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

In this section, we present our codes for data cleaning and panel data preparation. Package `dplyr` Wickham et al. (2015), `haven` Wickham and Miller (2018), `sqldf` Grothendieck (2017), `zoo` Zeileis and Grothendieck (2005), `plm` Croissant et al. (2008), are used in this process.

Notice that the provided data only contain users' browsing transactions with purchase. When we apply search engine filter on reference domain, besides google.com, yahoo.com, bing.com, some other search engines (msn.com, aol.com, live.com, mywebsearch.com) and five vendors (amazon.com, staples.com, dell.com, walmart.com, bestbuy.com) are also involved. For product categories, we only consider those sold at Circuit City and exclude three other types (Business machines, Office furniture, Office supplies).

For `CCStorePresent`, its value is the same as `Store_Close_Status` in original data. And we set `AfterStoreClosing` to 0 if the time of user transaction is before November, 2008 and 1 otherwise. For `BBStorePresent`, we set it as 1 for user transactions if there's any Best Buy stores around user's location and 0 otherwise, using the `bestbuyzipcodes_sample` dataset. `NoReferringDomain` is the variable we construct if the user directly came to the target website without any referring domain. And `ReferringDomainIsSearchEngine` is the variable we construct if the user was referred by search engine to the target website.

Finally, two panel data are constructed by concatenating `sales_allotherzipcode` with `sales_ccity0mile` and `sales_ccity5mile`. After the concatenation and aggregation, we found that the built panel data are unbalanced, in a sense that, for instance, `zip_code` 75080 only has 2 records, instead of 24 (2 years). It happens because (a) the provided data are a small sample from the whole original one; (b) the original data might not cover the full 2 years period. Unbalanced panel data has been studied by many researchers Baltagi and Song (2006), like unbalanced seemingly unrelated regression McDowell (2004). Here we adopt a naive solution: we impute the missing values for one `zip_code` by averaging those non-missing values of this `zip_code`.

```r
1  # load library
2  library('dplyr')
3  library('haven')
4  library('sqldf')
5  library('zoo')
6  library('plm')
7  library('stargazer')
8
9  # all data path
10 bb_zipcode_path <- 'data/bestbuyzipcodes_sample.sas7bdat'
11 sales_allother_zipcode_path <- 'data/sales_allotherzipcode_sample.sas7bdat'
12 sales_cc_0mile_path <- 'data/sales_ccity0milezipcode_sample.sas7bdat'
13 sales_cc_5miles_path <- 'data/sales_ccity5milezipcode_sample.sas7bdat'
14
15 # load data
16 bb_zipcode <- read_sas(bb_zipcode_path)
17 sales_allother_zipcode <- read_sas(sales_allother_zipcode_path)
18 sales_cc_0mile <- read_sas(sales_cc_0mile_path)
19 sales_cc_5miles <- read_sas(sales_cc_5miles_path)
20
21 # Data Mapping
22 sales_allother_zipcode$Store_Close_Status <- 0 # NaN means no CC in 5-miles radius, we change NaN to 0
23
24 # Exclude Data without purchase
25 # All data should be with purchase -> tran_flg == 1
```

```r
26 sales_allother_zipcode <- sales_allother_zipcode[sales_allother_zipcode$tran_flg == 1,]
27 sales_cc_0mile <- sales_cc_0mile[sales_cc_0mile$tran_flg == 1,]
28 sales_cc_5miles <- sales_cc_5miles[sales_cc_5miles$tran_flg == 1,]
29
30 # Filter Referring Domain
31
32 # groupby ref_domain and count
33 groupby_ref_domain_result <- aggregate(machine_id ~ ref_domain_name, rbind(sales_allother_zipcode, sales_cc_0mile, sales_cc_5miles), FUN = "length"
        )
34 groupby_ref_domain_result <- groupby_ref_domain_result[order(-groupby_ref_domain_result$machine_id), ]
35 # we identify some search engines
36 search_engine_to_consider1 <- c("GOOGLE.COM", "YAHOO.COM", "google.com", "yahoo.com",
37                               "MSN.COM", "msn.com", "aol.com", "AOL.COM", "LIVE.COM", "live.com",
38                               "MYWEBSEARCH.COM", "ASK.COM", "MYWAY.COM", "mywebsearch.com",
39                               "ask.com", "YAHOO.NET", "BIZRATE.COM", "bizrate.com",
40                               "amazon.com", "staples.com", "dell.com", "walmart.com", "bestbuy.com",
41                               "AMAZON.COM", "STAPLES.COM", "DELL.COM", "WALMART.COM", "BESTBUY.COM")
42
43 search_engine_to_consider2 <- c("GOOGLE.COM", "YAHOO.COM", "BING.COM", "google.com", "yahoo.com", "bing.com")
44
45 ref_domain_to_consider1 <- c("", "GOOGLE.COM", "YAHOO.COM", "google.com", "yahoo.com",
46                               "MSN.COM", "msn.com", "aol.com", "AOL.COM", "LIVE.COM", "live.com",
47                               "MYWEBSEARCH.COM", "ASK.COM", "MYWAY.COM", "mywebsearch.com",
48                               "ask.com", "YAHOO.NET", "BIZRATE.COM", "bizrate.com",
49                               "amazon.com", "staples.com", "dell.com", "walmart.com", "bestbuy.com",
50                               "AMAZON.COM", "STAPLES.COM", "DELL.COM", "WALMART.COM", "BESTBUY.COM")
51
52 ref_domain_to_consider2 <- c("", "GOOGLE.COM", "YAHOO.COM", "BING.COM", "google.com", "yahoo.com", "bing.com")
53
54 # Then we filter data by refer domain name
55 sales_allother_zipcode <- sales_allother_zipcode[(sales_allother_zipcode$ref_domain_name %in% ref_domain_to_consider1),]
56 sales_cc_0mile <- sales_cc_0mile[(sales_cc_0mile$ref_domain_name %in% ref_domain_to_consider1),]
57 sales_cc_5miles <- sales_cc_5miles[(sales_cc_5miles$ref_domain_name %in% ref_domain_to_consider1),]
58
59 # Filter Target Domain Name
60 groupby_target_domain_result <- aggregate(machine_id ~ domain_name, rbind(sales_allother_zipcode, sales_cc_5miles), FUN = "length")
61 groupby_target_domain_result <- groupby_target_domain_result[order(-groupby_target_domain_result$machine_id), ]
62 five_target_domain_to_consider <- c("amazon.com", "staples.com", "dell.com", "walmart.com", "bestbuy.com")
63 two_target_domain_to_consider <- c("amazon.com","bestbuy.com")
64
65 # we can choose what filter to apply
66 sales_allother_zipcode <- sales_allother_zipcode[sales_allother_zipcode$domain_name %in% five_target_domain_to_consider,]
67 sales_cc_0mile <- sales_cc_0mile[sales_cc_0mile$domain_name %in% five_target_domain_to_consider,]
68 sales_cc_5miles <- sales_cc_5miles[sales_cc_5miles$domain_name %in% five_target_domain_to_consider,]
69
70 # Product Categories
71 # 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
72 # Jay removed 28, 30, 39, 40
73 # We choose to remove 38 39 40
74 sort(unique(rbind(sales_allother_zipcode, sales_cc_0mile, sales_cc_5miles)$prod_category_id))
75 category_to_consider <- c(22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37)
76 experience_product <- c(24, 25, 26, 27, 28, 31, 32, 33, 34, 36, 37)
77 search_product <- c(22, 23, 24, 29, 30, 35)
78
79 sales_allother_zipcode <- sales_allother_zipcode[sales_allother_zipcode$prod_category_id %in% category_to_consider,]
80 sales_cc_0mile <- sales_cc_0mile[sales_cc_0mile$prod_category_id %in% category_to_consider,]
81 sales_cc_5miles <- sales_cc_5miles[sales_cc_5miles$prod_category_id %in% category_to_consider,]
82
83 # Date Transform
84 sales_allother_zipcode$event_date <- as.Date(sales_allother_zipcode$event_date)
85 sales_cc_0mile$event_date <- as.Date(sales_cc_0mile$event_date)
86 sales_cc_5miles$event_date <- as.Date(sales_cc_5miles$event_date)
87
88 # construct MonthYear - month of year
89 sales_allother_zipcode$MonthYear <- format(sales_allother_zipcode$event_date, "%Y-%m")
90 sales_cc_0mile$MonthYear <- format(sales_cc_0mile$event_date, "%Y-%m")
91 sales_cc_5miles$MonthYear <- format(sales_cc_5miles$event_date, "%Y-%m")
92
93 # Mark CC Closure
94
95 # CCStorePresent
96 # it is the same as Store_Close_Status
```

```r
97  sales_allother_zipcode$CCStorePresent <- sales_allother_zipcode$Store_Close_Status
98  sales_cc_0mile$CCStorePresent <- sales_cc_0mile$Store_Close_Status
99  sales_cc_5miles$CCStorePresent <- sales_cc_5miles$Store_Close_Status
100
101 # AfterStoreClosing
102 sales_allother_zipcode$AfterStoreClosing <- ifelse(sales_allother_zipcode$MonthYear < "2008-11", 0, 1)
103 sales_cc_0mile$AfterStoreClosing <- ifelse(sales_cc_0mile$MonthYear < "2008-11", 0, 1)
104 sales_cc_5miles$AfterStoreClosing <- ifelse(sales_cc_5miles$MonthYear < "2008-11", 0, 1)
105
106 # BBStorePresent
107 sales_allother_zipcode <- merge(sales_allother_zipcode, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)
108 sales_cc_0mile <- merge(sales_cc_0mile, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)
109 sales_cc_5miles <- merge(sales_cc_5miles, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)
110
111 sales_allother_zipcode$BBStorePresent <- na.fill(sales_allother_zipcode$BB_Store_Status, 0)
112 sales_cc_0mile$BBStorePresent <- na.fill(sales_cc_0mile$BB_Store_Status, 0)
113 sales_cc_5miles$BBStorePresent <- na.fill(sales_cc_5miles$BB_Store_Status, 0)
114
115 # Mark Referring Domain
116 # Question: How to group data?
117 sales_allother_zipcode$NoReferringDomain <- ifelse(sales_allother_zipcode$ref_domain_name == "", 1, 0)
118 sales_cc_0mile$NoReferringDomain <- ifelse(sales_cc_0mile$ref_domain_name == "", 1, 0)
119 sales_cc_5miles$NoReferringDomain <- ifelse(sales_cc_5miles$ref_domain_name == "", 1, 0)
120
121 sales_allother_zipcode$ReferringDomainIsSearchEngine <- ifelse(sales_allother_zipcode$ref_domain_name %in% search_engine_to_consider1, 1, 0)
122 sales_cc_0mile$ReferringDomainIsSearchEngine <- ifelse(sales_cc_0mile$ref_domain_name %in% search_engine_to_consider1, 1, 0)
123 sales_cc_5miles$ReferringDomainIsSearchEngine <- ifelse(sales_cc_5miles$ref_domain_name %in% search_engine_to_consider1, 1, 0)
124
125 # Aggregate Data
126 concat_data1 <- rbind(sales_allother_zipcode, sales_cc_0mile)
127 concat_data2 <- rbind(sales_allother_zipcode, sales_cc_5miles)
128 concat_data1_exp <- concat_data1[concat_data1$prod_category_id %in% experience_product, ]
129 concat_data1_search <- concat_data1[concat_data1$prod_category_id %in% search_product, ]
130 concat_data2_exp <- concat_data2[concat_data2$prod_category_id %in% experience_product, ]
131 concat_data2_search <- concat_data2[concat_data2$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. Package `stargazer` Hlavac (2015) and `estout` Jann (2004) are used to export estimation into LaTeX format.

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.

