Appendix: Source code

2024-05-21

```r
knitr::opts_chunk$set(echo = TRUE)
rm(list = ls())
gc()
```

```
##          used (Mb) gc trigger (Mb) limit (Mb) max used (Mb)
## Ncells 478484 25.6    1032385 55.2         NA   669422 35.8
## Vcells 899935  6.9    8388608 64.0      36864  1851803 14.2
```

```r
set.seed(1)
options(digits=5)
if (!require("pacman")) install.packages("pacman")
```

```
## Loading required package: pacman
```

```r
pacman::p_load(
  plm,
  ggplot2,
  tidyverse,
  fixest,
  knitr,
  kableExtra,
  tidymodels,
  modelsummary,
  ggplot2,
  skimr
)
```

Data

```r
orig_df <- read_csv("./input/WSDR.csv")
```

```
## Rows: 15600 Columns: 9
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): descrip
## dbl (8): store, upc, week, move, price, profit, custcoun, Brand
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
head(orig_df, 30)
```

```
## # A tibble: 30 x 9
##    store        upc  week  move  price profit descrip              custcoun Brand
##    <dbl>      <dbl> <dbl> <dbl>  <dbl>  <dbl> <chr>                   <dbl> <dbl>
##  1     5 1200000230     1   158 0.0250  20.1  PEPSI COLA N/R          18820 12000
##  2     5 1200000394     1    47 0.0195   4.02 PEPSI COLA N.R. BO~     18820 12000
##  3     5 1200000396     1    43 0.0195   4.02 DIET PEPSI N.R. BO~     18820 12000
##  4     5 1200000492     1    49 0.0250  20.1  CAFFEINE FREE PEPSI     18820 12000
##  5     5 1200000496     1    61 0.0250  20.1  DIET PEPSI CAFFEINE     18820 12000
##  6     5 3828100261     1    66 0.0208  33.6  DOM ORANGE SODA         18820 38281
##  7     5 4900000634     1    28 0.0388  22.9  COCA-COLA CLASSIC ~     18820 49000
##  8     5 4900000639     1   109 0.0250  20.1  COCA-COLA CLASSIC       18820 49000
##  9     5 4900000658     1    12 0.0388  22.9  COKE DIET CANS          18820 49000
## 10     5 5490000060     1   118 0.0176  13.4  DR PEPPER               18820 54900
## # i 20 more rows
```

Answer to Q1

```r
### Q1 Answer ###
stats <- orig_df %>%
  dplyr::select(move,price, profit, custcoun) %>%
  skimr::skim(.) %>%
  skimr::yank("numeric") %>%
  dplyr::select(skim_variable, mean, sd, p0, p100) %>%
  dplyr::mutate_at(vars(mean, sd, p0, p100), ~round(., 3)) %>%
  kable(format = "latex")
cat(stats)
```

```
##
## \begin{tabular}{l|r|r|r|r}
## \hline
## skim\_variable & mean & sd & p0 & p100\\
## \hline
## move & 132.106 & 263.430 & 1.000 & 6121.00\\
## \hline
## price & 0.024 & 0.007 & 0.008 & 0.04\\
## \hline
## profit & 20.357 & 13.068 & 0.000 & 95.65\\
## \hline
## custcoun & 19154.180 & 4857.582 & 8071.000 & 35340.00\\
## \hline
## \end{tabular}
```

Answer to Q2

```r
### Q2 Answer ###

# to factor
```

```r
df <- orig_df |>
  mutate(upc = factor(upc)) |>
  arrange(week, store, upc)

# create product dummy cols
dummy_vars <- model.matrix(~ upc - 1, data = df)
colnames(dummy_vars) <- gsub("upc", "dummy_", colnames(dummy_vars))
df <- cbind(df, dummy_vars)
# delete reference of dummy col
df <- df |>  select(-c(dummy_1200000230))

# compute market share
df <- df |>
  dplyr::group_by(store, week) |>
  mutate(M_t = mean(custcoun),
         tot_quant_t = sum(move),
         s_jt = move / M_t,
         s_0t = (M_t- tot_quant_t) / M_t,
         logit_share = log(s_jt/s_0t)) |>
  ungroup() |>
  select(-c(M_t, tot_quant_t))

# compute iV
df <- df |>
  mutate(whole_p_jt = price * (1 - profit*(0.01)))

# OLS estimation in Berry's logit
model1_OLS <- feols(logit_share ~  price + i(upc),
               df, vcov="hetero"
)

# IV estimation in Berry's logit
model1_IV <- feols(logit_share ~  i(upc) | price ~ whole_p_jt,
               df, vcov="hetero"
)

