y^{(t_k)}_{t_k+h-L}, W_{t_k-1}, I_z\Big) \tag{2}$$

where $Y^{(t_k)}_{t_k+h-1}$, are the lagged cases which are derived in the following way:

Lagged cases: For window k, at origin t_k, to predict lead time h, the model will require L lagged case values ending at $t_k + h - 1$. Fix a forecast horizon H with lag L, and sliding window $0 \leq k \leq K$. For predicting the case at origin t_k we use the

$$Y^{k,0}_z = O_z \text{ if } t_k - L + h \leq z \leq t_k - 1 \tag{3}$$

as the L- lagged case values. Suppose we wish to predict the cases on t_{k+h} for some $h = 1, 2, \ldots, H-1$ we will inductively on h use the following as L lagged case values

$$Y^{k,h}_z = \begin{cases} O_z & \text{if } t_k - L + h \leq z \leq t_k - 1 \\ P^k_z & \text{if } t_k \leq z \leq t_k + h - 1, \end{cases} \tag{4}$$

to predict $t_k + h$, where O_z are the reported cases and P^k_z are the predicted cases for the previous $h-1$ days when $h \geq 1$.

The following is an example representation for the above method:

Figure 1: At h=0, all L=5 inputs are real observations. As h grows, each new prediction replaces an observation, compounding the error with each step. By h=3, three of the five inputs are predictions. Each new window, k, resets to all-observed inputs.

2 Evaluation Methodology

The evaluation methodology uses multiple metrics, including both standard and operationally relevant metrics. It is based on a stratified evaluation process, where models are benchmarked across different lead times, outbreak regimes.

2.1 Operational Relevance

Standard error metrics like Root Mean Squared Error and Normalised Root Mean Squared Error focus on numeric differences between predicted and observed case counts which are insufficient to evaluate models that are to be used for public health decision making processes. They are useful as ad hoc metrics, but can hide under-prediction and regime-specific failure. While evaluating models for implementation, it is important to understand the requirements of the health system and the context in which the model will be used. Operational metrics evaluate whether forecast will be useful for resource allocation and planning. Operational metrics assess whether the model captures actionable signals such as risk categories, direction of change, timing of increases, and under-prediction during high-burden periods.

- Reliability: If the model is being used to predict for a longer time horizon, the degradation in performance should be assessed to ensure reliability
- Capturing the trend: If the models are to be used for early warning, they should be evaluated on their ability to capture the peak and rise in cases.
- Resource Allocation: If forecasts are used to allocate resources, then the models should be able to accurately predict the total burden.
- Data availability: If recent case reports arrive late, models should be evaluated under the same reporting-delay constraints.

2.2 Stratified Evaluation

The models are evaluated across different strata like:

Lead times To understand on how far ahead the forecast remains reliable and how much it degrades. A model that performs well one week ahead may not be useful for preparedness if its skill collapses at longer lead times, say 4 weeks ahead.

Case Percentile To assess model performance across varying levels of disease burden (low vs. high). Case counts during the evaluation period are divided into three categories: Low, representing values at or below the 33rd percentile; Medium, representing values between the 33rd and 67th percentiles; and High, representing values above the 67th percentile.

Season To compare model metrics across seasons: pre-monsoon, monsoon, and post-monsoon. This will show whether a model performs well during the peak transmission season or only in the off-season.

Regions Evaluate model performance across different regions to ensure the model works well in all regions.

Risk Categorisation Evaluate low, moderate, high, and critical risk classes separately. Since misclassifying a high-risk period as low risk is more costly,

2.3 Metric Categories

We use multiple metrics; both Standard Metrics and Operationally Relevant Metrics, since as discussed a single metric is not enough to capture all the aspects of forecast performance. Additionally, all these metrics are stratified by the regimes above

Magnitude accuracy Root Mean Squared Error and Normalised Root Mean Squared Error estimate how far predicted case counts are from observed counts.

Bias and LinEx loss Generally, under prediction is more costly than over-prediction in a health system. Bias identifies systematic under- or over-prediction and LinEx loss penalizes the more costly direction, here, underprediction.

Trend capture Pearson and trajectory correlation test whether forecasts track the rise, peak, and decline of cases.

Risk-band metrics Accuracy, precision, recall, and F1 evaluate whether forecasts classify actionable risk levels(low, medium, high, critical) correctly.

2.4 Benchmarking Models

We benchmark models against a naive persistence baseline to understand whether the models provide actionable improvement. We calculate skill scores to quantify this improvement.

Naive Persistence: The prediction at week t + h equals the observed value at week $t - k$ for a given lag k and, lead time, h :

$$\hat{y}_t = y_{t-1} \tag{5}$$

Skill Score can be used to quantify the performance of the model relative to the naive/baseline model. A positive skill score indicates improvement over the baseline For metrics where higher is better, such as accuracy, recall, F1, trajectory correlation, and direction accuracy, skill is computed as:

$$\text{Skill}(M) = \frac{M_{\text{model}} - M_{\text{NP}}}{1 - M_{\text{NP}}}.$$

For metrics where lower is better, such as LinEx, skill is computed as:

$$\text{Skill}(M) = \frac{M_{\text{NP}} - M_{\text{model}}}{M_{\text{NP}}}.$$

2.5 Qualification Criteria for the Ensemble of models

No single model is optimal across all regimes or geographies, hence an ensemble of models is used. The qualification criteria for a model to be included in the Ensemble must be context-driven. Model selection should consider the following aspects:

3 Appendix

3.1 Standard Metrics :

Let y_i denote the true values, $\hat{y}_i$ the predicted values, and n the number of samples.