```
1  # Table 1
2  table1_raw <-  rbind(read_sas(sales_allother_zipcode_path), read_sas(sales_cc_0mile_path))
3  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")
4  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.

```
1  # Table 2
2
3  table2_raw <-  rbind(read_sas(sales_allother_zipcode_path), read_sas(sales_cc_0mile_path))
4  table2_raw$direct_to_website <- ifelse(table2_raw$ref_domain_name == '', 1, 0)
5  table2_raw$referred_by_search <- ifelse(table2_raw$ref_domain_name %in% search_engine_to_consider1, 1, 0)
6  table2_raw$referred_by_other <- ifelse(!(table2_raw$ref_domain_name %in% ref_domain_to_consider1), 1, 0)
7  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.

```
1  # Table 3
2  temp <- read_sas(sales_allother_zipcode_path)
3  temp$Store_Close_Status <- 0
4  table3_0m_raw <-  rbind(temp, read_sas(sales_cc_0mile_path))
5  table3_5m_raw <-  rbind(temp, read_sas(sales_cc_5miles_path))
6
7  # Date Transform
8  table3_0m_raw$event_date <- as.Date(table3_0m_raw$event_date)
9  table3_5m_raw$event_date <- as.Date(table3_5m_raw$event_date)
10
11 # construct MonthYear - month of year
12 table3_0m_raw$MonthYear <- format(table3_0m_raw$event_date, "%Y-%m")
13 table3_5m_raw$MonthYear <- format(table3_5m_raw$event_date, "%Y-%m")
14
15 # Mark CC Closure
16
17 # CCStorePresent
```