# First stage
etable(model1_IV, stage = 1, fitstat=~ . + ivfall + ivwaldall.p,
       signif.code=c("***"=0.01,"**"=0.05,"*"=0.10),
       style.tex = style.tex("aer"),  tex = TRUE,
       digits=3, digits.stats=3)
```

```
## \begingroup
## \centering
## \begin{tabular}{lc}
##    \toprule
##                              & price\\
##                              & (1)\\
##    \midrule
##    Constant                  & 0.005$^{***}$\\
##                              & ($8.18\times 10^{-5}$)\\
##    whole\_p\_jt              & 0.879$^{***}$\\
##                              & (0.004)\\
```

```
##    upc $=$ 1200000394      & 0.001$^{***}$\\
##                            & ($7.49\times 10^{-5}$)\\
##    upc $=$ 1200000396      & 0.001$^{***}$\\
##                            & ($7.49\times 10^{-5}$)\\
##    upc $=$ 1200000492      & 0.0006$^{***}$\\
##                            & ($7.55\times 10^{-5}$)\\
##    upc $=$ 1200000496      & 0.0006$^{***}$\\
##                            & ($7.55\times 10^{-5}$)\\
##    upc $=$ 3828100261      & 0.002$^{***}$\\
##                            & ($8.77\times 10^{-5}$)\\
##    upc $=$ 4900000634      & 0.006$^{***}$\\
##                            & ($9.25\times 10^{-5}$)\\
##    upc $=$ 4900000639      & $-6.34\times 10^{-5}$\\
##                            & ($7.58\times 10^{-5}$)\\
##    upc $=$ 4900000658      & 0.006$^{***}$\\
##                            & ($9.25\times 10^{-5}$)\\
##    upc $=$ 5490000060      & 0.0006$^{***}$\\
##                            & ($8.38\times 10^{-5}$)\\
##     \\
##    Observations            & 15,600\\
##    R$^2$                   & 0.899\\
##    Adjusted R$^2$          & 0.898\\
##    F-test (IV only)        & 42,461.2\\
##    Wald (IV only), p-value & $NaN\times 10^{-Inf}$\\
##    \bottomrule
## \end{tabular}
## \par\endgroup
```

```r
# Estimation result
etable(model1_OLS, model1_IV, stage = 2, fitstat=~ . + ivfall + ivwaldall.p,
       signif.code=c("***"=0.01,"**"=0.05,"*"=0.10),
       style.tex = style.tex("aer"),  tex = TRUE,
       digits=3, digits.stats=3)
```

```
## \begingroup
## \centering
## \begin{tabular}{lcc}
##    \toprule
##     & \multicolumn{2}{c}{logit\_share}\\
##                            & (1)              & (2)\\
##    \midrule
##    Constant                & -1.09$^{***}$    & -0.715$^{***}$\\
##                            & (0.035)          & (0.039)\\
##    price                   & -159.7$^{***}$ & -177.8$^{***}$\\
##                            & (1.40)           & (1.63)\\
##    upc $=$ 1200000394      & -1.55$^{***}$    & -1.49$^{***}$\\
##                            & (0.023)          & (0.023)\\
##    upc $=$ 1200000396      & -1.92$^{***}$    & -1.86$^{***}$\\
##                            & (0.023)          & (0.022)\\
##    upc $=$ 1200000492      & -1.23$^{***}$    & -1.22$^{***}$\\
##                            & (0.023)          & (0.022)\\
##    upc $=$ 1200000496      & -1.20$^{***}$    & -1.19$^{***}$\\
##                            & (0.022)          & (0.021)\\
##    upc $=$ 3828100261      & -1.02$^{***}$    & -1.07$^{***}$\\
```

```
##                              & (0.025)         & (0.025)\\
##     upc $=$ 4900000634       & 0.837$^{***}$   & 1.10$^{***}$\\
##                              & (0.033)         & (0.034)\\
##     upc $=$ 4900000639       & -0.300$^{***}$ & -0.300$^{***}$\\
##                              & (0.021)         & (0.021)\\
##     upc $=$ 4900000658       & 0.629$^{***}$   & 0.889$^{***}$\\
##                              & (0.034)         & (0.035)\\
##     upc $=$ 5490000060       & -2.09$^{***}$   & -2.08$^{***}$\\
##                              & (0.022)         & (0.021)\\
##      \\
##     Observations             & 15,600          & 15,600\\
##     R$^2$                    & 0.673           & 0.669\\
##     Adjusted R$^2$           & 0.673           & 0.669\\
##     F-test (IV only)         &                 & 12,273.7\\
##     Wald (IV only), p-value &                 & $NaN\times 10^{-Inf}$\\
##     \bottomrule
## \end{tabular}
## \par\endgroup
```

