Proceedings of the 2nd International Conference on Computing Innovation and Applied Physics (CONF-CIAP 2023)
DOI: 10.54254/2753-8818/5/20230433

A Practical Significant Technic in Solving Overfitting: Regularization

Muyuan Li

Zhengzhou No.47 Middle &High School, Ping An Avenue No.6, Zhengzhou, Henan, China

muyuanli528@gmail.com

Abstract. The passage mainly discusses the solution to overfitting. Overfitting usually happens when people are training their machine learning models. When a model is overfitted, it only fits one particular dataset and misses most of the data points from another dataset. This problem affects the model's performance and makes it unable to use for its purpose. So how to solve this problem with significance and practical meaning? At the beginning of the passage, I will introduce some theoretical foundations for overfitting. Then I will define the concept of overfitting and show an example of overfitting in the machine learning model. After that, I will tell you how to pick the correct model with the testing set. Then, the passage focuses on the discussion of regularization, which is a helpful technique for solving overfitting. And I will compare the L1 and l2 regularization to help you find the suitable one.

Keywords: Machine Learning Model, Overfitting, Regularization, L1 Norm, L2 Norm.

1. Introduction

Today, more than 75% of US stock trades are placed by algorithmic trading. Computer programmers make all these trading strategies and use them by traders. And there is one trading strategy called the machine learning model, which is one of those popular strategies. We use machine learning strategies to train the model to fit the dataset and then use it to predict. That sounds useful. But one unique problem is coming with the benefits- overfitting. Overfitting often happens when people are testing their machine learning model, and the result shows that the model cannot fit the new data at all. So the model with overfitting cannot be used in practice simply because it cannot predict anything. So solving the overfitting is essential. In this passage, I will elaborate on why overfitting would happen and introduce a technique we use to solve the problem.

2. Theoretical Foundation for Overfitting

2.1. Mathematical concepts

To begin with, we need to list out a series of math knowledge as the background. And this content mainly relates to linear algebra.

First, vector space, we can define the vector space as a group of vectors combination. And the end-up vectors of addition or scalar multiplication must also be in the vector space.

DOI: 10.54254/2753-8818/5/20230433

Secondly, the matric. In machine learning, matrices and vectors are fundamental concepts. We can use vectors to represent individual variables as instances or sets of numbers. Variable sets can be represented using matrices. A matrix is simply an ordered collection of vectors in this sense. Consequently, column vectors are typically used, but when reading matrices, it is always a good idea to pay close attention to the authors' notation. Matrices are how we actually interact with data because computer screens have two dimensions [1].

Linear mappings, otherwise called linear transformation and linear function, demonstrate the correspondence between vectors in a vector space V and similar vectors in a different vector space W. In this way, linear mappings change vector spaces into others. Two characteristics must be held by transformation: The transformation of the sum of the vectors must be equivalent to adding up the transformations of each vector separately. A vector's transformation in a scaled form must be the same as taking the vector's transformation first and scaling the result.

Last but not least, the norms. We can think of the norm as a special map. Elements from the vector space V over the filed f, the F is either a real number or a complex number, and map those elements in the vector space to the positive real line. And we called the normed vector space like ||V||. And the norm satisfies the following properties: The first thing is the Triangle inequality. If you look at a vector sum of two vectors V1 and V2 from the vector space and you take the norm that will be less than or equal to the sum or plus of the norms of the individual vectors just themselves, and that is true for all the V1 and V2 in the vector space V. And the second if you look at the norm of a scalar and that is equal to the absolute value of the scalar times the norm of the vector. For the third property, if a norm of a vector evaluates to zero and that is equivalent to the vector itself must be the zero vector in the corresponding space.

2.2. Definition of overfitting

Before talking about overfitting, it is necessary to take a view of the process of machine learning. Once we have the historical data, we put them in a machine learning model as the training set. Then, to fit the best line, the computer will give us a very complex model that exactly fits all the data points. However, in most cases, overfitting occurs.

