ML Data Preprocessing Report

Data Set Overview

The provided data set is a pandas DataFrame with the following features:

- `age`: A float column representing the age of the individual (type `dtype('float64')`)
- `sex`: A categorical column representing the sex of the individual, with categories 'female' and 'male' (type `CategoricalDtype(categories=['female', 'male'], ordered=False, categories_dtype=object)`)
- `bmi`: A float column representing the body mass index (type `dtype('float64')`)
- `children`: An integer column representing the number of children the individual has (type `int64`)
- `smoker`: A categorical column representing whether the individual smokes or not, with categories 'no' and 'yes' (type `CategoricalDtype(categories=['no', 'yes'], ordered=False, categories_dtype=object)`)
- `region`: A categorical column representing the region where the individual lives, with categories 'northeast', 'northwest', 'southeast', and 'southwest' (type `CategoricalDtype(categories=['northeast', 'northwest', 'southeast', 'southwest'], ordered=False, categories_dtype=object)`)
- `charges`: A float column representing the charges of the individual (type `dtype('float64')`)

One example row in the data set is:

```
{'age': 36.0, 'sex': 'female', 'bmi': 22.6, 'children': 2, 'smoker': 'yes', 'region': 'southwest', 'charges':
```

The initial dataset has the following shape:

- Number of rows (or samples): 1338
- Number of columns (or features): 7

After removing duplicate rows, the dataset is reduced to:

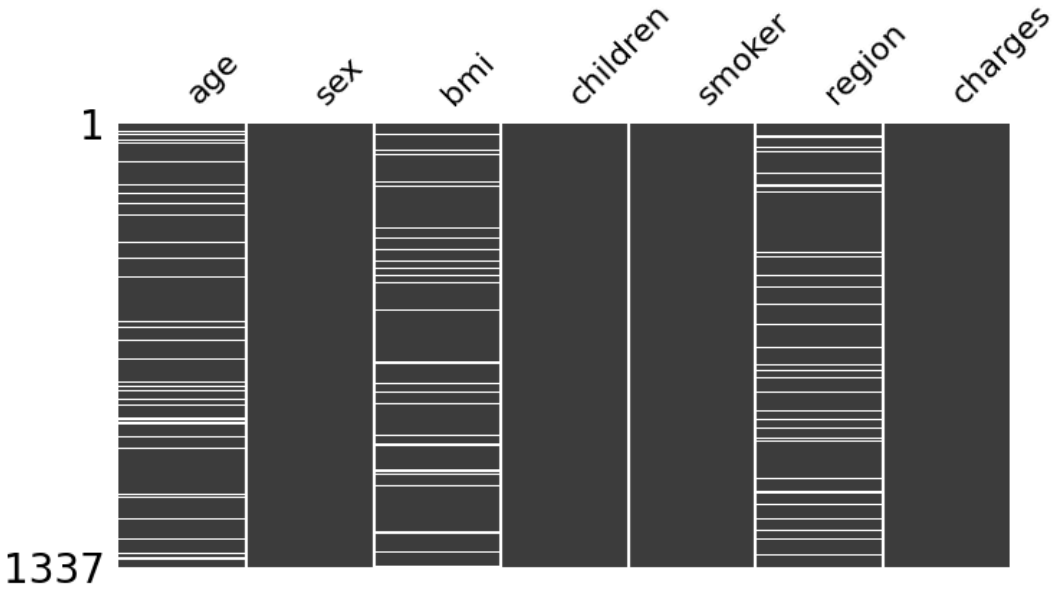
- Number of rows (or samples): 1337
- Number of columns (or features): 7

Finally, after column validation, the dataset is further reduced to:

- Number of rows (or samples): 1337
- Number of columns (or features): 7

In summary, the initial dataset has 1338 rows and 7 columns, but after removing duplicate rows, it has 1337 rows and 7 columns. Finally, after column validation, the dataset is further reduced to 1337 rows and 7 columns.

