HW 4

David Gao

2022-10-25

```
library(tidyverse)
```

```
## Registered S3 methods overwritten by 'tibble':
##   method     from  
##   format.tbl pillar
##   print.tbl  pillar
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.1     v dplyr   1.0.7
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
## Warning: package 'stringr' was built under R version 4.0.5
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(dplyr)
library(ggplot2)

oj<- read.csv("C:\\UW\\AUT 2022\\ECON 487\\HW 2\\oj.csv")
```

1. In the last assignment you calculated the MSE on a test set. Let’s expand that code to include 5-fold cross validation.

a. Create 5 partitions of the data of equal size.

```
n <- nrow(oj)
oj_temp <- oj %>%
  mutate(binom = rbinom(nrow(oj), 1, 0.4))

oj1 <- oj_temp %>%
  filter(binom == 0)

oj1 <- oj1 %>%
  mutate(binom = rbinom(nrow(oj1), 1, 0.333333))

oj11 <- oj1 %>%
  filter(binom == 0)

oj20 <- oj1 %>%
  filter(binom == 1)

oj111 <- oj11 %>%
  mutate(binom = rbinom(nrow(oj11), 1, 0.5))

oj40 <- oj111 %>%
  filter(binom == 0)

oj60 <- oj111 %>%
  filter(binom == 1)

oj2 <- oj_temp %>%
  filter(binom == 1)

oj2 <- oj2 %>%
  mutate(binom = rbinom(nrow(oj2), 1, 0.5))

oj80 <- oj2 %>%
  filter(binom == 0)

oj100 <- oj2 %>%
  filter(binom == 1)
```

b. Create 5 training datasets using 80% of the data for each one.

```r
# bind all fold
oj_new <- rbind(oj20, oj40, oj60, oj80, oj100)

# create lagged price (code copied from solution for HW3)
oj_new <- oj_new %>%
  arrange(week) %>%
  group_by(store, brand) %>%
  mutate(lag_price = ifelse(lag(week) + 1 == week, lag(price), NA)) %>%
  ungroup()

# replace all NA value in lagged price with mean price
oj_new[is.na(oj_new)] <- mean(oj$price)

MSE1 = c(0, 0, 0, 0, 0)
# number of rows for each fold
rows <- c(0, 5795, 5816, 5772, 5820, 5744)

for (i in 1:5) {
  start <- sum(rows[1:i])
  ends <- start + rows[i+1]
  oj_temp <- oj_new[start:ends,]

  # sampling training and testing data using binomial variable with P(x = 1) = 0.8
  oj_temp <- oj_temp %>%
  mutate(binom = rbinom(nrow(oj_temp), 1, 0.8))

  oj_train <- oj_temp %>%
    filter(binom == 1)

  oj_test <- oj_temp %>%
    filter(binom == 0)

  # fit model
  model <- lm(logmove ~ log(price) + feat + brand + brand*log(price) + log(lag_price) + AGE60 +
 EDUC + ETHNIC + INCOME + HHLARGE + WORKWOM + HVAL150 + SSTRDIST + SSTRVOL + CPDIST5 + CPWVOL5 +
EDUC*log(price) + HHLARGE*log(price), data = oj_train)

  # predict value
  logmove_hat <- predict(model, newdata = oj_test)

  # MSE for each fold
  MSE = sum((oj_test$logmove - logmove_hat)^2)/n

  # store MSE for each fold
  MSE1[i] <- MSE
}
avg_MSE <- mean(MSE1)

MSE1
```

```
## [1] 0.01440005 0.01440081 0.01470813 0.01449250 0.01601802
```

```
avg_MSE
```

```
## [1] 0.0148039
```

c. Estimate a complex model using OLS which includes price, featured, brand, brand*price and lagged price, all the sociodemographic variables and interactions of EDUC and HHSIZE with price on each of the training sets then the MSE on the test sets using the predict command.

i. Calculate the MSE for the model on the test set for each fold (e.g., there will be five sets of model parameters and five test set MSEs with 5-fold cross validation).

The MSE for each fold is 0.0144, 0.0144008, 0.0147081, 0.0144925, and 0.016018.

ii. Average across the MSEs to get the cross validated MSE for an OLS model run on that particular set of features.

The average MSE is 0.0148039.

2. Now lets take that same model from (1.c) and run a LASSO using glmnet which is a workhorse R package for LASSO, Ridge and Elastic Nets.

a. First remember to install the glmnet package and library to your R session.

```
library("glmnet")
```

```
## Warning: package 'glmnet' was built under R version 4.0.5
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
## Loaded glmnet 4.1-4
```

b. Remember to estimate a LASSO you must pass glmnet a matrix of data for candidate features and a vector as candidate outcomes:

```
oj_RHS <- oj_new %>%
  select(brand, store, week, feat, price, AGE60, EDUC, ETHNIC, INCOME, HHLARGE, WORKWOM, HVAL15
0, SSTRDIST, SSTRVOL, CPDIST5, CPWVOL5, lag_price)

oj_LHS <- oj_new %>%
  select(logmove)
```

c. Which are the parameters the cross validated LASSO model kicks out of the model? What is the ratio of number of features to number of observations? How might that relate to overfitting from “sampling error”?