```r
18  # it is the same as Store_Close_Status
19  table3_0m_raw$CCStorePresent <- table3_0m_raw$Store_Close_Status
20  table3_5m_raw$CCStorePresent <- table3_5m_raw$Store_Close_Status
21
22  # AfterStoreClosing
23  table3_0m_raw$AfterStoreClosing <- ifelse(table3_0m_raw$MonthYear < "2008-11", 0, 1)
24  table3_5m_raw$AfterStoreClosing <- ifelse(table3_5m_raw$MonthYear < "2008-11", 0, 1)
25
26  # BBStorePresent
27  table3_0m_raw <- merge(table3_0m_raw, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)
28  table3_5m_raw <- merge(table3_5m_raw, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)
29
30  table3_0m_raw$BBStorePresent <- na.fill(table3_0m_raw$BB_Store_Status, 0)
31  table3_5m_raw$BBStorePresent <- na.fill(table3_5m_raw$BB_Store_Status, 0)
32
33  # aggregate data
34
35  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")
36  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")
37
38  # Table 3 Gen Func
39  table3_gen <- function(table3_raw, domain_name_used, print_name){
40    # Amazon Sales
41    # for control
42    amazonsales_control_before <- table3_raw[(table3_raw$CCStorePresent == 0) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
        AfterStoreClosing == 0),]$TotalMonthlySales
43    amazonsales_control_after <- table3_raw[(table3_raw$CCStorePresent == 0) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
        AfterStoreClosing == 1),]$TotalMonthlySales
44
45    amazonsales_control_before <- log(amazonsales_control_before + 1)
46    amazonsales_control_after <- log(amazonsales_control_after + 1)
47    # t test
48    t_test.amazonsales_control <- t.test(amazonsales_control_after, amazonsales_control_before)
49    amazonsales_control_mean_diff_se <- t_test.amazonsales_control$stderr
50    t_test.amazonsales_control$p.value
51    amazonsales_control_after_mean <- t_test.amazonsales_control$estimate[["mean of x"]]
52    amazonsales_control_before_mean <- t_test.amazonsales_control$estimate[["mean of y"]]
53    amazonsales_control_mean_diff <- t_test.amazonsales_control$estimate[["mean of x"]] - t_test.amazonsales_control$estimate[["mean of y"]]
54
55    # Amazon Sales
56    # for treatment
57    amazonsales_treatment_before <- table3_raw[(table3_raw$CCStorePresent == 1) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
        AfterStoreClosing == 0),]$TotalMonthlySales
58    amazonsales_treatment_after <- table3_raw[(table3_raw$CCStorePresent == 1) & (table3_raw$domain_name == domain_name_used) & (table3_raw$
        AfterStoreClosing == 1),]$TotalMonthlySales
59
60    amazonsales_treatment_before <- log(amazonsales_treatment_before + 1)
61    amazonsales_treatment_after  <- log(amazonsales_treatment_after + 1)
62    # t test
63    t_test.amazonsales_treatment <- t.test(amazonsales_treatment_after, amazonsales_treatment_before)
64    amazonsales_treatment_mean_diff_se <- t_test.amazonsales_treatment$stderr
65    t_test.amazonsales_treatment$p.value
66    amazonsales_treatment_after_mean <- t_test.amazonsales_treatment$estimate[["mean of x"]]
67    amazonsales_treatment_before_mean <- t_test.amazonsales_treatment$estimate[["mean of y"]]
68    amazonsales_treatment_mean_diff <- t_test.amazonsales_treatment$estimate[["mean of x"]] - t_test.amazonsales_treatment$estimate[["mean of y"]]
69
70    # Amazon Sales DID
71    amazonsales_did <- amazonsales_treatment_mean_diff - amazonsales_control_mean_diff
72
73    # Amazon PagesPerDollar
74    # for control
75    amazonppd_control_before <- table3_raw[(table3_raw$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),]$TotalMonthlySales
76    amazonppd_control_after <- table3_raw[(table3_raw$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
 77
 78   amazonppd_control_before <- log(amazonppd_control_before + 1)
 79   amazonppd_control_after  <- log(amazonppd_control_after + 1)
 80   # t test
 81   t_test.amazonppd_control <- t.test(amazonppd_control_after, amazonppd_control_before)
 82   amazonppd_control_mean_diff_se <- t_test.amazonppd_control$stderr
 83   t_test.amazonppd_control$p.value
 84   amazonppd_control_after_mean <- t_test.amazonppd_control$estimate[["mean of x"]]
 85   amazonppd_control_before_mean <- t_test.amazonppd_control$estimate[["mean of y"]]
 86   amazonppd_control_mean_diff <- t_test.amazonppd_control$estimate[["mean of x"]] - t_test.amazonppd_control$estimate[["mean of y"]]
 87
 88   # Amazon PagesPerDollar
 89   # for treatment
 90   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),]$TotalMonthlySales
 91   amazonppd_treatment_after  <- table3_raw[(table3_raw$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),]$TotalMonthlySales
 92
 93   amazonppd_treatment_before <- log(amazonppd_treatment_before + 1)
 94   amazonppd_treatment_after  <- log(amazonppd_treatment_after + 1)
 95   # t test
 96   t_test.amazonppd_treatment <- t.test(amazonppd_treatment_after, amazonppd_treatment_before)
 97   amazonppd_treatment_mean_diff_se <- t_test.amazonppd_treatment$stderr
 98   t_test.amazonppd_treatment$p.value
 99   amazonppd_treatment_after_mean <- t_test.amazonppd_treatment$estimate[["mean of x"]]
100   amazonppd_treatment_before_mean <-t_test.amazonppd_treatment$estimate[["mean of y"]]
101   amazonppd_treatment_mean_diff <- t_test.amazonppd_treatment$estimate[["mean of x"]] - t_test.amazonppd_treatment$estimate[["mean of y"]]
102
103   # Amazon PagesPerDollar DID
104   amazonppd_did <- amazonppd_treatment_mean_diff - amazonppd_control_mean_diff
105
106   # Amazon MinsPerDollar
107   # for control
108   amazonmpd_control_before <- table3_raw[(table3_raw$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),]$TotalMonthlySales
109   amazonmpd_control_after  <- table3_raw[(table3_raw$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),]$TotalMonthlySales
110
111   amazonmpd_control_before <- log(amazonmpd_control_before + 1)
112   amazonmpd_control_after  <- log(amazonmpd_control_after + 1)
113   # t test
114   t_test.amazonmpd_control <- t.test(amazonmpd_control_after, amazonmpd_control_before)
115   amazonmpd_control_mean_diff_se <- t_test.amazonmpd_control$stderr
116   t_test.amazonmpd_control$p.value
117   amazonmpd_control_after_mean <- t_test.amazonmpd_control$estimate[["mean of x"]]
118   amazonmpd_control_before_mean <- t_test.amazonmpd_control$estimate[["mean of y"]]
119   amazonmpd_control_mean_diff <- t_test.amazonmpd_control$estimate[["mean of x"]] - t_test.amazonmpd_control$estimate[["mean of y"]]
120
121   # Amazon MinsPerDollar
122   # for treatment
123   amazonmpd_treatment_before <- table3_raw[(table3_raw$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),]$TotalMonthlySales
124   amazonmpd_treatment_after  <- table3_raw[(table3_raw$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),]$TotalMonthlySales
125
126   amazonmpd_treatment_before <- log(amazonmpd_treatment_before + 1)
127   amazonmpd_treatment_after  <- log(amazonmpd_treatment_after + 1)
128   # t test
129   t_test.amazonmpd_treatment <- t.test(amazonmpd_treatment_after, amazonmpd_treatment_before)
130   amazonmpd_treatment_mean_diff_se <- t_test.amazonmpd_treatment$stderr
131   t_test.amazonmpd_treatment$p.value
132   amazonmpd_treatment_after_mean <- t_test.amazonmpd_treatment$estimate[["mean of x"]]
133   amazonmpd_treatment_before_mean <- t_test.amazonmpd_treatment$estimate[["mean of y"]]
134   amazonmpd_treatment_mean_diff <- t_test.amazonmpd_treatment$estimate[["mean of x"]] - t_test.amazonmpd_treatment$estimate[["mean of y"]]
135
```

```
136    # Amazon MinsPerDollar DID
137    amazonmpd_did <- amazonmpd_treatment_mean_diff - amazonmpd_control_mean_diff
138
139    # construct table
140    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),
141             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),
142             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),
143             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),
144             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),
145             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))
146      )
147  }
148
149  # generate table
150  amazon_table3 <- table3_gen(table3_0m_aggregate, "amazon.com", "Amazon")
151  bestbuy_table3 <- table3_gen(table3_0m_aggregate, "bestbuy.com", "bestbuy.com")
152
153  #
154  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

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	0.014	-0.005	-0.002	-0.002
	(0.015)	(0.008)	(0.033)	(0.008)
β_2	-0.033	0.003	0.009	0.002
	(0.022)	(0.010)	(0.036)	(0.010)
Observations	68,472	75,096	14,664	16,848
R^2	0.00003	0.00001	0.00002	0.00000
Adjusted R^2	-0.044	-0.044	-0.045	-0.045
F Statistic	1.091 (df = 2; 65594)	0.278 (df = 2; 71942)	0.154 (df = 2; 14028)	0.035 (df = 2; 16121)

Note: $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$

Codes for generating Table 4 are listed below.

```
1  # Table 4 Data
2  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")
3  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")
4  # manually construct DID and THREEINTERACTION
5  data_0m_t4$DID <- data_0m_t4$CCStorePresent * data_0m_t4$AfterStoreClosing
6  data_0m_t4$THREEINTER <- data_0m_t4$CCStorePresent * data_0m_t4$AfterStoreClosing * data_0m_t4$BBStorePresent
7  data_5m_t4$DID <- data_5m_t4$CCStorePresent * data_5m_t4$AfterStoreClosing
8  data_5m_t4$THREEINTER <- data_5m_t4$CCStorePresent * data_5m_t4$AfterStoreClosing * data_5m_t4$BBStorePresent
9  # Table 4
10 ama.t4.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t4[data_0m_t4$domain_name == "amazon.com",], index = c("Zip_Code"
       , "MonthYear"), model = "within", effect = "twoways")
11 ama.t4.5mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_5m_t4[data_5m_t4$domain_name == "amazon.com",], index = c("Zip_Code"
       , "MonthYear"), model = "within", effect = "twoways")
12 bb.t4.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t4[data_0m_t4$domain_name == "bestbuy.com",], index = c("Zip_Code"
       , "MonthYear"), model = "within", effect = "twoways")
13 bb.t4.5mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_5m_t4[data_5m_t4$domain_name == "bestbuy.com",], index = c("Zip_Code"
       , "MonthYear"), model = "within", effect = "twoways")
```

Code 5: Table 4 Generation

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	0.003	-0.019^{***}	0.001	0.002	0.004	-0.021^{***}	0.001	0.003
	(0.012)	(0.007)	(0.016)	(0.004)	(0.012)	(0.007)	(0.013)	(0.003)
β_2	-0.068^{***}	0.018^{**}	0.003	-0.001	-0.057^{***}	0.022^{***}	0.0004	-0.002
	(0.018)	(0.009)	(0.018)	(0.005)	(0.017)	(0.008)	(0.014)	(0.004)
Observations	68,472	75,096	14,664	16,848	68,472	75,096	14,664	16,848
R^2	0.0004	0.0001	0.00003	0.00004	0.0003	0.0001	0.00001	0.0001
Adjusted R^2	-0.043	-0.044	-0.045	-0.045	-0.044	-0.044	-0.045	-0.045
F Statistic	12.530^{***} (df = 2; 65594)	3.985^{**} (df = 2; 71942)	0.202 (df = 2; 14028)	0.337 (df = 2; 16121)	8.867^{***} (df = 2; 65594)	5.187^{***} (df = 2; 71942)	0.046 (df = 2; 14028)	0.451 (df = 2; 16121)

