Bias Variance Tradeoff

By Thanuja Polani

Background

Every machine learning model can encounter errors in training. These errors are Bias and Variance and are inversely proportional

Bias-variance tradeoff is how we balance these errors to achieve optimal performance of the model

Bias

- Error due to overly simplistic models
- Occurs when a model is too simple and makes strong assumptions about the data.
- A high-bias model doesn't learn enough from the training data and performs poorly on both training and test data.
- A bias-prone model is like a student who only memorizes textbook definitions without understanding concepts, leading to poor performance on any new question.
- Eg: trying to fit a straight line (linear regression) to data that follows a curved pattern—it won’t capture the complexity of the data, leading to underfitting.

Variance

- Error due to overly complex models that fit noise
- Occurs when a model is too complex and captures noise along with the patterns in the training data.
- A high-variance model performs well on training data but poorly on new test data because it fails to generalize
- A variance-prone model is like a student who memorizes every question from past exams but struggles with any new format.
- Eg: A deep decision tree memorizing the training data instead of learning general patterns.

What's a good model?

- A good model should:
  - Capture enough complexity to learn important patterns (low bias).
  - Avoid memorizing unnecessary details to generalize well (low variance).

Model Complexity	Bias	Variance	Training Accuracy	Test Accuracy
Too Simple (Linear Regression)	High	Low	Low	Low
Too Complex (Deep Decision Tree)	Low	High	High	Low
Just Right (Balanced Model)	Low	Low	High	High

Tradeoff

- If a model underfits (high bias, low variance):
  - ✓ Use a more complex model (e.g., switch from linear regression to a decision tree).
  - ✓ Add more meaningful features (e.g., include location and number of bedrooms in house price prediction).
  - ✓ **Reduce** regularization (if using techniques like Lasso or Ridge regression).
- If a model overfits (low bias, high variance):
  - ✓ Simplify the model (e.g., prune a deep decision tree to reduce complexity).
  - ✓ **Use** regularization (L1/L2 regularization, dropout in neural networks).
  - ✓ Get more training data (helps the model generalize better).
  - Ensemble Methods: Use techniques like bagging (e.g., Random Forests) or boosting (e.g., XGBoost) to combine multiple models and reduce the overall variance.

Nutshell

- Bias -: Error due to overly simplistic assumption in model's algo
  - High bias causes model to miss relevant relationships between I/P features and target O/P
  - Effect: Underfitting
- Variance: errors due to excessive sensitivity to small fluctuations in training data
  - High variance cause the model to model the noise in the training data rather intended O/P
  - Effect: Overfitting (since it captures noise like a natural pattern)
- Tradeoff: Find the balance that minimizes the error (Bias+var). Typically it involves the adjusting the complexity of the model
  - Simplify a complex model to reduce variance - > which may increase bias
  - Complicate a model to reduce bias -> which may increase the var
- Sweet spot lies somewhere between
- Solution: **Regularization**, **cross-validation**, and **ensemble methods**

Learning Curves

- Graphical representation of a model's performance over time or with more training data.

In machine learning, learning curves typically plot:

•Training Error: The error (or loss) calculated on the training dataset.

•Test Error: The error (or loss) calculated on a separate unseen test dataset

•These errors are usually plotted against the number of training iterations (epochs), the size of the training data, or time

Learning curves help us visualize how bias and variance evolve during the training process.

1.Training Error and Bias:

- Training error tends to decrease as the model learns.
- Early in training, the model starts with high bias because it's simple and doesn't understand the data well. As training progresses, the model learns more and becomes better at fitting the training data, so the training error drops.
- Low training error typically indicates that the model has reduced its bias (i.e., it is becoming more capable of fitting the data).

2.Test Error and Variance:

- Test error initially decreases as the model gets better at fitting the data and generalizing to new, unseen data. However, after a certain point, the test error may increase. This increase is a sign that variance is dominating.
- As a model becomes more complex and fits the training data more closely, it starts capturing not just the true patterns but also the noise or randomness in the data. This leads to overfitting, where the model does well on the training data but fails to generalize well to new data, causing the test error to rise.

Key Phases in Learning Curves

Underfitting (High Bias):

- Early Training Stage:
  - At the start of training, both training error and test error are high.
  - The model is underfitting because it is too simple (high bias) and is not learning the data's underlying patterns.
  - In this phase, training error is still high because the model hasn't learned much yet.
  - Test error is also high because the model is not capturing enough complexity to generalize well.
- What to look for: When both the training error and test error are high, this indicates that the model has high bias (underfitting). The model may be too simple, and you should consider using a more complex model or adding features.

Key Phases in Learning Curves

Optimal Fit:

- Mid Training Stage:
  - As training continues, training error continues to decrease because the model is learning and improving.
  - Test error starts to decrease as well, reflecting the model's improved ability to generalize to unseen data.
  - This is the sweet spot where the model is learning to capture the underlying data patterns without fitting noise. At this point, bias is low, and variance is also balanced.
- What to look for: If the test error is decreasing and getting close to the training error, it indicates that the model is fitting the data well and generalizing effectively. This is the point of optimal model performance, where bias and variance are balanced.

Key Phases in Learning Curves

Later Training Stage:

- As the model becomes increasingly complex (e.g., by training for more steps or adding features), training error continues to decrease because the model is fitting the training data more closely.
- However, test error starts to increase after a certain point. This happens because the model is starting to fit noise in the training data, causing it to perform poorly on new, unseen data.
- This is a classic case of overfitting, where the model is too complex and variance dominates, leading to poor generalization.

•What to look for: When the training error keeps decreasing, but the test error starts increasing, it indicates high variance (overfitting). The model is too complex for the amount of training data, and you need to simplify it or apply regularization.

Interpreting Learning Curves

•Training Error: Starts high, decreases as the model learns more.
•Test Error: Decreases initially, but starts increasing once the model starts overfitting.
•Underfitting: Both training and test errors are high.
•Optimal Fit: Test error is low, and close to training error.
•Overfitting: Test error increases, even as training error decrease

By analyzing the learning curves, you can infer:
•Underfitting: When the test and training errors are both high, the model is too simple (high bias). You should try increasing model complexity, adding more features, or training for more epochs.
•Good Fit: If both training and test errors are low, the model is well-calibrated, and you have found the right balance between bias and variance.
•Overfitting: When the training error is low but the test error starts to rise, the model has overfit the data (high variance). You can reduce overfitting by:

- Simplifying the model (reducing complexity).
- Applying regularization techniques (L1, L2).
- Using cross-validation.
- Increasing training data.

Thank You