The impact of a newly implemented marketing strategy on the weekly sales levels at the treated store

Impact of the new marketing strategy on treated store's weekly sales

Objectives

The goal of this analysis is to evaluate the impact of a newly implemented marketing strategy on the weekly sales levels at Store 109 (the treated store), over the other stores' sales records within the brand. This evaluation will be conducted using several statistical methods to compare the effectiveness of the strategy, incorporating various internal and external factors that might influence sales outcomes.

Methodology

Analytical Techniques: Conducting 3 Analyses

1. **Simple Pre-Post Analysis:**
    - **Objective:** Assess the initial impacts of the new marketing strategy by comparing average sales before and after the implementation at Store 109.
    - **Procedure:** Calculate average sales for `p1sales` and `p2sales` before (`Post==0`) and after (`Post==1`) the strategy implementation. Use t-tests to assess if the changes in averages are statistically significant.
2. **Difference-in-Differences (DiD) Analysis:**
    - **Objective:** Examine the level difference of sales that comes from the marketing strategy by comparing the trend changes in sales at Store 109 to those at control stores.
    - **Procedure:**
        - Go through the visual inspection and formal test for checking the validity of common parallel trends assumptions.
        - Use a linear regression model: `sales ~ Post + W + Post*W` where `Post` is a binary indicator of the pre/post-treatment period, and `W` is a binary indicator differentiating between the treated store and control stores.
3. **Synthetic Control Method (SCM) Using the `Synth` Package:**
    - **Objective:** Construct a synthetic version of Store 109 using a weighted combination of control stores to create a robust counterfactual scenario, where we get an overview of the control stores that have similar trends with the treatment store.

- **Procedure:**
  - Identify predictors and weights: Use pre-treatment characteristics and sales data to determine the optimal weights for control stores that minimize the pre-treatment difference in sales between the treated store and the synthetic control.
  - Perform the analysis: Compare the post-treatment sales of Store 109 to its synthetic control to assess the impact.
  - Check the predictor balance after creating the synthetic control to ensure a good fit.

1. Pre-Post Analysis

```r
# /echo = False
library(dplyr)
```

```

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```r
library(tidyr)
library(ggplot2)
```

```
Warning: package 'ggplot2' was built under R version 4.3.3
```

```r
library(knitr)
library(kableExtra)
```

```

Attaching package: 'kableExtra'

The following object is masked from 'package:dplyr':

    group_rows
```

```r
# Load the data set
store_data <- read.csv("Store_data.csv", sep = ";")

# Filter the data for store 109
store_109_data <- filter(store_data, storeNum == 109)
```

```r
# Define the treatment week and year
treatment_week <- 26
treatment_year <- 2023
```

```r
# Split the data
pre_treatment_data <- filter(store_109_data, Year < treatment_year | (Year == treatment_year & We

post_treatment_data <- filter(store_109_data, Year > treatment_year | (Year == treatment_year & W

# Calculate statistics for pre and post
pre_treatment_stats <- pre_treatment_data %>% summarize(across(c(p1sales, p2sales), list(mean=mea

post_treatment_stats <- post_treatment_data %>% summarize(across(c(p1sales, p2sales), list(mean=m
```

```r
# Use kable to create an HTML table for pre-treatment statistics
kable(pre_treatment_stats, format = "html", caption = "Pre-Treatment Statistics") %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

Pre-Treatment Statistics

p1sales_mean	p1sales_sd	p1sales_min	p1sales_max	p2sales_mean	p2sales_sd	p2sales_min	p2sales_max
306.1299	11.51926	288	343	346.0909	23.77094	322	431

Findings on treatment effects

For product 1 (p1sales): the data shows a significant increase in sales after the new marketing strategy was introduced. This would imply that the strategy had a positive impact on product, p1. We can also visually see it happen after the treatment is implied on the graph.

For product 2 (p2sales): the data shows no sign of increase in sales for product p2. The trends remain the same after the new marketing strategy was introduced. The t-test displays a p-value of 0.284, which means there is no sign of evidence to conclude that the new implied strategy affected sales of product p2. Even with a slight increase of mean in sales for p2, it is not enough to conclude that the marketing strategy had an impact.

```r
# Use kable for post-treatment statistics as well
kable(post_treatment_stats, format = "html", caption = "Post-Treatment Statistics") %>%
  kable_styling(bootstrap_options = c("striped", "hover"))
```

Post-Treatment Statistics

p1sales_mean	p1sales_sd	p1sales_min	p1sales_max	p2sales_mean	p2sales_sd	p2sales_min	p2sales_max
332.1481	9.505884	322	361	353.5926	32.95832	327	420

```r
# Perform t-tests on the sales data for p1 and p2
t_test_p1 <- t.test(pre_treatment_data$p1sales, post_treatment_data$p1sales, var.equal = FALSE)
t_test_p2 <- t.test(pre_treatment_data$p2sales, post_treatment_data$p2sales, var.equal = FALSE)
```

