MIS7420

Seminar in Management Information Systems:

Paper Replication with R

Prof. Zhiqiang Zheng, Prof. Vijay Mookerjee

Author: Yihong Liu,

NETID: YXL180111,

Contents

List of Figures

List of Tables

List of Codes

1 Data Cleaning Process

```r
# load library
library('dplyr')
library('haven')
library('sqldf')
library('zoo')
library('plm')
library('stargazer')

# all data path
bb_zipcode_path <- 'data/bestbuyzipcodes_sample.sas7bdat'
sales_allother_zipcode_path <- 'data/sales_allotherzipcode_sample.sas7bdat'
sales_cc_0mile_path <- 'data/sales_ccity0milezipcode_sample.sas7bdat'
sales_cc_5miles_path <- 'data/sales_ccity5milezipcode_sample.sas7bdat'

# load data
bb_zipcode <- read_sas(bb_zipcode_path)
sales_allother_zipcode <- read_sas(sales_allother_zipcode_path)
sales_cc_0mile <- read_sas(sales_cc_0mile_path)
sales_cc_5miles <- read_sas(sales_cc_5miles_path)

# Data Mapping
sales_allother_zipcode$Store_Close_Status <- 0 # NaN means no CC in 5-miles radius, we change NaN to 0

# Exclude Data without purchase
# All data should be with purchase -> tran_flg == 1
sales_allother_zipcode <- sales_allother_zipcode[sales_allother_zipcode$tran_flg == 1,]
sales_cc_0mile <- sales_cc_0mile[sales_cc_0mile$tran_flg == 1,]
sales_cc_5miles <- sales_cc_5miles[sales_cc_5miles$tran_flg == 1,]

# Filter Referring Domain

# groupby ref_domain and count
groupby_ref_domain_result <- aggregate(machine_id ~ ref_domain_name, rbind(sales_allother_zipcode, sales_cc_0mile, sales_cc_5miles), FUN = "length"
      )
groupby_ref_domain_result <- groupby_ref_domain_result[order(-groupby_ref_domain_result$machine_id), ]
# we identify some search engines
search_engine_to_consider1 <- c("GOOGLE.COM", "YAHOO.COM", "google.com", "yahoo.com",
                            "MSN.COM", "msn.com", "aol.com", "AOL.COM", "LIVE.COM", "live.com",
                            "MYWEBSEARCH.COM", "ASK.COM", "MYWAY.COM", "mywebsearch.com",
                            "ask.com", "YAHOO.NET", "BIZRATE.COM", "bizrate.com",
                            "amazon.com", "staples.com", "dell.com", "walmart.com", "bestbuy.com",
                            "AMAZON.COM", "STAPLES.COM", "DELL.COM", "WALMART.COM", "BESTBUY.COM")

search_engine_to_consider2 <- c("GOOGLE.COM", "YAHOO.COM", "BING.COM", "google.com", "yahoo.com", "bing.com")

ref_domain_to_consider1 <- c("", "GOOGLE.COM", "YAHOO.COM", "google.com", "yahoo.com",
                            "MSN.COM", "msn.com", "aol.com", "AOL.COM", "LIVE.COM", "live.com",
                            "MYWEBSEARCH.COM", "ASK.COM", "MYWAY.COM", "mywebsearch.com",
                            "ask.com", "YAHOO.NET", "BIZRATE.COM", "bizrate.com",
                            "amazon.com", "staples.com", "dell.com", "walmart.com", "bestbuy.com",
                            "AMAZON.COM", "STAPLES.COM", "DELL.COM", "WALMART.COM", "BESTBUY.COM")

ref_domain_to_consider2 <- c("", "GOOGLE.COM", "YAHOO.COM", "BING.COM", "google.com", "yahoo.com", "bing.com")

# Then we filter data by refer domain name
sales_allother_zipcode <- sales_allother_zipcode[(sales_allother_zipcode$ref_domain_name %in% ref_domain_to_consider1),]
sales_cc_0mile <- sales_cc_0mile[(sales_cc_0mile$ref_domain_name %in% ref_domain_to_consider1),]
sales_cc_5miles <- sales_cc_5miles[(sales_cc_5miles$ref_domain_name %in% ref_domain_to_consider1),]

# Filter Target Domain Name
groupby_target_domain_result <- aggregate(machine_id ~ domain_name, rbind(sales_allother_zipcode, sales_cc_5miles), FUN = "length")
groupby_target_domain_result <- groupby_target_domain_result[order(-groupby_target_domain_result$machine_id), ]
five_target_domain_to_consider <- c("amazon.com", "staples.com", "dell.com", "walmart.com", "bestbuy.com")
two_target_domain_to_consider <- c("amazon.com","bestbuy.com")

# we can choose what filter to apply
sales_allother_zipcode <- sales_allother_zipcode[sales_allother_zipcode$domain_name %in% five_target_domain_to_consider,]
sales_cc_0mile <- sales_cc_0mile[sales_cc_0mile$domain_name %in% five_target_domain_to_consider,]
sales_cc_5miles <- sales_cc_5miles[sales_cc_5miles$domain_name %in% five_target_domain_to_consider,]
```

```r

# Product Categories
# 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
# Jay removed 28, 30, 39, 40
# We choose to remove 38 39 40
sort(unique(rbind(sales_allother_zipcode, sales_cc_0mile, sales_cc_5miles)$prod_category_id))
category_to_consider <- c(22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37)
experience_product <- c(24, 25, 26, 27, 28, 31, 32, 33, 34, 36, 37)
search_product <- c(22, 23, 24, 29, 30, 35)

sales_allother_zipcode <- sales_allother_zipcode[sales_allother_zipcode$prod_category_id %in% category_to_consider,]
sales_cc_0mile <- sales_cc_0mile[sales_cc_0mile$prod_category_id %in% category_to_consider,]
sales_cc_5miles <- sales_cc_5miles[sales_cc_5miles$prod_category_id %in% category_to_consider,]

# Date Transform
sales_allother_zipcode$event_date <- as.Date(sales_allother_zipcode$event_date)
sales_cc_0mile$event_date <- as.Date(sales_cc_0mile$event_date)
sales_cc_5miles$event_date <- as.Date(sales_cc_5miles$event_date)

# construct MonthYear - month of year
sales_allother_zipcode$MonthYear <- format(sales_allother_zipcode$event_date, "%Y-%m")
sales_cc_0mile$MonthYear <- format(sales_cc_0mile$event_date, "%Y-%m")
sales_cc_5miles$MonthYear <- format(sales_cc_5miles$event_date, "%Y-%m")

# Mark CC Closure

# CCStorePresent
# it is the same as Store_Close_Status
sales_allother_zipcode$CCStorePresent <- sales_allother_zipcode$Store_Close_Status
sales_cc_0mile$CCStorePresent <- sales_cc_0mile$Store_Close_Status
sales_cc_5miles$CCStorePresent <- sales_cc_5miles$Store_Close_Status

# AfterStoreClosing
sales_allother_zipcode$AfterStoreClosing <- ifelse(sales_allother_zipcode$MonthYear < "2008-11", 0, 1)
sales_cc_0mile$AfterStoreClosing <- ifelse(sales_cc_0mile$MonthYear < "2008-11", 0, 1)
sales_cc_5miles$AfterStoreClosing <- ifelse(sales_cc_5miles$MonthYear < "2008-11", 0, 1)

# BBStorePresent
sales_allother_zipcode <- merge(sales_allother_zipcode, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)
sales_cc_0mile <- merge(sales_cc_0mile, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)
sales_cc_5miles <- merge(sales_cc_5miles, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)

sales_allother_zipcode$BBStorePresent <- na.fill(sales_allother_zipcode$BB_Store_Status, 0)
sales_cc_0mile$BBStorePresent <- na.fill(sales_cc_0mile$BB_Store_Status, 0)
sales_cc_5miles$BBStorePresent <- na.fill(sales_cc_5miles$BB_Store_Status, 0)

# Mark Referring Domain
# Question: How to group data?
sales_allother_zipcode$NoReferringDomain <- ifelse(sales_allother_zipcode$ref_domain_name == "", 1, 0)
sales_cc_0mile$NoReferringDomain <- ifelse(sales_cc_0mile$ref_domain_name == "", 1, 0)
sales_cc_5miles$NoReferringDomain <- ifelse(sales_cc_5miles$ref_domain_name == "", 1, 0)

sales_allother_zipcode$ReferringDomainIsSearchEngine <- ifelse(sales_allother_zipcode$ref_domain_name %in% search_engine_to_consider1, 1, 0)
sales_cc_0mile$ReferringDomainIsSearchEngine <- ifelse(sales_cc_0mile$ref_domain_name %in% search_engine_to_consider1, 1, 0)
sales_cc_5miles$ReferringDomainIsSearchEngine <- ifelse(sales_cc_5miles$ref_domain_name %in% search_engine_to_consider1, 1, 0)

# Aggregate Data
concat_data1 <- rbind(sales_allother_zipcode, sales_cc_0mile)
concat_data2 <- rbind(sales_allother_zipcode, sales_cc_5miles)
concat_data1_exp <- concat_data1[concat_data1$prod_category_id %in% experience_product, ]
concat_data1_search <- concat_data1[concat_data1$prod_category_id %in% search_product, ]
concat_data2_exp <- concat_data2[concat_data2$prod_category_id %in% experience_product, ]
concat_data2_search <- concat_data2[concat_data2$prod_category_id %in% search_product, ]
```

Code 1: Data Preprocess

2 Paper Replication

In this section, we provide our replication for this paper. Names for subsections correspond to the tables in the published paper.

