

OLS (DC) peanut butter, by manufacturer

This initial analysis is performed on the peanut butter data and looks at individual treated store sales compared to an aggregate of the sales of the other stores in the market (DC Area). As of now this is only for the sales of captain, validator, and private label between stores. We believe that the treated retailer is 842.

Install packages and load library.

```
library(tidyverse)

## -- Attaching packages -----
## ----- tidyverse 1.2.1 --

## v ggplot2 3.1.0      v purrr  0.2.5
## v tibble  2.0.0      v dplyr  0.7.8
## v tidyr   0.8.2      v stringr 1.3.1
## v readr   1.3.1      v forcats 0.3.0

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

library(lubridate)

##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date
```

Unzip the 'peanutbutter.tgz' file. Read in 2011 movement and store files, merge them based on the store code, and filter for 'F' (food stores) and states North Dakota, Minnesota, Missouri, and D.C.

```
untar("peanutbutter.tgz", list = TRUE)

untar("peanutbutter.tgz")

move_11 <-
read_tsv("nielsen_extracts/RMS/2011/Movement_Files/0506_2011/1421_2011.tsv")

## Parsed with column specification:
## cols(
##   store_code_uc = col_double(),
##   upc = col_character(),
##   week_end = col_double(),
```

```

##  units = col_double(),
##  prmult = col_double(),
##  price = col_double(),
##  feature = col_double(),
##  display = col_double()
## )

stores_11 <-
read_tsv("nielsen_extracts/RMS/2011/Annual_Files/stores_2011.tsv")

## Parsed with column specification:
## cols(
##   store_code_uc = col_double(),
##   year = col_double(),
##   parent_code = col_double(),
##   retailer_code = col_double(),
##   channel_code = col_character(),
##   store_zip3 = col_character(),
##   fips_state_code = col_double(),
##   fips_state_descr = col_character(),
##   fips_county_code = col_double(),
##   fips_county_descr = col_character(),
##   dma_code = col_double(),
##   dma_descr = col_character()
## )

full_11 <- move_11 %>%
  inner_join(stores_11, by = "store_code_uc") %>%
  filter(channel_code == "F",
         fips_state_descr %in% c("ND", "MN", "MO", "DC"))

save(full_11, file = 'pb_full_11.RData')

rm(move_11, stores_11, full_11)

```

Perform the previous step for 2012 and 2013.

```

move_12 <-
read_tsv("nielsen_extracts/RMS/2012/Movement_Files/0506_2012/1421_2012.tsv")

## Parsed with column specification:
## cols(
##   store_code_uc = col_double(),
##   upc = col_character(),
##   week_end = col_double(),
##   units = col_double(),
##   prmult = col_double(),
##   price = col_double(),
##   feature = col_double(),
##   display = col_double()
## )

```

```
stores_12 <-  
read_tsv("nielsen_extracts/RMS/2012/Annual_Files/stores_2012.tsv")  
  
## Parsed with column specification:  
## cols(  
##   store_code_uc = col_double(),  
##   year = col_double(),  
##   parent_code = col_double(),  
##   retailer_code = col_double(),  
##   channel_code = col_character(),  
##   store_zip3 = col_character(),  
##   fips_state_code = col_double(),  
##   fips_state_descr = col_character(),  
##   fips_county_code = col_double(),  
##   fips_county_descr = col_character(),  
##   dma_code = col_double(),  
##   dma_descr = col_character()  
## )  
  
full_12 <- move_12 %>%  
  inner_join(stores_12, by = "store_code_uc") %>%  
  filter(channel_code == "F",  
         fips_state_descr %in% c("ND", "MN", "MO", "DC"))  
  
save(full_12, file = 'pb_full_12.RData')  
  
rm(move_12, stores_12, full_12)  
  
move_13 <-  
read_tsv("nielsen_extracts/RMS/2013/Movement_Files/0506_2013/1421_2013.tsv")  
  
## Parsed with column specification:  
## cols(  
##   store_code_uc = col_double(),  
##   upc = col_character(),  
##   week_end = col_double(),  
##   units = col_double(),  
##   prmult = col_double(),  
##   price = col_double(),  
##   feature = col_double(),  
##   display = col_double()  
## )  
  
stores_13 <-  
read_tsv("nielsen_extracts/RMS/2013/Annual_Files/stores_2013.tsv",  
         col_types = list(col_double(), col_double(),  
                           col_double(), col_double(), col_character(),  
                           col_character(), col_double(), col_character(),  
                           col_double(), col_character(), col_double(),  
                           col_character()))
```

```

full_13 <- move_13 %>%
  inner_join(stores_13, by = "store_code_uc") %>%
  filter(channel_code == "F",
         fips_state_descr %in% c("ND", "MN", "MO", "DC"))

save(full_13, file = 'pb_full_13.RData')

rm(move_13, stores_13)

