453 HW3

Max Tjen

2023-06-24

Packages

```
library(GGally)
library(nnet)
library(car)
library(VGAM)
library(dplyr)
library(tidyverse)
```

Question 9

Part A

In the context of DAVFs, independence would be when the rows and columns aren't related, so the odds for each classification (column) is the same for each symptom (row). Dependence would be when the rows and columns are related, so the classification odds shift depending on which symptom a patient is experiencing. To justify their classification system, the researchers would prefer symptoms and classification to be dependent. This is because there would then be an association and a clearer way to classify a patient based on their symptom.

Part B

```
# create new dataframe of data
table <- array (data = c(0, 1, 0, 0, 0, 0, 83,
                         0, 8, 0, 1, 1, 0, 17,
                         2, 1, 0, 0, 0, 0, 7,
                         1, 2, 6, 2, 1, 0, 6,
                         10, 0, 8, 1, 0, 0, 6,
                         19, 4, 2, 3, 0, 0, 1,
                         5, 1, 0, 0, 0, 6, 0),
                dim = c(7, 7),
                dimnames = list(Symptom = c("Hemmorage", "Intracranial hypertension",
                                            "Focal neurologic deficit", "Seizures",
                                            "Cardiac deficiency", "Myelopathy",
                                            "Non-aggressive symptoms"),
                                Classification = c("1", "2a", "2b", "2a and 2b",
```

```
                                                "3", "4", "5")))

table
```

```
##                             Classification
## Symptom                       1 2a 2b 2a and 2b  3  4 5
##   Hemmorage                   0  0  2         1 10 19 5
##   Intracranial hypertension   1  8  1         2  0  4 1
##   Focal neurologic deficit    0  0  0         6  8  2 0
##   Seizures                    0  1  0         2  1  3 0
##   Cardiac deficiency          0  1  0         1  0  0 0
##   Myelopathy                  0  0  0         0  0  0 6
##   Non-aggressive symptoms    83 17  7         6  6  1 0
```

```
chisq.test(table)
```

```
## Warning in chisq.test(table): Chi-squared approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test
##
## data:  table
## X-squared = 303.2, df = 36, p-value < 2.2e-16
```

Part C

If the alternative scenario was true, the Pearson chi square results would be invalidated because of the test's assumptions. One of the assumptions wouldn't be met, which is that cells in the table are mutually exclusive. This means that an individual can't belong to more than one cell, which would occur in the case of the alternative scenario.

