Homework #6: Segmentation

Raveena Kamal

Grading Note

This HW is worth 40 points in total. I've made notes in this document where those points were earned.

Homework tasks:

- Define a segmentation scheme for a women's apparel brand
- Gain practice with clustering techniques
  - Euclidean and Gower distance (similarly) measures
  - K-means clustering algorithm
- The apparel customer dataset contains data on customer characteristics
  - Cross-section of observations
  - We observe last year expenditures (on all products) by channel (retail and online)
  - We directly observe the customer's age and gender (direct demographics)
  - We impute Census demographics using a zip-code matching process
    - Income, white (fraction white households), college (fraction adults w/ degree)
  - Data file is: `apparel_customer_data.csv`

The variables in the dataset are:

Variable	Description
`iid`	Identifier for customer
`spend_online`	dollars spent last 12 months on online purchases
`spend_retail`	dollars spent last 12 months on retail purchases
`age`	customer age
`male`	1 = if consumer is male
`white`	proportion of households in customer zip code that are white
`college`	proportion of households in customer zip code that have college
`hh_inc`	median income of households in customer zip code ('000)

Read in the data

Q1 To begin, load the customer data into a dataframe named `DF`. Use `head()` and `summary()` to visualize the first few rows and to summarize the variables. **(1 point)**

```
DF<- read.csv("/Users/raveena/Desktop/Classroom - R/Marketing Analytics/data/apparel_customer_data_hw6.
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

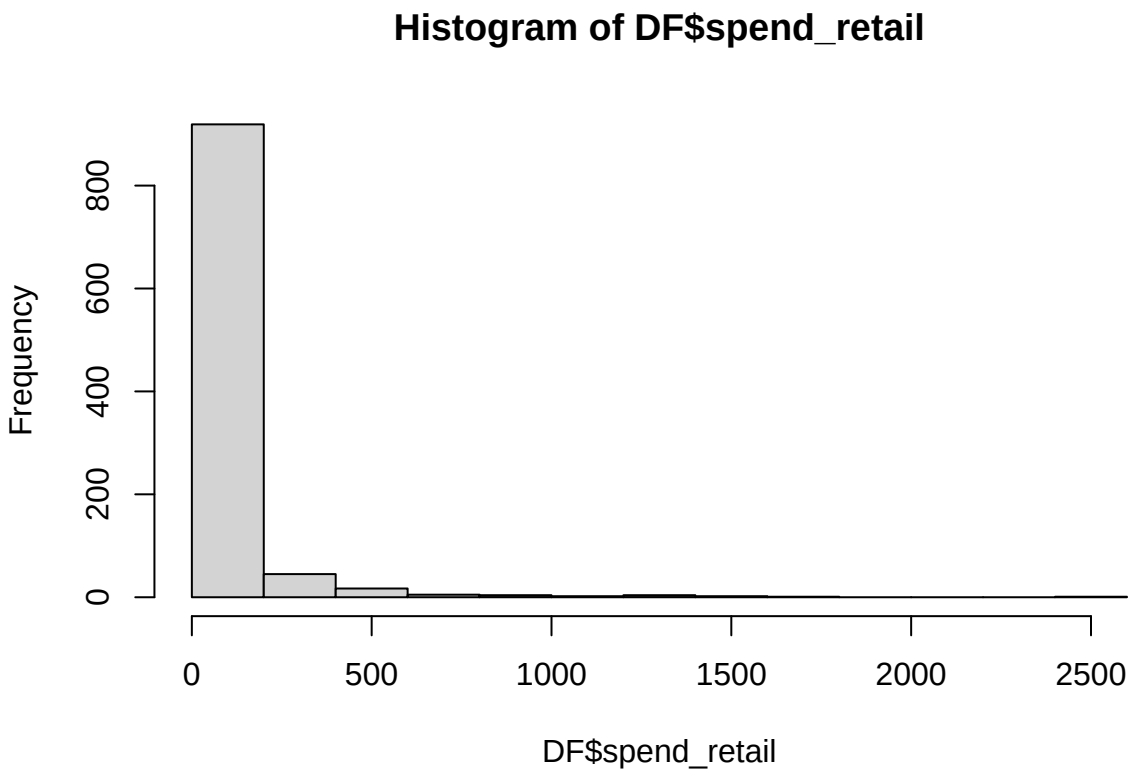
```
library(ggplot2)
summary(DF)
```

```
##       iid         spend_online      spend_retail          age
##  Min.   :   14   Min.   :   0.00   Min.   :   0.00   Min.   :18.00
##  1st Qu.: 2946   1st Qu.:   0.00   1st Qu.:   0.00   1st Qu.:33.00
##  Median : 5430   Median :  14.97   Median :  27.71   Median :41.00
##  Mean   : 5463   Mean   :  72.44   Mean   :  78.00   Mean   :40.91
##  3rd Qu.: 8110   3rd Qu.:  70.72   3rd Qu.:  78.00   3rd Qu.:49.00
##  Max.   :10589   Max.   :1985.75   Max.   :2421.91   Max.   :88.00
##      white           college           male           hh_inc
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.000   Min.   :  2.499
##  1st Qu.:0.7297   1st Qu.:0.3835   1st Qu.:0.000   1st Qu.: 59.356
##  Median :0.8550   Median :0.5580   Median :0.000   Median : 87.364
##  Mean   :0.7993   Mean   :0.5437   Mean   :0.091   Mean   : 96.254
##  3rd Qu.:0.9422   3rd Qu.:0.7136   3rd Qu.:0.000   3rd Qu.:122.602
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.000   Max.   :250.001
```

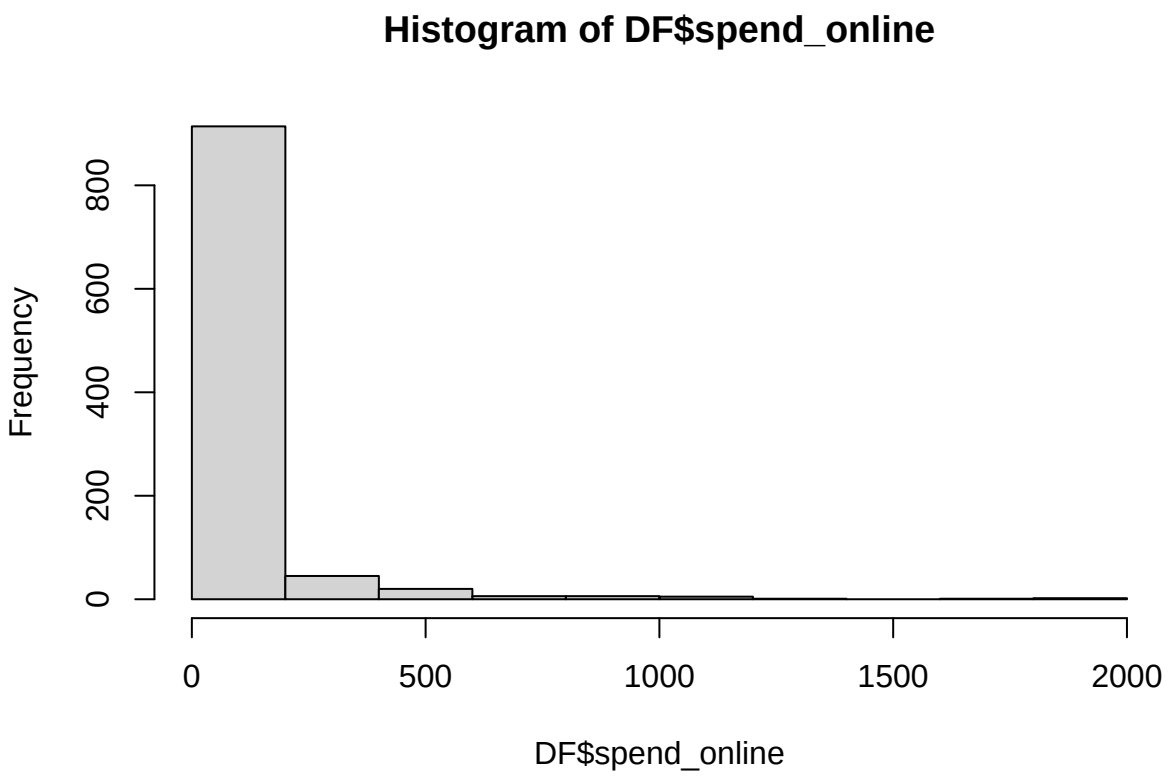
```
head(DF)
```

