Problem Set 3

Topics in Advanced Econometrics (ResEcon 703)
University of Massachusetts Amherst

Solutions

Rules

Email a single .pdf file of your problem set writeup, code, and output to mwoerman@umass.edu by
the date and time above. You may work in groups of up to three, and all group members can submit
the same code and output; indicate in your writeup who you worked with. You must submit a unique
writeup that answers the problems below. You can discuss answers with your fellow group members,
but your writeup must be in your own words. Problem 1 requires you to code your own estimators,
while Problem 2 allows you to use R's "canned routines." Each problem will indicate which R function
to use.

Data

Download the file `ps3_datasets.zip` from the course website (`github.com/woerman/ResEcon703`).
This zipped file contains two datasets, `travel_binary.csv` and `phones.csv`, that you will use for this
problem set. The `travel_binary.csv` dataset contains simulated data on the travel mode choice of
1000 UMass graduate students commuting to campus in the middle of winter when only driving a car
or taking a bus are feasible options (assume the weather is too severe for even the heartiest graduate
students to ride a bike or walk). The `phones.csv` dataset contains simulated data from 1000 customers
on the purchases or pre-orders of highly-anticipated phone models recently released by Apple and Google:
iPhone 11 and Pixel 4. See the file `data_descriptions.txt` for descriptions of the variables in each
dataset.

```
### Load packages for problem set
library(tidyverse)

## - Attaching packages ------------------ tidyverse 1.2.1 -
## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
## - Conflicts ------------------- tidyverse_conflicts() -
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(gmm)
```

```
## Loading required package:   sandwich

library(mlogit)

## Loading required package:   Formula
## Loading required package:   zoo
##
## Attaching package:   'zoo'
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
## Loading required package:   lmtest
```

Problem 1: Generalized Method of Moments

Use the `travel_binary.csv` dataset for this question.

a. Model the choice to drive to campus during winter as a binary logit model. Include the cost and the time of each alternative as explanatory variables with common coefficients; do not include an intercept. That is, the representative utility for alternative j is simply

$$V_{nj} = \beta_1 C_{nj} + \beta_2 T_{nj}$$

where C_j is the cost of alternative j and T_j is the time of alternative j. Estimate the parameters of this model by method of moments. The following steps can provide a rough guide to creating your own method of moments estimator:

   I. Write down moment conditions for this model. In this case, a set of two moment conditions based on residuals and data.

   II. Create a function that takes a set of parameters and a *matrix* of data as inputs: `function(parameters, data_matrix)`.

   III. Within that function, make the following calculations:

      i. Calculate the representative utility for each alternative and for each decision maker.
      ii. Calculate the choice probability of driving for each decision maker.
      iii. Calculate the econometric residual, or the difference between the outcome and the probability, for each decision maker.
      iv. Calculate each of the M moments for each decision maker.
      v. Return the $N \times M$ matrix of individual moments.

   IV. Find the MM estimator using `gmm()`. Your call of the `gmm()` function may look something like:

```
gmm(g = your_function, x = your_data_matrix, t0 = your_starting_guesses,
      vcov = 'iid', control = list(reltol = 1e-25, maxit = 10000))
```

Report your parameter estimates, standard errors, t-stats, and p-values. Briefly interpret these results.

Hint: For a binary logit model, characterizing one choice probability is sufficient because the two probabilities must sum to 100%. The probability of driving is

$$P_{nc} = \frac{e^{V_{nc}}}{e^{V_{nc}} + e^{V_{nb}}}$$
$$= \frac{1}{1 + e^{-V_{nc} + V_{nb}}}$$