```r
print(t_test_p1)
```

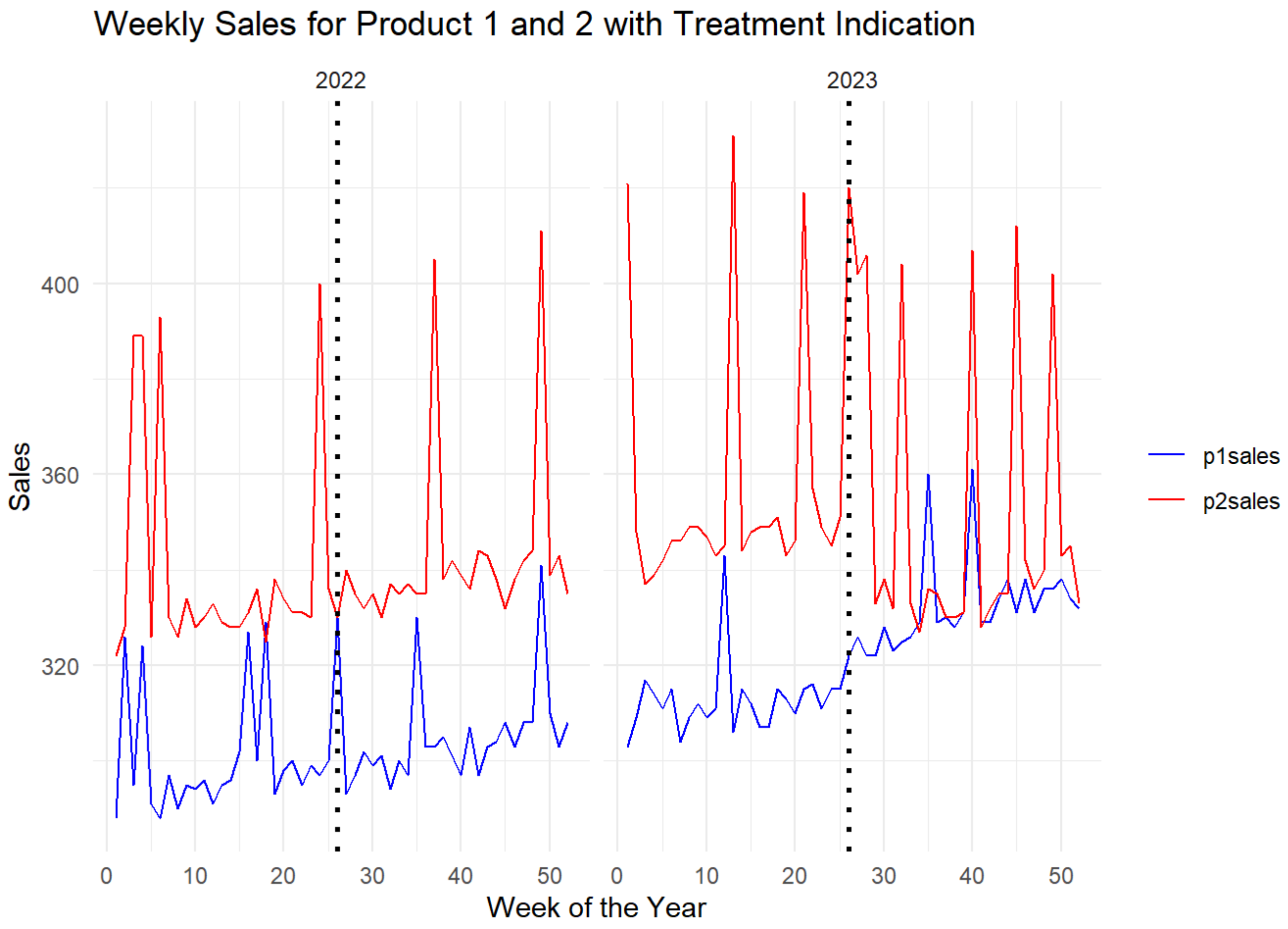
```

    Welch Two Sample t-test

data:  pre_treatment_data$p1sales and post_treatment_data$p1sales
t = -11.555, df = 54.707, p-value = 2.651e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -30.53127 -21.50528
sample estimates:
mean of x mean of y 
 306.1299  332.1481 
```

```r
print(t_test_p2)
```

```

    Welch Two Sample t-test

data:  pre_treatment_data$p2sales and post_treatment_data$p2sales
t = -1.0877, df = 35.941, p-value = 0.284
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -21.49044   6.48707
sample estimates:
mean of x mean of y 
 346.0909  353.5926 
```

```r
# Reshape the data to a long format for easier plotting with ggplot2
store_109_long <- store_109_data %>%
  pivot_longer(cols = c(p1sales, p2sales), names_to = "Product", values_to = "Sales")

# Create the plot with separate panels for each year
ggplot(data = store_109_long, aes(x = Week, y = Sales, group = Product, color = Product)) +
  geom_line() +
  geom_vline(xintercept = 26, linetype = "dotted", color = "black", size = 1) +
  facet_wrap(~Year, scales = 'free_x') +
  labs(title = "Weekly Sales for Product 1 and 2 with Treatment Indication", x = "Week of the Yea
  theme_minimal() +
  scale_color_manual(values = c("blue", "red")) +
  theme(legend.title = element_blank())
```

```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
ℹ Please use `linewidth` instead.
```



There could also be other reasons or confounders to why the sales may have gone up and could be factors such as economic trends, changes in consumer behavior or other promotions that may be happening at the same time.

For practical implications on a simple pre-post-analysis, the management might need to review the marketing strategy as to why it did not have an impact on product p2. Results might differ as we go deeper into the analysis and try a different approach by using DiD and Synthetic control analysis.

2. Difference-in-difference

```
# Load necessary libraries:
library(tidyverse)
```

```
Warning: package 'tidyverse' was built under R version 4.3.3
```

```
Warning: package 'readr' was built under R version 4.3.3
```

```
Warning: package 'forcats' was built under R version 4.3.3
```

```
Warning: package 'lubridate' was built under R version 4.3.3
```

```
── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
✔ forcats   1.0.0     ✔ readr     2.1.5
✔ lubridate 1.9.3     ✔ stringr   1.5.1
✔ purrr     1.0.2     ✔ tibble    3.2.1
── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
✖ dplyr::filter()          masks stats::filter()
✖ kableExtra::group_rows() masks dplyr::group_rows()
✖ dplyr::lag()             masks stats::lag()
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become
errors
```

```r
library(lubridate)
library(sjPlot)
```

```
Install package "strengejacke" from GitHub
(`devtools::install_github("strengejacke/strengejacke")`) to load all sj-packages at once!
```

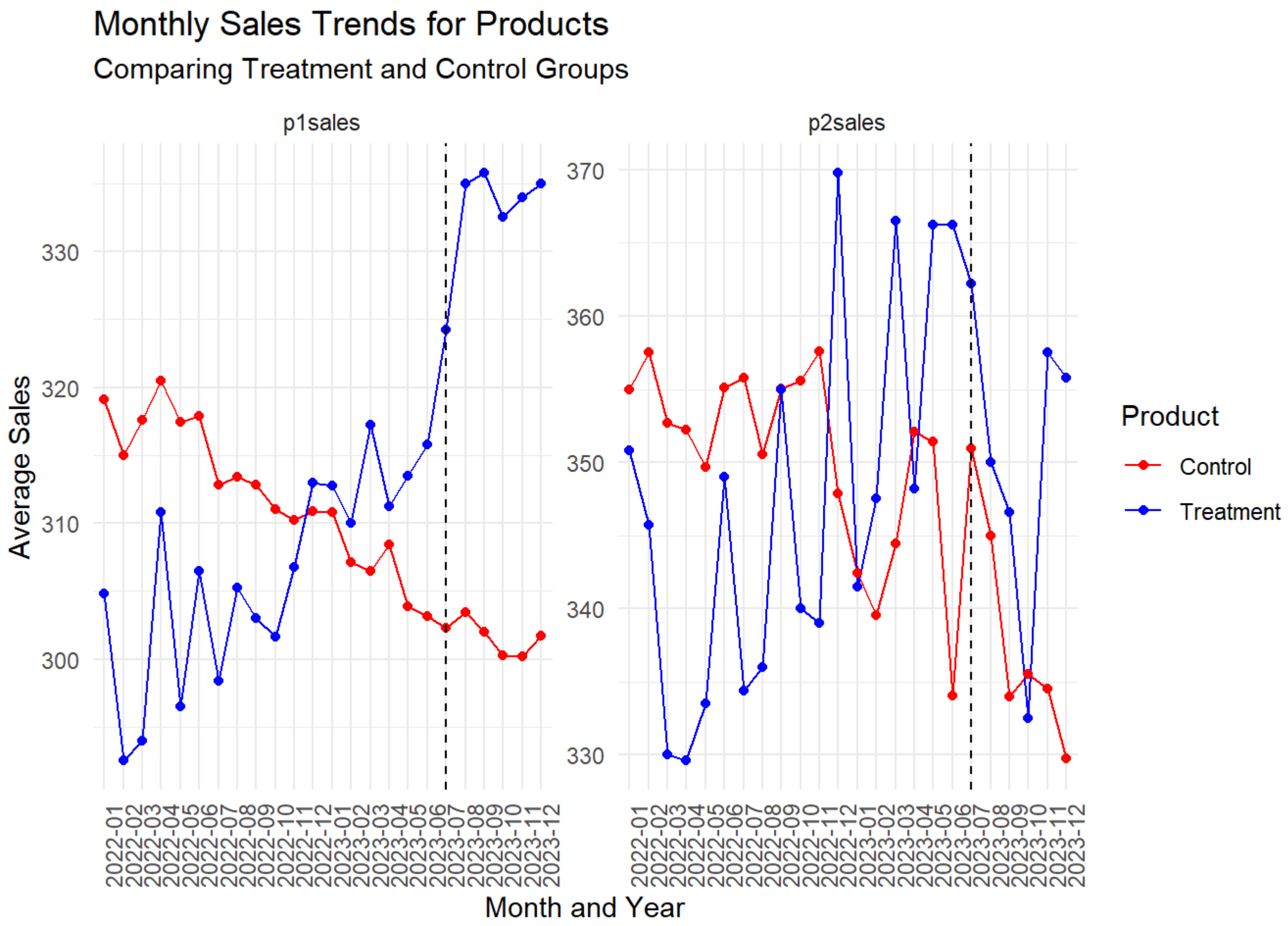
```r
# Read the dataset with the correct delimiter
store_data <- read.csv("Store_data.csv", sep = ";")