```
##    iid spend_online spend_retail age     white   college male  hh_inc
## 1  199       34.975        0.000  41 0.6403464 0.7028232    0 148.603
## 2 2298      220.135        0.000  41 0.2723192 0.2530814    0  25.469
## 3 9594       39.950        0.000  44 0.8428670 0.5984784    0  84.702
## 4 9542       34.975      480.355  40 0.9354839 0.5439673    0  83.125
## 5 1163       50.400        0.000  32 0.9256757 0.6632826    0 132.813
## 6 6013        0.000       46.000  40 1.0000000 0.4557109    0 128.558
```

```
hist(DF$spend_retail)
```




```
hist(DF$spend_online)
```




```
min(DF$spend_online)
```

```
## [1] 0
```

Q2: Which (continuous) variables stand out in terms of being high-skew? *(1 point)*

Answer Online Spend

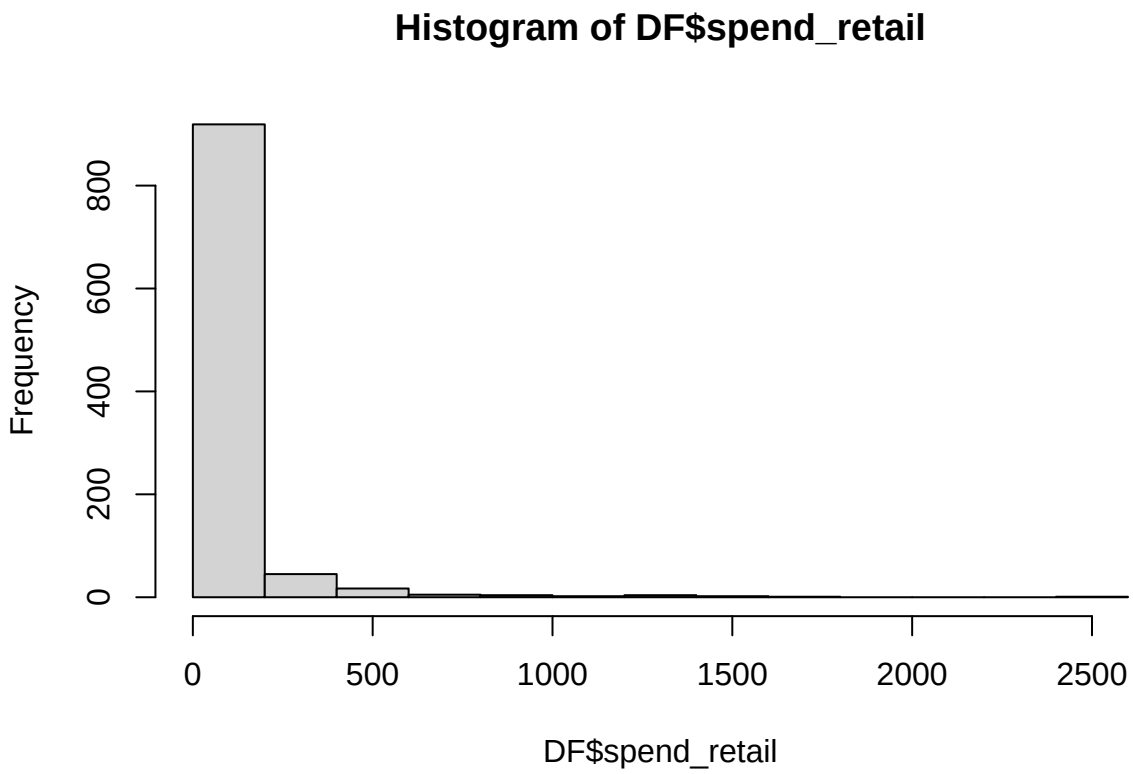
Q3: What is the minimum value of the high-skew variables? *(1 point)*

Answer 0

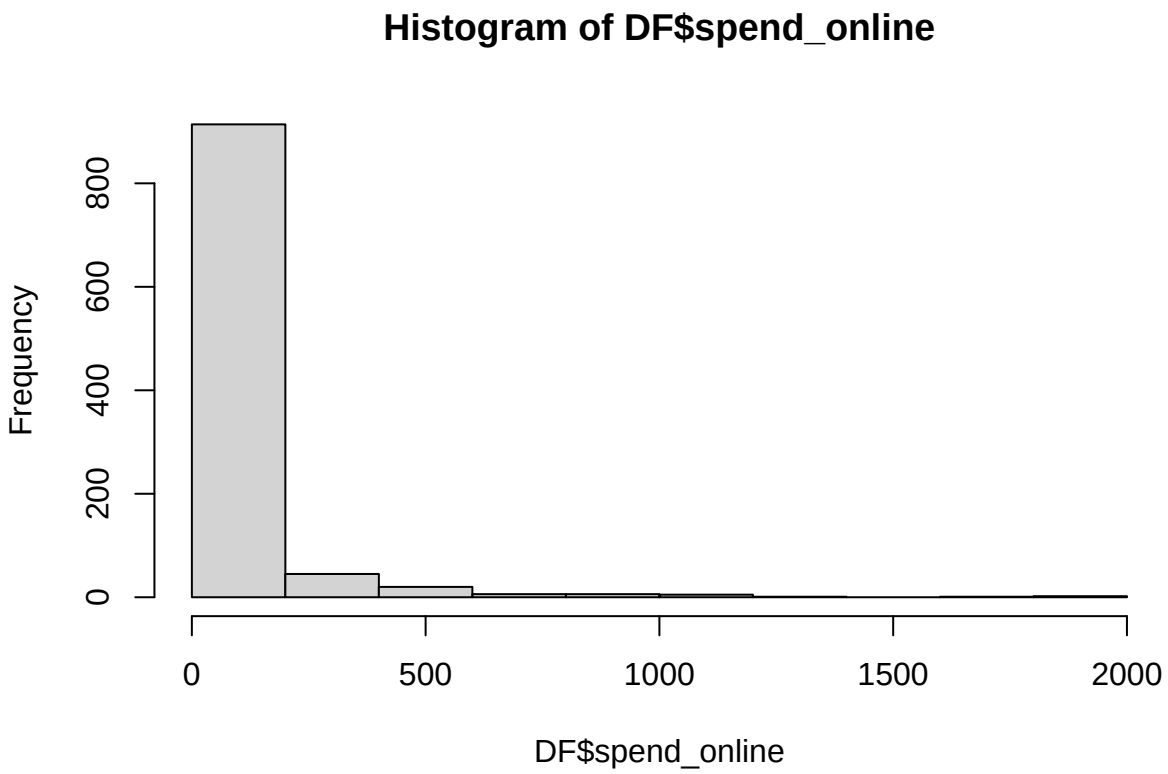
Histograms of all variables

Q4: Next, we wish to inspect the distribution of all the variables we might use for the cluster analysis. Generating histograms of each variable: *(2 points)*

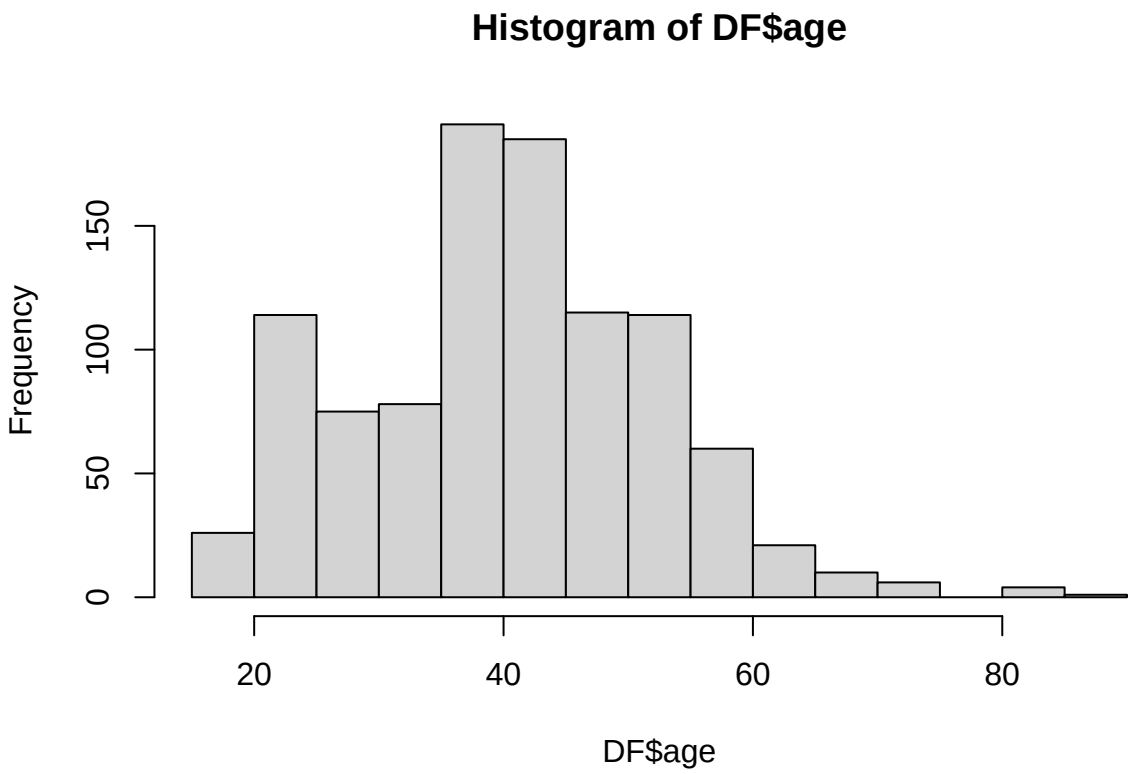
```
hist(DF$spend_retail)
```



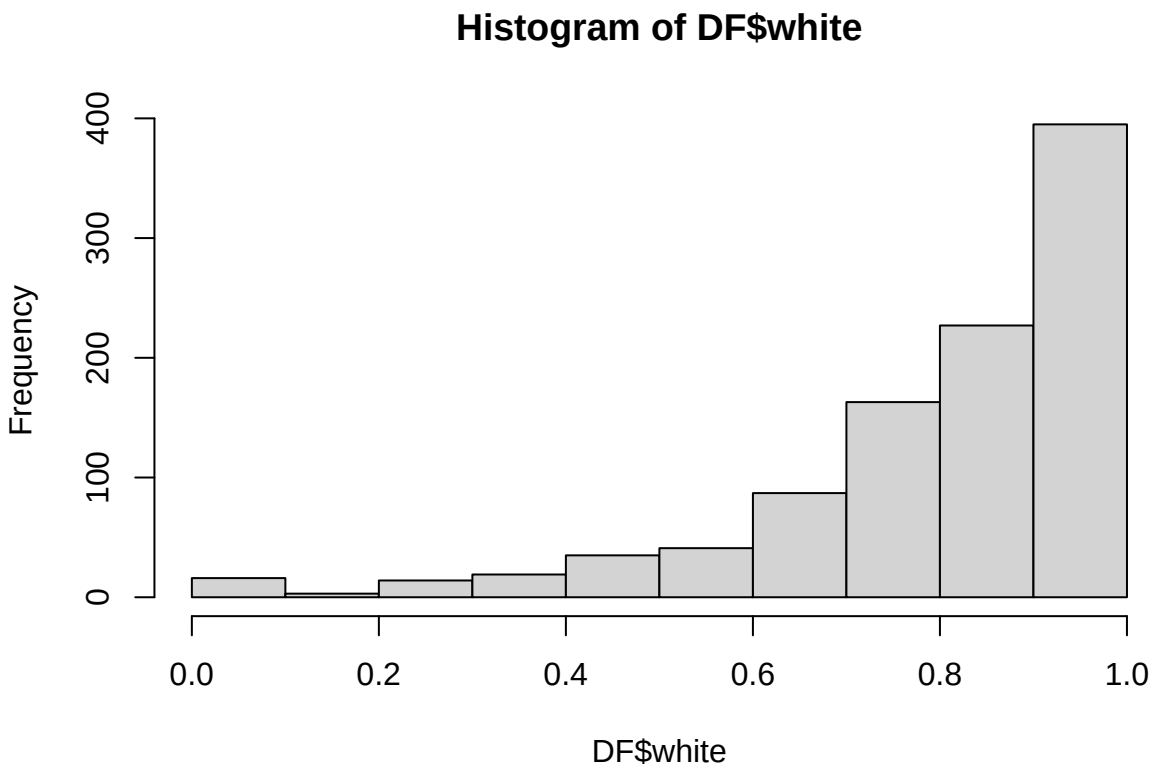