Both of these expressions for the probability of driving may be useful as you solve this problem.

```
### Part a
## Load dataset
travel_binary <- read_csv('travel_binary.csv')

## Parsed with column specification:
## cols(
##   mode = col_character(),
##   time_car = col_double(),
##   cost_car = col_double(),
##   time_bus = col_double(),
##   cost_bus = col_double(),
##   price_gas = col_double(),
##   snowfall = col_double(),
##   car_in_shop = col_double(),
##   age = col_double(),
##   income = col_double(),
##   marital_status = col_character()
## )

## Create matrix of data
data_1a <- travel_binary %>%
  mutate(choice = 1 * (mode == 'car'),
         cost_difference = cost_car - cost_bus,
         time_difference = time_car - time_bus) %>%
  select(choice, cost_difference, time_difference) %>%
  as.matrix()
## Function to calculate binary logit moments
calculate_moments_1a <- function(parameters, data){
  ## Extract explanatory variable data from matrix
  data_x <- data[, -1]
  ## Extract choice data from matrix
  data_y <- data[, 1]
  ## Calculate net utility of alternative given the parameters
  utility <- data_x %*% parameters
  ## Caclculate logit probability of alternative given the parameters
  probability_choice <- 1 / (1 + exp(-utility))
  ## Calculate residuals
  residuals <- data_y - probability_choice
```

```
  ## Create moment matrix
  moments <- c(residuals) * data_x
  return(moments)
}
## Use GMM to estimate model
model_1a <- gmm(calculate_moments_1a, data_1a, c(0, 0), vcov = 'iid',
                control = list(reltol = 1e-25, maxit = 10000))
## Summarize model results
model_1a %>%
  summary()


##
## Call:
## gmm(g = calculate_moments_1a, x = data_1a, t0 = c(0, 0), vcov = "iid",
##     control = list(reltol = 1e-25, maxit = 10000))
##
##
## Method:  twoStep
##
## Coefficients:
##            Estimate     Std. Error    t value      Pr(>|t|)
## Theta[1]  -4.9646e-01   8.8151e-02  -5.6320e+00   1.7816e-08
## Theta[2]  -5.8354e-02   6.5891e-03  -8.8561e+00   8.2867e-19
##
## J-Test: degrees of freedom is 0
##                   J-test                    P-value
## Test E(g)=0:    9.46762506511059e-23   *******
##
## #############
## Information related to the numerical optimization
## Convergence code =   0
## Function eval. =   191
## Gradian eval. =   NA
```

The cost and time parameters are both negative and statistically significant, indicating that both money and time spent commuting reduce utility. These results are intuitive, as we generally think that people like having both money and time.

b. Again model the choice to drive to campus during winter as a binary logit model, but now add an intercept term and alternative-specific coefficients. That is, the representative utilities for the alternatives are

$$V_{nc} = \beta_0 + \beta_1 C_{nc} + \beta_2 T_{nc}$$
$$V_{nb} = \beta_3 C_{nb} + \beta_4 T_{nb}$$

where C_{nj} is the cost of alternative j and T_{nj} is the time of alternative j. Estimate the parameters of this model by method of moments. Follow the same set of steps as in (a), but now you have five parameters and the representative utility calculation is different. Report your parameter estimates, standard errors, t-stats, and p-values. Briefly interpret these results.

```r
### Part b
## Create matrix of data
data_1b <- travel_binary %>%
  mutate(choice = 1 * (mode == 'car'),
         constant = 1,
         cost_bus = -cost_bus,
         time_bus = -time_bus) %>%
  select(choice, constant, cost_car, time_car, cost_bus, time_bus) %>%
  as.matrix()
## Function to calculate binary logit moments
calculate_moments_1b <- function(parameters, data){
  ## Extract explanatory variable data from matrix
  data_x <- data[, -1]
  ## Extract choice data from matrix
  data_y <- data[, 1]
  ## Calculate net utility of alternative given the parameters
  utility <- data_x %*% parameters
  ## Caclculate logit probability of alternative given the parameters
  probability_choice <- 1 / (1 + exp(-utility))
  ## Calculate residuals
  residuals <- data_y - probability_choice
  ## Create moment matrix
  moments <- c(residuals) * data_x
  return(moments)
}
## Use GMM to estimate model
model_1b <- gmm(calculate_moments_1b, data_1b, rep(0, 5), vcov = 'iid',
                control = list(reltol = 1e-25, maxit = 10000))
## Summarize model results
model_1b %>%
  summary()


##
## Call:
## gmm(g = calculate_moments_1b, x = data_1b, t0 = rep(0, 5), vcov = "iid",
##     control = list(reltol = 1e-25, maxit = 10000))
##
##
## Method:  twoStep
##
## Coefficients:
##            Estimate     Std. Error    t value      Pr(>|t|)
## Theta[1]    3.6826e+00   1.5654e+00    2.3525e+00   1.8647e-02
## Theta[2]   -3.2852e+00   8.3576e-01   -3.9308e+00   8.4650e-05
## Theta[3]   -4.8187e-01   7.6660e-02   -6.2858e+00   3.2617e-10
## Theta[4]   -2.9441e+00   2.6936e-01   -1.0930e+01   8.3086e-28
## Theta[5]   -2.4268e-01   2.5520e-02   -9.5092e+00   1.9218e-21
```

```
##
## J-Test: degrees of freedom is 0
##                   J-test              P-value
## Test E(g)=0:    8.74728598718018e-22   *******
##
## #############
## Information related to the numerical optimization
## Convergence code =  0
## Function eval. =  3908
## Gradian eval. =  NA
```

