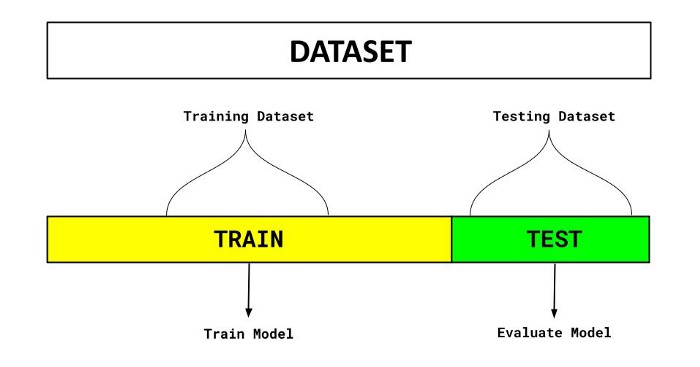
**Hold-Out Approach**

The hold-out approach involves splitting the data into multiple parts and using one part to train the model and the rest to validate and test it. It can be used for both model evaluation and selection.

In cases where every data is used for training the model, the problem remains with selecting the best model from a list of possible models. There is a need to have a mechanism that allows the model to be trained on one set of data and tested on another set of data. Primarily, we want to identify which model has the lowest generalization error or makes a better prediction on future or unseen datasets than all others.

**Hold-Out Method for Model Evaluation**

Model evaluation using the hold-out method entails splitting the dataset into training and test datasets, evaluating model performance, and determining the most optimal model. This diagram illustrates the hold-out method for model evaluation.



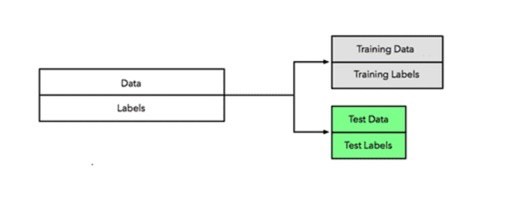
Hold-out method for model evaluation

There are two parts to the dataset in the diagram above. One split is a training set. Another set is held back for testing or evaluation of the model. The percentage of the split is determined based on the amount of training data available. A typical split of 70–30% is used in which 70% of the dataset is used for training and 30% is used for testing the model.

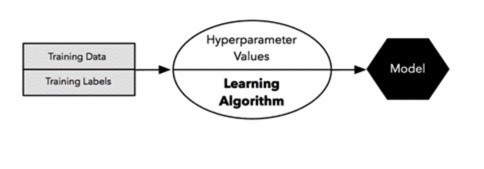
The objective of this technique is to select the best model based on its accuracy on the testing dataset and compare it with other models. Since the final model is trained to fit well (or overfit) the test data, it won’t generalize well to unknowns or future datasets. There is, however, the possibility that the model can be well-fitted to the test data using this technique. In other words, models are trained to improve model accuracy on test datasets based on the assumption that the test dataset represents the population. As a result, the test error becomes an optimistic generalization error estimation.

Follow the steps below for using the hold-out method for model evaluation:

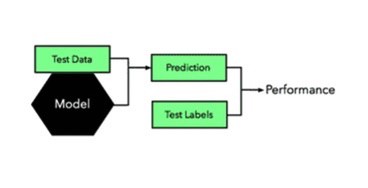
1. Split the dataset in two (preferably 70–30%; however, the split percentage can vary and should be random).



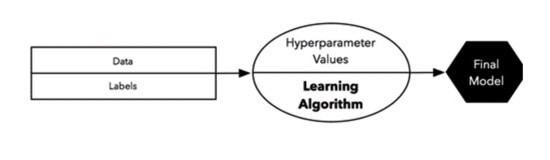
2. Now, we train the model on the training dataset by selecting some fixed set of hyperparameters while training the model.



3. Use the hold-out test dataset to evaluate the model.



4. Use the entire dataset to train the final model to generalize better on future datasets.



The dataset is split into training and test sets in this process, and a fixed set of hyperparameters is used to evaluate the model. There is another process in which data can also be split into three sets, and these sets can be used to select a model or to tune hyperparameters. We will discuss that technique next.

**Hold-Out Method for Model Selection**

Sometimes the model selection process is referred to as hyperparameter tuning. During the hold-out method of selecting a model, the dataset is separated into three sets — training, validation, and testing.



Hold-out method for model selection

Follow the steps below for using the hold-out method for model selection:

1. Divide the dataset into three parts: training dataset, validation dataset, and test dataset.
2. Now, different machine learning algorithms can be used to train different models. You can train your classification model, for example, using logistic regression, random forest, and XGBoost.
3. Tune the hyperparameters for models trained with different algorithms. Change the hyperparameter settings for each algorithm mentioned in step 2 and develop multiple models.
4. On the validation dataset, test the performance of each of these models (associating with each of the algorithms).
5. Choose the most optimal model from those tested on the validation dataset. The most optimal model will be set up with the most optimal hyperparameters. Using the example above, let’s suppose the model trained with XGBoost with the most optimal hyperparameters is selected.
6. Finally, test the performance of the most optimal model on the test dataset.