Data Science Lecture Notes

Date: 18.05.022

Overfitting and Underfitting in Machine Learning

Overfitting and Underfitting are the two main problems that occur in machine learning and degrade the performance of the machine learning models.

The main goal of each machine learning model is **to generalize well**. Here **generalization** defines the ability of an ML model to provide a suitable output by adapting the given set of unknown input. It means after providing training on the dataset, it can produce reliable and accurate output. Hence, the underfitting and overfitting are the two terms that need to be checked for the performance of the model and whether the model is generalizing well or not.

Before understanding the overfitting and underfitting, let's understand some basic term that will help to understand this topic well:

- **Signal:** It refers to the true underlying pattern of the data that helps the machine learning model to learn from the data.
- **Noise:** Noise is unnecessary and irrelevant data that reduces the performance of the model.
- **Bias:** Bias is a prediction error that is introduced in the model due to oversimplifying the machine learning algorithms. Or it is the difference between the predicted values and the actual values.
- **Variance:** If the machine learning model performs well with the training dataset, but does not perform well with the test dataset, then variance occurs.

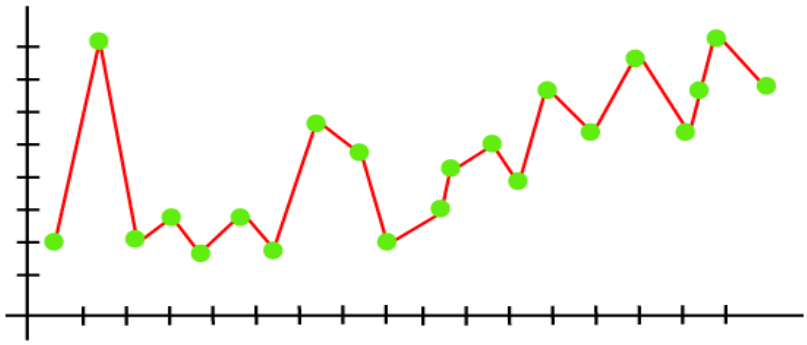
Overfitting

Overfitting occurs when our machine learning model tries to cover all the data points or more than the required data points present in the given dataset. Because of this, the model starts caching noise and inaccurate values present in the dataset, and all these factors reduce the efficiency and accuracy of the model. The over fitted model has **low bias** and **high variance.**

The chances of occurrence of overfitting increase as much we provide training to our model. It means the more we train our model, the more chances of occurring the overfitted model.

Overfitting is the main problem that occurs in supervised learning

.**Example:** The concept of the over fitting can be understood by the below graph of the linear regression output:



As we can see from the above graph, the model tries to cover all the data points present in the scatter plot. It may look efficient, but in reality, it is not so. Because the goal of the regression model to find the best fit line, but here we have not got any best fit, so, it will generate the prediction errors.

How to avoid the Overfitting in Model

Both overfitting and underfitting cause the degraded performance of the machine learning model. But the main cause is overfitting, so there are some ways by which we can reduce the occurrence of overfitting in our model.

- **Cross-Validation**
- **Training with more data**
- **Removing features**
- **Early stopping the training**
- **Regularization**
- **Ensembling**

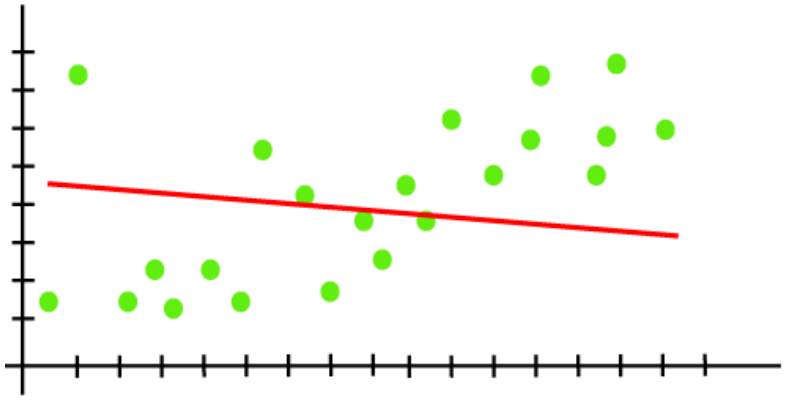
Underfitting

Underfitting occurs when our machine learning model is not able to capture the underlying trend of the data. To avoid the overfitting in the model, the fed of training data can be stopped at an early stage, due to which the model may not learn enough from the training data. As a result, it may fail to find the best fit of the dominant trend in the data.

In the case of underfitting, the model is not able to learn enough from the training data, and hence it reduces the accuracy and produces unreliable predictions.

An underfitted model has high bias and low variance.

Example: We can understand the underfitting using below output of the linear regression model:



As we can see from the above diagram, the model is unable to capture the data points present in the plot.

How to avoid underfitting:

- By increasing the training time of the model.
- By increasing the number of features.

Goodness of Fit

The "Goodness of fit" term is taken from the statistics, and the goal of the machine learning models to achieve the goodness of fit. In statistics modeling, *it defines how closely the result or predicted values match the true values of the dataset.*

The model with a good fit is between the underfitted and overfitted model, and ideally, it makes predictions with 0 errors, but in practice, it is difficult to achieve it.

