CytoAutoCluster

Overview

We are developing a project to study mass cytometry in cells. After analyzing various datasets, we have finalized our choice with the Levine_32dim dataset and initiated our exploration. Our analysis of the Levine_32dim dataset aims to understand its structure, examine feature distributions, and identify relationships among variables. This in-depth exploration employs various statistical techniques and visualizations, such as histograms, box plots, t-SNE, and PCA. Our goal is to extract insights from the data, evaluate its quality, and prepare for future modelling efforts. Gaining a clear understanding of the dataset's characteristics will guide our selection of analytical methods and strategies for subsequent machine learning applications.

1. Selecting the Right Dataset

Before diving into the analysis, it is crucial to explore and select the right dataset that aligns with the objectives of your analysis. This involves reviewing the dataset's documentation, understanding its context, and examining key aspects such as the number of samples, feature types, and potential biases. Conducting exploratory data analysis (EDA) at this stage helps identify whether the dataset is suitable for your intended analysis or if alternative datasets should be considered.

2. Null vs Non-Null Values

A bar plot is generated to visualize the counts of null and non-null values across each column. This analysis is crucial for identifying missing data and planning for imputation or removal strategies.

3. Class Label Distribution

A bar plot illustrates the distribution of class labels in the dataset. This visualization is essential for understanding class balance, which can impact model performance and selection strategies.

4. Histograms of Numerical Features

Histograms are plotted for each numerical feature to assess their distributions. This step helps identify patterns, such as normality, skewness, and potential outliers, which are important for data preprocessing and feature engineering.

5. Feature Distribution Comparison Using Histograms and KDE

This section compares the distributions of selected features using both histograms and Kernel Density Estimation (KDE) plots. This dual approach provides insights into the data's distribution shapes and overlaps, which can inform model selection and evaluation.

6. Box Plots and Count Plots

Box plots visualize the distribution of numerical features, highlighting the median, quartiles, and potential outliers. Count plots for categorical features reveal their frequency distribution, helping to identify dominant categories and patterns in the data.

7. Correlation Matrix

A heatmap of the correlation matrix is created to identify relationships between features. This analysis is useful for feature selection, as highly correlated features may lead to multicollinearity issues in modelling.

8. Skewness Analysis

The skewness of each feature is calculated and categorized (e.g., symmetric, positively skewed, negatively skewed) to understand the asymmetry of the data distribution. This analysis helps determine if transformations (like log or square root) are needed for normalization.

9. Kurtosis Analysis

Kurtosis is calculated for each feature, categorizing their distribution shapes (e.g., normal, heavy-tailed, light-tailed). Understanding kurtosis is important for assessing the presence of outliers and the data's overall behavior.

10. t-SNE Visualization

t-SNE is employed to reduce the dataset's dimensionality to two dimensions, allowing for a visual representation of potential clusters and patterns among data points. This technique is particularly useful for high-dimensional data, revealing structure that may not be apparent in the original feature space.

11. PCA Visualization

Principal Component Analysis (PCA) is conducted to reduce dimensionality while preserving variance. Visualizations are created for both 2D and 3D representations, helping to identify significant components that capture the majority of variance in the data.

12. Variance and Cumulative Proportion Analysis

The explained variance and cumulative proportion for the principal components are calculated and summarized. This analysis helps assess the significance of each component and determines how many components are necessary to retain a desired level of variance in the dataset.

13. **Binary Mask and Corrupted Data Analysis**:

- Apply a binary mask and generate a corrupted DataFrame by using the formula (x.values * (1 - m) + x_shuffled.values * m). This synthetic corruption aids in robustness testing.
- Create a new mask using (mask_new = 1 * (data_filtered != data_corrupted)) to highlight differences between original and corrupted data.

14. Data Splitting: Split the labeled dataset into training and testing subsets with a 70-30 split. This prepares the data for model training and evaluation.

15. Logistic Regression and XGBoost: Apply these machine learning models to establish a baseline and improve performance with more complex models. Logistic regression offers interpretability, while XGBoost provides robust performance for classification tasks.

16. **Data Encoding**: Apply encoding for categorical features to ensure compatibility with models, especially tree-based algorithms like XGBoost.