Statistical Methods for Discrete Response, Time Series, and Panel Data (W271): Lab 2

Heather Feinstein, David Harding, Charlotte Swavola

Strategic Placement of Products in Grocery Stores

Answer **Question 12 of chapter 3 (on page 189 and 190)** of Bilder and Loughin's *"Analysis of Categorical Data with R"*. Here is the background of this analysis, taken as an excerpt from this question:

In order to maximize sales, items within grocery stores are strategically placed to draw customer attention. This exercise examines one type of item:breakfast cereal. Typically, in large grocery stores, boxes of cereal are placed on sets of shelves located on one side of the aisle. By placing particular boxes of cereals on specific shelves, grocery stores may better attract customers to them. To investigate this further, a random sample of size 10 was taken from each of four shelves at a Dillons grocery store in Manhattan, KS. These data are given in the **cereal_dillons.csv** file. The response variable is the shelf number, which is numbered from bottom (1) to top (4), and the explanatory variables are the sugar, fat, and sodium content of the cereals.

```r
# Load libraries
library(Hmisc)
library(MASS)
library(nnet)
library(stargazer)

# Load dataset
cereal <- read.csv("cereal_dillons.csv")
head(cereal)
```

```
##   ID Shelf                                  Cereal size_g sugar_g fat_g
## 1  1     1 Kellog's Razzle Dazzle Rice Crispies     28      10     0
## 2  2     1            Post Toasties Corn Flakes     28       2     0
## 3  3     1                Kellogg's Corn Flakes     28       2     0
## 4  4     1              Food Club Toasted Oats     32       2     2
## 5  5     1                     Frosted Cheerios     30      13     1
## 6  6     1             Food Club Frosted Flakes     31      11     0
##   sodium_mg
## 1       170
## 2       270
## 3       300
## 4       280
## 5       210
## 6       180
```

```r
# describe(cereal)
```

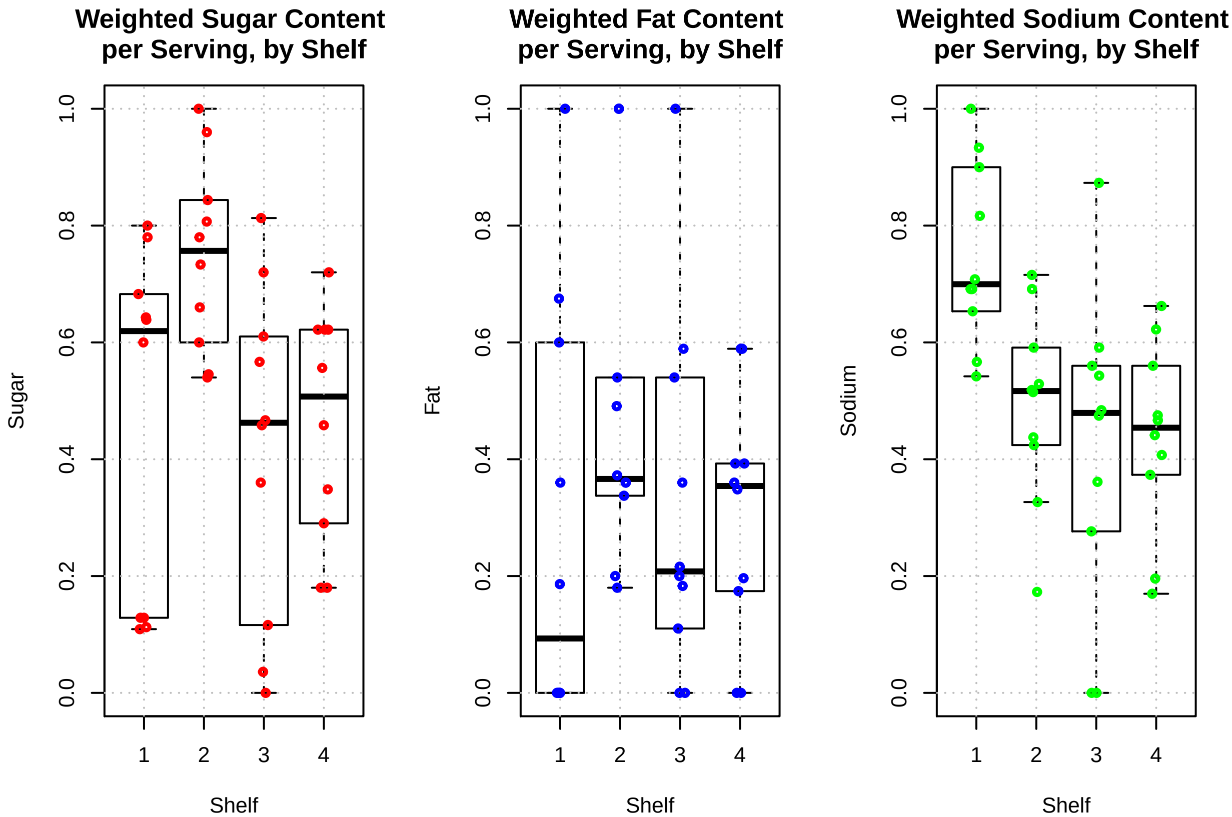
a. The explanatory variables need to be reformatted before proceeding further.
    - First, divide each explanatory variable by its serving size to account for the different serving sizes among the cereals.
    - Second, rescale each variable to be within 0 and 1.

```
stand01 <- function(x) {(x - min(x)) /( max(x) - min(x))}
cereal2 <- data.frame(Shelf = cereal$Shelf, sugar =
                      stand01(x = cereal$sugar_g/cereal$size_g),
                      fat = stand01(x = cereal$fat_g/cereal$size_g),
                      sodium = stand01 (x = cereal$sodium_mg/cereal$size_g))

summary(cereal2)
```