2.3. An example of overfitting

In this part, I will show a overfitting model compared with the model we want using the same dataset in Figure 1.

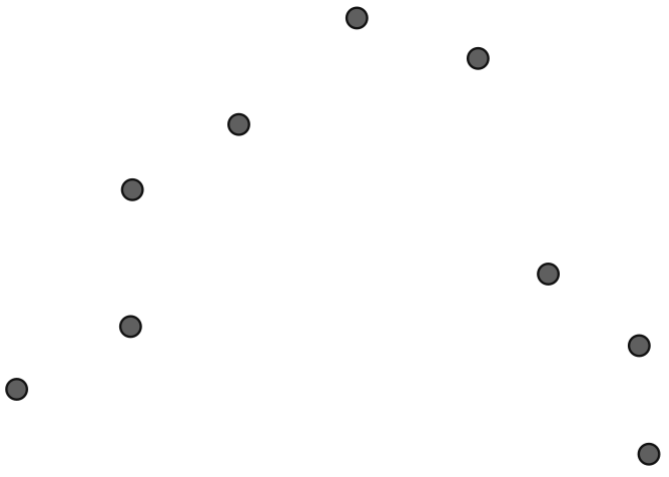

Figure 1. In this dataset, we train two models, which can indicate overfitting.

DOI: 10.54254/2753-8818/5/20230433

For humans, we can easily recognize this dataset as a parabola, which is a polynomial of degree 2. However, we can't force a computer to have that visual ability. The strategy computer use is trying many values for the degree of the polynomial and then picking the best one. So what probably happens here is that the computer gives a polynomial of degree 10. When we fit polynomials with degrees 2 and 10 into this dataset, Figure 1.2 shows these two polynomials' results.

Model 1

A polynomial of degree 2

Model 2

A polynomial of degree 10

Figure 2. Fitting these two models to the same dataset. Model 1 is a polynomial of degree 2 or a quadratic. Model 2 is a polynomial of degree 10.

As we can see, although model 1 didn't ideally cross each data point, it indicates the line well. On the contrary, model 2 catches every point of our dataset but completely misses the point. So the data is recognized as a parabola, although there is some noise, and model 2 draws an extremely complex line to catch each point in the dataset but cannot give us the essence or the trend of the dataset. So we won't choose it because it is too complex that overfitted.

To summarize, very complex models tend to overfit. And our goal is to find a model that is not too complex but can also indicate the dataset's trend, just like model 1.

The thing that is challenging is, as a human, we know that model 1 best fits the dataset. But as we mentioned, a computer cannot see. The computer can only calculate the error functions such as absolute error and square error. In this example, we will use absolute error. We take the average of the sums of the absolute values of the distances from the point to the line. Model 1's distances are small, so the error is small. However, the error of model 2 is zero because there is no distance between the line and the points. All the points exactly fall in the actual curve. So, the computer will think that model 2 is the best model among these two models. So, we need a way to tell the computer that the best model is model 1 and that model 2 is overfitting. The question is how we can do this.

3. Testing to Find the Right Model

In this part, we will discuss a way to determine whether overfitted by testing the model. To elaborate, testing a model requires us to pick a small part of points in the dataset and put them into the model, waiting to be chosen. In other words, we use this set of data points to test the model's performance. And this set of data is called the testing set (usually 10% of all the data). After using the training set to train the model, we use the testing set to evaluate the model. When testing a model, we can make sure that the model will do well in acting on unseen data instead of just memorizing the training set, which is what we call overfitting.

Now let's go back to our data and model with the testing set. Notice that the real problem of model 2 is not it doesn't fit the data but doesn't generalize well to new data. In another world, when some new point appears, model 2 is unreliable for making a good prediction because the model merely memorized the entire dataset without capturing its essence.

DOI: 10.54254/2753-8818/5/20230433

In Figure 2, we have drawn two white triangles in our dataset, representing the testing set. The training set corresponds to the black circles. In the figure, we can see how these models perform with both training and testing sets. Or, say we examine the model's error in both datasets, which we will refer to as the training error and the testing error.

