MACHINE LEARNING

Solution to most common problems in ML

BRENDA CASTILLO FERNANDEZ

ROBOTICS 9°B

Brenda Estefania Castillo Fernandez Robotics 9°B

Underfitting is a situation when the model is too simple for the data. More formally, the hypothesis about data distribution is wrong and too simple — for example, the data is quadratic and the model is linear. This situation is also called high bias. This means that the algorithm can do accurate predictions, but the initial assumption about the data is incorrect.

Underfitting refers to a model that can neither model the training data nor generalize to new data. An underfit machine learning model is not a suitable model and will be obvious as it will have poor performance on the training data. Underfitting is often not discussed as it is easy to detect given a good performance metric. The remedy is to move on and try alternate machine learning algorithms. Nevertheless, it does provide a good contrast to the problem of overfitting.

Techniques to Reduce Underfitting:

1. Increase model complexity.
2. Increase the number of features, performing feature engineering.
3. Remove noise from the data.
4. Increase the number of epochs or increase the duration of training to get better results.

Overfitting happens when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data. This means that the noise or random fluctuations in the training data is picked up and learned as concepts by the model. The problem is that these concepts do not apply to new data and negatively impact the models ability to generalize.

Overfitting is more likely with nonparametric and nonlinear models that have more flexibility when learning a target function. As such, many nonparametric machine learning algorithms also include parameters or techniques to limit and constrain how much detail the model learns.

For example, decision trees are a nonparametric machine learning algorithm that is very flexible and is subject to overfitting training data. This problem can be addressed by pruning a tree after it has learned in order to remove some of the detail it has picked up.

Techniques to Reduce Overfitting:

1. Increase training data.
2. Reduce model complexity.
3. Early stopping during the training phase (have an eye over the loss over the training period as soon as loss begins to increase stop training).
4. Ridge Regularization and Lasso Regularization.
5. Use dropout for neural networks to tackle overfitting.

Brenda Estefania Castillo Fernandez Robotics 9°B

Outlier is a data object that deviates significantly from the rest of the data objects and behaves in a different manner. An outlier is an object that deviates significantly from the rest of the objects. They can be caused by measurement or execution errors. The analysis of outlier data is referred to as outlier analysis or outlier mining. An outlier cannot be termed as a noise or error. Instead, they are suspected of not being generated by the same method as the rest of the data objects.

There are three types of outliers

1. Global Outliers

1. Definition: Global outliers are data points that deviate significantly from the overall distribution of a dataset.

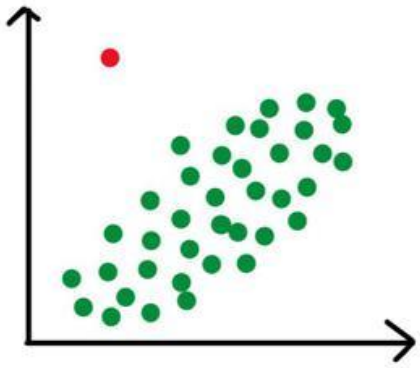
2. Causes: Errors in data collection, measurement errors, or truly unusual events can result in global outliers.

3. Impact: Global outliers can distort data analysis results and affect machine learning model performance.

4. Detection: Techniques include statistical methods (e.g., z-score, Mahalanobis distance), machine learning algorithms (e.g., isolation forest, one-class SVM), and data visualization techniques.

5. Handling: Options may include removing or correcting outliers, transforming data, or using robust methods.

6. Considerations: Carefully considering the impact of global outliers is crucial for accurate data analysis and machine learning model outcomes.



2. Collective Outliers

1. Definition: Collective outliers are groups of data points that collectively deviate significantly from the overall distribution of a dataset.

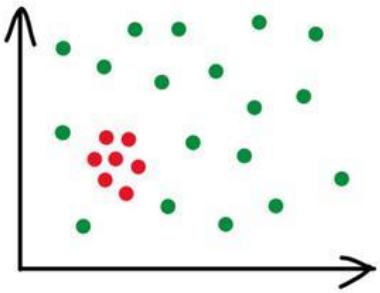
2. Characteristics: Collective outliers may not be outliers when considered individually, but as a group, they exhibit unusual behavior.

3. Detection: Techniques for detecting collective outliers include clustering algorithms, density-based methods, and subspace-based approaches.

4 Impact: Collective outliers can represent interesting patterns or anomalies in data that may require special attention or further investigation.

5. Handling: Handling collective outliers depends on the specific use case and may involve further analysis of the group behavior, identification of contributing factors, or considering contextual information.

6. Considerations: Detecting and interpreting collective outliers can be more complex than individual outliers, as the focus is on group behavior rather than individual data points. Proper understanding of the data context and domain knowledge is crucial for effective handling of collective outliers.



3. Contextual Outliers

1. Definition: Contextual outliers are data points that deviate significantly from the expected behavior within a specific context or subgroup.

2. Characteristics: Contextual outliers may not be outliers when considered in the entire dataset, but they exhibit unusual behavior within a specific context or subgroup.

