

Mark Paluta, Carlos Sancini, Krysten Thompson (W271): Lab 2

Professor Jeffrey Yau

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.

```
library(knitr)
opts_chunk$set(tidy.opts=list(width.cutoff=60),tidy=TRUE)
```

```
rm(list = ls())
```

```
# Load Libraries
```

```
library(dplyr)
library(car)
library(Hmisc)
library(skimr)
library(ggplot2)
library(gmodels)
library(vcd)
library(nnet)
```

```
df <- read.csv("cereal_dillons.csv")
#head(df)
```

```
str(df)
```

```
## 'data.frame':   40 obs. of  7 variables:
## $ ID          : int  1 2 3 4 5 6 7 8 9 10 ...
## $ Shelf       : int  1 1 1 1 1 1 1 1 1 1 ...
## $ Cereal      : Factor w/ 38 levels "Basic 4","Capn Crunch",...: 17 34 19 13 16 9 2 3 30 8 ...
## $ size_g      : int  28 28 28 32 30 31 27 27 29 33 ...
## $ sugar_g     : int  10 2 2 2 13 11 12 9 11 2 ...
## $ fat_g       : num  0 0 0 2 1 0 1.5 2.5 0.5 0 ...
```

```
## $ sodium_mg: int 170 270 300 280 210 180 200 200 220 330 ...
```

```
#describe(df)
```

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.

```
df$sugar_g = df$sugar_g/df$size_g
df$fat_g = df$fat_g/df$size_g
df$sodium_mg = df$sodium_mg/df$size_g
df$Shelf = as.factor(df$Shelf)
```

- Second, rescale each variable to be within 0 and 1.

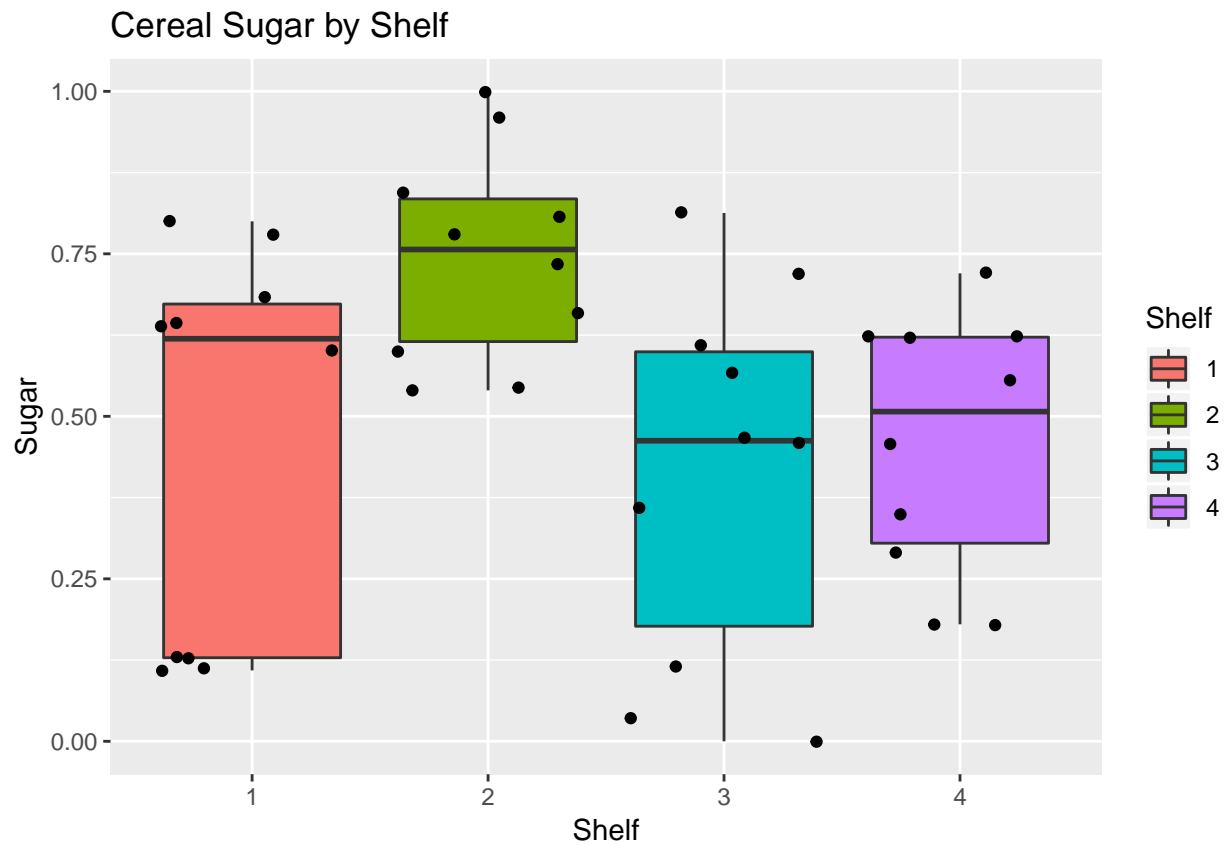
```
normalize = function(x) {
  (x - min(x))/(max(x) - min(x))
}
```

```
df = cbind(df[, 1:4], apply(df[,5:7], 2, normalize))
head(df) #confirmed normalization worked
```

