Universidad Politécnica de Yucatán

Machine Learning

“Solution to most common problems in ML”

Victor Alejandro Ortiz Santiago

Student:

Enrique Arturo Emmanuel Chi Góngora

UNIVERSIDAD
POLITÉCNICA
DE YUCATÁN

BIS

Solution to most common problems in ML

Overfitting & Underfitting

Overfitting and underfitting are essential concepts in machine learning that describe how well a model generalizes from the training data to new, unseen data. Overfitting happens when a model is overly complex and fits the training data so closely that it even captures noise and randomness, resulting in excellent performance on the training dataset but poor performance on new data. On the other hand, underfitting occurs when a model is too simplistic and fails to capture the underlying patterns in the training data, leading to mediocre or poor performance on both the training and test datasets. Overfitting is often caused by using a model with too many parameters or insufficient data, while underfitting arises from using a model that is too simple or ignoring critical features. To mitigate these issues, one can adjust model complexity, increase the amount of data, fine-tune hyperparameters, and ensure that relevant features are considered during training. Striking the right balance between complexity and generalization is crucial in building effective machine learning models.

Characteristics of outliers

Outliers are unusual data points that are very different from most of the other data. They can make the data look strange, especially when there are just a few of them. Outliers can happen because of mistakes when collecting data or because of rare events. When they're in the data, they can mess up the average and other calculations. Imagine you're counting how tall people are, and there's one person who's incredibly tall – that's an outlier. It can make the average height seem much higher than it really is. Finding outliers is important, but it can be tricky because sometimes what seems like an outlier is just a very unusual but valid data point. We use some math tricks to spot outliers, like looking at how far away a data point is from the average. Once we find them, we might remove them from the data or treat them differently in our analysis to get more accurate results. So, outliers are data points that stand out, and dealing with them helps us get better answers from our data.

The most common solutions for overfitting, underfitting and presence of outliers in datasets

When dealing with overfitting, a common approach is to obtain more training data to help the model generalize better. Alternatively, simplifying the model's complexity can mitigate overfitting. Regularization techniques like L1 and L2 can be applied to penalize complex models. Monitoring the model's performance during training and stopping early if it starts to overfit is another effective strategy. Feature selection by choosing only the most relevant features can reduce overfitting, and cross-validation methods like k-fold cross-validation can be used to assess and prevent overfitting.

For underfitting, it's essential to consider more complex machine learning models that can better capture data patterns. Feature engineering can improve feature representation by adding relevant features or transformations. In some cases, increasing the training data size may help an underfit model learn better. If excessive regularization is causing underfitting, reducing its strength can be beneficial. Fine-tuning hyperparameters is also valuable for addressing underfitting issues.

When dealing with the presence of outliers, it's crucial to first detect them using statistical techniques such as standard deviation, z-scores, interquartile range (IQR), or percentiles. Once identified, a decision needs to be made regarding their authenticity; if they are data errors, removal or correction is appropriate. Data transformations, such as taking logarithms, can reduce the impact of outliers. Using machine learning algorithms that are less sensitive to outliers, like decision trees or random forests, is another strategy. Leveraging domain knowledge is essential to assess the significance of outliers and determine their effect on the problem at hand.

Dimensionality problem

The dimensionality problem, also known as the "curse of dimensionality" in machine learning, happens when you have lots of things (called features or dimensions) in your data. When this occurs, it makes building good machine learning models tricky and requires a lot of computer power. Imagine having too many ingredients in a recipe; it can get confusing and not necessarily make the dish taste better. As the number of features increases, you need more and more data to make sense of it all, which can be challenging to gather. This curse also makes it hard to group similar things together when there are too many dimensions. To solve this problem, we use techniques like reducing the number of dimensions, using deep learning, and different ways to measure the similarity between things. These methods help make machine learning more manageable and accurate, even when dealing with lots of features.

The dimensionality reduction process

Dimensionality reduction is a crucial process in data analysis, aiming to simplify complex datasets by reducing the number of features or dimensions while preserving essential information. Think of it as summarizing a lengthy book into a shorter version that retains the main plot points. One widely used method for dimensionality reduction is Principal Component Analysis (PCA), which identifies key directions, called principal components, where the data exhibits the most variation. These components represent the core themes of the dataset, like the main ideas in a book. PCA then retains only the most significant components, discarding less crucial ones, effectively reducing the dataset's dimensions and making it more manageable.

Another approach involves leveraging deep learning techniques, where algorithms automatically discover meaningful data patterns and keep only the most relevant ones, acting like a smart reader highlighting critical sections of the book. Additionally, cosine similarity can be employed, measuring data point similarity based on angles rather than distances, which proves valuable when working with high-dimensional data due to its resistance to the curse of dimensionality.

Dimensionality reduction offers several advantages, such as quicker computation since smaller datasets with fewer dimensions require less processing time, improved model accuracy by focusing on essential features, and enhanced data visualization in 2D or 3D, aiding in better data interpretation.

In essence, dimensionality reduction simplifies intricate datasets by selecting the most significant features or patterns, streamlining the data for more efficient analysis by machine learning models while preserving critical information.

The bias-variance trade-off

The bias-variance trade-off in machine learning represents a crucial balance that models must strike. Bias refers to the error that occurs when a model oversimplifies its predictions, essentially ignoring training data, which results in high errors both during training and when applied to new data. On the other hand, variance measures the model's sensitivity to variations in the training data. A model with high variance fits the training data extremely well but struggles when faced with new, unseen data, leading to high error rates.

The trade-off arises because increasing a model's complexity, such as adding more parameters, can help reduce bias but also increases variance. Conversely, simplifying the model reduces variance but often increases bias. Striking the right balance between bias and variance is critical for building models that generalize well to new data without overfitting (fitting the training data too closely) or underfitting (oversimplifying the model). The goal is to minimize the total error by finding an optimal balance between bias and variance, ensuring that the model performs well on both training and test data. This trade-off is fundamental in designing effective prediction models.

References

[1] Sachin (2022) Overfitting vs underfitting, Medium. Available at: https://medium.com/mlearning-ai/overfitting-vs-underfitting-6a41b3c6a9ad (Accessed: 13 September 2023).

[2] Priya, B. (2022) How to detect outliers in machine learning – 4 methods for outlier detection, freeCodeCamp.org. Available at: https://www.freecodecamp.org/news/how-to-detect-outliers-in-machine-learning/ (Accessed: 13 September 2023).

[3] (2023) ML: Underfitting and overfitting, GeeksforGeeks. Available at: https://www.geeksforgeeks.org/underfitting-and-overfitting-in-machine-learning/ (Accessed: 13 September 2023).

[4] Sriram (2023) Curse of dimensionality in machine learning: How to solve the curse?, upGrad blog. Available at: https://www.upgrad.com/blog/curse-of-dimensionality-in-machine-learning-how-to-solve-the-curse/ (Accessed: 13 September 2023).

[5] Singh, S. (2018) Understanding the bias-variance tradeoff, Medium. Available at: https://towardsdatascience.com/understanding-the-bias-variance-tradeoff-165e6942b229 (Accessed: 13 September 2023).