In-Class Exercise and Homework 1: Data Preparation in R

For all R assignment items, use the patterns described in DataScience02a.R, DataScience02b.R, and DataScience02c.R

1. Using R Data Preparation
2. Get Indian Liver Patient Dataset:  
   **url <- “http://archive.ics.uci.edu/ml/machine-learningdatabases/00225/Indian%20Liver%20Patient%20Dataset%20(ILPD).csv”**  
   **ILPD <- read.csv(url, header=FALSE, stringsAsFactors=FALSE)**
3. Get the column headers from:  
   [http://archive.ics.uci.edu/ml/datasets/ILPD+(Indian+Liver+Patient+Dataset)#](http://archive.ics.uci.edu/ml/datasets/ILPD+(Indian+Liver+Patient+Dataset))
4. Manually construct a vector of column using  
   **headers <- c(<name1>, <name2>, …)**
5. Associate names with the dataframe  
   **names(<dataframe>) <- headers**
6. Using R Data Exploration on ILPD (Indian Liver Patient Dataset)
7. Use **head(ILPD)** to view the first 6 rows.
8. Determine the **mean**, **median**, and standard deviation (**sd**) of each column.
9. What does **na.rm = TRUE** do in sd(x, **na.rm = TRUE**)?
10. Create Histograms (**hist**) for each column where possible.

Use the **plot(ILPD)** function on this data frame to present a general overview of the data. You want to see a matrix of many plots. Your efforts may be thwarted because the Gender column is not numeric. You can skip the Gender column, or you can turn the gender column into a numeric column. You might need help from a fellow student or me. Look at the plots and answer:

1. What can you say about the data?
2. How can you tell if a vector contains continuous numbers or binary data?
3. How can you tell if two vectors are correlated?
4. Using Data Preparation concepts Create examples of the following data preparation  
   processes in R
   1. Remove Outliers: c(-1, 1, 5, 1, 1, 17, -3, 1, 1, 3)
   2. Relabel: c('BS', 'MS', 'PhD', 'HS', 'Bachelors', 'Masters', 'High School', 'BS', 'MS', 'MS')
   3. Normalize: c(-1, 1, 5, 1, 1, 17, -3, 1, 1, 3)
      1. Min-Max Normalization
      2. Z-score normalization
   4. Binarize: c('Red', 'Green', 'Blue', 'Blue', 'Blue', 'Blue', 'Blue', 'Red', 'Green', 'Blue')
   5. Discretize: c(3, 4, 4, 5, 5, 5, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6, 7, 7, 7, 7, 8, 8, 9, 12, 23, 23, 25, 81)
      1. 3 Bins of equal range
      2. 3 Bins Equal of near equal amounts (Do this by hand. Writing equalization code is tricky)
5. Combine the assignment items 1, 2, and 3 into a single R file.