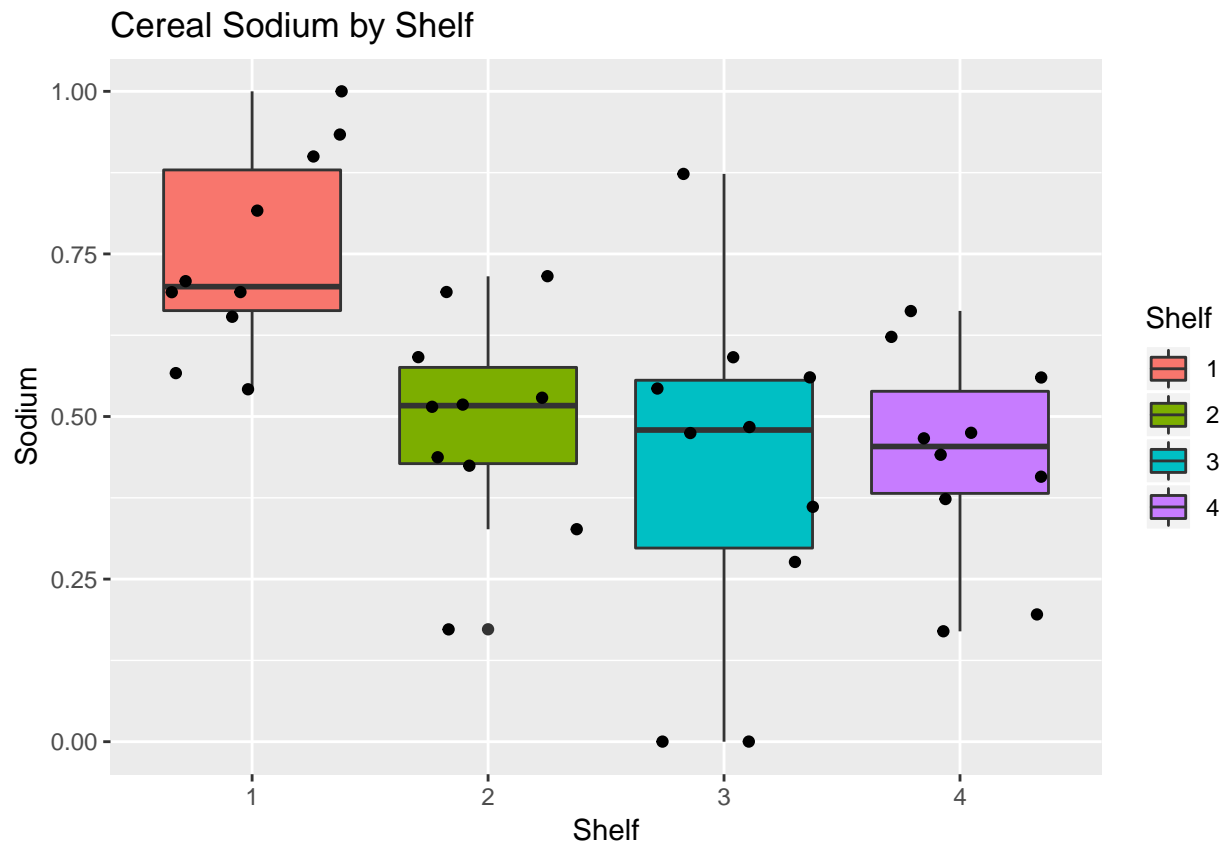
```
## ID Shelf Cereal size_g sugar_g fat_g
## 1 1 1 Kellogg's Razzle Dazzle Rice Crispies 28 0.6428571 0.000
## 2 2 1 Post Toasties Corn Flakes 28 0.1285714 0.000
## 3 3 1 Kellogg's Corn Flakes 28 0.1285714 0.000
## 4 4 1 Food Club Toasted Oats 32 0.1125000 0.675
## 5 5 1 Frosted Cheerios 30 0.7800000 0.360
## 6 6 1 Food Club Frosted Flakes 31 0.6387097 0.000
## sodium_mg
## 1 0.5666667
## 2 0.9000000
## 3 1.0000000
## 4 0.8166667
## 5 0.6533333
## 6 0.5419355
```