# Filter the data for store 109
store_109_data <- filter(store_data, store_data$storeNum == 109)

# Define the stored data for store 109 to be a daraframe:

store_109_data <- as.data.frame(store_109_data)
```

```r
# Filter control data (excluding Store 109)
control_data <- filter(store_data, store_data$storeNum != 109)

# Check the structure of the store 109 data:
str(store_109_data)
```

```
'data.frame':	104 obs. of  17 variables:
 $ storeNum : int  109 109 109 109 109 109 109 109 109 109 ...
 $ Year     : int  2022 2022 2022 2022 2022 2022 2022 2022 2022 2022 ...
 $ Week     : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Date     : chr  "2022-01-03" "2022-01-10" "2022-01-17" "2022-01-24" ...
 $ Weekind  : int  1 2 3 4 5 6 7 8 9 10 ...
 $ p1sales  : int  288 326 295 324 291 288 297 290 295 294 ...
 $ p2sales  : int  322 328 389 389 326 393 330 326 334 328 ...
 $ p1price  : num  2.19 2.19 2.29 2.29 2.79 2.49 2.79 2.79 2.99 2.79 ...
 $ p2price  : num  2.29 2.59 2.99 2.49 2.49 2.99 2.99 2.59 2.59 2.49 ...
 $ p1prom   : int  0 1 0 1 0 0 0 0 0 0 ...
 $ p2prom   : int  0 0 1 1 0 1 0 0 0 0 ...
 $ compind  : num  0.271 0.271 0.271 0.271 0.271 0.271 0.271 0.271 0.271 0.271 ...
 $ storesize: int  123 123 123 123 123 123 123 123 123 123 ...
 $ city     : chr  "STAV" "STAV" "STAV" "STAV" ...
 $ citysize : int  121610 121610 121610 121610 121610 121610 121610 121610 121610 121610 ...
 $ W        : int  1 1 1 1 1 1 1 1 1 1 ...
 $ Post     : int  0 0 0 0 0 0 0 0 0 0 ...
```

We can also have a look at monthly sales through 2022-2023 between two products, among treatment and control groups to observe the trends:

```
# Convert 'Weekind' to 'Date', then to 'MonthYear'
store_data <- store_data %>%
  mutate(
    Date = as.Date("2022-01-01") + (Weekind - 1) * 7,  # Create Date from 'weekind'
    MonthYear = format(Date, "%Y-%m")  # Create 'MonthYear' from 'Date'
  )

# Ensure 'MonthYear' is a factor and check if '2023-07' is a level (this step is for drawing the 

store_data$MonthYear <- factor(store_data$MonthYear)
if(!"2023-07" %in% levels(store_data$MonthYear)) {
  store_data$MonthYear <- factor(store_data$MonthYear, levels = c(levels(store_data$MonthYear), "
}



# Create 'Group' variable to categorize Treatment and Control
store_data <- store_data %>%
  mutate(Group = if_else(store_data$storeNum == 109, "Treatment", "Control"))

# Aggregate data by 'MonthYear' and 'Group'
monthly_sales <- store_data %>%
  group_by(MonthYear, Group) %>%
  summarise(
    Avg_p1sales = mean(p1sales, na.rm = TRUE),
    Avg_p2sales = mean(p2sales, na.rm = TRUE),
    .groups = 'drop'
  )

# Pivot data to long format for plotting
monthly_sales_long <- monthly_sales %>%
  pivot_longer(
    cols = c(Avg_p1sales, Avg_p2sales),
    names_to = "Product",
    values_to = "Sales",
    names_prefix = "Avg_"
  )

# Plot with a vertical dash line at '2023-07'

ggplot(monthly_sales_long, aes(x = MonthYear, y = Sales, color = Group, group = interaction(Produ
  geom_line() +
  geom_point() +
  facet_wrap(~Product, scales = "free_y") +
  scale_color_manual(values = c("Treatment" = "blue", "Control" = "red")) +
  labs(
    title = "Monthly Sales Trends for Products",
    subtitle = "Comparing Treatment and Control Groups",
    x = "Month and Year",
    y = "Average Sales",
    color = "Product"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  geom_vline(xintercept = which(levels(monthly_sales_long$MonthYear) == "2023-07"), linetype = "da
```



From the graph above, it is observed that the trends of both groups are following in the opposite direction, indicating that this setting might not meet the common parallel trends assumption, which is preliminary to applying Difference-in-Difference analysis. But let’s look at the formal test on their trends in the next part, to be more assertive.

Formal test on Common Parallel Trends Assumption:

We run two regressions on each treatment and control group on pre-treatment data only, to test their trends. Then check if there is a difference between the treatment and control groups’ trends to see if the difference is significant.

```
# Common or parallel trend assumptions:
# Run a linear regression on the pre-treatment data only
```

```r
# On product 1:

model1t <- lm(p1sales ~ Weekind, data = store_109_data[store_109_data$Post==0,])
tab_model(model1t)
```

	p1sales		
Predictors	*Estimates*	*CI*	*p*
(Intercept)	296.95	292.25 – 301.65	**<0.001**
Weekind	0.24	0.13 – 0.34	**<0.001**
Observations	78		
R^2 / R^2 adjusted	0.216 / 0.206		

```r
model1c <- lm(p1sales ~ Weekind, data = control_data[control_data$Post==0,])
tab_model(model1c)
```

	p1sales		
Predictors	*Estimates*	*CI*	*p*
(Intercept)	320.48	316.56 – 324.39	**<0.001**
Weekind	-0.21	-0.29 – -0.12	**<0.001**
Observations	1560		
R^2 / R^2 adjusted	0.014 / 0.014		

```r
# On product 2:

model2t <- lm(p2sales ~ Weekind, data = store_109_data[store_109_data$Post==0,])
tab_model(model2t)
```

	p2sales		
Predictors	*Estimates*	*CI*	*p*
(Intercept)	333.51	322.59 – 344.43	**<0.001**
Weekind	0.34	0.10 – 0.58	**0.006**
Observations	78		
R^2 / R^2 adjusted	0.096 / 0.084		