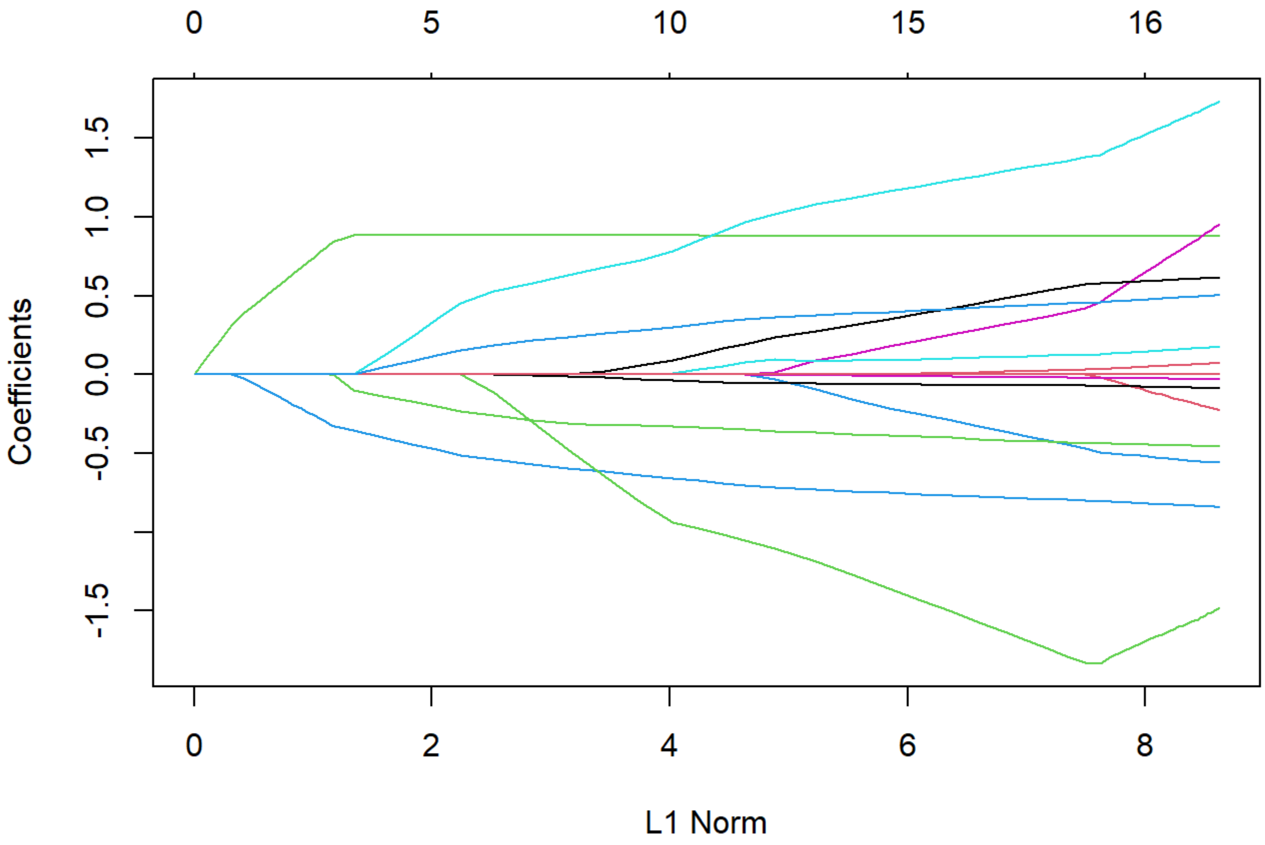
```
x <- as.matrix(oj_RHS)
y <- as.numeric(as.matrix(oj_LHS))

#x <- as.matrix(oj_new[ ,5:17])
#y <- as.numeric(as.matrix(oj_new[ ,4]))

lasso_v1 <- glmnet(x, y, alpha=1)
```

```
## Warning in storage.mode(xd) <- "double": NAs introduced by coercion
```

```
#Results
plot(lasso_v1)
```



```
coef(lasso_v1, s=lasso_v1$lambda.min)
```

```
## 18 x 72 sparse Matrix of class "dgCMatrix"
```

