STAT3355(HW-4)

Tulasi Janjanam

2024-10-22

Problem 1

```
library(ggplot2)

mobile_data <- read.csv("/Users/tulasijanjanam/Downloads/archive (4)/train.csv")
head(mobile_data)
```

```
##   battery_power blue clock_speed dual_sim fc four_g int_memory m_dep mobile_wt
## 1           842    0         2.2        0  1      0          7   0.6       188
## 2          1021    1         0.5        1  0      1         53   0.7       136
## 3           563    1         0.5        1  2      1         41   0.9       145
## 4           615    1         2.5        0  0      0         10   0.8       131
## 5          1821    1         1.2        0 13      1         44   0.6       141
## 6          1859    0         0.5        1  3      0         22   0.7       164
##   n_cores pc px_height px_width  ram sc_h sc_w talk_time three_g touch_screen
## 1       2  2        20      756 2549    9    7        19       0            0
## 2       3  6       905     1988 2631   17    3         7       1            1
## 3       5  6      1263     1716 2603   11    2         9       1            1
## 4       6  9      1216     1786 2769   16    8        11       1            0
## 5       2 14      1208     1212 1411    8    2        15       1            1
## 6       1  7      1004     1654 1067   17    1        10       1            0
##   wifi price_range
## 1    1           1
## 2    0           2
## 3    0           2
## 4    0           2
## 5    0           1
## 6    0           1
```

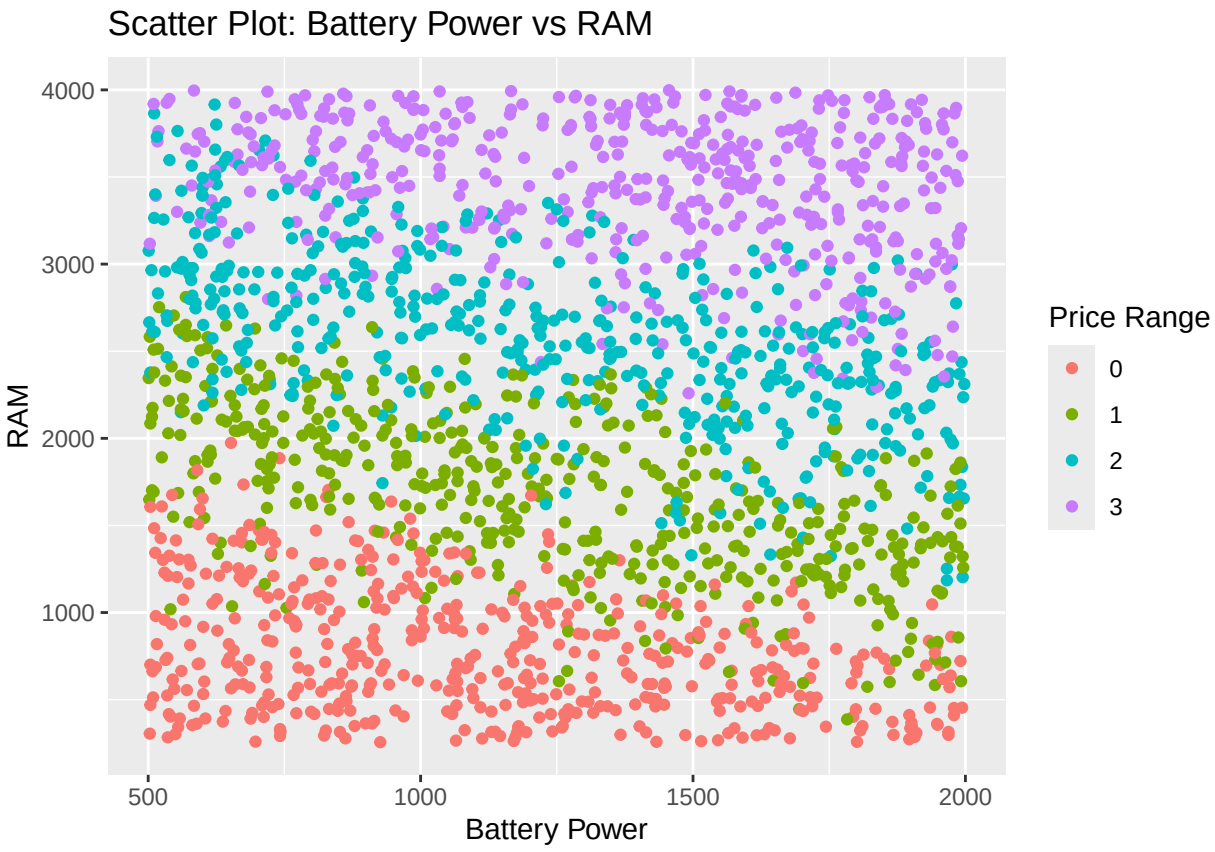
```
str(mobile_data)
```

```
## 'data.frame':    2000 obs. of  21 variables:
##  $ battery_power: int  842 1021 563 615 1821 1859 1821 1954 1445 509 ...
##  $ blue         : int  0 1 1 1 1 0 0 0 1 1 ...
##  $ clock_speed  : num  2.2 0.5 0.5 2.5 1.2 0.5 1.7 0.5 0.5 0.6 ...
##  $ dual_sim     : int  0 1 1 0 0 1 0 1 0 1 ...
##  $ fc           : int  1 0 2 0 13 3 4 0 0 2 ...
##  $ four_g       : int  0 1 1 0 1 0 1 0 0 1 ...
##  $ int_memory   : int  7 53 41 10 44 22 10 24 53 9 ...
```

```
##  $ m_dep        : num  0.6 0.7 0.9 0.8 0.6 0.7 0.8 0.8 0.7 0.1 ...
##  $ mobile_wt    : int  188 136 145 131 141 164 139 187 174 93 ...
##  $ n_cores      : int  2 3 5 6 2 1 8 4 7 5 ...
##  $ pc           : int  2 6 6 9 14 7 10 0 14 15 ...
##  $ px_height    : int  20 905 1263 1216 1208 1004 381 512 386 1137 ...
##  $ px_width     : int  756 1988 1716 1786 1212 1654 1018 1149 836 1224 ...
##  $ ram          : int  2549 2631 2603 2769 1411 1067 3220 700 1099 513 ...
##  $ sc_h         : int  9 17 11 16 8 17 13 16 17 19 ...
##  $ sc_w         : int  7 3 2 8 2 1 8 3 1 10 ...
##  $ talk_time    : int  19 7 9 11 15 10 18 5 20 12 ...
##  $ three_g      : int  0 1 1 1 1 1 1 1 1 1 ...
##  $ touch_screen : int  0 1 1 0 1 0 0 1 0 0 ...
##  $ wifi         : int  1 0 0 0 0 0 1 1 0 0 ...
##  $ price_range  : int  1 2 2 2 1 1 3 0 0 0 ...
```