```r
model2c <- lm(p2sales ~ Weekind, data = control_data[control_data$Post==0,])
tab_model(model2c)
```

	p2sales		
Predictors	*Estimates*	*CI*	*p*
(Intercept)	357.35	350.57 – 364.13	**<0.001**
Weekind	-0.17	-0.32 – -0.02	**0.027**
Observations	1560		
R^2 / R^2 adjusted	0.003 / 0.003		

```r
# On treatment and time interaction, to see if the trend in sales over time differs between the tr

model3_p1 <- lm(p1sales ~ Weekind*W, data = store_data[store_data$Post==0,])
tab_model(model3_p1)
```

	p1sales		
Predictors	*Estimates*	*CI*	*p*
(Intercept)	320.48	316.65 – 324.31	**<0.001**
Weekind	-0.21	-0.29 – -0.12	**<0.001**
W	-23.53	-41.09 – -5.97	**0.009**
Weekind × W	0.45	0.06 – 0.83	**0.024**
Observations	1638		
R^2 / R^2 adjusted	0.016 / 0.014		

```r
model3_p2 <- lm(p2sales ~ Weekind*W, data = store_data[store_data$Post==0,])
tab_model(model3_p2)
```

	p2sales		
Predictors	*Estimates*	*CI*	*p*
(Intercept)	357.35	350.71 – 363.99	**<0.001**
Weekind	-0.17	-0.31 – -0.02	**0.024**
W	-23.84	-54.27 – 6.58	0.124

Weekind × W	0.51	-0.16 – 1.18	0.134
Observations	1638		
R^2 / R^2 adjusted	0.004 / 0.002		

The ‘Weekind x W’ interaction term is not statistically significant (p-value=0.134) suggesting that the slope of ‘Weekind’ does not differ by group, and it supports the parallel trends assumption. However, in models 1t&1c, 2t&2c, both groups have similar slopes for both products but differ for different groups. While the treatment store has positive slopes, negative slopes have been shown for the control stores. By this point, the primary assumption of parallel trends for both groups is violated, therefore difference-in-difference method should not be considered in this case, as the result from this method would not be valid.

Check the Difference-in-Difference estimates’ result:

We estimate the ATET (average treatment effect on the treated) in the diff-in-diff from the following regression model $Sales_{it} = \alpha + \beta_1 W_i + \beta_2 POST_t + \boldsymbol{\beta_3} W_i * POST_t$ where β_3 represents the $ATET_{DID}$
.

```
# On product 1
model3a <- lm(p1sales ~ W*Post, data = store_data)
tab_model(model3a)
```

	p1sales		
Predictors	*Estimates*	*CI*	*p*
(Intercept)	312.26	310.25 – 314.27	**<0.001**
W	-5.93	-15.14 – 3.29	0.207
Post	-10.58	-14.60 – -6.55	**<0.001**
W × Post	36.78	18.35 – 55.21	**<0.001**
Observations	2184		
R^2 / R^2 adjusted	0.016 / 0.015		

```
# On product 2
model4a <- lm(p2sales ~ W*Post, data = store_data)
tab_model(model4a)
```

	p2sales		
Predictors	*Estimates*	*CI*	*p*
(Intercept)	350.68	347.38 – 353.98	**<0.001**

W	-3.64	-18.75 – 11.47	0.636
Post	-12.08	-18.67 – -5.48	**<0.001**
W × Post	16.08	-14.14 – 46.29	0.297
Observations	2184		
R^2 / R^2 adjusted	0.006 / 0.005		

For product 1, the coefficient of treatment assignment ‘W’ is negative (-5.93) indicating that the treatment group had ~6 fewer sales than the control group, on average, in the pre-treatment period, but its p-value is insignificant. ‘Post’ coefficient is also negative (-10.58) showing an effect in the decrease in sales for the control group. The interaction term ‘W x Post’ coefficient is the DiD estimator, is positive (36.78) meaning a lift of 36.78 difference in sales after the treatment on the treated group. But the R-squared is pretty low (1.6%), which means that the model is only able to explain 1.6% variability of the sales, which might suggest that the other factors have a bigger impact on influencing the sales.

A similar pattern was found in product 2’s estimate, the coefficient of the DiD estimator (16.08) has a positive influence on sales, but its CI [-14.14 ~46.29] and its p-value is 0.297, clearly proving this statistically insignificant. A relatively low R-square (0.006) also adds weight to the hypothesis that the sales were not influenced predominantly by the treatment.

According to the results above, the D-i-D estimator is not performing well to prove the impact on sales before and after treatment. The reason could be because the control group is a combination of the rest of the stores with variation in sales, where their trends were averaged, versus store 109, which is not comparable in terms of scaling.

3. Synthetic Control

In this analysis I will be using synthetic control analysis to assess if the new marketing strategy on sales at store 109 has an impact.

```r
library(Synth)
```

```
Warning: package 'Synth' was built under R version 4.3.3
```

```
##
## Synth Package: Implements Synthetic Control Methods.
```

```
## See https://web.stanford.edu/~jhain/synthpage.html for additional information.
```

```r
library(plm)
```

```
Warning: package 'plm' was built under R version 4.3.3
```

```

Attaching package: 'plm'
```

```
The following objects are masked from 'package:dplyr':

    between, lag, lead
```

```r
library(ggplot2)
library(corrplot)
```

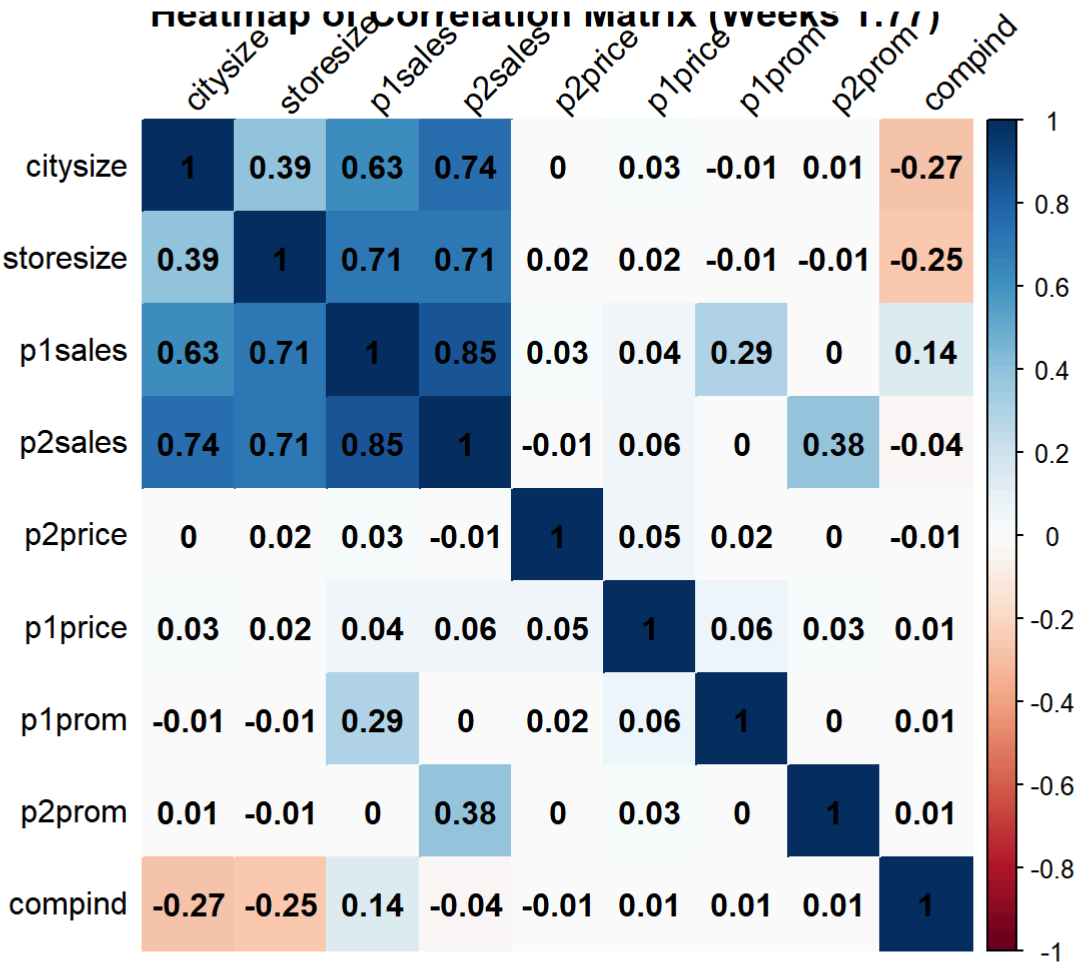
```
Warning: package 'corrplot' was built under R version 4.3.3
```