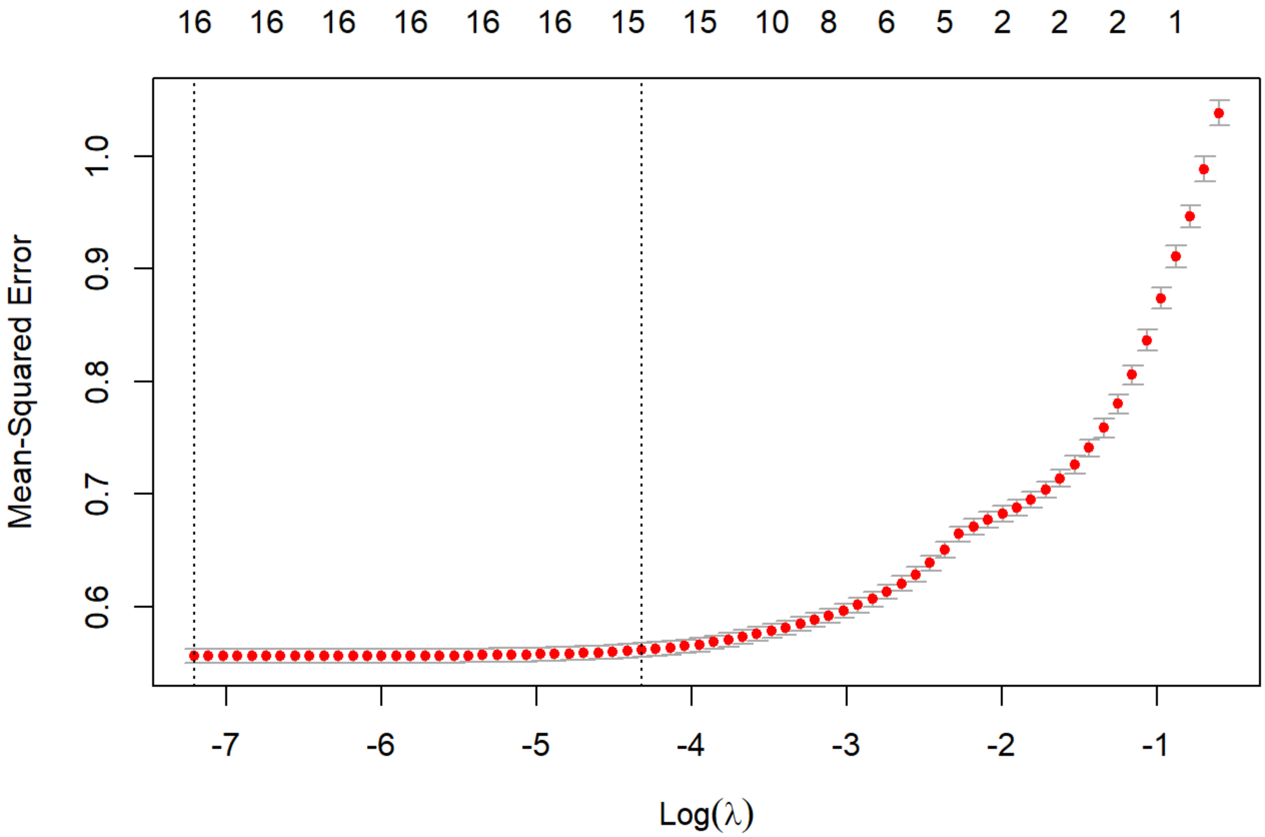
```
##    [[ suppressing 72 column names 's0', 's1', 's2' ... ]]
```

```
##
## (Intercept) 9.167864 9.1407774 9.116097 9.0936094  9.12952205  9.20635703
## brand       .        .         .        .          .           .
## store       .        .         .        .          .           .
## week        .        .         .        .          .           .
## feat        .        0.1141635 0.218185 0.3129655  0.38883467  0.44975799
## price       .        .         .        .         -0.02362047 -0.06361618
## AGE60       .        .         .        .          .           .
## EDUC        .        .         .        .          .           .
## ETHNIC      .        .         .        .          .           .
## INCOME      .        .         .        .          .           .
## HHLARGE     .        .         .        .          .           .
## WORKWOM     .        .         .        .          .           .
## HVAL150     .        .         .        .          .           .
## SSTRDIST    .        .         .        .          .           .
## SSTRVOL     .        .         .        .          .           .
## CPDIST5     .        .         .        .          .           .
## CPWVOL5     .        .         .        .          .           .
## lag_price   .        .         .        .          .           .
##
## (Intercept)  9.2763665  9.3401565  9.3982796  9.4512392  9.4994940  9.5434620
## brand        .          .          .          .          .          .
## store        .          .          .          .          .          .
## week         .          .          .          .          .          .
## feat         0.5052686  0.5558478  0.6019338  0.6439255  0.6821868  0.7170491
## price       -0.1000589 -0.1332641 -0.1635194 -0.1910870 -0.2162055 -0.2390926
## AGE60        .          .          .          .          .          .
## EDUC         .          .          .          .          .          .
## ETHNIC       .          .          .          .          .          .
## INCOME       .          .          .          .          .          .
## HHLARGE      .          .          .          .          .          .
## WORKWOM      .          .          .          .          .          .
## HVAL150      .          .          .          .          .          .
## SSTRDIST     .          .          .          .          .          .
## SSTRVOL      .          .          .          .          .          .
## CPDIST5      .          .          .          .          .          .
## CPWVOL5      .          .          .          .          .          .
## lag_price    .          .          .          .          .          .
##
## (Intercept)  9.5835205  9.6200238  9.6532842  9.6835899  9.7116301910
## brand        .          .          .          .          .
## store        .          .          .          .          .
## week         .          .          .          .          .
## feat         0.7488196  0.7777624  0.8041340  0.8281629  0.8500570616
## price       -0.2599454 -0.2789468 -0.2962601 -0.3120353 -0.3264091103
## AGE60        .          .          .          .          .
## EDUC         .          .          .          .          .
## ETHNIC       .          .          .          .          .
## INCOME       .          .          .          .          .
## HHLARGE      .          .          .          .          .
## WORKWOM      .          .          .          .          .
## HVAL150      .          .          .          .          .
## SSTRDIST     .          .          .          .          .
## SSTRVOL      .          .          .          .          .
## CPDIST5      .          .          .          .          .
## CPWVOL5      .          .          .          .         -0.0009725855
## lag_price    .          .          .          .          .
##
## (Intercept)  9.75944331  9.800299232  9.79743239  9.7916234  9.7864355
## brand        .           .            .           .          .
## store        .           .            .           .          .
## week         .           .            .           .          .
## feat         0.86989873  0.886494492  0.88612145  0.8859528  0.8857010
## price       -0.33972311 -0.354997450 -0.39917577 -0.4392481 -0.4758782
## AGE60        .           .            0.11674959  0.2404434  0.3531310
## EDUC         .           .            .           .          .
## ETHNIC       .           .            .           .          .
## INCOME       .           .            .           .          .
## HHLARGE      .           .            .           .          .
## WORKWOM      .           .            .           .          .
## HVAL150      .           .            .           .          .
## SSTRDIST     .           .            .           .          .
## SSTRVOL      .           .            .           .          .
## CPDIST5      .           .            .           .          .
## CPWVOL5     -0.05139639 -0.097301787 -0.13587851 -0.1706008 -0.2022379
## lag_price    .           0.004479254  0.04853761  0.0884877  0.1249719
##
## (Intercept)  9.7817091  9.7952058  9.825884528  9.854187889  9.87822186
## brand        .          .          .            .            .
## store        .          .          .            .            .
## week         .          .          .            .            .
