Part 2ai

This is a small dataset with 16 attributes for 690 rows of data. While it is identified as credit approval data, it goes on to say the attribute names and values have been changed for confidentiality purposes, meaning the dataset would be strictly for education purposes. I'm saying educational because of the diverse mix of values, and lack of values, in the dataset. For example, A2, A3, A8, A11, A14, and A15 are continuous numeric values (some appear to have floating point values). The remaining columns are discrete in nature and restricted to certain values. For example, A1 only has values 'b' or 'a' and A16 only has values '+' or '-'.

Part 2aiii

The str command is a useful function that shows a breakdown of each attribute, the type of value represented, and the different values and labels found within each attribute. For example, here we see that A1 consist of a factor with 3 different levels: "", "a", and "b". This is followed by a label for these factors for the first 10 rows in the dataset. In this case, "a" has a label of 2 and "b" has a label of "3". In the case of A2, it shows a type of num followed by the first 5 numbers rounded up.

Part 2bii

The "Min" value tells us the lowest value for this attribute in the dataset. The "1st Qu.", or first quartile, marks the upper end of the lower 25% of values for this attribute. "Median" is simply the middle point of the data in this attribute. The "Mean" is the calculated average of values in this attribute. The "3rd Qu.", or third quartile, marks the lower end of the top 25% of values of this attribute. "Max" tells us the highest value within this attribute. The "NA" field tells us how many columns have missing values in this attribute.

Part 2biii

When we run the summary command on a factor it does not return the six descriptive statistics. Rather, it returns a list of the values found in this set and the number of instances of each. For example, in credit$A1 there are 12 occurrences of "", 210 of "a", and 468 of "b". In credit$A4 there are 6 occurrences of "", 2 of "l", 519 of "u", and 163 of "y".

Part2biv

The str() command provided a general view of what each attribute was comprised of. It showed the type and a preview of the values within that attribute. The summary() command, on the other hand, provided descriptive statistics of each attribute. For example, when dealing with a number attribute it provided the min, 1st quartile, median, mean, 3rd quartile, max, and in some cases the NA values. When dealing with factors it showed the count of occurrence for each value.

Part 2ci

Discretization is the process of processing continuous numeric attributes into nominal categorical attributes algorithmically. This process can be done with various methods, both unsupervised and supervised. The overall purpose is to create discrete variables for other algorithms to use which cannot work with continuous variables. A simple example of this process would be dividing values of a column into equal-sized width breaks.

Part 2cii

In each command I used the credit$A2 data for purposes of comparing the results. I also specified 6 breaks in each command for the same reason. The first method used is the equal interval width. This method takes the number of breaks specified and splits the data into intervals of equal size. If we look at the command above, we can see each bin is approximately a width of 11. This method could be affected by outliers. The equal frequency method, on the other hand, splits the data so there are approximately the same number of values in each range. Noted above, each range, though not equal in size, does have close to the same number of values, 113+/-3. The third command uses K-means clustering. This method determines a center point for each break and ensures values within that range are closer to it's center than other breaks.

Part 2ciii

The equal interval width method is an unsupervised parametric algorithm that splits data into a specified number of equal breaks. While it is susceptible to extreme outliers, it would be useful if we had a large set of continuous variables and we need to convert this into discrete nominal variables. If we look at the car data provided in this week's tutorial, we could apply this method to the cars$city\_mpg field. In this case we could break our intervals into each width of 6 to more easily visual how many cars have an acceptable mpg rating per government guidelines. In this case there are 28 vehicles between 13-19 and 76 between 19-25. The equal frequency method creates a specified set of breaks and attempts to balance the number of values within each break, versus attempting to balance the size of the break. This method would be preferable over equal width if we needed to keep values with similar values in the same break. Looking back at our cars$city\_mpg field, in this case we could see how many vehicles have a similar mpg rating, versus having them locked into set intervals. K-means clustering determines the center value for each break and ensures every value assigned to a break is closer to it's center than any other. K-means would be the preferred method if we needed to maintain some resemblance of the original distribution. In the case of our cars$city\_mpg data this method would more reasonably associate a car's mpg rating with the appropriate break since it determines association with distance to center, versus applying set intervals or attempting to attain similar frequency counts. In this instance we can see there are over 66 vehicles with less than 21.5 mpg and only 1 that is over 47.5 mpg.

