CMDA-3654

Homework 4

Your name here

Due as a .pdf upload

Instructions:

Delete the Instructions section from your write-up!!

I have given you this assignment as an .Rmd (R Markdown) file.

- Change the name of the file to: `Lastname_Firstname_CMDA_3654_HW4.Rmd` , and your output should therefore match but with a `.pdf` extension.
- You need to edit the R Markdown file by filling in the chunks appropriately with your code. Output will be generated automatically when you compile the document.
- You also need to add your own text before and after the chunks to explain what you are doing or to interpret the output.
- Feel free to add additional chunks if needed. I **will not** be providing assignments to you like this for the entire semester, just long enough for you to learn how to do it for yourself.

Required: The final product that you turn in must be a .pdf file.

You MUST Knit this document directly to a PDF, you are not allowed to knit to any other file and then convert afterwards.

Be sure to include appropriate titles, axis names, and legends.

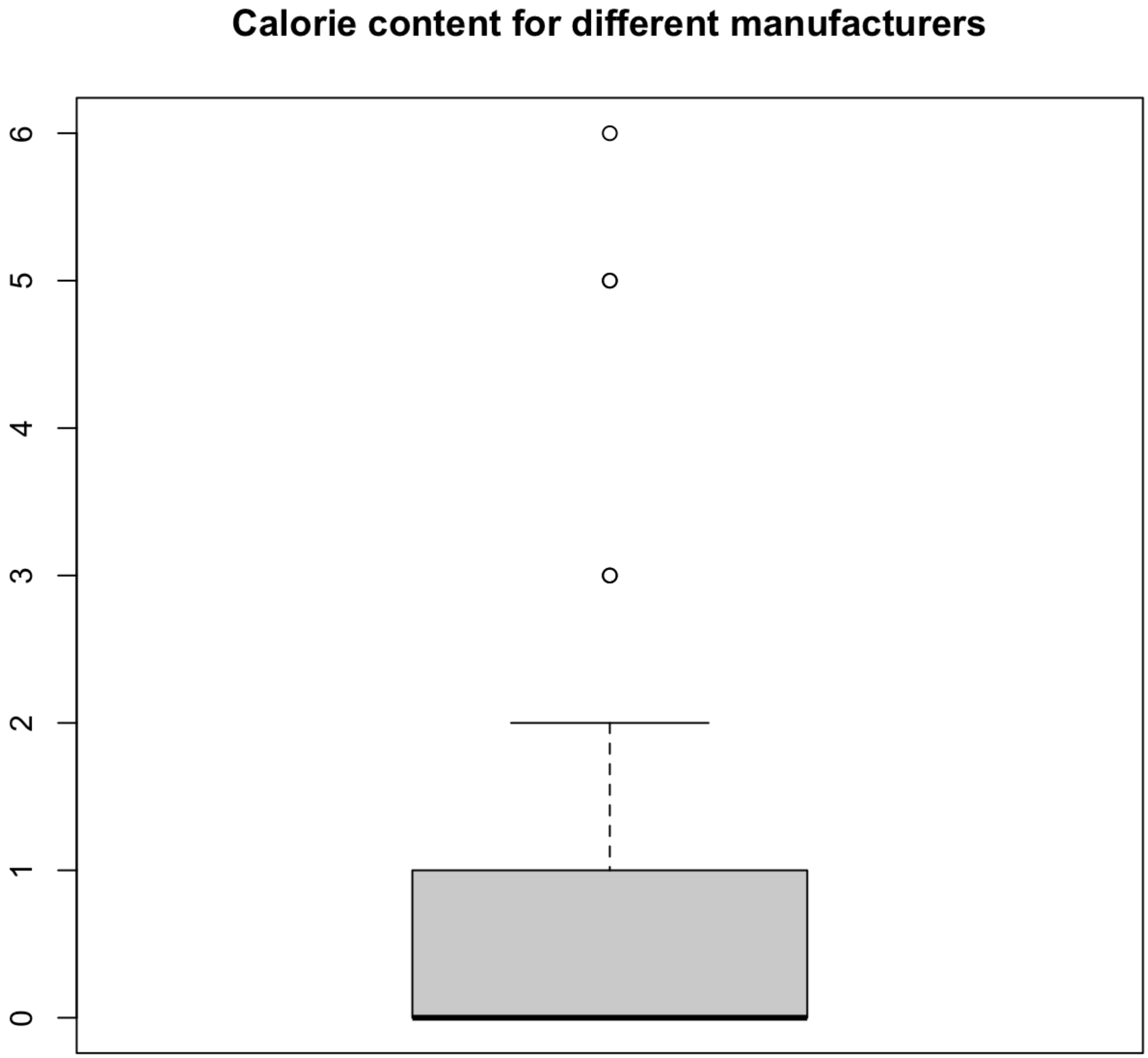
Problem 1: [70 pts] Plots using ggplot2

The plots must be made using ggplot2.