```
hist(DF$spend_online)
```


Histogram of DF$spend_online
Frequency
0
200
400
600
800
0
500
1000
1500
2000
DF$spend_online

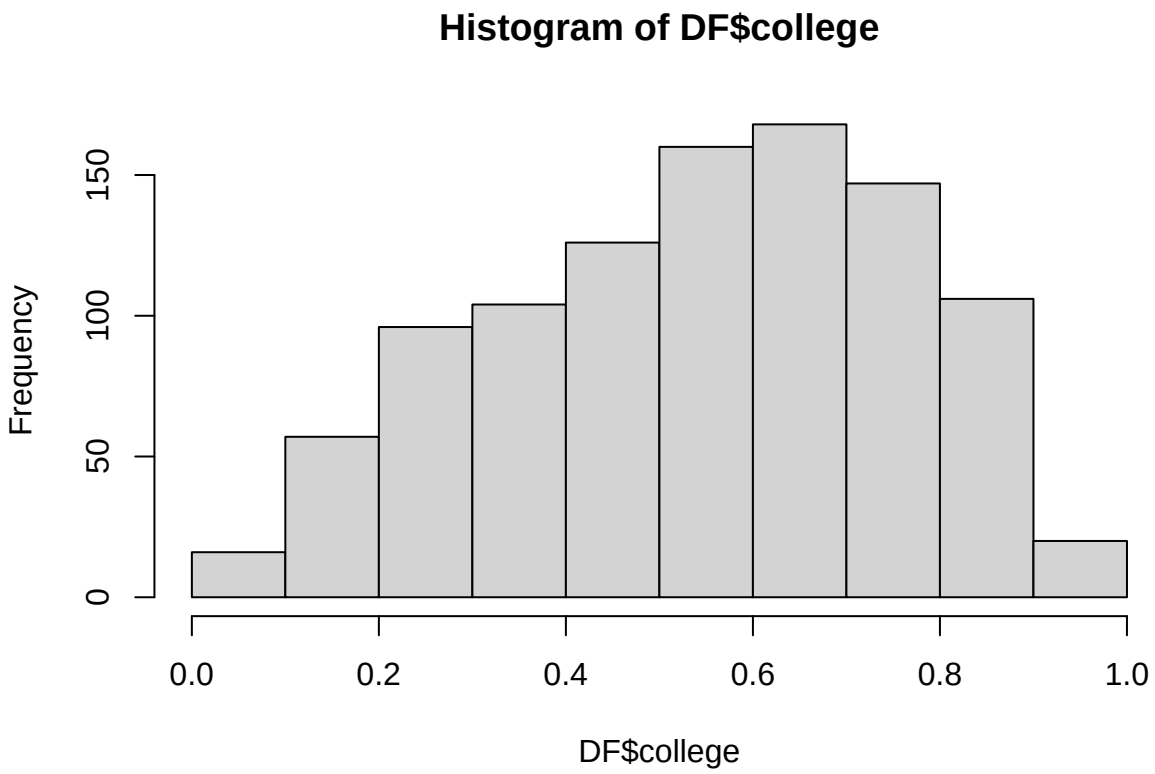
```
hist(DF$age)
```



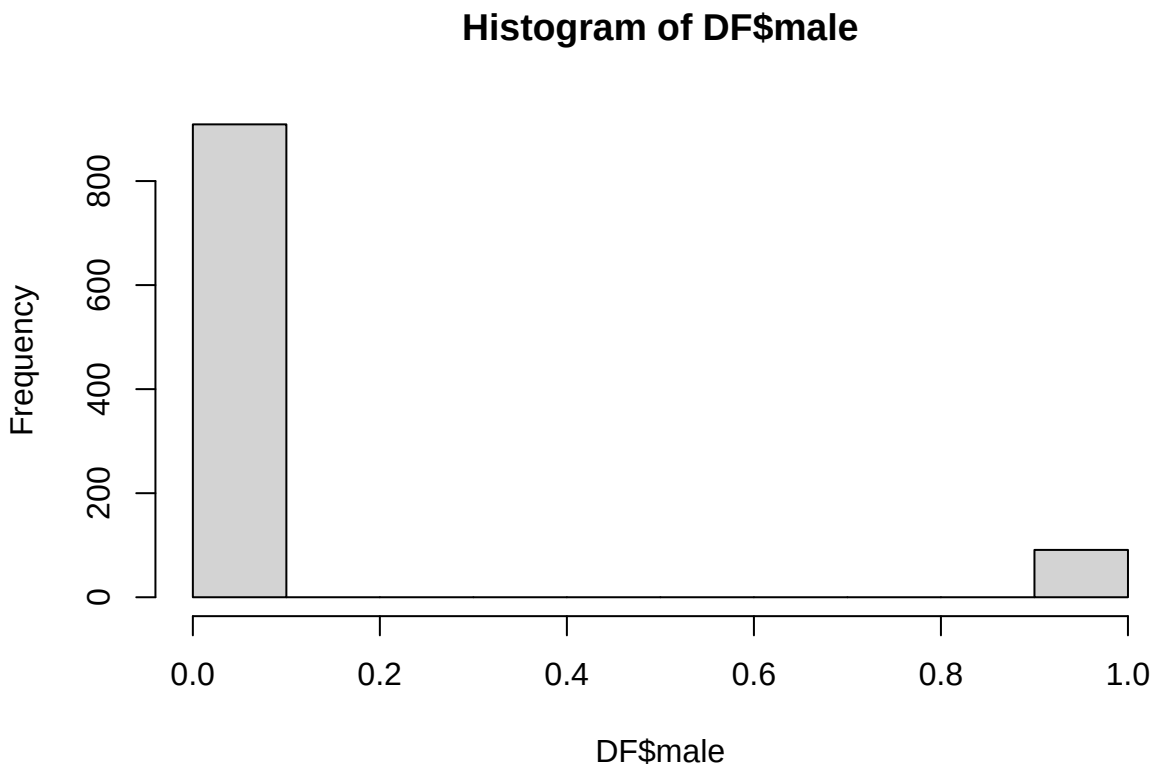