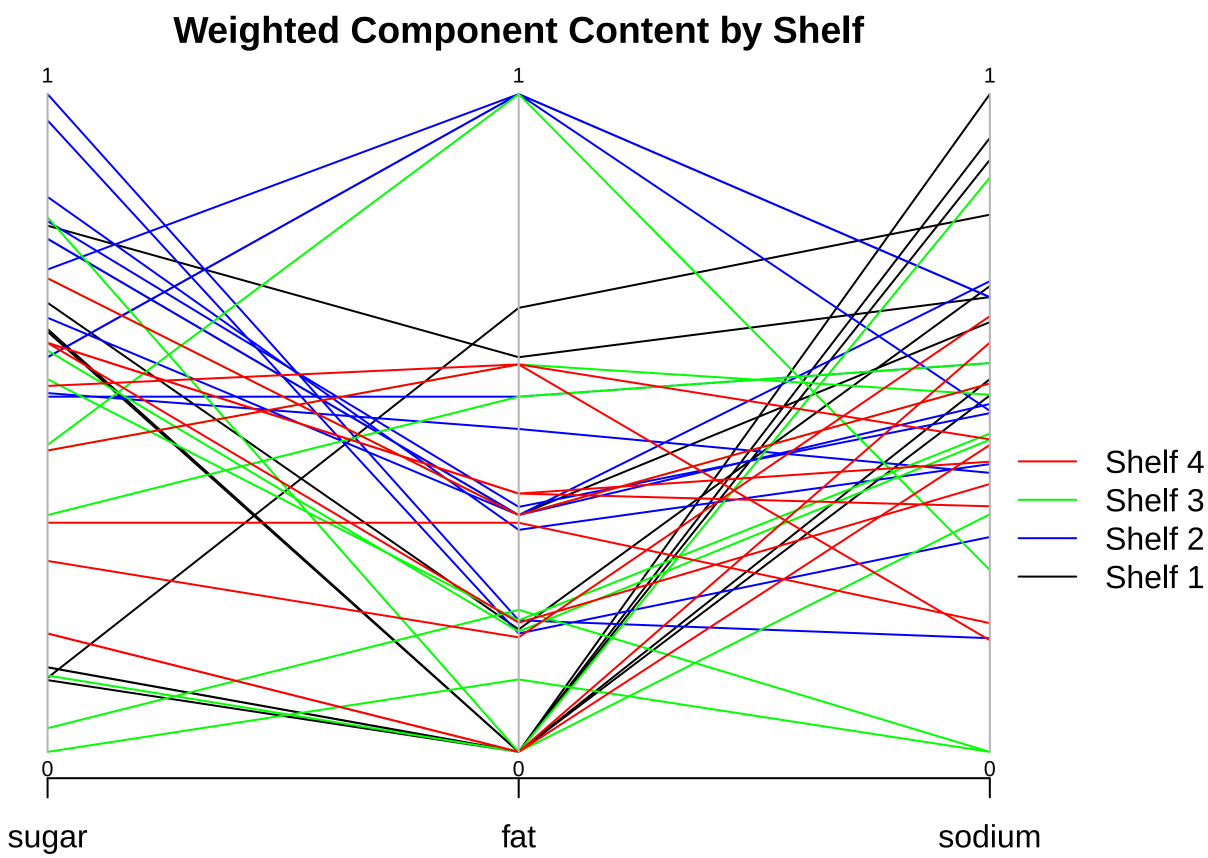
```
##      Shelf          sugar             fat              sodium
##  Min.   :1.00   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:1.75   1st Qu.:0.3339   1st Qu.:0.1582   1st Qu.:0.4200
##  Median :2.50   Median :0.6000   Median :0.3542   Median :0.5354
##  Mean   :2.50   Mean   :0.5209   Mean   :0.3476   Mean   :0.5240
##  3rd Qu.:3.25   3rd Qu.:0.7200   3rd Qu.:0.5400   3rd Qu.:0.6696
##  Max.   :4.00   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
```