The car intercept is positive and statistically significant, indicating that, *ceteris paribus*, driving a car is preferred to taking the bus. The alternative-specific parameters for cost and time are all negative and statistically significant, indicating that the cost and time of commuting by either alternative reduce utility.

c. You might be concerned that the cost and time of driving to campus are endogenous; for example, a student who enjoys driving is more likely to live farther from campus because they do not mind the extra time spent driving. The `travel_binary.csv` dataset includes three possible instruments: gasoline price (the cost of driving is proportional to the gasoline price), snowfall (the time to drive includes the time to clean off the car and the added time from driving cautiously), and an indicator if the student's car is in the shop (extra time or cost is required to borrow someone else's car). Again model the choice to drive to campus during winter as in (b). That is, the representative utilities for the alternatives are

$$V_{nc} = \beta_0 + \beta_1 C_{nc} + \beta_2 T_{nc}$$
$$V_{nb} = \beta_3 C_{nb} + \beta_4 T_{nb}$$

where C_{nj} is the cost of alternative j and T_{nj} is the time of alternative j. In this model, however, instrument for the cost and time of driving with the instruments described above. Thus, you will have six instruments: constant term, cost to take the bus, time to take the bus, gasoline price, snowfall, and an indicator if car is in the shop. Estimate the parameters of this model by generalized method of moments. Follow the same set of steps as in (a), but now you have five parameters and six moment conditions. Report your parameter estimates, standard errors, t-stats, and p-values. Briefly interpret these results. Also test for overidentifying restrictions in this model and interpret the results of this test.

```
### Part c
## Create matrix of data
data_1c <- travel_binary %>%
  mutate(choice = 1 * (mode == 'car'),
         constant = 1,
         cost_bus = -cost_bus,
         time_bus = -time_bus) %>%
  select(choice, constant, cost_car, time_car, cost_bus, time_bus,
         price_gas, snowfall, car_in_shop) %>%
  as.matrix()
## Function to calculate binary logit moments
```

```r
calculate_moments_1c <- function(parameters, data){
  ## Extract explanatory variable data from matrix
  data_x <- data[, 2:6]
  ## Extract choice data from matrix
  data_y <- data[, 1]
  ## Extract instrument data from matrix
  data_z <- data[, c(2, 5:9)]
  ## Calculate net utility of alternative given the parameters
  utility <- data_x %*% parameters
  ## Caclculate logit probability of alternative given the parameters
  probability_choice <- 1 / (1 + exp(-utility))
  ## Calculate residuals
  residuals <- data_y - probability_choice
  ## Create moment matrix
  moments <- c(residuals) * data_z
  return(moments)
}
## Use GMM to estimate model
model_1c <- gmm(calculate_moments_1c, data_1c, rep(0, 5), vcov = 'iid',
                control = list(reltol = 1e-25, maxit = 10000))
## Summarize model results
model_1c %>%
  summary()
```

```
##
## Call:
## gmm(g = calculate_moments_1c, x = data_1c, t0 = rep(0, 5), vcov = "iid",
##     control = list(reltol = 1e-25, maxit = 10000))
##
##
## Method:  twoStep
##
## Coefficients:
##            Estimate      Std. Error    t value        Pr(>|t|)
## Theta[1]    1.1607e+01    5.6793e+00    2.0437e+00    4.0980e-02
## Theta[2]   -7.4920e+00    3.0683e+00   -2.4417e+00    1.4618e-02
## Theta[3]   -1.8632e-01    2.0063e-01   -9.2865e-01    3.5307e-01
## Theta[4]   -3.2721e+00    5.4418e-01   -6.0129e+00    1.8218e-09
## Theta[5]   -2.0915e-01    4.6169e-02   -4.5301e+00    5.8952e-06
##
## J-Test: degrees of freedom is 1
##                   J-test    P-value
## Test E(g)=0:      0.28316   0.59464
##
## Initial values of the coefficients
##      Theta[1]      Theta[2]      Theta[3]      Theta[4]      Theta[5]
## -2.455663276   0.001196173  -0.681169864  -2.599205971  -0.257323604
```

```
##
## #############
## Information related to the numerical optimization
## Convergence code =  10
## Function eval. =  1892
## Gradian eval. =  NA

## Test overidentifying restrictions
model_1c %>%
  specTest()

##
##  ##  J-Test: degrees of freedom is 1  ##
##
##                 J-test    P-value
## Test E(g)=0:    0.28316  0.59464
```

