How to Easily Distinguish Between Validation and Test Sets

Authored by

stats writer

December 3, 2025

In the realm of machine learning and statistical modeling, the process of training and evaluating a predictive model requires careful partitioning of the available data. Two critical components often confused by practitioners are the Validation Set and the Test Set. While both are subsets of data used for evaluation, they serve fundamentally different purposes throughout the development lifecycle.

The Validation Set is primarily utilized during the training phase to tune the hyperparameters and optimize the internal structure of the learning algorithm. Its continuous use helps prevent overfitting to the training data. Conversely, the Test Set is strictly reserved until the very end; it provides a single, final assessment of the performance of the finalized model on truly unseen data, yielding an unbiased estimate of its real-world generalization capability.

The Essential Role of Data Splitting

To ensure that a predictive model is robust and generalizes effectively to new observations, standard practice dictates splitting the original dataset into three distinct portions. This separation is crucial for maintaining integrity in both the development and final evaluation stages. Each subset serves a unique function necessary for successful model deployment.

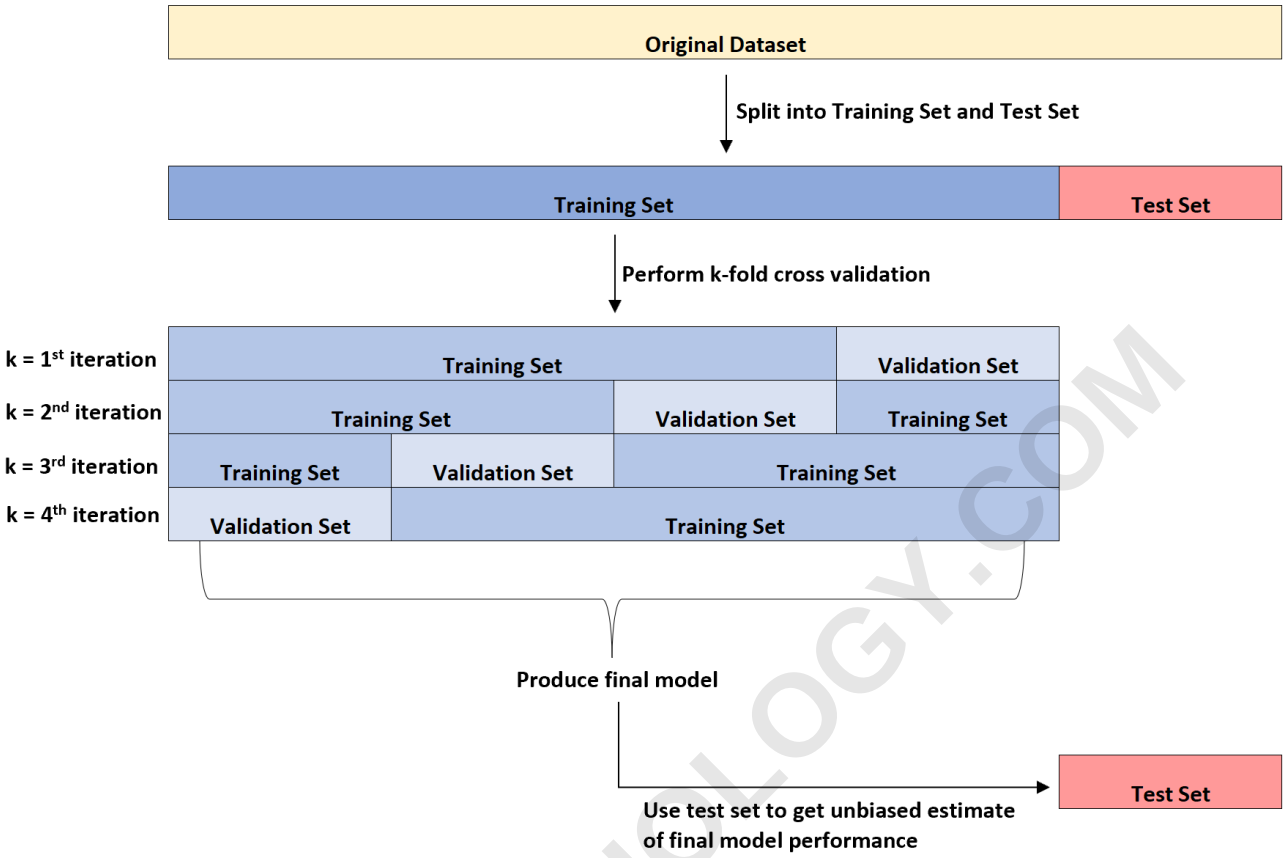
The allocation of data typically follows common ratios like 60/20/20 or 70/15/15, depending on the dataset size and complexity of the task. Adhering to these partitions ensures that the learning process, the selection process, and the final evaluation are all conducted independently, mitigating the risk of data leakage and overly optimistic performance metrics.