(a)

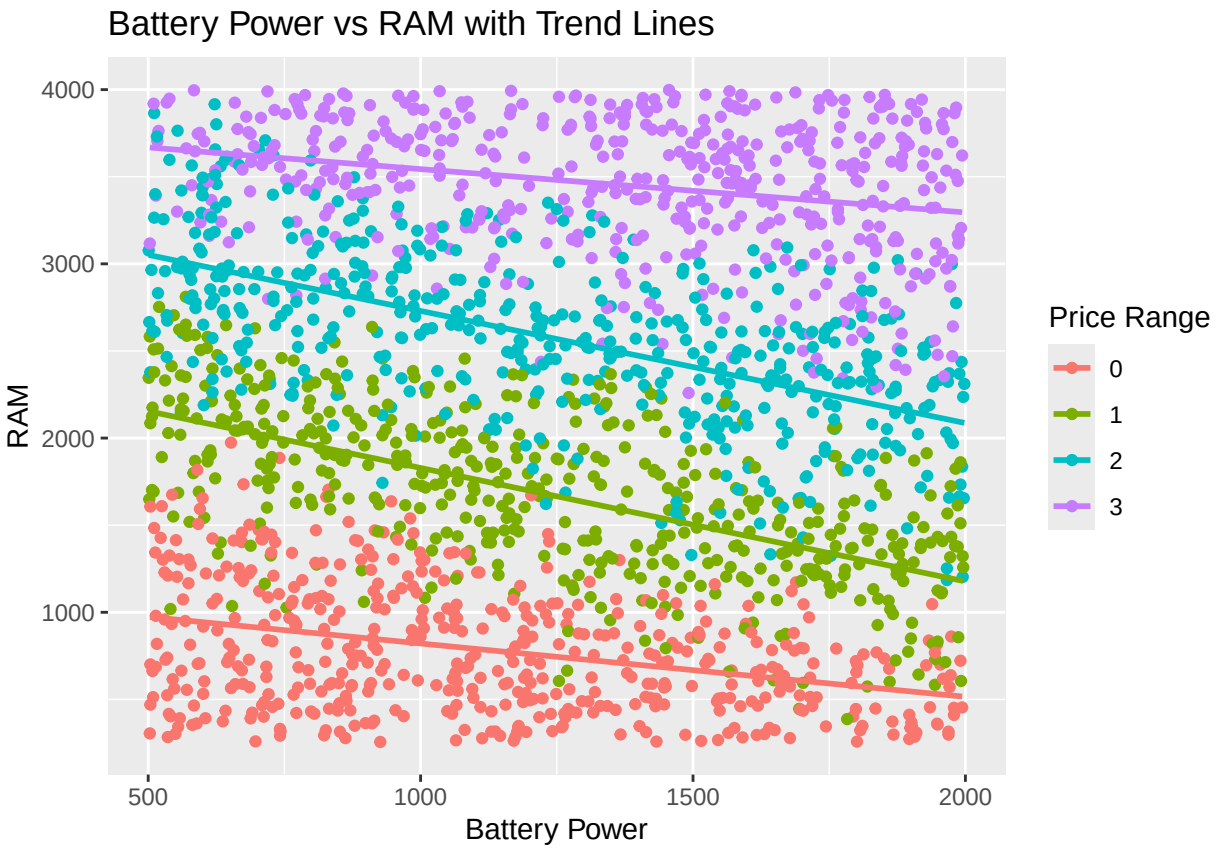
```
ggplot(mobile_data, aes(x = battery_power, y = ram, color = factor(price_range))) +
  geom_point() +
  labs(title = "Scatter Plot: Battery Power vs RAM", x = "Battery Power", y = "RAM", color = "Price Rang
```



(b)