When using these instruments, the parameters all have the same signs as in (b). The magnitudes are different and the parameter on the cost of driving is no longer significant, but the general interpretation of this model is roughly consistent with the model in (b). The overidentifying test fails to reject the null hypothesis, so we conclude that all empirical moments are sufficiently close to zero, implying that this model provides a good fit for our data.

Problem 2: Generalized Extreme Value Models

Use the phones.csv dataset for this question.

a. We are interested in understanding how consumers value two phone characteristics: internal storage and screen size. Model the purchase of a phone as a multinomial logit model. Include the amount of storage, the screen size, and the price of each phone as explanatory variables with common coefficients; do not include intercepts. That is, the representative utility for alternative j is simply

$$V_{nj} = \beta_1 GB_j + \beta_2 SS_j + \beta_3 p_j$$

where GB_j is the internal storage of alternative j, SS_j is the diagonal screen size of alternative j, and p_j is the price of alternative j. Estimate this multinomial logit model using the mlogit() function in R.

  i. Report your parameter estimates, standard errors, z-stats, and p-values. Briefly interpret these results.

  ii. Calculate the value consumers place on each gigabyte of internal storage and each 0.1 inch of diagonal screen size.

  iii. Apple sees that the iPhone 11 with 256 GB of internal storage has the greatest market share, so they are considering raising the price. Calculate the elasticity of each other phone with respect to the price of the iPhone 11 with 256 GB. (That is, calculate 10 different elasticities, although some might be equal.) Describe the substitution patterns Apple should expect based on these elasticities.

8

```
### Part a
## Load dataset
phones <- read_csv('phones.csv')


## Parsed with column specification:
## cols(
##   customer_id = col_double(),
##   phone_id = col_double(),
##   purchase = col_double(),
##   phone = col_character(),
##   storage = col_double(),
##   screen = col_double(),
##   price = col_double()
## )


## Convert dataset to mlogit format
data_2 <- phones %>%
  mlogit.data(shape = 'long', choice = 'purchase', alt.var = 'phone_id')


## Warning:   Setting row names on a tibble is deprecated.


## Model phone purchase as a multinomial logit
model_2a <- data_2 %>%
  mlogit(purchase ~ storage + screen + price | 0 | 0, data = .)
## Summarize model results
model_2a %>%
  summary()


##
## Call:
## mlogit(formula = purchase ~ storage + screen + price | 0 | 0,
##     data = ., method = "nr")
##
## Frequencies of alternatives:
##     1     2     3     4     5     6     7     8     9    10
## 0.073 0.008 0.078 0.005 0.208 0.355 0.009 0.081 0.014 0.169
##
## nr method
## 5 iterations, 0h:0m:0s
## g'(-H)^-1g = 9.32E-06
## successive function values within tolerance limits
##
## Coefficients :
##             Estimate  Std. Error z-value  Pr(>|z|)
## storage  0.01014179  0.00038324  26.463 < 2.2e-16 ***
## screen   1.88293072  0.15827858  11.896 < 2.2e-16 ***
## price   -0.00728440  0.00026080 -27.930 < 2.2e-16 ***
```