```

Bind full_11,12,13 into one tbl, 'full_11_12_13'. Overwrite the variable week_end to be in year-month-date format. Create sales variable, which is the number of units sold multiplied by the price.

```

load('pb_full_11.RData')
load('pb_full_12.RData')

full_11_12_13 <- full_11 %>%
  bind_rows(full_12) %>%
  bind_rows(full_13) %>%
  mutate(week_end = ymd(week_end), sales = units * price)

rm(full_11, full_12, full_13)

```

Read in products master file.

```

products <- read_tsv('products.tsv', quote = "")

## Parsed with column specification:
## cols(
##   upc = col_character(),
##   upc_ver_uc = col_double(),
##   upc_descr = col_character(),
##   product_module_code = col_double(),
##   product_module_descr = col_character(),
##   product_group_code = col_double(),
##   product_group_descr = col_character(),
##   department_code = col_double(),
##   department_descr = col_character(),
##   brand_code_uc = col_double(),
##   brand_descr = col_character(),
##   multi = col_double(),
##   size1_code_uc = col_double(),
##   size1_amount = col_double(),
##   size1_units = col_character(),
##   dataset_found_uc = col_character(),
##   size1_change_flag_uc = col_double()
## )

```

Merge products file and full_11_12_13.

```
full_11_12_13 <- full_11_12_13 %>%
  left_join(products, by = 'upc')

rm(products)
```

Filter the full dataset for data pertaining to the retailers in DC. Display the retailer codes in DC.

```
dc_data <- full_11_12_13 %>%
  filter(fips_state_descr == 'DC')

treated_stores <- dc_data %>% filter(retailer_code==842) %>%
  distinct(store_code_uc)
```

Assign manufacturer for each brand. Aggregate sales by manufacturer for all other stores not in 842. Divide all sales by average sales over entire pre and post period. Create treatment and control variables for the captain, validator, and private label.

```
dc_data <- dc_data %>%
  mutate(manuf_name = substr(dc_data$brand_descr, 1, 3)) %>%
  filter(manuf_name %in% c('SKI', 'JIF', 'SMU', 'SIM', 'SAN', 'ADA', 'CTL'))
%>%
  mutate(actual_manuf = if_else(manuf_name == 'CTL', 'CTL',
                                if_else(manuf_name == 'SKI', 'UNI', 'SMU')))

ag_dc <- dc_data %>% group_by(store_code_uc, actual_manuf, week_end) %>%
  summarize(sales=sum(sales))

treat_uni <- ag_dc %>% filter(actual_manuf=="UNI") %>%
  filter(store_code_uc==treated_stores) %>%
  mutate(scaled_sales=sales/mean(sales))

treat_ctl <- ag_dc %>% filter(actual_manuf=="CTL") %>%
  filter(store_code_uc==treated_stores) %>%
  mutate(scaled_sales=sales/mean(sales))

treat_smu <- ag_dc %>% filter(actual_manuf=="SMU") %>%
  filter(store_code_uc==treated_stores) %>%
  mutate(scaled_sales=sales/mean(sales))

control_uni <- ag_dc %>% filter(store_code_uc!=treated_stores) %>%
  filter(actual_manuf=="UNI") %>% group_by(week_end) %>%
  summarize(sales=sum(sales)) %>% mutate(scaled_sales=sales/mean(sales),
    store_code_uc=00000)

control_ctl <- ag_dc %>% filter(store_code_uc!=treated_stores) %>%
  filter(actual_manuf=="CTL") %>% group_by(week_end) %>%
  summarize(sales=sum(sales)) %>% mutate(scaled_sales=sales/mean(sales),
    store_code_uc=00000)
```

```

control_smu <- ag_dc %>% filter(store_code_uc!=treated_stores) %>%
filter(actual_manuf=="SMU") %>% group_by(week_end) %>%
  summarize(sales=sum(sales)) %>% mutate(scaled_sales=sales/mean(sales),
store_code_uc=00000)

ols_uni <- treat_uni %>% bind_rows(control_uni) %>%
mutate(treated=if_else(store_code_uc==treated_stores, 1, 0),
post=if_else(week_end >
"2012-07-16", 1, 0))

ols_ctl <- treat_ctl %>% bind_rows(control_ctl) %>%
mutate(treated=if_else(store_code_uc==treated_stores, 1, 0),
post=if_else(week_end >
"2012-07-16", 1, 0))

ols_smu <- treat_smu %>% bind_rows(control_smu) %>%
mutate(treated=if_else(store_code_uc==treated_stores, 1, 0),
post=if_else(week_end >
"2012-07-16", 1, 0))