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

```
ggplot(df, aes(Shelf, sugar_g)) +
  geom_boxplot(aes(fill = Shelf)) +
  geom_jitter() +
  ylab("Sugar") +
  xlab("Shelf") +
  ggtitle("Cereal Sugar by Shelf")
```



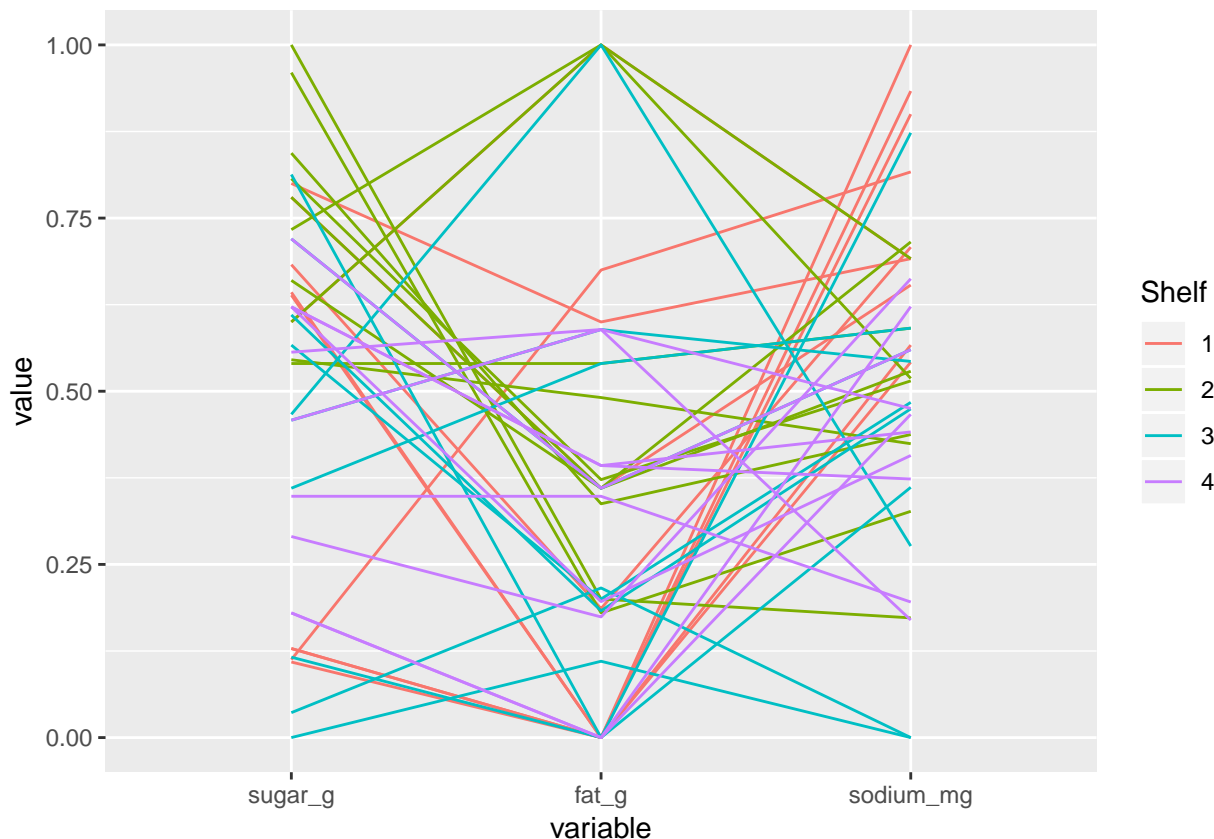
```
ggplot(df, aes(Shelf, fat_g)) +  
  geom_boxplot(aes(fill = Shelf)) +  
  geom_jitter() +  
  ylab("Fat") +  
  xlab("Shelf") +  
  ggtitle("Cereal Fat by Shelf")
```