9

```
## ---
## Signif. codes:   0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -1832.7


## Calculate estimated probabilities for each phone
probability_2a <- fitted(model_2a, type = 'probabilities')[1, ]
## Calculate value of storage and screen size
model_2a$coefficients[1:2] / -model_2a$coefficients[3] * c(1, 0.1)


##    storage      screen
##   1.392263 25.848822


## Calculate elasticities with respect to the price of alternative 6
model_2a$coefficients[3] * phones$price[6] *
  c(1 - probability_2a[6], -probability_2a[6]) %>%
  setNames(c('own', 'cross'))


##        own      cross
## -3.735641   2.448812
```

The parameter estimates on both storage and screen size are positive, and the parameter estimate on price is negative; all estimates are statistically significant. These result indicate that consumers get more utility from a phone with more storage and a larger screen size but lose utility as the price of a phone increases, all of which are intuitive. Using these parameter estimates, we see that each gigabyte of storage is worth $1.39 and each 0.1 inch of screen size is worth $25.85. The own-price elasticity of the iPhone 11 with 256 GB is -3.74, and the cross-price elasticity of all other phones with respect to the price of the iPhone 11 with 256 GB is 2.45. If Apple raises the price of the iPhone 11 with 256 GB, this model implies that consumers will substitute proportionally to all other phones in the choice set.

b. The multinomial logit model of (a) is not be the best model for this setting if we think a consumer's unobserved utility includes brand preference, which would create correlations among phones of the same brand. Model the purchase of a phone as a nested logit model with two nests, one for each brand. Model the purchase of a phone as in (a) with the addition of these nests. That is, the representative utility for alternative j is simply

$$V_{nj} = \beta_1 GB_j + \beta_2 SS_j + \beta_3 p_j$$

where GB_j is the internal storage of alternative j, SS_j is the diagonal screen size of alternative j, and p_j is the price of alternative j. Estimate this nested logit model using the `mlogit()` function in R.

   i. Report your parameter estimates, standard errors, z-stats, and p-values. Briefly interpret these results.

   ii. Calculate the value consumers place on each gigabyte of internal storage and each 0.1 inch of diagonal screen size.

10

Apple is still interested in raising the price of the iPhone 11 with 256 GB. Calculate the elasticity of each other phone with respect to the price of the iPhone 11 with 256 GB. (That is, calculate 10 different elasticities, although some might be equal.) Describe the substitution patterns Apple should expect based on these elasticities. Why are these substitution patterns different from what you found in (a)?

```r
### Part b
## Model phone purchase as a nested logit with brand nests
model_2b <- data_2 %>%
  mlogit(purchase ~ storage + screen + price | 0 | 0, data = .,
         nests = list(google = 1:4, apple = 5:10))
## Summarize model results
model_2b %>%
  summary()
```

```
##
## Call:
## mlogit(formula = purchase ~ storage + screen + price | 0 | 0,
##     data = ., nests = list(google = 1:4, apple = 5:10))
##
## Frequencies of alternatives:
##     1     2     3     4     5     6     7     8     9    10
## 0.073 0.008 0.078 0.005 0.208 0.355 0.009 0.081 0.014 0.169
##
## bfgs method
## 11 iterations, 0h:0m:0s
## g'(-H)^-1g =  21.8
## last step couldn't find higher value
##
## Coefficients :
##              Estimate  Std. Error  z-value  Pr(>|z|)
## storage    0.00620166  0.00046368  13.3749 < 2.2e-16 ***
## screen     1.02238660  0.15984596   6.3961 1.594e-10 ***
## price     -0.00461945  0.00037303 -12.3836 < 2.2e-16 ***
## iv:google  0.17941856  0.08697237   2.0629   0.03912 *
## iv:apple   0.67752903  0.05502902  12.3122 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -1829
```

```r
## Calculate estimated probabilities for each phone
probability_2b <- fitted(model_2b, type = 'probabilities')[1, ]
## Calculate value of storage and screen size
model_2b$coefficients[1:2] / -model_2b$coefficients[3] * c(1, 0.1)
```

```
##  storage   screen
##  1.34251 22.13221
```

```
## Calculate elasticities with respect to the price of alternative 6
c(model_2b$coefficients[3] * phones$price[6] *
    ((1 / model_2b$coefficients[5]) -
      ((1 - model_2b$coefficients[5]) / model_2b$coefficients[5] *
        (probability_2b[6] / sum(probability_2b[5:10]))) -
      probability_2b[6]),
  -model_2b$coefficients[3] * phones$price[6] * probability_2b[6] *
    (1 + (1 - model_2b$coefficients[5]) / model_2b$coefficients[5] *
      1 / sum(probability_2b[5:10])),
  -model_2b$coefficients[3] * phones$price[6] * probability_2b[6]) %>%
  setNames(c('own', 'within nest', 'other nest'))


##         own within nest   other nest
##   -3.371083    2.417472     1.534494
```