Answer to Q4

```r
### Q4 Answer ###

# compute own elasticity for each market
own_elas_jt <- df |>
  select(week, store, upc, descrip, price, move, s_jt, Brand) |>
  mutate(own_elas = model1_IV$coefficients["fit_price"] * price * (1 - s_jt))

# take median of own elasticity by upc
own_elas_df <- own_elas_jt |>
  select(upc, descrip, own_elas) |>
  group_by(upc, descrip) |>
  summarize(own_elas = median(own_elas), .groups = 'drop')

# compute cross elasticity for each market
cross_elas_jt <- df |>
  select(week, store, upc, descrip, price, move, s_jt, Brand) |>
  mutate(cross_elas = (-1) * model1_IV$coefficients["fit_price"] * price * s_jt)

# take median of cross elasticity by upc
cross_elas_df <- cross_elas_jt |>
  select(upc, descrip, cross_elas) |>
  group_by(upc, descrip) |>
  summarize(cross_elas = median(cross_elas), .groups = 'drop')

# create elasticity matrix
J <- nrow(cross_elas_df)
elas_mat_med <- matrix( rep(cross_elas_df$cross_elas, J), nrow = J, ncol = J)
diag(elas_mat_med) <- own_elas_df$own_elas
colnames(elas_mat_med) <- rownames(elas_mat_med) <- as.character(cross_elas_df$descrip)
res_elas_mat <- elas_mat_med |>
  knitr::kable(format = "latex", booktabs = TRUE, caption = "Median Elasticity matrix") |>
```

```r
  kableExtra::kable_styling(latex_options = "hold_position")
cat(res_elas_mat)
```

```
## \begin{table}[!h]
## \centering
## \caption{\label{tab:unnamed-chunk-4}Median Elasticity matrix}
## \centering
## \begin{tabular}[t]{lrrrrrrrrrr}
## \toprule
##   & PEPSI COLA N/R & PEPSI COLA N.R. BOTT & DIET PEPSI N.R. BOTT & CAFFEINE FREE PEPSI & DIET PEPSI C
## \midrule
## PEPSI COLA N/R & -3.61050 & 0.03905 & 0.03905 & 0.03905 & 0.03905 & 0.03905 & 0.03905 & 0.03905 & 0.0
## PEPSI COLA N.R. BOTT & 0.00594 & -4.14686 & 0.00594 & 0.00594 & 0.00594 & 0.00594 & 0.00594 & 0.0059
## DIET PEPSI N.R. BOTT & 0.00396 & 0.00396 & -4.14885 & 0.00396 & 0.00396 & 0.00396 & 0.00396 & 0.0039
## CAFFEINE FREE PEPSI & 0.01155 & 0.01155 & 0.01155 & -3.90668 & 0.01155 & 0.01155 & 0.01155 & 0.01155
## DIET PEPSI CAFFEINE & 0.01097 & 0.01097 & 0.01097 & 0.01097 & -3.90732 & 0.01097 & 0.01097 & 0.01097
## \addlinespace
## DOM ORANGE SODA & 0.02057 & 0.02057 & 0.02057 & 0.02057 & 0.02057 & -3.28010 & 0.02057 & 0.02057 & 0
## COCA-COLA CLASSIC CA & 0.01222 & 0.01222 & 0.01222 & 0.01222 & 0.01222 & 0.01222 & -6.87568 & 0.0122
## COCA-COLA CLASSIC & 0.02636 & 0.02636 & 0.02636 & 0.02636 & 0.02636 & 0.02636 & 0.02636 & -3.62885 &
## COKE DIET CANS & 0.01006 & 0.01006 & 0.01006 & 0.01006 & 0.01006 & 0.01006 & 0.01006 & 0.01006 & -6.8
## DR PEPPER & 0.00479 & 0.00479 & 0.00479 & 0.00479 & 0.00479 & 0.00479 & 0.00479 & 0.00479 & 0.00479 &
## \bottomrule
## \end{tabular}
## \end{table}
```

Setup function for Q5 & Q6

```r
# compute markup and marginal cost
compute_markup <- function(own_elas_jt, closs_elas_jt,  owner_mat, is_Multi=FALSE) {

  unique_week <- unique(own_elas_jt$week)
  unique_store <- unique(own_elas_jt$store)

  market_comb <- expand.grid(week = unique_week, store = unique_store)

  # store list of markup and mc
  markup_mc_list <- vector("list", nrow(market_comb))

  # calc markup and mc for each market
  for (i in 1:nrow(market_comb)) {