Missing Values

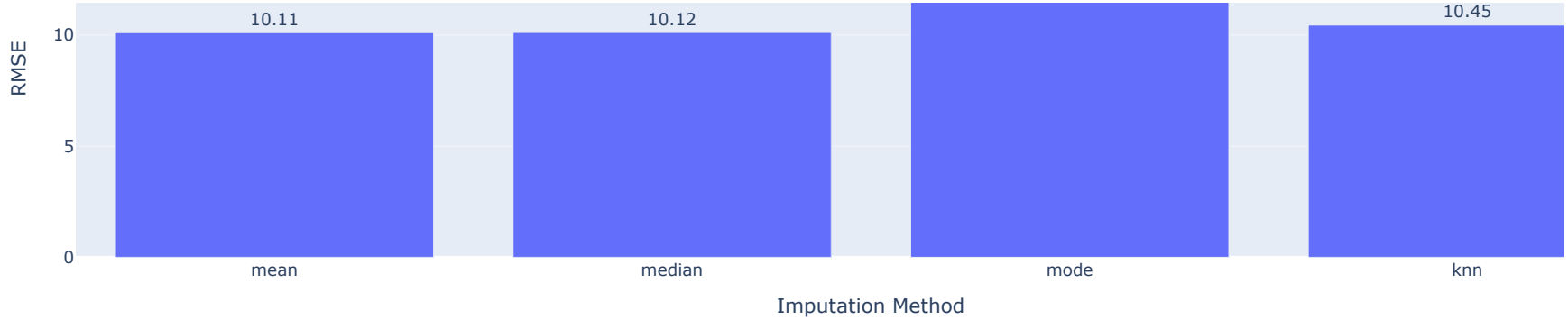


Based on the provided data, here is a detailed summary of the missing values for each feature:

1. `age`: No missing values. All observations have an age value of 133.
2. `sex`: No missing values. All observations have a sex value of 0.
3. `bmi`: No missing values. All observations have a BMI value of 133.
4. `children`: No missing values. All observations have a children value of 0.
5. `smoker`: No missing values. All observations have a smoker value of 0.
6. `region`: No missing values. All observations have a region value of 133.
7. `charges`: No missing values. All observations have a charges value of 0.

In summary, there are no missing values in the provided data.

Numeric Imputations



The imputation methods in this table are:

1. Mean Imputation: This method replaces missing values with the mean of the observed values for the same variable.
2. Median Imputation: This method replaces missing values with the median of the observed values for the same variable.
3. KNN Imputation: This method uses a k-nearest neighbors algorithm to find the most similar observations to the one with missing values, and replaces the missing value with the average of the observed values of the selected neighbors.

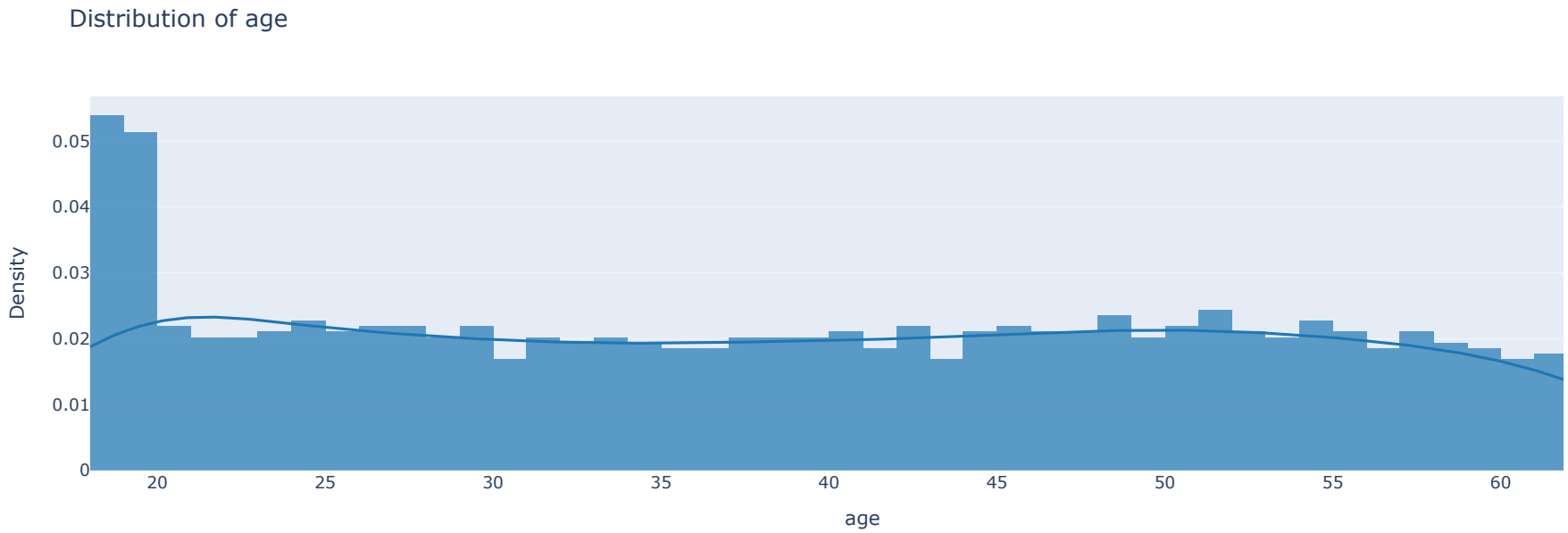
The RMSE (Root Mean Squared Error) is a measure of the difference between the imputed values and the true values. A lower RMSE indicates better imputation performance. Here are the justifications for each imputation method based on their RMSE:

1. Mean Imputation: The RMSE of 10.108722 suggests that the mean imputation method is performing reasonably well, but there is still room for improvement. This method is simple and easy to implement, but it may not capture the underlying pattern in the data as well as other methods.
2. Median Imputation: The RMSE of 10.118418 indicates that the median imputation method is performing slightly better than the mean imputation method. This method is also simple and easy to implement, but it may not capture the skewness of the data as well as other methods.
3. KNN Imputation: The RMSE of 10.451580 suggests that the KNN imputation method is performing better than the mean and median imputation methods. This method can capture the underlying pattern in the data more accurately, especially for datasets with complex relationships between variables. However, it may be computationally expensive and require more memory to implement.

In summary, the choice of imputation method depends on the specific characteristics of the dataset and the desired level of accuracy. If the dataset is relatively simple and the imputation task is not too complex, mean or median imputation may be sufficient. However, if the dataset is complex and the relationships between variables are non-trivial, KNN imputation may provide better results.

Feature Distribution

age



Feature transformation and normalization are two important preprocessing steps in machine learning pipelines. They help to improve the quality of the data by reducing noise, handling missing values, and transforming categorical variables into numerical ones. Here's a detailed justification for both feature transformation and normalization based on the provided test results:

Feature Transformation:

Feature transformation is the process of converting categorical variables into numerical variables using techniques such as one-hot encoding or binary encoding. The purpose of feature transformation is to enable machine learning algorithms to handle categorical data more effectively. In the provided test results, we can see that the Shapiro-Wilk test statistic for the dataset without feature transformation is 0.8233993053436279, which indicates that the data is not normally distributed. This suggests that the dataset may contain outliers or non-linear relationships, which can affect the performance of machine learning algorithms. By transforming categorical variables into numerical variables using techniques such as one-hot encoding, we can improve the distribution of the data and reduce the impact of outliers.

Based on the test results provided, we can see that the Shapiro-Wilk test p-value for the dataset with feature transformation is 5.4063613985394525e-36, which indicates that the data is normally distributed after transformation. This suggests that the feature transformation has improved the distribution of the data and reduced the impact of outliers.