The coefficient estimates are roughly comparable to those in (a). We also now estimate independence (or correlation) parameters for each brand and find both parameters are between zero and one and are statistically significant. These parameters indicate that the unobserved component of utility that a consumer receives from all phones of the same brand is correlated, suggesting unmodeled brand preferences among consumers. Using the coefficient estimates, we see that each gigabyte of storage is worth \$1.34 and each 0.1 inch of screen size is worth \$22.13. The own-price elasticity of the iPhone 11 with 256 GB is -3.37, the cross-price elasticity of all other Apple phones with respect to the price of the iPhone 11 with 256 GB is 2.42, and the cross-price elasticity of all Google phones with respect to the price of the iPhone 11 with 256 GB is 1.53. If Apple raises the price of the iPhone 11 with 256 GB, this model implies that consumers will substitute proportionally more to other Apple phones than to the Google phones.

c. Conduct a likelihood ratio test to compare the models in (a) and (b). Write down the null hypothesis that you are testing and describe this hypothesis in words. Conduct this likelihood ratio test using the function `lrtest()` in R. Do you reject your null hypothesis? What is the p-value of the test?

```
### Part c
## Conduct likelihood ratio test of the models in parts a and b
lrtest(model_2a, model_2b)


## Likelihood ratio test
##
## Model 1: purchase ~ storage + screen + price | 0 | 0
## Model 2: purchase ~ storage + screen + price | 0 | 0
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   3 -1832.8
## 2   5 -1829.0  2 7.5081    0.02342 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The null hypothesis we are testing is

$$H_0: \lambda_{Apple} = \lambda_{Google} = 1$$

which means there is no correlation in the unobserved components of utility. We reject this null hypothesis; the likelihood ratio test has a p-value of 0.023. Thus, we conclude there is correlation in unobserved utility within a brand, so the model in (b) provides a better fit than the model in (a).

d. Consumer preferences for a specific phone model may be stronger than brand preferences. We can nest by each phone model (i.e., Google Pixel 4, Google Pixel 4 XL, iPhone 11, etc.) to incorporate these phone model preferences into our nested logit model. Model the purchase of a phone as a nested logit model with five nests, one for each phone model. Model the purchase of a phone as in (a) with the addition of these nests. That is, the representative utility for alternative j is simply

$$V_{nj} = \beta_1 GB_j + \beta_2 SS_j + \beta_3 p_j$$

where GB_j is the internal storage of alternative j, SS_j is the diagonal screen size of alternative j, and p_j is the price of alternative j. Estimate this nested logit model using the `mlogit()` function in R.

  i. Report your parameter estimates, standard errors, z-stats, and p-values. Briefly interpret these results.

  ii. Calculate the value consumers place on each gigabyte of internal storage and each 0.1 inch of diagonal screen size.

  iii. Apple is still interested in raising the price of the iPhone 11 with 256 GB. Calculate the elasticity of each other phone with respect to the price of the iPhone 11 with 256 GB. (That is, calculate 10 different elasticities, although some might be equal.) Describe the substitution patterns Apple should expect based on these elasticities. Why are these substitution patterns different from what you found in (a) and (b)?

