Homework 1

Alex Soupir

August 28, 2019

Packages: HSAUR3, ggplot2, reshape, MASS

Collaborators:

1. Question 1.1, pg. 23 in Handbook *this question will require you to make some assumptions. List your assumptions and how you interpreted the question.*

- Statement/Question: Calculate the median profit for the companies in the US and the median profit for the companies in the UK, France, and Germany.

Assumptions: The companies are only located in a single country and not multiple countries. Countries that report NA don't have data on profits and therefore are to be removed from the median calculation otherwise the resulting profit calculated will be NA.

```
## country()
```

```
##         France        Germany United Kingdom  United States
##          0.190          0.230          0.205          0.240
```

- Discussion: The data frame was first subset by the countries that we are interested in using the *%in%* operator within subset. Initially the resulting table reported NA for the US and UK because some of the companies didn't have a value for profits, therefore, when running tapply *na.rm = TRUE* was added to make sure only those with reported profits were used in finding the median company profit for that country.

2. Question 1.2, pg. 23 in Handbook

- Statement/Question: Find all German companies with negative profit.

```
##                          name country                    category profits
## 350         Allianz Worldwide Germany                   Insurance   -1.23
## 364          Deutsche Telekom Germany Telecommunications services  -25.83
## 397                     E.ON Germany                   Utilities   -0.73
## 431       HVB-HypoVereinsbank Germany                     Banking   -0.87
## 500               Commerzbank Germany                     Banking   -0.31
## 798     Infineon Technologies Germany             Semiconductors   -0.51
## 869               BHW Holding Germany      Diversified financials   -0.38
## 926  Bankgesellschaft Berlin Germany                     Banking   -0.74
## 1034           W&W-Wustenrot Germany      Diversified financials   -0.08
## 1187         mg technologies Germany                   Chemicals   -0.13
## 1477 Nurnberger Beteiligungs Germany                   Insurance   -0.03
## 1887            SPAR Handels Germany                Food markets   -0.40
## 1994                Mobilcom Germany Telecommunications services   -3.62
```

- Discussion: The main table was subset based on what we wanted to look at; the country Germany and those with profits less than 0 (negative profits), then printed out the company name, the country, category, and the profits that they had. This allowed us to make sure that the table had the right values in it for profits and country. The results aren't unexpected. Initially the data was subset by country *than* by profits, but realized this could be done easier by subsetting both in the same step.

3. Question 1.3, pg. 23 in Handbook

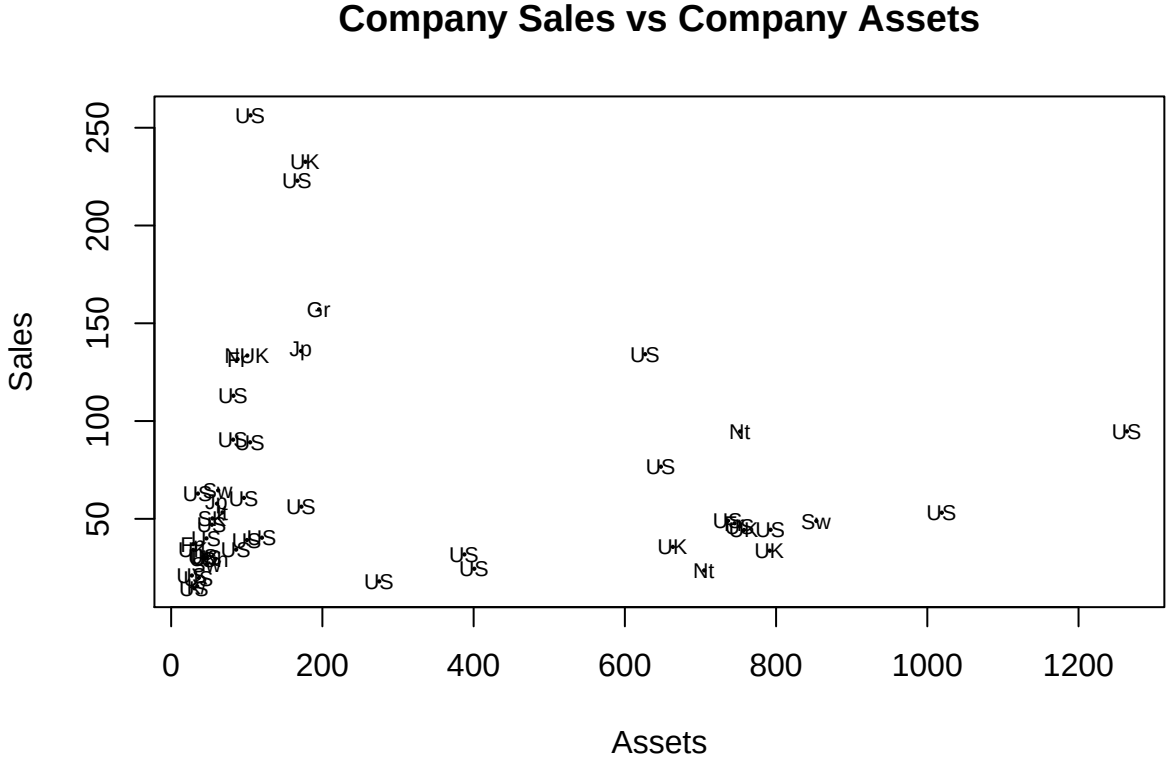
- Statement/Question: To which business category do most of the Bermuda island companies belong?