2.1 Table 1

Table 1 shows the summary statistics of top five vendors by sales volume.

Table 1: Summary Statistics of Top Five Vendors by Sales Volume

Domain Name	Total Transactions	Total Sales	Total Pages Viewed	Pages Per Dollar	Total Duration	Mins Per Dollar
dell.com	1,620	483,703.300	66,953	0.138	57,225.660	0.118
amazon.com	10,904	354,573.300	464,383	1.310	369,227.900	1.041
staples.com	5,927	236,982.300	247,163	1.043	166,189.900	0.701
walmart.com	1,977	156,606.100	80,397	0.513	68,434.890	0.437
bestbuy.com	1,230	149,950.400	50,627	0.338	36,735.900	0.245

Codes for generating Table 1 are listed below.

```
# Table 1
table1_raw <-  rbind(read_sas(sales_allother_zipcode_path), read_sas(sales_cc_0mile_path))
table1 <- sqldf("SELECT domain_name as DomainName, count(*) as TotalTransaction, SUM(prod_totprice) AS TotalSales, SUM(pages_viewed) AS
      TotalPagesViewed, SUM(pages_viewed)/SUM(prod_totprice) AS PagesPerDollar, SUM(duration) AS TotalDuration, SUM(duration)/SUM(prod_totprice) AS
       MinsPerDollar FROM table1_raw GROUP BY domain_name ORDER BY TotalSales DESC")
stargazer(table1[1:5,], align=TRUE, summary = FALSE, rownames = FALSE, title="Summary Statistics of Top Five Vendors by Sales Volume")
```

Code 2: Table 1 Generation

2.2 Table 2

Table 2 summarizes the frequency of referral channels for various online retailers.

Table 2: Summary Statistics of Referring Domain Categories

Domain Name	Total Transactions	Referred by Search Engine	Direct to Website	Referred by Others
amazon.com	10,904	2,955(27.1%)	7,018(64.4%)	931(8.6%)
bestbuy.com	1,230	258(21.0%)	901(73.3%)	71(5.8%)
All Others	36,794	6,999(19.0%)	25,483(69.3%)	4,312(11.7%)
All Transactions	48,928	10,212(20.9%)	33,402(68.3%)	5,314(10.9%)

Codes for generating Table 2 are listed below.

```r
# Table 2

table2_raw <-  rbind(read_sas(sales_allother_zipcode_path), read_sas(sales_cc_0mile_path))
table2_raw$direct_to_website <- ifelse(table2_raw$ref_domain_name == '', 1, 0)
table2_raw$referred_by_search <- ifelse(table2_raw$ref_domain_name %in% search_engine_to_consider1, 1, 0)
table2_raw$referred_by_other <- ifelse(!(table2_raw$ref_domain_name %in% ref_domain_to_consider1), 1, 0)
table2_raw$domain_name[!(table2_raw$domain_name %in% c('amazon.com', 'bestbuy.com'))] <- "All Others"
```

Code 3: Table 2 Generation

2.3 Table 3

Table 3 reports the model-free average DID values for some outcome variables.

Table 3: Average Difference-in-Difference (DID) of the Outcome Variables

Outcome Variable	Groups	After Store Closure	Before Store Closure	First Difference (se)	DID
Amazon Sales	Control	3.418	3.303	0.115 (0.031)	-0.167
	Treatment	3.351	3.403	-0.052 (0.212)	
Amazon PagesPerDollar	Control	1.188	1.147	0.041 (0.025)	0.257
	Treatment	1.363	1.065	0.298 (0.153)	
Amazon MinsPerDollar	Control	1.016	0.975	0.041 (0.025)	0.263
	Treatment	1.187	0.882	0.304 (0.137)	
bestbuy.com Sales	Control	3.418	3.303	0.354 (0.031)	0.623
	Treatment	3.351	3.403	0.976 (0.212)	
bestbuy.com PagesPerDollar	Control	1.188	1.147	-0.109 (0.025)	0.074
	Treatment	1.363	1.065	-0.035 (0.153)	
bestbuy.com MinsPerDollar	Control	1.016	0.975	-0.084 (0.025)	-0.012
	Treatment	1.187	0.882	-0.096 (0.137)	

