Cross Validation

Amisi FIKIRINI AMISI
Elie MULAMBA NGAMBWA
Salomon MUHIRWA

African Master in Machine Intelligence (AMMI), Senegal

Supervised by: Moustapha Cisse

Fondation of Machine Learning and Deep Learning

April 13, 2023

AIMS | African Institute for Mathematical Sciences SENEGAL

Contents

1. Introduction

Training a machine learning model is a very important step as this process allows us to confirm or deny if the model is good or not. To do this, the process requires dividing the general dataset into two slices called **Training data** and **Test data**.

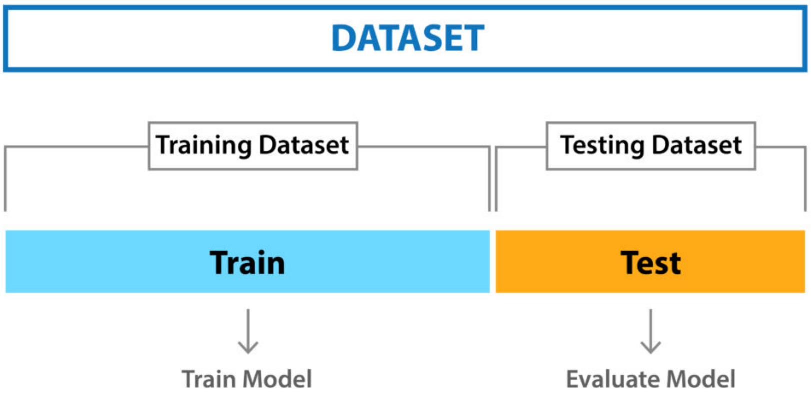


Figure 1.1: Dataset splitted in Training and Test Data.

So, once the training of the model is done, it is not certain that the model will perform well on data that it has never seen before. In order words, it is not sure that the model will have the desired precision and variance in front of data it has never seen before. However, we need some assurance about the accuracy of the predictions that the model makes; for this, we need to validate the model.

1.1 Definition

Model validation is the process of verifying that a model meets all the requirements that have been set for it. This includes verifying that the model is accurate and complete, as well as verifying it is consistent with other models.

1.2 Importance of model validation

As metionned in the introduction, validate a model is very important in order to be sure with the prediction's accuracy. So, there are some reasons about the importance of model validation :

- It helps ensure that the data we are working with is clean and accurate;

- It can help to catch errors early on in the build process, before they cause major problems down the line;
- It helps us build more robust and reliable models overall.

1.3 Working Principle

Besides the traditional method, there is another method of splitting data in training and test data for model validation called **Cross Validation**.

Cross-validation is a technique used to assess the accuracy of a model. It works by splitting the data into a training set and a test set. The model is fit on the training set, and then predictions are made on the test set. The accuracy of the model is then assessed by comparing the predictions to the actual values in the test set.