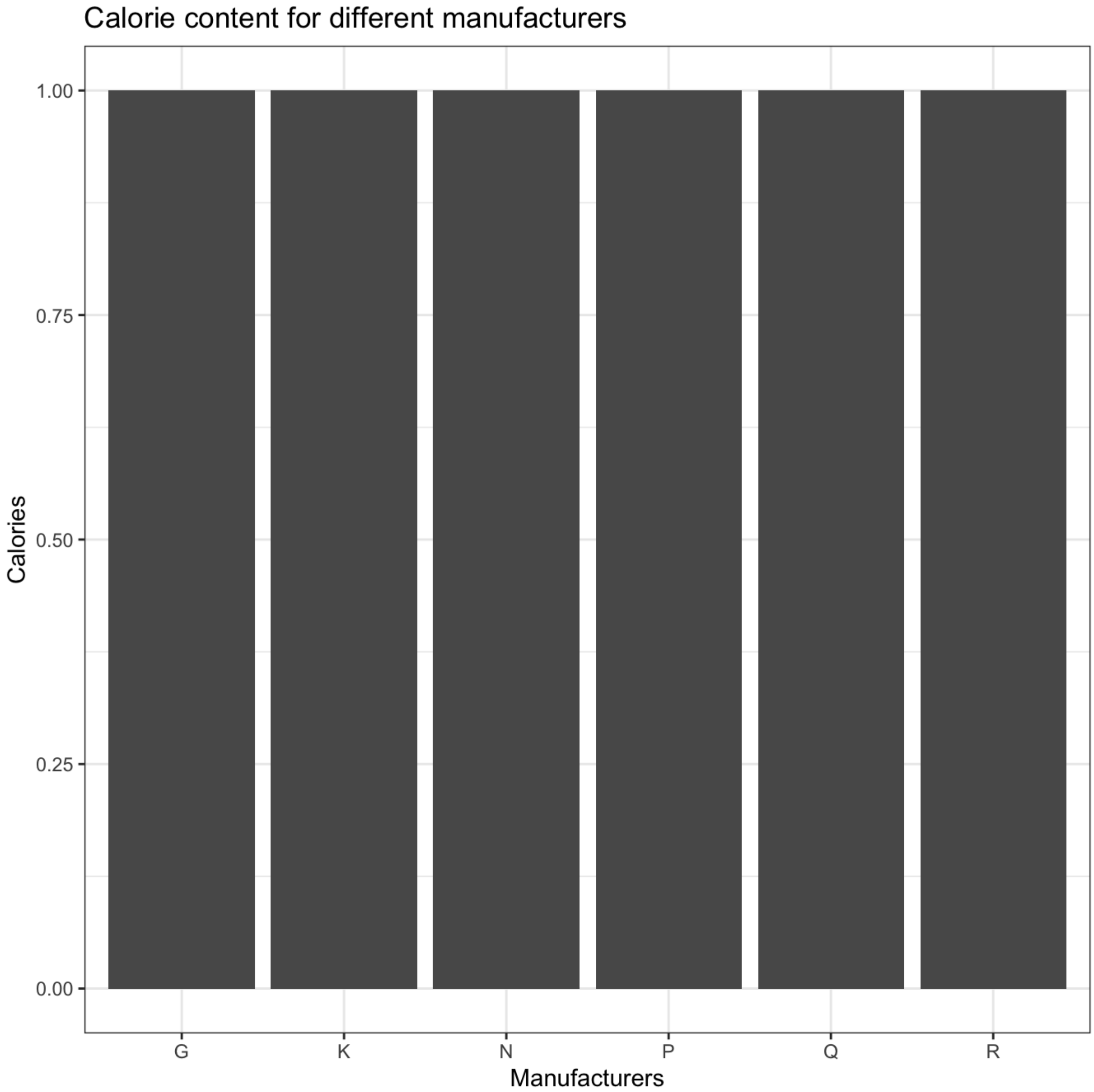
Load the `MASS` library. Within this library is a dataset called `UScereal` (you should read the help summary regarding this dataset). Note that `fibre = fiber` (due to British spelling).

a. Create side-by-side boxplots that display the calorie content for the different manufacturers. Make sure that the calories axis has breaks every 25 units apart. Flip the coordinates so that the boxplots are displayed horizontally. The different manufacturers should have different colors and their full name should be provided on the axis. Finally, overlay a jittered stripchart on top of the boxplots as well. If you are having trouble fitting everything on your html/pdf properly, you can turn off the legend or put it underneath the plot.