b. Construct side-by-side box plots with dot plots overlaid for each of the explanatory variables.

```
par(mfrow=c(1,3))
boxplot(formula = sugar ~ Shelf, data = cereal2, ylab = "Sugar", xlab = "Shelf",
        pars = list(outpch = NA),main="Weighted Sugar Content \nper Serving, by Shelf")
stripchart(x = cereal2$sugar ~ cereal2$Shelf, lwd = 2, col = "red", method = "jitter",
           vertical = TRUE, pch = 1, add = TRUE,
           panel.first = grid(col = "gray", lty = "dotted"))

boxplot(formula = fat ~ Shelf, data = cereal2, ylab = "Fat", xlab = "Shelf",
        pars = list(outpch = NA),main="Weighted Fat Content \nper Serving, by Shelf")
stripchart(x = cereal2$fat ~ cereal2$Shelf, lwd = 2, col = "blue", method = "jitter",
           vertical = TRUE, pch = 1, add = TRUE,
           panel.first = grid(col = "gray", lty = "dotted"))

boxplot(formula = sodium ~ Shelf, data = cereal2, ylab = "Sodium", xlab = "Shelf",
        pars = list(outpch = NA),main="Weighted Sodium Content \nper Serving, by Shelf")
stripchart(x = cereal2$sodium ~ cereal2$Shelf, lwd = 2, col = "green", method = "jitter",
           vertical = TRUE, pch = 1, add = TRUE,
           panel.first = grid(col = "gray", lty = "dotted"))
```




```r
par(mar=c(2, 1, 2, 5),xpd = TRUE)
shelf.colors<-ifelse(test = cereal2$Shelf=="1", yes = "black",
                     no = ifelse(test = cereal2$Shelf=="2", yes = "blue", no = ifelse(test = ce
parcoord(x = cereal2[, c(2,3,4)], col = shelf.colors, var.label = TRUE)
title(main="Weighted Component Content by Shelf")
legend(3,0.5, legend = c("Shelf 4", "Shelf 3", "Shelf 2", "Shelf 1"), lty = "solid",
       col=c("red", "green", "blue", "black"), bty = 'n')
```



b. (cont'd) Discuss if possible content differences exist among the shelves.

- Shelf 1 has a higher, narrow sodium distribution relative to the wide distributions of other components and the sodium distribution - for other shelves.
- Shelf 2 has a higher, narrow distribution for sugar content, and hits the maximum fat content (though not for all samples).
- Shelf 3 maintains a wide distribution across all three components. It has the samples with the lowest sugar, fat, and sodium content.
- Shelf 4 has wide distributions for all components that are generally lower than the component distributions of other shelves.

c. The response has values of $1, 2, 3$, and 4. Under what setting would it be desirable to take into account ordinality. Do you think that this setting occurs here?

Ordinality could be considered for visibiility by a specific audience. For example, if you were examining kids' cereal specifically, the shelves would likely have the levels 2,1,3,4, for ordinality, as they are the most likely to be seen in that order by that specific audience. For fancy organic, perhaps 3,4,2,1- which corresponds to the eye levels of (most) adults. In this case, we haven't specified a cereal type or a particular audience to target, so an ordinality setting is not relevant.

d. Estimate a **multinomial regression model with linear forms of the sugar, fat, and sodium variables**. Perform **LRTs** to examine the importance of each explanatory variable.

```
#Estimate model
mod.fit<-multinom(formula = Shelf ~ sugar + fat + sodium, data=cereal2)
```

```
## # weights:  20 (12 variable)
## initial  value 55.451774
## iter  10 value 37.329384
## iter  20 value 33.775257
## iter  30 value 33.608495
## iter  40 value 33.596631
## iter  50 value 33.595909
## iter  60 value 33.595564
## iter  70 value 33.595277
## iter  80 value 33.595147
## final  value 33.595139
## converged
```

```
summary(mod.fit)
```

```
## Call:
## multinom(formula = Shelf ~ sugar + fat + sodium, data = cereal2)
##
## Coefficients:
##   (Intercept)      sugar        fat    sodium
## 2    6.900708   2.693071  4.0647092 -17.49373
## 3   21.680680 -12.216442 -0.5571273 -24.97850
## 4   21.288343 -11.393710 -0.8701180 -24.67385
##
## Std. Errors:
##   (Intercept)    sugar      fat   sodium
## 2    6.487408 5.051689 2.307250 7.097098
## 3    7.450885 4.887954 2.414963 8.080261
## 4    7.435125 4.871338 2.405710 8.062295
##
## Residual Deviance: 67.19028
## AIC: 91.19028
```

```
# LRT for sugar_g:
mod.fit.Ho_sugar<-multinom(formula = Shelf ~ fat + sodium, data=cereal2, trace=FALSE)
anova(mod.fit.Ho_sugar, mod.fit)
```

```
## Likelihood ratio tests of Multinomial Models
##
## Response: Shelf
##                  Model Resid. df Resid. Dev   Test    Df LR stat.
## 1         fat + sodium       111   89.95511
## 2 sugar + fat + sodium       108   67.19028 1 vs 2     3 22.76484
##        Pr(Chi)
## 1
## 2 4.520699e-05
```