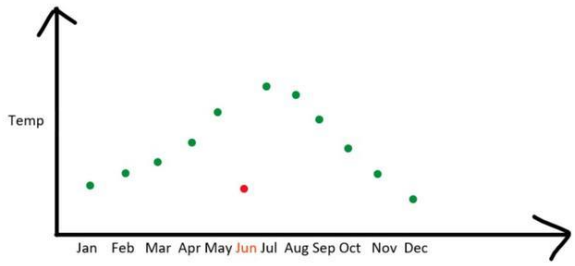
3. Detection: Techniques for detecting contextual outliers include contextual clustering, contextual anomaly detection, and context-aware machine learning approaches.

4. Contextual Information: Contextual information such as time, location, or other relevant factors are crucial in identifying contextual outliers.

5. Impact: Contextual outliers can represent unusual or anomalous behavior within a specific context, which may require further investigation or attention.

6. Handling: Handling contextual outliers may involve considering the contextual information, contextual normalization or transformation of data, or using context-specific models or algorithms.

7. Considerations: Proper understanding of the context and domain-specific knowledge is crucial for accurate detection and interpretation of contextual outliers, as they may vary based on the specific context or subgroup being considered.



Dimensionality problems or Curse of Dimensionality refers to a set of problems that arise when working with high-dimensional data. The dimension of a dataset corresponds to the number of attributes/features that exist in a dataset. A dataset with a large number of attributes, generally of the order of a hundred or more, is referred to as high dimensional data. Some of the difficulties that come with high dimensional data manifest during analyzing or visualizing the data to identify patterns, and some manifest while training machine learning models. The difficulties related to training machine learning models due to high dimensional data is referred to as ‘Curse of Dimensionality’.

Dimensionality reduction is a technique used to reduce the number of features or variables in a dataset while preserving the essential information and structure. This process helps in mitigating the curse of dimensionality, improving computational efficiency, and often enhancing the performance of machine learning models. There are two main types of dimensionality reduction: feature selection and feature extraction.

Dimensionality reduction can help to mitigate these problems by reducing the complexity of the model and improving its generalization performance. There are two main approaches to dimensionality reduction: feature selection and feature extraction.

- Feature Selection:
  Feature selection involves selecting a subset of the original features that are most relevant to the problem at hand. The goal is to reduce the dimensionality of the dataset while retaining the most important features. There are several methods for feature selection, including filter methods, wrapper methods, and embedded methods. Filter methods rank the features based on their relevance to the target variable, wrapper methods use the model performance as the criteria for selecting features, and embedded methods combine feature selection with the model training process.

- Feature Extraction:
  Feature extraction involves creating new features by combining or transforming the original features. The goal is to create a set of features that captures the essence of the original data in a lower-dimensional space. There are several methods for feature extraction, including principal component analysis (PCA), linear discriminant analysis (LDA), and t-distributed stochastic neighbor embedding (t-SNE). PCA is a popular technique that projects the original features onto a lower-dimensional space while preserving as much of the variance as possible.

Components of Dimensionality Reduction

Feature selection: In this, we try to find a subset of the original set of variables, or features, to get a smaller subset which can be used to model the problem. It usually involves three ways:

- Filter
- Wrapper
- Embedded

Feature extraction: This reduces the data in a high dimensional space to a lower dimension space, i.e. a space with lesser no. of dimensions.

Methods of Dimensionality Reduction

The various methods used for dimensionality reduction include:

Principal Component Analysis (PCA)

Linear Discriminant Analysis (LDA)

Generalized Discriminant Analysis (GDA)

Dimensionality reduction may be both linear and non-linear, depending upon the method used. The prime linear method, called Principal Component Analysis, or PCA, is discussed below.

Principal Component Analysis

This method was introduced by Karl Pearson. It works on the condition that while the data in a higher dimensional space is mapped to data in a lower dimension space, the variance of the data in the lower dimensional space should be maximum. It involves the following steps:

- Construct the covariance matrix of the data.
- Compute the eigenvectors of this matrix.
- Eigenvectors corresponding to the largest eigenvalues are used to reconstruct a large fraction of variance of the original data.

Bias: Bias refers to the error due to overly simplistic assumptions in the learning algorithm. These assumptions make the model easier to comprehend and learn but might not capture the underlying complexities of the data. It is the error due to the model's inability to represent the true relationship between input and output accurately. When a model has poor performance both on the training and testing data means high bias because of the simple model, indicating underfitting.

Variance: Variance, on the other hand, is the error due to the model's sensitivity to fluctuations in the training data. It's the variability of the model's predictions for different instances of training data. High variance occurs when a model learns the training data's noise and random fluctuations rather than the underlying pattern. As a result, the model performs well on the training data but poorly on the testing data, indicating overfitting.

References:

(N.d.). Machinelearningmastery.com. Retrieved September 15, 2023, from https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/#:~:text=Finally%2C%20you%20learned%20about%20the,poor%20generalization%20to%20other%20data

Understanding curse of Dimensionality. (2020, October 1). Great Learning Blog: Free Resources What Matters to Shape Your Career! https://www.mygreatlearning.com/blog/understanding-curse-of-dimensionality/

Introduction to dimensionality reduction. (2017, June 1). GeeksforGeeks. https://www.geeksforgeeks.org/dimensionality-reduction/

Follow, R. (2021, July 1). *Types of outliers in data mining*. GeeksforGeeks. https://www.geeksforgeeks.org/types-of-outliers-in-data-mining/

Follow, D. (2017, November 23). *ML*. GeeksforGeeks. https://www.geeksforgeeks.org/underfitting-and-overfitting-in-machine-learning/