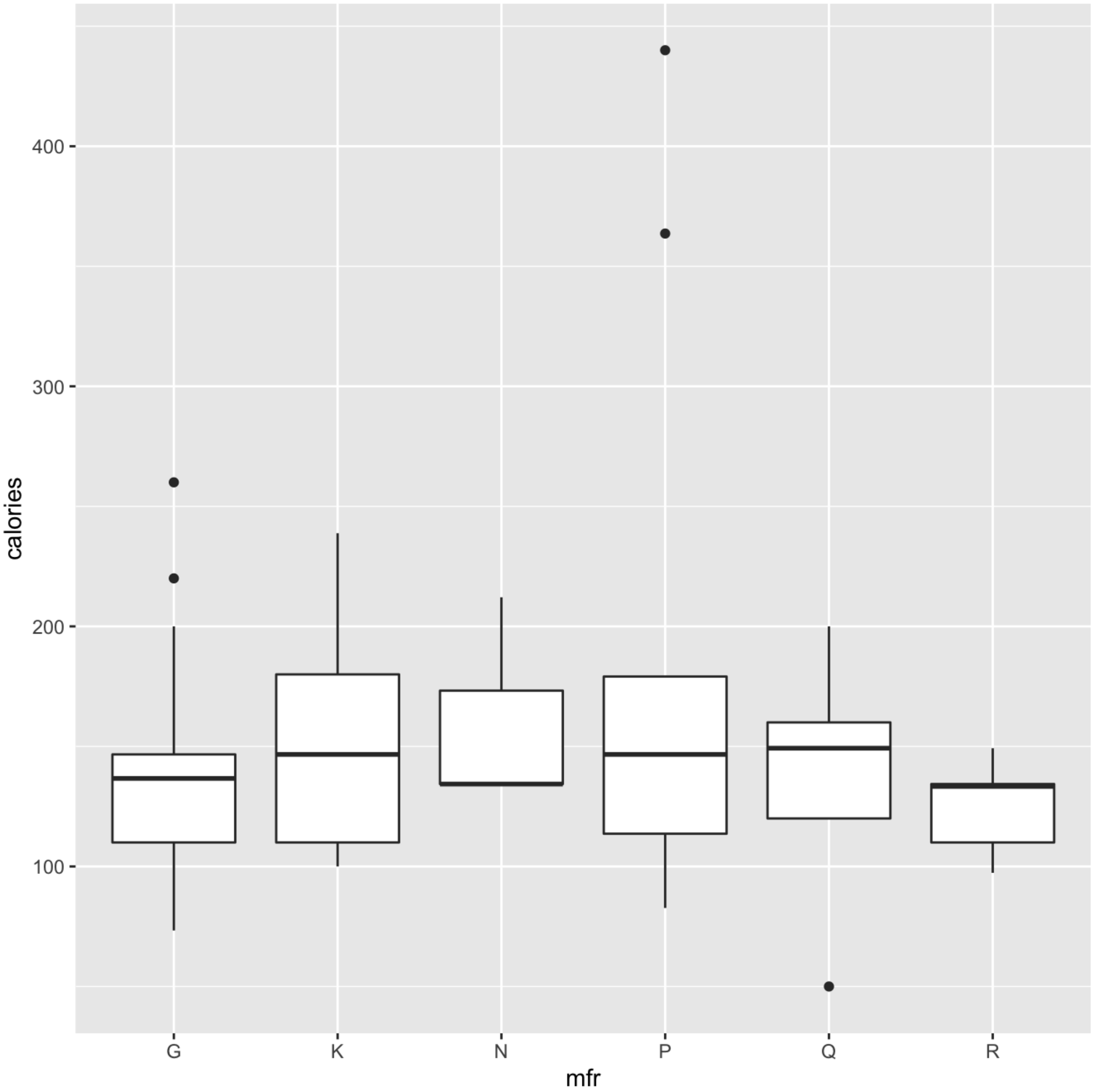
```
library(MASS)
data("UScereal",package = "MASS")
library(ggplot2)
data("ggplot",package = "ggplot2")

Manufac<-table(UScereal$mfr,UScereal$calories)

freq<-Manufac/rowSums(Manufac)
boxplot(Manufac,main="Calorie content for different manufacturers" )
```



```
ggplot(UScereal) + theme_bw() +
  geom_bar(mapping = aes(x = mfr, fill = calories), position = "fill") +
  labs(x = "Manufacturers", y = "Calories",
       title = "Calorie content for different manufacturers") +
  scale_fill_manual(values = c("cyan", "orange"))
```