The p-value for the likelihood ratio test is less than 0.05, therefore we can reject the null hypothesis for sugar, concluding that sugar is significant to the model.

```
# LRT for fat_g:
mod.fit.Ho_fat<-multinom(formula = Shelf ~ sugar + sodium, data=cereal2, trace=FALSE)
anova(mod.fit.Ho_fat, mod.fit)
```

```
## Likelihood ratio tests of Multinomial Models
##
## Response: Shelf
##                  Model Resid. df Resid. Dev   Test    Df LR stat.
## 1       sugar + sodium       111   72.47384
## 2 sugar + fat + sodium       108   67.19028 1 vs 2     3  5.28356
##     Pr(Chi)
## 1
## 2 0.1521727
```

We cannot reject the null hypothesis that fat is not significant to the model.

```
# LRT for sodium_mg:
mod.fit.Ho_sodium<-multinom(formula = Shelf ~ sugar + fat, data=cereal2, trace=FALSE)
anova(mod.fit.Ho_sodium, mod.fit)
```

```
## Likelihood ratio tests of Multinomial Models
##
## Response: Shelf
##                  Model Resid. df Resid. Dev   Test    Df LR stat.
## 1          sugar + fat       111   93.81001
## 2 sugar + fat + sodium       108   67.19028 1 vs 2     3 26.61974
##        Pr(Chi)
## 1
## 2 7.073281e-06
```

For sodium, the p-value is less than 0.05. We can reject the null hypothesis and conclude that sodium is significant to the model

e. Show that there are no significant interactions among the explanatory variables (including an interaction among all three variables).

```r
# LRT for sugar*sodium interaction term:
mod.fit.Ho_sugar_sodium <- multinom(formula = Shelf ~ sugar + fat + sodium + sugar:sodium, data
anova(mod.fit.Ho_sugar_sodium, mod.fit)
```

```
## Likelihood ratio tests of Multinomial Models
##
## Response: Shelf
##                                    Model Resid. df Resid. Dev   Test    Df
## 1                   sugar + fat + sodium       108   67.19028
## 2 sugar + fat + sodium + sugar:sodium       105   64.83988 1 vs 2     3
##   LR stat.  Pr(Chi)
## 1
## 2 2.350397 0.502935
```

```r
# LRT for sugar*fat interaction term:
mod.fit.Ho_sugar_fat <- multinom(formula = Shelf ~ sugar + fat + sodium + sugar:fat, data=cerea
anova(mod.fit.Ho_sugar_fat, mod.fit)
```

```
## Likelihood ratio tests of Multinomial Models
##
## Response: Shelf
##                                 Model Resid. df Resid. Dev   Test    Df
## 1                sugar + fat + sodium       108   67.19028
## 2 sugar + fat + sodium + sugar:fat       105   61.82907 1 vs 2     3
##   LR stat.   Pr(Chi)
## 1
## 2  5.36121 0.1471795
```

```r
# LRT for fat*sodium interaction term:
mod.fit.Ho_fat_sodium <- multinom(formula = Shelf ~ sugar + fat + sodium + sugar:sodium, data=c
anova(mod.fit.Ho_fat_sodium, mod.fit)
```

```
## Likelihood ratio tests of Multinomial Models
##
## Response: Shelf
##                                    Model Resid. df Resid. Dev   Test    Df
## 1                   sugar + fat + sodium       108   67.19028
## 2 sugar + fat + sodium + sugar:sodium       105   64.83988 1 vs 2     3
##   LR stat.  Pr(Chi)
## 1
## 2 2.350397 0.502935
```

```r
# LRT for sugar*fat*sodium interaction term:
mod.fit.Ho_sugar_fat_sodium <- multinom(formula = Shelf ~ sugar + fat + sodium + sugar:fat:sodi
anova(mod.fit.Ho_sugar_fat_sodium, mod.fit)
```

```
## Likelihood ratio tests of Multinomial Models
##
```

```
## Response: Shelf
##                                      Model Resid. df Resid. Dev   Test
## 1                     sugar + fat + sodium       108   67.19028
## 2 sugar + fat + sodium + sugar:fat:sodium       105   65.04570 1 vs 2
##      Df LR stat.   Pr(Chi)
## 1
## 2     3  2.14458 0.5429468
```

We cannot reject any of the null hypotheses for the possible interaction coefficients. These interaction coefficients are not significant to the model.

f. Kellogg's Apple Jacks (http://www.applejacks.com) is a cereal marketed toward children. For a serving size of 28 grams, its sugar content is 12 grams, fat content is 0.5 grams, and sodium content is 130 milligrams. Estimate the shelf probabilities for Apple Jacks.

```r
# Data for Apple Jacks standardized
stand01.spec <- function(w,x) {(w - min(x)) /( max(x) - min(x))}
newdata <- data.frame(sugar = stand01.spec(w = 12/28, x = cereal$sugar_g/cereal$size_g),
                      fat = stand01.spec(w = 0.5/28, x = cereal$fat_g/cereal$size_g),
                      sodium = stand01.spec(w = 130/28, x = cereal$sodium_mg/cereal$size_g))
newdata
```

```
##       sugar       fat    sodium
## 1 0.7714286 0.1928571 0.4333333
```