Codes for generating Table 3 are listed below.

```r
# Table 3
temp <- read_sas(sales_allother_zipcode_path)
temp$Store_Close_Status <- 0
table3_0m_raw <-  rbind(temp, read_sas(sales_cc_0mile_path))
```

```r
table3_5m_raw <-  rbind(temp, read_sas(sales_cc_5miles_path))

# Date Transform
table3_0m_raw$event_date <- as.Date(table3_0m_raw$event_date)
table3_5m_raw$event_date <- as.Date(table3_5m_raw$event_date)

# construct MonthYear - month of year
table3_0m_raw$MonthYear <- format(table3_0m_raw$event_date, "%Y-%m")
table3_5m_raw$MonthYear <- format(table3_5m_raw$event_date, "%Y-%m")

# Mark CC Closure

# CCStorePresent
# it is the same as Store_Close_Status
table3_0m_raw$CCStorePresent <- table3_0m_raw$Store_Close_Status
table3_5m_raw$CCStorePresent <- table3_5m_raw$Store_Close_Status

# AfterStoreClosing
table3_0m_raw$AfterStoreClosing <- ifelse(table3_0m_raw$MonthYear < "2008-11", 0, 1)
table3_5m_raw$AfterStoreClosing <- ifelse(table3_5m_raw$MonthYear < "2008-11", 0, 1)

# BBStorePresent
table3_0m_raw <- merge(table3_0m_raw, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)
table3_5m_raw <- merge(table3_5m_raw, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)

table3_0m_raw$BBStorePresent <- na.fill(table3_0m_raw$BB_Store_Status, 0)
table3_5m_raw$BBStorePresent <- na.fill(table3_5m_raw$BB_Store_Status, 0)

# aggregate data

table3_0m_aggregate <- sqldf("SELECT Zip_Code, MonthYear, domain_name, count(*) AS TotalTransactions, SUM(pages_viewed) as TotalPages, SUM(prod_
      totprice) as TotalMonthlySales, SUM(duration) as TotalMins, SUM(pages_viewed) / SUM(prod_totprice) AS PagesPerDollar, SUM(duration) / SUM(
      prod_totprice) AS MinsPerDollar, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS
      AfterStoreClosing FROM table3_0m_raw GROUP BY Zip_Code, MonthYear, domain_name")
table3_5m_aggregate <- sqldf("SELECT Zip_Code, MonthYear, domain_name, count(*) AS TotalTransactions, SUM(pages_viewed) as TotalPages, SUM(prod_
      totprice) as TotalMonthlySales, SUM(duration) as TotalMins, SUM(pages_viewed) / SUM(prod_totprice) AS PagesPerDollar, SUM(duration) / SUM(
      prod_totprice) AS MinsPerDollar, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS
      AfterStoreClosing FROM table3_5m_raw GROUP BY Zip_Code, MonthYear, domain_name")

# Table 3 Gen Func
table3_gen <- function(table3_raw, domain_name_used, print_name){
  # Amazon Sales
  # for control
  amazonsales_control_before <- table3_raw[(table3_raw$CCStorePresent == 0) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
      AfterStoreClosing == 0),]$TotalMonthlySales
  amazonsales_control_after <- table3_raw[(table3_raw$CCStorePresent == 0) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
      AfterStoreClosing == 1),]$TotalMonthlySales

  amazonsales_control_before <- log(amazonsales_control_before + 1)
  amazonsales_control_after <- log(amazonsales_control_after + 1)
  # t test
  t_test.amazonsales_control <- t.test(amazonsales_control_after, amazonsales_control_before)
  amazonsales_control_mean_diff_se <- t_test.amazonsales_control$stderr
  t_test.amazonsales_control$p.value
  amazonsales_control_after_mean <- t_test.amazonsales_control$estimate[["mean of x"]]
  amazonsales_control_before_mean <- t_test.amazonsales_control$estimate[["mean of y"]]
  amazonsales_control_mean_diff <- t_test.amazonsales_control$estimate[["mean of x"]] - t_test.amazonsales_control$estimate[["mean of y"]]

  # Amazon Sales
  # for treatment
  amazonsales_treatment_before <- table3_raw[(table3_raw$CCStorePresent == 1) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
      AfterStoreClosing == 0),]$TotalMonthlySales
  amazonsales_treatment_after <- table3_raw[(table3_raw$CCStorePresent == 1) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
      AfterStoreClosing == 1),]$TotalMonthlySales

  amazonsales_treatment_before <- log(amazonsales_treatment_before + 1)
  amazonsales_treatment_after  <- log(amazonsales_treatment_after + 1)
  # t test
  t_test.amazonsales_treatment <- t.test(amazonsales_treatment_after, amazonsales_treatment_before)
  amazonsales_treatment_mean_diff_se <- t_test.amazonsales_treatment$stderr
  t_test.amazonsales_treatment$p.value
  amazonsales_treatment_after_mean <- t_test.amazonsales_treatment$estimate[["mean of x"]]
```

```r
amazonsales_treatment_before_mean <- t_test.amazonsales_treatment$estimate[["mean of y"]]
amazonsales_treatment_mean_diff <- t_test.amazonsales_treatment$estimate[["mean of x"]] - t_test.amazonsales_treatment$estimate[["mean of y"]]

# Amazon Sales DID
amazonsales_did <- amazonsales_treatment_mean_diff - amazonsales_control_mean_diff

# Amazon PagesPerDollar
# for control
amazonppd_control_before <- table3_raw[(table3_raw$CCStorePresent == 0) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
    AfterStoreClosing == 0),]$TotalPages / table3_raw[(table3_raw$CCStorePresent == 0) & (table3_raw$domain_name == domain_name_used) & (table3_
    raw$AfterStoreClosing == 0),]$TotalMonthlySales
amazonppd_control_after <- table3_raw[(table3_raw$CCStorePresent == 0) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
    AfterStoreClosing == 1),]$TotalPages / table3_raw[(table3_raw$CCStorePresent == 0) & (table3_raw$domain_name == domain_name_used) & (table3_
    raw$AfterStoreClosing == 1),]$TotalMonthlySales

amazonppd_control_before <- log(amazonppd_control_before + 1)
amazonppd_control_after  <- log(amazonppd_control_after + 1)
# t test
t_test.amazonppd_control <- t.test(amazonppd_control_after, amazonppd_control_before)
amazonppd_control_mean_diff_se <- t_test.amazonppd_control$stderr
t_test.amazonppd_control$p.value
amazonppd_control_after_mean <- t_test.amazonppd_control$estimate[["mean of x"]]
amazonppd_control_before_mean <- t_test.amazonppd_control$estimate[["mean of y"]]
amazonppd_control_mean_diff <- t_test.amazonppd_control$estimate[["mean of x"]] - t_test.amazonppd_control$estimate[["mean of y"]]

# Amazon PagesPerDollar
# for treatment
amazonppd_treatment_before <- table3_raw[(table3_raw$CCStorePresent == 1) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
    AfterStoreClosing == 0),]$TotalPages / table3_raw[(table3_raw$CCStorePresent == 1) & (table3_raw$domain_name == domain_name_used) & (table3_
    raw$AfterStoreClosing == 0),]$TotalMonthlySales
amazonppd_treatment_after  <- table3_raw[(table3_raw$CCStorePresent == 1) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
    AfterStoreClosing == 1),]$TotalPages / table3_raw[(table3_raw$CCStorePresent == 1) & (table3_raw$domain_name == domain_name_used) & (table3_
    raw$AfterStoreClosing == 1),]$TotalMonthlySales

amazonppd_treatment_before <- log(amazonppd_treatment_before + 1)
amazonppd_treatment_after  <- log(amazonppd_treatment_after + 1)
# t test
t_test.amazonppd_treatment <- t.test(amazonppd_treatment_after, amazonppd_treatment_before)
amazonppd_treatment_mean_diff_se <- t_test.amazonppd_treatment$stderr
t_test.amazonppd_treatment$p.value
amazonppd_treatment_after_mean <- t_test.amazonppd_treatment$estimate[["mean of x"]]
amazonppd_treatment_before_mean <-t_test.amazonppd_treatment$estimate[["mean of y"]]
amazonppd_treatment_mean_diff <- t_test.amazonppd_treatment$estimate[["mean of x"]] - t_test.amazonppd_treatment$estimate[["mean of y"]]

# Amazon PagesPerDollar DID
amazonppd_did <- amazonppd_treatment_mean_diff - amazonppd_control_mean_diff

# Amazon MinsPerDollar
# for control
amazonmpd_control_before <- table3_raw[(table3_raw$CCStorePresent == 0) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
    AfterStoreClosing == 0),]$TotalMins / table3_raw[(table3_raw$CCStorePresent == 0) & (table3_raw$domain_name == domain_name_used) & (table3_
    raw$AfterStoreClosing == 0),]$TotalMonthlySales
amazonmpd_control_after  <- table3_raw[(table3_raw$CCStorePresent == 0) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
    AfterStoreClosing == 1),]$TotalMins / table3_raw[(table3_raw$CCStorePresent == 0) & (table3_raw$domain_name == domain_name_used) & (table3_
    raw$AfterStoreClosing == 1),]$TotalMonthlySales

amazonmpd_control_before <- log(amazonmpd_control_before + 1)
amazonmpd_control_after  <- log(amazonmpd_control_after + 1)
# t test
t_test.amazonmpd_control <- t.test(amazonmpd_control_after, amazonmpd_control_before)
amazonmpd_control_mean_diff_se <- t_test.amazonmpd_control$stderr
t_test.amazonmpd_control$p.value
amazonmpd_control_after_mean <- t_test.amazonmpd_control$estimate[["mean of x"]]
amazonmpd_control_before_mean <- t_test.amazonmpd_control$estimate[["mean of y"]]
amazonmpd_control_mean_diff <- t_test.amazonmpd_control$estimate[["mean of x"]] - t_test.amazonmpd_control$estimate[["mean of y"]]

# Amazon MinsPerDollar
# for treatment
amazonmpd_treatment_before <- table3_raw[(table3_raw$CCStorePresent == 1) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
    AfterStoreClosing == 0),]$TotalMins / table3_raw[(table3_raw$CCStorePresent == 1) & (table3_raw$domain_name == domain_name_used) & (table3_
    raw$AfterStoreClosing == 0),]$TotalMonthlySales
amazonmpd_treatment_after  <- table3_raw[(table3_raw$CCStorePresent == 1) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
```

```
      AfterStoreClosing == 1),]$TotalMins / table3_raw[(table3_raw$CCStorePresent == 1) & (table3_raw$domain_name == domain_name_used) & (table3_
      raw$AfterStoreClosing == 1),]$TotalMonthlySales

  amazonmpd_treatment_before <- log(amazonmpd_treatment_before + 1)
  amazonmpd_treatment_after  <- log(amazonmpd_treatment_after + 1)
  # t test
  t_test.amazonmpd_treatment <- t.test(amazonmpd_treatment_after, amazonmpd_treatment_before)
  amazonmpd_treatment_mean_diff_se <- t_test.amazonmpd_treatment$stderr
  t_test.amazonmpd_treatment$p.value
  amazonmpd_treatment_after_mean <- t_test.amazonmpd_treatment$estimate[["mean of x"]]
  amazonmpd_treatment_before_mean <- t_test.amazonmpd_treatment$estimate[["mean of y"]]
  amazonmpd_treatment_mean_diff <- t_test.amazonmpd_treatment$estimate[["mean of x"]] - t_test.amazonmpd_treatment$estimate[["mean of y"]]

  # Amazon MinsPerDollar DID
  amazonmpd_did <- amazonmpd_treatment_mean_diff - amazonmpd_control_mean_diff

  # construct table
  return(rbind(c(paste(print_name,"Sales"),"Control", amazonsales_control_after_mean, amazonsales_control_before_mean, amazonsales_control_mean_
      diff, amazonsales_control_mean_diff_se, amazonsales_did),
               c(paste(print_name,"Sales"),"Treatment", amazonsales_treatment_after_mean, amazonsales_treatment_before_mean, amazonsales_treatment_
      mean_diff, amazonsales_treatment_mean_diff_se, amazonsales_did),
               c(paste(print_name,"PagesPerDollar"),"Control", amazonppd_control_after_mean, amazonppd_control_before_mean, amazonppd_control_mean_
      diff, amazonppd_control_mean_diff_se, amazonppd_did),
               c(paste(print_name,"PagesPerDollar"),"Treatment", amazonppd_treatment_after_mean, amazonppd_treatment_before_mean, amazonppd_
      treatment_mean_diff, amazonppd_treatment_mean_diff_se, amazonppd_did),
               c(paste(print_name,"MinsPerDollar"),"Control", amazonmpd_control_after_mean, amazonmpd_control_before_mean, amazonmpd_control_mean_
      diff, amazonmpd_control_mean_diff_se, amazonmpd_did),
               c(paste(print_name,"MinsPerDollar"),"Treatment", amazonmpd_treatment_after_mean, amazonmpd_treatment_before_mean, amazonmpd_
      treatment_mean_diff, amazonmpd_treatment_mean_diff_se, amazonmpd_did))
  )
}

# generate table
amazon_table3 <- table3_gen(table3_0m_aggregate, "amazon.com", "Amazon")
bestbuy_table3 <- table3_gen(table3_0m_aggregate, "bestbuy.com", "bestbuy.com")

#
stargazer(rbind(amazon_table3, bestbuy_table3), align=TRUE, summary = FALSE, rownames = FALSE, title="Summary Statistics of Top Five Vendors by
      Sales Volume")
```