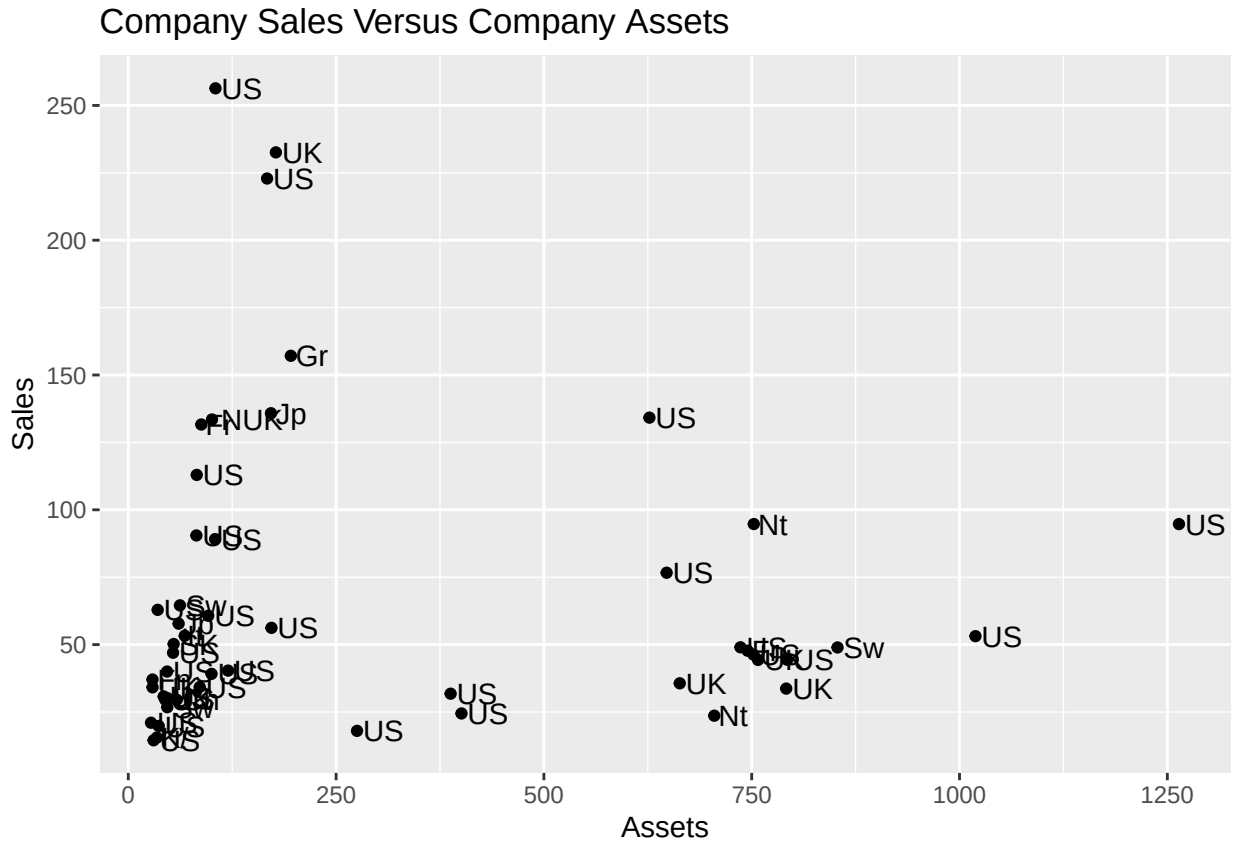
```
## Insurance
##        10
```

- Discussion: In order to get the results of this query, the original data needed to be subset to only include the companies that are from the Bermuda islands. After subsetting, a summary of the companies category was created, then sorted to get the category counts in ascending order. With the category in ascending order I finally selected the last category using the *length()* function, giving us Insurance companies being the most numerous from the Forbes2000 data set that reside in Bermuda.
  - This method was mostly trial and error, following the path to figuring out what acheived the next step toward getting the top category count for Bermuda. I'm sure there are easier ways using some of the other functions within packages that I haven't used before, but this seemed to work out. As far as if this is a result that was expected, from just what has been heard in the past Bermuda sounds like a tax-haven so I would have thought that more conflomerates or software companies would be located there. It also probably of importance taht Insurance counts are **5x** higher than the next business class of oil and gas.

4. Question 1.4, pg. 23 in Handbook

- Statement/Question: For the 50 companies in the Forbes data set with the highest profits, plot sales against assets (or some suitable transformation of each variable), labeling each point with the appropriate country name which may need to be abbreviated (using abbreviate) to avoid making the plot look too 'messy'.





- Discussion: Just like the previous example, the data has to be sorted in order to determine the companies that have the highest profits by *order(-**column**)* and then subset including only the companies with the top 50 profits. With the subset data, sales were plotted against assets and each point was labeled with the country that the company is from. An issue that I ran into was trying to add a legend that has the abbreviations to what the long name, but wasn't sure how to do that. Also thought about adding color but there would be a lot of colors to make sense of and some were really similar and hard to tell the difference between.
  - An interesting point here is how the United States is scattered throughout the whole plot, and there appears to be a large number from the United States. This seems to be pretty expected given that the United States seems to have a good atmosphere and environment for businesses to thrive today, so 20 years ago I could see the same being said. The United Kingdom appears to have a wide spread as well, but much less in the number of companies. There is a greenland company that has low **Assets** but maintains a fair level of sales.
  - There was some complication in getting ggplot to work with knitting. I'm not sure what the issue was, or how it was fixed, but it kept saying that I was trying to apply *is.na()* to a non-(list or vector)? Works now though, which is nice.

5. Question 1.5, pg. 23 in Handbook

- Statement/Question: Find the average value of sales for the companies in each country in the Forbes data set, and find the number of companies in each country with profits above 5 billion US dollars.