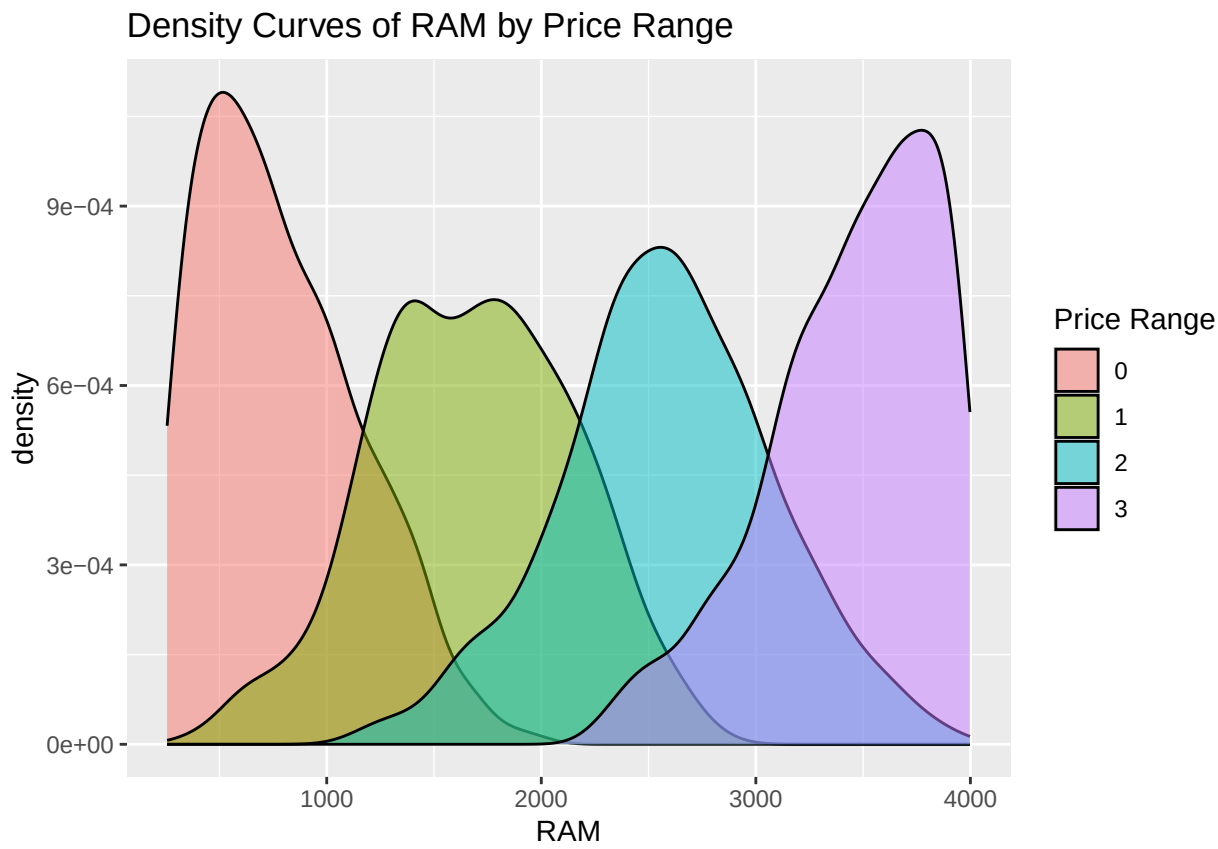
```
ggplot(mobile_data, aes(x = battery_power, y = ram, color = factor(price_range))) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  labs(title = "Battery Power vs RAM with Trend Lines", x = "Battery Power", y = "RAM", color = "Price
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



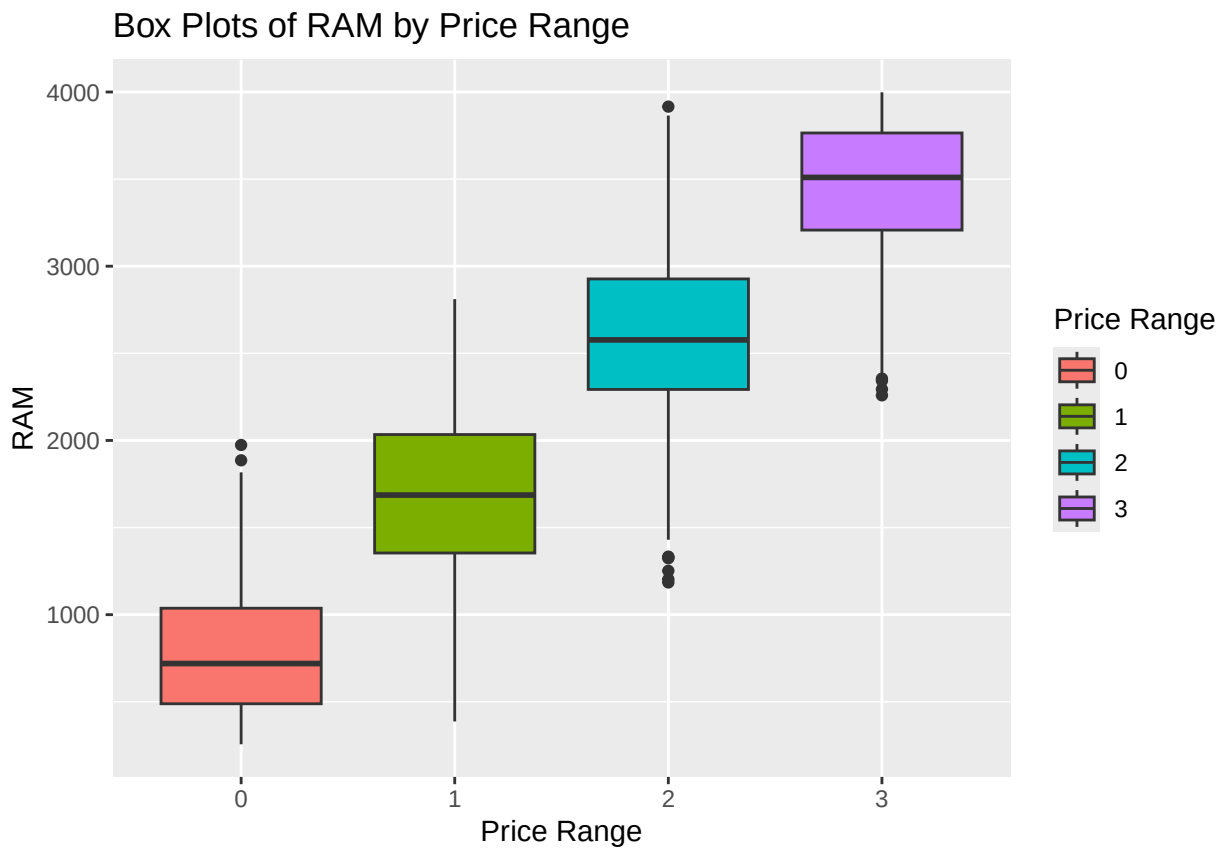
(c)