Defining the Three Data Subsets

When training a machine learning algorithm, three primary data partitions are utilized:

Training Set: This is the largest portion of the data, used exclusively to train the model--that is, allowing the algorithm to learn the relationships and patterns inherent in the data by adjusting its internal weights and biases.
Validation Set: This subset is used during the training phase to fine-tune the structural choices or hyperparameters (e.g., learning rate, number of layers, regularization strength) of the model. It helps the developer select the best configuration without contaminating the final evaluation.
Test Set: This dataset is held back completely and is used only once, after the model is fully trained and all optimization decisions are complete, to obtain an unbiased estimate of its performance on completely novel data.

The following diagram visually illustrates how these three subsets relate to the total available data:



Distinguishing Validation from Testing

A common source of confusion among data science students is the precise operational difference between the validation set and the test set, as both are used for evaluating performance. The distinction lies entirely in the stage of development at which they are employed and the fundamental purpose of the evaluation.

The validation set acts as a crucial proxy for unseen data during the iterative process of model building. By measuring the model's error on this set, researchers can decide which model architecture or set of hyperparameters performs best. Because the model structure is chosen based on the validation set performance, the validation error is inherently optimistic and biased towards the training procedure.

The test set, conversely, provides a clean, final assessment. It is critical that this data remains untouched until the very end, ensuring that the final performance metric accurately reflects the model's true capability to generalize. If the test set were used repeatedly for optimization, the resulting performance estimate would also become biased, rendering the entire evaluation process unreliable.

Addressing Estimation Bias through the Test Set

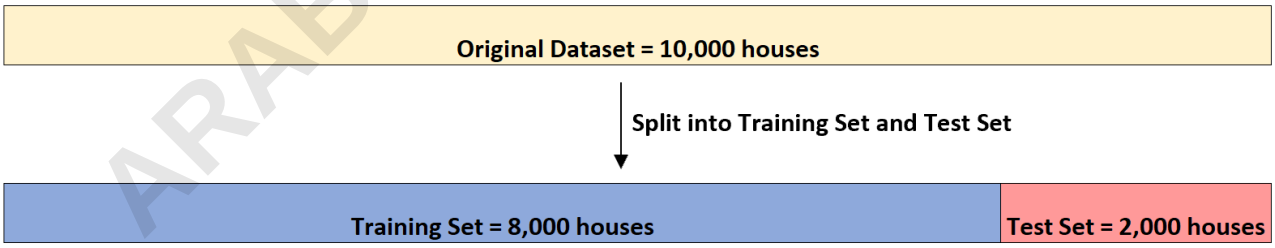
Statistical theory demonstrates that error rates measured during the iterative selection process--even those derived from sophisticated techniques like k-fold cross validation--tend to slightly underestimate the true error rate that the model will exhibit when applied to entirely new, real-world data. This underestimation occurs because the model selection process inadvertently incorporates knowledge derived from observing these interim evaluation sets.

To counteract this inherent optimism, we must rely on the dedicated test set. By fitting the finalized, best-performing model (selected via the validation process) to this completely reserved dataset, we obtain an unbiased estimate of the true error rate. This final metric is the most trustworthy predictor of how the model will perform once deployed in a production environment.

