Feature Engineering - Handling Outliers

By Binoy Patra

**Introduction:**

Outliers are data points that significantly deviate from the other observations in a dataset. These values lie outside the typical range of the data and can result from variability in the data, measurement errors, or data entry mistakes.

**Why Do Outliers Exist in a Dataset?**

1. **Measurement Errors:** Incorrect readings or data entry issues.
2. **Natural Variation:** Rare occurrences in real-world data (e.g., extreme weather events).
3. **Data Merging Issues:** Combining datasets with different scales or standards.
4. **Anomalies:** Genuine rare events that should be investigated further.

**How to Detect Outliers?**

1. **Statistical Methods:**
   * **Z-Score:** Values with z-scores greater than 3 (or less than -3) are considered outliers. (in case of Normal Distribution).
   * **IQR (Interquartile Range):** Values below Q1 - 1.5*IQR or above Q3 + 1.5*IQR. (in case of Skewed Distribution).
   * **Percentile-Based Approach:** This method identifies outliers by analyzing data beyond a specific percentile range, such as the 1st and 99th percentiles. (for any distribution)
2. **Visualization Methods:**
   * **Boxplot:** Highlights points outside the whiskers.
   * **Scatter Plot:** Reveals isolated data points.
   * **Histogram:** Shows unusual peaks or gaps.
3. **Domain Knowledge:** Leverage industry-specific thresholds to identify unexpected values.

**Effects of Outliers on Analysis and Machine Learning Models**

1. **Inaccurate Results:** Skews mean, standard deviation, and other statistics.
2. **Poor Model Performance:**
   * Can lead to overfitting in sensitive algorithms like linear regression and KNN.
   * Distorts relationships between variables.
3. **Bias in Decision-Making:** Outliers may lead to misleading conclusions.
4. **Inefficient Resource Use:** May result in increased computation time and complexity.

**Why Remove Outliers?**

1. To enhance data quality and reliability.
2. To improve the performance and accuracy of machine learning models.
3. To ensure insights are representative of most of the data.
4. To reduce noise and make visualizations cleaner and more interpretable.

**Algorithms Most Impacted by Outliers**

1. **Linear Regression**
   * Outliers can disproportionately influence the regression line, leading to incorrect predictions.
   * **Why?** Linear regression minimizes the squared errors, so large deviations (outliers) get amplified.
2. **Logistic Regression**
   * Outliers in independent variables can distort the decision boundary, affecting classification accuracy.
   * **Why?** Logistic regression assumes a linear relationship between predictors and the log-odds, which outliers can skew.
3. **K-Nearest Neighbours (KNN)**
   * Outliers can distort distance calculations, leading to misclassification.
   * **Why?** KNN relies on proximity, so extreme values can shift the neighbourhood.
4. **Support Vector Machines (SVM)**
   * Outliers can affect the placement of the hyperplane, leading to suboptimal margins.
   * **Why?** SVM aims to maximize the margin but can be skewed by extreme points.
5. **Clustering Algorithms (e.g., K-Means)**
   * Outliers can pull centroids away from the true clusters, affecting cluster assignments.
   * **Why?** K-means minimizes the sum of squared distances, making it sensitive to extreme values.
6. **Principal Component Analysis (PCA)**
   * Outliers can dominate the principal components, distorting the reduced-dimensional representation.
   * **Why?** PCA maximizes variance, which outliers can heavily influence.

**Algorithms Less Impacted by Outliers**

1. **Tree-Based Models (e.g., Decision Trees, Random Forests, Gradient Boosting)**
   * Relatively robust to outliers, as splits are based on thresholds rather than distance or variance.
   * **Why?** These models are driven by feature splits, not statistical measures like mean or variance.
2. **Ensemble Methods (e.g., XGBoost, LightGBM)**
   * Moderately robust, though extreme outliers can still influence predictions if not handled.

**Different Ways to Remove Outliers:**

**1. Statistical Methods**

* **Z-Score Method**:  
  Remove data points with z-scores greater than a chosen threshold (e.g., ±3). Use when your data follows a normal distribution, and you want to identify extreme values based on standard deviations from the mean.
  + **When to Use**: For datasets with continuous variables that are symmetric and approximately bell-shaped.
  + **Best For**: Removing extreme outliers in normally distributed data.
* **IQR Method**:  
  Remove points outside the range [Q1−1.5∗IQR,Q3+1.5∗IQR][Q1 - 1.5\*IQR, Q3 + 1.5\*IQR]. Use when your data is not normally distributed or contains skewed distributions.
  + **When to Use**: For datasets with non-normal distributions where you want a robust method based on percentiles.
  + **Best For**: Detecting and handling outliers in most datasets, as it's less sensitive to non-normal data.

**2. Capping and Flooring**

* Replace outliers with the maximum and minimum threshold values (e.g., cap at Q3+1.5∗IQRQ3 + 1.5\*IQR and floor at Q1−1.5∗IQRQ1 - 1.5\*IQR).
* Use when you want to retain data points but limit their influence.
* **When to Use**: For datasets where outliers might still carry useful information, and you want to reduce their impact without removing them entirely.
* **Best For**: Financial data, where extreme values could still be relevant for analysis.

**3. Winsorization**

* Use to limit extreme values to a defined range without removing them, ensuring no data points are dropped.
* **When to Use**: When the dataset is small or losing data could result in significant information loss.
* **Best For**: Sensitive datasets like medical or social science data.

**4. Transformations**

* **Log Transformation**:  
  Use for right-skewed data to compress large values and stabilize variance.
  + **When to Use**: When you want to make data more normal-like or reduce the influence of large values.
* **Square Root Transformation**:  
  Use for mildly skewed data or when you want to moderate the effect of larger values.
  + **When to Use**: For datasets with moderate positive skewness.
* **Box-Cox Transformation**:  
  Use when you want a more general approach to stabilize variance and make data normal.
  + **When to Use**: For advanced modeling that requires normally distributed data.

**5. Removing Data Points**

* Use when outliers are errors, anomalies, or irrelevant to the analysis.
* **When to Use**: For large datasets where removing a few rows won’t significantly affect the analysis.
* **Best For**: Datasets with evident outlier errors (e.g., negative age or impossible values).

**General Guidelines:**

* **If data loss isn’t a concern:** Use **IQR Method** or **Z-Score Method** to remove outliers.
* **If retaining data is critical:** Use **Capping and Flooring** or **Winsorization**.
* **For skewed distributions:** Use **Transformations**.
* **For error removal:** Directly drop the rows.

**5 Key point to Remember:**

**1. Understand the Nature of Your Data**

* Ensure that you know the distribution of your data (normal or skewed). Use Z-Score for normal data and IQR for skewed data. Outliers should be identified based on the data type.

**2. Domain Knowledge Matters**

* In some domains (e.g., finance or medical data), outliers may represent important information. Removing them blindly can lead to data loss, so always consider the context before removing outliers.

**3. Choose the Right Method**

* Select the appropriate method based on your data distribution. Z-Score is for normally distributed data, IQR is better for skewed data, and Percentile-based methods are suitable for mixed distributions.

**4. Evaluate Impact on the Model**

* After removing outliers, evaluate how the changes affect the performance of your model. Removing outliers can improve or degrade model accuracy depending on the algorithm used.

**5. Document and Reproduce**

* Always document the outlier removal process (method, thresholds, impact) to ensure transparency and reproducibility, especially for future reference or collaboration.