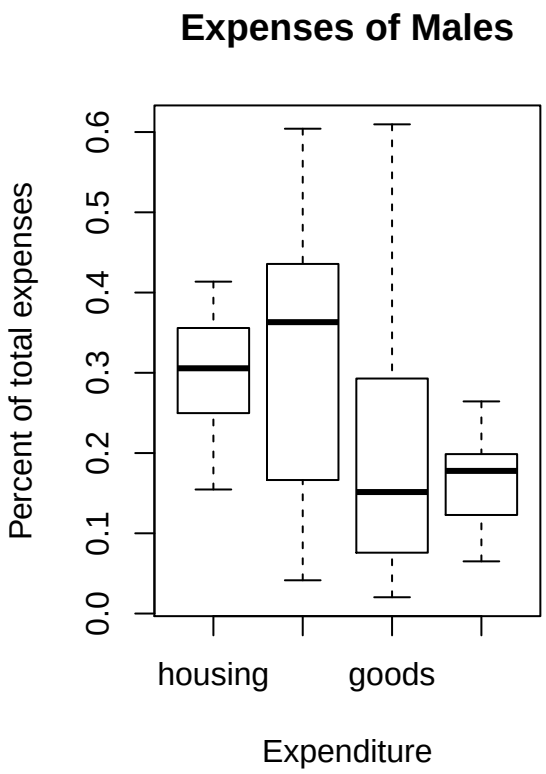
```
##                       Country Mean_Sales Profits_Greater_Than_5bil
## 60                United States  10.058256                       20
## 52                  Switzerland  12.456765                        3
## 56               United Kingdom  10.445109                        3
## 37 Netherlands/ United Kingdom  92.100000                        1
## 18                      Germany  20.781385                        1
```

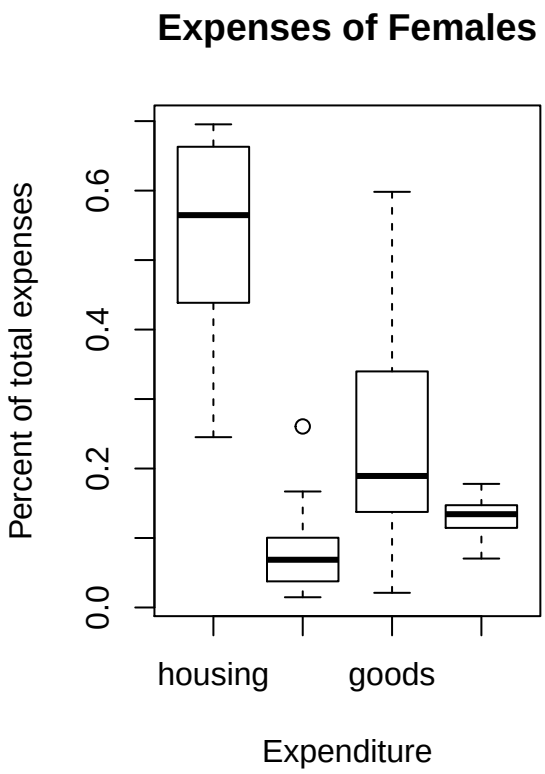
```
## 16                     France  20.102063                     1
## 28                      Japan  10.190633                     1
## 49                South Korea   7.969333                     1
## 12                      China   5.099600                     1
## 36                Netherlands  17.020714                     0
## 31                      Korea  15.005000                     0
## 33                 Luxembourg  14.185000                     0
## 3    Australia/ United Kingdom  11.595000                     0
## 39                     Norway  10.780000                     0
## 15                    Finland  10.291818                     0
## 27                      Italy  10.213902                     0
## 6                     Belgium  10.114444                     0
## 57   United Kingdom/ Australia  10.010000                     0
## 50                      Spain   7.843448                     0
## 46                     Russia   7.672500                     0
```