Code 4: Table 3 Generation

2.4 Table 4

Codes for generating Table 4 are listed below.

```
# Table 4 Data
data_0m_t4 <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(prod_totprice) AS TotalMonthlySales, AVG(CCStorePresent) AS CCStorePresent, AVG(
      BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS AfterStoreClosing FROM concat_data1 GROUP BY Zip_Code, MonthYear, domain_name")
data_5m_t4 <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(prod_totprice) AS TotalMonthlySales, AVG(CCStorePresent) AS CCStorePresent, AVG(
      BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS AfterStoreClosing FROM concat_data2 GROUP BY Zip_Code, MonthYear, domain_name")
# manually construct DID and THREEINTERACTION
data_0m_t4$DID <- data_0m_t4$CCStorePresent * data_0m_t4$AfterStoreClosing
data_0m_t4$THREEINTER <- data_0m_t4$CCStorePresent * data_0m_t4$AfterStoreClosing * data_0m_t4$BBStorePresent
data_5m_t4$DID <- data_5m_t4$CCStorePresent * data_5m_t4$AfterStoreClosing
data_5m_t4$THREEINTER <- data_5m_t4$CCStorePresent * data_5m_t4$AfterStoreClosing * data_5m_t4$BBStorePresent
# Table 4
ama.t4.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t4[data_0m_t4$domain_name == "amazon.com",], index = c("Zip_Code"
      , "MonthYear"), model = "within", effect = "twoways")
ama.t4.5mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_5m_t4[data_5m_t4$domain_name == "amazon.com",], index = c("Zip_Code"
      , "MonthYear"), model = "within", effect = "twoways")
bb.t4.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t4[data_0m_t4$domain_name == "bestbuy.com",], index = c("Zip_Code"
      , "MonthYear"), model = "within", effect = "twoways")
bb.t4.5mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_5m_t4[data_5m_t4$domain_name == "bestbuy.com",], index = c("Zip_Code"
      , "MonthYear"), model = "within", effect = "twoways")
```

Code 5: Table 4 Generation

Table 4: Results of the Sales Effect (All Product Categories)

	log(TotalMonthlySales + 1)			
	Amazon-0 Mile	Amazon-5 Miles	BestBuy-0 Mile	BestBuy-5 Miles
	(1)	(2)	(3)	(4)
β_1	0.138	−0.202	0.090	−1.002
	(0.524)	(0.198)	(1.171)	(0.896)
β_2	0.109	0.096		1.116
	(0.668)	(0.245)		(1.224)
Observations	5,126	5,781	729	837
R^2	0.0002	0.001	0.0001	0.012
Adjusted R^2	-1.279	-1.199	-6.744	-6.512
F Statistic	0.204 (df = 2; 2248)	0.672 (df = 2; 2627)	0.006 (df = 1; 94)	0.647 (df = 2; 110)

Note: *p<0.1; **p<0.05; ***p<0.01

2.5 Table 5

Table 5: Results of the Search Effect (All Product Categories)

	log(PagesPerDollar + 1)				log(MinsPerDollar + 1)			
	Amazon-0 Mile	Amazon-5 Miles	BestBuy-0 Mile	BestBuy-5 Miles	Amazon-0 Mile	Amazon-5 Miles	BestBuy-0 Mile	BestBuy-5 Miles
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
β_1	−0.078	0.072	0.476	0.373	0.149	0.063	0.127	0.513
	(0.411)	(0.156)	(0.680)	(0.511)	(0.400)	(0.152)	(0.544)	(0.425)
β_2	−0.263	−0.056		−0.028	−0.387	−0.047		−0.309
	(0.524)	(0.193)		(0.697)	(0.510)	(0.188)		(0.580)
Observations	5,126	5,781	729	837	5,126	5,781	729	837
R^2	0.0005	0.0001	0.005	0.008	0.0003	0.0001	0.001	0.014
Adjusted R^2	-1.279	-1.200	-6.705	-6.537	-1.279	-1.200	-6.740	-6.490
F Statistic	0.552 (df = 2; 2248)	0.109 (df = 2; 2627)	0.490 (df = 1; 94)	0.458 (df = 2; 110)	0.347 (df = 2; 2248)	0.089 (df = 2; 2627)	0.055 (df = 1; 94)	0.805 (df = 2; 110)

Note: *p<0.1; **p<0.05; ***p<0.01

Codes for generating Table 5 are listed below.

```
# Table 5 Data
data_0m_t5 <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(pages_viewed) / SUM(prod_totprice) AS PagesPerDollar, SUM(duration) / SUM(prod_
      totprice) AS MinsPerDollar, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS
      AfterStoreClosing FROM concat_data1 GROUP BY Zip_Code, MonthYear, domain_name")
data_5m_t5 <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(pages_viewed) / SUM(prod_totprice) AS PagesPerDollar, SUM(duration) / SUM(prod_
      totprice) AS MinsPerDollar, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS
      AfterStoreClosing FROM concat_data2 GROUP BY Zip_Code, MonthYear, domain_name")
# manually construct DID and THREEINTERACTION
```

```
data_0m_t5$DID <- data_0m_t5$CCStorePresent * data_0m_t5$AfterStoreClosing
data_0m_t5$THREEINTER <- data_0m_t5$CCStorePresent * data_0m_t5$AfterStoreClosing * data_0m_t5$BBStorePresent
data_5m_t5$DID <- data_5m_t5$CCStorePresent * data_5m_t5$AfterStoreClosing
data_5m_t5$THREEINTER <- data_5m_t5$CCStorePresent * data_5m_t5$AfterStoreClosing * data_5m_t5$BBStorePresent
# Table 5
# For PagesPerDollar
ama.t5.pagesperdollar.0mile <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t5[data_0m_t5$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t5.pagesperdollar.5mile <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t5[data_5m_t5$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t5.pagesperdollar.0mile  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t5[data_0m_t5$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t5.pagesperdollar.5mile  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t5[data_5m_t5$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
# For MinsPerDollar
ama.t5.minsperdollar.0mile <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t5[data_0m_t5$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t5.minsperdollar.5mile <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t5[data_5m_t5$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t5.minsperdollar.0mile <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t5[data_0m_t5$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t5.minsperdollar.5mile <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t5[data_5m_t5$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
```

Code 6: Table 5 Generation

2.6 Table 6

Table 6: Results of the Sales Effect: Experience and Search Products

	log(TotalMonthlySales + 1)						
	Amazon 0 Mile Experience	Amazon 5 Miles Experience	Amazon 0 Mile Search	Amazon 5 Miles Search	BestBuy 0 Mile Experience	BestBuy 5 Miles Experience	BestBuy 5 Miles Search
	(1)	(2)	(3)	(4)	(5)	(6)	(7)
β_1	−1.783	0.245	0.517	−0.304*	−1.574	−2.021	−2.515**
	(1.821)	(0.697)	(0.634)	(0.180)	(2.994)	(2.054)	(0.993)
β_2	1.785	−0.173	−0.069	0.281		2.867	
	(2.015)	(0.772)	(0.816)	(0.229)		(2.277)	
Observations	1,788	2,022	3,741	4,229	461	529	342
R^2	0.002	0.0003	0.001	0.002	0.024	0.090	0.130
Adjusted R^2	-3.195	-2.923	-1.437	-1.343	-39.794	-29.016	-5.901
F Statistic	0.480 (df = 2; 425)	0.071 (df = 2; 515)	0.704 (df = 2; 1533)	1.424 (df = 2; 1802)	0.276 (df = 1; 11)	0.795 (df = 2; 16)	6.417** (df = 1; 43)

