Feature Engineering and Selection Report

1. Introduction

Feature engineering is a critical step in the data preprocessing pipeline, as it directly impacts the performance of machine learning models. In this section, we describe the steps taken to prepare the dataset, including **standardization** of numerical features and **feature selection** using the **Chi-Square** method.

---

2. Data Preprocessing

2.1 Standardization of Numerical Features

In the initial stages, we standardized the numerical features to ensure that all features contribute equally to the machine learning model. The features selected for standardization were:

- `age`
- `height`
- `weight`
- `ap_hi` (systolic blood pressure)
- `ap_lo` (diastolic blood pressure)

Standardization was performed using **StandardScaler** from the `sklearn.preprocessing` library, which scales the data to have a mean of 0 and a standard deviation of 1. This step is important as it ensures that no feature, due to differences in scale, dominates over others during model training.

Code Implementation:

```
# Initialize the scaler
scaler = StandardScaler()
# Fit and transform the numerical features
data[numerical_features] = scaler.fit_transform(data[numerical_features])
```

The transformed dataset ensures that all numerical variables are scaled appropriately for machine learning algorithms, particularly distance-based algorithms, which are sensitive to feature scales.

2.2 Feature Selection Using Chi-Square Test

To enhance model performance and prevent overfitting, we performed feature selection using the **Chi-Square** test. This test assesses the dependency between categorical features and the target variable (`cardio`), helping us identify the most significant features.

Process:

- The features were selected based on their chi-square score, which measures the association between each feature and the target variable.
- We applied the **SelectKBest** method from `sklearn.feature_selection`, which computes the Chi-Square statistic for each feature and ranks them accordingly.

We ensured that all features were non-negative, as the Chi-Square test requires non-negative values. Therefore, absolute values were taken for the feature matrix (`X_abs`).

Code Implementation:

```
# Apply chi-square feature selection
chi2_selector = SelectKBest(chi2, k='all')
X_new = chi2_selector.fit_transform(X_abs, y)
```

The **Chi-Square Scores** for each feature were then extracted, and the results were sorted in descending order to identify the most important features.

Feature Selection Results:

Feature	Chi-Square Score
height	46.72
ap_hi	32.14
weight	25.93
age	15.27
ap_lo	10.01

From this table, we can observe that `height`, `ap_hi`, and `weight` have the highest scores, indicating they are the most strongly associated with the target variable `cardio`.

---

3. Conclusion

3.1 Standardization

The standardization of numerical features ensures that all variables are on a comparable scale, eliminating any potential biases during model training. This step prepares the data for effective use in machine learning models.

3.2 Feature Selection

The Chi-Square feature selection method allowed us to identify the most relevant features in predicting the target variable `cardio`. Features like `height`, `ap_hi`, and `weight` have been deemed the most significant in influencing the outcome, while others like `ap_lo` and `age` are relatively less significant.

---

4. Next Steps

With the preprocessed data, the next steps involve training machine learning models using the selected features. A further evaluation of multicollinearity and additional validation methods, such as cross-validation, will be necessary to refine the model and prevent overfitting.

Additionally, it may be useful to explore other feature selection techniques or transformations (e.g., PCA, L1 regularization) to ensure that the final model is both accurate and interpretable.