## feat         0.8854711  0.8850665  0.884789803  0.884831502  0.88431657
## price       -0.5092549 -0.5399329 -0.567763486 -0.592776381 -0.61625894
## AGE60        0.4558077  0.5312661  0.582946836  0.631593089  0.68073126
## EDUC         .          .          .            .            .
## ETHNIC       .          .          .            .            0.02200277
## INCOME       .          .          .            .            .
## HHLARGE      .         -0.1089823 -0.303073198 -0.471771880 -0.64322154
## WORKWOM      .          .          .            .            .
## HVAL150      .          .          .            .            .
## SSTRDIST     .          .          .            .            .
## SSTRVOL      .          .         -0.003843246 -0.009940525 -0.01841941
## CPDIST5      .          .          .            .            .
## CPWVOL5     -0.2310644 -0.2608058 -0.287680635 -0.309688104 -0.32042363
## lag_price    0.1582154  0.1885531  0.216096769  0.241029341  0.26405362
##
## (Intercept)  9.89744399  9.906433e+00  9.8826856907  9.8621022975  9.8433490720
## brand        .           .             .             .             .
## store        .           .             .             .             .
## week         .           1.515152e-05  0.0001172607  0.0002102957  0.0002950626
## feat         0.88424507  8.837712e-01  0.8834620383  0.8830495881  0.8826722120
## price       -0.63722014 -6.567590e-01 -0.6738939634 -0.6896658674 -0.7040385247
## AGE60        0.73023306  7.818766e-01  0.8519272626  0.9140562588  0.9706654402
## EDUC         .           .             .             .             .
## ETHNIC       0.05925220  9.454537e-02  0.1318315228  0.1656140074  0.1963929640
## INCOME       .           .             .             .             .
## HHLARGE     -0.80929311 -9.354095e-01 -0.9747864998 -1.0144166655 -1.0505246504
## WORKWOM      .           .             .             .             .
## HVAL150      .           7.559847e-03  0.0330039566  0.0558304347  0.0766302195
## SSTRDIST     .           .             .             .             .
## SSTRVOL     -0.02874737 -3.709236e-02 -0.0426751799 -0.0477726595 -0.0524163018
## CPDIST5      .           .             .             .             .
## CPWVOL5     -0.32252314 -3.262653e-01 -0.3324794792 -0.3382332198 -0.3434777368
## lag_price    0.28459565  3.036483e-01  0.3211901979  0.3372935232  0.3519675494
##
## (Intercept)  9.8435511445  9.844869e+00  9.8555125962  9.8584222651
## brand        .             .             .             .
## store        .             9.724568e-05  0.0001938888  0.0002745576
## week         0.0003749228  4.483885e-04  0.0005149638  0.0005765807
## feat         0.8823419061  8.822679e-01  0.8822361483  0.8821546723
## price       -0.7169625297 -7.283266e-01 -0.7386799714 -0.7481175604
## AGE60        1.0206707445  1.084494e+00  1.1239656166  1.1630833645
## EDUC         0.0147040702  8.897228e-02  0.1292842221  0.1815588264
## ETHNIC       0.2386012065  2.787731e-01  0.3150216357  0.3507661382
## INCOME       .             .             .             .
## HHLARGE     -1.1007406593 -1.183184e+00 -1.2668560359 -1.3548540801
## WORKWOM     -0.0311415713 -8.994466e-02 -0.1553135212 -0.2115504062
## HVAL150      0.0957323206  8.798047e-02  0.0924840203  0.0914476945
## SSTRDIST    -0.0019408041 -4.314857e-03 -0.0064719818 -0.0084250766
## SSTRVOL     -0.0539045008 -5.690344e-02 -0.0593421889 -0.0616885110
## CPDIST5      .             .             .             0.0022883007
## CPWVOL5     -0.3581492033 -3.677203e-01 -0.3756515318 -0.3835057104
## lag_price    0.3657186047  3.782576e-01  0.3896386902  0.4001333622
##
## (Intercept)  9.8498894154  9.8417160318  9.8354091272  9.8298317864
## brand        .             .             .             .
## store        0.0003366823  0.0003948374  0.0004479622  0.0004964731
## week         0.0006335718  0.0006854927  0.0007328624  0.0007760457
## feat         0.8822238430  0.8821973602  0.8822179270  0.8822509478
## price       -0.7565266347 -0.7642813316 -0.7712984991 -0.7776767216
## AGE60        1.1988488055  1.2341636919  1.2640322064  1.2907901873
## EDUC         0.2166632000  0.2563751056  0.2900781870  0.3199715514
## ETHNIC       0.3893859608  0.4241035896  0.4556004922  0.4842800604
## INCOME       .             .             .             .
## HHLARGE     -1.4320260380 -1.5044463028 -1.5718995122 -1.6334864996
## WORKWOM     -0.2538964962 -0.2922343773 -0.3285926045 -0.3618921608
## HVAL150      0.1001392307  0.1040858446  0.1087696372  0.1134013743
## SSTRDIST    -0.0102898456 -0.0119651813 -0.0134922196 -0.0148844193
## SSTRVOL     -0.0627368615 -0.0640593628 -0.0651908394 -0.0661958164
## CPDIST5      0.0075380505  0.0122325703  0.0165236528  0.0204371675
## CPWVOL5     -0.3927116958 -0.4006251869 -0.4078891186 -0.4145302247
## lag_price    0.4096355376  0.4183771501  0.4263058136  0.4335182063
##
## (Intercept)  9.8242353567  9.8197154422  9.8150775859  9.8107706442
## brand        .             .             .             .
## store        0.0005405083  0.0005809777  0.0006175138  0.0006507928
## week          0.0008153394  0.0008512458  0.0008838579  0.0009135822
## feat          0.8822456113  0.8822950864  0.8822862831  0.8822778678
## price        -0.7835269405 -0.7887963760 -0.7936583782 -0.7980880968
## AGE60         1.3164279453  1.3384453657  1.3597364800  1.3793897462
## EDUC          0.3491251742  0.3739061418  0.3981499493  0.4208329321
## ETHNIC        0.5104677508  0.5341917845  0.5559422186  0.5757374692
## INCOME        .             .             .             .           
## HHLARGE      -1.6891038141 -1.7405732826 -1.7867484458 -1.8288740588
## WORKWOM      -0.3916610725 -0.4194350263 -0.4441700766 -0.4666288524
## HVAL150       0.1167624594  0.1205774575  0.1233527393  0.1255995275
## SSTRDIST     -0.0161509965 -0.0173054694 -0.0183569045 -0.0193139612
## SSTRVOL      -0.0671754959 -0.0680123656 -0.0688285981 -0.0695904545
## CPDIST5       0.0239929738  0.0272368329  0.0301892853  0.0328748526
## CPWVOL5      -0.4205219276 -0.4260322812 -0.4310009419 -0.4355157915
## lag_price     0.4401191527  0.4460900358  0.4515737022  0.4565712526
##                                                                     
## (Intercept)  9.9573158501 10.1762802131 10.3459423927 10.5045515802
## brand         .             .             .             .           
## store         0.0006885859  0.0007226497  0.0007538007  0.0007819502
## week          0.0009400077  0.0009637189  0.0009855503  0.0010054210
## feat          0.8821570958  0.8821274170  0.8821077605  0.8820851030
## price        -0.8022732977 -0.8060040835 -0.8093924094 -0.8124837757
## AGE60         1.3965957699  1.4312967745  1.4591845159  1.4869110901
## EDUC          0.4615060943  0.5129857358  0.5546214238  0.5948744111
## ETHNIC        0.5835007723  0.5842578654  0.5874509819  0.5900303689
## INCOME       -0.0149172685 -0.0383253552 -0.0564482029 -0.0734960686
## HHLARGE      -1.8349436609 -1.7883023040 -1.7604122480 -1.7313833472
## WORKWOM      -0.4900335525 -0.4963663697 -0.5037645622 -0.5089030804
## HVAL150       0.1286255664  0.1340783222  0.1384995028  0.1420230511
## SSTRDIST     -0.0201944495 -0.0210903075 -0.0218916481 -0.0226274638
## SSTRVOL      -0.0714946615 -0.0728855391 -0.0741915815 -0.0753833650
## CPDIST5       0.0362729930  0.0400098994  0.0432257855  0.0461826421
## CPWVOL5      -0.4367489912 -0.4385456177 -0.4401607464 -0.4416967835
## lag_price     0.4611919639  0.4652850117  0.4690248216  0.4724339052
##                                                                     
## (Intercept) 10.6457674198 10.7776862188 10.8943271857 10.9995728265
## brand         .             .             .             .           
## store         0.0008077977  0.0008311499  0.0008526263  0.0008723017
## week          0.0010235502  0.0010400466  0.0010551092  0.0010688458
## feat          0.8820692858  0.8820501006  0.8820398919  0.8820351952
## price        -0.8152957618 -0.8178625372 -0.8201936496 -0.8223128480
## AGE60         1.5102600170  1.5334421664  1.5527779998  1.5696138150
## EDUC          0.6298859339  0.6635021677  0.6924668610  0.7181018913
## ETHNIC        0.5926275989  0.5947464933  0.5969474948  0.5990271704
## INCOME       -0.0885851228 -0.1027711504 -0.1152362175 -0.1264460138
## HHLARGE      -1.7080738542 -1.6837591321 -1.6646681363 -1.6483799824
## WORKWOM      -0.5149886249 -0.5191522528 -0.5242059714 -0.5293615670
## HVAL150       0.1455242275  0.1483923852  0.1512875756  0.1541089786
## SSTRDIST     -0.0232923289 -0.0239034729 -0.0244552368 -0.0249560239
## SSTRVOL      -0.0764773784 -0.0774684845 -0.0783763261 -0.0792022990
## CPDIST5       0.0488520672  0.0513081175  0.0535202812  0.0555281987
## CPWVOL5      -0.4430328212 -0.4443111623 -0.4454230186 -0.4464137343
## lag_price    0.4755386043  0.4783691096  0.4809447041  0.4832884721
##                                                                    
## (Intercept) 11.0996491334 11.1864927627 11.2695493927 11.3405654598
## brand        .             .             .             .           
## store        0.0008899835  0.0009063147  0.0009209729  0.0009345984
## week         0.0010813159  0.0010927448  0.0011030958  0.0011126161
## feat         0.8820189459  0.8820222342  0.8820073680  0.8820207113
## price       -0.8242568092 -0.8260073916 -0.8276227759 -0.8290635276
## AGE60        1.5869345476  1.6013735856  1.6158492963  1.6277795297
## EDUC         0.7433603242  0.7648883370  0.7859652874  0.8036432579
## ETHNIC       0.6006103272  0.6023708851  0.6036896455  0.6052130290
## INCOME      -0.1371931480 -0.1464756975 -0.1554002157 -0.1629972843
## HHLARGE     -1.6300739674 -1.6162602699 -1.6009993688 -1.5899492219
## WORKWOM     -0.5326916167 -0.5365432850 -0.5392435180 -0.5424004435
## HVAL150      0.1563649214  0.1585860964  0.1604159694  0.1622398983
## SSTRDIST    -0.0254179843 -0.0258347519 -0.0262184217 -0.0265638074
## SSTRVOL     -0.0799512185 -0.0806337210 -0.0812569404 -0.0818215457
## CPDIST5      0.0573880570  0.0590517336  0.0605957204  0.0619683539
## CPWVOL5     -0.4473712026 -0.4482193715 -0.4490163075 -0.4497284887
## lag_price    0.4854308460  0.4873711180  0.4891508818  0.4907537855
##                                                                    
## (Intercept) 11.4093582356 11.466907022 11.523608174 11.5693492808 11.6231443254
## brand        .             .            .            .             .           