Note: *p<0.1; **p<0.05; ***p<0.01

Codes for generating Table 6 are listed below.

```
# Table 6 Data
data_0m_t6_exp <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(prod_totprice) AS TotalMonthlySales, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS AfterStoreClosing FROM concat_data1_exp GROUP BY Zip_Code, MonthYear, domain_name")
data_0m_t6_search <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(prod_totprice) AS TotalMonthlySales, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS AfterStoreClosing FROM concat_data1_search GROUP BY Zip_Code, MonthYear, domain_name")
data_5m_t6_exp <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(prod_totprice) AS TotalMonthlySales, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS AfterStoreClosing FROM concat_data2_exp GROUP BY Zip_Code, MonthYear, domain_name")
```

```
data_5m_t6_search <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(prod_totprice) AS TotalMonthlySales, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS AfterStoreClosing FROM concat_data2_search GROUP BY Zip_Code, MonthYear, domain_name")
# manually construct DID and THREEINTERACTION
data_0m_t6_exp$DID <- data_0m_t6_exp$CCStorePresent * data_0m_t6_exp$AfterStoreClosing
data_0m_t6_exp$THREEINTER <- data_0m_t6_exp$CCStorePresent * data_0m_t6_exp$AfterStoreClosing * data_0m_t6_exp$BBStorePresent
data_0m_t6_search$DID <- data_0m_t6_search$CCStorePresent * data_0m_t6_search$AfterStoreClosing
data_0m_t6_search$THREEINTER <- data_0m_t6_search$CCStorePresent * data_0m_t6_search$AfterStoreClosing * data_0m_t6_search$BBStorePresent
data_5m_t6_exp$DID <- data_5m_t6_exp$CCStorePresent * data_5m_t6_exp$AfterStoreClosing
data_5m_t6_exp$THREEINTER <- data_5m_t6_exp$CCStorePresent * data_5m_t6_exp$AfterStoreClosing * data_5m_t6_exp$BBStorePresent
data_5m_t6_search$DID <- data_5m_t6_search$CCStorePresent * data_5m_t6_search$AfterStoreClosing
data_5m_t6_search$THREEINTER <- data_5m_t6_search$CCStorePresent * data_5m_t6_search$AfterStoreClosing * data_5m_t6_search$BBStorePresent
# Table 6
# AmazonTotalMonthlySales & BBTotalMonthlySale vs Experience and Search Product
ama.t6.0mile.exp <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t6_exp[data_0m_t6_exp$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t6.5mile.exp <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_5m_t6_exp[data_5m_t6_exp$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t6.0mile.search <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t6_search[data_0m_t6_search$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t6.5mile.search <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_5m_t6_search[data_5m_t6_search$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t6.0mile.exp <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t6_exp[data_0m_t6_exp$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t6.5mile.exp <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_5m_t6_exp[data_5m_t6_exp$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t6.0mile.search <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t6_search[data_0m_t6_search$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t6.5mile.search <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_5m_t6_search[data_5m_t6_search$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
```

Code 7: Table 6 Generation

2.7 Table 7

Table 7: Results of the Online Search Effect: Experience Products

	log(PagesPerDollar + 1)				log(MinsPerDollar + 1)			
	Amazon-0 Mile	Amazon-5 Miles	BestBuy-0 Mile	BestBuy-5 Miles	Amazon-0 Mile	Amazon-5 Miles	BestBuy-0 Mile	BestBuy-5 Miles
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
β_1	0.663	−0.067	1.091	0.136	1.046	0.039	0.886	0.386
	(1.451)	(0.547)	(0.800)	(0.634)	(1.406)	(0.536)	(0.625)	(0.481)
β_2	−1.167	0.145		0.043	−1.366	−0.021		−0.122
	(1.605)	(0.606)		(0.703)	(1.555)	(0.593)		(0.533)
Observations	1,788	2,022	461	529	1,788	2,022	461	529
R^2	0.002	0.0002	0.145	0.013	0.002	0.00002	0.155	0.069
Adjusted R^2	-3.198	-2.924	-34.767	-31.568	-3.197	-2.924	-34.355	-29.730
F Statistic	0.358 (df = 2; 425)	0.040 (df = 2; 515)	1.861 (df = 1; 11)	0.106 (df = 2; 16)	0.390 (df = 2; 425)	0.004 (df = 2; 515)	2.011 (df = 1; 11)	0.591 (df = 2; 16)

Note: *p<0.1; **p<0.05; ***p<0.01

Codes for generating Table 7 are listed below.

```
# Table 7 & 8 Data
data_0m_t7_exp    <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(pages_viewed) / SUM(prod_totprice) AS PagesPerDollar, SUM(duration) / SUM(prod_totprice) AS MinsPerDollar, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS AfterStoreClosing FROM concat_data1_exp GROUP BY Zip_Code, MonthYear, domain_name")
data_0m_t8_search <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(pages_viewed) / SUM(prod_totprice) AS PagesPerDollar, SUM(duration) / SUM(prod_totprice) AS MinsPerDollar, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS AfterStoreClosing FROM concat_data1_search GROUP BY Zip_Code, MonthYear, domain_name")
```

```
data_5m_t7_exp    <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(pages_viewed) / SUM(prod_totprice) AS PagesPerDollar, SUM(duration) / SUM(prod_totprice) AS MinsPerDollar, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS AfterStoreClosing FROM concat_data2_exp GROUP BY Zip_Code, MonthYear, domain_name")
data_5m_t8_search <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(pages_viewed) / SUM(prod_totprice) AS PagesPerDollar, SUM(duration) / SUM(prod_totprice) AS MinsPerDollar, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS AfterStoreClosing FROM concat_data2_search GROUP BY Zip_Code, MonthYear, domain_name")
# manually construct DID and THREEINTERACTION
data_0m_t7_exp$DID <- data_0m_t7_exp$CCStorePresent * data_0m_t7_exp$AfterStoreClosing
data_0m_t7_exp$THREEINTER <- data_0m_t7_exp$CCStorePresent * data_0m_t7_exp$AfterStoreClosing * data_0m_t7_exp$BBStorePresent
data_0m_t8_search$DID <- data_0m_t8_search$CCStorePresent * data_0m_t8_search$AfterStoreClosing
data_0m_t8_search$THREEINTER <- data_0m_t8_search$CCStorePresent * data_0m_t8_search$AfterStoreClosing * data_0m_t8_search$BBStorePresent
data_5m_t7_exp$DID <- data_5m_t7_exp$CCStorePresent * data_5m_t7_exp$AfterStoreClosing
data_5m_t7_exp$THREEINTER <- data_5m_t7_exp$CCStorePresent * data_5m_t7_exp$AfterStoreClosing * data_5m_t7_exp$BBStorePresent
data_5m_t8_search$DID <- data_5m_t8_search$CCStorePresent * data_5m_t8_search$AfterStoreClosing
data_5m_t8_search$THREEINTER <- data_5m_t8_search$CCStorePresent * data_5m_t8_search$AfterStoreClosing * data_5m_t8_search$BBStorePresent
# Table 7
ama.t7.pagesperdollar.0mile.exp <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t7_exp[data_0m_t7_exp$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t7.pagesperdollar.5mile.exp <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t7_exp[data_5m_t7_exp$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t7.pagesperdollar.0mile.exp  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t7_exp[data_0m_t7_exp$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t7.pagesperdollar.5mile.exp  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t7_exp[data_5m_t7_exp$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t7.minsperdollar.0mile.exp  <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t7_exp[data_0m_t7_exp$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t7.minsperdollar.5mile.exp  <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t7_exp[data_5m_t7_exp$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t7.minsperdollar.0mile.exp   <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t7_exp[data_0m_t7_exp$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t7.minsperdollar.5mile.exp   <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t7_exp[data_5m_t7_exp$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
```

Code 8: Table 7 Generation

2.8 Table 8

Table 8: Results of the Online Search Effect: Search Products

	log(PagesPerDollar + 1)			log(MinsPerDollar + 1)		
	Amazon-0 Mile	Amazon-5 Miles	BestBuy-5 Miles	Amazon-0 Mile	Amazon-5 Miles	BestBuy-5 Miles
	(1)	(2)	(3)	(4)	(5)	(6)
β_1	0.051	0.181	1.038	0.395	0.151	1.522**
	(0.554)	(0.160)	(0.752)	(0.555)	(0.159)	(0.711)
β_2	−0.320	−0.214		−0.778	−0.182	
	(0.713)	(0.203)		(0.715)	(0.203)	
Observations	3,741	4,229	342	3,741	4,229	342
R^2	0.0002	0.001	0.042	0.001	0.001	0.096
Adjusted R^2	-1.439	-1.344	-6.593	-1.438	-1.345	-6.166
F Statistic	0.182 (df = 2; 1533)	0.704 (df = 2; 1802)	1.907 (df = 1; 43)	0.613 (df = 2; 1533)	0.501 (df = 2; 1802)	4.583** (df = 1; 43)

Note: *p<0.1; **p<0.05; ***p<0.01

Codes for generating Table 8 are listed below.

```
# Table 8
ama.t8.pagesperdollar.0mile.search <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t8_search[data_0m_t8_search$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
```

```r
ama.t8.pagesperdollar.5mile.search <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t8_search[data_5m_t8_search$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t8.pagesperdollar.0mile.search  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t8_search[data_0m_t8_search$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t8.pagesperdollar.5mile.search  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t8_search[data_5m_t8_search$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t8.minsperdollar.0mile.search  <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t8_search[data_0m_t8_search$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t8.minsperdollar.5mile.search  <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t8_search[data_5m_t8_search$domain_name == "amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t8.minsperdollar.0mile.search   <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t8_search[data_0m_t8_search$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t8.minsperdollar.5mile.search   <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t8_search[data_5m_t8_search$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
```

Code 9: Table 8 Generation

2.9 Table 9

Table 9: Results of Logistic Regression for Referring Domain

	ReferringDomainIsSearchEngine		NoReferringDomain	
	Amazon-0 Mile	BestBuy-0 Mile	Amazon-0 Mile	BestBuy-0 Mile
	(1)	(2)	(3)	(4)
β_1	-0.817^{*}	-15.12^{***}	0.325	-0.223
	(0.337)	(0.611)	(0.346)	(1.259)
β_2	0.697	14.43^{***}	-0.415	0.916
	(0.564)	(0.944)	(0.544)	(1.615)
Observations	10,791	1,225	10,791	1,225