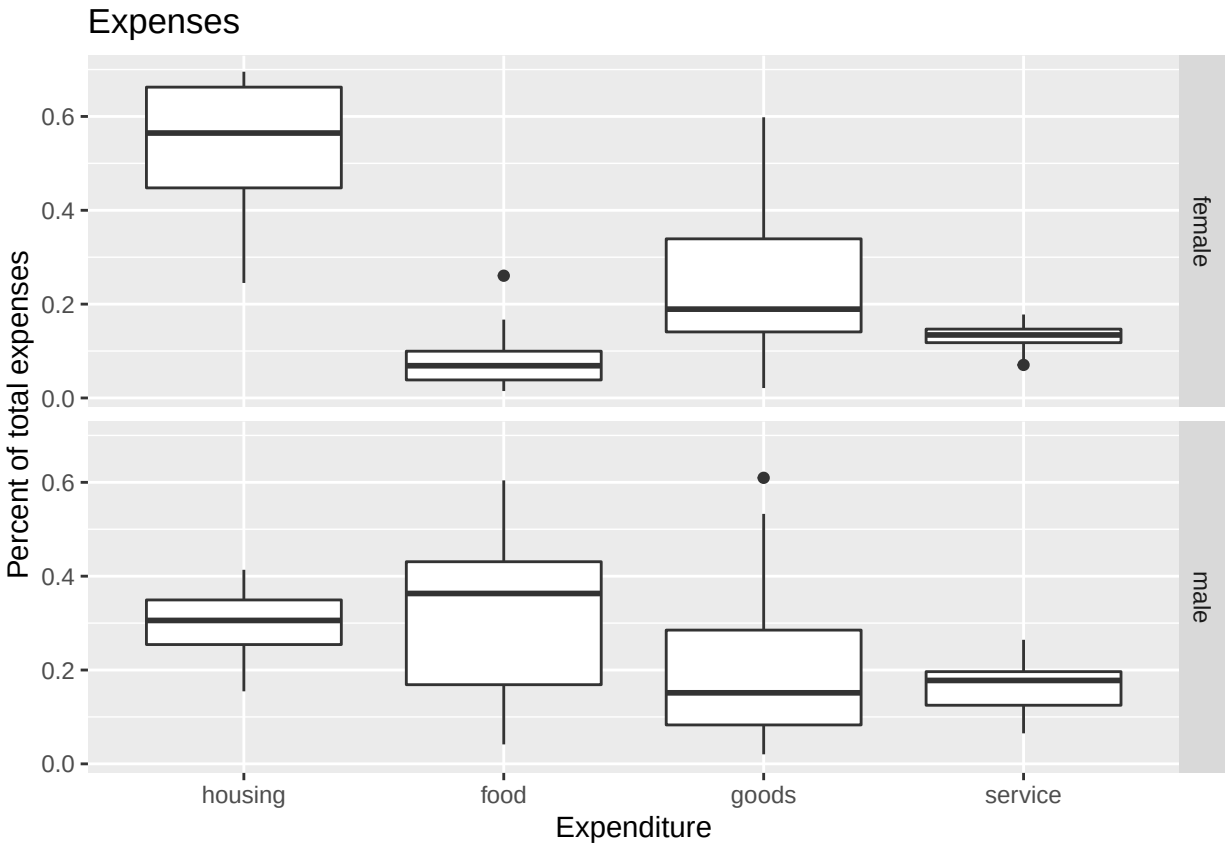
- Discussion: First, the mean sales of each country is calculated using *tapply()*. Than, the Forbes data set is subset to a new dataframe that includes only those companies that have profits greater than 5 billion. Similarly to the Bermuda problem, the summary of the country column is taken and this gives a vector of countries and the count of companies within those countries that have profits greather than 5 billion. In order to see them together in the same table, each vector was converted to a data frame and then merged together based on the row name which was the name of the country. Following the merging of the data frames, the rows were renamed appropriately, and then sorted by mean sales first, than companies with profits greater than 5 billion.
  - There weren't really any complications doing this. was pretty straight forward other than mean sales defaulting to an array after being calculated rather than just a vector, which then had to be converted to numeric in the data frame in order to sort by that column.
  - An interesting piece of information from the final table is that the united states has the most companies with profits greater than 5 billion, but they don't have the highest average sales. The country with the highest average sales is actually the Netherlands/United Kingdom with 92.1 billion while the United States has 20 businesses with profits greater than 5 billion but the average sales of companies from the United States on the Forbes list is only 10.1 billion.
  - I think it would be important to note that the average sales in a country seems to not fully related to the number of companies from that country that were able to pull in more than 5 billion in sales. For example, there were 14 countries that had higher average sales but had less companies with profits greater than 5 billion than the United States. Seems like an interesting insight into the data.
  - Just like in a previous example, it was kind of expected that the United States would have the highest number of companies with profits greater than 5 billion dollars, however, it was unexpected that the mean sales of companies in the US would be lower than other countries. This might be due to a fairly large skew in the data towards lower mean sales since there were so many that are from the US in the data set that can be seen on the graphs above.

6. Question 2.1, pg. 39 in Handbook (see Chapter 6 of R Graphcis Cookbook for GGPlot)

- Statement/Question: Data was collected from 20 males and 20 females about their expenditures on housing, food, goods, and services. We want to look at 2 different things: 1. The split of the 4 different spending groups based on the total expenditure (Do those that spend more over all have a different spending habit or is the distribution fairly similar to those that spend less); 2. Does the relationship between expenditures differ between the males and females?


Expenses of Males
Percent of total expenses
0.0
0.1
0.2
0.3
0.4
0.5
0.6
housing
goods
Expenditure


Expenses of Females
Percent of total expenses
0.0
0.2
0.4
0.6
housing
goods
Expenditure