```
hist(DF$white)
```



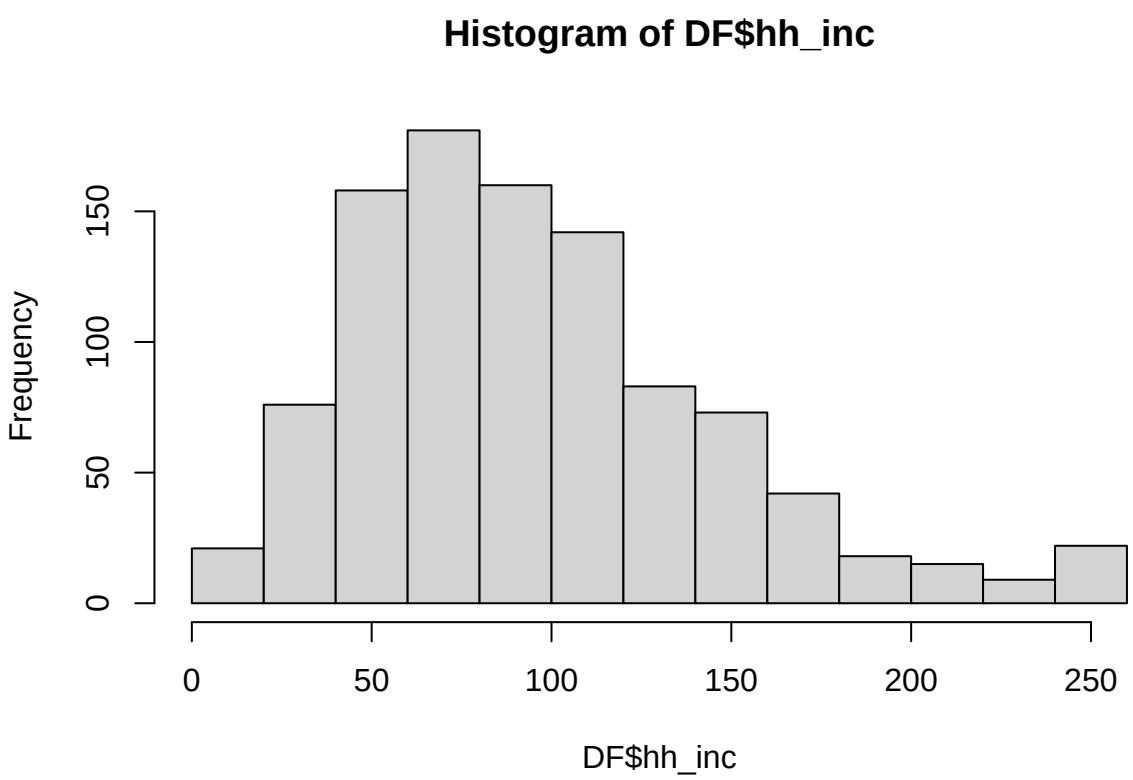

```
hist(DF$college)
```


Histogram of DF$college
Frequency
0
50
100
150
0.0
0.2
0.4
0.6
0.8
1.0
DF$college

```
hist(DF$male)
```




```
hist(DF$hh_inc)
```



Q5: By inspecting the histograms, which variables are continuous, and which are binary? *(1 point)*

Answer Continuous variables are spend_online, spend_retail, hh_inc, age, college, white Binary - male

Q6: Which variables demonstrate high-skew in their histograms? *(1 point)*

Answer spend_online, spend_retail, white

Q7: What do we conclude about: (a) which variables should be log-transformed, and (b) which distance metric would be appropriate for these data (assuming all variables will be used)? *(1 point)*

Answer (a) spend_online, spend_retail should we log transformed since these are highly skewed. (b) for all the continous variables like spend_retail, spend_onlie, hh_income, age Euclidean can be used while for categorical variables like male Gower can be used.

Q8: Which variables should be log-transformed? *(1 point)*

Answer spend_retail, spend_online

Clustering steps

Here we go through the clustering steps outlined in the lecture slides.

1. Select variables to use for clustering

Since we have a limited number of variables, and because all look potentially relevant, we will include all variables in our initial analysis.

Often, we do this iteratively, such that we may subsequently omit variables that contribute little to distinguishing the clusters or are impractical for developing targeted marketing strategies.

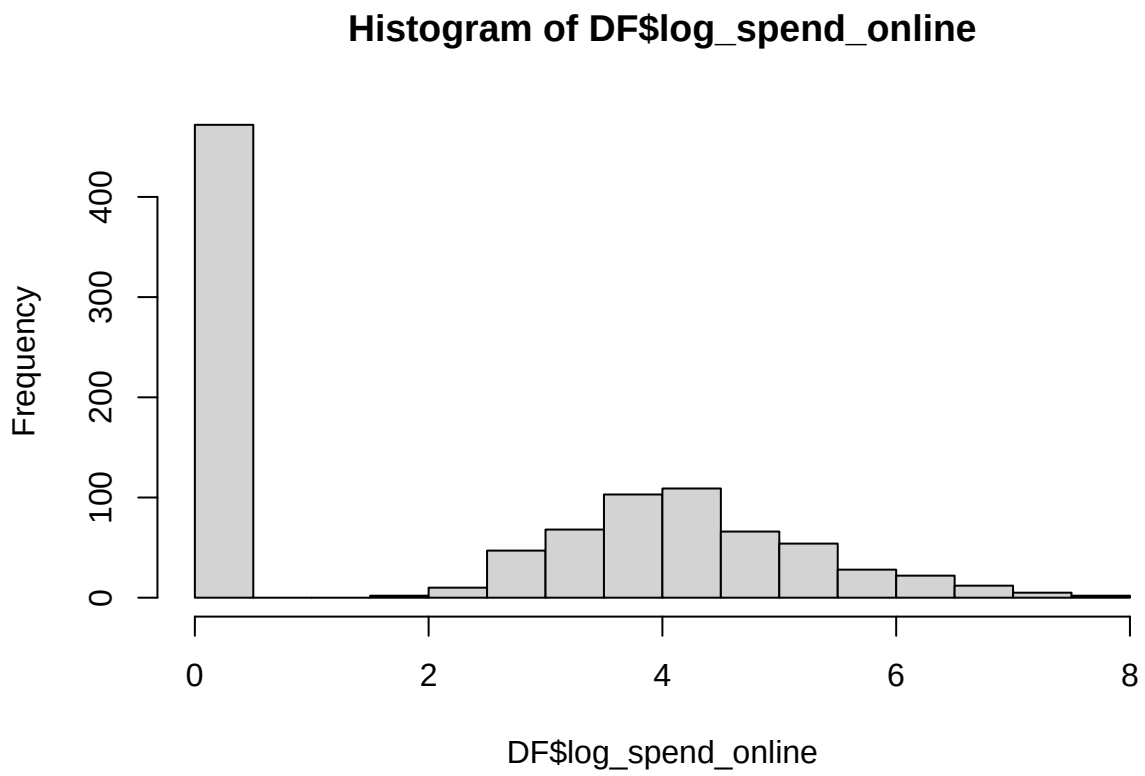
1.2. Log-transformation of skewed variables

Having observed the distributions of the variables, some stand out as different from the rest. Since clusting algorithms tend to perform poorly with highly skewed variables, we will transform them in a way that reduces skew.

The usual way to quickly handle skewed distributions such as these is to take the log-transform, which usually will give the data a more normal-shaped distribution. **Important:** The minmum value of these variables might be zero, and **log of zero (or negative numbers) is not possible. To deal with both problems, we transform the expenditure levels by taking log(1+x), where x is the untransformed variable.**

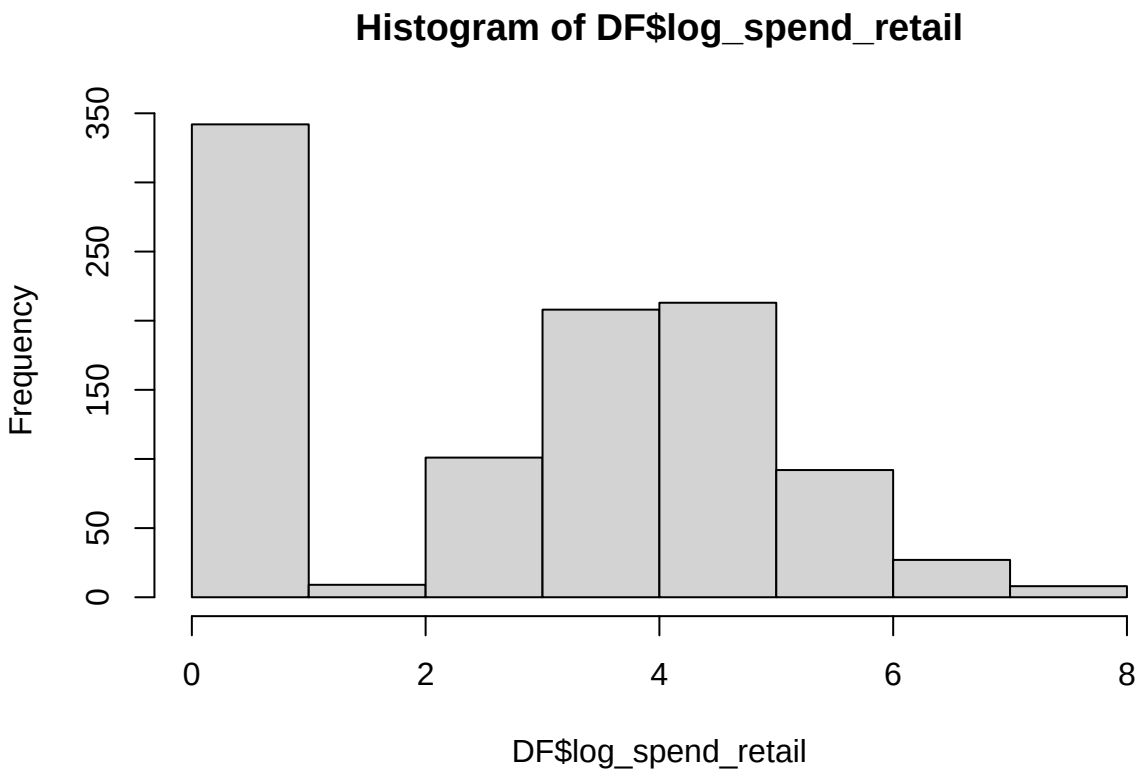
Specifically, to the dataframe `DF`, add variables named `log_variable` by taking the `log(1 DF$variable)` transformation of each variable that is skewed. Plot histograms for these. For example, `DF$spend_online` is one of the problem variables. Here is what we do:

```
DF$log_spend_online <- log(1+DF$spend_online)
hist(DF$log_spend_online)
```



Q9: Repeat this for the other skewed variable *(1 point)*

```
DF$log_spend_retail <- log(1+DF$spend_retail)
hist(DF$log_spend_retail)
```



Q10: How would you characterize the distribution of the transformed variables? Do the distributions appear more like the normal distribution (bell curve)? Are there multiple modes (peaks)? *(1 point)*

Answer The distribution looks a bit normal for all the values except 0. Yes there are multiple peaks. Since log tranformed doesn't take 0 into account which is why we add 1 in the command DF$log_spend_retail < -log(1 + DF$spend_retail), hence there's a spearate peak for 0.

1.3 Create dataframe with finalized cluster variables (only)

To make matters easier later, create a separate dataframe with *only* the cluster variables we intend to use to generate clustering (segmentation) purposes.

This code will create a dataframe called `DF` that *only* has the following variables: `log_spend_online`, `log_spend_retail`, `age`, `white`, `college`, `male`, `hh_inc`:

```
# create dataframe with transformed variables, omit non-cluster variables
DF <- DF
DF$iid <- NULL
DF$spend_online <- NULL
DF$spend_retail <- NULL

DF<- data.frame(
  log_spend_online = DF$log_spend_online,
  log_spend_retail = DF$log_spend_retail,
  age = DF$age,
  white = DF$white,
```

```
  hh_inc = DF$hh_inc,
  college = DF$college,
  male = DF$male
)
```

2 Define distance measure between individuals

2.1 Euclidean distance

We measure similarity between two customers by calculating the "distance" between them in terms of their observable characteristics.

Recall from basic geometry that we can find the distance (d) between two points (x_1,y_1) and (x_2,y_2) as: $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$. This is simply a version of the Pythagorean theorem, which relates the length of a triangle's hypotenuse (longest edge) to the length of its sides (generally expressed $c^2 = a^2 + b^2$, where c is the hypotenuse).

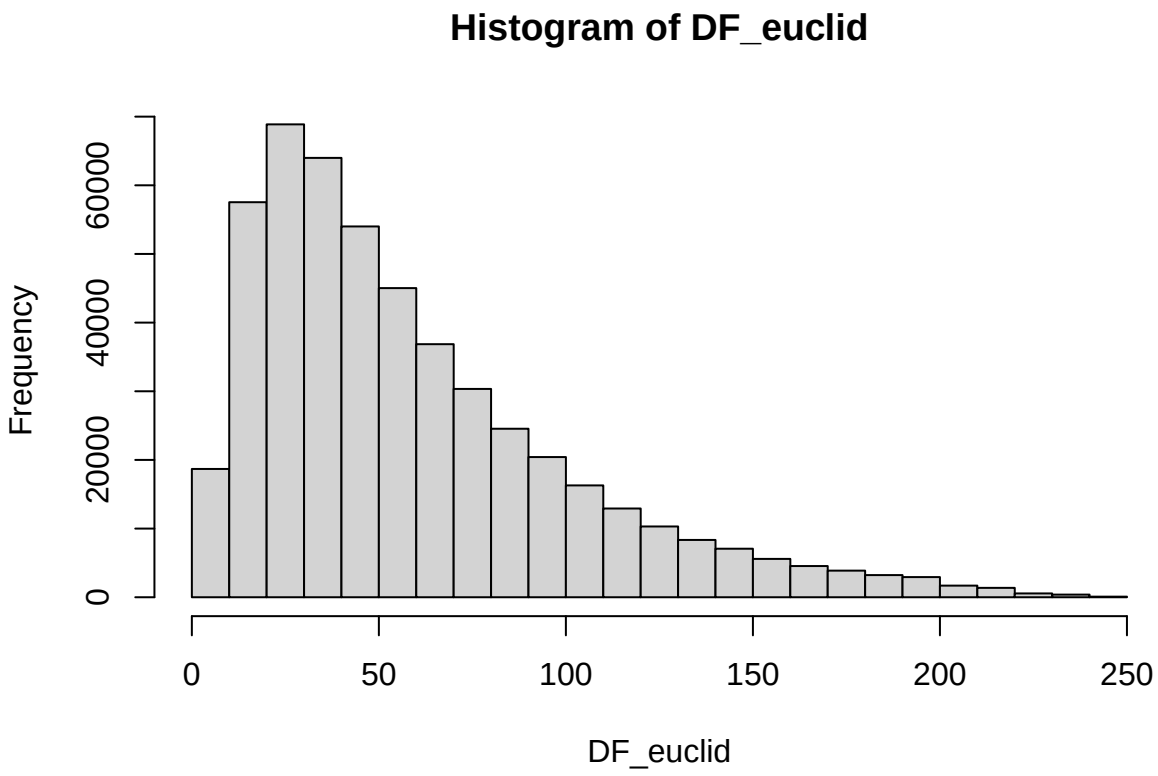
Rather than thinking of points in physical space, we can think of points in "characteristic" space. For example, the x-axis could represent a person's age and the y-axis could represent a person's income. The "distance" between two people in this case would be the square root of the squared difference in their age plus the squared difference in their income.

Consistent with its geometric origins, distance defined in this way is known as *Euclidean* distance. Note that the distance concept extends to higher dimensions, such that for $k = 1, ..., K$ dimensions, distance is given by: $d = \sqrt{(x_{11} - x_{21})^2 + (x_{12} - x_{22})^2 + ... + (x_{1K} - x_{2K})^2}$

Euclidean distance is (most) appropriately applied to a set of continuous variables. For data that is a mixture of continuous and binary/categorical variables, other distance metrics (e.g. Gower distance) are preferred.

This code calculate the (unstandardized) Euclidean distance between all pairs of consumers across the variables in dataframe `DF`.