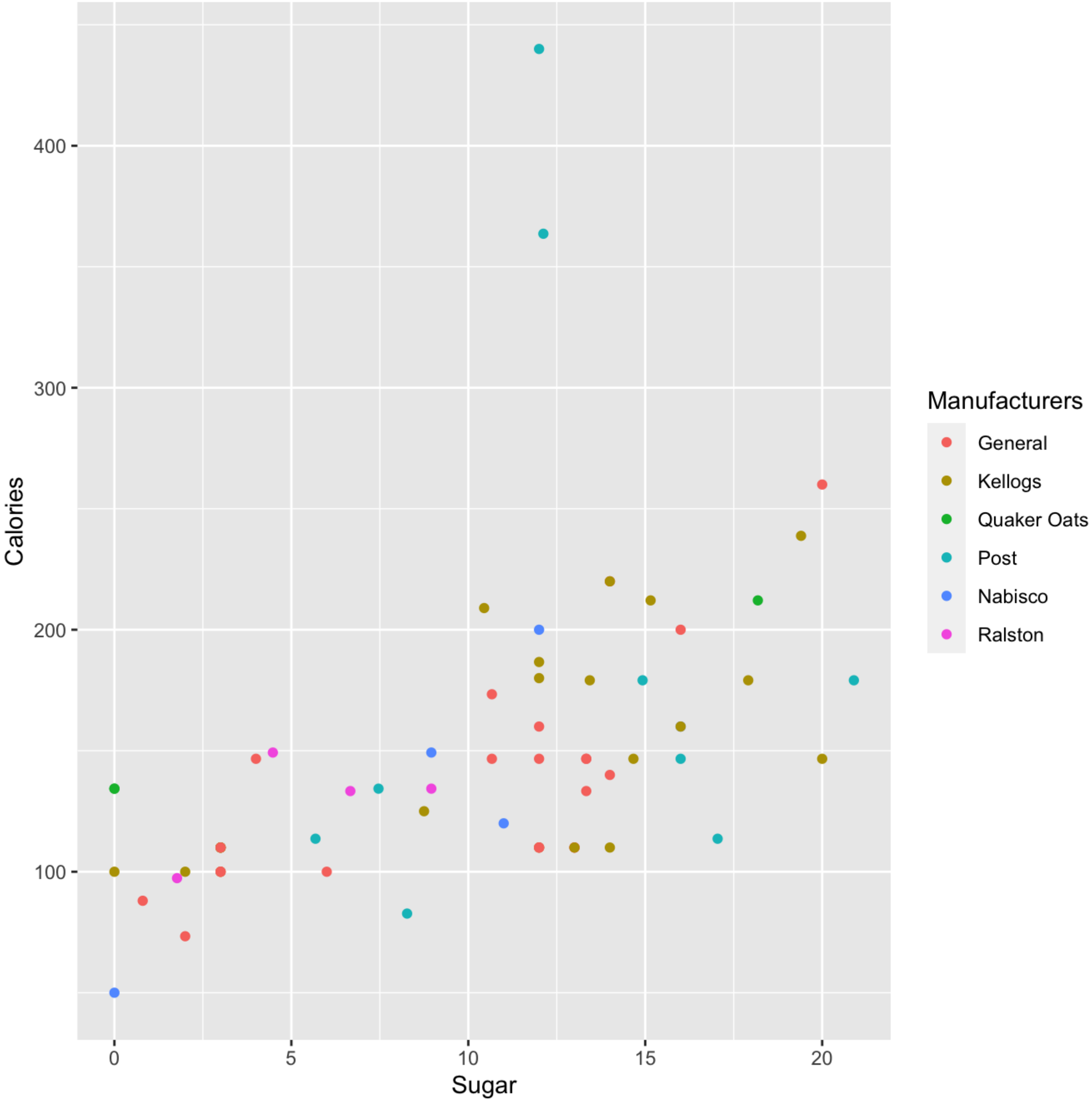


```
ggplot(data = UScereal,
       aes(x = mfr, y = calories), na.rm=T, orientation=H, coef=0.25, values=c("green"))
+
  stat_boxplot()
```