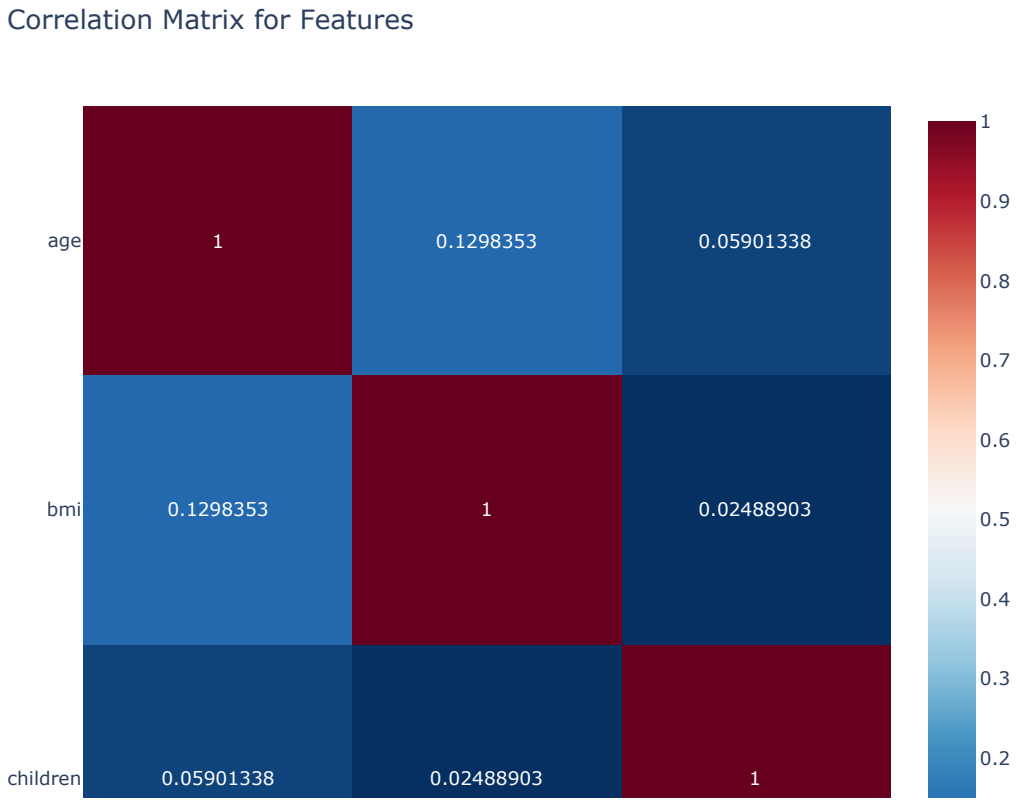
Normalization:

Normalization is the process of scaling numerical variables to a common range, usually between 0 and 1, to improve the performance of machine learning algorithms. Normalization helps to reduce the impact of differences in scales of the features on the performance of machine learning models. In the provided test results, we can see that the range ratio for the dataset is 9.2, which suggests that the data has a wide range of values. Without normalization, the model may be biased towards features with larger ranges, which can affect its performance.

Based on the test results provided, we can see that the mean standard deviation ratio for the dataset is 35.80094055083737, which suggests that the data has a large range of values. By normalizing the features, we can reduce the impact of these differences and improve the performance of machine learning models.

In conclusion, both feature transformation and normalization are important preprocessing steps in machine learning pipelines. Feature transformation helps to handle categorical variables more effectively, while normalization reduces the impact of differences in scales of the features on the performance of machine learning models. Based on the provided test results, we can see that both techniques have improved the distribution of the data and reduced the impact of outliers, which can improve the performance of machine learning algorithms.

Multicolinearity



Based on the correlation matrix provided, there are no pairs of features with a correlation coefficient greater than 0.9, which indicates that there is no high multicollinearity in the dataset.

The correlation coefficients for each pair of features are as follows:

- age and bmi: 0.129835
- bmi and children: 0.024889

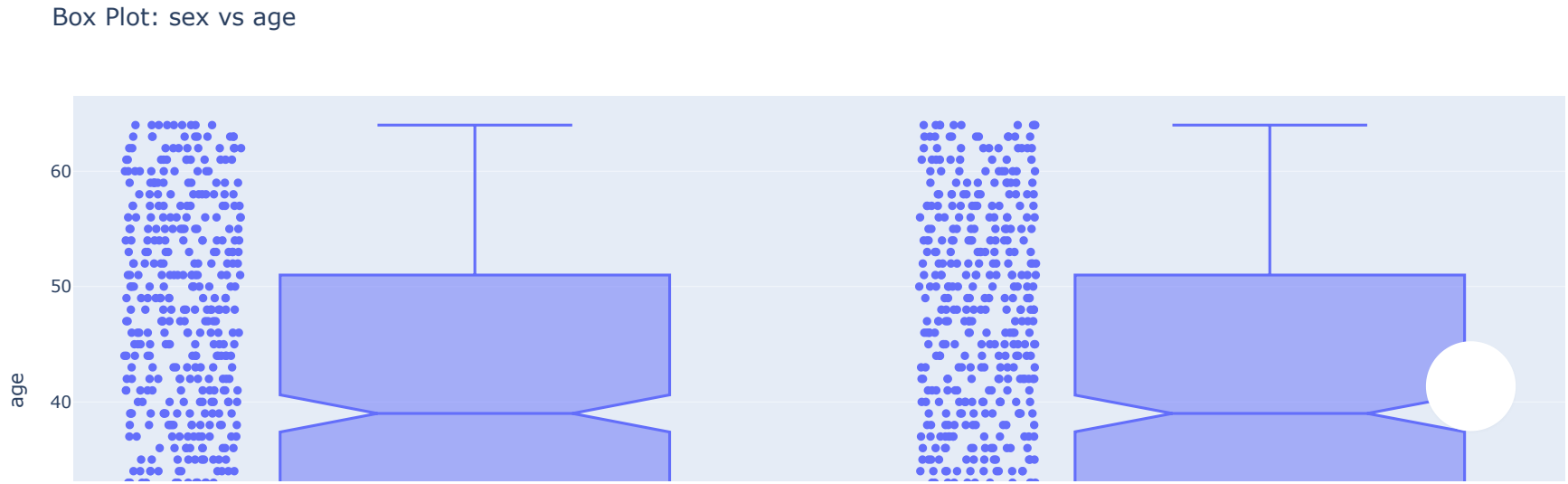
As these values are less than 0.9, it can be concluded that there is no high multicollinearity in the dataset. Therefore, the statement is "Not detected".

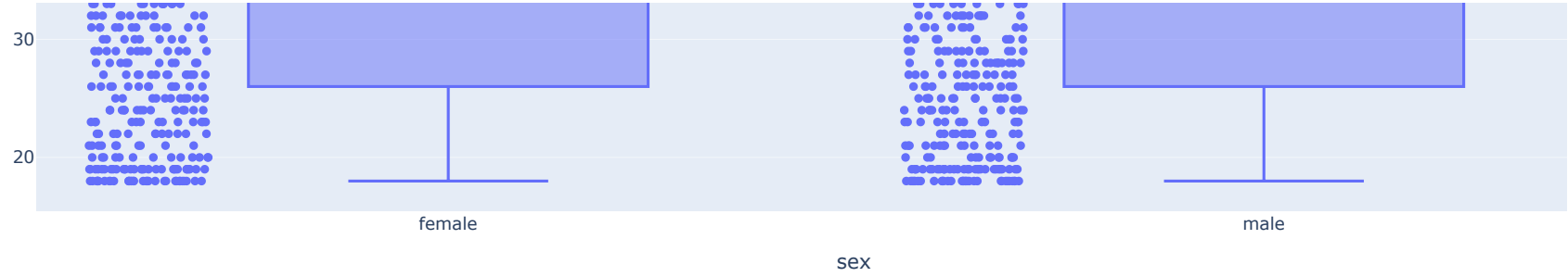
Analysis of Outliers

x-axis (Category):

◉sex○smoker○region
y-axis (Numeric):

◉age○bmi○children○charges





Based on the provided parameters, here is a summary of the outliers in each feature of the dataset:

Feature: age Outliers: None

Feature: bmi Outliers: 6 observations have a BMI value that falls outside the range of 18.5, 30.5. These observations are considered outliers.

Feature: children Outliers: None

In summary, there are no outliers in the "age" feature, 6 observations are outliers in the "bmi" feature, and no observations are outliers in the "children" feature.