```
ggplot(mobile_data, aes(x = ram, fill = factor(price_range))) +
  geom_density(alpha = 0.5) +
  labs(title = "Density Curves of RAM by Price Range", x = "RAM", fill = "Price Range")
```



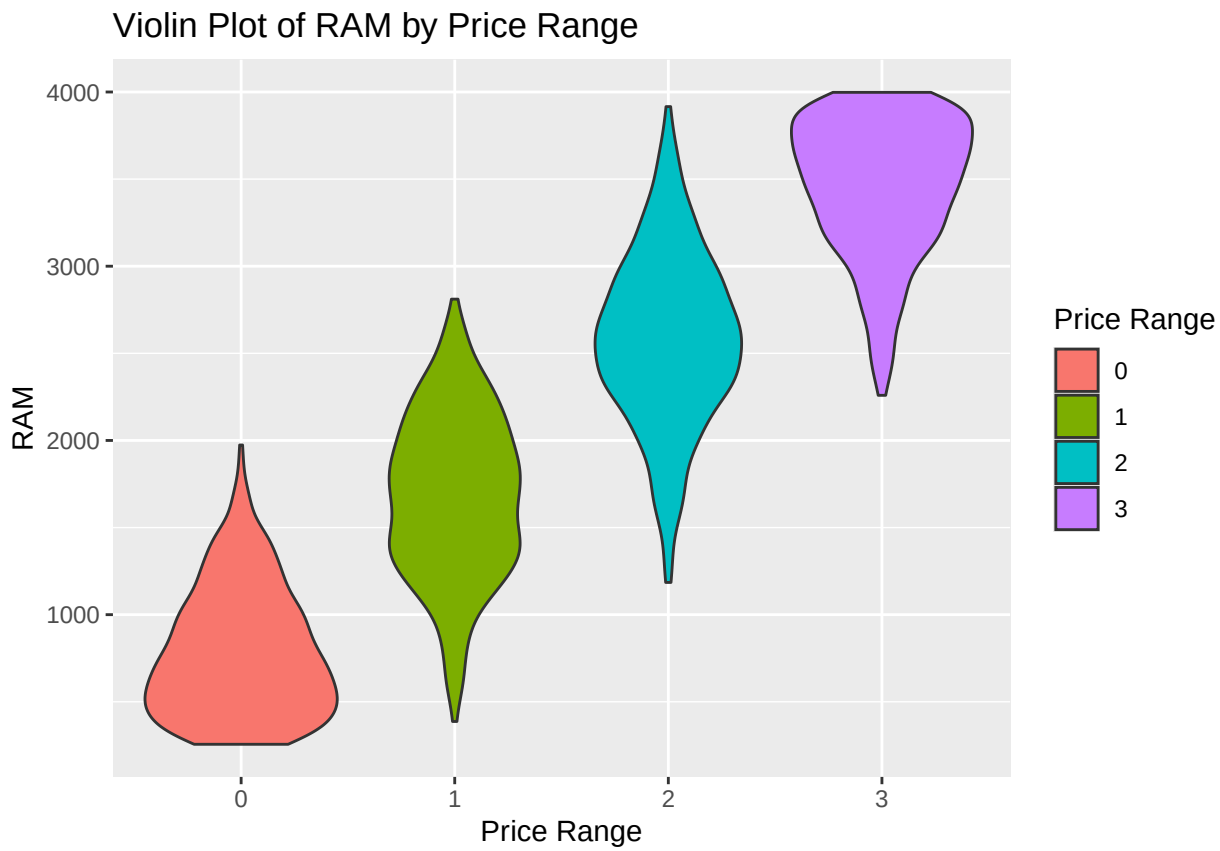
(d)

```
ggplot(mobile_data, aes(x = factor(price_range), y = ram, fill = factor(price_range))) +
  geom_boxplot() +
  labs(title = "Box Plots of RAM by Price Range", x = "Price Range", y = "RAM", fill = "Price Range")
```



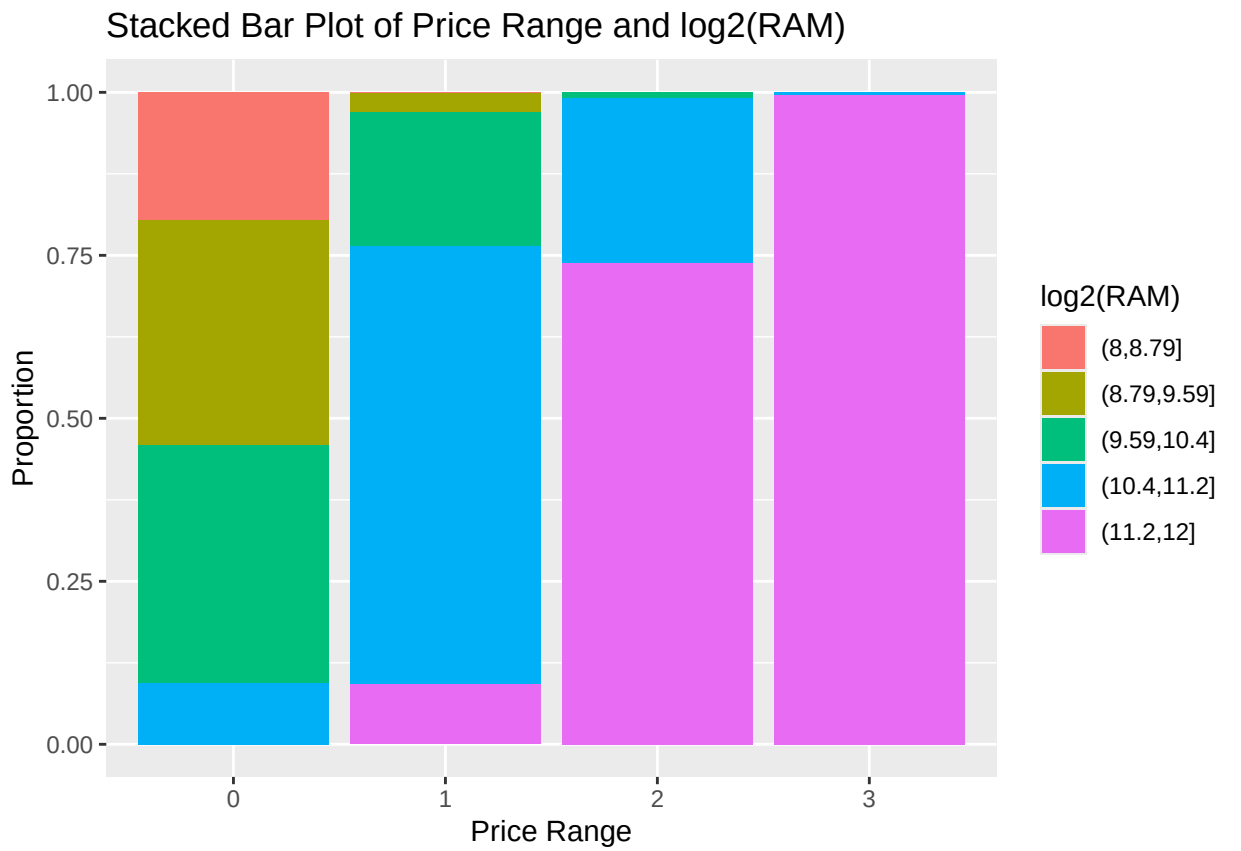
(e)

