λ-Guard: Structural & Stability Overfitting Index for Boosting

Fabrizio Di Sciorio, PhD[1]

[1]Department of Economics and Business, University of Almeria, Spain

16/02/2026

Overview

λ-Guard is a framework to detect overfitting **without using a test set**. Traditional overfitting measures rely on a held-out dataset to detect performance drops. λ-Guard instead analyzes:

- **Geometric structure** of the learned representation (how the model partitions the input space)
- **Stability** of predictions under small input perturbations

The model is decomposed into two key conceptual spaces:

1. **Representation Space (Capacity)** – measures how "rich" or complex the model representation is.
2. **Prediction Trajectory Space (Alignment)** – measures how effectively the model's components (trees) contribute to predicting the target.

Each tree in Gradient Boosting partitions the input space into leaf regions. We define a binary matrix Z where each row corresponds to an observation and each column to a leaf region across all trees:

$$Z_{i,j} = \begin{cases} 1 & \text{if observation } i \text{ falls into leaf } j \\ 0 & \text{otherwise} \end{cases}$$

This matrix is analogous to the **hat matrix** H in linear regression: it encodes how the model projects training data into its learned representation.

Mathematical Formulation

1. Leaf Membership Matrix Z

Given a dataset $X \in \mathbb{R}^{n\times d}$ and T trees, each tree t has L_t leaves. Define the total number of leaf regions as:

$$L = \sum_{t=1}^{T} L_t$$

Then $Z \in \mathbb{R}^{n\times L}$ is defined as above. Each row i represents the embedding of observation x_i into leaf space, while each column j represents a specific leaf region. Effectively, Z encodes the **geometric projection of the training data** into the model's functional representation.

2. Capacity C

Capacity quantifies the intrinsic dimensionality of the learned representation:

$$\bar{Z} = \frac{1}{n}\sum_{i=1}^{n} Z_i, \quad C = \frac{1}{n}\sum_{i=1}^{n} \|Z_i - \bar{Z}\|_2^2 = \mathrm{Var}(Z)$$

Intuition:

- High C → observations spread in many independent directions in leaf space → complex partitioning → more degrees of freedom → higher overfitting risk.
- Low C → most observations lie in few effective leaf combinations → simpler model.

Equivalently, in functional terms:

$$C = \mathrm{Var}(f(X)) = \frac{1}{n}\sum_{i=1}^{n}(f(x_i) - \bar{f})^2$$

3. Alignment A

Alignment measures how well the learned representation predicts the target $y \in \mathbb{R}^n$:

$$A = \mathrm{Corr}(f(X), y) = \frac{\mathrm{Cov}(f(X), y)}{\sigma_{f(X)}\sigma_y}$$

Intuition:

- High A → each tree contributes independent information toward predicting the target → efficient representation.
- Low A → later trees largely redundant → model may have wasted capacity.

4. Generalization Index GI

$$GI = \frac{A}{C}, \quad G_{\text{norm}} = \frac{A}{A+C} \in [0,1]$$

Interpretation:

- $G_{\text{norm}} \to 1 \to$ strong generalization, alignment dominates
- $G_{\text{norm}} \to 0 \to$ high capacity with low alignment → risk of overfitting

5. Instability Index S

$$S = \frac{1}{n} \sum_{i=1}^{n} \frac{|f(x_i) - f(x_i + \epsilon_i)|}{\sigma_f}, \quad \epsilon_i \sim \mathcal{N}(0, \sigma_\epsilon^2)$$

Interpretation:

- High S → model is unstable; small changes in input produce large prediction differences → overfitting risk
- Low S → model robust

6. Overfitting Index λ

$$\lambda = \frac{C}{A+C} \cdot S, \quad \lambda_{\text{norm}} = \frac{\lambda - \min(\lambda)}{\max(\lambda) - \min(\lambda)} \in [0,1]$$

Interpretation:

- High λ → many independent leaf regions that do not contribute to prediction + unstable predictions → strong overfitting signal
- Computable entirely on **training data**, no test set required