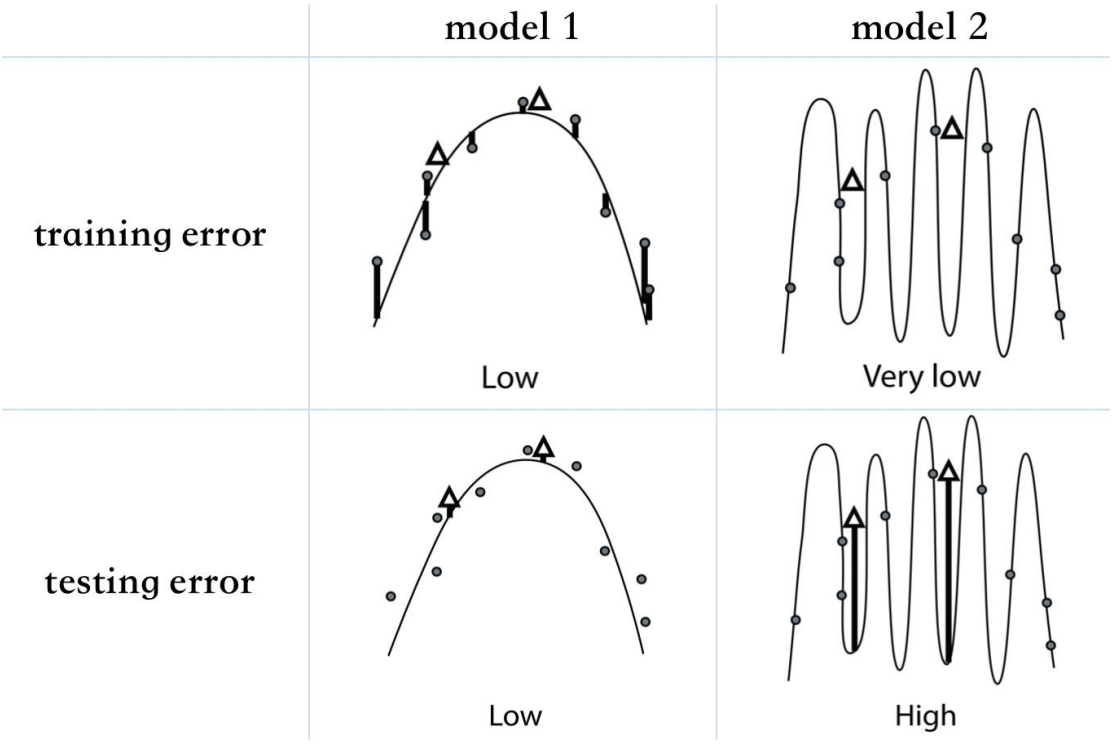

Figure 3. The comparison of training and testing errors between models 1 and 2.

We use this table to show how our model performs in the training set and the testing model. The solid circles are the training set; the white triangle is the testing set. The vertical lines from each point to the curve are the errors of each point. The error of each point is the absolute error, which is the average of these vertical lengths. From this table, we can conclude that between these two models, the best one is model 1 because it gives us a lower testing error compared with model 2.

I was comparing these two models. Model 1, with a small testing error, is good because it fits the training and the testing model well. However, model 2 produces a large testing error, which means it did a terrible job fitting the testing set yet such a good job in the training set, so we conclude that model 3 overfits.

To conclude, overfits means the model is too complex for the dataset. And it acts well in the training set, so the training error of an overfitted model is small. But it acts poorly in the testing set, indicating a large testing error. And if both errors are low, then it is a good model.

What can we do when choosing the right model among many models? The model complexity graph.

4. Another Useful Way To Solve Overfitting: Regularization

4.1. Basic information on regularization

In this part, we discuss a useful technique to solve the overfitting in the models. And this method does not require us to train many models and then find which one is the best. Instead, we train the model once because we will train the model's performance and reduce the model's complexity simultaneously. The essence of doing this is that we need to measure performance and complexity simultaneously [2].