```
corrplot 0.92 loaded
```

```r
store_data <- read.csv("Store_data.csv", sep = ";")
```

```r
store_filtered <- subset(store_data, Weekind >= 1 & Weekind <= 77)
# Select only the required columns for the heatmap
store_selected <- store_filtered[, c('p1sales', 'p2sales', 'p1price', 'p2price', 'p1prom', 'p2pro
```

```r
# Calculate the correlation matrix
corr_matrix <- cor(store_selected, use = "complete.obs")

# Generate the heatmap
corrplot(corr_matrix, method = "color", order = "hclust",
         tl.col = "black", tl.srt = 45, addCoef.col = "black",
         title = "Heatmap of Correlation Matrix (Weeks 1:77)")
```



	citysize	storesize	p1sales	p2sales	p2price	p1price	p1prom	p2prom	compind
citysize	1	0.39	0.63	0.74	0	0.03	-0.01	0.01	-0.27
storesize	0.39	1	0.71	0.71	0.02	0.02	-0.01	-0.01	-0.25
p1sales	0.63	0.71	1	0.85	0.03	0.04	0.29	0	0.14
p2sales	0.74	0.71	0.85	1	-0.01	0.06	0	0.38	-0.04
p2price	0	0.02	0.03	-0.01	1	0.05	0.02	0	-0.01
p1price	0.03	0.02	0.04	0.06	0.05	1	0.06	0.03	0.01
p1prom	-0.01	-0.01	0.29	0	0.02	0.06	1	0	0.01
p2prom	0.01	-0.01	0	0.38	0	0.03	0	1	0.01
compind	-0.27	-0.25	0.14	-0.04	-0.01	0.01	0.01	0.01	1

Here we can see which columns will be the predictors for p1sales and p2sales

```
treated_unit <- 109
control_units <- setdiff(unique(store_data$storeNum), treated_unit)
```

```
dependent_variable <- "p1sales"

store_data$storeNum <- as.numeric(store_data$storeNum)
store_data <- data.frame(store_data)
store_data$storeNum_char <- as.character(store_data$storeNum)
```

```
dataprep_out_p1 <- dataprep(
  foo = store_data,
  predictors = c("p2sales", "p1prom","p2prom","citysize", "storesize"),
  dependent = "p1sales",
  unit.variable = "storeNum",
  time.variable = "Weekind",
  treatment.identifier = treated_unit,
  controls.identifier = control_units,
  time.predictors.prior = c(1:77),
  special.predictors = list(
    list("p1sales", 34:64, "mean"),
    list("p1sales", 1, "mean"),
    list("p1sales", 63, "mean")),
  time.optimize.ssr = c(1:77),
  unit.names.variable = "storeNum_char",
  time.plot = c(1:104)
)
```

```
synth_out_p1 <- synth(data.prep.obj = dataprep_out_p1)
```

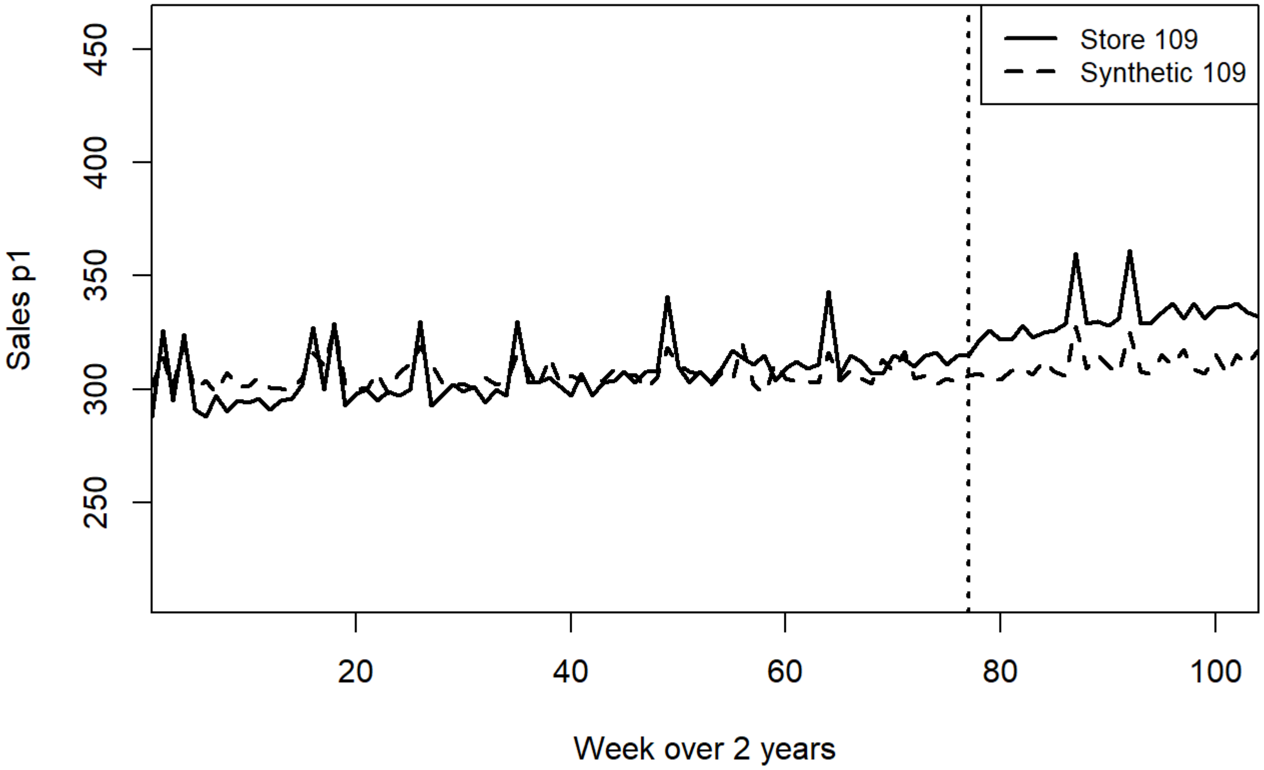
```
X1, X0, Z1, Z0 all come directly from dataprep object.