Note: $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$

Codes for generating Table 5 are listed below.

```
1  # Table 5 Data
2  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")
3  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")
4  # manually construct DID and THREEINTERACTION
5  data_0m_t5$DID <- data_0m_t5$CCStorePresent * data_0m_t5$AfterStoreClosing
6  data_0m_t5$THREEINTER <- data_0m_t5$CCStorePresent * data_0m_t5$AfterStoreClosing * data_0m_t5$BBStorePresent
7  data_5m_t5$DID <- data_5m_t5$CCStorePresent * data_5m_t5$AfterStoreClosing
8  data_5m_t5$THREEINTER <- data_5m_t5$CCStorePresent * data_5m_t5$AfterStoreClosing * data_5m_t5$BBStorePresent
9  # Table 5
10 # For PagesPerDollar
11 ama.t5.pagesperdollar.0mile <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t5[data_0m_t5$domain_name == "amazon.com",], index =
         c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
12 ama.t5.pagesperdollar.5mile <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t5[data_5m_t5$domain_name == "amazon.com",], index =
         c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
13 bb.t5.pagesperdollar.0mile  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t5[data_0m_t5$domain_name == "bestbuy.com",], index =
         c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
14 bb.t5.pagesperdollar.5mile  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t5[data_5m_t5$domain_name == "bestbuy.com",], index =
         c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
15 # For MinsPerDollar
16 ama.t5.minsperdollar.0mile <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t5[data_0m_t5$domain_name == "amazon.com",], index = c(
         "Zip_Code", "MonthYear"), model = "within", effect = "twoways")
17 ama.t5.minsperdollar.5mile <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t5[data_5m_t5$domain_name == "amazon.com",], index = c(
         "Zip_Code", "MonthYear"), model = "within", effect = "twoways")
18 bb.t5.minsperdollar.0mile  <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t5[data_0m_t5$domain_name == "bestbuy.com",], index = c(
         "Zip_Code", "MonthYear"), model = "within", effect = "twoways")
19 bb.t5.minsperdollar.5mile  <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t5[data_5m_t5$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

	Amazon-0 Mile-Exp	Amazon-5 Miles-Exp	Amazon-0 Mile-Search	Amazon-5 Miles-Search	BestBuy-0 Mile-Exp	BestBuy-5 Miles-Exp	BestBuy-0 Mile-Exp	BestBuy-5 Miles-Search
				log(TotalMonthlySales + 1)				
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
β_1	0.005	-0.007	0.005	-0.008	-0.011	-0.009	-0.001	-0.010
	(0.017)	(0.010)	(0.013)	(0.006)	(0.009)	(0.007)	(0.023)	(0.008)
β_2	-0.043^*	0.009	-0.002	0.009		0.013	0.000	0.009
	(0.024)	(0.012)	(0.018)	(0.008)		(0.008)	(0.028)	(0.010)
Observations	32,112	35,568	52,392	57,648	10,224	11,712	5,664	6,600
R^2	0.0002	0.00002	0.00000	0.00003	0.0001	0.0002	0.00000	0.0002
Adjusted R^2	-0.044	-0.044	-0.044	-0.044	-0.046	-0.045	-0.048	-0.047
F Statistic	2.775^* (df = 2; 30749)	0.318 (df = 2; 34061)	0.101 (df = 2; 50184)	0.774 (df = 2; 55221)	1.377 (df = 1; 9774)	1.297 (df = 2; 11199)	0.004 (df = 2; 5403)	0.746 (df = 2; 6300)

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

Codes for generating Table 6 are listed below.

```
1  # Table 6 Data
2  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")
3  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")
4  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")
5  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")
6  # manually construct DID and THREEINTERACTION
7  data_0m_t6_exp$DID <- data_0m_t6_exp$CCStorePresent * data_0m_t6_exp$AfterStoreClosing
8  data_0m_t6_exp$THREEINTER <- data_0m_t6_exp$CCStorePresent * data_0m_t6_exp$AfterStoreClosing * data_0m_t6_exp$BBStorePresent
9  data_0m_t6_search$DID <- data_0m_t6_search$CCStorePresent * data_0m_t6_search$AfterStoreClosing
10 data_0m_t6_search$THREEINTER <- data_0m_t6_search$CCStorePresent * data_0m_t6_search$AfterStoreClosing * data_0m_t6_search$BBStorePresent
11 data_5m_t6_exp$DID <- data_5m_t6_exp$CCStorePresent * data_5m_t6_exp$AfterStoreClosing
12 data_5m_t6_exp$THREEINTER <- data_5m_t6_exp$CCStorePresent * data_5m_t6_exp$AfterStoreClosing * data_5m_t6_exp$BBStorePresent
13 data_5m_t6_search$DID <- data_5m_t6_search$CCStorePresent * data_5m_t6_search$AfterStoreClosing
14 data_5m_t6_search$THREEINTER <- data_5m_t6_search$CCStorePresent * data_5m_t6_search$AfterStoreClosing * data_5m_t6_search$BBStorePresent
15 # Table 6
16 # AmazonTotalMonthlySales & BBTotalMonthlySale vs Experience and Search Product
17 ama.t6.0mile.exp <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t6_exp[data_0m_t6_exp$domain_name == "amazon.com",], index =
       c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
18 ama.t6.5mile.exp <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_5m_t6_exp[data_5m_t6_exp$domain_name == "amazon.com",], index =
       c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
19 ama.t6.0mile.search <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t6_search[data_0m_t6_search$domain_name == "amazon.com",],
       index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
20 ama.t6.5mile.search <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_5m_t6_search[data_5m_t6_search$domain_name == "amazon.com",],
       index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
21 bb.t6.0mile.exp <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t6_exp[data_0m_t6_exp$domain_name == "bestbuy.com",], index =
       c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
22 bb.t6.5mile.exp <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_5m_t6_exp[data_5m_t6_exp$domain_name == "bestbuy.com",], index =
       c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
23 bb.t6.0mile.search <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t6_search[data_0m_t6_search$domain_name == "bestbuy.com",],
       index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
24 bb.t6.5mile.search <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_5m_t6_search[data_5m_t6_search$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	0.007	-0.037^{***}	0.006^{**}	0.001	0.006	-0.039^{***}	0.003	0.001
	(0.015)	(0.008)	(0.002)	(0.002)	(0.015)	(0.008)	(0.002)	(0.001)
β_2	-0.077^{***}	0.030^{***}		-0.0001	-0.067^{***}	0.034^{***}		-0.001
	(0.020)	(0.010)		(0.002)	(0.020)	(0.010)		(0.002)
Observations	32,112	35,568	10,224	11,712	32,112	35,568	10,224	11,712
R^2	0.001	0.001	0.001	0.00003	0.001	0.001	0.0003	0.0001
Adjusted R^2	-0.043	-0.044	-0.045	-0.046	-0.044	-0.044	-0.046	-0.046
F Statistic	12.857^{***} (df = 2; 30749)	10.009^{***} (df = 2; 34061)	5.763^{**} (df = 1; 9774)	0.143 (df = 2; 11199)	10.349^{***} (df = 2; 30749)	11.626^{***} (df = 2; 34061)	2.508 (df = 1; 9774)	0.438 (df = 2; 11199)