Note: *p<0.05; **p<0.01; ***p<0.001

Stata codes for generating Table 9 are listed below.

```
* build variables
gen DID =  CCStorePresent *  AfterStoreClosing
gen THREEINTER = DID *  BBStorePresent
egen Code_Time = group(Zip_Code MonthYear)

* Amazon - ReferringDomainIsSearchEngine & NoReferringDomain
eststo: logit ReferringDomainIsSearchEngine DID THREEINTER if domain_name == "amazon.com", vce(cluster Code_Time) noconstant
eststo: logit NoReferringDomain DID THREEINTER if domain_name == "amazon.com", vce(cluster Code_Time) noconstant
* BestBuy - ReferringDomainIsSearchEngine & NoReferringDomain
eststo: logit ReferringDomainIsSearchEngine DID THREEINTER if domain_name == "bestbuy.com", vce(cluster MonthYear) noconstant
```

```
eststo: logit NoReferringDomain DID THREEINTER if domain_name == "bestbuy.com", vce(cluster MonthYear) noconstant
```

Code 10: Table 9 Generation

2.10 Table 10

Table 10: Results of the Online Sales and Search Effect (All Product Categories)

	log(SalesPerTransaction + 1)		log(PagesPerTransaction + 1)		log(MinsPerTransaction + 1)	
	Amazon-0 Mile	BestBuy-0 Mile	Amazon-0 Mile	BestBuy-0 Mile	Amazon-0 Mile	BestBuy-0 Mile
	(1)	(2)	(3)	(4)	(5)	(6)
β_1	0.104	0.064	0.137	0.929	0.366	0.468
	(0.449)	(1.148)	(0.324)	(0.683)	(0.403)	(0.741)
β_2	0.133		−0.446		−0.559	
	(0.572)		(0.413)		(0.514)	
Observations	5,126	729	5,126	729	5,126	729
R^2	0.0002	0.00003	0.001	0.019	0.001	0.004
Adjusted R^2	-1.279	-6.744	-1.278	-6.595	-1.279	-6.712
F Statistic	0.240 (df = 2; 2248)	0.003 (df = 1; 94)	0.800 (df = 2; 2248)	1.850 (df = 1; 94)	0.596 (df = 2; 2248)	0.399 (df = 1; 94)

Note: *p<0.1; **p<0.05; ***p<0.01

Codes for generating Table 10 are listed below.

```
# Table 10 Data
data_0m_t10 <- sqldf("SELECT Zip_Code, MonthYear, domain_name, AVG(pages_viewed) AS PagesPerTransaction, AVG(duration) AS MinsPerTransaction, AVG(
      prod_totprice) AS SalesPerTransaction, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing)
      AS AfterStoreClosing FROM concat_data1 GROUP BY Zip_Code, MonthYear, domain_name")

data_0m_t10$DID <- data_0m_t10$CCStorePresent * data_0m_t10$AfterStoreClosing
data_0m_t10$THREEINTER <- data_0m_t10$CCStorePresent * data_0m_t10$AfterStoreClosing * data_0m_t10$BBStorePresent
# Table 10
# SalesPerTransaction; PagesPerTransaction; MinsPerTransaction; for Ama & BB
ama.t10.0mile.SalesPerTransaction <- plm(log(SalesPerTransaction + 1) ~ DID + THREEINTER, data = data_0m_t10[data_0m_t10$domain_name == "amazon.com
      ",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t10.0mile.SalesPerTransaction  <- plm(log(SalesPerTransaction + 1) ~ DID + THREEINTER, data = data_0m_t10[data_0m_t10$domain_name == "bestbuy.
      com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t10.0mile.PagesPerTransaction <- plm(log(PagesPerTransaction + 1) ~ DID + THREEINTER, data = data_0m_t10[data_0m_t10$domain_name == "amazon.com
      ",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t10.0mile.PagesPerTransaction  <- plm(log(PagesPerTransaction + 1) ~ DID + THREEINTER, data = data_0m_t10[data_0m_t10$domain_name == "bestbuy.
      com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t10.0mile.MinsPerTransaction <- plm(log(MinsPerTransaction + 1) ~ DID + THREEINTER, data = data_0m_t10[data_0m_t10$domain_name == "amazon.com"
      ,], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t10.0mile.MinsPerTransaction  <- plm(log(MinsPerTransaction + 1) ~ DID + THREEINTER, data = data_0m_t10[data_0m_t10$domain_name == "bestbuy.com"
      ,], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
```

Code 11: Table 10 Generation

2.11 Table 11

Codes for generating Table 11 are listed below.

Table 11: Results of the Online Sales and Search Effect After Matching Zip Codes: TotalMonthlySales, PagesPerDollar, and MinsPerDollar (All Product Categories)

	log(TotalMonthlySales + 1) Amazon-0 Mile	log(PagesPerDollar + 1) Amazon-0 Mile	log(MinsPerDollar + 1) Amazon-0 Mile
	(1)	(2)	(3)
β_1	−0.069	−0.137	0.009
	(0.833)	(0.455)	(0.403)
β_2	0.035	−0.112	−0.286
	(0.849)	(0.464)	(0.411)
Observations	67	67	67
R^2	0.054	0.213	0.235
Adjusted R^2	-19.809	-16.304	-15.837
F Statistic (df = 2; 3)	0.086	0.407	0.460

Note: *p<0.1; **p<0.05; ***p<0.01

```
library(cem)
#matching based on zipcode demographics (cross-sectional)
data_0m_t11 <- sqldf("SELECT Zip_Code, SUM(prod_totprice) AS TotalMonthlySales,
                     AVG(CCStorePresent) AS CCStorePresent,
                     AVG(household_size) AS HoHSize,
                     AVG(hoh_oldest_age) AS HoHAge,
                     AVG(household_income) AS HoHIncome,
                     AVG(children) AS HoHChildren,
                     AVG(connection_speed) AS HoHSpeed
                     FROM concat_data1 GROUP BY Zip_Code")

#check imblance within data set
vars <- c("HoHSize", "HoHAge", "HoHIncome", "HoHChildren", "HoHSpeed")
imbalance(group=data_0m_t11$CCStorePresent, data = data_0m_t11[vars])

# Default is not 1-1 matching in CEM. Use k2k = "True" to enforce 1 to 1 matching.
todrop <- c("TotalMonthlySales")
todrop2 <- c("TotalMonthlySales", "Zip_Code")
# mat <- cem(treatment = "CCStorePresent", data = data_0m_t11, drop = todrop, k2k ="True")

mat <- cem(treatment = "CCStorePresent",
           data = data_0m_t11,
           drop = todrop2,
           k2k = TRUE,
           method = "euclidean")
mat

# We got 110 zipcodes in total. We checked 2 dataframe from CEM results, "w" and "matched", and both have 110 values.
# Fortunately, they are the same. In the future, just use data from "matched". Note that this is only ID of row value of Zipcode

# assign ID of row value of zipcode from "matched"
zipcheck <- c()

for (i in 1:length(mat$matched)){
```

```
  if (mat$matched[i] == "TRUE") zipcheck <-c(zipcheck,i)
}

data.frame(zipcheck)

# assign ID of row value of zipcode from "w"
zipcheck1 <- c()

for (i in 1:length(mat$w)){
  if (mat$w[i] == 1) zipcheck1 <-c(zipcheck1,i)
}

data.frame(zipcheck1)

# Test both dataframe, and they are same.
all.equal(zipcheck,zipcheck1)

# add specific Zipcode by mapping from ID of row of matched zipcode
ziplist <- c()
for (i in 1:length(data_0m_t11$Zip_Code)){
  if ( i %in% zipcheck) ziplist <-c(ziplist,data_0m_t11$Zip_Code[i])
}

data.frame(ziplist)

# assign matched zipcode to dataset
concat_data1$Zipmatch <- ifelse(concat_data1$Zip_Code %in% ziplist, 1, 0)
data_0m_t11 <- sqldf("SELECT Zip_Code, Zipmatch, MonthYear, domain_name, SUM(prod_totprice) AS TotalMonthlySales, SUM(pages_viewed) / SUM(prod_
      totprice) AS PagesPerDollar, SUM(duration) / SUM(prod_totprice) AS MinsPerDollar, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent)
      AS BBStorePresent, AVG(AfterStoreClosing) AS AfterStoreClosing FROM concat_data1 GROUP BY Zip_Code, MonthYear, domain_name")
data_0m_t11$DID <- data_0m_t11$CCStorePresent * data_0m_t11$AfterStoreClosing
data_0m_t11$THREEINTER <- data_0m_t11$DID * data_0m_t11$BBStorePresent

# result for Amazon regarding TotalMonthlySales, PagesPerDollar, MinsPerDollar
ama.t11.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t11[(data_0m_t11$domain_name == "amazon.com") & (data_0m_t11$
      Zipmatch == 1),], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t11.pagesperdollar.0mile <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t11[(data_0m_t11$domain_name == "amazon.com") & (
      data_0m_t11$Zipmatch == 1),], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t11.minsperdollar.0mile <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t11[(data_0m_t11$domain_name == "amazon.com") & (data_
      0m_t11$Zipmatch == 1),], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
# result for Bestbuy regarding TotalMonthlySales, PagesPerDollar, MinsPerDollar
bb.t11.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t11[(data_0m_t11$domain_name == "bestbuy.com") & (data_0m_t11$
      Zipmatch == 1),], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t11.pagesperdollar.0mile <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t11[(data_0m_t11$domain_name == "bestbuy.com") & (
      data_0m_t11$Zipmatch == 1),], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t11.minsperdollar.0mile <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t11[(data_0m_t11$domain_name == "bestbuy.com") & (data_
      0m_t11$Zipmatch == 1),], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
```