- Also, construct a **parallel coordinates plot** for the explanatory variables and the shelf

*#This library is an extension of ggplot2 and has functions that
#reduce complexity of combining geometric objects with transformed data; it was
#recommended when researching parallel coordinates plots*

```
library(GGally, quietly = TRUE, warn.conflicts = FALSE)
df$Shelf <- as.factor(df$Shelf)
ggparcoord(df, columns = 5:7, groupColumn='Shelf', scale = 'globalminmax')
```



Based on the parallel coordinates plot, Shelf 1 appears to contain cereals with low levels of fat, high levels of sodium, and a mix of sugar levels (some high and some low).

Shelf 2 contains cereals with some of the highest levels of sugar, moderate to high levels of fat, and a mix of sodium levels (some high, some low). This appears to be the shelf focused on children.

Shelf 3 contains cereals that span both low and high levels of sugar, fat, and sodium. This shelf could target adults who monitor their intake of any of these ingredients.

Shelf 4 trends similarly to Shelf 3, with cereals spanning the range of low to high for all three ingredients.

- 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 of the dependent variable is a good data model when we feel confident in a proportional odds model. Given our knowledge of grocery store marketing strategy, we tend to think a proportional odds model is not a good model. Desirability of shelves may not vary predictably from shelf to shelf. There may be “sweet spots”, depending on the type of cereal. For example, let’s say that the average child’s height is at shelf 2. Knowing that there is more sugar in a cereal (all else being equal), may mean a large jump in cumulative odds at shelf 2 but rather small jumps adding in shelves 3 & 4. We are concerned this may lead to a violation of proportional odds that will cause us to have a less useful model than simply treating the shelves as categorical data.

- 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.

Multinomial Regression Formula

$$\hat{\pi}_j = \frac{\exp(\beta_j 0 + \beta_{j1}x_1 + \cdots + \beta_{jp}x_p)}{1 + \sum_{j=2}^J \exp(\beta_j 0 + \beta_{j1}x_1 + \cdots + \beta_{jp}x_p)}$$

Likelihood Ratio Formula

$$\begin{aligned} -2\log(\Lambda) &= -2\log\left(\frac{L(\hat{\beta}^{(0)}|y_1, \dots, y_n)}{L(\hat{\beta}^{(a)}|y_1, \dots, y_n)}\right) \\ &= -2\sum y_i \log\left(\frac{\hat{\pi}_i^{(0)}}{\hat{\pi}_i^{(a)}}\right) + (1 - y_i) \log\left(\frac{1 - \hat{\pi}_i^{(0)}}{1 - \hat{\pi}_i^{(a)}}\right) \end{aligned}$$

```
#Transformed Shelf back to Int for modeling
```

```
df$Shelf <- as.integer(df$Shelf)
```

```
mod.fit <- multinom(formula = Shelf ~ sugar_g + fat_g + sodium_mg, data=df,
                    epsilon = 0.0001, maxit = 1000, trace = F) #trace=F is suppressing convergence
summary(mod.fit)
```

```
## Call:
```

```
## multinom(formula = Shelf ~ sugar_g + fat_g + sodium_mg, data = df,
```

```
##     epsilon = 1e-04, maxit = 1000, trace = F)
```

```
##
```

```
## Coefficients:
```

```
## (Intercept)    sugar_g    fat_g sodium_mg
```

```
## 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_g    fat_g sodium_mg
```

```
## 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
```

```
round(coef(mod.fit), 2)
```

```
## (Intercept) sugar_g fat_g sodium_mg
```

```
## 2          6.90    2.69    4.06    -17.49
```

```
## 3          21.68   -12.22   -0.56    -24.98
```

```
## 4          21.29   -11.39   -0.87    -24.67
```

```
Anova(mod.fit)
```

```
## Analysis of Deviance Table (Type II tests)
```

```
##
```

```
## Response: Shelf
##           LR Chisq Df Pr(>Chisq)
## sugar_g    22.7648  3  4.521e-05 ***
## fat_g       5.2836  3    0.1522
## sodium_mg  26.6197  3  7.073e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The model equations:

$$\log(\hat{\pi}_2/\hat{\pi}_1) = 6.90 + 2.69 * sugar + 4.06 * fat - 17.49 * sodium$$

$$\log(\hat{\pi}_3/\hat{\pi}_1) = 21.68 - 12.22 * sugar - 0.56 * fat - 24.98 * sodium$$

$$\log(\hat{\pi}_4/\hat{\pi}_1) = 21.29 - 11.39 * sugar - 0.87 * fat - 24.67 * sodium$$

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