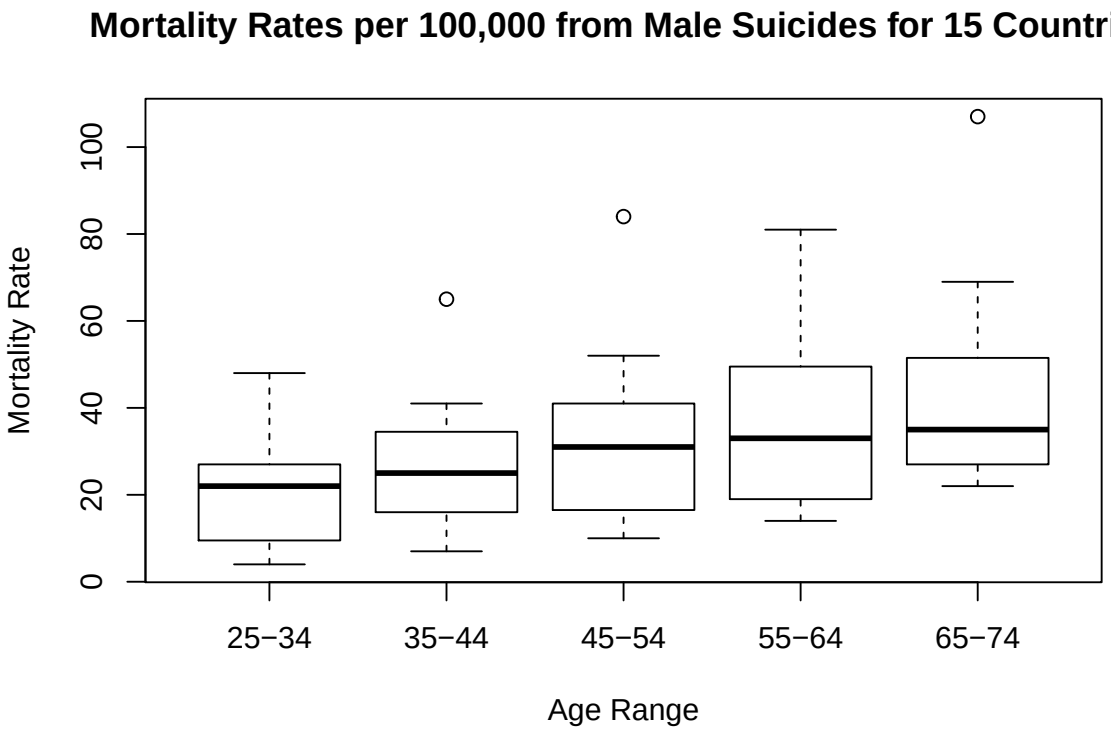


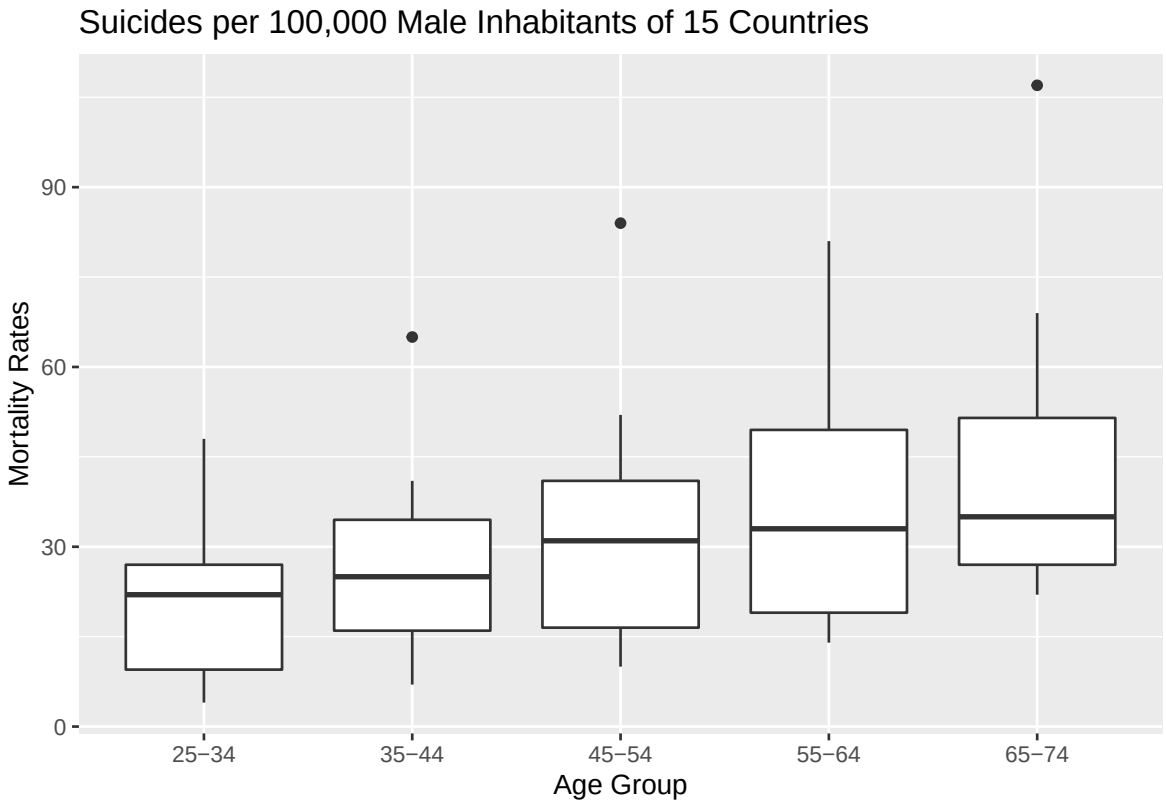
- Discussion: The data was already preloaded into R under *household*. In order to get all of the expenditures to have similar meaning, I think they should be divided by the total expenses to get a proportion. Then the total column is deleted so all that is left is the proportion of the total that each expense is per person. Since I don't know how to make graphs with each of the columns being a different variable, I shifted the data from a wide format to a long format which allows the selection of a column to be automatically split based on factor level. Then each expense is plotted using the base R *boxplot()* function and a *ggplot()* function.
- For the first question that we wanted answered, if the expense on commodity group depends on the total expenses, we can see that for some categories like the food and service commodities for females doesn't really change in proportions no matter how much they spend over all, where the housing and goods categories do change a fair amount; females spent between about 25% and 70 percent on housing while spending close to 0% to 18% (with the exception of 1 data point that is an outlier) on food. Males also had shown some differences. Males were fairly consitent in the percent of their spending that went to housing and services while being really variable in their food and goods expenditures.
- For the second question, differences in males and females can be seen in the housing and food categories. Men appeared to be more consistent in their housing spending and higher variation in their food spending while females tended to be more consistent in their food spending, but more variable in their housing spending. However, both males and females had a large variation in their goods spending depending on how much they spent overall and both were rather consistent in their service spending.
  - A difficulty was getting the graphs to look right with them being on the same 'plate'. The *boxplot()* function doesn't look very good when they are plotted over top of each other, while ggplot does. The function from ggplot also makes it easier to visualize when gender is stacked rather than side by side. Also, figuring out that I couldn't (or didn't know how) to use the same data frame for both the base R and the ggplot2 functions to make similar output took some time. I referred back to the lecture **.R** file for the ggplot problem since the R Graphics Cookbook didn't help.
  - I think it is important that the proportions of total expenses of males and females differ in Housing

and food. It is also really interesting that goods and services were similar in variation between males and females. From the way the plots were calculated, it shows that males and females are pretty consistent in spending about 15% of their total expenditures on services.