Part 2civ

As we can see from the exercise above, removing an attribute is trivial, we only need to assign it to a "NULL" value. There are several reasons we might want to remove a variable from a dataset. One example, if we are getting ready to merge two different datasets and both have an ID column, there might be conflicting values and removing the whole column and starting new might be the best move. Additionally, we might only need a subset of the data frame and decide to remove data that will not be used in our analytics.

Part 2diii

Managing missing values in your dataset is a critical part of data preprocessing. If data is missing it could disrupt the results of the algorithm we are attempting to run. While some missing values could be acceptable, it is important for us to assess this and make an educated decision on whether to leave the values as NULL, replace with the attribute mean value, or fill with some other predetermined value to meet the requirements of the analysis we are conducting.

Part 2dv

Two important reasons we want to sort our data is readability for the user and algorithm efficiency. The first of these seems obvious enough, it is much easier for a person to read a sorted list, alphabetical or numeric, versus an unsorted list. In terms of algorithm efficiency, let's discuss the difference between a merge sort and a bubble sort algorithm, and then how a binary search works. Merge sort takes a divide-and-conquer approach to sorting elements. This method is O (n log n), a combination of linear and logarithmic complexity. Bubble sort, on the other hand, is O (n^2). This quadratic complexity is since a bubble sort runs through list for each element, checking to see if it is greater than or less than the next element. As we can see sorting can be an expensive operation, so we would want to do it upfront, consider our data preprocessing, before we need to use the sorted data to ensure our analysis is occurring as quickly as possible. A great example of this is the binary search, which is O (log n), or logarithmic complexity. Given a presorted list, a binary search can quickly cut a list in half, then half again, and then again to quickly find what it is looking for. If the data were not already presorted this operation would not be possible (Sam, 2014).

Sam. (2014, December 10). *The Idiots Guide to Big O*. Retrieved from Core Java Interview Questions: http://www.corejavainterviewquestions.com/idiots-guide-big-o/

Part 2ei

When we use the plot command, our goal is visual our datasets. In this case, I have converted credit$A6 into a table with the frequency count of each value within this attribute and specified a type "h" which draws vertical lines like a histogram. I also specified the lines be blue with the "col" definition. As we can see in the plot above, there are an approximately 140 instances of "c", but less than 20 of "i". This is a great tool if we need to see how our data looks, versus reading through rows and columns. The plot command can also be turned into a step, line, and/or points plot by changing the type to either "s", "l", "p", or "b".

Part 2fi

Data preprocessing is an important first step of data mining. We cannot, or should not, proceed with data analysis until the data has been properly prepared. For example, if we have gone through the proper steps of preparing our data by properly cleaning, integrating, and transforming the data as needed we can rest easy knowing we have a complete dataset ready to be analyzed. Another advantage to data preprocessing would be fewer data redundancies, and overall better algorithm efficiency as the dataset will be streamlined for processing. There are several consequences if data preprocessing is not conducted. For instance, the dataset will likely have a lot of noise in it, erroneous data and extreme outliers, which will comprise and/or completely hinder proper algorithm functionality. Additionally, if the datasets frequently have errors in the dataset users will eventually come to not trust the data at all rendering any analysis useless.

Part 2fii

The primary difference between variable and row filters is the subset of data they affect. The variable filters were applied to an entire column of data points across the rows. An example of a variable filter is applying the equal width interval discretization method to an attribute for categorizing data into set width intervals. An example of when we would use this would be if we had an attribute of employee salaries and wanted to consolidate the salaries into equal width bins for grouping them for better visualization of the different level of salaries in the company. The row filters will affect the number of rows being selected. For example, we could specify a subset of rows we need to work with specifically, versus the entire dataset. This might be useful if we only need to analyze a sampling of an employee dataset for quality assurance purposes.