```
ggplot(mobile_data, aes(x = factor(price_range), y = ram, fill = factor(price_range))) +
  geom_violin() +
  labs(title = "Violin Plot of RAM by Price Range", x = "Price Range", y = "RAM", fill = "Price Range")
```



(f)

```
mobile_data$log2_ram <- log2(mobile_data$ram)
# stacked bar plot
ggplot(mobile_data, aes(x = factor(price_range), fill = factor(cut(log2_ram, breaks = 5)))) +
  geom_bar(position = "fill") +
  labs(title = "Stacked Bar Plot of Price Range and log2(RAM)", x = "Price Range", y = "Proportion", fi
```



Problem 2 **(a)**

```
# CRAN mirror
options(repos = c(CRAN = "https://cran.rstudio.com/"))

install.packages("UsingR")
```

```
##
## The downloaded binary packages are in
##  /var/folders/21/75_sfw8d6mb3b6s8mftqhsm00000gn/T//RtmpgkAe7D/downloaded_packages
```

```
library(UsingR)
```

```
## Loading required package: MASS

## Loading required package: HistData

## Loading required package: Hmisc

##
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```r
data("UScereal")
str(UScereal)
```

```
## 'data.frame':    65 obs. of  11 variables:
##  $ mfr      : Factor w/ 6 levels "G","K","N","P",..: 3 2 2 1 2 1 6 4 5 1 ...
##  $ calories : num  212 212 100 147 110 ...
##  $ protein  : num  12.12 12.12 8 2.67 2 ...
##  $ fat      : num  3.03 3.03 0 2.67 0 ...
##  $ sodium   : num  394 788 280 240 125 ...
##  $ fibre    : num  30.3 27.3 28 2 1 ...
##  $ carbo    : num  15.2 21.2 16 14 11 ...
##  $ sugars   : num  18.2 15.2 0 13.3 14 ...
##  $ shelf    : int  3 3 3 1 2 3 1 3 2 1 ...
##  $ potassium: num  848.5 969.7 660 93.3 30 ...
##  $ vitamins : Factor w/ 3 levels "100%","enriched",..: 2 2 2 2 2 2 2 2 2 2 ...
```

```r
levels(UScereal$mfr) <- c("General Mills", "Kelloggs", "Nabisco", "Post", "Quaker Oats", "Ralston Purina
levels(UScereal$mfr)
```

```
## [1] "General Mills"  "Kelloggs"       "Nabisco"        "Post"
## [5] "Quaker Oats"    "Ralston Purina"
```

(b)

```r
# Convert shelf to factor
UScereal$shelf <- factor(UScereal$shelf, levels = c(1, 2, 3), labels = c("Lower", "Middle", "Upper"))
str(UScereal$shelf)
```

```
##  Factor w/ 3 levels "Lower","Middle",..: 3 3 3 1 2 3 1 3 2 1 ...
```

(c)