## store        0.0009467609  0.000958147  0.000968233  0.0009776951  0.0009857453
## week         0.0011212055  0.001129151  0.001136279  0.0011429294  0.0011486872
## feat         0.8820064230  0.882033219  0.882019071  0.8820647608  0.8819937944
## price       -0.8304069879 -0.831584281 -0.832702473 -0.8336513469 -0.8346500897
## AGE60        1.6396789429  1.649488982  1.659184677  1.6672432223  1.6766248754
## EDUC         0.8210522168  0.835436833  0.849712001  0.8612771107  0.8750683059
## ETHNIC       0.6063261410  0.607659337  0.608627527  0.6098153507  0.6102500347
## INCOME      -0.1703828800 -0.176547016 -0.182626549 -0.1875407785 -0.1933258742
## HHLARGE     -1.5774613895 -1.568815219 -1.558772894 -1.5522479153 -1.5404509289
## WORKWOM     -0.5447376426 -0.547308759 -0.549378440 -0.5513884430 -0.5525744427
## HVAL150      0.1637771725  0.165280780  0.166591135  0.1678143344  0.1687651391
## SSTRDIST    -0.0268820719 -0.027167761 -0.027431721 -0.0276672072 -0.0278928290
## SSTRVOL     -0.0823397815 -0.082803861 -0.083233906 -0.0836077917 -0.0839816547
## CPDIST5      0.0632493919  0.064377283  0.065438957  0.0663582452  0.0672934076
## CPWVOL5     -0.4503854794 -0.450991159 -0.451532673 -0.4520718858 -0.4524937508
## lag_price    0.4922330364  0.493551682  0.494781855  0.4958580552  0.4969240411
##                                                                               
## (Intercept) 11.6631888469 11.702442220 11.739463243 11.766448145 11.794380777
## brand        .             .            .            .            .          
## store        0.0009935202  0.001000389  0.001006625  0.001012330  0.001017466
## week         0.0011541952  0.001159109  0.001163571  0.001167808  0.001171541
## feat         0.8820286307  0.882017528  0.882003233  0.882059099  0.882052801
## price       -0.8354409075 -0.836212915 -0.836921038 -0.837483459 -0.838065486
## AGE60        1.6843791369  1.691623865  1.698588556  1.704195854  1.709390834
## EDUC         0.8857008362  0.895994572  0.905815435  0.913242941  0.920593062
## ETHNIC       0.6110705805  0.611737704  0.612245225  0.613010148  0.613671250
## INCOME      -0.1976743331 -0.201915109 -0.205924353 -0.208875137 -0.211892809
## HHLARGE     -1.5332687922 -1.525739607 -1.518106148 -1.513828741 -1.508994301
## WORKWOM     -0.5535063214 -0.554490017 -0.555183067 -0.555802919 -0.556642035
## HVAL150      0.1695821614  0.170349319  0.170984539  0.171575546  0.172196409
## SSTRDIST    -0.0280909976 -0.028274680 -0.028442800 -0.028589987 -0.028729275
## SSTRVOL     -0.0842928562 -0.084586956 -0.084858488 -0.085071122 -0.085282335
## CPDIST5      0.0680704083  0.068805073  0.069482204  0.070044474  0.070591746
## CPWVOL5     -0.4529666865 -0.453365141 -0.453727257 -0.454155920 -0.454482084
## lag_price    0.4978187721  0.498667656  0.499443787  0.500097581  0.500739240
##
## (Intercept) 11.821665117 11.84760856 11.871887894 11.886540216 11.910534025
## brand        .            .           .            .            .
## store        0.001022154  0.00102644  0.001030346  0.001033252  0.001036939
## week         0.001174910  0.00117797  0.001180755  0.001183458  0.001185660
## feat         0.882036716  0.88202038  0.882005339  0.882070745  0.881994019
## price       -0.838607786 -0.83910367 -0.839555510 -0.839867785 -0.840325271
## AGE60        1.714415805  1.71927719  1.723944059  1.727787148  1.731880917
## EDUC         0.927743089  0.93460651  0.941123993  0.945748187  0.951821151
## ETHNIC       0.614144091  0.61448358  0.614737214  0.615207959  0.615451130
## INCOME      -0.214839545 -0.21764815 -0.220284652 -0.221938230 -0.224509507
## HHLARGE     -1.503708465 -1.49825857 -1.492850450 -1.490173373 -1.484711044
## WORKWOM     -0.557318921 -0.55778462 -0.558061926 -0.557837412 -0.558363605
## HVAL150      0.172733701  0.17317564  0.173530327  0.173797461  0.174153146
## SSTRDIST    -0.028856650 -0.02897311 -0.029079640 -0.029171268 -0.029266290
## SSTRVOL     -0.085485000 -0.08567413 -0.085848484 -0.085923375 -0.086118308
## CPDIST5      0.071101902  0.07157275  0.072005358  0.072341895  0.072739705
## CPWVOL5     -0.454755679 -0.45499918 -0.455221960 -0.455673085 -0.455701945
## lag_price    0.501331127  0.50187116  0.502363004  0.502746043  0.503209575
##
## (Intercept) 11.924173591
## brand        .
## store        0.001039380
## week         0.001187888
## feat         0.882038929
## price       -0.840596739
## AGE60        1.735316072
## EDUC         0.956025409
## ETHNIC       0.615756342
## INCOME      -0.226041501
## HHLARGE     -1.481792330
## WORKWOM     -0.558106712
## HVAL150      0.174354544
## SSTRDIST    -0.029343077
## SSTRVOL     -0.086190754
## CPDIST5      0.073029362
## CPWVOL5     -0.456051532
## lag_price    0.503535302
```

```
# Now ready for cross validation version of the object
#x <- as.matrix(oj_new[ ,5:17])
#y <- as.numeric(as.matrix(oj_new[ ,4]))

x <- as.matrix(oj_RHS[,2:17])
y <- as.numeric(as.matrix(oj_LHS))

cvfit <- cv.glmnet(x, y, alpha=1)
#Results
plot(cvfit)
```



```
cvfit$lambda.min
```

```
## [1] 0.0007396846
```

```
log(cvfit$lambda.min)
```

```
## [1] -7.209287
```

```
coef(cvfit, s = "lambda.min")
```

```
## 17 x 1 sparse Matrix of class "dgCMatrix"
##                      s1
## (Intercept) 11.924173543
## store        0.001039380
## week         0.001187888
## feat         0.882038929
## price       -0.840596739
## AGE60        1.735316778
## EDUC         0.956025399
## ETHNIC       0.615756329
## INCOME      -0.226041557
## HHLARGE     -1.481791378
## WORKWOM     -0.558105762
## HVAL150      0.174354565
## SSTRDIST    -0.029343076
## SSTRVOL     -0.086190735
## CPDIST5      0.073029373
## CPWVOL5     -0.456051517
## lag_price    0.503535302
```