DOI: 10.54254/2753-8818/5/20230433

How do we measure the complexity of a model? L1 and L2 norm

L1: The sum of the absolute values of the coefficients

L2: The sum of the squares of the coefficients

These coefficients do not include bias because this number is not associated with the complexity of the model.

Let's take the example from the previous model1 and model 2. Assuming these two models' formulas are the following:

Model 1: $y = -x^2 + 6x - 2$

Model 2:$y = x^9 + 4x^8 - 9x^7 + 3x^6 - 14x^5 - 2x^4 - 9x^3 + x^2 + 6x + 10$

The L1 and L2 norms are calculated as follows:

L1 norm:

Model 1: |-1|+|6| = 7

Model 2: |1|+|4|+|-9|+|3|+|-14|+|-2|+|-9|+|1|+|6| = 49

L2 norm:

Model 1: $(-1)^2 + 6^2 = 37$

Model 2:$1^2 + 4^2 + (-9)^2 + 3^2 + (-14)^2 + (-2)^2 + (-9)^2 + 1^2 + 6^2 = 425$

So now, we have two measures for models. Let's define them again.

Regression error: A measure of the quality of the model. It can be absolute or square errors.

Regularization term: A measure of the complexity of the model. It can be the L1 and L2 norm of the model.

So the sum of regression error and the regularization term is what we want to minimize to find a suitable model.

Error = Regression error + Regularization term

We can also divide the regularization into two terms depending on what norm we use. For example, if we train our regression model using the L1 norm, the model is called lasso regression. So the error function follows:

Lasso regression error = Regression error + L1 norm

And if we train the model using the L2 norm, it is called ridge regression. The error function follows:

Ridge regression error = Regression error + L2 norm

4.2. Comparison between L1 and L2

We need to understand the effects of L1 and L2 regularization on the model's coefficients to choose the right regularisation term. When we add regularization and train the model again, we end up with a simpler model. If we use the L1 norm, we end up with a model with fewer coefficients because the L1 norm turns some of the coefficients into zero. If we use the L2 norm, we end up with a model with smaller coefficients because L2 shrink all the coefficients but rarely turns them into zero. Thus, depending on what kind of equation we want, we can decide between using L1 and L2 regularization. In other words, L1 regularization estimates the median of the dataset, and L2 regularization attempts to estimate the mean of the data to evade overfitting. And another point that needs to notice, because it is a square of weight, the closed form solution for L2 exists.On the other hand, because L1 is a non-differentiable function and contains an absolute value, it does not have a closed-form solution [3].

L1 is computationally more expensive, cannot be solved using matrix measurement, and heavily relies on approximations, as stated above.

Although L2 regularization incurs higher computational costs, it is probably more accurate in all circumstances.

5. Conclusion

To conclude, overfitting is a mistake that could happen when people are training a machine learning model. Overfitting occurs when the model tailors a particular dataset and cannot generalize or predict

DOI: 10.54254/2753-8818/5/20230433

another dataset. We use a testing set to compare models after training and find the fitted model. However, this way is not high-efficient. Then we find a way named regularization, which uses L1 and L2 norms in the error functions to simultaneously calculate the models' performance and complexity.

References

[1] *Catbug88@home: ~$*. Introduction to Linear Algebra for Applied Machine Learning with Python. (n.d.). Retrieved October 23, 2022, from https://pabloinsente.github.io/intro-linear-algebra.

[2] Serrano, L. G., & Thrun, S. (2021). *Grokking Machine Learning*. Manning Publications Co.

[3] Tyagi, N. (n.d.). *L2 vs L1 regularization in machine learning: Ridge and Lasso regularization*. L2 vs L1 Regularization in Machine Learning | Ridge and Lasso Regularization. Retrieved October 23, 2022, from https://www.analyticssteps.com/blogs/l2-and-l1-regularization-machine-learning