```
# epsilon = 0.0001 and maxit = 1000 were set since some models were not converging
model.H0 = multinom(
  formula = Shelf ~ sugar_g + fat_g + sodium_mg,
  data=df, trace = F, epsilon = 0.0001, maxit = 1000)

model.sugar.fat = multinom(
  formula = Shelf ~ sugar_g + fat_g + sodium_mg + sugar_g:fat_g,
  data=df, trace = F, epsilon = 0.0001, maxit = 1000)

model.sugar.sodium = multinom(
  formula = Shelf ~ sugar_g + fat_g + sodium_mg + sugar_g:sodium_mg,
  data=df, trace = F, epsilon = 0.0001, maxit = 1000)

model.fat.sodium = multinom(
  formula = Shelf ~ sugar_g + fat_g + sodium_mg + fat_g:sodium_mg,
  data=df, trace = F, epsilon = 0.0001, maxit = 1000)

model.sugar.fat.sodium = multinom(
  formula = Shelf ~ sugar_g + fat_g + sodium_mg + sugar_g:fat_g:sodium_mg,
  data=df, trace = F, epsilon = 0.0001, maxit = 1000)

# interactions significance test
print('Null vs Sugar:Fat')
```

```
## [1] "Null vs Sugar:Fat"
```

```
anova(model.H0, model.sugar.fat)
```

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

```
##
```

```
## Response: Shelf
```

##		Model	Resid. df	Resid. Dev	Test
## 1	sugar_g + fat_g + sodium_mg		108	67.19028	
## 2	sugar_g + fat_g + sodium_mg + sugar_g:fat_g		105	61.81491	1 vs 2


```

##      Df LR stat.   Pr(Chi)
## 1
## 2      3 5.375364 0.1462862

print('Null vs Sugar:Sodium')

## [1] "Null vs Sugar:Sodium"

anova(model.H0, model.sugar.sodium)

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

print('Null vs Fat:Sodium')

## [1] "Null vs Fat:Sodium"

anova(model.H0, model.fat.sodium)

## Likelihood ratio tests of Multinomial Models
##
## Response: Shelf
##
##              Model Resid. df Resid. Dev
## 1              sugar_g + fat_g + sodium_mg      108   67.19028
## 2 sugar_g + fat_g + sodium_mg + fat_g:sodium_mg      105   60.72048
##      Test      Df LR stat.   Pr(Chi)
## 1
## 2 1 vs 2      3 6.469799 0.09086119

print('Null vs Sugar:Fat:Sodium')

## [1] "Null vs Sugar:Fat:Sodium"

anova(model.H0, model.sugar.fat.sodium)

## Likelihood ratio tests of Multinomial Models
##
## Response: Shelf
##
##              Model Resid. df
## 1              sugar_g + fat_g + sodium_mg      108
## 2 sugar_g + fat_g + sodium_mg + sugar_g:fat_g:sodium_mg      105
## Resid. Dev   Test      Df LR stat.   Pr(Chi)
## 1    67.19028
## 2    65.04570 1 vs 2      3  2.14458 0.5429468

```

The LRT hypothesis tests for each of the interactions shows that none of them have

low enough p-values to be considered statistically significant.

- 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.

```
df_unscaled <- read.csv("cereal_dillons.csv")
```

```
new_data = data.frame(
  sugar_g = 12/28/max(df_unscaled$sugar_g),
  fat_g = .5/28/max(df_unscaled$fat_g),
  sodium_mg = 130/28/max(df_unscaled$sodium_mg))
pi.hat = predict(object = mod.fit, newdata = new_data, type = "probs")
round(pi.hat*100,1)
```

```
##      1      2      3      4
## 0.0  0.0 59.2 40.8
```

- 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.