7. Question 2.3, pg. 41 in Handbook (see Chapter 6 of R Graphcis Cookbook for GGPlot)

- Statement/Question: Construct side-by-side box plots for the data for the different age groups in the *suicide2* data frame and comment on what the plot tells me about the data.





- Discussion: To make the plots, I used the same methods as the previous problem. The *suicides2* data frame was converted from wide to long so that all of the age groups could be plotted on the same plot. Both base R and ggplots are made using this long version of the data frame. For the X-axis labels, I changed the column names to have an easier to understand format; i.e. A25.34 was changed to 25-34. From the plot, the mortality rate increases as age increases. Also, the distribution of mortality per age group decreases from 25-34 to 35-44, but than increases again to 45-54, decreases at 55-64, than increases again to 65-74. However, 35-44, 45-54, and 65-74 contain an outlier on the high end of mortality rates. The age groups 25-34 and 65-74 are skewed when looking at the median.
  - Using question 6's method for question 7, there weren't any complications in this problem. Once converting from wide to long was figured out, the plotting of the boxplots was fairly simple.
  - The results are very unexpected. I would have expected the 25-34 age range would have been the highest with the amount of change that happens at these ages. However, it would be interesting to explore reasons behind the high mortality rate in older ages.

8. Using a single R expression, calculate the median absolute deviation, $1.4826 \cdot median|x - \mu|$, where μ is the sample median. Use the dataset **chickwts**. Use the R function mad() to verify your answer.

```
## [1] 91.9212
```

```
## [1] TRUE
```

- Discussion: This problem was fairly straight forward. A new column was created with the value of the **x - μ** was created and than the median of that column was found using *median()*. This value was than multiplied by the given 1.4826 and then reported. The boolean operator = was used make sure that this value was the same value found using the *mad()* function. **TRUE** means the values are the same.

No quesiton number 9?

10. Using the data matrix **state.x77**, find the state with the minimum per capita income in the New England region as defined by the factor *state.division*. Use the vector *state.name* to get the state name.

```
## [1] "Maine"
```

- Discussion: To work with the state.x77 matrix, I converted it to a data frame and then added columns for the division and state name. The data frame was than subset to include only those states in the division "New England". The new data frame was than sorted by a calculation of income divided by population and the new column sorted from smallest to largest. From the sorted data frame, the first name in the column "name" was pulled and reported to be Maine. Maine has a per capita income of $3694,

11. Use subscripting operations on the dataset **Cars93** to find the vehicles with highway mileage of less than 25 miles per gallon (variable *MPG.highway*) and weight (variable *Weight*) over 3500lbs. Print the model name, the price range (low, high), highway mileage, and the weight of the cars that satisfy these conditions.

```
##         Model Min.Price Max.Price MPG.highway Weight
## 16 Lumina_APV      14.7      18.0          23   3715
## 17      Astro      14.7      18.6          20   4025
## 26    Caravan      13.6      24.4          21   3705
## 28    Stealth      18.5      33.1          24   3805
## 36   Aerostar      14.5      25.3          20   3735
## 48        Q45      45.4      50.4          22   4000
## 49      ES300      27.5      28.4          24   3510
## 50      SC300      34.7      35.6          23   3515
## 56        MPV      16.6      21.7          24   3735
## 63   Diamante      22.4      29.9          24   3730
## 66      Quest      16.7      21.5          23   4100
## 70 Silhouette      19.5      19.5          23   3715
## 87     Previa      18.9      26.6          22   3785
## 89    Eurovan      16.6      22.7          21   3960
```

- Discussion: Similarly to question 10, the *Cars93* data was subset to include data that includes highway MPG less than 25, and a weight greater than 3500 pounds. Following subsetting of the data, the model name, min and max price, the vehicles highway MPG, as well as the vehicle weight were printed out.

12. Form a matrix object named **mycars** from the variables *Min.Price, Max.Price, MPG.city, MPG.highway, EngineSize, Length, Weight* from the **Cars93** dataframe from the **MASS** package. Use it to create a list object named *cars.stats* containing named components as follows:

    a) A vector of means, named *Cars.Means*

    b) A vector of standard errors of the means, named *Cars.Std.Errors*

```
## $Cars.Means
##   Min.Price   Max.Price    MPG.city MPG.highway  EngineSize      Length
##   17.125806   21.898925   22.365591   29.086022    2.667742  183.204301
##      Weight
## 3072.903226
##
## $Cars.Std.Errors
##   Min.Price   Max.Price    MPG.city MPG.highway  EngineSize      Length
##   0.9069210   1.1438051   0.5827473   0.5528742   0.1075695   1.5141964
##      Weight
##  61.1694186
```

- Discussion: In order to calculate the *Cars.Means* vector, *colMeans()* was used. However to calculate the standard error of the column means, a temporary vector was created using the *Cars.Means* vector. Each column was looped over in order to calculate the standard error of the mean wish the value being written to the appropriate place in the vector. The new vector was then added to the *Cars.Stats* list

under *Cars.Std.Errors* title.