Seems like none of the variable is kicked out by the LASSO model from the. And the lasso_v1 part, as both do not have coefficient values in the whole process. The ratio should be 17:28947. This result is obtained with min of lambda, which means overfitting might happen as variables that are unnecessary might be inclued in the model. (brand is removed as I ended up with *error: object of type ‘closure’ is not subsettable* when trying `X <- model.matrix(formula, df_RHS)` and can not find a way to solve it lol)

d. Can you look that the glmnet objects and figure out what the out of sample (e.g., test set) average MSE was with the cross validated LASSO model relative to the model in 1.c?

The MSE of LASSO model should be smaller than OLS model as OLS included more unnecessary variables that might lead to overfitting, making smaller MSE in trainning data but larger MSE in testing and actual data.

e. What is the advantage of using LASSO for choosing model complexity as opposed to using your intuition as an economist?

The advantage of LASSO is that we can get an simpler model with only related variables, whereas unrelated variables will have coefficient 0, which is opposed to adding more variables into the model to make it more accurate.

i. In what part of this process did you use your intuition as an economist? (HINT: what’s in the X matrix?)

We assume all varaibles are relative to log quantity and put all of them into the LASSO model for cross validation.

3. Now estimate the model with only the variable selected with the LASSO procedure but with OLS to avoid attenuation bias in the coefficients.

```
model <- lm(logmove ~ log(price) + feat + brand + brand*log(price) + log(lag_price) + log(lag_pr
ice)*log(price) + AGE60 + EDUC + ETHNIC + INCOME + HHLARGE + WORKWOM + HVAL150 + SSTRDIST + SSTR
VOL + CPDIST5 + CPWVOL5 + EDUC*log(price) + HHLARGE*log(price), data = oj_new)

summary(model)
```

```
## 
## Call:
## lm(formula = logmove ~ log(price) + feat + brand + brand * log(price) + 
##     log(lag_price) + log(lag_price) * log(price) + AGE60 + EDUC + 
##     ETHNIC + INCOME + HHLARGE + WORKWOM + HVAL150 + SSTRDIST + 
##     SSTRVOL + CPDIST5 + CPWVOL5 + EDUC * log(price) + HHLARGE * 
##     log(price), data = oj_new)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -5.5145 -0.3868  0.0008  0.3716  2.9010 
## 
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)    
## (Intercept)               12.768202   0.332073  38.450  < 2e-16 ***
## log(price)                -3.080147   0.082968 -37.125  < 2e-16 ***
## feat                       0.807220   0.010060  80.244  < 2e-16 ***
## brandminute.maid           0.390128   0.037021  10.538  < 2e-16 ***
## brandtropicana             0.370037   0.047876   7.729 1.12e-14 ***
## log(lag_price)             0.866612   0.053190  16.293  < 2e-16 ***
## AGE60                      2.015590   0.125461  16.066  < 2e-16 ***
## EDUC                      -1.903624   0.142650 -13.345  < 2e-16 ***
## ETHNIC                     0.665949   0.036646  18.173  < 2e-16 ***
## INCOME                    -0.260161   0.032791  -7.934 2.20e-15 ***
## HHLARGE                    3.545933   0.430114   8.244  < 2e-16 ***
## WORKWOM                   -0.358290   0.144477  -2.480   0.0131 *  
## HVAL150                    0.365561   0.041153   8.883  < 2e-16 ***
## SSTRDIST                  -0.020209   0.001442 -14.010  < 2e-16 ***
## SSTRVOL                   -0.053340   0.009612  -5.549 2.89e-08 ***
## CPDIST5                    0.064056   0.006189  10.350  < 2e-16 ***
## CPWVOL5                   -0.501326   0.025287 -19.825  < 2e-16 ***
## log(price):brandminute.maid  0.425130   0.050349   8.444  < 2e-16 ***
## log(price):brandtropicana    1.031117   0.056477  18.257  < 2e-16 ***
## log(price):log(lag_price)   -0.469751   0.061263  -7.668 1.80e-14 ***
## log(price):EDUC            3.537077   0.127758  27.686  < 2e-16 ***
## log(price):HHLARGE        -5.612750   0.465137 -12.067  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.6462 on 28925 degrees of freedom
## Multiple R-squared:  0.5985, Adjusted R-squared:  0.5982 
## F-statistic:  2053 on 21 and 28925 DF,  p-value: < 2.2e-16
```

i. For Dominicks when the lagged price is $1 (NOTE: did you interact lagged price with current period price?) If not, does lagged price impact the elasticity this period or log move this period.

The elasticity of Dominicks when lagged price is $1 is -3.080147 + 0.866612*1 + -0.469751 = -2.683286 as Dominicks is taken as the reference in this model. And the lagged price influences the elasticity this preriod.

ii. For Tropicana

The elasticity for Tropicana is -3.080147 + 0.866612 + 1.031117 + -0.469751 = -1.652169.

iii. For Tropicana when its featured

The elasticity for Tropicana is -3.080147 + 0.807220 + 0.866612 + 1.031117 + -0.469751 = -0.844949.

iv. What is the 95% confidence intervals for Tropicana

The 95% confidence interval for elastisity of Tropicana is [-1.652169 - 1.96 · 0.253898, -1.652169 + 1.96 · 0.253898], which is [-2.149809, -1.154529].

b. Which product has the most elastic demand?

Looks like Tropicana has Tropicana has the most elastic demand as it has the highest elasticity.

i. Should that product have the highest markup over costs or lowest markup over costs? Why?

It should have the lowest markup as highest elasticity means small change in price will lead to huge impact on quantity, thus the slope for demand curve is relatively flat, which leads to low markup.

4. Go back to using logmove and log(price).

a. Estimate a 3x3 matrix own price and cross price elasticities for Dominicks, Minute Maid, and Tropicana using only the current week's prices.

```
oj_price <- oj %>%
  select(store, week, brand, price) %>%
  group_by(store, week, brand) %>%
  mutate(brand_temp = brand) %>%
  pivot_wider(id_cols = c(store, week), names_from = brand, values_from = price)

oj_join <- oj %>%
  select(store, week, logmove, brand) %>%
  left_join(oj_price, by = c('store', 'week'))

oj_trop <- oj_join %>%
  filter(brand == "tropicana")

oj_minu <- oj_join %>%
  filter(brand == "minute.maid")

oj_domi <- oj_join %>%
  filter(brand == "dominicks")

model1 <- lm(logmove ~ log(tropicana) + log(minute.maid) + log(dominicks), data = oj_trop)
model2 <- lm(logmove ~ log(tropicana) + log(minute.maid) + log(dominicks), data = oj_minu)
model3 <- lm(logmove ~ log(tropicana) + log(minute.maid) + log(dominicks), data = oj_domi)


tropicana <- summary(model1)$coefficients[, "Estimate"][2:4]
minute.maid <- summary(model2)$coefficients[, "Estimate"][2:4]
dominicks <- summary(model3)$coefficients[, "Estimate"][2:4]

matrix_e <- rbind(tropicana, minute.maid, dominicks)
matrix_e
```

```
##             log(tropicana) log(minute.maid) log(dominicks)
## tropicana      -2.86609240         0.450694      0.2648684
## minute.maid     1.17688039        -3.866215      0.9035890
## dominicks       0.02698408         1.174361     -3.5410460
```