```
library(cluster)
DF_euclid <- daisy(DF, metric = "euclidean",warnType=FALSE)
hist(DF_euclid)
```



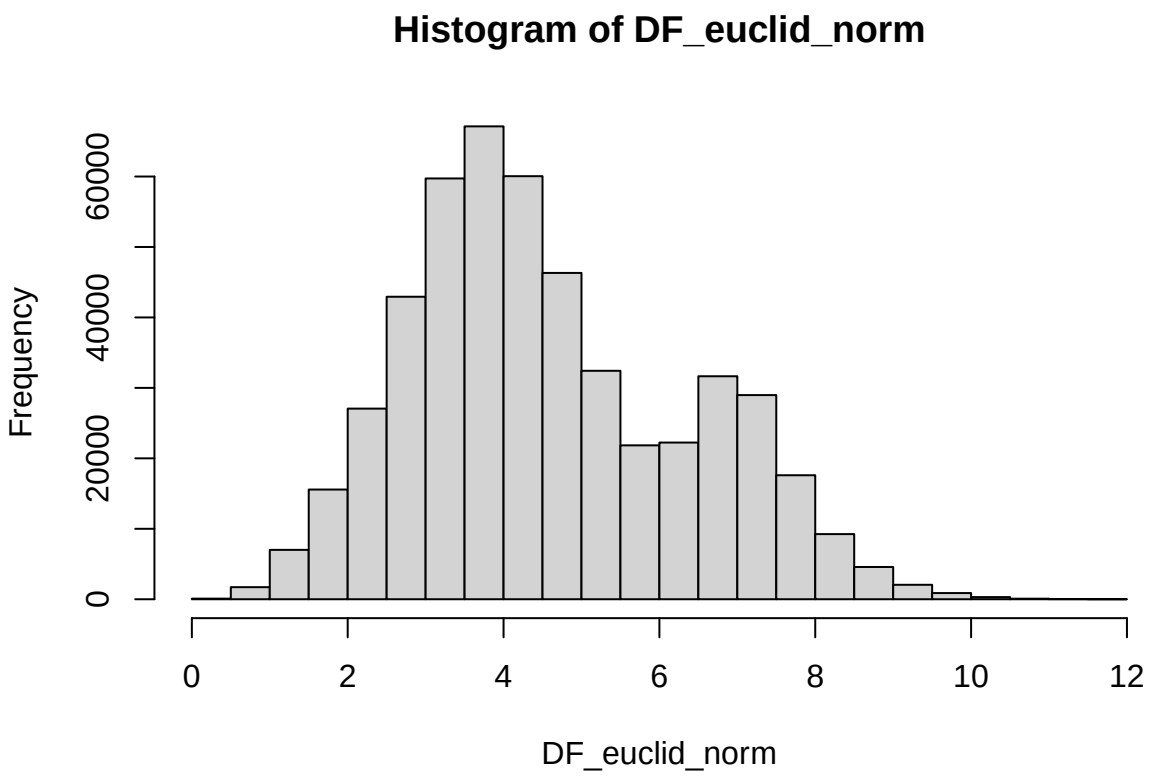
2.2 Standardized Euclidean distance

As previously mentioned, clustering algorithms tend to work best with input variables that are (approximately) normally distributed. This is principally because clustering algorithms tend to work best when the resulting *distance distribution* is normally distributed, and this tends to occur when the underlying variables are normally distributed.

In addition to log-transforming highly skewed variables, *standardizing* variables can result in distance distributions that are closer to being normally-distributed. Standardizing variables means that the variables are rescaled so that each variable has zero mean and unit (1) variance, e.g. $\tilde{x}_i = \frac{x_i - \bar{x}}{\sigma_x}$. The rationale for standardizing is that putting all variables on the "same scale" should give each variable roughly equal weight in contributing to the distance between points (consumers).

Q11: Calculate the standardized Euclidean distance between all pairs of consumers across the variables in dataframe `DF` (as defined in 3.1.2). The `stand=TRUE` option to the `daisy()` function can be useful for this task. Call the resulting list of pairwise distances `DF_euclid_norm`. Also, generate a histogram of `hist_euclid_norm`. *(2 points)*

```
DF_euclid_norm <- daisy(DF, metric = "euclidean", warnType = FALSE, stand = TRUE)
hist(DF_euclid_norm)
```



Q12: Characterize the shape of the standardized Euclidean distance distribution. How does it compare to the non-standardized distance distribution? What does this imply for the results of our clustering later? *(2 points)*

Answer The standardized Euclidean distance distribution looks a bit like bell-shaped and normally distributed when compared with non-standardized Euclidean distance distribution which is skewed at one end. This implies that all the variables are given equal weight and contribute equally to the distance calculation regardless of their scale, units or variance. Standardization can improve the quality of clustering results by ensuring that high variance or large-scale features do not dominate.

2.3 Gower distance

In many cases, we have a *mixture* of continuous and binary/categorical variables. In such cases, Euclidean distance metrics can perform poorly.

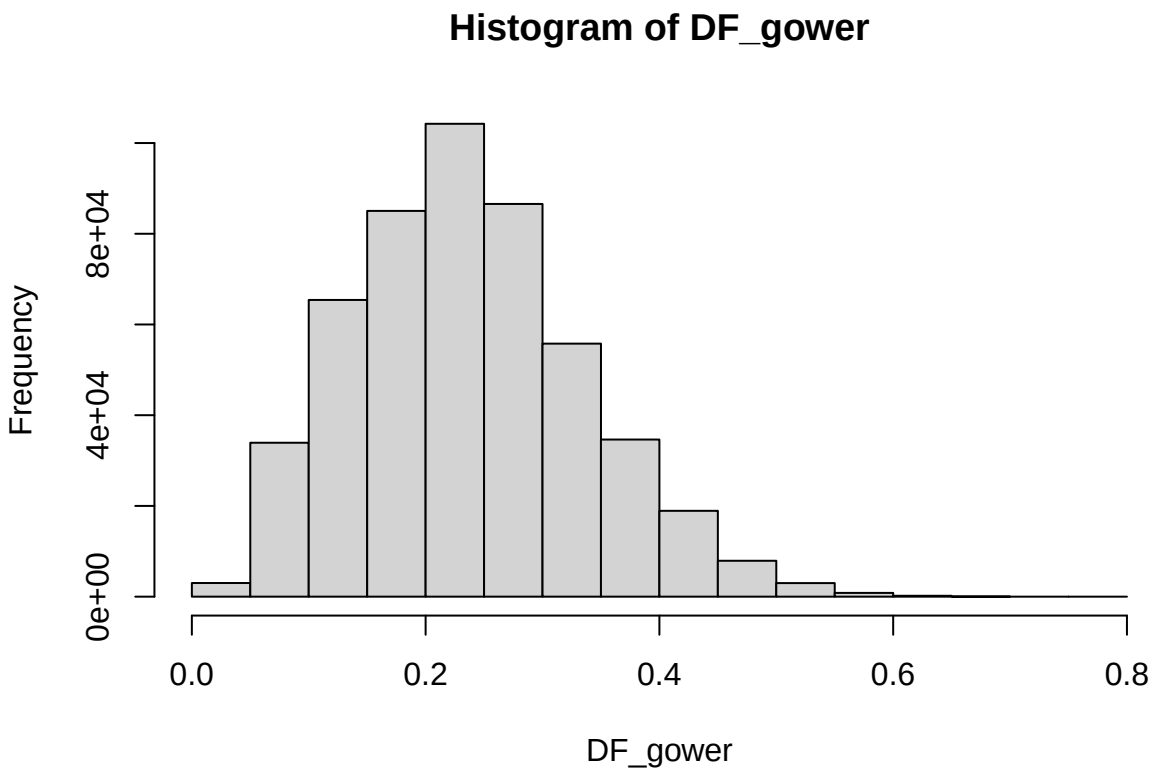
For mixed continuous & binary data, a better option is the Gower distance metric, which defines the distance between individuals i and j on variable k (e.g. age, income, etc.) as follows:

$$d_{ijk} = \begin{cases} \frac{|x_{ik}-x_{jk}|}{max(x_k)-min(x_k)} & x_k \ continuous \\ 0 & x_k \ binary, \ x_{ik} = x_{ij} \\ 1 & x_k \ binary, \ x_{ik} \neq x_{ij} \end{cases}$$

The total distance between individuals i and j is then just the sum over all observed variables, $d_{ij} = \sum_k d_{ijk}$. Note that the Gower metric "automatically" standardizes variables by construction. For continuous variables, the distance between any two individuals is normalized with respect to the maximum distance possible between any two individuals. The result is to map the original variable into the range [0,1], which is the same scale as binary variables.

Q13: Calculate the Gower distance between all pairs of consumers across the variables in dataframe `DF`. Call the resulting list of pairwise distances `DF_gower`. Also, generate a histogram of `DF_gower`. *(1 point)*