```r
UScereal$Product <- rownames(UScereal)
str(UScereal)
```

```
## 'data.frame':    65 obs. of  12 variables:
##  $ mfr      : Factor w/ 6 levels "General Mills",..: 3 2 2 1 2 1 6 4 5 1 ...
##  $ calories : num  212 212 100 147 110 ...
##  $ protein  : num  12.12 12.12 8 2.67 2 ...
##  $ fat      : num  3.03 3.03 0 2.67 0 ...
##  $ sodium   : num  394 788 280 240 125 ...
##  $ fibre    : num  30.3 27.3 28 2 1 ...
##  $ carbo    : num  15.2 21.2 16 14 11 ...
##  $ sugars   : num  18.2 15.2 0 13.3 14 ...
##  $ shelf    : Factor w/ 3 levels "Lower","Middle",..: 3 3 3 1 2 3 1 3 2 1 ...
##  $ potassium: num  848.5 969.7 660 93.3 30 ...
##  $ vitamins : Factor w/ 3 levels "100%","enriched",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ Product  : chr  "100% Bran" "All-Bran" "All-Bran with Extra Fiber" "Apple Cinnamon Cheerios" ...
```

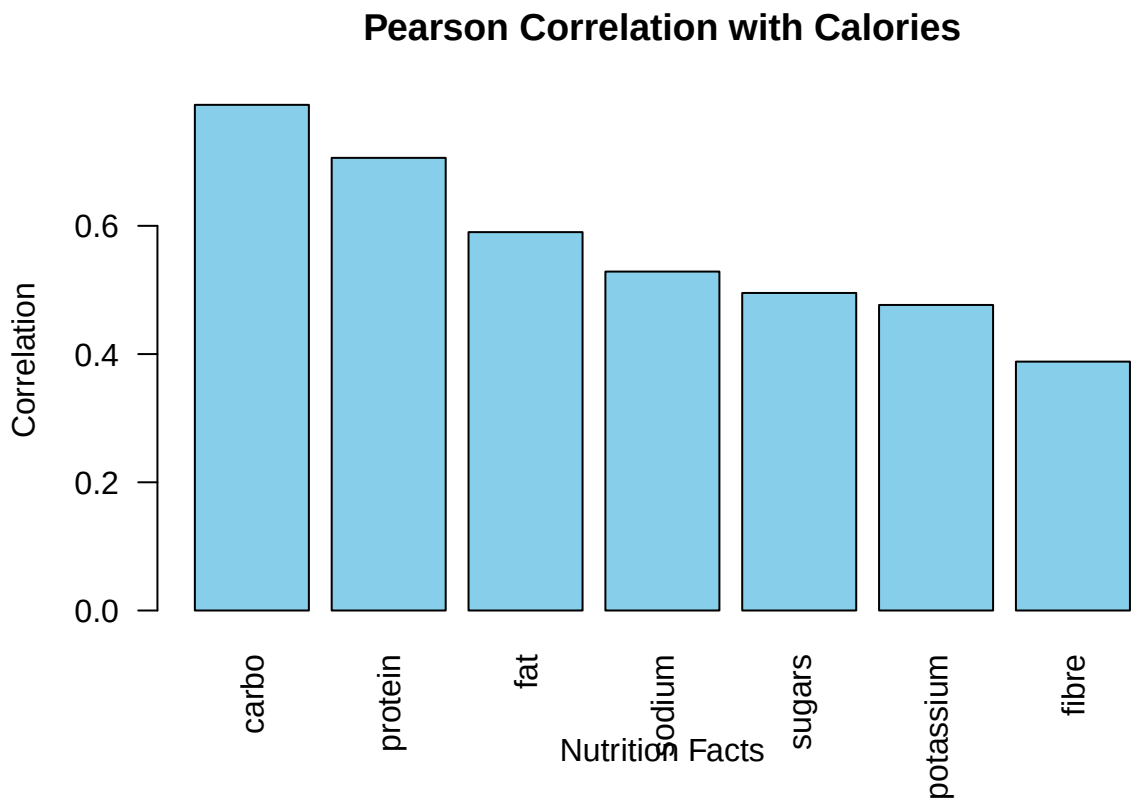
(d)

```r
nutrition_vars <- c("protein", "fat", "sodium", "fibre", "carbo", "sugars", "potassium")
correlations <- sapply(nutrition_vars, function(x) cor(UScereal$calories, UScereal[[x]], use = "complete
correlations
```

```
##   protein       fat    sodium     fibre     carbo    sugars potassium
## 0.7060105 0.5901757 0.5286552 0.3882179 0.7887227 0.4952942 0.4765955
```

(e)

```r
ordered_correlations <- sort(correlations, decreasing = TRUE)
# Bar plot
barplot(ordered_correlations, main = "Pearson Correlation with Calories",
        ylab = "Correlation", xlab = "Nutrition Facts", col = "skyblue", las = 2)
```

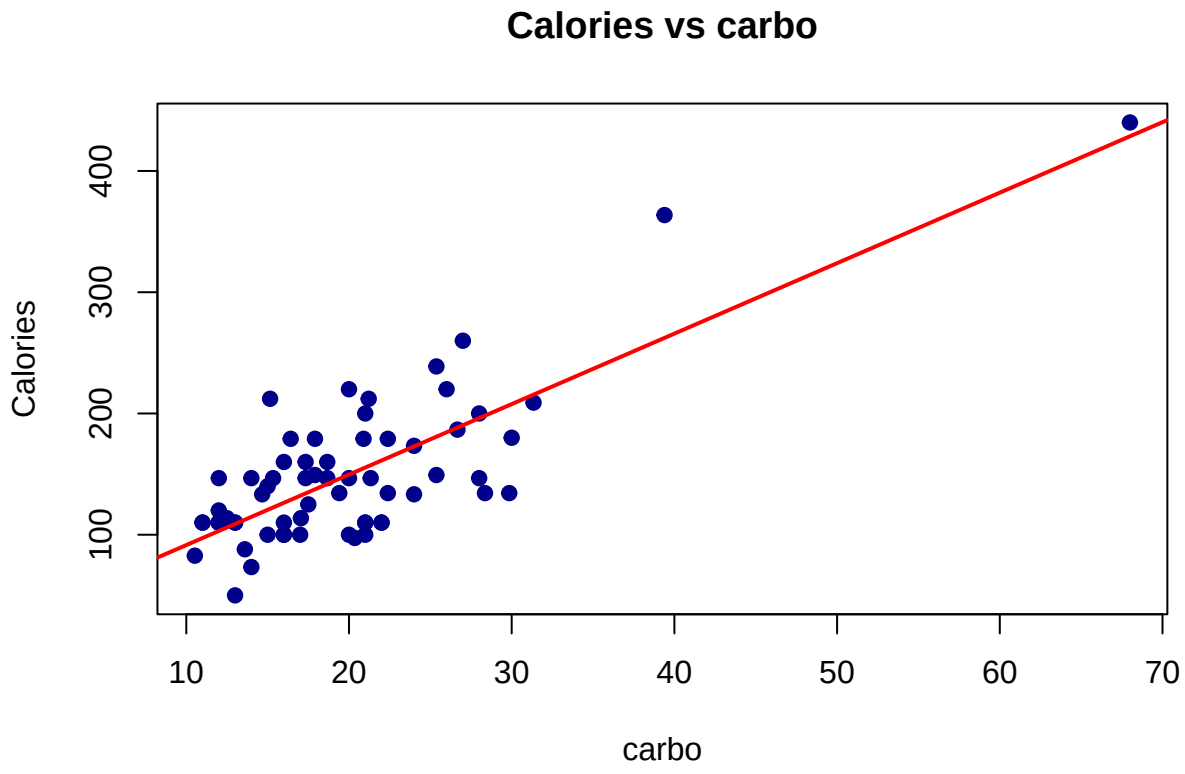