****************
 searching for synthetic control unit


****************
****************
****************

MSPE (LOSS V): 79.32724

solution.v:
 0.3451499 0.03861501 0.0006204239 0.1178336 0.02261248 0.1375817 0.2510741 0.08651274

solution.w:
 6.33e-08 0.1221107 1.199e-07 6.8e-08 1.504e-07 2.687e-07 5.15e-08 1.208e-07 0.5327373 4.883e-07
1.6472e-06 3.5767e-06 4.308e-07 0.2579961 4.871e-07 1.843e-07 2.035e-07 1.677e-07 1.63e-07
0.08714767
```

```
print(synth.tab(dataprep.res = dataprep_out_p1, synth.res = synth_out_p1))
```

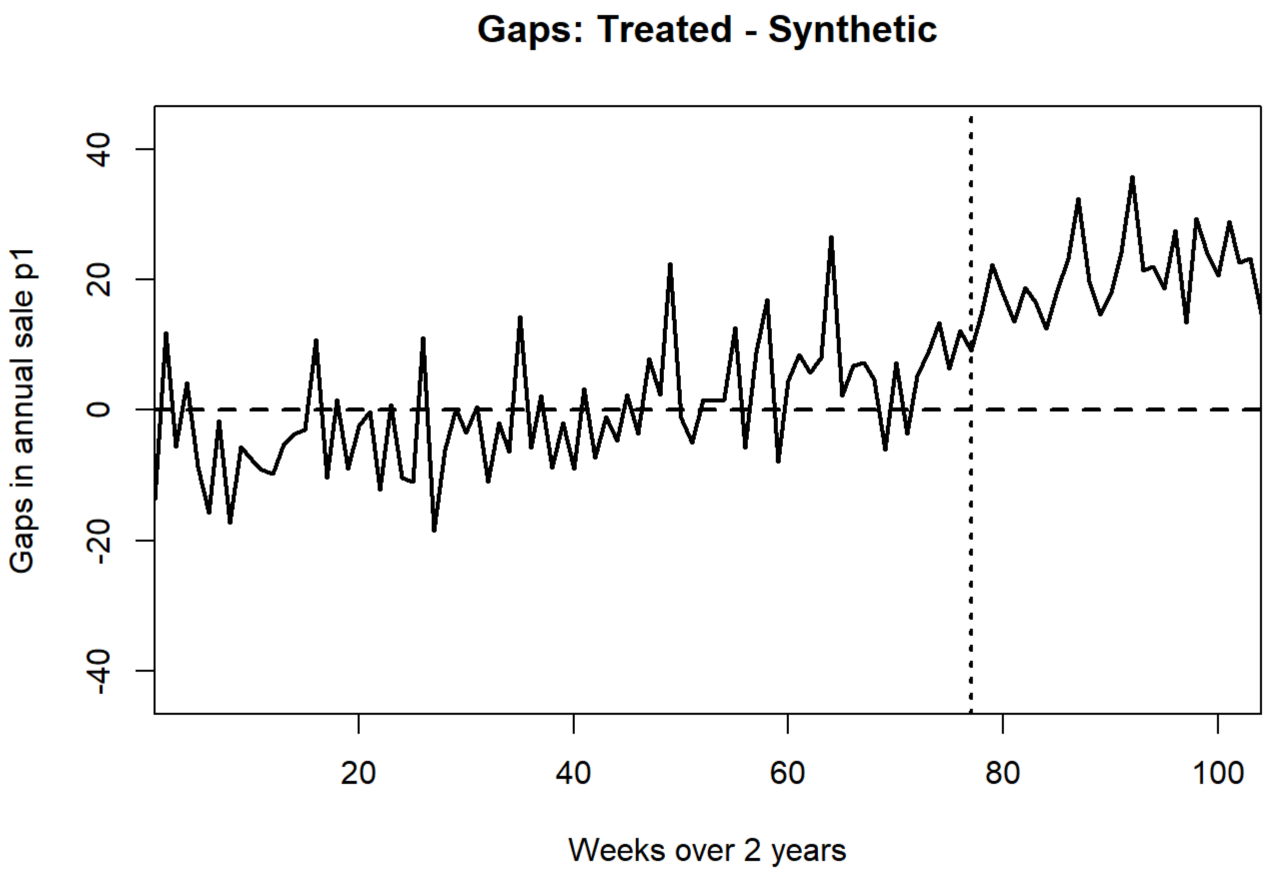
```
$tab.pred
                         Treated  Synthetic Sample Mean
p2sales                  346.091    333.629     350.850
p1prom                     0.104      0.133       0.175
p2prom                     0.117      0.156       0.166
citysize              121610.000 163587.223  213992.350
storesize                123.000    106.643     118.250
special.p1sales.34.64    309.452    306.793     310.205
special.p1sales.1        288.000    301.687     320.500
special.p1sales.63       311.000    302.978     309.650

$tab.v
                      v.weights
p2sales               0.345
p1prom                0.039
p2prom                0.001
citysize              0.118
```

```
storesize             0.023
special.p1sales.34.64 0.138
special.p1sales.1     0.251
special.p1sales.63    0.087

$tab.w
    w.weights unit.names unit.numbers
101     0.000        101          101
102     0.122        102          102
103     0.000        103          103
104     0.000        104          104
105     0.000        105          105
106     0.000        106          106
107     0.000        107          107
108     0.000        108          108
110     0.533        110          110
111     0.000        111          111
112     0.000        112          112
113     0.000        113          113
114     0.000        114          114
115     0.258        115          115
116     0.000        116          116
117     0.000        117          117
118     0.000        118          118
119     0.000        119          119
120     0.000        120          120
121     0.087        121          121

$tab.loss
         Loss W   Loss V
[1,] 0.08605624 79.32724
```

```
# Path plot to visualize the fit
path.plot(synth_out_p1,
          dataprep.res = dataprep_out_p1,
          Ylab = "Sales p1",
          Xlab = "Week over 2 years",
          Legend = c("Store 109","Synthetic 109"),
          tr.intake = 77)
```



```
# Computing the treatment effect
gaps.plot(synth_out_p1,
          dataprep.res = dataprep_out_p1,
          Ylab = "Gaps in annual sale p1",
          Xlab = "Weeks over 2 years",
          tr.intake = 77)
```