b. Create a scatter plot of calories versus sugars. Color and shape the points according to the different manufacturers. Provide the actual names of the manufacturers in the legend, not just the single letter label. Does the number of calories seem to be correlated with sugar content? Do you notice any interesting patterns or unusual points in the plot?

```
#Scatter ggplot
ggplot(data=UScereal,aes(y=calories,x=sugars,color=mfr))+
  labs(y="Calories",x="Sugar", tittle="Calories vs Sugar by Manufacturers")+ geom_point
() +
  scale_color_discrete(name="Manufacturers",labels=c("General","Kellogs","Quaker Oats",
"Post","Nabisco","Ralston"))
```



c. People often use cereal to immediately obtain fiber first thing in the morning. Let’s consider cereal that has at least 4 grams of fiber per serving to be classified as a “high fiber” cereal. First provide a table that summaries the min, median, and max fibre contents for each manufacturer. Then generate a two-way table that summarizes the median sugar content for all cereals depending on if they are “high fiber” versus “lower fiber” and by vitamin content.

```
#List of manufacturers
Marcas<-c("G","K","Q","P","N","R")
#Create matrix
Matrize<-matrix(ncol=3,nrow=0)
#for loop to get each manufacturer in the matrix
for(i in Marcas){
  category<-subset(UScereal,mfr==i)
  valor<-round(c(max(category$fibre),median(category$fibre),min(category$fibre),2))
  Matrize<-rbind(Matrize,valor)
}
#Naming labels
colnames(Matrize)<-c("Maximum Fibre","Median Fibre", "Minimun Fibre")
rownames(Matrize)<-c("General","Kellogs","Quaker Oats","Post","Nabisco","Ralston")

Matrize
```

```
            Maximum Fibre Median Fibre Minimun Fibre
General                 5            2             0
Kellogs                28            1             0
Quaker Oats             4            1             0
Post                   12            7             0
Nabisco                30            6             4
Ralston                 6            1             0
```

```
#subsetting sugar and fibres
low_fibre_high_vitamin<-subset(UScereal,UScereal$fibre<4&UScereal$vitamins=="100%")
high_fibre_low_vitamin<-subset(UScereal,UScereal$fibre>=4&UScereal$vitamins=="enriched")
low_fibre_no_vitamin<-subset(UScereal,UScereal$fibre<4&UScereal$vitamins=="none")
high_fibre_high_vitamin<-subset(UScereal,UScereal$fibre>=4&UScereal$vitamins=="100%")
low_fibre_low_vitamin<-subset(UScereal,UScereal$fibre<4&UScereal$vitamins=="enriched")
high_fibre_no_vitamin<-subset(UScereal,UScereal$fibre>=4&UScereal$vitamins=="none")

#Create matrix for second table
Matrize_2<-matrix(nrow=2,ncol=3)
rownames(Matrize_2)<-c("High fibre","Low fibre")
colnames(Matrize_2)<-c("100%","enriched","none")

#Plotting points
Matrize_2[1,1]<-median(low_fibre_high_vitamin$sugar)
Matrize_2[1,2]<-median(low_fibre_low_vitamin$sugar)
Matrize_2[1,3]<-median(low_fibre_no_vitamin$sugar)
Matrize_2[2,1]<-median(high_fibre_high_vitamin$sugar)
Matrize_2[2,2]<-median(high_fibre_low_vitamin$sugar)
Matrize_2[2,3]<-median(high_fibre_no_vitamin$sugar)

Matrize_2
```

```
           100% enriched none
High fibre    3     11.5    0
Low fibre    14     14.0    0
```

d. Which visual method(s) would be the most useful for determining whether cereals with more sugar are being marketed more towards children? There are a few acceptable answers, but please show whatever visuals you like to investigate this situation and make your argument.

```
#I believe a classic baxplot would be the most useful since on the x axis I could plot t
he ages on the x axis and the sugar quantities on the y. Thic could potentially show the
amount of sugar by age group and we could see if it is true that cereals with more sugar
target children.
```

e. A common belief is that low-fat foods often contain added sugar in order to taste better, which negates any true health benefits. A counter argument is that fat is an essential nutrient so it's not necessarily a bad thing to have it in your diet in moderation.