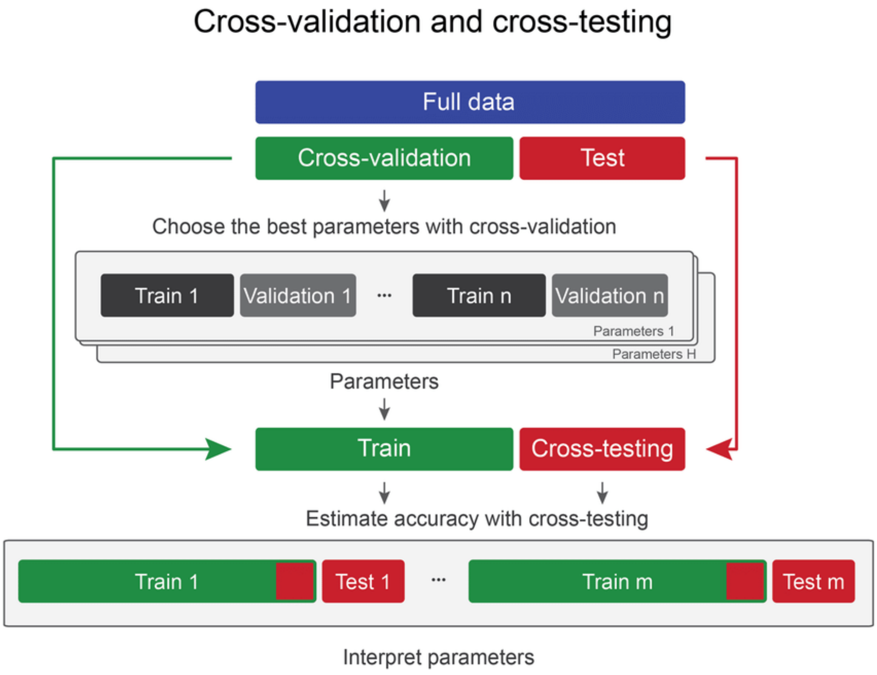


Figure 1.2: Cross Validation Process.

2. Types of Cross Validation

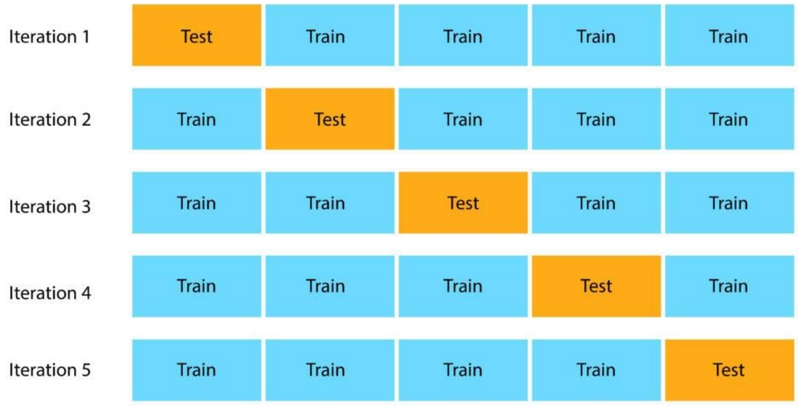
It is important to know that there are several types of cross validation of which, within the framework of this work we have chosen only 3 following :

2.1 K-Fold cross-validation

K-fold is a technique for assessing the performance of a machine learning model. The basic idea behind is to split the available data into k equal subsets or folds.

2.1.1 Working Principle

The K-fold method involves randomly splitting the dataset into *K folds* (typically K $=$ 5 as presented in the figure 2.1). The model is then fit on K-1 folds and evaluated on the remaining fold. This process is repeated *K times*, with each fold serving as the test set once. The evaluation metric, such as accuracy or mean squared error, is recorded for each iteration. After all k iterations are complete, the average performance metric is computed across all iterations as an estimate of the model's performance. i.e, The final model is then evaluated on the entire dataset. This approach can be used when there is a limited amount of data available.



Iteration 1	Test	Train	Train	Train	Train
Iteration 2	Train	Test	Train	Train	Train
Iteration 3	Train	Train	Test	Train	Train
Iteration 4	Train	Train	Train	Test	Train
Iteration 5	Train	Train	Train	Train	Test

Figure 2.1: KFold Cross Validation

2.1.2 Algorithm

Let's S=$\bigcup_{i=1}^{k} S_i$

K= number of folds

$\forall(i,j)$:S$_i \cap S_j = \phi$

For i =1 to K

- Train h_i on $A = S - S_i$
- Test h_i on S_i and Compute the $Error_i$
- Average $Error_i$

Generalize Error

2.1.3 Advantages

More accurate estimate of model performance : K-fold cross-validation provides a more accurate estimate of the model's performance than other methods like a traintest split because it uses all available data for training and testing.

Better utilization of data : By dividing the data into K folds, k-fold cross-validation enables us to use all available data for both training and testing.

Reduced risk of overfitting : K-fold cross-validation reduces the risk of overfitting, as the model is tested on different subsets of the data during the validation process.

Improved model generalization : K-fold cross-validation allows us to estimate how well the model will generalize to new, unseen data.

2.1.4 Disadvantages

Increased computational time : K-fold cross-validation requires fitting the model K times.

Sensitivity to data imbalance : If the data set is imbalanced, meaning that some classes have significantly more samples than others, k-fold cross-validation can result in biased estimates of model performance.

Higher variance in performance estimates : The performance estimates obtained from k-fold cross-validation can have higher variance than other methods.

Increased complexity of hyperparameter tuning : K-fold cross-validation can make hyperparameter tuning more complex, as it requires fitting the model multiple times for each combination of hyperparameters.