    week_i <- market_comb$week[i]
    store_i <- market_comb$store[i]

    # subset of a particular market
    sub_own_elas <- own_elas_jt |> filter(week == week_i & store == store_i)
    sub_cross_elas <- cross_elas_jt |> filter(week == week_i & store == store_i)

    # number of product
    J <- nrow(sub_cross_elas)
```

```r
    # if multi-product bertrand nash
    if (is_Multi) {
      for (j in 1:J) {
        for (k in 1:J) {
          # if j's brand is equal to k's brand, take 1
          if (sub_cross_elas$Brand[j] == sub_cross_elas$Brand[k]){
            owner_mat[j,k] = 1
          }
        }
      }
    }

    # calculate elasticity matrix
    elas_mat <- matrix(rep(sub_cross_elas$cross_elas, J), nrow = J, ncol = J)
    diag(elas_mat) <- sub_own_elas$own_elas

    # calculate matrix S(p)
    price_mat <- matrix(rep((1/sub_cross_elas$price), J), nrow = J, ncol = J)
    share_mat <- t(matrix(rep(sub_cross_elas$s_jt, J), nrow = J, ncol = J))
    S_p_mat <- (-1) * elas_mat * price_mat * share_mat

    # calculate markup for a paticular market
    D_p_mat <- owner_mat * S_p_mat
    markup <- solve(D_p_mat) %*% sub_cross_elas$s_jt
    mc <- sub_cross_elas$price - markup

    return_df <- data.frame(week = market_comb$week[i],
                            store = market_comb$store[i],
                            Brand = sub_cross_elas$Brand,
                            upc = sub_cross_elas$upc,
                            descrip = sub_cross_elas$descrip,
                            price = sub_cross_elas$price,
                            move = sub_cross_elas$move,
                            markup = markup,
                            mc = mc) %>%
      mutate(markup_dev_p = markup/price)

    markup_mc_list[[i]] <- return_df
  }

  result_df <- do.call(rbind, markup_mc_list)
  return(result_df)
}
```

Answer to Q5

```r
### Q5 Answer ###

# compute single product Nash equilibrium for each market
J <- nrow(cross_elas_df)
Omega_sin <- diag(J)
```

```r
markup_sin <- compute_markup(own_elas_jt, cross_elas_jt, Omega_sin)
markup_sin_med <- markup_sin |>
  select(Brand, descrip, markup, markup_dev_p, mc) |>
  group_by(Brand, descrip) |>
  summarize(markup = median(markup), markup_dev_p = median(markup_dev_p),
            mc = median(mc), .groups = 'drop')

# compute multi product Nash equilibrium for each market
Omega_multi <- matrix(0, J, J)
markup_multi <- compute_markup(own_elas_jt, cross_elas_jt, Omega_multi, is_Multi=TRUE)
markup_multi_med <- markup_multi |>
  select(Brand, descrip, markup, markup_dev_p, mc) |>
  group_by(Brand, descrip) |>
  summarize(markup = median(markup), markup_dev_p = median(markup_dev_p),
            mc = median(mc), .groups = 'drop')

# joint pricing of all brands
J <- nrow(cross_elas_df)
Omega_joint <- matrix(1, nrow = J, ncol = J)
markup_joint <- compute_markup(own_elas_jt, cross_elas_jt, Omega_joint)
markup_joint_med <- markup_joint |>
  select(Brand, descrip, markup, markup_dev_p, mc) |>
  group_by(Brand, descrip) |>
  summarize(markup = median(markup), markup_dev_p = median(markup_dev_p),
            mc = median(mc), .groups = 'drop')

res_markup <- left_join(markup_sin_med |> select(Brand, descrip, markup, markup_dev_p),
                        markup_multi_med |> select(Brand, descrip, markup, markup_dev_p),
                        by = c("Brand", "descrip")) |>
                left_join(markup_joint_med |> select(Brand, descrip, markup, markup_dev_p),
                         by = c("Brand", "descrip"), suffix = c(".multi", ".joint"))

res_markup <- res_markup %>%
  knitr::kable(format = "latex", booktabs = TRUE, caption = "Estimated markup") %>%