```
fat_mean <- mean(df$fat_g)
```

```
sodium_mean <- mean(df$sodium_mg)
```

```
# Estimate model
```

```
mod.fit.sugar <- multinom(formula = Shelf ~ sugar_g + fat_g + sodium_mg, data = df, trace=F) #
beta.hat <- coefficients(mod.fit.sugar)
```

```
# Plot each pi_j
```

```
#Shelf 1
```

```
curve(expr = 1/
  (1 + exp(beta.hat[1,1] + beta.hat[1,2]*x + beta.hat[1,3]*fat_mean + beta.hat[1,4]*sodium_mean) +
  exp(beta.hat[2,1] + beta.hat[2,2]*x + beta.hat[2,3]*fat_mean + beta.hat[2,4]*sodium_mean) +
  exp(beta.hat[3,1] + beta.hat[3,2]*x + beta.hat[3,3]*fat_mean + beta.hat[3,4]*sodium_mean)),
  col = "black", lty = "solid", lwd = 2, n = 40,
  xlim = c(min(df$sugar_g), max(df$sugar_g)), ylim=c(0,1), xlab = "Sugar (normalized)",
  ylab = expression(hat(pi)), panel.first = grid(col = "gray", lty = "dotted"))
```

```
# Shelf 2
```

```
curve(expr = exp(beta.hat[1,1] + beta.hat[1,2]*x + beta.hat[1,3]*fat_mean + beta.hat[1,4]*sodium_mean) /
  (1 + exp(beta.hat[1,1] + beta.hat[1,2]*x + beta.hat[1,3]*fat_mean + beta.hat[1,4]*sodium_mean) +
  exp(beta.hat[2,1] + beta.hat[2,2]*x + beta.hat[2,3]*fat_mean + beta.hat[2,4]*sodium_mean) +
  exp(beta.hat[3,1] + beta.hat[3,2]*x + beta.hat[3,3]*fat_mean + beta.hat[3,4]*sodium_mean)),
  col = "green", lty = "dotdash", lwd = 2, n = 40, add = TRUE)
```

```
#Shelf 3
```

```
curve(expr = exp(beta.hat[2,1] + beta.hat[2,2]*x + beta.hat[2,3]*fat_mean + beta.hat[2,4]*sodium_mean) /
  (1 + exp(beta.hat[1,1] + beta.hat[1,2]*x + beta.hat[1,3]*fat_mean + beta.hat[1,4]*sodium_mean) +
  exp(beta.hat[2,1] + beta.hat[2,2]*x + beta.hat[2,3]*fat_mean + beta.hat[2,4]*sodium_mean) +
  exp(beta.hat[3,1] + beta.hat[3,2]*x + beta.hat[3,3]*fat_mean + beta.hat[3,4]*sodium_mean)),
  col = "red", lty = "dotted", lwd = 2, n = 40, add = TRUE)
```

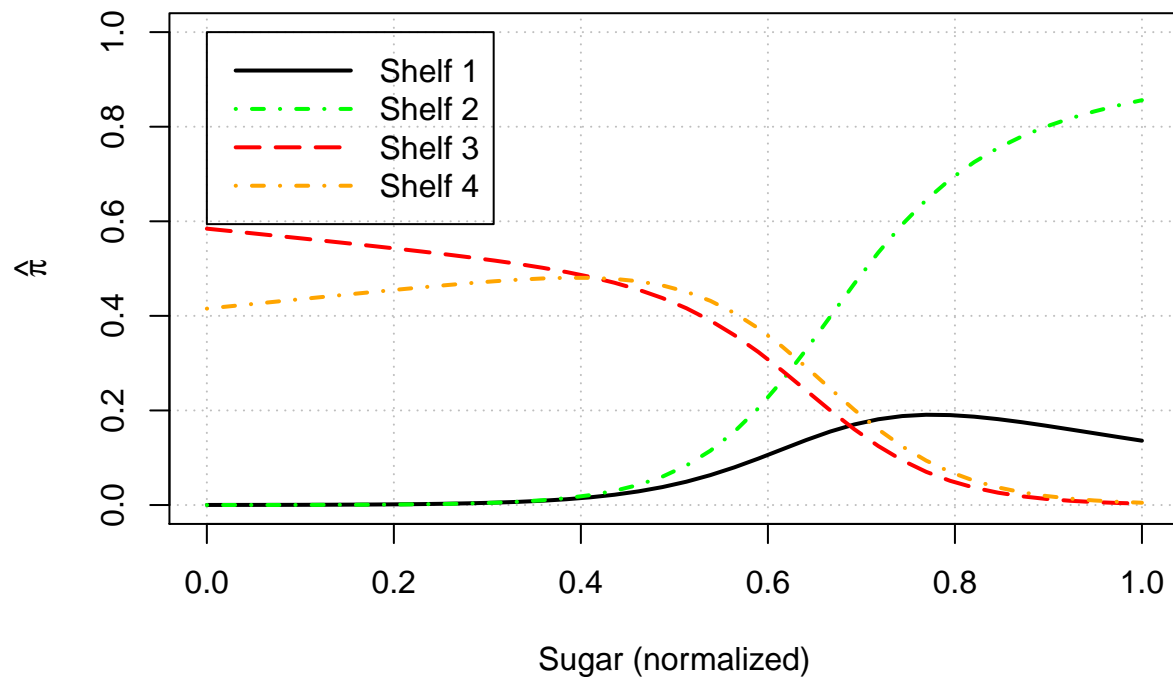
```