```

Run OLS regression for captain, validator, and private label. Plot graphs showing sales over time.

```

ols_uni %>% lm(scaled_sales~treated + post + I(treated*post), data=.) %>%
summary()

##
## Call:
## lm(formula = scaled_sales ~ treated + post + I(treated * post),
##     data = .)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.71758 -0.19337 -0.01462  0.14785  1.50988
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.91803    0.03469  26.467  < 2e-16 ***
## treated         0.02938    0.04905   0.599  0.549618
## post           0.16933    0.04985   3.397  0.000771 ***
## I(treated * post) -0.06070    0.07050  -0.861  0.389949
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3122 on 310 degrees of freedom
## Multiple R-squared:  0.04991,    Adjusted R-squared:  0.04072
## F-statistic: 5.428 on 3 and 310 DF,  p-value: 0.00119

ols_ctl %>% lm(scaled_sales~treated + post + I(treated*post), data=.) %>%
summary()

```

```
##
## Call:
## lm(formula = scaled_sales ~ treated + post + I(treated * post),
##     data = .)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.73397 -0.15807 -0.02386  0.13233  1.19397
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.98092    0.03357  29.222 < 2e-16 ***
## treated        -0.15815    0.04747  -3.331 0.000969 ***
## post           0.03941    0.04825   0.817 0.414644
## I(treated * post) 0.32671    0.06823   4.788 2.61e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3021 on 310 degrees of freedom
## Multiple R-squared:  0.1582, Adjusted R-squared:  0.15
## F-statistic: 19.42 on 3 and 310 DF,  p-value: 1.46e-11

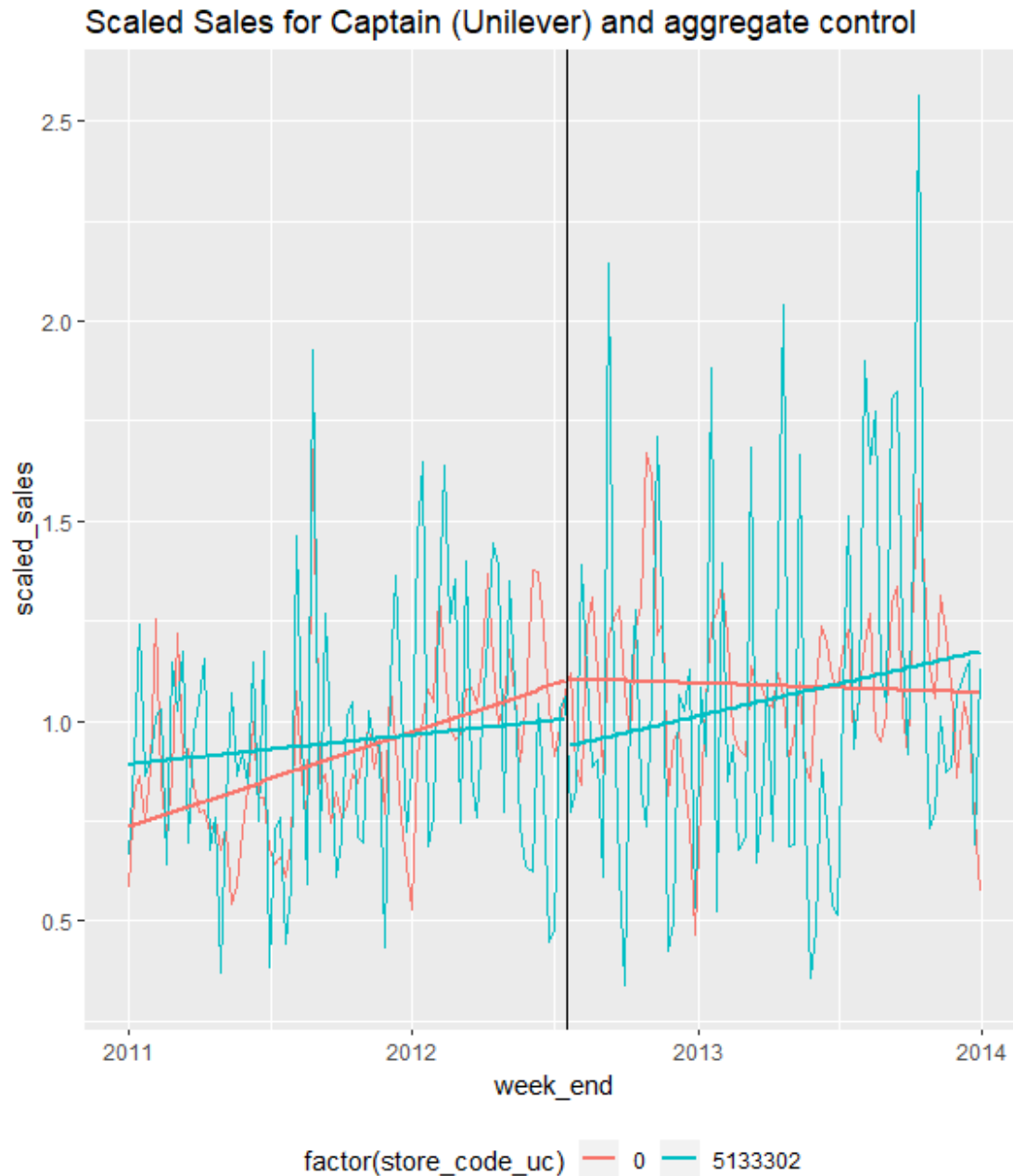
ols_smu %>% lm(scaled_sales~treated + post + I(treated*post), data=.) %>%
summary()

##
## Call:
## lm(formula = scaled_sales ~ treated + post + I(treated * post),
##     data = .)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.54018 -0.11124 -0.00449  0.09048  0.78728
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.940779    0.020580  45.713 < 2e-16 ***