Code 12: Table 11 Generation

2.12 Table 12

Codes for generating Table 12 are listed below.

```
# Table 12 Data
data_0m_t12 <- sqldf("SELECT Zip_Code, MonthYear, domain_name, SUM(prod_totprice) AS TotalMonthlySales, SUM(pages_viewed) / SUM(prod_totprice) AS
      PagesPerDollar, SUM(duration) / SUM(prod_totprice) AS MinsPerDollar, AVG(household_size) AS HoHSize, AVG(hoh_oldest_age) AS HoHAge, AVG(
      household_income) AS HoHIncome, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent) AS BBStorePresent, AVG(AfterStoreClosing) AS
      AfterStoreClosing FROM concat_data1 GROUP BY Zip_Code, MonthYear, domain_name")
data_0m_t12$DID <- data_0m_t12$CCStorePresent * data_0m_t12$AfterStoreClosing * data_0m_t12$HoHSize * data_0m_t12$HoHAge * data_0m_t12$HoHIncome
data_0m_t12$THREEINTER <- data_0m_t12$CCStorePresent * data_0m_t12$AfterStoreClosing * data_0m_t12$BBStorePresent
# Table 12
ama.t12.0m.PagesPerDollar <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t12[data_0m_t12$domain_name == "amazon.com",], index =
      c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t12.0m.MinsPerDollar <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t12[data_0m_t12$domain_name == "amazon.com",], index = c(
      "Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.t12.0m.TotalMonthlySales <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t12[data_0m_t12$domain_name == "amazon.com",],
      index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
```

Table 12: Results of the Online Sales and Search Effect with Zip Code Demographics as Interactions and Time Fixed Effects (All Product Categories)

	log(TotalMonthlySales + 1)		log(PagesPerDollar + 1)		log(MinsPerDollar + 1)	
	Amazon-0 Mile	BestBuy-0 Mile	Amazon-0 Mile	BestBuy-0 Mile	Amazon-0 Mile	BestBuy-0 Mile
	(1)	(2)	(3)	(4)	(5)	(6)
β_1	−0.001	−0.012	0.002	−0.003	0.002	0.0001
	(0.002)	(0.018)	(0.002)	(0.010)	(0.002)	(0.008)
β_2	0.352	1.008	−0.493	0.718	−0.427	0.123
	(0.457)	(1.779)	(0.359)	(1.034)	(0.349)	(0.829)
Observations	5,126	729	5,126	729	5,126	729
R^2	0.0003	0.005	0.001	0.006	0.001	0.001
Adjusted R^2	-1.279	-6.788	-1.278	-6.779	-1.277	-6.823
F Statistic	0.348 (df = 2; 2248)	0.239 (df = 2; 93)	1.116 (df = 2; 2248)	0.291 (df = 2; 93)	1.216 (df = 2; 2248)	0.027 (df = 2; 93)

Note: *p<0.1; **p<0.05; ***p<0.01

```
bb.t12.0m.PagesPerDollar <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t12[data_0m_t12$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t12.0m.MinsPerDollar <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t12[data_0m_t12$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
bb.t12.0m.TotalMonthlySales <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t12[data_0m_t12$domain_name == "bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
```

Code 13: Table 12 Generation

2.13 Table 13

Codes for generating Table 13 are listed below.

```
library(cem)
#matching based on zipcode demographics (cross-sectional)
data_0m_t11 <- sqldf("SELECT Zip_Code, SUM(prod_totprice) AS TotalMonthlySales,
                     AVG(CCStorePresent) AS CCStorePresent,
                     AVG(household_size) AS HoHSize,
                     AVG(hoh_oldest_age) AS HoHAge,
                     AVG(household_income) AS HoHIncome,
                     AVG(children) AS HoHChildren,
                     AVG(connection_speed) AS HoHSpeed
                     FROM concat_data1 GROUP BY Zip_Code")
# CEM
todrop2 <- c("TotalMonthlySales", "Zip_Code")
mat <- cem(treatment = "CCStorePresent",
           data = data_0m_t11,
           drop = todrop2,
           k2k = TRUE,
           method = "euclidean")
mat

# Check Matching
zipcheck <- c()

for (i in 1:length(mat$matched)){
  if (mat$matched[i] == "TRUE") zipcheck <-c(zipcheck,i)
}

```

Table 13: Results of the Online Sales and Search Effect While Ignoring Time Series (All Product Categories)

	log(TotalMonthlySales + 1) Amazon-0 Mile	log(PagesPerDollar + 1) Amazon-0 Mile	log(MinsPerDollar + 1) Amazon-0 Mile
	(1)	(2)	(3)
β_1	0.106	0.160	0.548
	(0.812)	(0.593)	(0.499)
β_2	2.281	−1.116	−1.369
	(1.519)	(1.109)	(0.934)
Observations	107	107	107
R^2	0.101	0.035	0.069
Adjusted R^2	-2.075	-2.301	-2.184
F Statistic (df = 2; 31)	1.738	0.556	1.143

Note: *p<0.1; **p<0.05; ***p<0.01

```
data.frame(zipcheck)

# assign ID of row value of zipcode from "w"
zipcheck1 <- c()

for (i in 1:length(mat$w)){
  if (mat$w[i] == 1) zipcheck1 <-c(zipcheck1,i)
}

data.frame(zipcheck1)

# Test both dataframe, and they are same.
all.equal(zipcheck,zipcheck1)

# add specific Zipcode by mapping from ID of row of matched zipcode
ziplist <- c()
for (i in 1:length(data_0m_t11$Zip_Code)){
  if ( i %in% zipcheck) ziplist <-c(ziplist,data_0m_t11$Zip_Code[i])
}

data.frame(ziplist)

# Assign matched zipcode to dataset
concat_data1$Zipmatch <- ifelse(concat_data1$Zip_Code %in% ziplist, 1, 0)
data_0m_t13 <- sqldf("SELECT Zip_Code, Zipmatch, MonthYear, domain_name, SUM(prod_totprice) AS TotalMonthlySales, SUM(pages_viewed) / SUM(prod_
      totprice) AS PagesPerDollar, SUM(duration) / SUM(prod_totprice) AS MinsPerDollar, AVG(CCStorePresent) AS CCStorePresent, AVG(BBStorePresent)
      AS BBStorePresent, AVG(AfterStoreClosing) AS AfterStoreClosing FROM concat_data1 GROUP BY Zip_Code, MonthYear, domain_name")
data_0m_t13$DID <- data_0m_t13$CCStorePresent * data_0m_t13$AfterStoreClosing
data_0m_t13$THREEINTER <- data_0m_t13$DID * data_0m_t13$BBStorePresent

# Table 13
ama.0m.t13.sales <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t13[(data_0m_t13$domain_name == "amazon.com") & (data_0m_t13$
      Zipmatch == 1),], index = c("Zip_Code"), model = "within")
ama.0m.t13.ppd   <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t13[(data_0m_t13$domain_name == "amazon.com") & (data_0m_t13$
      Zipmatch == 1),], index = c("Zip_Code"), model = "within")
```

```
ama.0m.t13.mpd   <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t13[(data_0m_t13$domain_name == "amazon.com") & (data_0m_t13$
      Zipmatch == 1),], index = c("Zip_Code"), model = "within")

bb.0m.t13.sales <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t13[(data_0m_t13$domain_name == "bestbuy.com") & (data_0m_t13$
      Zipmatch == 1),], index = c("Zip_Code"), model = "within")
bb.0m.t13.ppd   <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t13[(data_0m_t13$domain_name == "bestbuy.com") & (data_0m_t13$
      Zipmatch == 1),], index = c("Zip_Code"), model = "within")
bb.0m.t13.mpd   <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t13[(data_0m_t13$domain_name == "bestbuy.com") & (data_0m_t13$
      Zipmatch == 1),], index = c("Zip_Code"), model = "within")
```

Code 14: Table 13 Generation

2.14 Table 14

Table 14: Results of the Online Sales and Search Effect with Arbitrary Variance-Covariance Matrix Corrections (All Product Categories)

	log(TotalMonthlySales + 1) Amazon-0 Mile	log(PagesPerDollar + 1) Amazon-0 Mile	log(MinsPerDollar + 1) Amazon-0 Mile
	(1)	(2)	(3)
β_1	1.910	0.054	0.372
	(1.061)	(0.348)	(0.256)
β_2	-1.699	-1.048^{*}	-1.310^{***}
	(1.190)	(0.381)	(0.307)
Observations	107	107	107
R^2	0.078	0.034	0.059
Adjusted R^2	-11.213	-2.658	-2.562
F Statistic	0.340 (df = 2; 8)	0.488 (df = 2; 28)	0.880 (df = 2; 28)
Note:			*p<0.01; **p<0.001; ***p<0.0001