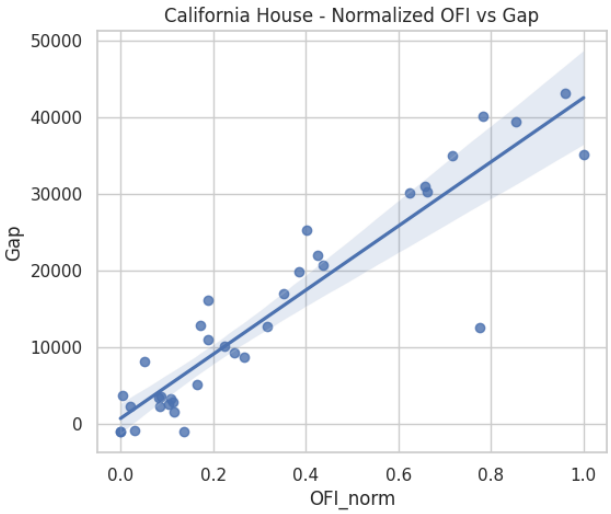

Figure 1: RMSE Test/Train gap vs λ guard - Correlation analysis

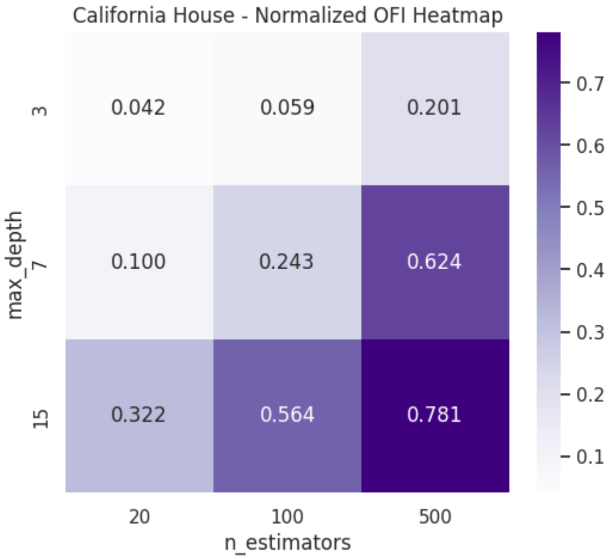

Figure 2: λ guard distribution

Geometric Interpretation

1. Z maps each observation into a high-dimensional leaf space
2. Capacity C measures the "spread" of points in this space
3. Alignment A captures how well this spread correlates with the target
4. Instability S detects whether the representation is sensitive to small input perturbations
5. λ combines both aspects into an overfitting score
6. Essentially, λ-Guard generalizes the hat matrix H concept to Gradient Boosting

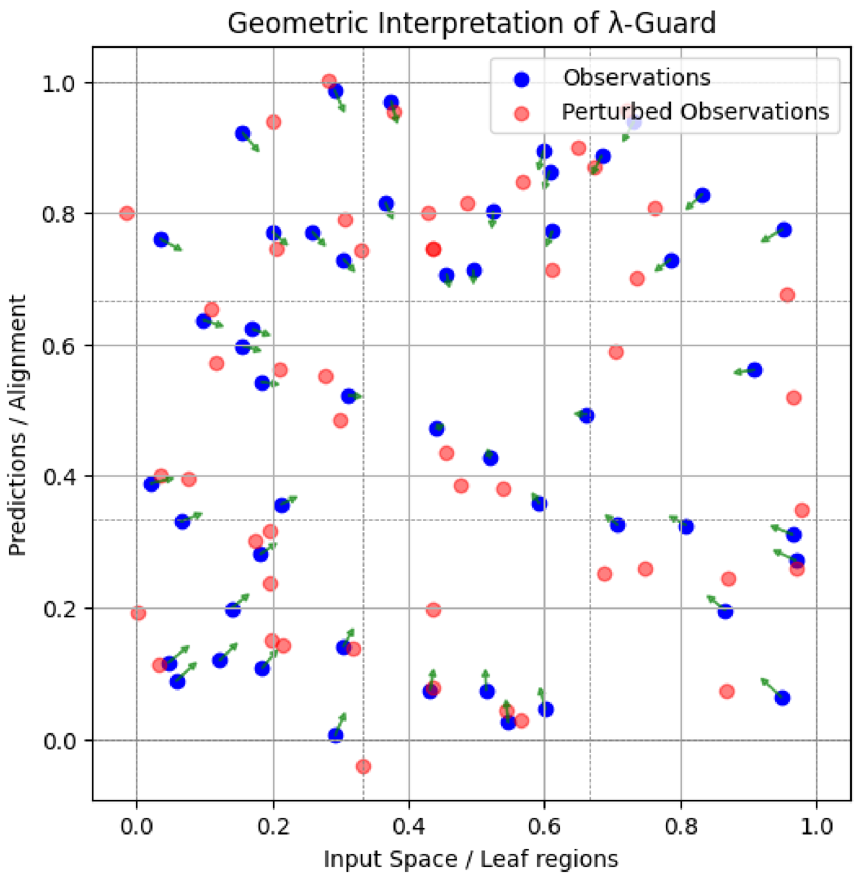

Figure 3: Geometric interpretation of λ-Guard. Gray squares: leaf regions, blue points: original observations, red points: instability, green arrows: alignment. High λ occurs when capacity is high, alignment low, and instability high.