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

Codes for generating Table 7 are listed below.

```
1  # Table 7 & 8 Data
2  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")
3  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")
4  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")
5  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")
6  # manually construct DID and THREEINTERACTION
7  data_0m_t7_exp$DID <- data_0m_t7_exp$CCStorePresent * data_0m_t7_exp$AfterStoreClosing
8  data_0m_t7_exp$THREEINTER <- data_0m_t7_exp$CCStorePresent * data_0m_t7_exp$AfterStoreClosing * data_0m_t7_exp$BBStorePresent
9  data_0m_t8_search$DID <- data_0m_t8_search$CCStorePresent * data_0m_t8_search$AfterStoreClosing
10 data_0m_t8_search$THREEINTER <- data_0m_t8_search$CCStorePresent * data_0m_t8_search$AfterStoreClosing * data_0m_t8_search$BBStorePresent
11 data_5m_t7_exp$DID <- data_5m_t7_exp$CCStorePresent * data_5m_t7_exp$AfterStoreClosing
12 data_5m_t7_exp$THREEINTER <- data_5m_t7_exp$CCStorePresent * data_5m_t7_exp$AfterStoreClosing * data_5m_t7_exp$BBStorePresent
13 data_5m_t8_search$DID <- data_5m_t8_search$CCStorePresent * data_5m_t8_search$AfterStoreClosing
14 data_5m_t8_search$THREEINTER <- data_5m_t8_search$CCStorePresent * data_5m_t8_search$AfterStoreClosing * data_5m_t8_search$BBStorePresent
15 # Table 7
16 ama.t7.pagesperdollar.0mile.exp <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t7_exp[data_0m_t7_exp$domain_name == "amazon.com"
       ,], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
17 ama.t7.pagesperdollar.5mile.exp <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t7_exp[data_5m_t7_exp$domain_name == "amazon.com"
       ,], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
18 bb.t7.pagesperdollar.0mile.exp  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t7_exp[data_0m_t7_exp$domain_name == "bestbuy.com
       ",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
19 bb.t7.pagesperdollar.5mile.exp  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t7_exp[data_5m_t7_exp$domain_name == "bestbuy.com
       ",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
20 ama.t7.minsperdollar.0mile.exp  <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t7_exp[data_0m_t7_exp$domain_name == "amazon.com"
       ,], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
21 ama.t7.minsperdollar.5mile.exp  <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t7_exp[data_5m_t7_exp$domain_name == "amazon.com"
       ,], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
22 bb.t7.minsperdollar.0mile.exp   <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t7_exp[data_0m_t7_exp$domain_name == "bestbuy.com"
       ,], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
23 bb.t7.minsperdollar.5mile.exp   <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t7_exp[data_5m_t7_exp$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-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	0.001	0.006	0.001	0.009*	0.003	0.004	0.0001	0.009*
	(0.012)	(0.006)	(0.014)	(0.005)	(0.012)	(0.006)	(0.012)	(0.005)
β_2	−0.019	−0.002	−0.000	−0.007	−0.019	0.001	−0.000	−0.008
	(0.017)	(0.008)	(0.017)	(0.006)	(0.017)	(0.008)	(0.015)	(0.006)
Observations	52,392	57,648	5,664	6,600	52,392	57,648	5,664	6,600
R^2	0.00005	0.00002	0.00000	0.001	0.00004	0.00003	0.00000	0.001
Adjusted R^2	-0.044	-0.044	-0.048	-0.047	-0.044	-0.044	-0.048	-0.047
F Statistic	1.138 (df = 2; 50184)	0.553 (df = 2; 55221)	0.011 (df = 2; 5403)	1.590 (df = 2; 6300)	0.935 (df = 2; 50184)	0.696 (df = 2; 55221)	0.0001 (df = 2; 5403)	1.927 (df = 2; 6300)

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

Codes for generating Table 8 are listed below.

```
1  # Table 8
2  ama.t8.pagesperdollar.0mile.search <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t8_search[data_0m_t8_search$domain_name == "
       amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
3  ama.t8.pagesperdollar.5mile.search <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t8_search[data_5m_t8_search$domain_name == "
       amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
4  bb.t8.pagesperdollar.0mile.search  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t8_search[data_0m_t8_search$domain_name == "
       bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
5  bb.t8.pagesperdollar.5mile.search  <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t8_search[data_5m_t8_search$domain_name == "
       bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
6  ama.t8.minsperdollar.0mile.search  <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t8_search[data_0m_t8_search$domain_name == "
       amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
7  ama.t8.minsperdollar.5mile.search  <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t8_search[data_5m_t8_search$domain_name == "
       amazon.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
8  bb.t8.minsperdollar.0mile.search   <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t8_search[data_0m_t8_search$domain_name == "
       bestbuy.com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
9  bb.t8.minsperdollar.5mile.search   <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_5m_t8_search[data_5m_t8_search$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.

```
1  * build variables
2  gen DID =  CCStorePresent *  AfterStoreClosing
3  gen THREEINTER = DID *  BBStorePresent
4  egen Code_Time = group(Zip_Code MonthYear)
5
6  * Amazon - ReferringDomainIsSearchEngine & NoReferringDomain
7  eststo: logit ReferringDomainIsSearchEngine DID THREEINTER if domain_name == "amazon.com", vce(cluster Code_Time) noconstant
8  eststo: logit NoReferringDomain DID THREEINTER if domain_name == "amazon.com", vce(cluster Code_Time) noconstant
9  * BestBuy - ReferringDomainIsSearchEngine & NoReferringDomain
10 eststo: logit ReferringDomainIsSearchEngine DID THREEINTER if domain_name == "bestbuy.com", vce(cluster MonthYear) noconstant
11 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	0.012	-0.001	0.004	0.0002	0.006	0.0002
	(0.013)	(0.032)	(0.009)	(0.017)	(0.011)	(0.020)
β_2	-0.018	0.010	-0.021^*	0.005	-0.021	-0.003
	(0.019)	(0.034)	(0.013)	(0.018)	(0.016)	(0.021)
Observations	68,472	14,664	68,472	14,664	68,472	14,664
R^2	0.00002	0.00003	0.0001	0.00004	0.00003	0.00001
Adjusted R^2	-0.044	-0.045	-0.044	-0.045	-0.044	-0.045
F Statistic	0.539 (df = 2; 65594)	0.213 (df = 2; 14028)	1.867 (df = 2; 65594)	0.304 (df = 2; 14028)	0.939 (df = 2; 65594)	0.066 (df = 2; 14028)

Note: $^*p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$

Codes for generating Table 10 are listed below.

```
1  # Table 10 Data
2  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")
3
4  data_0m_t10$DID <- data_0m_t10$CCStorePresent * data_0m_t10$AfterStoreClosing
5  data_0m_t10$THREEINTER <- data_0m_t10$CCStorePresent * data_0m_t10$AfterStoreClosing * data_0m_t10$BBStorePresent
6  # Table 10
7  # SalesPerTransaction; PagesPerTransaction; MinsPerTransaction; for Ama & BB
8  ama.t10.0mile.SalesPerTransaction <- plm(log(SalesPerTransaction + 1) ~ DID + THREEINTER, data = data_0m_t10[data_0m_t10$domain_name == "amazon.com
       ",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
9  bb.t10.0mile.SalesPerTransaction  <- plm(log(SalesPerTransaction + 1) ~ DID + THREEINTER, data = data_0m_t10[data_0m_t10$domain_name == "bestbuy.
       com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
10 ama.t10.0mile.PagesPerTransaction <- plm(log(PagesPerTransaction + 1) ~ DID + THREEINTER, data = data_0m_t10[data_0m_t10$domain_name == "amazon.com
       ",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
11 bb.t10.0mile.PagesPerTransaction  <- plm(log(PagesPerTransaction + 1) ~ DID + THREEINTER, data = data_0m_t10[data_0m_t10$domain_name == "bestbuy.
       com",], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
12 ama.t10.0mile.MinsPerTransaction <- plm(log(MinsPerTransaction + 1) ~ DID + THREEINTER, data = data_0m_t10[data_0m_t10$domain_name == "amazon.com"
       ,], index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
13 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

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

| | log(TotalMonthlySales + 1) | | log(PagesPerDollar + 1) | | log(MinsPerDollar + 1) | |
| | Amazon-0 Mile | BesyBuy-0 Mile | Amazon-0 Mile | BesyBuy-0 Mile | Amazon-0 Mile | BesyBuy-0 Mile |
	(1)	(2)	(3)	(4)	(5)	(6)
β_1	0.019	−0.0002	0.006	−0.001	0.003	−0.0002
	(0.019)	(0.002)	(0.012)	(0.003)	(0.011)	(0.002)
β_2	−0.026		−0.023		−0.024*	
	(0.024)		(0.016)		(0.013)	
Observations	1,776	384	1,776	384	1,776	384
R^2	0.001	0.00002	0.001	0.0001	0.002	0.00003
Adjusted R^2	-0.058	-0.113	-0.057	-0.113	-0.056	-0.113
F Statistic	0.740 (df = 2; 1677)	0.008 (df = 1; 344)	1.183 (df = 2; 1677)	0.030 (df = 1; 344)	1.931 (df = 2; 1677)	0.012 (df = 1; 344)

Codes for generating Table 11 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")

#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()
```

```r
42
43  for (i in 1:length(mat$w)){
44    if (mat$w[i] == 1) zipcheck1 <-c(zipcheck1,i)
45  }
46
47  data.frame(zipcheck1)
48
49  # Test both dataframe, and they are same.
50  all.equal(zipcheck,zipcheck1)
51
52  # add specific Zipcode by mapping from ID of row of matched zipcode
53  ziplist <- c()
54  for (i in 1:length(data_0m_t11$Zip_Code)){
55    if ( i %in% zipcheck) ziplist <-c(ziplist,data_0m_t11$Zip_Code[i])
56  }
57
58  data.frame(ziplist)
59
60  # assign matched zipcode to dataset
61  concat_data1$Zipmatch <- ifelse(concat_data1$Zip_Code %in% ziplist, 1, 0)
62  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")
63  data_0m_t11$DID <- data_0m_t11$CCStorePresent * data_0m_t11$AfterStoreClosing
64  data_0m_t11$THREEINTER <- data_0m_t11$DID * data_0m_t11$BBStorePresent
65
66  # result for Amazon regarding TotalMonthlySales, PagesPerDollar, MinsPerDollar
67  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")
68  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")
69  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")
70  # result for Bestbuy regarding TotalMonthlySales, PagesPerDollar, MinsPerDollar
71  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")
72  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")
73  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