```r
highest_corr_nutrition <- names(ordered_correlations)[1]
highest_corr_nutrition
```

```
## [1] "carbo"
```

(f)

```r
# Scatter plot
plot(UScereal[[highest_corr_nutrition]], UScereal$calories,
     main = paste("Calories vs", highest_corr_nutrition),
     xlab = highest_corr_nutrition, ylab = "Calories", pch = 19, col = "darkblue")
# trend line
abline(lm(UScereal$calories ~ UScereal[[highest_corr_nutrition]]), col = "red", lwd = 2)
```




```r
lm_model <- lm(UScereal$calories ~ UScereal[[highest_corr_nutrition]])
summary(lm_model)
```

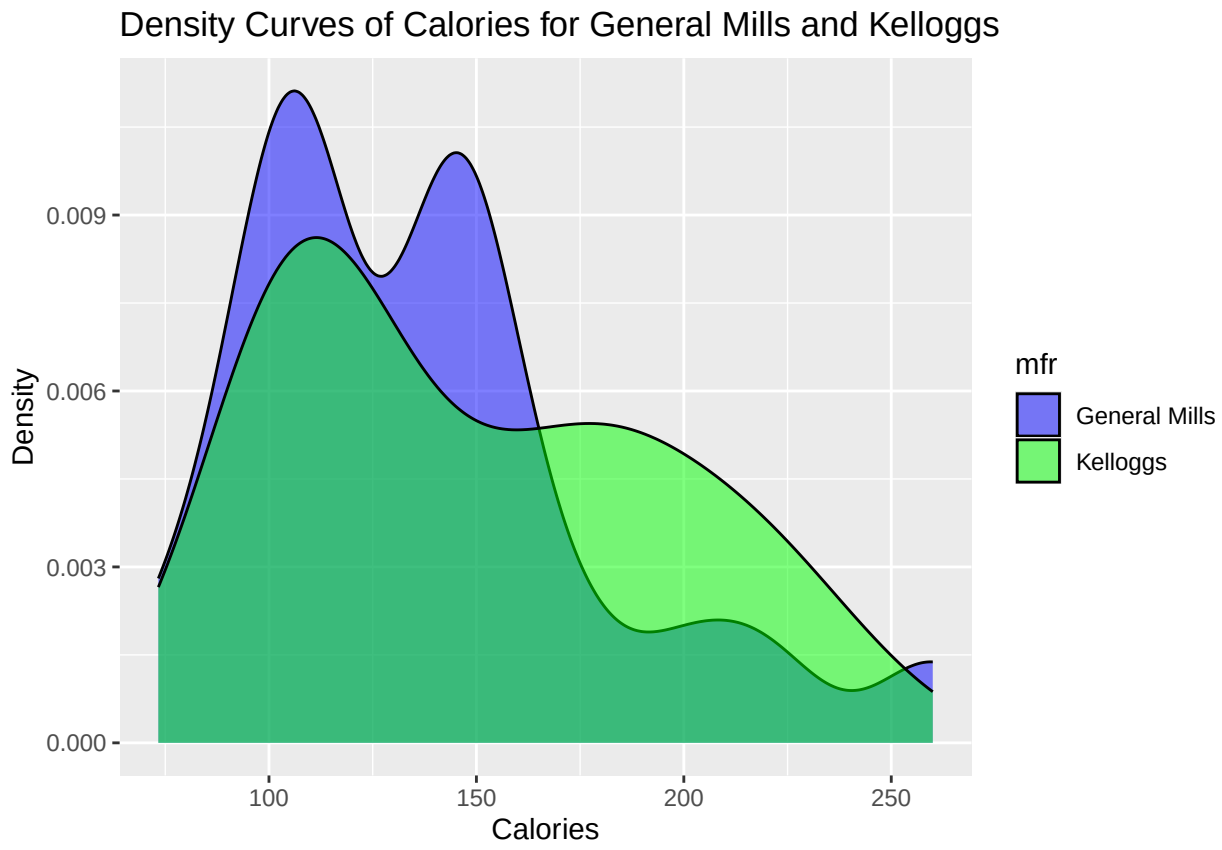
```
##
## Call:
## lm(formula = UScereal$calories ~ UScereal[[highest_corr_nutrition]])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -72.529 -27.725   1.093  19.468 101.306
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       33.3401    12.3658   2.696  0.00899 **
## UScereal[[highest_corr_nutrition]]   5.8128     0.5708  10.183 6.13e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 38.67 on 63 degrees of freedom
## Multiple R-squared:  0.6221, Adjusted R-squared:  0.6161
## F-statistic: 103.7 on 1 and 63 DF,  p-value: 6.132e-15
```

(g)

```r
# Subset the data for General Mills and Kelloggs
gm_kelloggs <- UScereal[UScereal$mfr %in% c("General Mills", "Kelloggs"), ]