## treated        0.008124    0.029105   0.279   0.780
## post           0.122338    0.029580   4.136 4.56e-05 ***
## I(treated * post) -0.016782    0.041832  -0.401   0.689
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1852 on 310 degrees of freedom
## Multiple R-squared:  0.08781, Adjusted R-squared:  0.07898
## F-statistic: 9.947 on 3 and 310 DF,  p-value: 2.804e-06

pre_cc_uni <- ols_uni %>%
filter(week_end <= as.Date("2012-07-16"))
```

```
post_cc_uni <- ols_uni %>%
  filter(week_end > as.Date("2012-07-16"))

ggplot(ols_uni, aes(week_end, scaled_sales, col = factor(store_code_uc))) +
  geom_line() +
  geom_smooth(aes(week_end, scaled_sales), data = pre_cc_uni, method = 'lm',
se = FALSE) +
  geom_smooth(aes(week_end, scaled_sales), data = post_cc_uni, method = 'lm',
se = FALSE) +
  geom_vline(xintercept = as.numeric(as.Date("2012-07-16"))) +
  ggtitle("Scaled Sales for Captain (Unilever) and aggregate control") +
  theme(legend.position="bottom")
```

```
pre_cc_ctl <- ols_ctl %>%
  filter(week_end <= as.Date("2012-07-16"))

post_cc_ctl <- ols_ctl %>%
  filter(week_end > as.Date("2012-07-16"))

ggplot(ols_ctl, aes(week_end, scaled_sales, col = factor(store_code_uc))) +
  geom_line() +
  geom_smooth(aes(week_end, scaled_sales), data = pre_cc_ctl, method = 'lm',
    se = FALSE) +
  geom_smooth(aes(week_end, scaled_sales), data = post_cc_ctl, method = 'lm',
```

```
se = FALSE) +
  geom_vline(xintercept = as.numeric(as.Date("2012-07-16"))) +
  ggtitle("Scaled Sales for Private Label and aggregate control") +
  theme(legend.position="bottom")
```



```
pre_cc_smu <- ols_smu %>%
  filter(week_end <= as.Date("2012-07-16"))

post_cc_smu <- ols_smu %>%
  filter(week_end > as.Date("2012-07-16"))
```

```
ggplot(ols_smu, aes(week_end, scaled_sales, col = factor(store_code_uc))) +
  geom_line() +
  geom_smooth(aes(week_end, scaled_sales), data = pre_cc_smu, method = 'lm',
se = FALSE) +
  geom_smooth(aes(week_end, scaled_sales), data = post_cc_smu, method = 'lm',
se = FALSE) +
  geom_vline(xintercept = as.numeric(as.Date("2012-07-16"))) +
  ggtitle("Scaled Sales for Validator (Smuckers) and aggregate control") +
  theme(legend.position="bottom")
```