Investigate this dataset based upon all of the variables in the dataset. Use summaries, tables, and whatever plots you feel are necessary for investigating whether we can determine which cereals might actually be healthy. Is there a manufacturer that tends to produce healthier cereal than the others? This is an incredibly open-ended question but do your best to be creative and give a powerful argument. Do not simply reuse the scatterplot from part (b). Feel free to reference outside material to support your answers, such as Daily Recommended Intake information.

```
#For this I used Tableau and have attached the tables I used to analyze this data since
 it say to use whatever plots feel necessary for the investigation.
#After extensive analysis it seems that the all bran brand is the healthiest option yet,
it is important to note that the all-bran with extra fiber is not available on shelf. It
is one of the healthiest options since it has more vitamins than any other cereal and al
so has a lot of protein, potasseoum, fiber and a lot of fat. As a person that is into nu
trition a lot, fat and protein are the most essential items for breakfast, that is why e
ggs are the best breakfast. I also found out by analyzing the data that there is mistake
on it since, it is stating that: Honey Cumb, Apple Jacks, Fruit Loops and Cocoa Puffs ha
ve the least amount of sugar out of all the available manufacturers. But, after looking
 into the nutrition value of this specific cereals and comparing it to the All Bran, the
re is indeed a mistake in the dataframe. Grape-Nuts woul've been the healthiest option i
f it weren't for the amount of sugar it has.
```

Problem 2: [30 pts] Area graphs using ggplot2.

a. Load the `uspopage` dataset in R (found in the `gcookbook` library), and describe the dataset in your own words, in 2-3 lines.

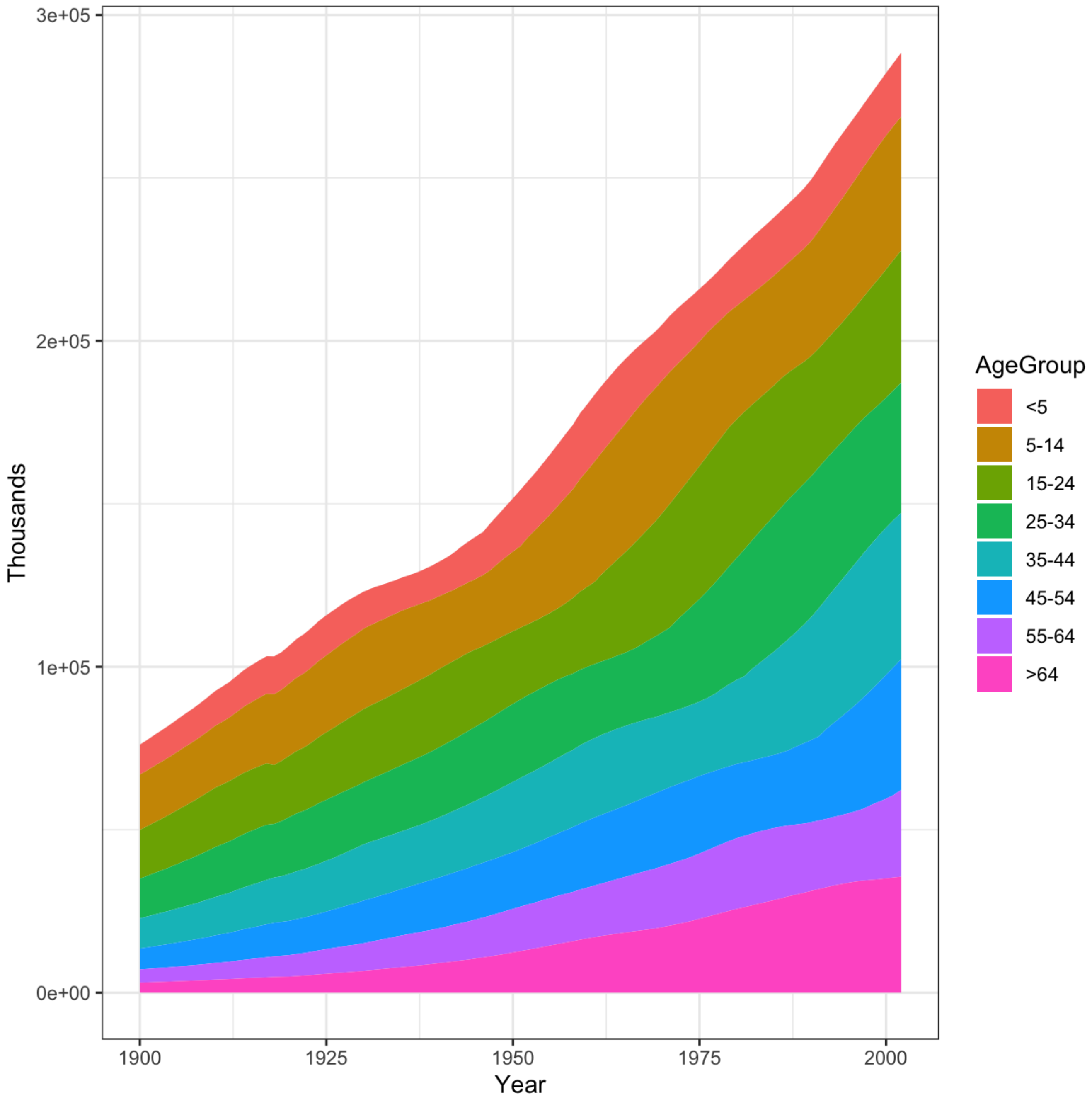
```r
library(gcookbook)
data("uspopage",package = "gcookbook")
#This dataset is the estimated values by U.S Census which includes age group, year, and
 thousands: number of people in thousands
```