  kableExtra::kable_styling(latex_options = "hold_position")
cat(res_markup)
```

```
## \begin{table}[!h]
## \centering
## \caption{\label{tab:unnamed-chunk-6}Estimated markup}
## \centering
## \begin{tabular}[t]{rlrrrrrr}
## \toprule
## Brand & descrip & markup.x & markup\_dev\_p.x & markup.y & markup\_dev\_p.y & markup & markup\_dev\_
## \midrule
## 12000 & CAFFEINE FREE PEPSI & 0.00564 & 0.25597 & 0.00574 & 0.25908 & 0.00594 & 0.27667\\
## 12000 & DIET PEPSI CAFFEINE & 0.00564 & 0.25593 & 0.00574 & 0.25908 & 0.00594 & 0.27667\\
## 12000 & DIET PEPSI N.R. BOTT & 0.00563 & 0.24103 & 0.00574 & 0.24528 & 0.00594 & 0.25294\\
## 12000 & PEPSI COLA N.R. BOTT & 0.00563 & 0.24115 & 0.00574 & 0.24528 & 0.00594 & 0.25294\\
## 12000 & PEPSI COLA N/R & 0.00568 & 0.27697 & 0.00574 & 0.27895 & 0.00594 & 0.29092\\
## \addlinespace
## 38281 & DOM ORANGE SODA & 0.00566 & 0.30487 & 0.00566 & 0.30487 & 0.00594 & 0.31504\\
## 49000 & COCA-COLA CLASSIC & 0.00566 & 0.27557 & 0.00570 & 0.27836 & 0.00594 & 0.29169\\
```

```
## 49000 & COCA-COLA CLASSIC CA & 0.00564 & 0.14544 & 0.00570 & 0.14748 & 0.00594 & 0.15593\\
## 49000 & COKE DIET CANS & 0.00563 & 0.14539 & 0.00570 & 0.14748 & 0.00594 & 0.15593\\
## 54900 & DR PEPPER & 0.00563 & 0.27412 & 0.00563 & 0.27412 & 0.00594 & 0.29885\\
## \bottomrule
## \end{tabular}
## \end{table}
```

Answer to Q6

```
res_mc <- left_join(markup_sin_med |> select(Brand, descrip, mc),
                        markup_multi_med |> select(Brand, descrip, mc),
                        by = c("Brand", "descrip")) |>
                left_join(markup_joint_med |> select(Brand, descrip, mc),
                          by = c("Brand", "descrip"), suffix = c(".multi", ".joint"))

res_mc <- res_mc %>%
  knitr::kable(format = "latex", booktabs = TRUE, caption = "Estimated markup") %>%
  kableExtra::kable_styling(latex_options = "hold_position")
cat(res_mc)
```

```
## \begin{table}[!h]
## \centering
## \caption{\label{tab:unnamed-chunk-7}Estimated markup}
## \centering
## \begin{tabular}[t]{rlrrr}
## \toprule
## Brand & descrip & mc.x & mc.y & mc\\
## \midrule
## 12000 & CAFFEINE FREE PEPSI & 0.01639 & 0.01632 & 0.01601\\
## 12000 & DIET PEPSI CAFFEINE & 0.01639 & 0.01632 & 0.01601\\
## 12000 & DIET PEPSI N.R. BOTT & 0.01773 & 0.01763 & 0.01746\\
## 12000 & PEPSI COLA N.R. BOTT & 0.01773 & 0.01763 & 0.01746\\
## 12000 & PEPSI COLA N/R & 0.01486 & 0.01482 & 0.01461\\
## \addlinespace
## 38281 & DOM ORANGE SODA & 0.01289 & 0.01289 & 0.01274\\
## 49000 & COCA-COLA CLASSIC & 0.01489 & 0.01486 & 0.01461\\
## 49000 & COCA-COLA CLASSIC CA & 0.03311 & 0.03304 & 0.03277\\
## 49000 & COKE DIET CANS & 0.03312 & 0.03304 & 0.03277\\
## 54900 & DR PEPPER & 0.01492 & 0.01492 & 0.01434\\
## \bottomrule
## \end{tabular}
## \end{table}
```

Setup function for Q7

```
compute_price_eq = function(df, init_p, owner_mat, alpha_hat, beta_hat, is_Multi =TRUE){

  # iteration stop parameters
  epsilon <- 1e-5
  df$price <- init_p
```

```r
error <- 10000
count_max <- 20
counter <- 0

while (error > epsilon & counter < count_max) {
  # check counter
  counter = counter + 1
  print(counter)
  print(error)

  unique_week <- unique(df$week)
  unique_store <- unique(df$store)

  market_comb <- expand.grid(week = unique_week, store = unique_store)

  # store list of p
  p_k_list <- vector("list", nrow(market_comb))

  # calc eta_(p) for each market
  for (i in 1:nrow(market_comb)) {

    week_i <- market_comb$week[i]
    store_i <- market_comb$store[i]

    # subset of a particular market
    df_t <- df |> filter(week == week_i & store == store_i)

    # compute own elasticity for each market