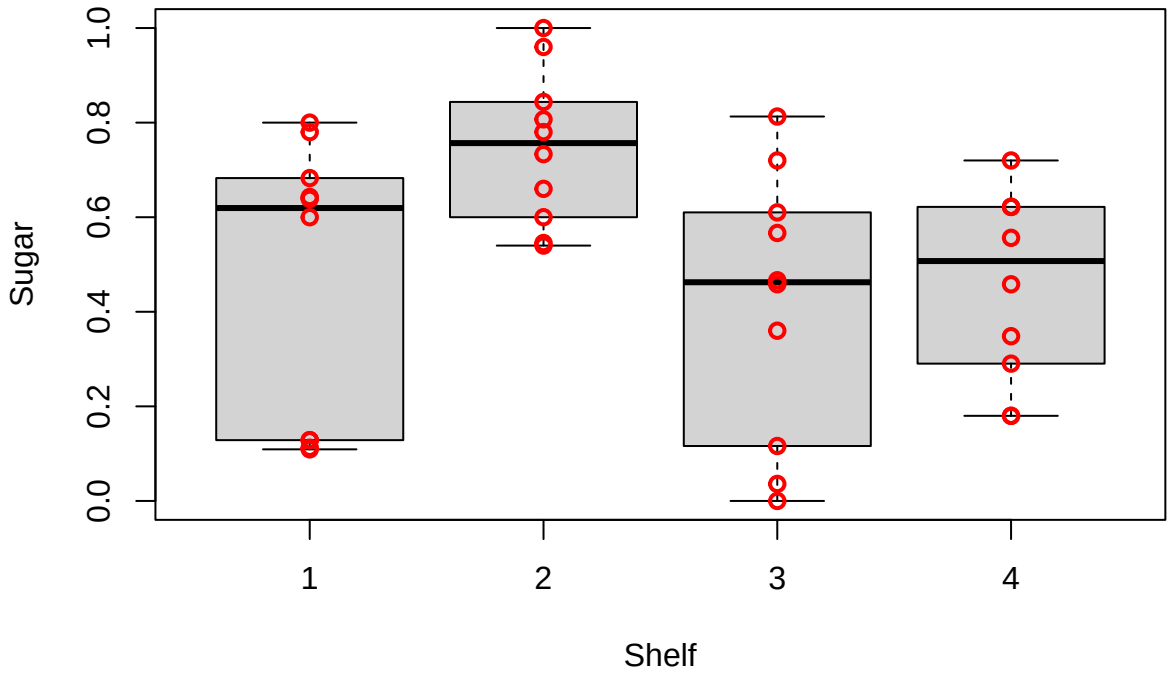
Question 12

Part A

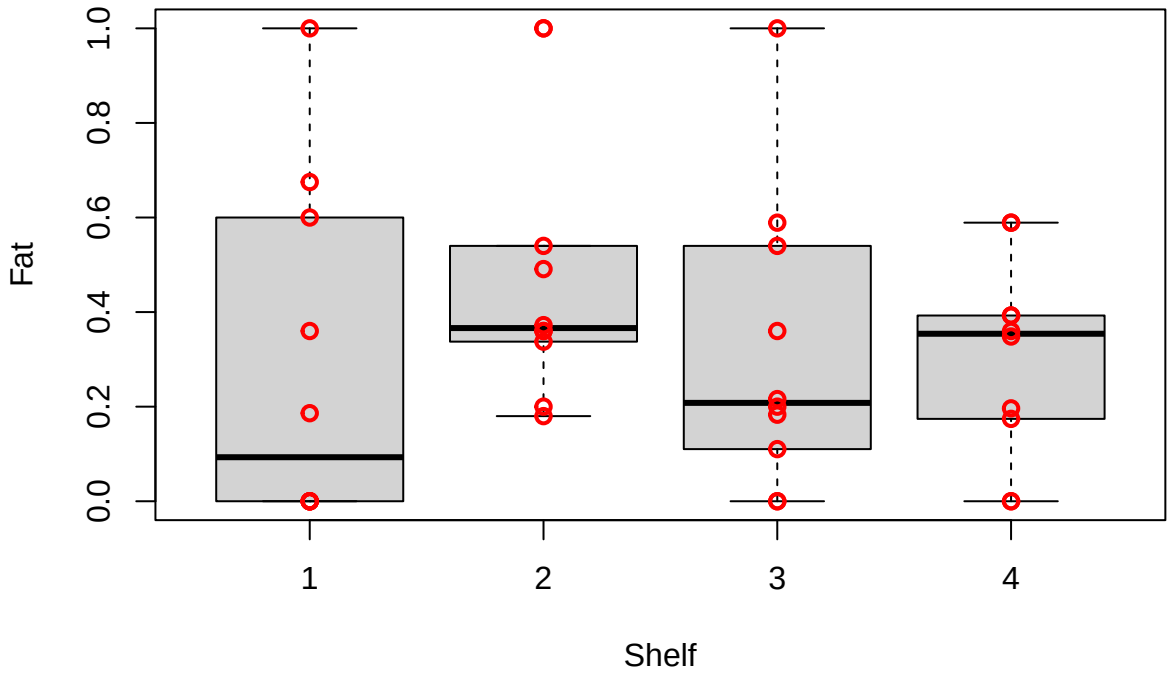
```
data12 <- read_csv("/Users/mtjen/Desktop/453/hw3/cereal_dillons.csv")

stand01 <- function(x) { (x - min(x))/(max(x) - min(x)) }
data12 <- data.frame(Shelf = data12$Shelf,
                     sugar = stand01(x = data12$sugar_g/data12$size_g),
                     fat = stand01(x = data12$fat_g/data12$size_g),
                     sodium = stand01(x = data12$sodium_mg/data12$size_g))

data12 <- data12 |> mutate(Shelf = factor(Shelf))
```

Part B

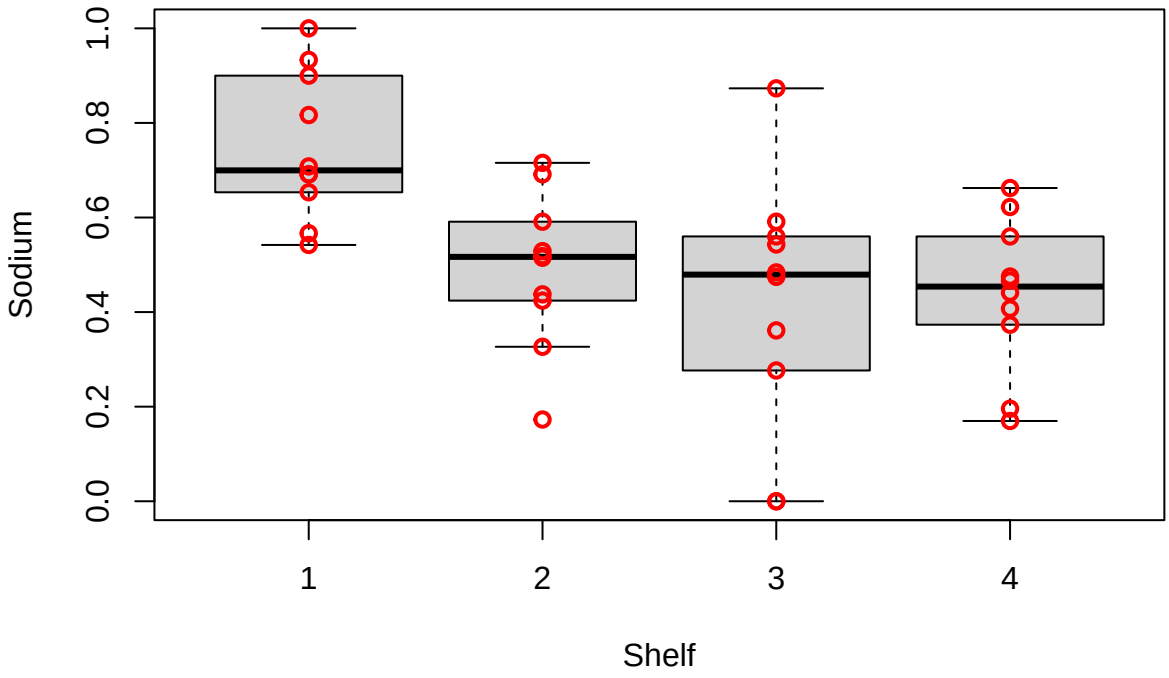
```r
# sugar
boxplot(formula = sugar ~ Shelf, data = data12, ylab = "Sugar",
        xlab = "Shelf", pars = list(outpch=NA))
stripchart(x = data12$sugar ~ data12$Shelf, lwd = 2, col = "red",
           vertical = TRUE, pch = 1, add = TRUE)
```