```
DF_gower <- daisy(DF, metric = "gower", warnType = FALSE)
hist(DF_gower)
```



Q14: Based on the data types in `DF` and the shapes of the distance distributions, I am going to go with the Gower distance metric for the rest of this homework. Why is that? *(2 points)*

Answer Because Gower metric gives a more normalized distribution that Euclidean metric since the dataframe has a mix of continous and binary variables, Euclidean can perform poorly and Gower metric might be more useful.

3 Select clustering procedure

Using the pair-wise distance measures, clustering algorithms are used to group individuals into segments (clusters). There are many different types of clustering algorithms, which generally fall into 2 categories: hierarchical and non-hierarchical.

We focus on non-hierarchical methods, and the k-means (`kmeans()`) clustering algorithm in particular. We choose k-means because it tends to be the most general purpose method in terms of applicability and performance. Non-hierarchical methods like k-means determine clusters by optimizing (maximizing/minimizing) some measure of clustering "fit".

In the case of the k-means algorithm, the objective is to minimize the total within-cluster sum of squares. That is, for a fixed number of clusters, the algorithm minimizes pairwise distances within the clusters. To

determine cluster membership, the k-means algorithm begins by assigning k individuals at random to the k clusters. Then, the algorithm iterates between: (a) assigning individuals to the cluster with the closest centroid (mean variable values for all cluster members), and (b) recomputing the cluster cenrtoid values. The algorithm coverges (stops) when further iterations do not change the membership of the clusters.

In this section, we will perform k-means clustering using the **Gower distance matrix**. We will estimate cluster solutions for segments of size 2, 3 and 4. We will analyze these clustering solutions in section 5.

3.1 K-means (Gower), 2 segments

Q15: Using the Gower distance matrix, perform a k-means cluster analysis with K = 2 clusters. Use a minimum of 10 initial starting points. Save the result to `clu_gower_2`. Finally, add the cluster assignments to the orginal dataframe, `DF` – name the column `clu_gower_2`: *(1 point)*

```
clu_gower_2 <- kmeans(DF_gower, centers = 2, nstart = 10)
DF$clu_gower_2 <- clu_gower_2$cluster
```

3.2 K-means (Gower), 3 segments

Q16: Using the Gower distance matrix, perform a k-means cluster analysis with K = 3 clusters. Save the result to `clu_gower_3`. Finally, add the cluster assignments to the orginal dataframe, `DF` – name the column `clu_gower_3`: *(1 point)*

```
clu_gower_3 <- kmeans(DF_gower, centers = 3, nstart = 10)
DF$clu_gower_3 <- clu_gower_3$cluster
```

3.3 K-means (Gower), 4 segments

Q17: Using the Gower distance matrix, perform a k-means cluster analysis with K = 4 clusters. Save the result to `clu_gower_4`. Finally, add the cluster assignments to the orginal dataframe, `DF` – name the column `clu_gower_4`: *(1 point)*

```
clu_gower_4 <- kmeans(DF_gower, centers = 4, nstart = 10)
DF$clu_gower_4 <- clu_gower_4$cluster
```

4 Select number of clusters

4.1 Elbow plot

Here will will use an elbow plot to assist with determining the number of clusters. Generate an elbow plot for 1 to 10 clusters.

Recall that the elbow plot graphs the within-cluster sum of squares vs. the number of clusters. You can access the within-cluster sum of squares using `$withinss`, as in `clu_gower_2$wihtinss`. Note further that the within-cluster sum of squares returned from `$withinss` is a *list*, with 1 list element per cluster – so, to get the total (across clusters) within-cluster sum of squares, we would for example calculate `sum(clu_gower_2$wihtinss)`.

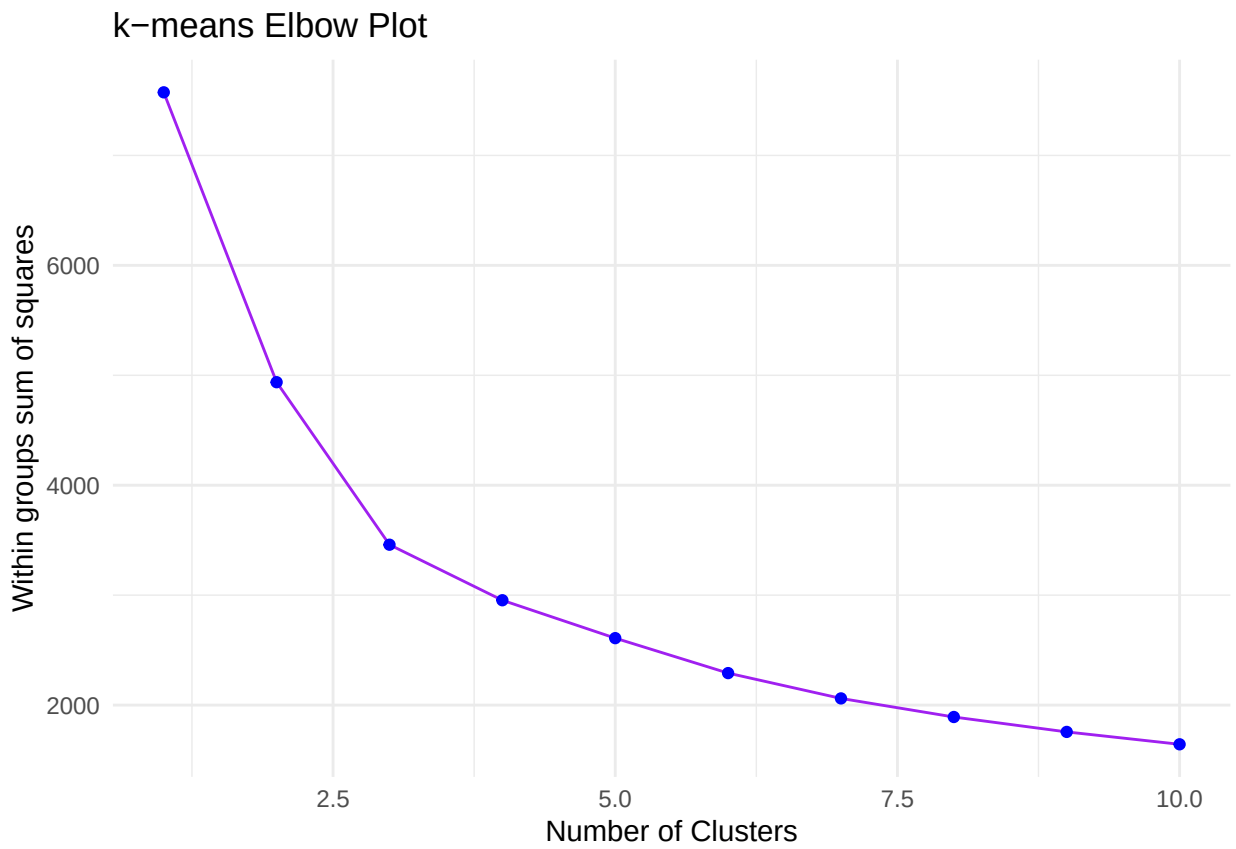
Hint: A loop is a straightforward way to approach this problem.

Q18: Make the elbow plot *(3 points)*

```r
max_clusters <- 10
wss <- rep(0, max_clusters)

for (i in 1:max_clusters) {
  clu_gower_10 <- kmeans(DF_gower, centers = i, nstart=10)
  wss[i] <- sum(clu_gower_10$withinss)
}

as.data.frame(wss)
```

```
##         wss
## 1  7573.920
## 2  4937.121
## 3  3458.796
## 4  2953.573
## 5  2608.835
## 6  2290.804
## 7  2060.684
## 8  1891.541
## 9  1755.923
## 10 1643.243
```