2.2 Leave-one-out cross-validation

Provides train/test indices to split data in train/test sets. Each sample is used once as a test set (singleton) while the remaining samples form the training set.

2.2.1 Working Principle

Leave-One-Out technique involves leaving out one data point from the dataset and fitting the model on the remaining data points. The model is then evaluated on the data point that was left out. This process is repeated for all data points in the dataset. This approach can be used when there is a limited amount of data available and when each data point is important.

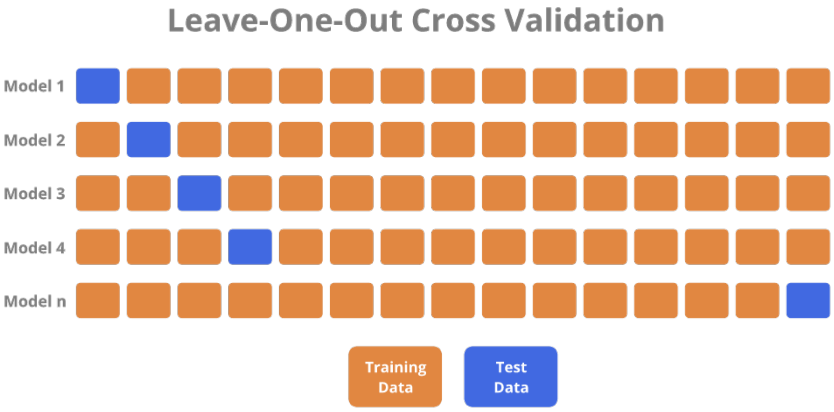


Figure 2.2: Leave-One-Out Cross Validation

2.2.2 Algorithm

1. Splitting

   Randomly split S_{train} into m disjoint subsets of training examples. where m = the size of S_{train} Let's call these subsets $S_1, ...S_k$

2. Training For i= 1 to m

   - Train h_i on $S_1 \cup ... \cup S_{i-1} \cup S_{i+1} \cup ... \cup S_m$
   - Test h_1 on S_i => Compute the $Error_i$
   - Average the $Error_i$

   Generalize the Error

2.2.3 Advantages

Lower Bias : uses almost all of the data for training in each fold

$$bias\_loo = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2 \tag{2.2.1}$$

$$bias\_k - fold = \frac{1}{k}\sum_{j=1}^{k}\frac{1}{m_j}\sum_{i=1}^{j}(y_i - \hat{y}_i)^2 \tag{2.2.2}$$

Good for small dataset : because it provides a more accurate estimate of the model's true performance.

2.2.4 Disadvantages

Higher variance can be expressed as:

$$var\_LOO = \frac{1}{n}\sum_{i=1}^{n},(y_i - f_i)^2 - bias_{LOO})^2 \tag{2.2.3}$$

$$var_{k-fold} = \frac{1}{k}\sum_{j=1}^{k}\frac{1}{n_j}\sum_{i=1}^{nj}(y_i - f_i)^2 - bias_{k-fold})^2 \tag{2.2.4}$$

Where var_{LOO} and var_{k-fold} are the variances of the prediction errors

Computationally expensive : LOO can be computationally expensive.

2.3 Stratified K-Fold cross-validation

Sometimes we may face a large imbalance of the target value in the dataset. For example, in the case of classification, in the cats and dogs dataset there might be a large shift towards the dog class. Stratified k-Fold is a variation of the standard k-Fold CV technique which is designed to be effective in such cases of target imbalance. i.e, is similar to k-fold cross-validation, but it ensures that each fold contains an equal proportion of data from each class (if the dataset is labeled). This approach can be used when there are a small number of data points available.

2.3.1 Working Principle

Stratified k-Fold splits the dataset on *k* folds such that each fold contains approximately the same percentage of samples of each target class as the complete set. In the case of regression, Stratified k-Fold makes sure that the mean target value is approximately equal in all the folds.