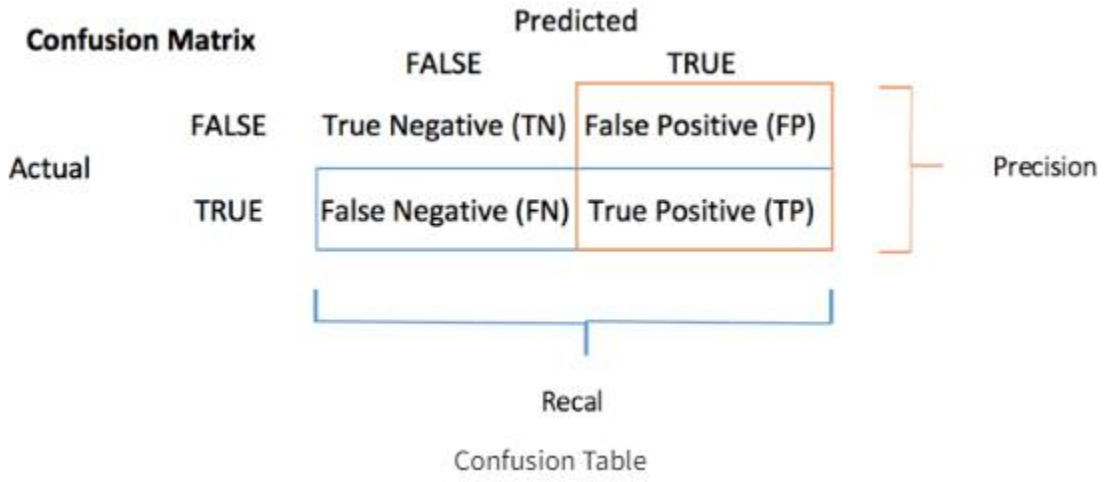
As when we train our model for a time, the errors in the training data go down, and the same happens with test data. But if we train the model for a long duration, then the performance of the model may decrease due to the overfitting, as the model also learn the noise present in the dataset. The errors in the test dataset start increasing, *so the point, just before the raising of errors, is the good point, and we can stop here for achieving a good model.*

What is Confusion Matrix?

A confusion matrix is a performance measurement technique for Machine learning classification. It is a kind of table which helps you to the know the performance of the classification model on a set of test data for that the true values are known. The term confusion matrix itself is very simple, but its related terminology can be a little confusing.

Four outcomes of the confusion matrix

The confusion matrix visualizes the accuracy of a classifier by comparing the actual and predicted classes. The binary confusion matrix is composed of squares:



Confusion Table

TP: True Positive: Predicted values correctly predicted as actual positive

FP: Predicted values incorrectly predicted an actual positive. i.e., Negative values predicted as positive

FN: False Negative: Positive values predicted as negative

TN: True Negative: Predicted values correctly predicted as an actual negative

You can compute the **accuracy test** from the confusion matrix:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Example of Confusion Matrix:
Confusion Matrix is a useful machine learning method which allows you to measure Recall, Precision, Accuracy, and AUC-ROC curve. Below given is an example to know the terms True Positive, True Negative, False Negative, and True Negative.
True Positive:
ou projected positive and its turn out to be true. For example, you had predicted that France would win the world cup, and it won.
True Negative:
When you predicted negative, and it's true. You had predicted that England would not win and it lost.
False Positive:
Your prediction is positive, and it is false.
You had predicted that England would win, but it lost.
False Negative:
Your prediction is negative, and result it is also false.
You had predicted that France would not win, but it won.
You should remember that we describe predicted values as either True or False or Positive and Negative.

Other Important Terms using a Confusion matrix
Positive Predictive Value(PVV): This is very much near to precision. One significant difference between the two-term is that PVV considers prevalence. In the situation where the classes are perfectly balanced, the positive predictive value is the same as precision.
Null Error Rate: This term is used to define how many times your prediction would be wrong if you can predict the majority class. You can consider it as a baseline metric to compare your classifier.
F Score: F1 score is a weighted average score of the true positive (recall) and precision.
Roc Curve: Roc curve shows the true positive rates against the false positive rate at various cut points. It also demonstrates a trade-off between sensitivity (recall and specificity or the true negative rate).
Precision: The precision metric shows the accuracy of the positive class. It measures how likely the prediction of the positive class is correct.

$$Precision = \frac{TP}{TP + FP}$$

The maximum score is 1 when the classifier perfectly classifies all the positive values. Precision alone is not very helpful because it ignores the negative class. The metric is usually paired with Recall metric. Recall is also called sensitivity or true positive rate.
Sensitivity: Sensitivity computes the ratio of positive classes correctly detected. This metric gives how good the model is to recognize a positive class.

$$Recall = \frac{TP}{TP + FN}$$

Why you need Confusion matrix?

Here are pros/benefits of using a confusion matrix:

- It shows how any classification model is confused when it makes predictions.
- Confusion matrix not only gives you insight into the errors being made by your classifier but also types of errors that are being made.
- This breakdown helps you to overcomes the limitation of using classification accuracy alone.
- Every column of the confusion matrix represents the instances of that predicted class.
- Each row of the confusion matrix represents the instances of the actual class.
- It provides insight not only the errors which are made by a classifier but also errors that are being made.