Handling missing values

For project 1, we have considered only 20 columns from the dataset to answer our SMART questions. In the second phase of the project which involves model building we have decided to include all of the 186 columns in the dataset. Since, we have not preprocessed the rest of the columns in project 1, we proceeded with the cleaning process first. Calculation of missing values in each column was done first. Missing values were replaced with NA’s for ease of understanding. We have employed multiple methods to handle the missing values. First step involves removing columns with more than twenty three percent of missing values from the dataset. This step was necessary as imputing a quarter of data would generate a reasonable amount of bias.

Next, shape of each column’s distribution was calculated to ensure whether to impute mean, median or mode in that particular column’s missing values. To achieve this, we had to loop through all the columns and generate histograms for each column. After understanding the distribution of the columns, imputation of mean was done to columns with normal (Gaussian) distribution (mean = median), median to columns with skewed distribution (if left skewed, mean < median else mean > median) and mode in categorical columns respectively. We’ve experimented with cluster specific imputation and regression based imputations as well. But they were computationally intensive and were taking a long time to run, which is why we had to drop those methods.

Feature Selection

Feature selection is of one of the most important steps before building the model. Since, we have 186 independent variables, it was of high significance for us to make sure that we have chosen appropriate features for our model. Hence, to accomplish this we have employed four feature selection methods and have chosen features from these. The methods used are, Point Biserial correlation technique, chi-square test of independence, Step-wise selection and Lasso regularization based feature importance.

Point biserial correlation method

The point-biserial correlation coefficient is a measure of association that quantifies the strength and direction of the relationship between a continuous numerical variable and a binary (dichotomous) variable. In the context of feature importance methods, the point-biserial correlation is commonly used to assess the relationship between individual continuous features and a binary target variable.

The provided code aims to evaluate the point-biserial correlation coefficients between each feature in the dataset 'g' and the binary target variable 'hospital\_death.' First, the target variable is extracted from the dataset. Then, the script iterates through each feature, calculating the point-biserial correlation coefficient only for numeric features while handling non-numeric features appropriately, marking them as 'NA.' This step ensures that the analysis focuses on the relationship between continuous numerical features and the binary target variable. The correlation coefficients are stored in the 'correlation\_with\_target' vector. Finally, a threshold, in this case, an absolute correlation value greater than 0.1, is applied to select features deemed to have a meaningful association with the target variable. The names of the selected features are stored in the 'selected\_features' vector, which can be printed or used for subsequent analyses.

This code provides a straightforward method for feature selection based on point-biserial correlation, helping to identify features with notable associations with the binary target variable. The correlation of 0.1 which is low is considered because we were aiming to capture features even with a small correlation as our’s is a hospital – patient dataset where every small things matter.