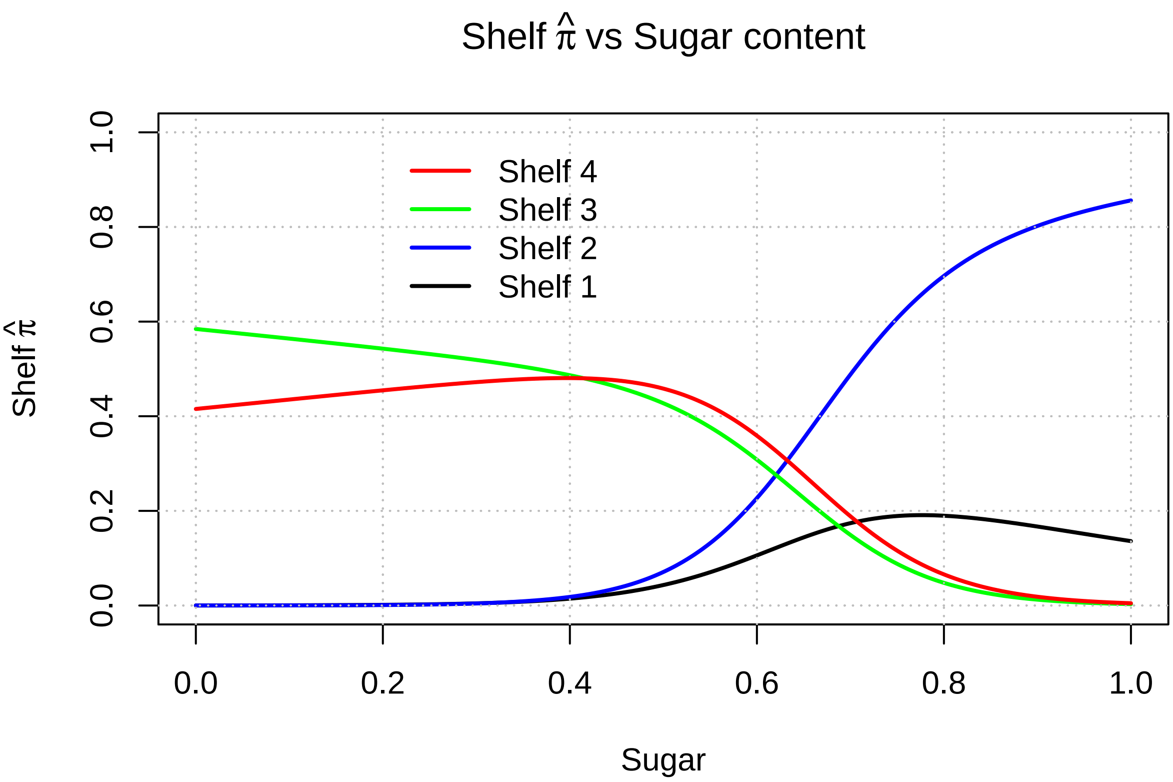
```r
# pi^
pi.hat<-predict(object = mod.fit, newdata = newdata, type = "probs")
round(pi.hat, 2)
```

```
##    1    2    3    4
## 0.05 0.47 0.20 0.27
```

g. Construct a plot similar to **Figure 3.3** where the estimated probability for a shelf is on the *y-axis* and the sugar content is on the *x-axis*. Use the mean overall fat and sodium content as the corresponding variable values in the model. Interpret the plot with respect to sugar content.