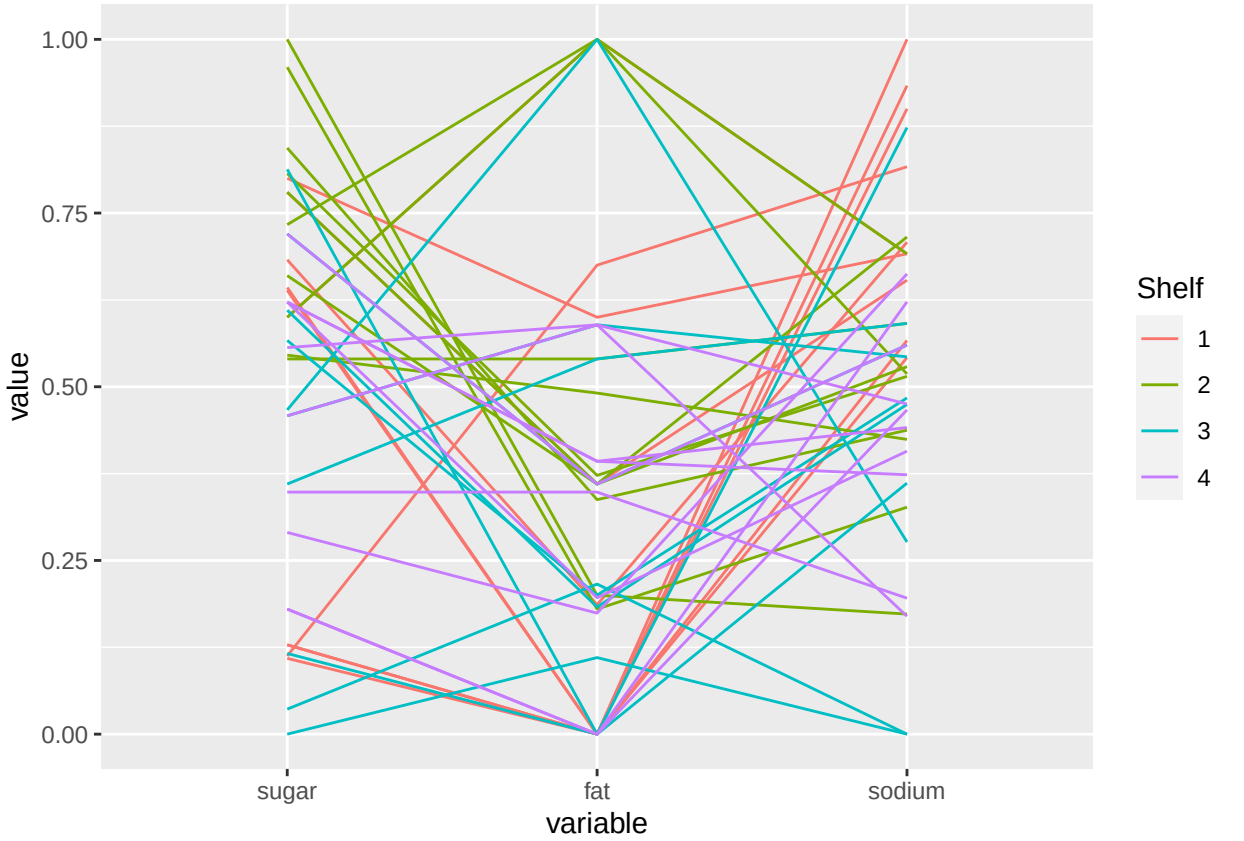

```r
# fat
boxplot(formula = fat ~ Shelf, data = data12, ylab = "Fat",
        xlab = "Shelf", pars = list(outpch=NA))
stripchart(x = data12$fat ~ data12$Shelf, lwd = 2, col = "red",
           vertical = TRUE, pch = 1, add = TRUE)
```




```
# sodium
boxplot(formula = sodium ~ Shelf, data = data12, ylab = "Sodium",
        xlab = "Shelf", pars = list(outpch=NA))
stripchart(x = data12$sodium ~ data12$Shelf, lwd = 2, col = "red",
           vertical = TRUE, pch = 1, add = TRUE)
```




```
# parallel coordinates plot
ggparcoord(data = data12, columns = c(2:4), groupColumn = "Shelf", scale = "uniminmax")
```



In the first plot, we can see that shelf 2 has higher sugar content than the other shelves which are pretty similar. For fat content, the shelves are pretty similar with the median for shelf 1 being noticeably lower than the others. The last plot shows us that shelf 1 has higher sodium content than the other shelves, which are pretty similar to each other.