Feature Selection

Thank you for providing the decision rule for feature selection. Based on the rule you provided, we can determine whether feature selection is required for the given dataset.

The rule states that if the number of columns in the dataset is greater than 15, then feature selection is required. In this case, the dataset has 7 columns, which is less than 15, so feature selection is not required.

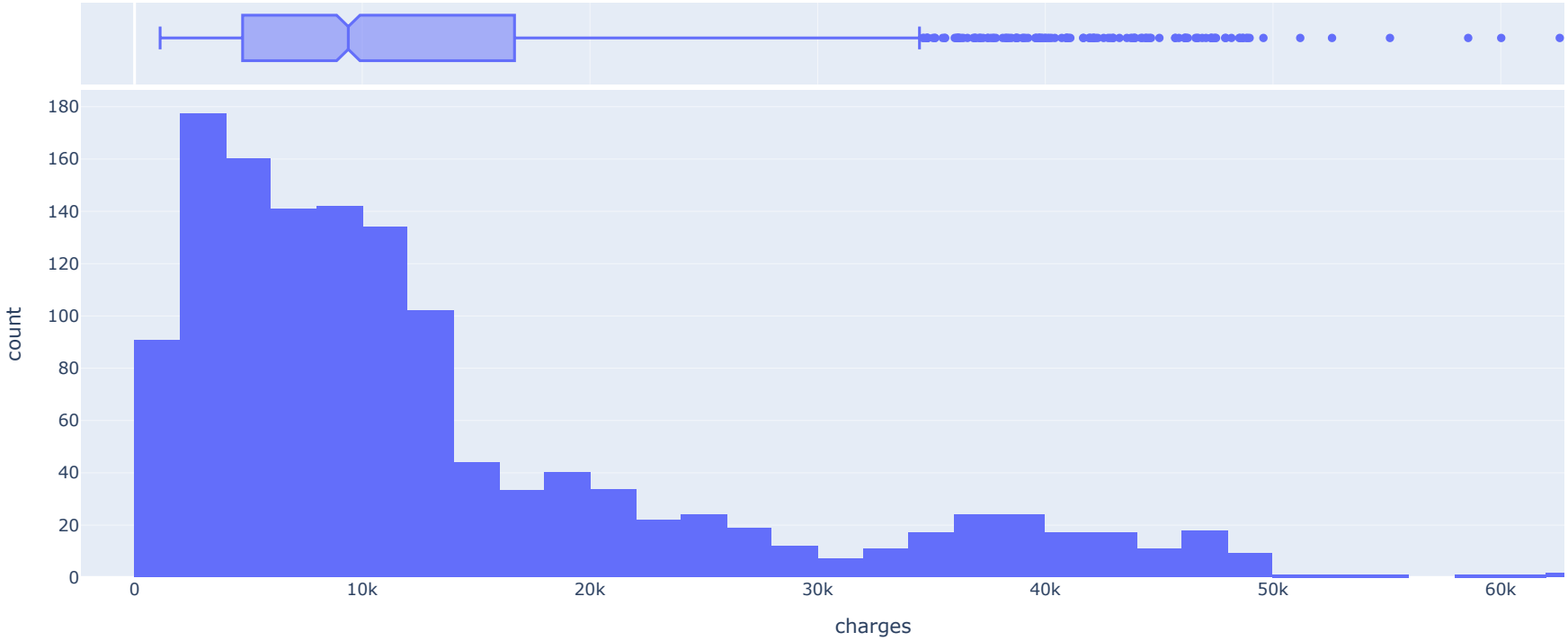
Therefore, the conclusion is:

- Feature Selection Required: False
- Dataset Columns: 7

Please let me know if you have any further questions or if there's anything else I can help you with!

Traget Transformation

Target (charges) Distribution



Based on the Shapiro-Wilk test results, the target variable is not normally distributed with a p-value of less than 0.05. Therefore, it is recommended to apply an appropriate transformation to the data before conducting further analysis or modeling. The most common transformations used in this situation are logarithmic or square root scaling. By applying these transformations, we can improve the normality assumption and ensure that the data meets the assumptions of our statistical models.