b. Do the same but add in interactions for whether or not each brand is featured.

```
oj_join <- oj %>%
  select(store, week, logmove, brand, feat) %>%
  left_join(oj_price, by = c('store', 'week'))

oj_trop <- oj_join %>%
  filter(brand == "tropicana")

oj_minu <- oj_join %>%
  filter(brand == "minute.maid")

oj_domi <- oj_join %>%
  filter(brand == "dominicks")

model1 <- lm(logmove ~ log(tropicana) + log(minute.maid) + log(dominicks) + feat*log(tropicana)
 + feat*log(minute.maid) + feat*log(dominicks), data = oj_trop)
model2 <- lm(logmove ~ log(tropicana) + log(minute.maid) + log(dominicks) + feat*log(tropicana)
 + feat*log(minute.maid) + feat*log(dominicks), data = oj_minu)
model3 <- lm(logmove ~ log(tropicana) + log(minute.maid) + log(dominicks) + feat*log(tropicana)
 + feat*log(minute.maid) + feat*log(dominicks), data = oj_domi)

summary(model1)
```

```
##
## Call:
## lm(formula = logmove ~ log(tropicana) + log(minute.maid) + log(dominicks) +
##     feat * log(tropicana) + feat * log(minute.maid) + feat *
##     log(dominicks), data = oj_trop)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3191 -0.3801  0.0011  0.3661  2.6433
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)            10.92239    0.04339 251.738  < 2e-16 ***
## log(tropicana)         -2.15199    0.03728 -57.721  < 2e-16 ***
## log(minute.maid)        0.29909    0.03741   7.995 1.44e-15 ***
## log(dominicks)          0.14741    0.02916   5.056 4.36e-07 ***
## feat                    1.50603    0.09893  15.224  < 2e-16 ***
## log(tropicana):feat    -1.76597    0.09519 -18.553  < 2e-16 ***
## log(minute.maid):feat   0.68484    0.11157   6.138 8.67e-10 ***
## log(dominicks):feat     0.13039    0.10214   1.277    0.202
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5946 on 9641 degrees of freedom
## Multiple R-squared:  0.508,  Adjusted R-squared:  0.5077
## F-statistic:  1422 on 7 and 9641 DF,  p-value: < 2.2e-16
```

```
summary(model2)
```

```
## 
## Call:
## lm(formula = logmove ~ log(tropicana) + log(minute.maid) + log(dominicks) + 
##     feat * log(tropicana) + feat * log(minute.maid) + feat * 
##     log(dominicks), data = oj_minu)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -2.6794 -0.3689 -0.0186  0.3440  2.7242 
## 
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)    
## (Intercept)            10.12732    0.04042 250.559   <2e-16 ***
## log(tropicana)          0.32778    0.03661   8.954   <2e-16 ***
## log(minute.maid)       -2.34650    0.04532 -51.778   <2e-16 ***
## log(dominicks)          0.56193    0.03212  17.495   <2e-16 ***
## feat                    1.11161    0.09245  12.024   <2e-16 ***
## log(tropicana):feat     0.77534    0.07599  10.203   <2e-16 ***
## log(minute.maid):feat -1.35931    0.08394 -16.194   <2e-16 ***
## log(dominicks):feat     0.11503    0.06263   1.837   0.0663 .  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.5768 on 9641 degrees of freedom
## Multiple R-squared:  0.6575, Adjusted R-squared:  0.6573 
## F-statistic:  2644 on 7 and 9641 DF,  p-value: < 2.2e-16
```

```
summary(model3)
```

```
##
## Call:
## lm(formula = logmove ~ log(tropicana) + log(minute.maid) + log(dominicks) +
##     feat * log(tropicana) + feat * log(minute.maid) + feat *
##     log(dominicks), data = oj_domi)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8297 -0.5104 -0.0001  0.5053  3.0648
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           10.09993    0.06167 163.768  < 2e-16 ***
## log(tropicana)        -0.26228    0.05161  -5.082 3.81e-07 ***
## log(minute.maid)       0.79993    0.05483  14.589  < 2e-16 ***
## log(dominicks)        -2.86366    0.04786 -59.828  < 2e-16 ***
## feat                  -0.12432    0.11363  -1.094    0.274
## log(tropicana):feat    0.61740    0.09660   6.391 1.72e-10 ***
## log(minute.maid):feat  0.73616    0.12180   6.044 1.56e-09 ***
## log(dominicks):feat   -0.57050    0.08964  -6.365 2.05e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8257 on 9641 degrees of freedom
## Multiple R-squared:  0.5212, Adjusted R-squared:  0.5209
## F-statistic:  1499 on 7 and 9641 DF,  p-value: < 2.2e-16
```

```
tropicana <- summary(model1)$coefficients[, "Estimate"][2:4]
minute.maid <- summary(model2)$coefficients[, "Estimate"][2:4]
dominicks <- summary(model3)$coefficients[, "Estimate"][2:4]

matrix_e <- rbind(tropicana, minute.maid, dominicks)
matrix_e
```

```
##             log(tropicana) log(minute.maid) log(dominicks)
## tropicana       -2.1519941        0.2990872      0.1474123
## minute.maid      0.3277803       -2.3464968      0.5619305
## dominicks       -0.2622846        0.7999289     -2.8636642
```

i. How do the estimates change?

The cross price elasticity for Dominicks and tropicana changes from positive to negative.

ii. What product’s sales suffer the most when Minute Maid is both featured and lowers its price?

Dominicks would suffer the most as the cross elasticity of Dominicks and Minute Maid is the largest, and the coefficient for Minute Maid being featured is also the largest in model3 for quantity of Dominicks.

c. Which two products are the most competitive with each other?

Looks like dominicks and Minute Maid are competitive with each other

i. How did you infer that looking at the cross price elasticity?

The cross elasticity of Minute Maid and Dominicks is positive and has the largest absolute value, which means one percent increase in price of Minute Maid will lead to 0.8 percent increase in quantity of Dominicks, which means they are substitute and compete with each other.

ii. What do you expect that to mean about the correlation of the prices of those two products? Would they be more correlated or less correlated than the price of other pairs of products?

Based on the cross elasticity of them, one percent increase in price of Minute Maid will lead to 0.8 percent increase in quantity of Dominicks, which means they are substitute and compete with each other. And they are more correlated than other pairs as the corss elasticity between Minute Maid and Dominicks has the largest absolute value.