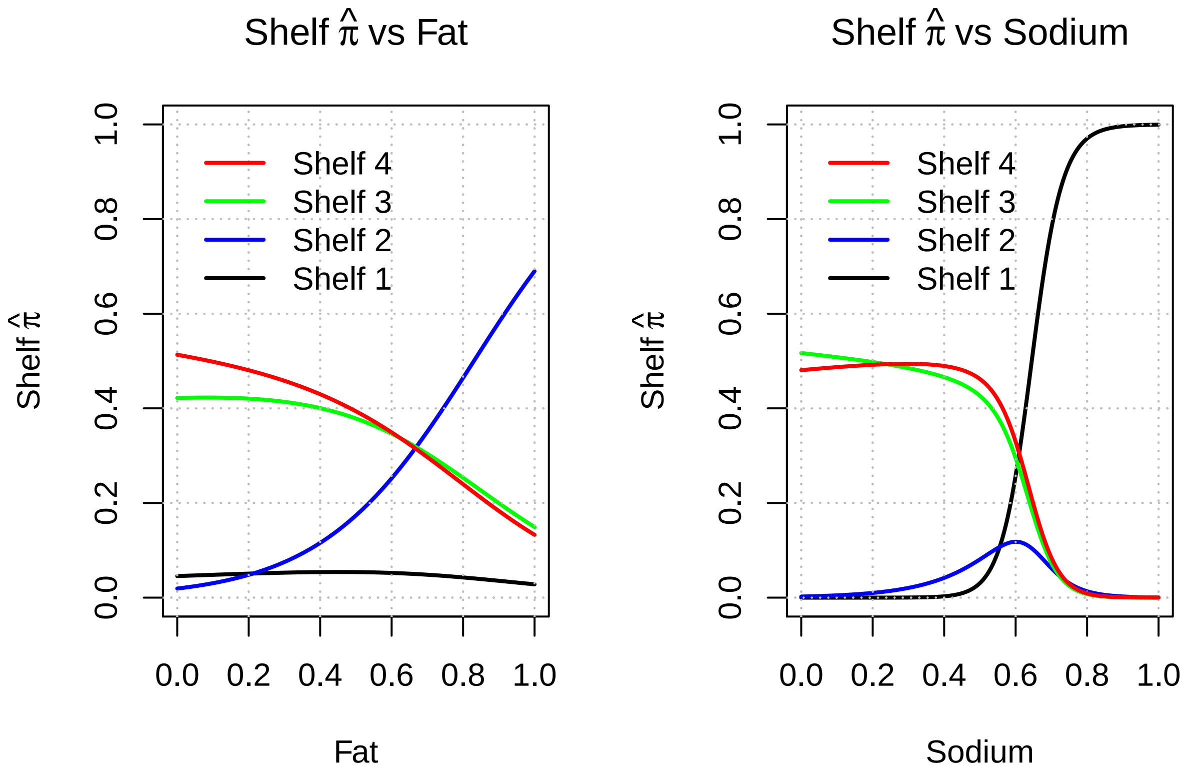
```r
curve(expr = predict(object = mod.fit, newdata = data.frame(sugar = x,
      fat = mean(cereal2$fat), sodium = mean(cereal2$sodium)), type = "probs")[,1],
      main= expression(Shelf~hat(pi)~"vs Sugar content"),
      ylab = expression(Shelf~hat(pi)), xlab = "Sugar", xlim = c(min(cereal2$sugar), max(cerea
      ylim = c(0,1), col = "black", lty = "solid", lwd = 2, n = 1000,
      panel.first = grid(col = "gray", lty = "dotted"))

curve(expr = predict(object = mod.fit, newdata = data.frame(sugar = x,
      fat = mean(cereal2$fat), sodium = mean(cereal2$sodium)), type = "probs")[,2],
      col = "blue", lty = "solid", lwd = 2, n = 1000,
      add = TRUE, panel.first = grid(col = "gray", lty = "dotted"))

curve(expr = predict(object = mod.fit, newdata = data.frame(sugar = x,
      fat = mean(cereal2$fat), sodium = mean(cereal2$sodium)), type = "probs")[,3],
```

```
      col = "green", lty = "solid", lwd = 2, n = 1000,
      add = TRUE, panel.first = grid(col = "gray", lty = "dotted"))

curve(expr = predict(object = mod.fit, newdata = data.frame(sugar = x,
      fat = mean(cereal2$fat), sodium = mean(cereal2$sodium)), type = "probs")[,4],
      col = "red", lty = "solid", lwd = 2, n = 1000,
      add = TRUE, panel.first = grid(col = "gray", lty = "dotted"))

legend(x=0.2,y=1, legend=c("Shelf 4","Shelf 3","Shelf 2", "Shelf 1"), lty=c("solid"), col=c("re
```




```
par(mfrow=c(1,2))
curve(expr = predict(object = mod.fit, newdata = data.frame(sugar = mean(cereal2$sugar),
      fat =x, sodium = mean(cereal2$sodium)), type = "probs")[,1],
      main= expression(Shelf~hat(pi)~"vs Fat"),
      ylab = expression(Shelf~hat(pi)), xlab = "Fat", xlim = c(min(cereal2$fat), max(cereal2$fa
      ylim = c(0,1), col = "black", lty = "solid", lwd = 2, n = 1000,
      panel.first = grid(col = "gray", lty = "dotted"))

curve(expr = predict(object = mod.fit, newdata = data.frame(sugar = mean(cereal2$sugar),
      fat =x, sodium = mean(cereal2$sodium)), type = "probs")[,2],
      col = "blue", lty = "solid", lwd = 2, n = 1000,
      add = TRUE, panel.first = grid(col = "gray", lty = "dotted"))
```

```
curve(expr = predict(object = mod.fit, newdata = data.frame(sugar = mean(cereal2$sugar),
      fat =x, sodium = mean(cereal2$sodium)), type = "probs")[,3],
      col = "green", lty = "solid", lwd = 2, n = 1000,
      add = TRUE, panel.first = grid(col = "gray", lty = "dotted"))

curve(expr = predict(object = mod.fit, newdata = data.frame(sugar = mean(cereal2$sugar),
      fat =x, sodium = mean(cereal2$sodium)), type = "probs")[,4],
      col = "red", lty = "solid", lwd = 2, n = 1000,
      add = TRUE, panel.first = grid(col = "gray", lty = "dotted"))

legend(x=0,y=1, legend=c("Shelf 4","Shelf 3","Shelf 2", "Shelf 1"), lty=c("solid"), col=c("red

curve(expr = predict(object = mod.fit, newdata = data.frame(sugar = mean(cereal2$sugar),
      fat = mean(cereal2$fat), sodium = x), type = "probs")[,1],
      main= expression(Shelf~hat(pi)~"vs Sodium"),
      ylab = expression(Shelf~hat(pi)), xlab = "Sodium", xlim = c(min(cereal2$sodium), max(cere
      ylim = c(0,1), col = "black", lty = "solid", lwd = 2, n = 1000,
      panel.first = grid(col = "gray", lty = "dotted"))

curve(expr = predict(object = mod.fit, newdata = data.frame(sugar = mean(cereal2$sugar),
      fat = mean(cereal2$fat), sodium = x), type = "probs")[,2],
      col = "blue", lty = "solid", lwd = 2, n = 1000,
      add = TRUE, panel.first = grid(col = "gray", lty = "dotted"))

curve(expr = predict(object = mod.fit, newdata = data.frame(sugar = mean(cereal2$sugar),
      fat = mean(cereal2$fat), sodium = x), type = "probs")[,3],
      col = "green", lty = "solid", lwd = 2, n = 1000,
      add = TRUE, panel.first = grid(col = "gray", lty = "dotted"))

curve(expr = predict(object = mod.fit, newdata = data.frame(sugar = mean(cereal2$sugar),
      fat = mean(cereal2$fat), sodium = x), type = "probs")[,4],
      col = "red", lty = "solid", lwd = 2, n = 1000,
      add = TRUE, panel.first = grid(col = "gray", lty = "dotted"))

legend(x=0,y=1, legend=c("Shelf 4","Shelf 3","Shelf 2", "Shelf 1"), lty=c("solid"), col=c("red
```