13. Use the `apply()` function on the three-dimensional array **iris3** to compute:

    a) Sample means of the variables *Sepal Length, Sepal Width, Petal Length, Petal Width*, for each of the three species *Setosa, Versicolor, Virginica*

```
##            Sepal L. Sepal W. Petal L. Petal W.
## Setosa        5.006    3.428    1.462    0.246
## Versicolor    5.936    2.770    4.260    1.326
## Virginica     6.588    2.974    5.552    2.026
```

    b) Sample means of the variables *Sepal Length, Sepal Width, Petal Width* for the entire data set.

```
## Sepal L. Sepal W. Petal L. Petal W.
## 5.843333 3.057333 3.758000 1.199333
```

- Discussion: Margins: 1 is replication, 2 is descriptive metric, 3 is species. In order to find the mean of each metric by the species name, 2 margins have to passed to the apply function; 3 for the species, and 2 for the metric, with ***c(3,2)***. For the whole data set means

14. Use the data matrix **state.x77** and the `tapply()` function to obtain:

    a) The mean per capita income of the states in each of the four regions defined by the factor *state.region*

```
##     Northeast         South North Central          West
##      4570.222      4011.938      4611.083      4702.615
```

    b) The maximum illiteracy rates for states in each of the nine divisions defined by the factor *state.division*

```
##        New England    Middle Atlantic     South Atlantic
##          0.9166667          1.1666667          1.5000000
## East South Central West South Central East North Central
##          1.9500000          2.0000000          0.8000000
## West North Central           Mountain            Pacific
##          0.6285714          0.9500000          1.1400000
```

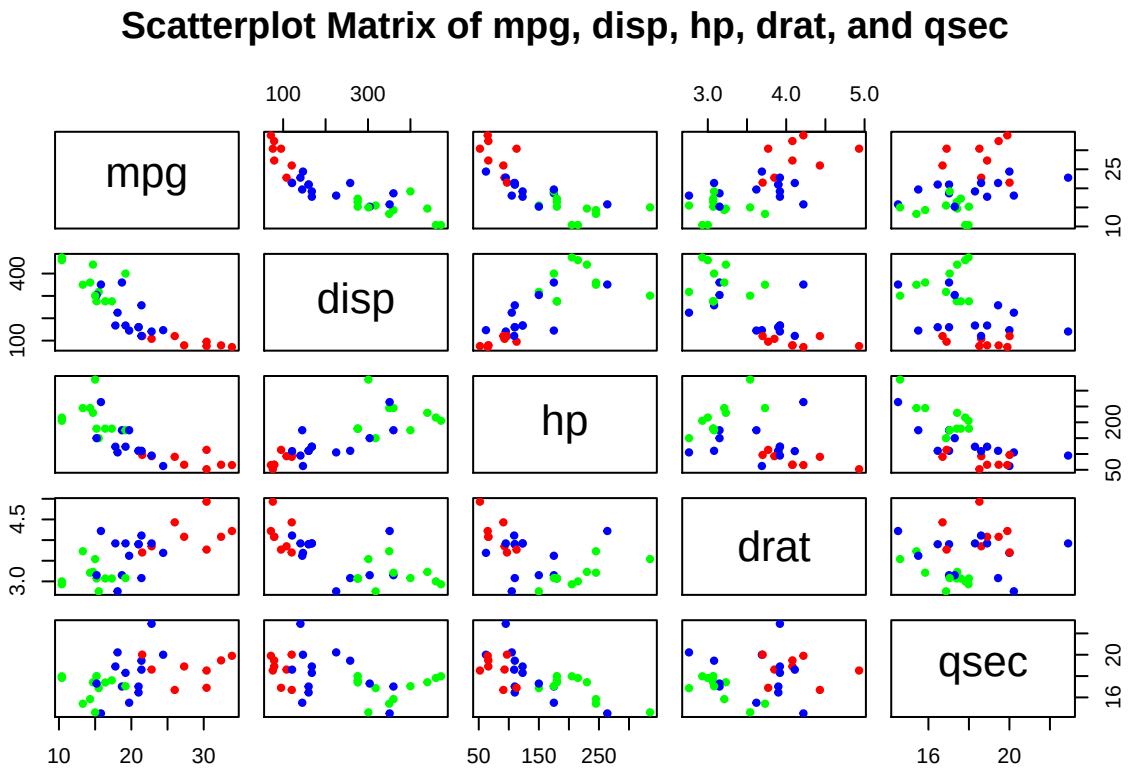
    c) The number of states in each region

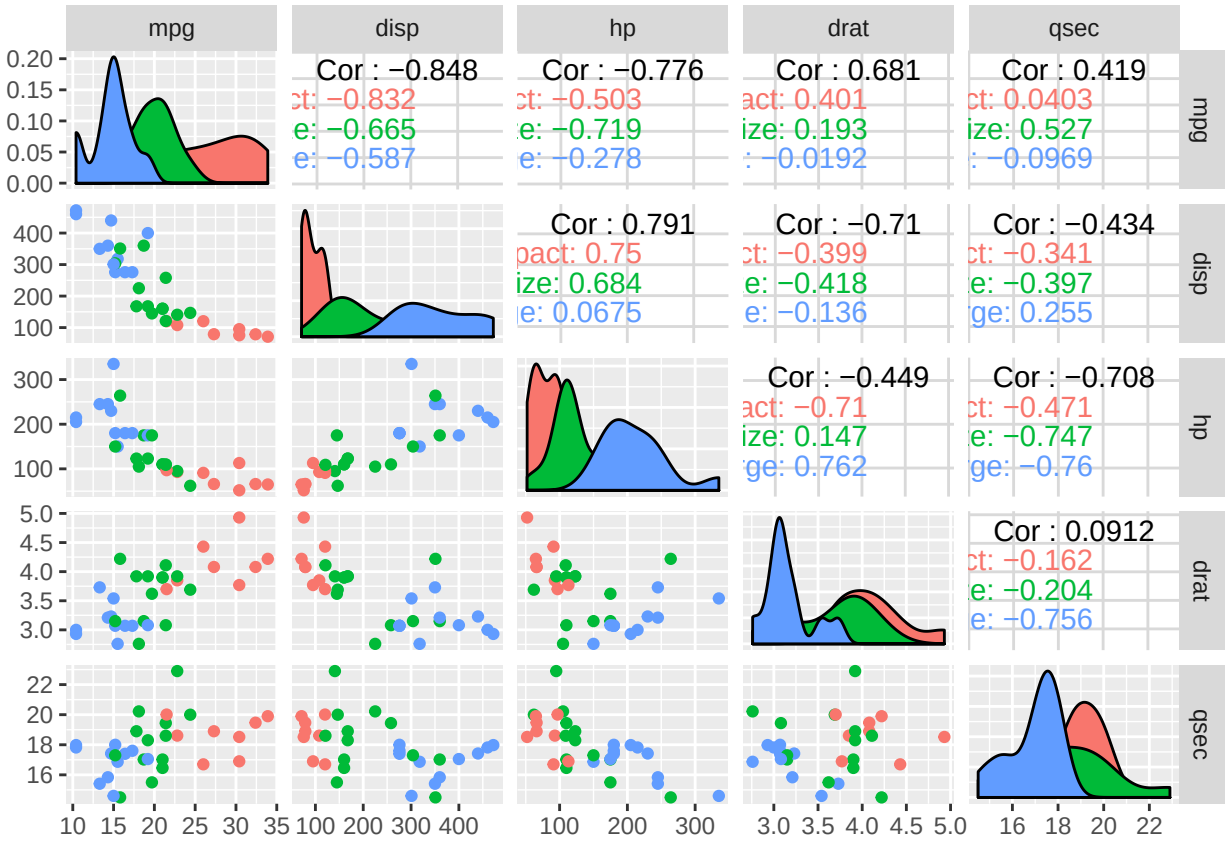
```
##     Northeast         South North Central          West
##             9            16            12            13
```