Part C

It would be desirable to take ordinality into account when there is a clear order of consistency across the shelves. From the plots before, there isn't any clear order of the shelves in terms of healthiness or content, so ordinality shouldn't be taken into account.

Part D

```
mod12 <- multinom(Shelf ~ sugar + fat + sodium,
                  data = data12, trace = FALSE)

Anova(mod12)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Shelf
##        LR Chisq Df Pr(>Chisq)
## sugar   22.7648  3  4.521e-05 ***
```

```
## fat      5.2836  3     0.1522
## sodium  26.6197  3  7.073e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

By running a likelihood ratio test for each explanatory variable in our multinomial regression model, we can see that the sugar and sodium variables are statistically significant. They both have very small p-values while the p-value of fat isn't very small at 0.152. This leads us to believe that sugar and sodium content are associated with a cereal's shelf placement while fat isn't.

Part E

```
mod12e <- multinom(Shelf ~ sugar + fat + sodium +
                     sugar * fat + sugar * sodium + fat * sodium +
                     sugar * fat * sodium,
                   data = data12, trace = FALSE)

Anova(mod12e)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Shelf
##                   LR Chisq Df Pr(>Chisq)
## sugar              19.2525  3  0.0002424 ***
## fat                 6.1167  3  0.1060686
## sodium             30.8407  3  9.183e-07 ***
## sugar:fat           3.2309  3  0.3573733
## sugar:sodium        3.0185  3  0.3887844
## fat:sodium          3.1586  3  0.3678151
## sugar:fat:sodium    2.5884  3  0.4595299
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

By adding interaction terms between each pair of explanatory variables as well as with all three, we can see that the likelihood ratio test doesn't return a statistically significant p-value for any of the new terms.

Part F

```
apple <- read_csv("/Users/mtjen/Desktop/453/hw3/cereal_dillons.csv")

# add row
apple[nrow(apple) + 1,] = list(41, 1, "Apple Jacks", 28, 12, 0.5, 130)

# retransform
apple <- data.frame(Shelf = apple$Shelf,
                    sugar = stand01(x = apple$sugar_g/apple$size_g),
                    fat = stand01(x = apple$fat_g/apple$size_g),
                    sodium = stand01(x = apple$sodium_mg/apple$size_g)) |>
  mutate(Shelf = factor(Shelf))
```

```r
tail(apple)
```

```
##    Shelf     sugar       fat    sodium
## 36     4 0.3483871 0.3483871 0.1956989
## 37     4 0.4581818 0.5890909 0.1696970
## 38     4 0.6218182 0.3927273 0.4412121
## 39     4 0.5563636 0.5890909 0.4751515
## 40     4 0.1800000 0.0000000 0.6222222
## 41     1 0.7714286 0.1928571 0.4333333
```

```r
# create and predict test data
testData <- tibble(sugar = apple[41,]$sugar,
                   fat = apple[41,]$fat,
                   sodium = apple[41,]$sodium)

predict(mod12, newdata = testData, type = "probs")
```

```
##          1          2          3          4
## 0.05326849 0.47194264 0.20042742 0.27436145
```

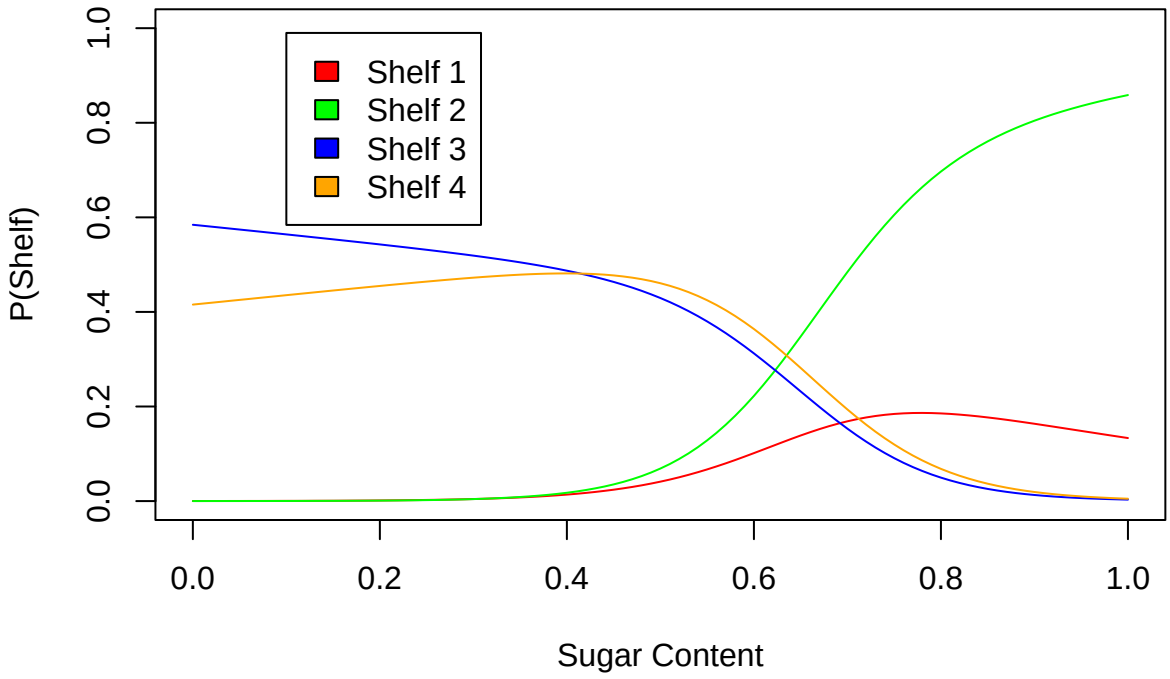
The predicted shelf probabilities for Apple Jacks given the information provided is:

- $\text{P}(\text{shelf} = 1) = 0.053$
- $\text{P}(\text{shelf} = 2) = 0.472$
- $\text{P}(\text{shelf} = 3) = 0.200$
- $\text{P}(\text{shelf} = 4) = 0.274$

Part G

```r
meanFat <- mean(apple$fat)
meanSodium <- mean(apple$sodium)

plotData <- data.frame(sugar = seq(from = 0, to = 1, by = 0.01),
                       fat = rep(meanFat, 101),
                       sodium = rep(meanSodium, 101))

predictedVals <- data.frame(predict(mod12, newdata = plotData, type = "probs"))
colnames(predictedVals) = c("P(Shelf 1)", "P(Shelf 2)", "P(Shelf 3)", "P(Shelf 4)")

comb <- cbind(plotData, predictedVals)

# plot probabilities
plot(comb$sugar, comb$`P(Shelf 1)`,
     type ="l",col="red",
     xlab = "Sugar Content", ylab = "P(Shelf)",
     ylim = c(0, 1))
lines(comb$sugar, comb$`P(Shelf 2)`, col="green")
lines(comb$sugar, comb$`P(Shelf 3)`, col="blue")
lines(comb$sugar, comb$`P(Shelf 4)`, col="orange")
```

```
# add legend to plot
legend(0.1, 0.99,
       legend = c("Shelf 1", "Shelf 2", "Shelf 3", "Shelf 4"),
       fill = c("red","green", "blue", "orange"))
```



With constant fat and sodium contents, we can see the impact of changing sugar content levels. If a cereal's sugar content is below 0.6, it has a relatively similar probability of being on shelf 3 or 4. A cereal with sugar content higher than 0.6 has the highest probability of being on shelf 2, particularly as the content level increases. Shelf 1 has a relatively low probability of being where the cereal is across all sugar content levels.

Part H

```
# odds ratios, confidence intervals
#round(exp(0.15*2.693071), 2)
#round(exp(0.15*(2.693071-12.216442)), 2)
```

???????

Question 16

Part A

```
data16 <- read_csv("/Users/mtjen/Desktop/453/hw3/ice_cream.csv")

xtabs(count ~ fat + rating,
      data = data16)
```

```
##       rating
## fat     1  2  3  4  5  6  7  8  9
##   0     4 17  8 16  5  6  4  2  1
##   0.04  1  1  5  6  7  9 21 12  0
##   0.08  0  2  2  2  4 13 16 21  3
##   0.12  1  1  1  3  4 11 15 23  4
##   0.16  0  3  2  6  3  7 17 17  5
##   0.2   0  1  3  8  4 13 14 11  8
##   0.24  1  5  4 14  2 13 13  7  2
##   0.28  4  6  9 11  5  9  7  8  3
```