    sub_own_elas <- df_t |>
      select(week, store, upc, descrip, price, move, s_jt, Brand, mc) |>
      mutate(own_elas = alpha_hat * price * (1 - s_jt))

    # compute cross elasticity for each market
    sub_cross_elas <- df_t |>
      select(week, store, upc, descrip, price, move, s_jt, Brand, mc) |>
      mutate(cross_elas = (-1) * alpha_hat * price * s_jt)

    # number of product
    J <- nrow(sub_cross_elas)

    # if multi-product bertrand nash
    if (is_Multi) {
      for (j in 1:J) {
        for (k in 1:J) {
          # if j's brand is equal to k's brand, take 1
          if (sub_cross_elas$Brand[j] == sub_cross_elas$Brand[k]){
            owner_mat[j,k] = 1
          }
        }
      }
    }

    # calculate elasticity matrix
```

```
      elas_mat <- matrix(rep(sub_cross_elas$cross_elas, J), nrow = J, ncol = J)
      diag(elas_mat) <- sub_own_elas$own_elas

      # calculate matrix S(p)
      price_mat <- matrix(rep((1/sub_cross_elas$price), J), nrow = J, ncol = J)
      share_mat <- t(matrix(rep(sub_cross_elas$s_jt, J), nrow = J, ncol = J))
      S_p_mat <- (-1) * elas_mat * price_mat * share_mat

      # calculate p_k for a paticular market
      D_p_mat <- owner_mat * S_p_mat
      p_k <- sub_cross_elas$mc + solve(D_p_mat) %*% sub_cross_elas$s_jt

      # calculate new share
      X_j <- as.matrix(df_t[, grepl("dummy_", colnames(df_t))])
      xi_hat <- log(df_t$s_jt/df_t$s_0t) - (alpha_hat * df_t$price + X_j %*% beta_hat)
      numerator <- exp(alpha_hat * p_k + X_j %*% beta_hat + xi_hat)
      s_jt_old <- df$s_jt
      s_jt <- numerator / (1 + sum(numerator))

      # to dataframe
      market_df <- data.frame(week = market_comb$week[i],
                              store = market_comb$store[i],
                              upc = sub_cross_elas$upc,
                              s_jt_post = s_jt,
                              price_post = p_k)
      # store dataframe
      p_k_list[[i]] <- market_df
    }
    return_df <- do.call(rbind, p_k_list)

    #update
    error <- max(abs(return_df$price_post - df$price))
    df$price <- return_df$price_post
    df$s_jt <- return_df$s_jt_post
  }
  return(return_df)
}
```

Answer to Q7

```
# compute price equilibrium for counterfactural exercise
df <- left_join(df, markup_multi |> select(store, week, upc, mc),
                by = c("store", "week", "upc"))

# initial price
init_p <- rep(0.025, nrow(df))

# computation result
alpha_hat <- model1_IV$coefficients["fit_price"]
beta_hat <- as.vector(model1_IV$coefficients)[3:11]
df <- df |> mutate(mc_post = mc * 1.1)# add 10% increase
p_eq_ct <- df %>%
```

```r
  mutate(mc = mc_post) %>% # set marginal cost 10% increase
  compute_price_eq(., init_p, Omega_multi, alpha_hat, beta_hat)
```

```
## [1] 1
## [1] 10000
## [1] 2
## [1] 0.01859
## [1] 3
## [1] 0.024061
## [1] 4
## [1] 0.022543
## [1] 5
## [1] 0.004114
## [1] 6
## [1] 0.0038043
## [1] 7
## [1] 0.0025594
## [1] 8
## [1] 0.0043849
## [1] 9
## [1] 0.0033577
## [1] 10
## [1] 0.0029579
## [1] 11
## [1] 0.0019816
## [1] 12
## [1] 0.0021967
## [1] 13
## [1] 0.0024741
## [1] 14
## [1] 0.0027788
## [1] 15
## [1] 0.0034923
## [1] 16
## [1] 0.0037226
## [1] 17
## [1] 0.0036609
## [1] 18
## [1] 0.0035939
## [1] 19
## [1] 0.0028778
## [1] 20
## [1] 0.0015755
```

```r
# join result
df <- left_join(df, p_eq_ct, c("store", "week", "upc")) |>
  mutate(price_pre = price)

# results to tex
res_p_eq_ct <- df |>
  select(Brand, descrip, price_pre, price_post, mc, mc_post) |>
  group_by(Brand, descrip) |>
  summarize(price_pre = median(price_pre),
```

```r
            price_post = median(price_post),