- Discussion: This was more simple thn the iris data, most likely since it's only working with 2 dimension. The average income for each region is stated in **a)**, after adding *state.region* to the *state* data frame used in Question 10.
- Illiteracy rate was also relatively easy since the data frame is already created. The illiteracy rate for the West North Central (our division) being the lowest was unexpected. I expected the illiteracy rate being the lowest in the New England division.
- In order to find the number of states in a particular region, I first tried to use the *sum()* function but found out that it wouldn't recognise an input of factors and then find the sum of them, so instead the unique function was used to find the different states in the *name* column based on the region, then the length of the unique names list was output.
  - The only complication found in this exercise was the writing of the function in ***c)***, however this was a quick fix. It might be important that the income of the West region being higher might not be related to the standard of living. It may also be important that the number of states in each region aren't evenly distributed, with the Northeast having the least number and the South having the greatest number.

15. Using the dataframe **mtcars**, produce a scatter plot matrix of the variables *mpg, disp, hp, drat, qsec*. Use different colors to identify cars belonging to each of the categories defined by the *carsize* variable in different colors.

```
carsize = cut(mtcars[,"wt"], breaks=c(0, 2.5, 3.5, 5.5),
+ labels = c("Compact","Midsize","Large"))
```





- Discussion: The plots were made with both base R (*pairs*) and *ggpairs*. The given code for carsize was used to create a new column of "size" and that column was used to set colors to red, blue, and green for the base R plot. The column of size was given directly to the *ggpairs* funciton and it chose the colors.
  - There looks to be a relationship between mpg and disp, mpg and hp, and disp and hp, which makes some sense; mpg goes down with more power and larger engine, and hp would increase with a larger engine.
  - The *ggpairs* plot is much more informative than the plot from base R because it provides more information like the correlation between the 2 continuous variables. The *ggpairs* also gives a density plot which is easier to observe than with just the scatterplot.

16. Use the function `aov()` to perform a one-way analysis of variance on the **chickwts** data with *feed* as the treatment factor. Assign the result to an object named *chick.aov* and use it to print an ANOVA table.

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## feed         5 231129   46226   15.37 5.94e-10 ***
## Residuals   65 195556    3009
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Discussion: The main source of variance in the data is between the different feeds, and not within individual feeds. The p-value states that we can reject the null hypothesis that the mean chick weight of 6 different feed types are equal, and conclude there some feed types produce higher weights in chicks. The assumptions that need to be made in order to do an analysis of variance is normality of the distribution and that the variances are equal.

17. Write an R function named `ttest()` for conducting a one-sample t-test. Return a list object containing the two components:

- the t-statistic named T;
- the two-sided p-value named P.

Use this function to test the hypothesis that the mean of the *weight* variable (in the **chickwts** dataset) is equal to 240 against the two-sided alternative. *For this problem, please show the code of function you created as well as show the output.*

```
#create t-test function
ttest = function(x, mu){
    m = mean(x)
    n = length(x)
    s = sd(x)
    t = (m - mu)/(s/sqrt(n))
    p = 2*pt(-abs(t), df=n-1)
    return(list("T" = t, "P" = p))
}

ttest(chickwts$weight, 240)
```

```
## $T
## [1] 2.299879
##
## $P
## [1] 0.02444107
```

- Checking if it matches *t.test()*

```
##
##  One Sample t-test
##
## data:  chickwts$weight
## t = 2.2999, df = 70, p-value = 0.02444
## alternative hypothesis: true mean is not equal to 240
## 95 percent confidence interval:
##  242.8301 279.7896
## sample estimates:
## mean of x
##  261.3099
```

- Discussion: In order to perform a one-sample t-test, an assumption being made is that the data is normally distributed. The 4 pieces of information needed to calculate the t-statistic in R is the mean, standard deviation, and length of the data array, as well as the value that we want to see if it is similar to. The t-statistic can than be calculated and with the t-statistic the pvalue can be calculated.
  - The only complication I had was trying to figure out how to do the p-value. The creation of the function was straight forward.
  - The p-value is below 0.05, at 0.02, which means that there is sufficient evidence to reject the null hypothesis that the overall chick weight mean is equal to 240.

Resources Used:

- StackOverflow
- rkabacoff.github.io/datavis/Customizing.html
- nabble.com
- cyclismo.org