exp(beta.hat[2,1] + beta.hat[2,2]*x + beta.hat[2,3]*fat_mean + beta.hat[2,4]*sodium,
exp(beta.hat[3,1] + beta.hat[3,2]*x + beta.hat[3,3]*fat_mean + beta.hat[3,4]*sodium,
col = "red", lty = "longdash", lwd = 2, n = 40, add = TRUE)

# Shelf 4
curve(expr = exp(beta.hat[3,1] + beta.hat[3,2]*x + beta.hat[3,3]*fat_mean + beta.hat[3,4]*sodium,
(1 + exp(beta.hat[1,1] + beta.hat[1,2]*x + beta.hat[1,3]*fat_mean + beta.hat[1,4]*sodium,
exp(beta.hat[2,1] + beta.hat[2,2]*x + beta.hat[2,3]*fat_mean + beta.hat[2,4]*sodium,
exp(beta.hat[3,1] + beta.hat[3,2]*x + beta.hat[3,3]*fat_mean + beta.hat[3,4]*sodium,
col = "orange", lty = "dotdash", lwd = 2, n = 40, add = TRUE)

#Legend
legend(x = 0, y = 1, legend=c("Shelf 1", "Shelf 2", "Shelf 3", "Shelf 4"),
lty=c("solid","dotdash", "longdash","dotdash"),
col=c("black","green","red", "orange"), lwd = c(2,2,2,2), seg.len = 4)

```



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

```

# Use one standard deviation as c values for interpretability
sd.cereal <- apply(X = df[, -c(1,3:4)], MARGIN = 2, FUN = sd)
c.value <- c(1, sd.cereal)
round(c.value, 2)

```

```

##           Shelf  sugar_g  fat_g sodium_mg
##      1.00    1.13    0.27    0.30    0.23

```

```

# Store Wald confidence intervals
conf.int <- confint(object = mod.fit, level = 0.95)

```

```
# Odds Ratio Confidence Interval - Shelf 2 to Shelf 1  
round(exp(conf.int[2:4,,1]*c.value[3]),2)
```

```
##          2.5 % 97.5 %  
## sugar_g    0.14  29.68  
## fat_g      0.88  10.09  
## sodium_mg  0.00   0.38
```

```
# Odds Ratio Confidence Interval - Shelf 3 to Shelf 1  
round(exp(conf.int[2:4,,2]*c.value[4]),2)
```

```
##          2.5 % 97.5 %  
## sugar_g    0.00   0.45  
## fat_g      0.21   3.49  
## sodium_mg  0.00   0.06
```

```
# Odds Ratio Confidence Interval - Shelf 4 to Shelf 1  
round(exp(conf.int[2:4,,3]*c.value[5]),2)
```

```
##          2.5 % 97.5 %  
## sugar_g    0.01   0.65  
## fat_g      0.28   2.42  
## sodium_mg  0.00   0.13
```