```
### Part d
## Model phone purchase as a nested logit with model nests
model_2d <- data_2 %>%
  mlogit(purchase ~ storage + screen + price | 0 | 0, data = .,
         nests = list(model_1 = 1:2,
                      model_2 = 3:4,
                      model_3 = 5:6,
                      model_4 = 7:8,
                      model_5 = 9:10))
## Summarize model results
model_2d %>%
  summary()

##
## Call:
## mlogit(formula = purchase ~ storage + screen + price | 0 | 0,
##     data = ., nests = list(model_1 = 1:2, model_2 = 3:4, model_3 = 5:6,
##         model_4 = 7:8, model_5 = 9:10))
##
## Frequencies of alternatives:
##      1     2     3     4     5     6     7     8     9    10
## 0.073 0.008 0.078 0.005 0.208 0.355 0.009 0.081 0.014 0.169
```

```
##
## bfgs method
## 15 iterations, 0h:0m:0s
## g'(-H)^-1g = 3.68E-07
## gradient close to zero
##
## Coefficients :
##              Estimate  Std. Error  z-value  Pr(>|z|)
## storage     0.00695180  0.00050983  13.6355 < 2.2e-16 ***
## screen      1.50203616  0.17404636   8.6301 < 2.2e-16 ***
## price      -0.00526048  0.00042449 -12.3924 < 2.2e-16 ***
## iv:model_1  0.03687200  0.01091284   3.3788 0.0007281 ***
## iv:model_2  0.02945388  0.00867669   3.3946 0.0006873 ***
## iv:model_3  1.00915663  0.13884278   7.2683 3.639e-13 ***
## iv:model_4  0.55181846  0.09364091   5.8929 3.794e-09 ***
## iv:model_5  0.52297779  0.07551492   6.9255 4.345e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -1759.3


## Calculate estimated probabilities for each phone
probability_2d <- fitted(model_2d, type = 'probabilities')[1, ]
## Calculate value of storage and screen size
model_2d$coefficients[1:2] / -model_2d$coefficients[3] * c(1, 0.1)


##   storage    screen
##  1.321514 28.553201


## Calculate elasticities with respect to the price of alternative 6
c(model_2d$coefficients[3] * phones$price[6] *
    ((1 / model_2d$coefficients[6]) -
       ((1 - model_2d$coefficients[6]) / model_2d$coefficients[6] *
          (probability_2d[6] / sum(probability_2d[5:6]))) -
       probability_2d[6]),
  -model_2d$coefficients[3] * phones$price[6] * probability_2d[6] *
    (1 + (1 - model_2d$coefficients[6]) / model_2d$coefficients[6] *
       1 / sum(probability_2d[5:6])),
  -model_2d$coefficients[3] * phones$price[6] * probability_2d[6]) %>%
  setNames(c('own', 'within nest', 'other nest'))


##       own within nest  other nest
##  -2.860307    1.565319    1.590929
```

The coefficient estimates are roughly comparable to those in (a) and (b). The independence (or correlation) parameter for the iPhone 11 is not statistically different from one, indicating that there is no correlation of unobserved utility between the two phones of that model. The independence

(or correlation) parameters for the other models are all between zero and one and are statistically significant, indicating that the unobserved component of utility that a consumer receives from phones of the same model is correlated for all models other than the iPhone 11. Using the coefficient estimates, we see that each gigabyte of storage is worth $1.32 and each 0.1 inch of screen size is worth $28.55. The own-price elasticity of the iPhone 11 with 256 GB is -2.86, the cross-price elasticity of the other iPhone 11 with respect to the price of the iPhone 11 with 256 GB is 1.56, and the cross-price elasticity of all other models with respect to the price of the iPhone 11 with 256 GB is 1.59. If Apple raises the price of the iPhone 11 with 256 GB, this model implies that consumers will substitute roughly proportionally to all other phones in the choice set.

e. Suppose you want to compare the models in (b) and (d). Can you use a likelihood ratio test to compare these models? If so, write down the null hypothesis of your test, conduct the test using the function `lrtest()` in R, and describe your conclusion. If not, explain specifically why the likelihood ratio test is not appropriate for this model comparison.

The likelihood ratio test is used to test a restricted model against an unrestricted model. In this case, however, neither model imposes a restriction on the other. Instead, when moving from brand nests to model nests, some parameters are restricted, but other parameter restrictions are relaxed; for example, we impose a restriction on the correlation between different models by the same brand, but we also allow for greater flexibility by estimating a separate parameter for each model. Thus, the likelihood ratio test cannot be used to compare these models.