b. Construct a stacked area graph with Year in the x-axis, population (in thousands) in the y-axis, and different age groups in different layers.

Making a stacked area graph is new to us, but it’s easy to make using `ggplot()` as you can see in the following link.

https://www.r-graph-gallery.com/136-stacked-area-chart/ (https://www.r-graph-gallery.com/136-stacked-area-chart/)

```r
ggplot(uspopage,aes(y=Thousands,x=Year, fill=AgeGroup))+geom_area()+theme_bw()
```



c. Next, for each Year, compute the contribution from each age group to the total population as a fraction of the total population. That is, what proportion of the total population does each age group make up for each year. Store this is a new data frame. We don't want to see the whole data frame as an output, so just show the years 1946, 1955, 1972.

Look at the small example to see what it should look like. I purposely did this a very inefficient way for the Year == 1900 only. Your code act upon the entire data frame automatically!

```
library(gcookbook)
data("uspopage")

# I'm only getting the year 1900
my1900 <- uspopage[uspopage$Year == 1900, ]

# Compute the proportion of the total population each age group makes up.
prop_of_total <- my1900$Thousands / sum(my1900$Thousands)

# Store in new data frame.
my1900 <- data.frame(my1900, prop_of_total)
print(my1900)
```

```
  Year AgeGroup Thousands prop_of_total
1 1900       <5      9181    0.12065340
2 1900     5-14     16966    0.22296107
3 1900    15-24     14951    0.19648067
4 1900    25-34     12161    0.15981549
5 1900    35-44      9273    0.12186243
6 1900    45-54      6437    0.08459274
7 1900    55-64      4026    0.05290825
8 1900      >64      3099    0.04072594
```