Root Mean Squared Error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{6}$$

RMSE measures the square root of the average squared differences between the predicted and actual values. It penalizes larger errors more heavily due to squaring, making it sensitive to outliers. Lower RMSE indicates better model performance.

Normalized Root Mean Squared Error (NRMSE)

$$\text{NRMSE} = \frac{\text{RMSE}}{\frac{1}{n}\sum_{i=1}^{n} y_i} \tag{7}$$

NRMSE normalizes RMSE by the mean of the observed values, making it scale-independent. This allows for comparisons across regions(here, zones) and models.

Coefficient of Determination (R^2)

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{8}$$

where

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \tag{9}$$

R^2 measures the proportion of variance in the observed data that is explained by the model. Values closer to 1 indicate better fit.

Pearson Correlation Coefficient (ρ)

$$\rho = \frac{\sum_{i=1}^{n}(y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2 \sum_{i=1}^{n}(\hat{y}_i - \bar{\hat{y}})^2}} \tag{10}$$

where

$$\bar{\hat{y}} = \frac{1}{n}\sum_{i=1}^{n} \hat{y}_i \tag{11}$$

Pearson correlation coefficient measures the linear relationship between the true and predicted values. It ranges from -1 (perfect negative correlation) to 1 (perfect positive correlation), with 0 indicating no linear relationship. A higher correlation indicates that the model can capture trends (rises/falls) in the data.

3.2 Operational Metrics

Let y_i denote the observed case count and $\hat{y}_i$ denote the forecast for observation i. Let $b_i \in \{1, 2, 3, 4\}$ denote the observed risk band and $\hat{b}_i \in \{1, 2, 3, 4\}$ denote the predicted risk band, where the bands correspond to Low, Moderate, High, and Critical risk.

Risk-Band Classification Metrics For risk-band evaluation, forecasts are first converted into discrete risk bands using predefined endemic-channel thresholds. A confusion matrix C is then constructed, where

$$C_{ab} = \sum_i \mathbf{1}(b_i = a,\ \hat{b}_i = b),$$

with a indexing the observed band and b indexing the predicted band.

Overall accuracy measures the proportion of predictions assigned to the correct risk band:

$$\text{Accuracy} = \frac{\sum_{k=1}^{4} C_{kk}}{\sum_{a=1}^{4}\sum_{b=1}^{4} C_{ab}}.$$

For each band k, precision, recall, and F1 score are defined as:

$$\text{Precision}_k = \frac{C_{kk}}{\sum_{a=1}^{4} C_{ak}}, \qquad \text{Recall}_k = \frac{C_{kk}}{\sum_{b=1}^{4} C_{kb}},$$

$$\text{F1}_k = \frac{2 \cdot \text{Precision}_k \cdot \text{Recall}_k}{\text{Precision}_k + \text{Recall}_k}.$$

Precision answers: when the model predicts a given risk band, how often is it correct? Recall answers: when that risk band actually occurs, how often does the model detect it? F1 balances these two quantities and is useful when missing high-risk periods and over-alerting both matter.

Macro-averaged metrics assign equal weight to each risk band:

$$\text{Macro Recall} = \frac{1}{4}\sum_{k=1}^{4} \text{Recall}_k.$$

Weighted F1 weights each band by its observed support:

$$\text{Weighted F1} = \sum_{k=1}^{4} \frac{\sum_{b=1}^{4} C_{kb}}{N} \text{F1}_k,$$

where N is the total number of evaluated observations.

Bias

$$\text{Bias} = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i) \tag{12}$$

Bias measures the average tendency of predictions to overestimate or underestimate the true values. A positive bias indicates overprediction, while a negative bias indicates underprediction.

LinEx Loss Residual on raw scale:

$$e_i = y_i - \hat{y}_i$$

LinEx Loss:

$$\mathcal{L}_{\text{LinEx}} = \frac{1}{n}\sum_{i=1}^{n} [\exp(\alpha\, e_i) - \alpha\, e_i - 1]\, where, e_i = y_i - \hat{y}_i$$

where $\alpha = 0.75$ and $e_i = y_{\text{true}} - y_{\text{pred}}$ (raw values, no log transform).

Underprediction ($e > 0$) is penalised more than overprediction ($e < 0$). Loss $\to 0$ as $e \to 0$; asymmetry controlled by α.

Trajectory Correlation

$$\rho = \frac{\sum_t \left(y_{\text{true}}^{(t)} - \bar{y}_{\text{true}}\right)\left(y_{\text{pred}}^{(t)} - \bar{y}_{\text{pred}}\right)}{\sqrt{\sum_t \left(y_{\text{true}}^{(t)} - \bar{y}_{\text{true}}\right)^2 \cdot \sum_t \left(y_{\text{pred}}^{(t)} - \bar{y}_{\text{pred}}\right)^2}}$$

Defined when $n \geq 3$, $\text{std}(y_{\text{true}}) > 0$, $\text{std}(y_{\text{pred}}) > 0$; returns `NaN` otherwise.

Arrays must be sorted chronologically before computing.