Shelf $\hat{\pi}$ vs Fat
Shelf 4
Shelf 3
Shelf 2
Shelf 1
Shelf $\hat{\pi}$
0.0
0.2
0.4
0.6
0.8
1.0
Fat
Shelf $\hat{\pi}$ vs Sodium
Shelf 4
Shelf 3
Shelf 2
Shelf 1
Shelf $\hat{\pi}$
Sodium

h. Estimate odds ratios and calculate corresponding confidence intervals for each explanatory variable. Relate your interpretations back to the plots constructed for this exercise.

```
# Information about each variable to help with choosing c. Leave out Shelf column
sd.cereal2<-apply(X = cereal2[,-c(1)], MARGIN = 2, FUN = sd)
# sd.cereal2

#convert sd into (g/serving) units for interpretation. 0-1 are percentages of the overall rang
sd_convert<-function(sd,df_column,df_serving=cereal$size_g){
  var_range<-max(df_column/df_serving)-min(df_column/df_serving)
  return(sd*var_range)
}


c.value<-c(1, sd.cereal2)  # class = 1 is first value
c.value<-c.value[2:4] # drop intercept from c.value
round(c.value,2)
```

```
##  sugar    fat sodium
##   0.27   0.30   0.23
```

```
units<-c(g_serving=sd_convert(c.value[1],cereal$sugar_g),g_serving=sd_convert(c.value[2],cereal
round(units,2)
```

```
##   g_serving.sugar     g_serving.fat mg_serving.sodium
##              0.15              0.03              2.46
```

```
# beta.hat_jr for r = 1, 2, 3  and j = 2, 3, 4
beta.hat2<-coefficients(mod.fit)[1,2:4]
beta.hat3<-coefficients(mod.fit)[2,2:4]
beta.hat4<-coefficients(mod.fit)[3,2:4]
```

```
# Odds ratios for j = 2 vs. j = 1
OR2_1<-exp(c.value*beta.hat2)
OR1_2<-1/exp(c.value*beta.hat2)

# Odds ratios for j = 3 vs. j = 2
OR3_2<-exp(c.value*beta.hat3)
OR2_3<-1/exp(c.value*beta.hat3)

# for j = 3 vs j = 1
OR3_1<-OR3_2*OR2_1
OR1_3<-1/OR3_1

# Odds ratios for j = 4 vs. j = 3
OR4_3<-exp(c.value*beta.hat4)
OR3_4<-1/exp(c.value*beta.hat4)

# for j = 4 vs j = 1
OR4_1<-OR4_3*OR3_1
```

```r
OR1_4<-1/OR4_1

# for j = 4 vs j = 2
OR4_2<-OR4_3*OR3_2
OR2_4<-1/OR4_2
```

```r
#build dataframes
OR_base=data.frame(OR2_1=round(OR2_1,2),
                   OR3_1=round(OR3_1,2),
                   OR4_1=round(OR4_1,2),
                   "-"=c("-","-","-"),
                   OR1_2=round(OR1_2,2),
                   OR3_2=round(OR3_2,2),
                   OR4_2=round(OR4_2,2),
                   "-"=c("-","-","-"),
                   OR1_3=round(OR1_3,2),
                   OR2_3=round(OR2_3,2),
                   OR4_3=round(OR4_3,2),
                   "-"=c("-","-","-"),
                   OR1_4=round(OR1_4,2),
                   OR2_4=round(OR2_4,2),
                   OR3_4=round(OR3_4,2))
```

```r
OR_base
```

```
##        OR2_1 OR3_1 OR4_1 X. OR1_2 OR3_2 OR4_2 X..1    OR1_3  OR2_3 OR4_3
## sugar   2.06  0.08   0.0  -  0.48  0.04  0.00    -    12.98  26.81  0.05
## fat     3.37  2.85   2.2  -  0.30  0.85  0.65    -     0.35   1.18  0.77
## sodium  0.02  0.00   0.0  - 55.74  0.00  0.00    - 17355.07 311.36  0.00
##        X..2      OR1_4    OR2_4  OR3_4
## sugar     -     278.95   575.96  21.48
## fat       -       0.45     1.53   1.30
## sodium    - 5038277.64 90390.00 290.31
```

```r
# Wald CIs
conf.beta<-confint(object = mod.fit, level = 0.95)
# round(conf.beta,2)  # Results are stored in a 3D array
# conf.beta[2:4,1:2,1]  # C.I.s for beta_2r
# conf.beta[2:4,1:2,2]  # C.I.s for beta_3r
# conf.beta[2:4,1:2,3]  # C.I.s for beta_4r
#CI for probability based on variable entry

# CIs for OR
ci.OR2<-exp(c.value*conf.beta[2:4,1:2,1])
ci.OR3<-exp(c.value*conf.beta[2:4,1:2,2])
ci.OR4<-exp(c.value*conf.beta[2:4,1:2,3])

"Shelf 2,3,4 vs Shelf 1"
```

```
## [1] "Shelf 2,3,4 vs Shelf 1"
```

```
round(data.frame(low = ci.OR2[,1], up = ci.OR2[,2]), 2) #RELATIVE TO SHELF 1
```

```
##          low    up
## sugar   0.14 29.68
## fat     0.87 13.04
## sodium  0.00  0.44
```

```
round(data.frame(low = ci.OR3[,1], up = ci.OR3[,2]), 2)
```

```
##          low   up
## sugar   0.00 0.49
## fat     0.21 3.49
## sodium  0.00 0.12
```

```
round(data.frame(low = ci.OR4[,1], up = ci.OR4[,2]), 2)
```

```
##          low   up
## sugar   0.00 0.61
## fat     0.19 3.16
## sodium  0.00 0.13
```

```
"Shelf 3 vs Shelf 2"
```

```
## [1] "Shelf 3 vs Shelf 2"
```

```
round(data.frame(low = ci.OR3[,1]/ci.OR2[,1], up = ci.OR3[,2]/ci.OR2[,2]), 2) #shelf 3 relativ
```

```
##          low   up
## sugar   0.02 0.02
## fat     0.24 0.27
## sodium  0.11 0.28
```

```
"Shelf 4 vs Shelf 3"
```

```
## [1] "Shelf 4 vs Shelf 3"
```

```
round(data.frame(low = ci.OR4[,1]/ci.OR3[,1], up = ci.OR4[,2]/ci.OR3[,2]), 2) #shelf 4 relativ
```

```
##          low   up
## sugar   1.26 1.24
## fat     0.92 0.91
## sodium  1.08 1.06
```