            mc_price_share_diff = median((mc_post - mc)/price_post),
            .groups = 'drop') |>
  knitr::kable(format = "latex", booktabs = TRUE,
               caption = "Counterfactural Exercise (MC increases by 10%)") |>
  kableExtra::kable_styling(latex_options = "hold_position")
cat(res_p_eq_ct)
```

```
## \begin{table}[!h]
## \centering
## \caption{\label{tab:unnamed-chunk-9}Counterfactural Exercise (MC increases by 10%)}
## \centering
## \begin{tabular}[t]{rlrrr}
## \toprule
## Brand & descrip & price\_pre & price\_post & mc\_price\_share\_diff\\
## \midrule
## 12000 & CAFFEINE FREE PEPSI & 0.02203 & 0.02372 & 0.06883\\
## 12000 & DIET PEPSI CAFFEINE & 0.02203 & 0.02372 & 0.06883\\
## 12000 & DIET PEPSI N.R. BOTT & 0.02336 & 0.02516 & 0.07007\\
## 12000 & PEPSI COLA N.R. BOTT & 0.02336 & 0.02516 & 0.07007\\
## 12000 & PEPSI COLA N/R & 0.02055 & 0.02204 & 0.06723\\
## \addlinespace
## 38281 & DOM ORANGE SODA & 0.01854 & 0.01990 & 0.06476\\
## 49000 & COCA-COLA CLASSIC & 0.02055 & 0.02201 & 0.06748\\
## 49000 & COCA-COLA CLASSIC CA & 0.03875 & 0.04203 & 0.07861\\
## 49000 & COKE DIET CANS & 0.03875 & 0.04203 & 0.07861\\
## 54900 & DR PEPPER & 0.02055 & 0.02204 & 0.06770\\
## \bottomrule
## \end{tabular}
## \end{table}
```

Setup function for Q8

```r
compute_walfare = function(df, alpha_hat, beta_hat){

    unique_week <- unique(df$week)
    unique_store <- unique(df$store)

    # market list
    market_comb <- expand.grid(week = unique_week, store = unique_store)

    # store list of p
    walfare_list <- vector("list", nrow(market_comb))

    # calculating walfare for each market
    for (i in 1:nrow(market_comb)) {
      week_i <- market_comb$week[i]
      store_i <- market_comb$store[i]

      # subset of a particular market
      df_t <- df |> filter(week == week_i & store == store_i)
```

```r
      # matrix X_j
      X_j <- as.matrix(df_t[, grepl("dummy_", colnames(df_t))])

      # calculate pre delta_jt
      xi_hat <- log(df_t$s_jt_old/df_t$s_0t) - (alpha_hat * df_t$price_old + X_j %*% beta_hat)
      delta<- alpha_hat * df_t$price + X_j %*% beta_hat + xi_hat

      # caliculate CV based on Small and Rosen (1981)
      #CV <- (1 / (-1) * alpha_hat) * (1+log(sum(exp(delta_post))) - (1+ log(sum(exp(delta_pre)))))
      CS <- 1+log(sum(exp(delta)))

      # calculate producer surplus for each brand in a paticular market
      M_t <- mean(df_t$custcoun)
      PS_f_t <- df_t |>
        mutate(calc_profit = (price - mc) * s_jt * M_t) |>
        group_by(Brand, week, store) |>
        summarize(calc_profit = sum(calc_profit), .groups = 'drop')

      # summation accross firm in the market
      PS  <-  PS_f_t |>
        summarize(calc_profit = sum(calc_profit), .groups = 'drop')

      # to dataframe
      walfare_df <- data.frame(week = week_i,
                               store = store_i,
                               CS = CS,
                               PS = PS$calc_profit,
                               TS = CS + PS$calc_profit)

      # store dataframe
      walfare_list[[i]] <- walfare_df
    }
    return_df <- do.call(rbind, walfare_list)
    return(return_df)
}
```

Answer to Q8

```r
# walfare in pre counter factural exercise
walfare_pre <- df |>
  mutate(price_old = price, s_jt_old = s_jt) |> # set variables for xi
  compute_walfare(alpha_hat, beta_hat)

# walfare in counter factural exercise
walfare_post <- df |>
  mutate(price_old = price, s_jt_old = s_jt,
         price = price_post, mc = mc_post, s_jt = s_jt_post) |> # use updated variables
  compute_walfare(alpha_hat, beta_hat)

# take difference of each surplus
Diff_CS <- walfare_post$CS - walfare_pre$CS
