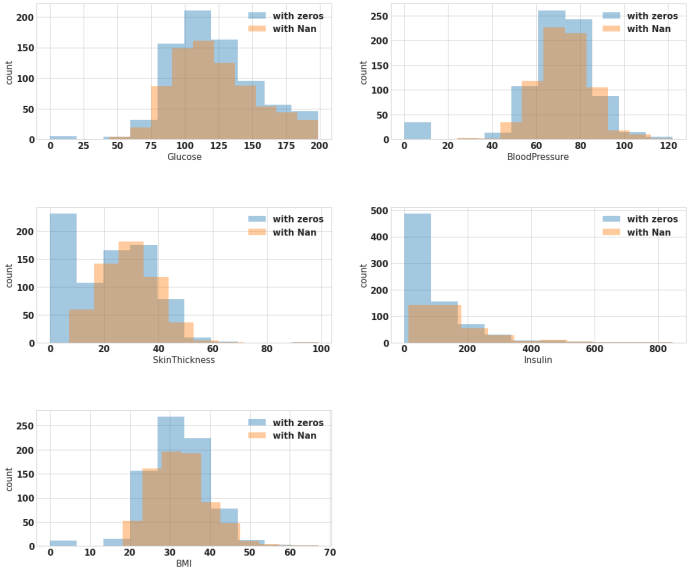
Exercise Sheet 6 – Missing Data and Feature selection

INF161 Autumn Semester 2020

Exercise 1 ***Missing Values:*** *In many of the real world machine learning problems, there are a lot of datapoints that have missing attributes. There are a few approaches when it comes to dealing with these values.*

- *Load the* ***"diabetes.csv"*** *file into a dataframe. This dataset consist of several feature variables and one target variable,* ***Outcome****. Features include the number of pregnancies the patient has had, their BMI, insulin level, age, and so on.*
- *Count the number of missing values which are denoted by* ***0*** *in columns:* ***(Glucose, BloodPressure, SkinThickness, Insulin, BMI)****.*
- *Plot the histogram of the columns mentioned above, before and after marking zero values as missing or Nan. Compare the distribution of the columns in both cases like below. Explain why it is important to use Nan instead of zero for missing values indication.*

Figure 1: Histogram of Columns with missing values



- *Split the rows of the dataset into train, validation, and test sets with corresponding ratios of 0.6, 0.2, 0.2, respectively.*
- *Replace the NaN values in the training and validation datasets with the column mean, column median, column most frequent value in training set. For each of these cases measure the performance of a K-Neighbours classifier trained on the training set, on the validation set by accuracy of classification. What method of imputation performs best on the validation set? (You can use the package sklearn.impute.SimpleImputer for this task)*
- *Impute the test set using the best performing imputation on validation set and measure the performance of the Logistic Regression model from previous step on test set. What is the value of the accuracy obtained?*

Exercise 2 ***Feature selection***

- *Load the titanic dataset into a dataframe.*
- *We want to make sure that we can predict the survival rate of a person not only from their ticket fare but from their name as well. But we can not feed the name directly to the prediction model. Therefore, some preprocessing is required so that the name can be used in prediction. Instead of name we want to extract the title of each person in the Name column. That is whether they are refered to as "Mr.", "Miss.", "Lady.", etc. Extract this pattern from the* ***"Name"*** *column and assign it to a separate column called* ***"Title"****.*
  *(**hint:** in order to extract patterns from strings, you can use Regular Expressions library or the built-in function of pandas.Series.str.extract() with a given pattern. In this scenario the required pattern starts with a capital letter after a space, is followed by one or few small letters and at the end there is a* ***"."****. So the required pattern is* `" ([A-Za-z]+)\."`*). For more information on string pattern matching read here*
- *Plot the bar plot of the number of different titles you have found.*
- *Make sure the title can either be ("Master", "Miss", "Mr", "Mrs"). Put any other title in the column as "Other".*
- *Plot the average survival rate of a person based on their title in a bar plot.*
- *Assign a number between 0 and 5 to each of the titles to make them usable for a prediction model.*
- *Split the dataset into train and validation sets with corresponding proportions of 0.8 and 0.2. Fit a K-Neighbours classifier on training set once with ticket fare and once with the pair of (Ticket Fare, Title) as input features. Compare the accuracy of predictions on the validation set.*