Codes for generating Table 14 are listed below.

```
# Table 14
library(lmtest)
library(sandwich)

# Create Baseline
ama.0m.t14.sale.base <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t13[(data_0m_t13$domain_name == "amazon.com") & (data_0m_
      t13$Zipmatch == 1),], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
ama.0m.t14.ppd.base  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t13[(data_0m_t13$domain_name == "amazon.com") & (data_0m_t13
      $Zipmatch == 1),], index = c("Zip_Code"), model = "within", effect = "twoways")
ama.0m.t14.mpd.base  <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t13[(data_0m_t13$domain_name == "amazon.com") & (data_0m_t13$
      Zipmatch == 1),], index = c("Zip_Code"), model = "within", effect = "twoways")

bb.0m.t14.sale.base <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t13[(data_0m_t13$domain_name == "bestbuy.com") & (data_0m_
      t13$Zipmatch == 1),], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
```

```
bb.0m.t14.ppd.base  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t13[(data_0m_t13$domain_name == "bestbuy.com") & (data_0m_t13$Zipmatch == 1),], index = c("Zip_Code"), model = "within", effect = "twoways")
bb.0m.t14.mpd.base  <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t13[(data_0m_t13$domain_name == "bestbuy.com") & (data_0m_t13$Zipmatch == 1),], index = c("Zip_Code"), model = "within", effect = "twoways")

# Correlation
coeftest(ama.0m.t14.sale.base, vcovDC)
coeftest(ama.0m.t14.ppd.base, vcovDC)
coeftest(ama.0m.t14.mpd.base, vcovDC)
```

Code 15: Table 14 Generation

2.15 Table C1

Table 15 shows

Table 15: Change in Demographics after Circuit City Store Closure

Group	Before Store Closure			After Store Closure			First Difference of Mean (p-value)		
	Mean Age	Mean Income	Mean Education	Mean Age	Mean Income	Mean Education	Mean Age	Mean Income	Mean Education
Control	7.048	4.479	97.957	6.937	4.498	97.999	-0.111 (<0.0001)	0.019 (0.300)	0.042 (0.639)
Treated	7.68	4.971	98.632	6.645	4.739	96.843	-1.035 (<0.0001)	-0.232 (0.029)	-1.789 (0.004)

Codes for generating Table 15 are listed below.

```
temp <- read_sas(sales_allother_zipcode_path)
temp$Store_Close_Status <- 0
table_C1_0m_raw <-  rbind(temp, read_sas(sales_cc_0mile_path))
table_C1_5m_raw <-  rbind(temp, read_sas(sales_cc_5miles_path))

# Date Transform
table_C1_0m_raw$event_date <- as.Date(table_C1_0m_raw$event_date)
table_C1_5m_raw$event_date <- as.Date(table_C1_5m_raw$event_date)

# construct MonthYear - month of year
table_C1_0m_raw$MonthYear <- format(table_C1_0m_raw$event_date, "%Y-%m")
table_C1_5m_raw$MonthYear <- format(table_C1_5m_raw$event_date, "%Y-%m")

# Mark CC Closure

# CCStorePresent
# it is the same as Store_Close_Status
table_C1_0m_raw$CCStorePresent <- table_C1_0m_raw$Store_Close_Status
table_C1_5m_raw$CCStorePresent <- table_C1_5m_raw$Store_Close_Status

# AfterStoreClosing
table_C1_0m_raw$AfterStoreClosing <- ifelse(table_C1_0m_raw$MonthYear < "2008-11", 0, 1)
table_C1_5m_raw$AfterStoreClosing <- ifelse(table_C1_5m_raw$MonthYear < "2008-11", 0, 1)

# BBStorePresent
table_C1_0m_raw <- merge(table_C1_0m_raw, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)
table_C1_5m_raw <- merge(table_C1_5m_raw, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)

```

```r
table_C1_0m_raw$BBStorePresent <- na.fill(table_C1_0m_raw$BB_Store_Status, 0)
table_C1_5m_raw$BBStorePresent <- na.fill(table_C1_5m_raw$BB_Store_Status, 0)

# t test
control_before_age    <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 0)&(table_C1_0m_raw$AfterStoreClosing==0),]$hoh_oldest_age
control_before_income <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 0)&(table_C1_0m_raw$AfterStoreClosing==0),]$household_income
control_before_edu    <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 0)&(table_C1_0m_raw$AfterStoreClosing==0),]$hoh_most_education

control_after_age    <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 0)&(table_C1_0m_raw$AfterStoreClosing==1),]$hoh_oldest_age
control_after_income <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 0)&(table_C1_0m_raw$AfterStoreClosing==1),]$household_income
control_after_edu    <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 0)&(table_C1_0m_raw$AfterStoreClosing==1),]$hoh_most_education

test.control.age <- t.test(control_before_age, control_after_age)
test.control.income <- t.test(control_before_income, control_after_income)
test.control.edu <- t.test(control_before_edu, control_after_edu)

treated_before_age    <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 1)&(table_C1_0m_raw$AfterStoreClosing==0),]$hoh_oldest_age
treated_before_income <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 1)&(table_C1_0m_raw$AfterStoreClosing==0),]$household_income
treated_before_edu    <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 1)&(table_C1_0m_raw$AfterStoreClosing==0),]$hoh_most_education

treated_after_age    <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 1)&(table_C1_0m_raw$AfterStoreClosing==1),]$hoh_oldest_age
treated_after_income <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 1)&(table_C1_0m_raw$AfterStoreClosing==1),]$household_income
treated_after_edu    <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 1)&(table_C1_0m_raw$AfterStoreClosing==1),]$hoh_most_education

test.treated.age <- t.test(treated_before_age, treated_after_age)
test.treated.income <- t.test(treated_before_income, treated_after_income)
test.treated.edu <- t.test(treated_before_edu, treated_after_edu)

# Construct Variables
test.control.age.before.mean    <- test.control.age$estimate[["mean of x"]]
test.control.income.before.mean <- test.control.income$estimate[["mean of x"]]
test.control.edu.before.mean    <- test.control.edu$estimate[["mean of x"]]

test.control.age.after.mean    <- test.control.age$estimate[["mean of y"]]
test.control.income.after.mean <- test.control.income$estimate[["mean of y"]]
test.control.edu.after.mean    <- test.control.edu$estimate[["mean of y"]]

test.control.age.diff    <- test.control.age.after.mean - test.control.age.before.mean
test.control.age.diff.se <- test.control.age$p.value
test.control.income.diff    <- test.control.income.after.mean - test.control.income.before.mean
test.control.income.diff.se <- test.control.income$p.value
test.control.edu.diff    <- test.control.edu.after.mean - test.control.edu.before.mean
test.control.edu.diff.se <- test.control.edu$p.value

test.treated.age.before.mean    <- test.treated.age$estimate[["mean of x"]]
test.treated.income.before.mean <- test.treated.income$estimate[["mean of x"]]
test.treated.edu.before.mean    <- test.treated.edu$estimate[["mean of x"]]

test.treated.age.after.mean    <- test.treated.age$estimate[["mean of y"]]
test.treated.income.after.mean <- test.treated.income$estimate[["mean of y"]]
test.treated.edu.after.mean    <- test.treated.edu$estimate[["mean of y"]]

test.treated.age.diff    <- test.treated.age.after.mean - test.treated.age.before.mean
test.treated.age.diff.se <- test.treated.age$p.value
test.treated.income.diff    <- test.treated.income.after.mean - test.treated.income.before.mean
test.treated.income.diff.se <- test.treated.income$p.value
test.treated.edu.diff    <- test.treated.edu.after.mean - test.treated.edu.before.mean
test.treated.edu.diff.se <- test.treated.edu$p.value

# Construct Table
tabc1 <- rbind(c("Control", test.control.age.before.mean, test.control.income.before.mean, test.control.edu.before.mean,
                 test.control.age.after.mean, test.control.income.after.mean, test.control.edu.after.mean,
                 test.control.age.diff, test.control.age.diff.se, test.control.income.diff, test.control.income.diff.se, test.control.edu.diff,
      test.control.edu.diff.se),
      c("Treated", test.treated.age.before.mean, test.treated.income.before.mean, test.treated.edu.before.mean,
                 test.treated.age.after.mean, test.treated.income.after.mean, test.treated.edu.after.mean,
                 test.treated.age.diff, test.treated.age.diff.se, test.treated.income.diff, test.treated.income.diff.se, test.treated.edu.diff,
      test.treated.edu.diff.se))
```

Code 16: Table C1 Generation

2.16 Table D1-D4

2.17 Table E1-E2

2.18 Table G1-G3

3 Advanced Method

3.1 Generalized Synthetic Control Method

3.2 PSM and LA-PSM

3.3 Causal Forest

4 References