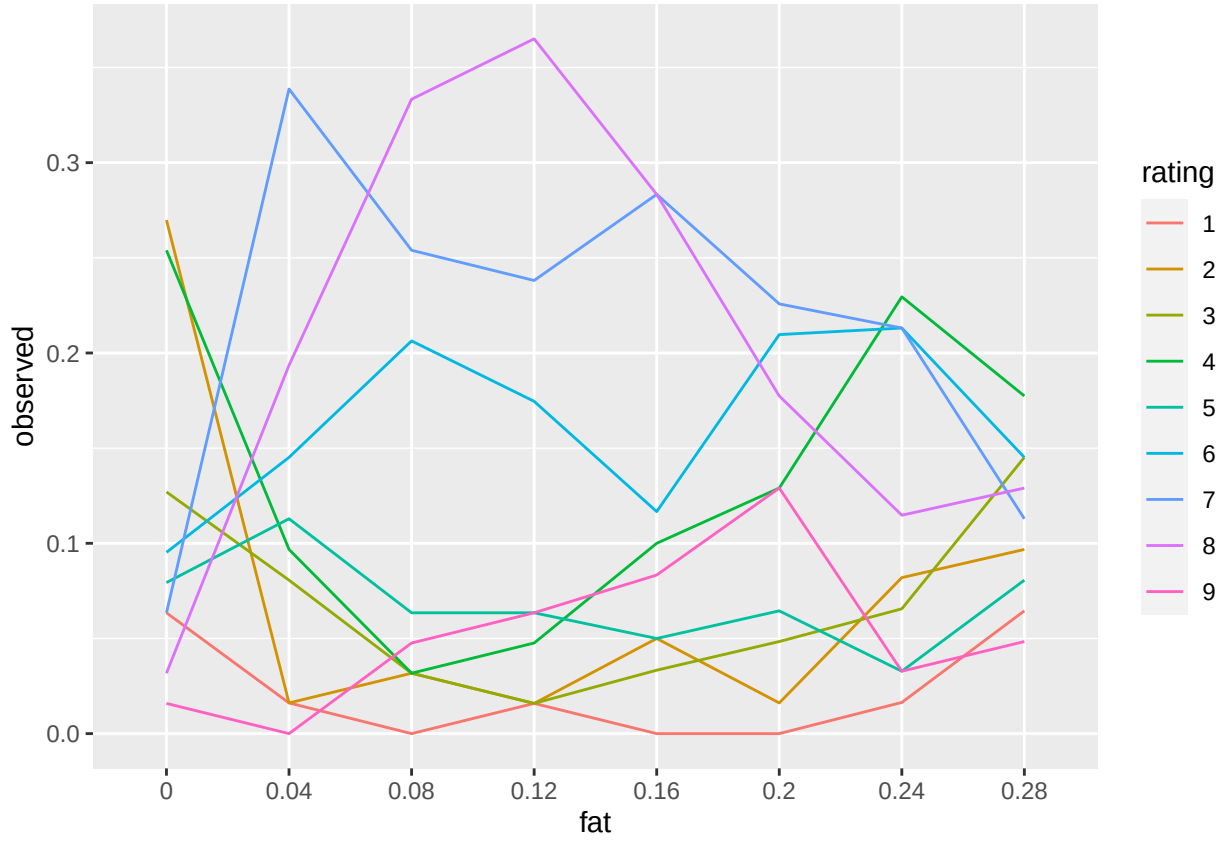
Part B

```
# contingency table to dataframe
freq16 <- data.frame(xtabs(count ~ fat + rating,
                           data = data16))

# group by fat and create new variable
freq16 <- freq16 |>
  group_by(fat) |>
  mutate(observed = Freq / sum(Freq))

xtabs(observed ~ fat + rating,
      data = freq16)
```

```
##       rating
## fat             1          2          3          4          5          6
##   0    0.06349206 0.26984127 0.12698413 0.25396825 0.07936508 0.09523810
##   0.04 0.01612903 0.01612903 0.08064516 0.09677419 0.11290323 0.14516129
##   0.08 0.00000000 0.03174603 0.03174603 0.03174603 0.06349206 0.20634921
##   0.12 0.01587302 0.01587302 0.01587302 0.04761905 0.06349206 0.17460317
##   0.16 0.00000000 0.05000000 0.03333333 0.10000000 0.05000000 0.11666667
##   0.2  0.00000000 0.01612903 0.04838710 0.12903226 0.06451613 0.20967742
##   0.24 0.01639344 0.08196721 0.06557377 0.22950820 0.03278689 0.21311475
##   0.28 0.06451613 0.09677419 0.14516129 0.17741935 0.08064516 0.14516129
##       rating
## fat             7          8          9
##   0    0.06349206 0.03174603 0.01587302
##   0.04 0.33870968 0.19354839 0.00000000
##   0.08 0.25396825 0.33333333 0.04761905
##   0.12 0.23809524 0.36507937 0.06349206
```

```
##   0.16 0.28333333 0.28333333 0.08333333
##   0.2  0.22580645 0.17741935 0.12903226
##   0.24 0.21311475 0.11475410 0.03278689
##   0.28 0.11290323 0.12903226 0.04838710
```

```
ggplot(data = freq16, aes(x = fat, y = observed,
                          group = rating, color = rating)) +
  geom_line()
```



Part C

Pearson χ^2 and likelihood ratio tests wouldn't be the ideal forms of analysis for independence on this data because they are focused on whether or not the values for each variable are independent from the other. In this instance for both fat content and rating, there is an order as to which values are higher than others, which isn't accounted for by the aforementioned tests.

Part D

```
data16 <- data16 |>
  mutate(rating = factor(rating))

mod16 <- MASS::polr(rating ~ fat + I(fat^2),
```

```
                  data = data16, method = "logistic", weights = count)

Anova(mod16)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: rating
##          LR Chisq Df Pr(>Chisq)
## fat        97.717  1  < 2.2e-16 ***
## I(fat^2)   99.513  1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The Anova test returns a very small and statistically significant p-value for the quadratic term, indicating that it would be helpful in prediction.

Part E

```
data16 <- data16 |>
  mutate(rating = factor(rating,
                         order = TRUE,
                         levels = c("1", "2", "3",
                                    "4", "5", "6",
                                    "7", "8", "9")))

# with proportional odds assumption
mod16ePO <- vglm(rating ~ fat + I(fat^2),
                 family = cumulative(parallel = TRUE),
                 data = data16[data16$count != 0,],
                 weights = count)