Case Study: Predictive Modeling in Real Estate

To solidify the distinction between these two datasets, let us consider a practical scenario involving a real estate investor aiming to predict house selling prices using a machine learning model. The investor uses three key features for prediction: **(1)** number of bedrooms, **(2)** total square feet, and **(3)** number of bathrooms.

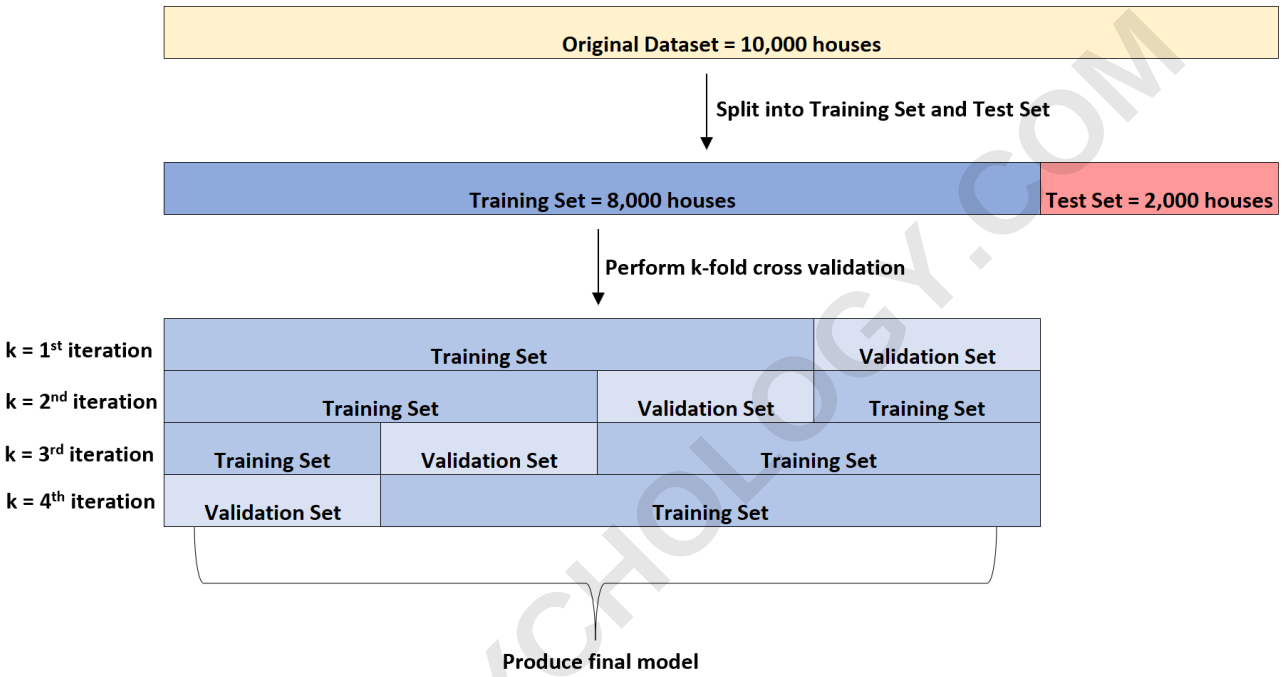
Suppose the investor possesses a comprehensive dataset containing detailed information on 10,000 houses. The first and most critical step is the initial partitioning of this entire dataset into the training data and the final evaluation data. In this example, the investor splits the data into a large 8,000-house portion for training and internal validation, and a 2,000-house portion strictly reserved as the test set.



Utilizing K-Fold Cross Validation for Optimization

With the 2,000-house test set safely isolated, the investor then focuses on the remaining 8,000 houses to train and optimize the model. To robustly select the optimal regression algorithm (e.g., standard linear regression vs. ridge regression) and tune its hyperparameters, the investor employs a rigorous process known as k-fold cross validation.