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	−0.00001	−0.00001	0.0001	0.00001	0.0001	0.00000
	(0.0001)	(0.0002)	(0.0001)	(0.0001)	(0.0001)	(0.0001)
β_2	−0.00001	−0.0001	−0.0002*	0.00002	−0.0002	0.00001
	(0.0002)	(0.0002)	(0.0001)	(0.0001)	(0.0001)	(0.0001)
Observations	68,472	14,664	68,472	14,664	68,472	14,664
R^2	0.00000	0.00004	0.00005	0.00002	0.00003	0.00001
Adjusted R^2	-0.044	-0.045	-0.044	-0.045	-0.044	-0.045
F Statistic	0.019 (df = 2; 65594)	0.255 (df = 2; 14028)	1.478 (df = 2; 65594)	0.114 (df = 2; 14028)	1.131 (df = 2; 65594)	0.053 (df = 2; 14028)

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

Codes for generating Table 12 are listed below.

```
1  # Table 12 Data
2  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")
3  data_0m_t12$DID <- data_0m_t12$CCStorePresent * data_0m_t12$AfterStoreClosing * data_0m_t12$HoHSize * data_0m_t12$HoHAge * data_0m_t12$HoHIncome
4  data_0m_t12$THREEINTER <- data_0m_t12$CCStorePresent * data_0m_t12$AfterStoreClosing * data_0m_t12$BBStorePresent
5  # Table 12
6  ama.t12.0m.PagesPerDollar <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t12[data_0m_t12$domain_name == "amazon.com",], index =
       c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
7  ama.t12.0m.MinsPerDollar <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t12[data_0m_t12$domain_name == "amazon.com",], index = c(
       "Zip_Code", "MonthYear"), model = "within", effect = "twoways")
8  ama.t12.0m.TotalMonthlySales <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t12[data_0m_t12$domain_name == "amazon.com",],
       index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
9  bb.t12.0m.PagesPerDollar <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t12[data_0m_t12$domain_name == "bestbuy.com",], index =
       c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
10 bb.t12.0m.MinsPerDollar <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_t12[data_0m_t12$domain_name == "bestbuy.com",], index = c(
       "Zip_Code", "MonthYear"), model = "within", effect = "twoways")
11 bb.t12.0m.TotalMonthlySales <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_t12[data_0m_t12$domain_name == "bestbuy.com",],
       index = c("Zip_Code", "MonthYear"), model = "within", effect = "twoways")
```

Code 13: Table 12 Generation

2.13   Table 13

Table 13: Results of the Online Sales and Search Effect After Matching Zip Codes: TotalMonthlySales, PagesPerDollar, and MinsPerDollar (All Product Categories)

	log(TotalMonthlySales + 1)		log(PagesPerDollar + 1)		log(MinsPerDollar + 1)	
	Amazon-0 Mile	BesyBuy-0 Mile	Amazon-0 Mile	BesyBuy-0 Mile	Amazon-0 Mile	BesyBuy-0 Mile
	(1)	(2)	(3)	(4)	(5)	(6)
β_1	0.023	−0.001	0.007	−0.003	0.009	−0.001
	(0.015)	(0.003)	(0.010)	(0.006)	(0.008)	(0.004)
β_2	−0.026		−0.023		−0.024*	
	(0.024)		(0.015)		(0.013)	
Observations	1,776	208	1,776	208	1,776	208
R^2	0.001	0.0003	0.001	0.001	0.002	0.0004
Adjusted R^2	-0.043	-0.083	-0.043	-0.083	-0.042	-0.083
F Statistic	1.169 (df = 2; 1700)	0.052 (df = 1; 191)	1.182 (df = 2; 1700)	0.197 (df = 1; 191)	1.663 (df = 2; 1700)	0.078 (df = 1; 191)

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

Codes for generating Table 13 are listed below.

```
1  library(cem)
2  #matching based on zipcode demographics (cross-sectional)
3  data_0m_t11 <- sqldf("SELECT Zip_Code, SUM(prod_totprice) AS TotalMonthlySales,
4                    AVG(CCStorePresent) AS CCStorePresent,
5                    AVG(household_size) AS HoHSize,
6                    AVG(hoh_oldest_age) AS HoHAge,
7                    AVG(household_income) AS HoHIncome,
8                    AVG(children) AS HoHChildren,
9                    AVG(connection_speed) AS HoHSpeed
10                   FROM concat_data1 GROUP BY Zip_Code")
11 # CEM
12 todrop2 <- c("TotalMonthlySales", "Zip_Code")
13 mat <- cem(treatment = "CCStorePresent",
14          data = data_0m_t11,
15          drop = todrop2,
16          k2k = TRUE,
17          method = "euclidean")
18 mat
19
20 # Check Matching
21 zipcheck <- c()
22
23 for (i in 1:length(mat$matched)){
24   if (mat$matched[i] == "TRUE") zipcheck <-c(zipcheck,i)
25 }
26
27 data.frame(zipcheck)
28
29 # assign ID of row value of zipcode from "w"
30 zipcheck1 <- c()
31
32 for (i in 1:length(mat$w)){
33   if (mat$w[i] == 1) zipcheck1 <-c(zipcheck1,i)
34 }
35
36 data.frame(zipcheck1)
37
38 # Test both dataframe, and they are same.
```

```
39  all.equal(zipcheck,zipcheck1)
40
41  # add specific Zipcode by mapping from ID of row of matched zipcode
42  ziplist <- c()
43  for (i in 1:length(data_0m_t11$Zip_Code)){
44    if ( i %in% zipcheck) ziplist <-c(ziplist,data_0m_t11$Zip_Code[i])
45  }
46
47  data.frame(ziplist)
48
49  # Assign matched zipcode to dataset
50  concat_data1$Zipmatch <- ifelse(concat_data1$Zip_Code %in% ziplist, 1, 0)
51  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")
52  data_0m_t13$DID <- data_0m_t13$CCStorePresent * data_0m_t13$AfterStoreClosing
53  data_0m_t13$THREEINTER <- data_0m_t13$DID * data_0m_t13$BBStorePresent
54
55  # Table 13
56  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")
57  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")
58  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")
59
60  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")
61  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")
62  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) | | 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	0.019	−0.001	0.006	−0.003	0.003	−0.001
	(0.019)	(0.003)	(0.006)	(0.006)	(0.011)	(0.004)
β_2	−0.026		−0.023***		−0.024***	
	(0.028)		(0.001)		(0.006)	
Observations	1,776	208	1,776	208	1,776	208
R^2	0.001	0.0002	0.001	0.001	0.002	0.0003
Adjusted R^2	-0.058	-0.156	-0.057	-0.156	-0.056	-0.156
F Statistic	0.740 (df = 2; 1677)	0.036 (df = 1; 179)	1.183 (df = 2; 1677)	0.136 (df = 1; 179)	1.931 (df = 2; 1677)	0.054 (df = 1; 179)

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

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", "MonthYear"), 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", "MonthYear"), 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", "MonthYear"), 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", "MonthYear"), 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: 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.