```

```r
Diff_PS <- walfare_post$PS - walfare_pre$PS
Diff_TS <- walfare_post$TS - walfare_pre$TS

# results to tex
res_walfare <- walfare_post |>
  mutate(Diff_CS = Diff_CS, Diff_PS = Diff_PS, Diff_TS = Diff_TS) |>
  select(week, store, Diff_CS, Diff_PS, Diff_TS) |>
  summarize(sum_Diff_CS = sum(Diff_CS), sum_Diff_PS = sum(Diff_PS), sum_Diff_TS = sum(Diff_TS),
            mean_Diff_CS = mean(Diff_CS), mean_Diff_PS = mean(Diff_PS), mean_Diff_TS = mean(Diff_TS),
            .groups = 'drop') |>
  knitr::kable(format = "latex", booktabs = TRUE,
               caption = "Summary of each surplus difference") |>
  kableExtra::kable_styling(latex_options = "hold_position")
cat(res_walfare)
```

```
## \begin{table}[!h]
## \centering
## \caption{\label{tab:unnamed-chunk-11}Summary of each surplus difference}
## \centering
## \begin{tabular}[t]{rrrrrr}
## \toprule
## sum\_Diff\_CS & sum\_Diff\_PS & sum\_Diff\_TS & mean\_Diff\_CS & mean\_Diff\_PS & mean\_Diff\_TS\\
## \midrule
## -401.63 & 363.19 & -38.437 & -0.25745 & 0.23282 & -0.02464\\
## \bottomrule
## \end{tabular}
## \end{table}
```

Answer to Q9

```r
### Q9 Answer ###

# set merged brand code 99999
## Note: Ownership matrix is automatically calculated based on merged brand code in this function
df <- df %>%
  mutate(Brand_merged = if_else(Brand %in% c(12000, 49000), 99999, Brand))


res_merged <- df %>%
  mutate(Brand = Brand_merged) %>% # set merged brand code
  compute_price_eq(., init_p, Omega_multi, alpha_hat, beta_hat) %>%
  rename(price_merged = price_post)
```

```
## [1] 1
## [1] 10000
## [1] 2
## [1] 0.018056
## [1] 3
## [1] 0.024164
## [1] 4
## [1] 0.022458
```

```
## [1] 5
## [1] 0.0047659
## [1] 6
## [1] 0.0040204
## [1] 7
## [1] 0.0036371
## [1] 8
## [1] 0.0043358
## [1] 9
## [1] 0.0041812
## [1] 10
## [1] 0.0030343
## [1] 11
## [1] 0.0020701
## [1] 12
## [1] 0.0034541
## [1] 13
## [1] 0.0022893
## [1] 14
## [1] 0.0024913
## [1] 15
## [1] 0.002979
## [1] 16
## [1] 0.0036706
## [1] 17
## [1] 0.0034566
## [1] 18
## [1] 0.0033758
## [1] 19
## [1] 0.0030004
## [1] 20
## [1] 0.0014941
```

```r
# join result
df <- left_join(df, res_merged, c("store", "week", "upc")) |>
  mutate(price_pre = price)

res_merged <- df |>
  select(Brand, Brand_merged,  descrip, price_pre, price_merged) |>
  group_by(Brand, Brand_merged,  descrip) |>
  summarize(price_pre = median(price_pre),
            price_merged = median(price_merged),
            .groups = 'drop') |>
  knitr::kable(format = "latex", booktabs = TRUE,
               caption = "Merge impact on the equilibrium price (median across market)") |>
  kableExtra::kable_styling(latex_options = "hold_position")
cat(res_merged)
```

```
## \begin{table}[!h]
## \centering
## \caption{\label{tab:unnamed-chunk-12}Merge impact on the equilibrium price (median across market)}
## \centering
## \begin{tabular}[t]{rrlrr}
## \toprule
```

```
## Brand & Brand\_merged & descrip & price\_pre & price\_merged\\
## \midrule
## 12000 & 99999 & CAFFEINE FREE PEPSI & 0.02203 & 0.02217\\
## 12000 & 99999 & DIET PEPSI CAFFEINE & 0.02203 & 0.02217\\
## 12000 & 99999 & DIET PEPSI N.R. BOTT & 0.02336 & 0.02352\\
## 12000 & 99999 & PEPSI COLA N.R. BOTT & 0.02336 & 0.02352\\
## 12000 & 99999 & PEPSI COLA N/R & 0.02055 & 0.02068\\
## \addlinespace
## 38281 & 38281 & DOM ORANGE SODA & 0.01854 & 0.01862\\
## 49000 & 99999 & COCA-COLA CLASSIC & 0.02055 & 0.02067\\
## 49000 & 99999 & COCA-COLA CLASSIC CA & 0.03875 & 0.03886\\
## 49000 & 99999 & COKE DIET CANS & 0.03875 & 0.03886\\
## 54900 & 54900 & DR PEPPER & 0.02055 & 0.02055\\
## \bottomrule
## \end{tabular}
## \end{table}
```