What is Cross Validation in Machine learning? Types of Cross Validation

By Great Learning Team / Updated on Jun 6, 2022 / 46564

Table of contents

What is Cross Validation?

Cross-validation is a statistical method used to estimate the performance (or accuracy) of machine learning models. It is used to protect against overfitting in a pred
limited. In cross-validation, you make a fixed nu
and then average the overall error estimate.

When dealing with a Machine Learning task, you
suitable algorithm which can give you the best

Say, you have trained the model with the datas
perform. One approach can be that you are goi
may not be a good practice.

So what is wrong with testing the model on the
represents all the possible scenarios of real-wo
model should be able to work well on the real-w
represents a small set of all the possible data p

So to know the real score of the model, it should
data is usually called testing set. But if we split

some important information that the test dataset may hold? Let us see the different types of cross validation and find out the answers?

Types of Cross Validation

There are different types of cross validation methods, and they could be classified into two broad categories – Non-exhaustive and Exhaustive Methods. We're going to look at a few examples from both the categories.

Non-exhaustive Methods

Non-exhaustive cross validation methods, as the name suggests do not compute all ways of splitting the original data. Let us go through the methods to get a clearer understanding.

Holdout method

This is a quite basic and simple approach in which we divide our entire dataset into two parts viz- training data and testing data. As the name, we train the model on training data and then evaluate on the testing set. Usually, the size of training data is set more than twice that of testing data, so the data is split in the ratio of 70:30 or 80:20.

In this approach, the data is first shuffled randomly before splitting. As the model is trained on a different combination of data points, the model can give different results every time we train it, and this can be a cause of instability. Also, we can never assure that the train set we picked is representative of the whole dataset.

Also when our dataset is not too large, there is a high possibility that the testing data may contain some important information that we lose as we do not train the model on the testing set.

The hold-out method is good to use when you have a very large dataset, you're on a time crunch, or you are starting to build an initial model in your data science project.

K fold cross validation

K-fold cross validation is one way to improve th
model does not depend on the way we picked t
and the holdout method is repeated k number

1. Randomly split your entire dataset into k numb
2. For each fold in your dataset, build your mode
   for kth fold
3. Repeat this until each of the k-folds has serve
4. The average of your k recorded accuracy is c
   metric for the model.

Because it ensures that every observation from
set, this method generally results in a less biase
we have limited input data.

The disadvantage of this method is that the trai
takes k times as much computation to make ar

Stratified K Fold Cross Validation

Using K Fold on a classification problem can be tricky. Since we are randomly shuffling the data and then dividing it into folds, chances are we may get highly imbalanced folds which may cause our training to be biased. For example, let us somehow get a fold that has majority belonging to one class(say positive) and only a few as negative class. This will certainly ruin our training and to avoid this we make stratified folds using stratification.

Stratification is the process of rearranging the data so as to ensure that each fold is a good representative of the whole. For example, in a binary classification problem where each class comprises of 50% of the data, it is best to arrange the data such that in every fold, each class comprises of about half the instances.

Exhaustive Methods

Exhaustive cross validation methods and test on all possible ways to divide the original sample into a training and a validation set.

Leave-P-Out cross validation

When using this exhaustive method, we take p number of points out from the total number of data points in the dataset(say n). While training the model we train it on these (n – p) data points and test the model on p data points. We repeat this process for all the possible combinations of p from the original dataset. Then to get the final accuracy, we average the accuracies from all these iterations.

This is an exhaustive method as we train the model on every possible combination of data points. Remember if we choose a higher value for p, then the number of combinations will be more and we can say the method gets a lot more exhaustive.

Leave-one-out cross validation

This is a simple variation of Leave-P-Out cross validation and the value of p is set as one. This makes the method much less exhaustive as now for n data points

What is Rolling Cross Valida

For time-series data the above-mentioned met
reasons as to why this is not an ideal way to go:

1. Shuffling the data messes up the time section
2. Using cross-validation, there is a chance that
   golden rule in time series i.e. "peaking in the fu

Keeping these points in mind we perform cross

1. We create the fold (or subsets) in a forward-c
2. Suppose we have a time series for stock prices
   The folds would be created like:

```
iteration 1: training [1], test [2]
iteration 2: training [1 2], test [3]
iteration 3: training [1 2 3], test [4]
iteration 4: training [1 2 3 4], test [5]
iteration 5: training [1 2 3 4 5], test [6]
.
.
.
iteration n: training [1 2 3 ….. n-1], test [n]
```

Here as we can see in the first iteration, we train on the data of the first year and then test it on 2nd year. Similarly in the next iteration, we train the on the data of first and second year and then test on the third year of data.

Note: It is not necessary to divide the data into years, I simply took this example to make it more understandable and easy.

FAQs

1. What is the purpose of cross validation?

The **purpose of cross–validation** is to test the ability of a machine learning model to predict new data. It is also used to flag problems like overfitting or selection bias and gives insights on how the model will generalize to an independent dataset.

2. How do you explain cross validation?

Cross-validation is a statistical method used to estimate the performance (or accuracy) of machine learning models. It is used to protect against overfitting in a pred
limited. In cross-validation, you make a fixed nu
and then average the overall error estimate.

3. What are the types of cross validation?

The 4 Types of Cross Validation in Machine Lear

- Holdout Method
- K-Fold Cross-Validation
- Stratified K-Fold Cross-Validation
- Leave-P-Out Cross-Validation

4. What is cross validation and why we need it

Cross-Validation is a very useful technique to a
cases where you need to mitigate overfitting. It
the sense that which parameters will result in th

5. Does cross validation reduce Overfitting?

Cross-validation is a procedure that is used to avoid overfitting and estimate the skill of the model on new data. There are common tactics that you can use to select the value of k for your dataset.

This brings us to the end of this article where we learned about cross validation and some of its variants. To get a in-depth experience and knowledge about machine learning, take the free course from the great learning academy. Click the banner to know more.

Sharing is caring:

G

Great Learning's Blog covers the latest deve
build rewarding careers. You'll find career gu
the fast-ch

Recommended for you

Real-time Face detection | Face Mask Detection using OpenCV

Face Detection using Viola Jones Algorithm

Random forest Algorithm in Machine learning: An Overview

Application of Graph Theory in 2023

Label Encoding in Python – 2023

OpenCV Tutorial: A Guide to Learn OpenCV in Python

Leave a Comment

Your email address will not be published. Required fi

Type here..

Name*

Email*

Save my name, email, and website in this browser for the next time I comment.

I'm not a robot

reCAPTCHA
Privacy - Terms

Post Comment »

Free Courses

Python for Machine Learning

Data Science Foundations

Deep Learning with Python

Introduction to Cyber Security

Introduction to Digital Marketing

Java Programming

View More →

Blog Categories

Data Science

Artificial Intelligence

Career

Cybersecurity

Full Stack Development

Popular Courses

PGP In Data Science and Business Analytics

PGP In Artificial Intelligence And Machine Learning

PGP In Management

PGP In Cloud Computing

Software Engineering Course

PGP In Digital Marketing

View More →

Salary Blogs

Salary Calculator

Data Architect Salary

Cloud Engineer Salary

Software Engineer Salary

Product Manager Salary

Interview Questions

Java Interview Questions

Python Interview Questions

SQL Interview Questions

Selenium Interview Questions

Machine Learning Interview Questions

NLP Interview Questions

View More →

Quick Links

About Us

Contact Us

Privacy Policy

Terms of Use

Great Learning Careers

Get our android app

Get our ios app