```
dataprep_out_p2 <- dataprep(
  foo = store_data,
  predictors = c( "p1sales", "p2prom"),
  dependent = "p2sales",
  unit.variable = "storeNum",
  time.variable = "Weekind",
  treatment.identifier = treated_unit,
  controls.identifier = control_units,
  time.predictors.prior = c(1:77),
  special.predictors = list(
    list("p2sales", 34:50, "mean"),
    list("p2sales", 22:33, "mean"),
    list("p2sales", 63, "mean")),
  time.optimize.ssr = c(1:77),
  unit.names.variable = "storeNum_char",
  time.plot = c(1:104)
)
```

```
synth_out_p2 <- synth(data.prep.obj = dataprep_out_p2)
```

X1, X0, Z1, Z0 all come directly from dataprep object.

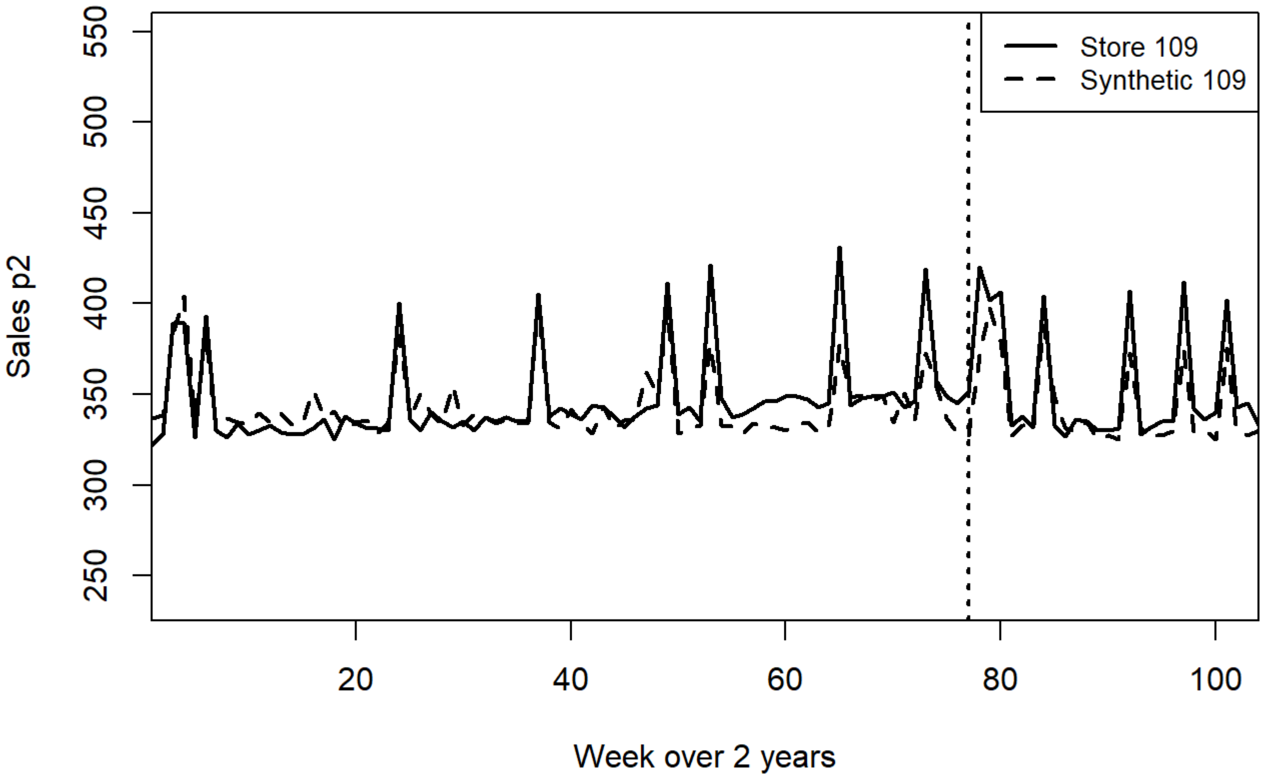
```
****************
 searching for synthetic control unit


****************
****************
****************

MSPE (LOSS V): 195.3678

solution.v:
 0.03859132 0.007291292 0.5744152 0.3788705 0.0008316703

solution.w:
 2.02649e-05 0.188446 1.14826e-05 1.66846e-05 2.7637e-05 1.17592e-05 7.32403e-05 2.93397e-05
0.7965401 9.691e-07 8.1e-09 1.56725e-05 0.0002514441 0.01094223 0.003448618 9.4999e-06 8.4088e-06
7.01302e-05 3.7695e-05 3.87529e-05
```

```r
print(synth.tab(dataprep.res = dataprep_out_p2, synth.res = synth_out_p2))
```

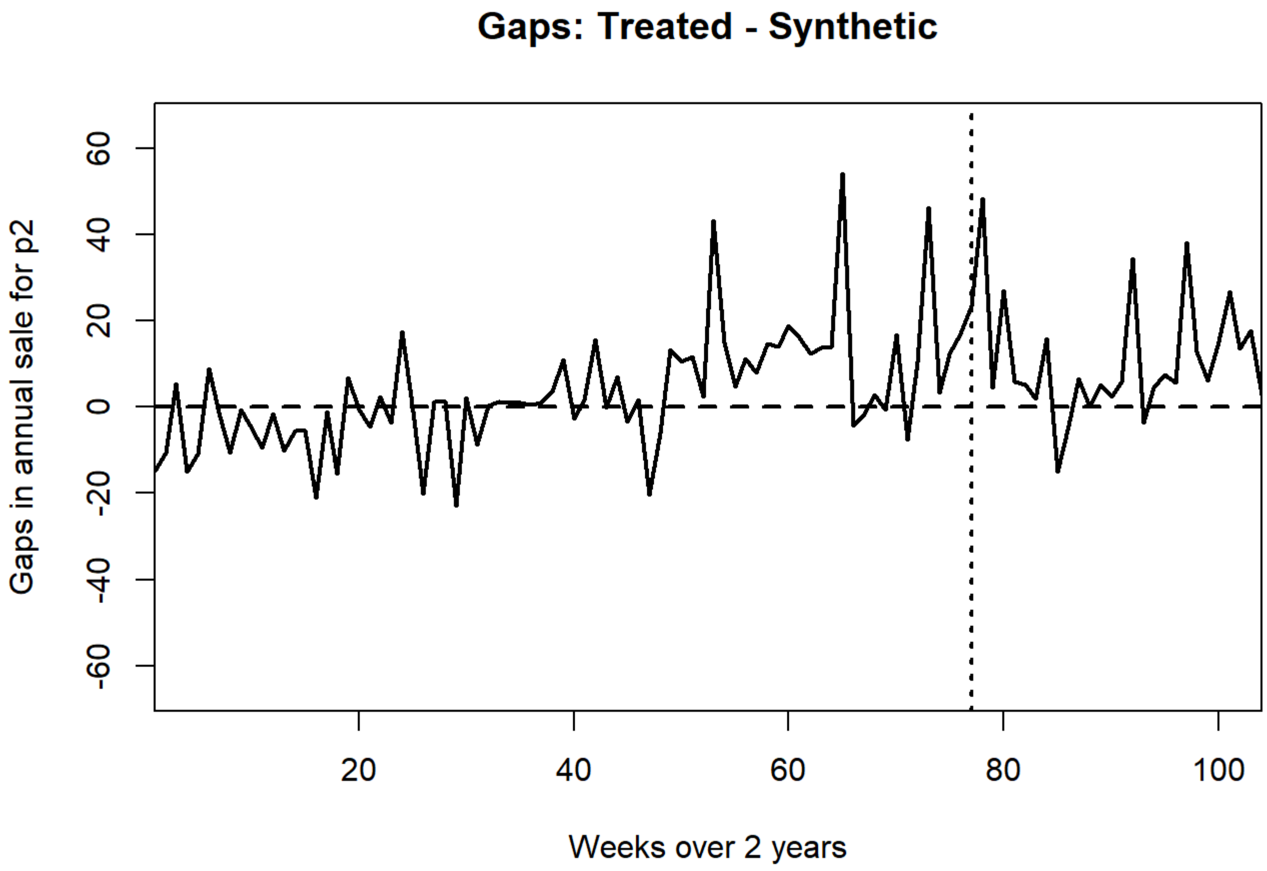
```
$tab.pred
                      Treated Synthetic Sample Mean
p1sales               306.130   308.886     312.418
p2prom                  0.117     0.130       0.166
special.p2sales.34.50 346.941   344.896     354.759
special.p2sales.22.33 339.250   341.770     353.638
special.p2sales.63    343.000   329.188     344.100