# with proportional odds assumption
mod16eNPO <- vglm(rating ~ fat + I(fat^2),
                  family = cumulative(parallel = FALSE),
                  data = data16[data16$count != 0,],
                  weights = count)

lr <- deviance(mod16ePO) - deviance(mod16eNPO)
dfs <- mod16ePO@df.residual - mod16eNPO@df.residual
1 - pchisq(q = lr, df = dfs)
```

```
## [1] 0.4998997
```

The p-value from the likelihood ratio test is statistically insignificant at 0.500, indicating that there isn't enough evidence that the proportional odds assumption is violated.

Part F

```r
c.value <- c(sd(data16$fat), 1)
round(1/exp(c.value*(-mod16$coefficients)), 2)
```

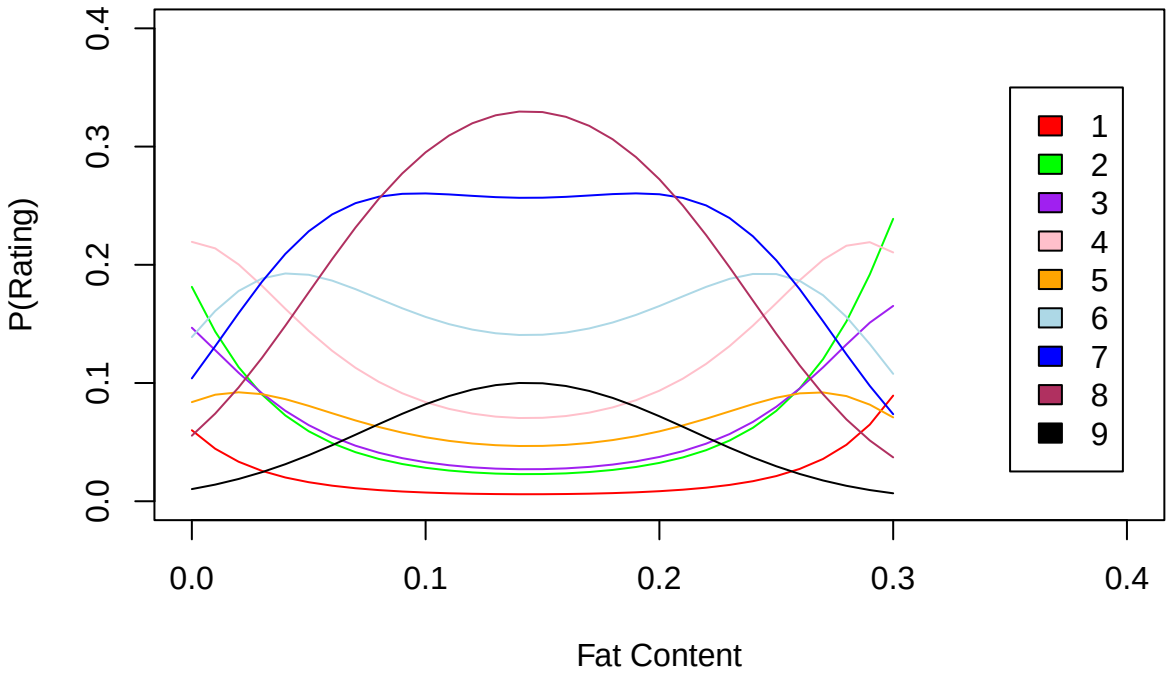
```
##      fat I(fat^2)
##    21.03     0.00
```

```r
c.value <- c(1, sd(data16$fat))
round(1/exp(c.value*(-mod16$coefficients)), 2)
```

```
##          fat     I(fat^2)
## 2.152115e+14 0.000000e+00
```

With a one standard deviation change in fat content, it appears that there is a very high likelihood that the ice cream rating will increase.

```r
# create and predict test data
testData <- tibble(fat = seq(from = 0, to = 0.3, by = 0.01))

predictions <- as.data.frame(predict(mod16, newdata = testData, type = "probs"))
colnames(predictions) <- c("1", "2", "3",
                           "4", "5", "6",
                           "7", "8", "9")

predictions$fat <- seq(from = 0, to = 0.3, by = 0.01)

# plot probabilities
plot(predictions$fat, predictions$`1`,
     type ="l",col="red",
     xlab = "Fat Content", ylab = "P(Rating)",
     xlim = c(0, 0.4),ylim = c(0, 0.4))
lines(predictions$fat, predictions$`2`, col="green")
lines(predictions$fat, predictions$`3`, col="purple")
lines(predictions$fat, predictions$`4`, col="pink")
lines(predictions$fat, predictions$`5`, col="orange")
lines(predictions$fat, predictions$`6`, col="lightblue")
lines(predictions$fat, predictions$`7`, col="blue")
lines(predictions$fat, predictions$`8`, col="maroon")
lines(predictions$fat, predictions$`9`, col="black")