```r
elbow_data <- data.frame(k = 1:max_clusters, WCSS = wss)

ggplot(elbow_data, aes(x = k, y = WCSS)) +
  geom_line(color = "purple") +
  geom_point(color = "blue") +
  labs(title = "k-means Elbow Plot",
       x = "Number of Clusters",
       y = "Within groups sum of squares") +
  theme_minimal()
```



Q19: We are going to use 3 clusters from here. Why did I choose that? *(1 point)*

Answer Because the within sum of squared error reduced considerably when going from 0 to 3 but beyond 3 it sort of got stable and hence creating clusters more than 3 wouldn't give a lot of difference between different clusters. For the difference within the clusters to be less and the difference outside of the clusters (between different clusters) to be more, 3 would be a good choice.

5 Profile and interpret the clusters

The final stage of the cluster analysis is to profile the clusters and analyze the results. Profiling a cluster entails two things:

1. Calculating the market share associated with the cluster (segment).
2. Calculating cluster (segment) centroids, i.e. the mean variable values across all cluster members.

We analyze cluster profiles primarily by assessing them with respect to the segmentation criteria:

1. Substantial – Segment market shares are large enough to warrant serving. A counter-example for 3 segments might be market shares of 98%, 1% and 1%. Unless the 1% segments are known to be associated with very high willingness to pay customers, such a scheme would have little practical value.
2. Actionable – Segment characteristics can be translated into targeted marketing policies (e.g. using age/income differences to craft different promotional vehicles). Targeted policies must also be consistent with firm competencies.
3. Differentiable – Differences between segments should be clearly defined. That is, differences across segments must be large enough to generate different (actionable) marketing policies.

5.1 K-means (Gower), 3 segments

NOTE: In case you were wondering, the labeling of segments is arbitrary – i.e., the segment with 55.2% of the customers could have been labeled segment 1 or segment 2. Some software packages use the convention that segments are labeled in order of decreasing size – R is apparently not one of them.

Calculate and print the fraction of customers assigned to each of the K = 3 segments. **Q20: 1 point**

Calculate and print the cluster centroids (mean values of the variables for customers in the segment). **Q21: 1 point**

```
DF$cluster <- clu_gower_3$cluster
library(dplyr)
cluster_size_3 <- DF |>
  group_by(cluster) |>
  summarise(size = n(),
            proportion = round(n()/nrow(DF), 3))
print(cluster_size_3)
```

```
## # A tibble: 3 x 3
##   cluster  size proportion
##     <int> <int>      <dbl>
## 1       1   423      0.423
## 2       2   107      0.107
## 3       3   470      0.47
```

```
cluster_means_3<- DF |>
  group_by(cluster) |>
  summarise(across(c(log_spend_online, log_spend_retail, age, white, college, male, hh_inc), mean)) |>
  round(3)
print(cluster_means_3)
```

```
## # A tibble: 3 x 8
##   cluster log_spend_online log_spend_retail   age white college  male hh_inc
##     <dbl>            <dbl>            <dbl> <dbl> <dbl>   <dbl> <dbl>  <dbl>
## 1       1            4.38             1.35  40.2 0.804   0.501  0      86.1
## 2       2            2.35             2.46  40.5 0.694   0.498  0.85   97.6
## 3       3            0.325            3.98  41.6 0.819   0.592  0     105.
```

Q22: Attempt to label the segments in the most descriptive but brief terms possible (e.g. "online affluent") *(1 point)*

Answer Cluster 1: Online Spend Dominant Cluster 2: Retail Spend Affluent Cluster 3: Male Affluent Spenders

Q23: Which segment is biggest? smallest? How do those segments differ in characteristics? *(1 point)*

Answer The biggest segment is cluster 2 with 47% customers and smallest is cluster 3 with 10.7% customers. The clusters differ in characteristics with cluster 2 having high hh income and high retail spending, suggesting they may value in-store experiences.Cluster 3 has male majority, high hh income and balanced retail and online spending.

Q24: Evaluate these segments on the basis of the segmentation criteria (substantial, actionable, differentiable) *(1 point)*

*Substantial – The segments are substantial, with each segment representing distinct sizes. While cluster 2 has a large size consisting of financially well-off people with retail shopping behaviour, cluster 3 is niche group with male majority.

*Actionable – The differences in spending habits (online vs. retail) and demographics (such as the male dominance in Cluster 3) suggest actionable strategies. Marketing strategies can be tailored to the preferences of each segment, with digital campaigns focused on Cluster 1, in-store promotions for Cluster 2, and gender-targeted approaches for Cluster 3.

*Differentiable – The segments are differntiable with unique spending behaviors, income and demographics. Cluster 1 is conservative, cluster 2 is affluent in store shoppers , while cluster 3 is balanced male shoppers.

Final 10 points

Repeat this task for the 2 and 4 segment solutions. Then, explain how these differ, and what scheme you would recommend using (2, 3, or 4)

For Segment 2:

```
DF$cluster <- clu_gower_2$cluster
library(dplyr)
cluster_size_2 <- DF |>
  group_by(cluster) |>
  summarise(size = n(),
            proportion = round(n()/nrow(DF), 3))
print(cluster_size_2)
```

```
## # A tibble: 2 x 3
##   cluster  size proportion
##     <int> <int>      <dbl>
## 1       1   448      0.448
## 2       2   552      0.552
```

```
cluster_means_2<- DF |>
  group_by(cluster) |>
  summarise(across(c(log_spend_online, log_spend_retail, age, white, college, male, hh_inc), mean)) |>
  round(3)
print(cluster_means_2)
```

```
## # A tibble: 2 x 8
##   cluster log_spend_online log_spend_retail   age white college  male hh_inc
##     <dbl>            <dbl>            <dbl> <dbl> <dbl>   <dbl> <dbl>  <dbl>
## 1       1            4.27              1.09  40.0 0.776   0.482 0.123   84.9
## 2       2            0.625             4.01  41.6 0.819   0.594 0.065  106.
```

Q22: Attempt to label the segments in the most descriptive but brief terms possible (e.g. "online affluent") *(1 point)*

Answer Cluster 1: Affluent Retail Shoppers Cluster 2: Conservative Online Shoppers

Q23: Which segment is biggest? smallest? How do those segments differ in characteristics? *(1 point)*

Answer The biggest segment is cluster 1 with 55.2% customers and smallest is cluster 2 with 44.8% customers. The primary differences between these segments are in their shopping preferences (online vs. retail), household income levels, and to some extent, demographic profiles (e.g., proportion of college-educated individuals and gender distribution)

Q24: Evaluate these segments on the basis of the segmentation criteria (substantial, actionable, differentiable) *(1 point)*

*Substantial – The segments are substantial, with each segment representing distinct sizes.

*Actionable – the difference in demographic profile, spending habits and household income can help draft marketing strategies for example, with online promotions targeted at Cluster 2 and more traditional retail-focused strategies for Cluster 1.

*Differentiable – The segments are distinctly differentiable with distinct preference for online v/s retail shoppers, demographic profiles, and household income.

For Segment 4:

```
DF$cluster <- clu_gower_4$cluster
library(dplyr)
cluster_size_4 <- DF |>
  group_by(cluster) |>
  summarise(size = n(),
            proportion = round(n()/nrow(DF), 3))
print(cluster_size_4)
```

```
## # A tibble: 4 x 3
##   cluster  size proportion
##     <int> <int>      <dbl>
## 1       1   102      0.102
## 2       2   303      0.303
## 3       3   428      0.428
## 4       4   167      0.167
```

```
cluster_means_4<- DF |>
  group_by(cluster) |>
  summarise(across(c(log_spend_online, log_spend_retail, age, white, college, male, hh_inc), mean)) |>
  round(3)
print(cluster_means_4)
```

```
## # A tibble: 4 x 8
##   cluster log_spend_online log_spend_retail   age white college  male hh_inc
##     <dbl>            <dbl>            <dbl> <dbl> <dbl>   <dbl> <dbl>  <dbl>
## 1       1            2.22             2.47   40.8 0.71    0.505 0.892   98.5
## 2       2            4.16             0.008  40.4 0.795   0.484 0       85.1
## 3       3            0.006            3.91   41.6 0.816   0.582 0      102.
## 4       4            4.60             4.64   40.3 0.82    0.578 0       99.2
```

Q22: Attempt to label the segments in the most descriptive but brief terms possible (e.g. "online affluent") *(1 point)*

Answer Cluster 1: Digital Shopper Cluster 2: Affluent Male Shoppers Cluster 3: Traditional Retail Affluent Shopper Cluster 4: Omnichannel Shoppers

Q23: Which segment is biggest? smallest? How do those segments differ in characteristics? *(1 point)*

Answer The biggest segment is cluster 3 with 42.8% customers and smallest is cluster 2 with 10.2% customers. The clusters differ in characteristics with biggest difference in shopping channel preferences, cluster 3 prefers retail shopping while cluster 2 has balanced shopping preferences and buyers are skewed towards male demographic.

Q24: Evaluate these segments on the basis of the segmentation criteria (substantial, actionable, differentiable) *(1 point)*

*Substantial – The segments are substantial, with each segment representing distinct sizes.The segments are large enough showing different customer base.

*Actionable – the difference in demographic profile, spending habits and household income can help draft marketing strategies for example, luxury or premium products for affluent customers while dicounts/promos or budget-conscious offerings to low hh income shoppers.

*Differentiable – The segments are distinctly differentiable based on spending habits, demographics, and household income.

I would recommend scheme 4, where there are 4 clusters since these customers represent groups which are highly distinct from each other with high difference in some characteristics or the other in terms of shopping behaviour, gender,hh income,.