2.3.2 Algorithm

Spit data into different specific classes

$S = \bigcup_{i=1}^{K} S_i$

for i = 1 to K

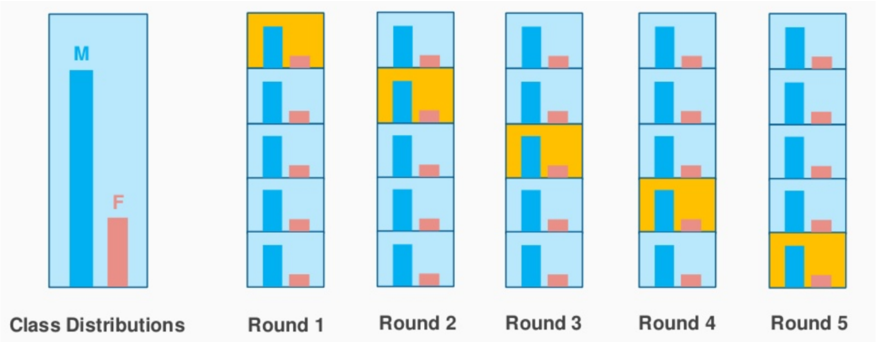


Figure 2.3: Stratified Cross Validation

$S_{train} = S – S_{i-merged}$

$S_{test} = S_{i-merged}$

$Calculate\ the\ Error_i$

Generalization Error

2.3.3 Advantages

Stratified KFold ensures that the proportion of the feature of interest is the same across the original data, training set and the test set.

Checking Model Generalization : Cross-validation gives the idea about how the model will generalize to an unknown dataset.

Checking Model Performance : Cross-validation helps to determine a more accurate estimate of model prediction performance.

2.3.4 Disadvantages

Higher Training Time : with cross-validation, we need to train the model on multiple training sets.

Expensive Computation : Cross-validation is computationally very expensive as we need to train on multiple training sets.

3. Implementation and Experiments

In order to evaluate the effectiveness of three different cross-validation techniques, we employed the logistic regression algorithm and assessed the performance of each model using each technique. This allowed us to compare the results of each approach and determine which method provided the best results in terms of accuracy and loss. Through this evaluation process, we gained valuable insights into the relative strengths and weaknesses of each technique, which can inform future model development efforts.

Logistic regression is a popular statistical model used for binary classification tasks, where the goal is to predict the probability of an instance belonging to one of two classes. In this model, a linear function of the input features is passed through a sigmoid activation function, which maps the output to a probability value between 0 and 1. This probability is then thresholded to produce a binary prediction.

3.1 Data Presentation

To generate a dataset for a binary classification task, we utilized Scikit-Learnś make_moons function. The resulting dataset comprises 1000 data points, each possessing two unique features. In order to introduce a degree of variability into the dataset, we incorporated noise into the generated data. To achieve this, we specified the noise parameter during the data generation process, setting it to 3 percent.

Adding noise to a dataset can be accomplished in various ways, depending on the nature and characteristics of the data. For instance, in the case of the make_moons function, noise is introduced by displacing the data points from their original positions along the half-circle shapes that form the basis of the dataset. By varying the amount of noise, we can control the degree of difficulty involved in the classification task and ultimately assess the performance of the machine learning algorithm.