Odds ratio interpretations: The c values (in $e^{c\beta_{jr}}$) used to evaluate the odds ratios for each component were calculated as 1 standard deviation unit. In grams per serving, these c values are 0.15 g/serving of sugar, 0.03 g/serv fat, and 2.46 mg/serv sodium.

Evaluating sugar first, for each c-value increase in sugar, the odds that the cereal would be on shelf 3 are 21.48 times as large as the odds it would be on shelf 4, 12.98 times as large that it'd be on shelf 1 as shelf 3, and 2.06 times as large that it would be on shelf 2 as shelf 1. This matches the $\hat{\pi}$ graph for sugar as shelves 1 and 2 have the highest probabilities for increased sugar content, however doesn't match the relatioship between 3 and 4. This discrepancy may be a result of a smoothing error, as shelf 3 has an overall wider range of sugar values, including some at higher sugar levels as the odds ratios suggest, but also has more samples at lower sugar values than shelf 4. Incorporating the confidence intervals, the CI of the odds ratio that the cereal will appear on shelf 2 vs shelf 1 includes 1, so we cannot definitively say the odds the cereal will be on shelf 2 are larger than shelf 1. We can, however, say that the odds the cereal will be on shelves 3 or 4 relative to shelf 1 are less than 1 for each unit increase in sugar.

Looking at fat, again the odds that a cereal will be on shelf 2 are at least twice as large as the odds of the other shelves. The $\hat{\pi}$ graphs echo this, for as the fat content increases, the likelihood the cereal will appear 2 increases while all others decrease. Again, The odds ratio confidence intervals span 1 for shelf 2 vs shelf 1, however are entirely below 1 for shelf 3 vs shelf 2 and shelf 4 vs shelf 3, indicating that the odds of the cereal to be on shelf 2 are larger than shelf 3 or 4 for an increase in fat.

Now, the odds ratios for sodium reach extreme numbers, especially relative to shelf 1. The behavior of the $/hat/pi$ vs sodium content for shelf 1 increases from 0 to 1 in about 0.3 units, almost the same as our c-value, potentially because only high-sodium samples appear on shelf 1. This is reiterated by the confidence intervals, as all the other shelf intervals are below 0.5 relative to shelf 1. This means that for each unit increase in sodium, the odds that a cereal appears on any other shelf are at most 1/2 as large as the odds that the cereal appears on shelf 1.