```
1  temp <- read_sas(sales_allother_zipcode_path)
2  temp$Store_Close_Status <- 0
3  table_C1_0m_raw <-  rbind(temp, read_sas(sales_cc_0mile_path))
4  table_C1_5m_raw <-  rbind(temp, read_sas(sales_cc_5miles_path))
5
6  # Date Transform
7  table_C1_0m_raw$event_date <- as.Date(table_C1_0m_raw$event_date)
8  table_C1_5m_raw$event_date <- as.Date(table_C1_5m_raw$event_date)
9
10 # construct MonthYear - month of year
11 table_C1_0m_raw$MonthYear <- format(table_C1_0m_raw$event_date, "%Y-%m")
12 table_C1_5m_raw$MonthYear <- format(table_C1_5m_raw$event_date, "%Y-%m")
13
14 # Mark CC Closure
15
16 # CCStorePresent
17 # it is the same as Store_Close_Status
18 table_C1_0m_raw$CCStorePresent <- table_C1_0m_raw$Store_Close_Status
19 table_C1_5m_raw$CCStorePresent <- table_C1_5m_raw$Store_Close_Status
20
21 # AfterStoreClosing
22 table_C1_0m_raw$AfterStoreClosing <- ifelse(table_C1_0m_raw$MonthYear < "2008-11", 0, 1)
23 table_C1_5m_raw$AfterStoreClosing <- ifelse(table_C1_5m_raw$MonthYear < "2008-11", 0, 1)
24
25 # BBStorePresent
26 table_C1_0m_raw <- merge(table_C1_0m_raw, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)
27 table_C1_5m_raw <- merge(table_C1_5m_raw, bb_zipcode, by.x ="Zip_Code", by.y = "Zip_Code", all.x = TRUE)
28
29 table_C1_0m_raw$BBStorePresent <- na.fill(table_C1_0m_raw$BB_Store_Status, 0)
30 table_C1_5m_raw$BBStorePresent <- na.fill(table_C1_5m_raw$BB_Store_Status, 0)
31
32 # t test
33 control_before_age    <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 0)&(table_C1_0m_raw$AfterStoreClosing==0),]$hoh_oldest_age
34 control_before_income <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 0)&(table_C1_0m_raw$AfterStoreClosing==0),]$household_income
35 control_before_edu    <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 0)&(table_C1_0m_raw$AfterStoreClosing==0),]$hoh_most_education
36
37 control_after_age     <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 0)&(table_C1_0m_raw$AfterStoreClosing==1),]$hoh_oldest_age
38 control_after_income  <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 0)&(table_C1_0m_raw$AfterStoreClosing==1),]$household_income
39 control_after_edu     <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 0)&(table_C1_0m_raw$AfterStoreClosing==1),]$hoh_most_education
40
41 test.control.age <- t.test(control_before_age, control_after_age)
42 test.control.income <- t.test(control_before_income, control_after_income)
43 test.control.edu <- t.test(control_before_edu, control_after_edu)
44
45 treated_before_age    <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 1)&(table_C1_0m_raw$AfterStoreClosing==0),]$hoh_oldest_age
46 treated_before_income <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 1)&(table_C1_0m_raw$AfterStoreClosing==0),]$household_income
47 treated_before_edu    <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 1)&(table_C1_0m_raw$AfterStoreClosing==0),]$hoh_most_education
```

```
48
49  treated_after_age     <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 1)&(table_C1_0m_raw$AfterStoreClosing==1),]$hoh_oldest_age
50  treated_after_income  <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 1)&(table_C1_0m_raw$AfterStoreClosing==1),]$household_income
51  treated_after_edu     <- table_C1_0m_raw[(table_C1_0m_raw$CCStorePresent == 1)&(table_C1_0m_raw$AfterStoreClosing==1),]$hoh_most_education
52
53  test.treated.age <- t.test(treated_before_age, treated_after_age)
54  test.treated.income <- t.test(treated_before_income, treated_after_income)
55  test.treated.edu <- t.test(treated_before_edu, treated_after_edu)
56
57  # Construct Variables
58  test.control.age.before.mean    <- test.control.age$estimate[["mean of x"]]
59  test.control.income.before.mean <- test.control.income$estimate[["mean of x"]]
60  test.control.edu.before.mean    <- test.control.edu$estimate[["mean of x"]]
61
62  test.control.age.after.mean    <- test.control.age$estimate[["mean of y"]]
63  test.control.income.after.mean <- test.control.income$estimate[["mean of y"]]
64  test.control.edu.after.mean    <- test.control.edu$estimate[["mean of y"]]
65
66  test.control.age.diff    <- test.control.age.after.mean - test.control.age.before.mean
67  test.control.age.diff.se <- test.control.age$p.value
68  test.control.income.diff    <- test.control.income.after.mean - test.control.income.before.mean
69  test.control.income.diff.se <- test.control.income$p.value
70  test.control.edu.diff    <- test.control.edu.after.mean - test.control.edu.before.mean
71  test.control.edu.diff.se <- test.control.edu$p.value
72
73  test.treated.age.before.mean    <- test.treated.age$estimate[["mean of x"]]
74  test.treated.income.before.mean <- test.treated.income$estimate[["mean of x"]]
75  test.treated.edu.before.mean    <- test.treated.edu$estimate[["mean of x"]]
76
77  test.treated.age.after.mean    <- test.treated.age$estimate[["mean of y"]]
78  test.treated.income.after.mean <- test.treated.income$estimate[["mean of y"]]
79  test.treated.edu.after.mean    <- test.treated.edu$estimate[["mean of y"]]
80
81  test.treated.age.diff    <- test.treated.age.after.mean - test.treated.age.before.mean
82  test.treated.age.diff.se <- test.treated.age$p.value
83  test.treated.income.diff    <- test.treated.income.after.mean - test.treated.income.before.mean
84  test.treated.income.diff.se <- test.treated.income$p.value
85  test.treated.edu.diff    <- test.treated.edu.after.mean - test.treated.edu.before.mean
86  test.treated.edu.diff.se <- test.treated.edu$p.value
87
88  # Construct Table
89  tabc1 <- rbind(c("Control", test.control.age.before.mean, test.control.income.before.mean, test.control.edu.before.mean,
90                   test.control.age.after.mean, test.control.income.after.mean, test.control.edu.after.mean,
91                   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),
92        c("Treated", test.treated.age.before.mean, test.treated.income.before.mean, test.treated.edu.before.mean,
93                   test.treated.age.after.mean, test.treated.income.after.mean, test.treated.edu.after.mean,
94                   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

2.16.1   Table D1

Table 16: Search Intensity Effects on Sales for Amazon

	(1)
	Log(Sales + 1)
β_1	2.376***
	(0.0435)
β_2	3.194***
	(0.0675)
β_3	-2.153***
	(0.0744)
Observations	10791

Standard errors in parentheses

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Stata codes for generating Table 16 are listed below.

```
1  eststo: reg LogSales MinsPerPage ExperienceGood tD1INTER if domain_name == "amazon.com", vce(cluster Code_Time) noconstant
2
3  esttab using tableD1.tex, se label replace booktabs title(Search Intensity Effects on Sales for Amazon\label{tab:tabD1})
4
5  eststo clear
```

Code 17: Table D1 Generation

2.16.2   Table D2

Table 17: Product Characteristics Effects on Search Intensity for Amazon

	(1)	(2)
	Log(PagesViewed)	Log(MinsSpent + 1)
β_1	0.00465***	0.00450***
	(0.000843)	(0.000785)
β_2	3.156***	2.923***
	(0.0581)	(0.0558)
Observations	10791	10791

Standard errors in parentheses

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Stata codes for generating Table 17 are listed below.

```
1  eststo: reg LogPagesViewed prod_totprice ExperienceGood if domain_name == "amazon.com", vce(cluster Code_Time) noconstant
2
3  eststo: reg LogMinsSpent prod_totprice ExperienceGood if domain_name == "amazon.com", vce(cluster Code_Time) noconstant
4
5  esttab using tableD2.tex, se label replace booktabs title(Product Characteristics Effects on Search Intensity for Amazon\label{tab:tabD2})
6
7  eststo clear
```

Code 18: Table D2 Generation

2.16.3   Table D3

Table 18: Product Characteristics Effects on Search Intensity for Amazon

	(1)	(2)
	RefDomainIsAmazon	ReferringDomainIsSearchEngine
ExperienceGood	-4.274***	-0.828***
	(0.310)	(0.0672)
Observations	10791	10791

Standard errors in parentheses

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Stata codes for generating Table 18 are listed below.

```
1  eststo: logit RefDomainIsAmazon ExperienceGood if domain_name == "amazon.com", vce(cluster MonthYear) noconstant
2
3  eststo: logit ReferringDomainIsSearchEngine ExperienceGood if domain_name == "amazon.com", vce(cluster MonthYear) noconstant
4
5  esttab using tableD3.tex, se label replace booktabs title(Product Characteristics Effects on Search Intensity for Amazon\label{tab:tabD3})
6
7  eststo clear
```

Code 19: Table D3 Generation

2.17 Table E1-E3

2.17.1 Table E1

Table 19: Results of the Sales Effect (All Product Categories)

| | log(TotalMonthlySales + 1) | | | |
| | staples.com-0 Mile | walmart.com-0 Mile | dell.com-0 Mile | circuitcity.com-0 Mile |
	(1)	(2)	(3)	(4)
β_1	−0.027	0.026	−0.006	0.003
	(0.064)	(0.018)	(0.018)	(0.036)
β_2	0.082	−0.034	−0.018	0.013
	(0.075)	(0.022)	(0.026)	(0.051)
Observations	8,352	24,912	19,440	3,940
R^2	0.0003	0.0001	0.0001	0.0001
Adjusted R^2	-0.046	-0.044	-0.045	-0.058
F Statistic	1.004 (df = 2; 7979)	1.332 (df = 2; 23849)	0.834 (df = 2; 18605)	0.094 (df = 2; 3722)