In this k-fold cross validation scheme, the 8,000-house dataset is repeatedly divided. For instance, in a 4-fold process, the data might be split such that 6,000 houses are used for immediate training (the training set), and 2,000 houses are designated as the validation set. This rotation allows every part of the 8,000-house block to act as both training and validation data across different iterations. The investor repeats this process across several different types of regression models to find the one exhibiting the lowest average error on the validation folds.

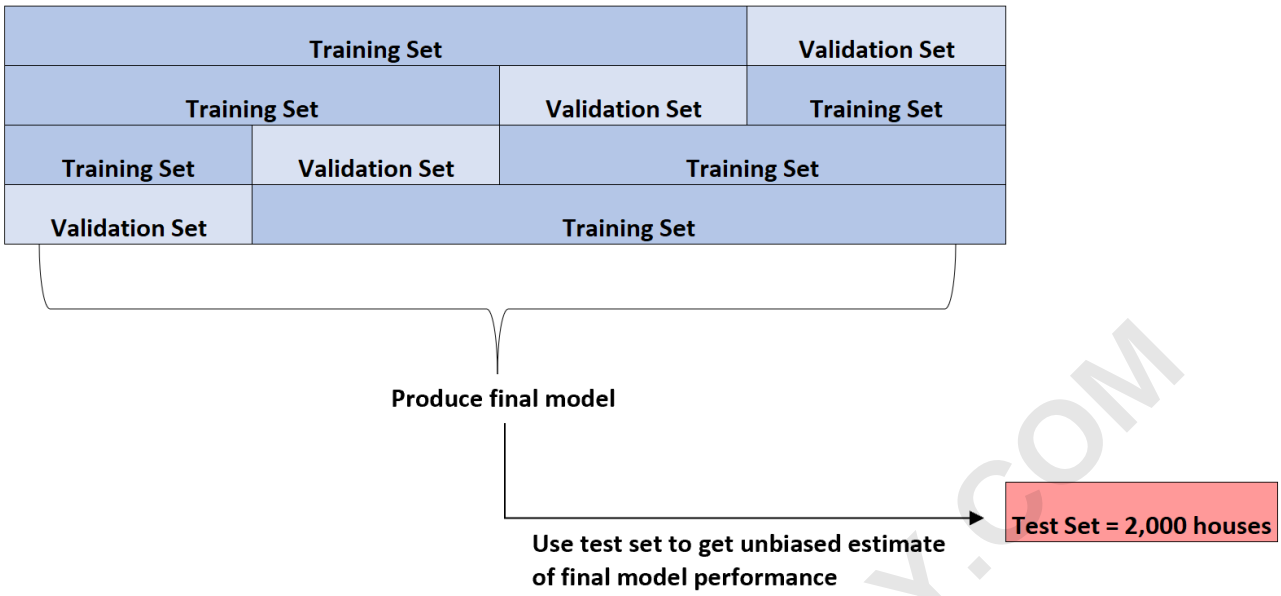


Final Evaluation Using the Unseen Test Set

Only after the investor has thoroughly performed the k-fold cross validation--and identified the best possible model structure based on the validation performance--will they proceed to the final, crucial step: evaluation using the reserved test set.

Imagine the optimization process identifies a specific type of regression model that yields a mean absolute error (MAE) of **$8,345** across the validation folds. This MAE is the metric used for selection. However, because this figure was used to select the model, it is a biased representation of performance.

The investor then applies this exact, finalized model structure to the 2,000 houses in the held-out test set--data that the model has never encountered in any phase of training or optimization. Upon testing, the investor finds that the true MAE of the model on this unseen data is **$8,847**. This higher figure is the definitive, unbiased estimate of the true mean absolute error, providing a realistic expectation of the model's performance in the market.



Summary of Roles

In summary, while the validation set is instrumental in guiding the development process and selecting the best hyperparameters, it does not provide the final, trustworthy measure of generalization. That definitive task belongs exclusively to the test set, which must remain separate to ensure the final performance metric is an unbiased estimate of how the chosen predictive system will perform in the real world.

This strict adherence to data separation is the cornerstone of reliable machine learning evaluation, ensuring that results reported are not inflated by selection bias.

How to Perform K-Fold Cross Validation in Python