1 The λ-Guard Test for Structural Overfitting

The λ-Guard framework is designed to detect *structural overfitting* in gradient boosting models **without using a test set**. While traditional measures of overfitting rely on the difference between training and validation error, λ-Guard examines the *structural dependence* of the model on each training observation, providing insights into both global model flexibility and local memorization.

1.1 Leaf Membership Matrix

Each tree in a gradient boosting ensemble partitions the input space into leaf regions. We define a binary *leaf membership matrix* $Z \in \mathbb{R}^{n \times L}$, where n is the number of training samples and $L = \sum_{t=1}^{T} L_t$ is the total number of leaf nodes across all trees:

$$Z_{i,j} = \begin{cases} 1 & \text{if observation } x_i \text{ falls into leaf } j, \\ 0 & \text{otherwise.} \end{cases}$$

Here, each row Z_i represents the embedding of a sample x_i in leaf space, and each column represents a specific leaf region. This matrix is analogous to the *hat matrix H* in linear regression, encoding how the training data is projected into the model's learned functional representation.

1.2 Leverage and Observed Statistics

For each training point i, we define the *leverage H_{ii}* as:

$$H_{ii} \approx \sum_{m=1}^{M} \frac{\eta}{|\mathrm{leaf}_m(x_i)|}$$

where η is the learning rate and $|\mathrm{leaf}_m(x_i)|$ is the number of observations in the leaf containing x_i for tree m. From the leverage vector $H = [H_{11}, \dots, H_{nn}]$, we define two key statistics:

- **Effective Degrees of Freedom ratio (global complexity)**:

  $$T_1 = \frac{1}{n} \sum_{i=1}^{n} H_{ii}$$

  Higher T_1 indicates that the model uses many degrees of freedom per point, suggesting potential *global overfitting*.

- **Peak leverage ratio (local memorization)**:

  $$T_2 = \frac{\max_i H_{ii}}{\frac{1}{n} \sum_{i=1}^{n} H_{ii}}$$

  Higher T_2 indicates that a few points dominate the fit, suggesting *local memorization*.

1.3 Bootstrap Null Distribution

To evaluate whether T_1 and T_2 are unusually large, we generate B bootstrap samples of the training data with replacement. For each bootstrap sample b, we compute the leverage $H^{(b)}$ and the statistics:

$$T_1^{(b)} = \frac{1}{n} \sum_{i=1}^{n} H_{ii}^{(b)}, \quad T_2^{(b)} = \frac{\max_i H_{ii}^{(b)}}{\frac{1}{n} \sum_{i=1}^{n} H_{ii}^{(b)}}$$

This yields empirical null distributions for both statistics under the assumption of a stable model.

1.4 Hypothesis Testing

We formulate the following one-sided hypothesis test:

- H_0: the model is structurally stable (no overfitting)
- H_1: the model exhibits structural overfitting, either globally (high T_1) or locally (high T_2)

Critical values are obtained from the $(1-\alpha)$ quantiles of the bootstrap distributions:

$$q_1 = \text{quantile}_{1-\alpha}(T_1^{(b)}), \quad q_2 = \text{quantile}_{1-\alpha}(T_2^{(b)})$$

Empirical p-values are computed as:

$$p_1 = \frac{1}{B}\sum_{b=1}^{B} \mathbf{1}\{T_1^{(b)} \geq T_1^{\text{obs}}\}, \quad p_2 = \frac{1}{B}\sum_{b=1}^{B} \mathbf{1}\{T_2^{(b)} \geq T_2^{\text{obs}}\}$$

The null hypothesis is rejected if:

$$\text{Reject } H_0 \quad \text{if } p_1 < \alpha \text{ OR } p_2 < \alpha$$

This logical OR ensures that either global overfitting or local memorization is sufficient to flag structural overfitting.

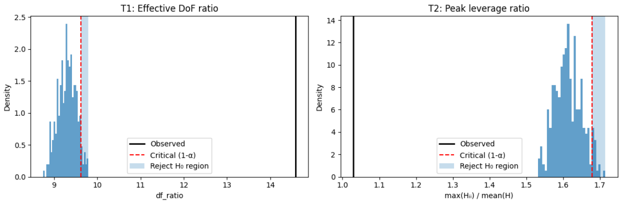

Figure 4: Test

1.5 Interpretation and Connection to λ

The diagonal H_{ii} serves as the **local λ index** in -Guard:

- mean(H_{ii}) → global complexity (T1)
- max(H_{ii})/mean(H_{ii}) → local memorization (T2)

Together, these measures allow classification of the model into one of four regimes:

1. Stable / smooth generalization
2. Global overfitting / interpolation
3. Local memorization / spike-dominated
4. Extreme interpolation (both T1 and T2 high)

References / Inspirations

- Hat matrix H in linear regression
- Gradient Boosting as a functional additive model
- Generalization Index (GI) framework
- λ in H Boosting matrix (pseudo residuals)