# add legend to plot
legend(0.35, 0.35,
       legend = c("1", "2", "3",
                  "4", "5", "6",
                  "7", "8", "9"),
       fill = c("red","green", "purple",
                "pink","orange", "lightblue",
                "blue","maroon", "black"))
```



From the rating probabilities, it looks like the higher rating's ice cream probabilities are highest between 0.1 and 0.2. Conversely, lower ratings have higher probabilities when fat content is either low or high. As such, we would recommend an ice cream fat content in the middle, particularly between of 0.1 and 0.2.

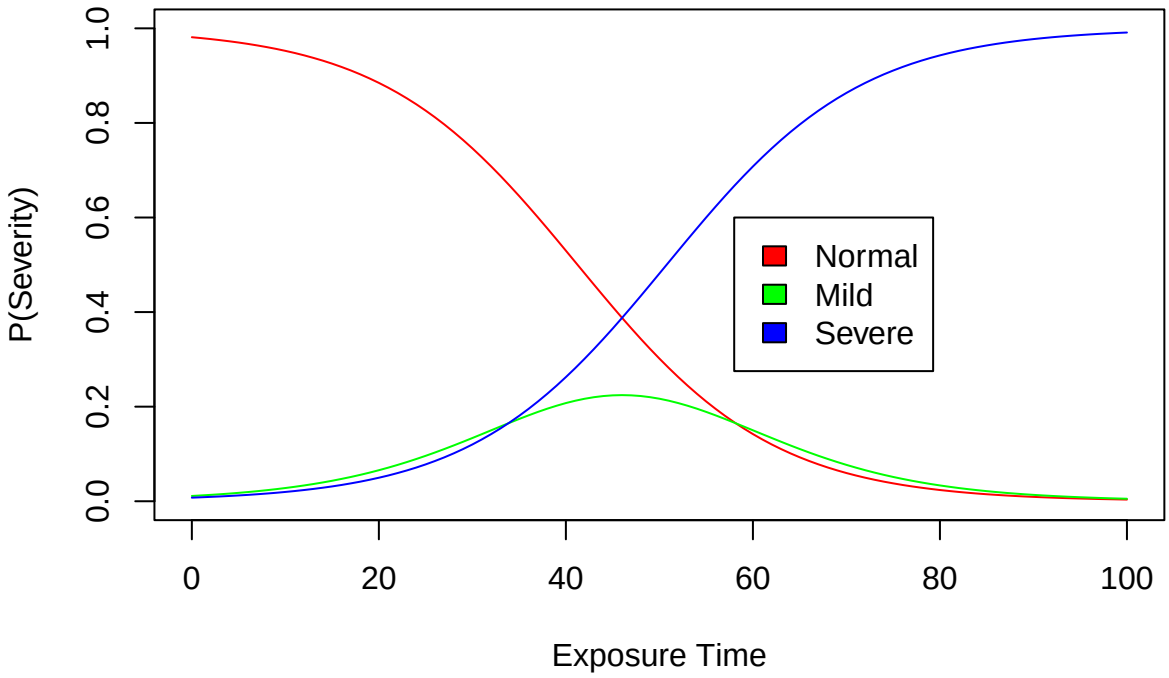
Question 21

```
data(pneumo)

normalCounts <- data.frame(exposure = pneumo$exposure.time,
                           count = pneumo$normal, severity = "normal")

mildCounts <- data.frame(exposure = pneumo$exposure.time,
                           count = pneumo$mild, severity = "mild")

severeCounts <- data.frame(exposure = pneumo$exposure.time,
                           count = pneumo$severe, severity = "severe")

data21 <- rbind(normalCounts, mildCounts, severeCounts) |>
  mutate(exposure = as.numeric(exposure),
         count = as.numeric(count),
         severity = factor(severity, order = TRUE,
                           levels = c("normal", "mild", "severe")))

mod21 <- MASS::polr(severity ~ exposure,
```

```
                    data = data21, weights = count)

Anova(mod21)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: severity
##          LR Chisq Df Pr(>Chisq)
## exposure   88.243  1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# create and predict test data
testData <- tibble(exposure = seq(from = 0, to = 100, by = 1))

predictions <- as.data.frame(predict(mod21, newdata = testData, type = "probs"))
colnames(predictions) <- c("Normal", "Mild", "Severe")
predictions$time <- seq(from = 0, to = 100, by = 1)

# plot probabilities
plot(predictions$time, predictions$Normal,
     type ="l",col="red",
     xlab = "Exposure Time", ylab = "P(Severity)",
     ylim = c(0, 1))
lines(predictions$time, predictions$Mild, col="green")
lines(predictions$time, predictions$Severe, col="blue")

# add legend to plot
legend(58, 0.6,
       legend = c("Normal", "Mild", "Severe"),
       fill = c("red","green", "blue"))
```



From the Anova test and the plot of probabilities, we can see that severity is quite dependent on exposure time. Normal and severe severity probabilities are almost inverse of each other, with normal being the most probable outcome from 0-47 and then severe being the most probable from 48-100. Mild has a relatively low probability across exposure times.

```
predictions |> filter(time %in% c(5, 10, 15, 20, 25))
```

```
##       Normal       Mild     Severe time
## 6  0.9700065 0.01773892 0.01225455    5
## 11 0.9524295 0.02792389 0.01964663   10
## 16 0.9253445 0.04329931 0.03135614   15
## 21 0.8847052 0.06560393 0.04969085   20
## 26 0.8261010 0.09601474 0.07788430   25
```

Here are the predicted severity probabilities for exposure times of 5, 10, 15, 20, and 25 years.