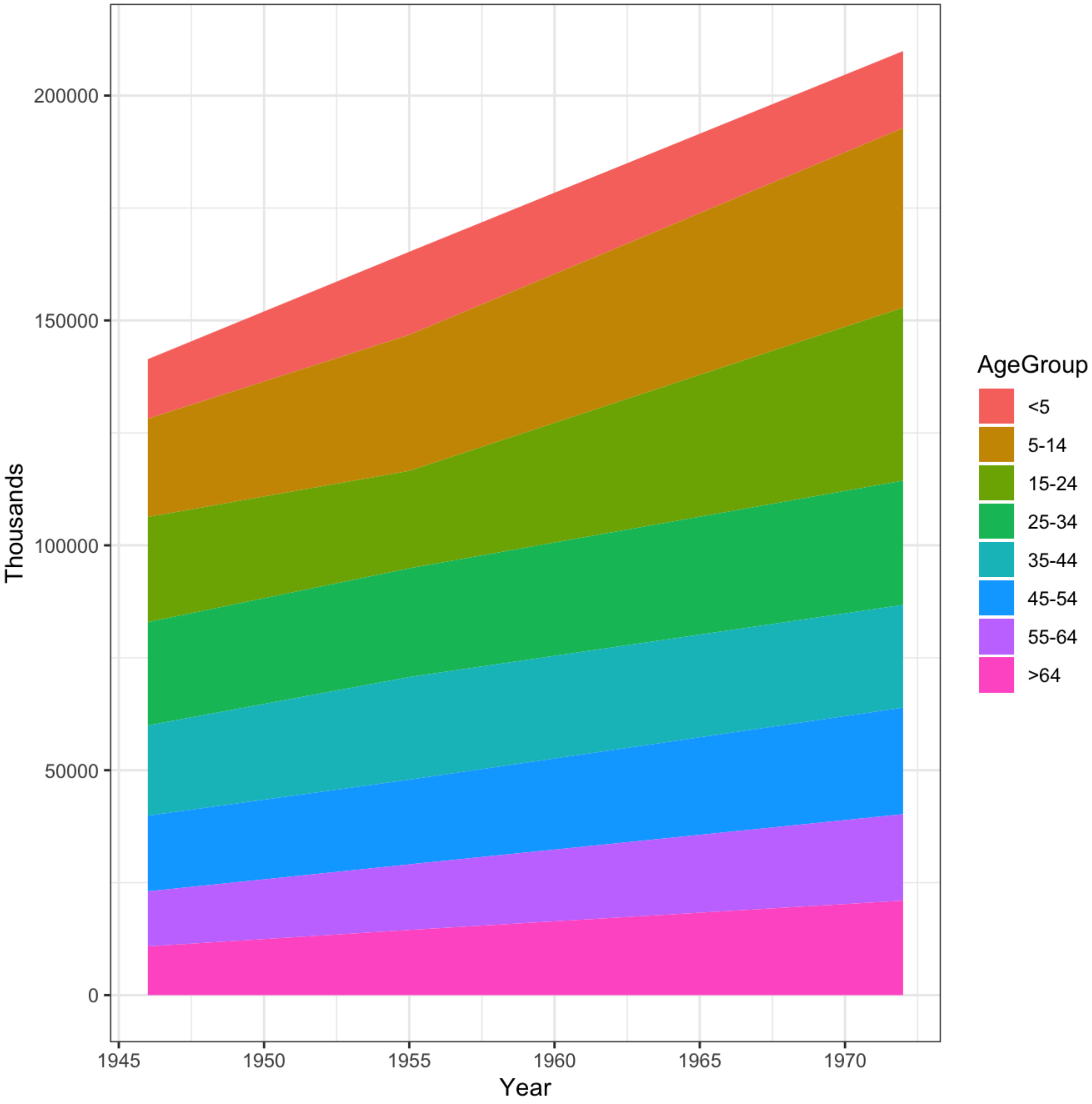
```
my_prop<-function(uspopage){
  #Creating matrix to be able to store the desired years
  Pmatrize<-matrix(ncol=4,nrow=0)
#Making list of years as numerical
  num_years<-as.numeric(levels(as.factor(uspopage$Year)))
  for(i in num_years){
    for_year<-uspopage[uspopage$Year==i,]
    section<-for_year$Thousands/sum(for_year$Thousands)
    selected_year<-cbind(for_year,section)
    Pmatrize<-rbind(Pmatrize,selected_year)

}
return(Pmatrize)
}

#display all selected years
display<-subset(my_prop(uspopage),Year==1946|Year==1955|Year==1972)
display
```

	Year <int>	AgeGroup <fct>	Thousands <int>	section <dbl>
369	1946	<5	13244	0.09367065
370	1946	5-14	21844	0.15449575
371	1946	15-24	23382	0.16537354
372	1946	25-34	22954	0.16234643
373	1946	35-44	20073	0.14197003
374	1946	45-54	16820	0.11896258
375	1946	55-64	12244	0.08659797
376	1946	>64	10828	0.07658304
441	1955	<5	18467	0.11173566
442	1955	5-14	30248	0.18301729

1-10 of 24 rows Previous **1** 2 3 Next

d. Finally, plot the proportional stacked area graph. This can be done two ways. It can be done using the data frame from (c) or it can be done using the right extra option in `geom_area()` on the original data frame.

```
ggplot(display,aes(y=Thousands,x=Year, fill=AgeGroup))+geom_area()+theme_bw()
```



e. Are there any interesting trends in the data?

Problem 3: [10 pts **Extra Credit**] Using the `GGally` library.

The `GGally` library has some interesting tools. https://ggobi.github.io/ggally/ (https://ggobi.github.io/ggally/)

Many of these tools may seem too advanced at the moment, but the two that make scatterplot matrices are quite nifty.

Redo problem 1e from HW3 but this time using a function from `GGally` . That is, make a scatterplot matrix using the `ggpairs()` (or `ggscatmat()`) function for all numeric variables. Color the points in the scatterplot matrix using different colors depending on race. A legend should be easy to make in this case (should be automatic).

Reminder: The data is `birthwt` from the `MASS` library.