# density plots
library(ggplot2)
ggplot(gm_kelloggs, aes(x = calories, fill = mfr)) +
  geom_density(alpha = 0.5) +
  labs(title = "Density Curves of Calories for General Mills and Kelloggs", x = "Calories", y = "Density
  scale_fill_manual(values = c("General Mills" = "blue", "Kelloggs" = "green"))
```



(h)

```r
# Box plot
ggplot(gm_kelloggs, aes(x = mfr, y = calories, fill = mfr)) +
  geom_boxplot() +
  labs(title = "Comparison of Calories Between General Mills and Kelloggs", x = "Manufacturer", y = "Ca
  scale_fill_manual(values = c("General Mills" = "blue", "Kelloggs" = "green"))
```

(i)

```
install.packages("reshape2")
```

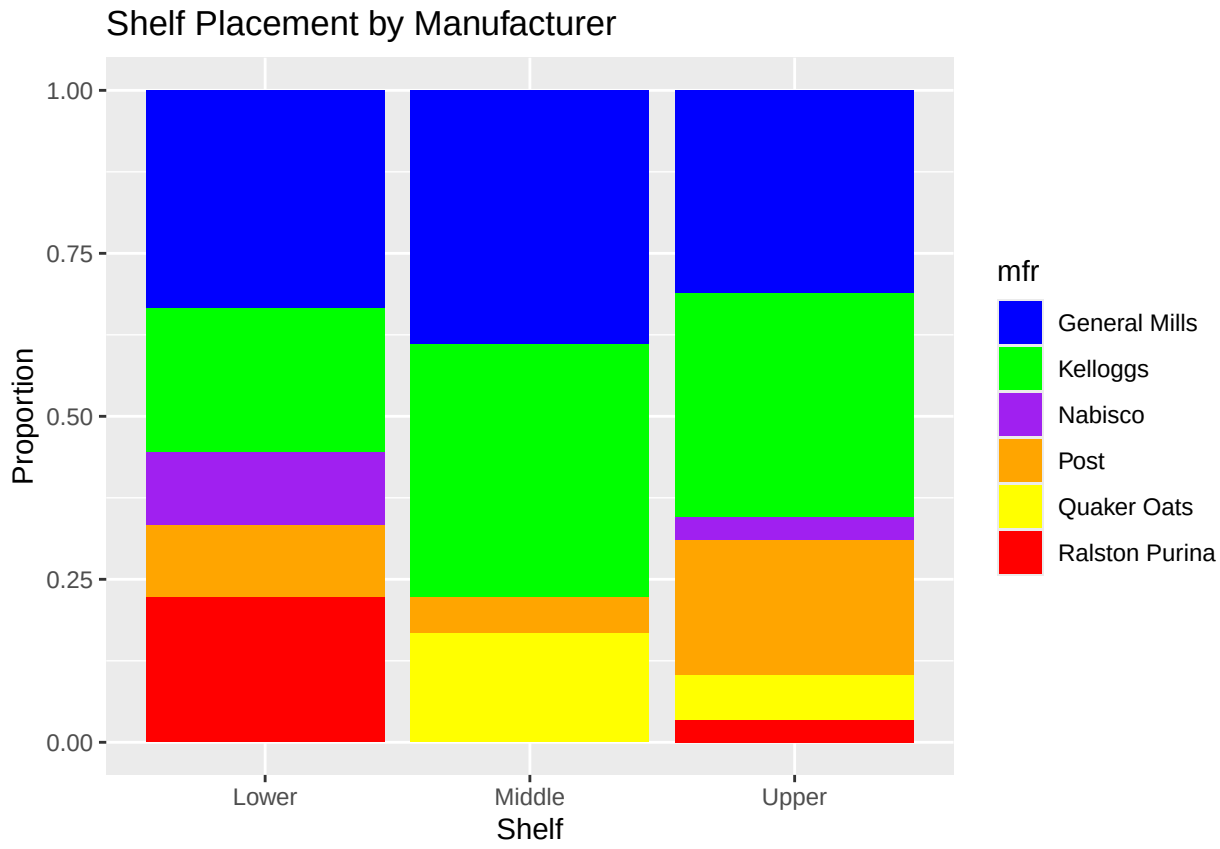
```
##
## The downloaded binary packages are in
##  /var/folders/21/75_sfw8d6mb3b6s8mftqhsm00000gn/T//RtmpgkAe7D/downloaded_packages
```

```
library(reshape2)
melted_data <- melt(UScereal, id.vars = "mfr", measure.vars = nutrition_vars)

#side-by-side box plots
ggplot(melted_data, aes(x = mfr, y = value, fill = mfr)) +
  geom_boxplot() +
  facet_wrap(~ variable, scales = "free") +
  labs(title = "Comparison of Nutrition Facts Across Manufacturers", x = "Manufacturer", y = "Nutrition
  scale_fill_manual(values = c("General Mills" = "blue", "Kelloggs" = "green", "Nabisco" = "purple",
                               "Post" = "orange", "Quaker Oats" = "yellow", "Ralston Purina" = "red"))
```

(j)

```
# Stacked bar plot
ggplot(UScereal, aes(x = shelf, fill = mfr)) +
  geom_bar(position = "fill") +
  labs(title = "Shelf Placement by Manufacturer", x = "Shelf", y = "Proportion") +
  scale_fill_manual(values = c("General Mills" = "blue", "Kelloggs" = "green", "Nabisco" = "purple",
                               "Post" = "orange", "Quaker Oats" = "yellow", "Ralston Purina" = "red"))
```


Shelf Placement by Manufacturer
1.00
0.75
0.50
0.25
0.00
Proportion
Lower
Middle
Upper
Shelf
mfr
General Mills
Kelloggs
Nabisco
Post
Quaker Oats
Ralston Purina