Here is an excerpt from the data collection

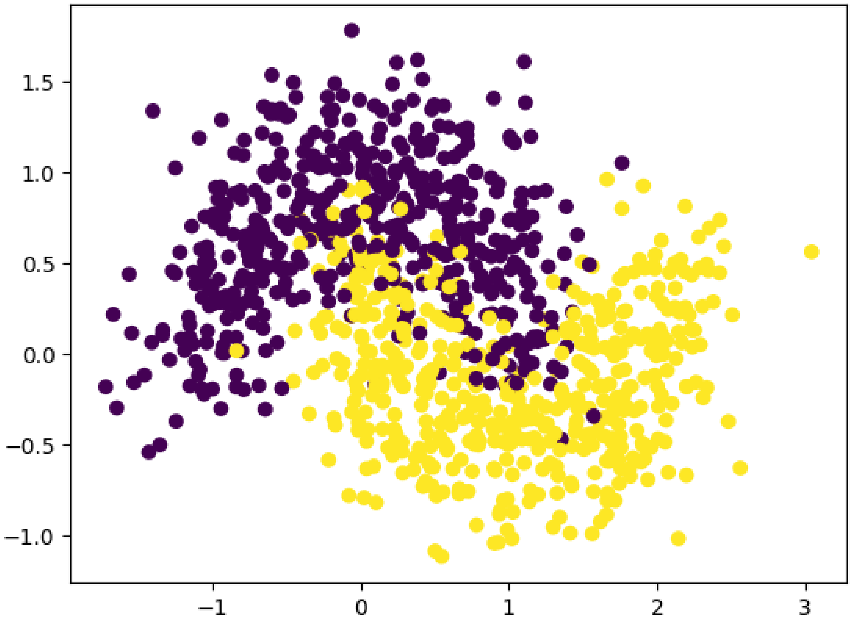


Figure 3.1: Plot of Data

3.2 Results

About the results obtain, here we have compared the results using from scratch process and using sklearn library.

Here, we are going to compare the accurancy and loss based on use of the traditionnal approach and cross validation approach

CV technique	Lost	Train Accuracy	Test Accuracy
Traditionnal split	0.36	85.86	79.0
K-fold	0.18	82.5	79.0
Leave-One-Out	0.0	100	58
Stratified	0.03	81	91

Table 3.1: Cross validation technique based on logistic regression model from scratch

3.3 Discussion

Firstly, we can observe that the traditional split method has a high loss value of 0.36 and a test accuracy score of 79.0% while achieving a relatively high training accuracy score of 85.86%. This suggests that the model may be overfitting the training data and not generalizing well to unseen data.

K-fold CV has a lower loss value of 0.18 and a test accuracy score of 79.0%, which is similar to the traditional split method. However, it achieves a lower training accuracy score of 82.5%. This suggests that K-fold CV may be better at generalizing to unseen data and avoiding overfitting compared to the traditional split method.

Leave-One-Out CV has a perfect training accuracy score of 100%, but a very low test accuracy score of 58%. This suggests that the model is overfitting to the training data and not generalizing well to unseen data. Leave-One-Out CV may not be suitable for this problem and may lead to poor generalization performance.

Stratified CV achieves a low loss value of 0.03 and a test accuracy score of 79.0%, which is similar to the traditional split and K-fold CV methods. However, it achieves a lower training accuracy score of 81.91% compared to the traditional split method. This suggests that stratified CV may be better at generalizing to unseen data and avoiding overfitting compared to the traditional split method.

Overall, the choice of CV technique depends on the specific problem and dataset. In this case, K-fold CV and stratified CV appear to be better at generalizing to unseen data and avoiding overfitting compared to the traditional split method. Leave-One-Out CV may not be suitable for this problem and may lead to poor generalization performance. However, it is important to experiment with different techniques and evaluate their performance to determine the most appropriate CV technique for a given problem.

4. Conclusion

In conclusion, the choice of CV technique depends on the specific problem and dataset. Each CV technique has its advantages and limitations. The traditional split method may be simple and easy to use, but it may lead to overfitting and poor generalization performance. K-fold CV and stratified CV may be better at avoiding overfitting and generalizing to unseen data. However, Leave-One-Out CV may not be suitable for some problems, and it may lead to overfitting.

Therefore, it is essential to experiment with different CV techniques and evaluate their performance to determine the most appropriate technique for a given problem. By selecting an appropriate CV technique, we can estimate the model's performance on unseen data, avoid overfitting, and build models that generalize well to new data.

References

Sylvain Arlot and Alain Celisse. A survey of cross-validation procedures for model selection. *Statistics surveys*, 4:40–79, 2010.

Yoshua Bengio and Yves Grandvalet. Cross-validation: what does it estimate and how well does it do it? In *Proceedings of the 2004 International Conference on Artificial Intelligence and Statistics*, volume 14, pages 23–30, 2004.

Bradley Efron and Robert J Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall/CRC, 1994.

Ron Kohavi. Cross-validation and the bootstrap: Estimating the error rate of a prediction rule. *Machine learning*, 14(2):113–143, 1995.

Andrew Ng, editor. *CS229 Lecture notes on Machine Learning*. Stanford University.

Web1. Importance and types of cross validation techniques. Towards Data Science, https://towardsdatascience.com/why-and-how-to-cross-validate-a-model-d6424b45261f, Accessed April 2023.

Web2. Stratified cross-validation in machine learning. Towards Data Science, https://towardsdatascience.com/what-is-stratified-cross-validation-in-machine-learning-8844f3e7ae8e, Accessed April 2023.