$tab.v
                      v.weights
p1sales               0.039
p2prom                0.007
special.p2sales.34.50 0.574
special.p2sales.22.33 0.379
special.p2sales.63    0.001

$tab.w
    w.weights unit.names unit.numbers
101     0.000        101          101
102     0.188        102          102
103     0.000        103          103
104     0.000        104          104
105     0.000        105          105
106     0.000        106          106
107     0.000        107          107
108     0.000        108          108
110     0.797        110          110
111     0.000        111          111
112     0.000        112          112
```

```
113      0.000        113          113
114      0.000        114          114
115      0.011        115          115
116      0.003        116          116
117      0.000        117          117
118      0.000        118          118
119      0.000        119          119
120      0.000        120          120
121      0.000        121          121

$tab.loss
          Loss W   Loss V
[1,] 0.002699052 195.3678
```

```
# Path plot to visualize the fit
path.plot(synth_out_p2,
          dataprep.res = dataprep_out_p2,
          Ylab = "Sales p2",
          Xlab = "Week over 2 years",
          Legend = c("Store 109","Synthetic 109"),
          tr.intake = 77)
```



```
# Computing the treatment effect
gaps.plot(synth_out_p2,
          dataprep.res = dataprep_out_p2,
          Ylab = "Gaps in annual sale for p2",
          Xlab = "Weeks over 2 years",
          tr.intake = 77)
```



Interpretation and Findings

After week 78, which is week 26 of the second year, there's a clear jump in sales for product p1 in Store 109. This shows that the marketing strategy might be working. We've used data that closely matches the sales of p1 before the strategy started, and we got an MSPE of 79. This number helps us trust that our synthetic 'comparison' store reflects what could have happened without the new marketing. When we look at the charts, it's clear that Store 109 is doing better in selling p1 compared to the synthetic version. This is good news—it looks like the marketing strategy has a positive effect. The tables tell us that the sales of product p2 are really important in predicting how well p1 will sell.

For product 2, things look a bit different. The actual sales at Store 109 don't seem to go up like they did for p1, and the charts don't show a clear boost after the new strategy began. Also, the MSPE is higher for p2, meaning that our synthetic control might not be as accurate here. We can also see some spike in volatility in the 2nd graph of p2 indicating that it might be influenced by factors not captured by synthetic control. These things could be seasonal effects, sporadic promotions or other unobserved variables.

Results & Discussion

Challenges in the Data

The analysis confronts a relatively short post-treatment period that may not adequately capture the long-term effects of the marketing strategy. There is also a dependency on data from other stores to generate the synthetic control, premised on the assumption that these stores share comparable attributes with Store 109. For product 2, the higher MSPE indicates potential issues in capturing pretreatment characteristics as effectively as for product 1.

Discussion

The marketing strategy appears effective for product 1, as evidenced by the sales uplift in the post-treatment phase. This effect can be quantified by examining the average gap between the treated and synthetic sales post-treatment. It remains essential to monitor the strategy's impact over a more extended period to ensure sustained effectiveness and to inform any necessary strategy adjustments. For product 2, the absence of a post-treatment uplift and a higher MSPE calls for a more nuanced approach, possibly involving a re-examination of the model or the inclusion of additional data and variables to improve fit and predictive accuracy. We can therefore conclude or recommend to find a new marketing strategy that would also work for the sales of product 2. But as well figure out what is working for product 1 in terms of this marketing strategy.

Comparison of three analyses:

The simple pre-post barely analyses the sales data of the pre and post-treatment for the treated store within a 2-year period, where a significant increase in sales of product 1 can be instantly seen from the graph, while product 2 remains the same in sales level. Even though it is a straightforward approach, it lacks a causal framework, making it a naive method as it did not control the external variables which could influence the sales.

Difference-in-difference is supposed to show the difference in sales level after the treatment, meaningfully explaining the causal effect of the new marketing strategy, is not considered an appropriate method since the preliminary assumption of its valid usage was not proved.

Synthetic control predicts the sales by using the control group without knowing which of the stores is the better fit for the treatment group. The relevant predictor was chosen by checking the correlation with the sales of the products. This method also reduces the influence of confounding variables, for its data comes from several units of the control group. Moreover, it has an improved causal inference by explicitly modelling the relationship between the treated units and the weighted combination of control units that best replicates the pre-treatment characteristics of the treated units. Allows more flexibility in selecting control units and weighing them to best match the treated units' pre-treatment characteristics.

The effectiveness of the new strategy

By conducting these three analyses and inspecting their appropriateness in refining the effectiveness of the new strategy, we can conclude that it has an impact on product 1, but not on product 2. The synthetic control analysis results show the increasing gap in sales of the treated store, in comparison to its synthetic control to prove the causal impact of the treatment. On the other hand, this method did not predict well in product 2 for the mean square prediction error (MSPE) was high. Hence, the treatment effect applied to product 2 was not explained by this method, statistically, as well as shown on some graphs that the improvement of product 2's sales was not considerable.

The expected sales uplift in average can be extracted from the simple pre-post analysis:

- Product 1: p1sales_mean post - p1sales_mean pre = 332.1 - 306.1 = 26

- Product 2: p2sales_mean post - p2sales_mean pre = 353.6 - 346.1 = 7.5

Conclusions

Recommendations for the marketing management:

- Focus on improving product 2 sales, by re-targeting and segmenting the customers for better tailoring the marketing approaches.

- Extend the post-treatment period to have a longer post-treatment period, to capture the longer-term effects of the strategy, especially on product 2, for better observing the significance of this marketing strategy.

- Extend the analysis framework for exploring other analytical techniques. For instance, considering the possibility of bias in DiD method and its inappropriate implementation, matching could be integrated to enhance the robustness of causal inference, as we encountered the preliminary assumption of DiD was not satisfied.

- Continuous monitoring to track sales performance and record real-time changes and feedback, as well as identify opportunities for optimizing sales, especially in a dynamic market environment.