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

Codes for generating Table 19 are listed below.

```
# Table E1
staple.tE1.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tE_balanced[data_0m_tE_balanced$domain_name == "staples.com"
    ,], index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
walmart.tE1.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tE_balanced[data_0m_tE_balanced$domain_name == "walmart.com"
    ,], index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
dell.tE1.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tE_balanced[data_0m_tE_balanced$domain_name == "dell.com",],
    index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
cc.tE1.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tE_balanced[data_0m_tE_balanced$domain_name == "circuitcity.com"
    ,], index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
```

Code 20: Table E1 Generation

2.17.2 Table E2

Table 20: Results of the Online Search Effect (All Product Categories)

| | log(PagesPerDollar + 1) | | | |
| | staples.com-0 Mile | walmart.com-0 Mile | dell.com-0 Mile | circuitcity.com-0 Mile |
	(1)	(2)	(3)	(4)
β_1	0.010	0.004	0.001	-0.002
	(0.027)	(0.009)	(0.004)	(0.012)
β_2	-0.017	-0.002	-0.002	0.0002
	(0.031)	(0.011)	(0.005)	(0.016)
Observations	8,352	24,912	19,440	3,940
R^2	0.00004	0.00001	0.00001	0.00002
Adjusted R^2	-0.047	-0.045	-0.045	-0.058
F Statistic	0.171 (df = 2; 7979)	0.123 (df = 2; 23849)	0.083 (df = 2; 18605)	0.030 (df = 2; 3722)

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

Codes for generating Table 20 are listed below.

```
1  # Table E2
2  staple.tE2.0mile <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_tE_balanced[data_0m_tE_balanced$domain_name == "staples.com",],
       index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
3  walmart.tE2.0mile <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_tE_balanced[data_0m_tE_balanced$domain_name == "walmart.com",],
       index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
4  dell.tE2.0mile <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_tE_balanced[data_0m_tE_balanced$domain_name == "dell.com",], index
       = c("Zip_Code", "Time"), model = "within", effect = "twoways")
5  cc.tE2.0mile <- plm(log(PagesPerDollar + 1) ~ DID + THREEINTER, data = data_0m_tE_balanced[data_0m_tE_balanced$domain_name == "circuitcity.com",],
       index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
```

Code 21: Table E2 Generation

2.17.3 Table E3

Table 21: Results of the Online Search Effect (All Product Categories)

	log(MinsPerDollar + 1)			
	staples.com-0 Mile	walmart.com-0 Mile	dell.com-0 Mile	circuitcity.com-0 Mile
	(1)	(2)	(3)	(4)
β_1	−0.011	0.002	−0.001	−0.002
	(0.022)	(0.008)	(0.003)	(0.010)
β_2	0.008	−0.001	−0.001	0.00003
	(0.027)	(0.010)	(0.005)	(0.014)
Observations	8,352	24,912	19,440	3,940
R^2	0.00003	0.00000	0.00001	0.00002
Adjusted R^2	-0.047	-0.045	-0.045	-0.058
F Statistic	0.137 (df = 2; 7979)	0.056 (df = 2; 23849)	0.124 (df = 2; 18605)	0.037 (df = 2; 3722)

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

Codes for generating Table 21 are listed below.

```
# Table E3
staple.tE3.0mile <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_tE_balanced[data_0m_tE_balanced$domain_name == "staples.com",],
    index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
walmart.tE3.0mile <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_tE_balanced[data_0m_tE_balanced$domain_name == "walmart.com",],
    index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
dell.tE3.0mile <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_tE_balanced[data_0m_tE_balanced$domain_name == "dell.com",], index
    = c("Zip_Code", "Time"), model = "within", effect = "twoways")
cc.tE3.0mile <- plm(log(MinsPerDollar + 1) ~ DID + THREEINTER, data = data_0m_tE_balanced[data_0m_tE_balanced$domain_name == "circuitcity.com",],
    index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
```

Code 22: Table E3 Generation

2.18   Table G1-G3

2.18.1   Table G1

Table 22: Results of the Sales Effect (Music, Movies and Videos, Console Video Games)

	log(TotalMonthlySales + 1)		
	amazon.com-0 Mile	bestbuy.com-0 Mile	circuitcity.com-0 Mile
	(1)	(2)	(3)
β_1	0.005	-0.001	-0.002
	(0.013)	(0.024)	(0.043)
β_2	0.008	0.000	
	(0.019)	(0.028)	
Observations	52,416	5,808	810
R^2	0.00002	0.00000	0.00000
Adjusted R^2	-0.044	-0.048	-0.092
F Statistic	0.535 (df = 2; 50207)	0.004 (df = 2; 5541)	0.001 (df = 1; 741)

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

Codes for generating Table 22 are listed below.

```
1  # Table G1
2  ama.tG.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tG_balanced[data_0m_tG_balanced$domain_name == "amazon.com",],
          index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
3  bb.tG.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tG_balanced[data_0m_tG_balanced$domain_name == "bestbuy.com",],
          index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
4  stp.tG.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tG_balanced[data_0m_tG_balanced$domain_name == "staples.com",],
          index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
5  walmart.tG.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tG_balanced[data_0m_tG_balanced$domain_name == "dell.com",],
          index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
6  cc.tG.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tG_balanced[data_0m_tG_balanced$domain_name == "circuitcity.com"
          ,], index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
```

Code 23: Table G1 Generation

2.18.2 Table G2

Table 23: Results of the Sales Effect (All Products; All Online Sellers in the Control Group)

	log(TotalMonthlySales + 1)				
	amazon.com-0 Mile	bestbuy.com-0 Mile	staples.com-0 Mile	walmart.com-0 Mile	circuitcity.com-0 Mile
	(1)	(2)	(3)	(4)	(5)
β_1	0.014	−0.002	−0.027	−0.006	0.003
	(0.015)	(0.033)	(0.064)	(0.018)	(0.036)
β_2	−0.033	0.009	0.082	−0.018	0.013
	(0.022)	(0.036)	(0.075)	(0.026)	(0.051)
Observations	68,472	14,664	8,352	19,440	3,940
R^2	0.00003	0.00002	0.0003	0.0001	0.0001
Adjusted R^2	-0.044	-0.045	-0.046	-0.045	-0.058
F Statistic	1.091 (df = 2; 65594)	0.154 (df = 2; 14028)	1.004 (df = 2; 7979)	0.834 (df = 2; 18605)	0.094 (df = 2; 3722)

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

Codes for generating Table 23 are listed below.

```
1  # Table G2
2  # Do not filter control group on target_domain_to_consider and no category filtering
3  ama.tG.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tG_balanced[data_0m_tG_balanced$domain_name == "amazon.com",],
       index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
4  bb.tG.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tG_balanced[data_0m_tG_balanced$domain_name == "bestbuy.com",],
       index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
5  stp.tG.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tG_balanced[data_0m_tG_balanced$domain_name == "staples.com",],
       index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
6  walmart.tG.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tG_balanced[data_0m_tG_balanced$domain_name == "dell.com",],
       index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
7  cc.tG.0mile <- plm(log(TotalMonthlySales + 1) ~ DID + THREEINTER, data = data_0m_tG_balanced[data_0m_tG_balanced$domain_name == "circuitcity.com"
       ,], index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
```

Code 24: Table G2 Generation

2.18.3   Table G3

Table 24: Effect Referring Domain on Amazon Sales

	RefferingDomainIsSearchEngineRatio Amazon	NoReferringDomainRatio Amazon
	(1)	(2)
β_1	-0.010**	0.009*
	(0.005)	(0.005)
β_2	0.011	-0.008
	(0.007)	(0.007)
Observations	73,416	73,416
R^2	0.0001	0.00004
Adjusted R^2	-0.044	-0.044
F Statistic (df = 2; 70332)	1.961	1.422

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

Codes for generating Table **??** are listed below.

```
# Table G3
ama.tG3.0mile.r1 <- plm(ReferringDomainIsSearchEngineRatio ~ DID + THREEINTER, data = data_0m_tG3_balanced[data_0m_tG3_balanced$domain_name == "
    amazon.com",], index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
ama.tG3.0mile.r2 <- plm(NoReferringDomainRatio ~ DID + THREEINTER, data = data_0m_tG3_balanced[data_0m_tG3_balanced$domain_name == "amazon.com",],
    index = c("Zip_Code", "Time"), model = "within", effect = "twoways")
```

Code 25: Table G3 Generation

3 Advanced Method

In this section, we present some advanced methods on the provided data.

3.1 Generalized Synthetic Control Method

3.2 PSM and LA-PSM

3.3 Causal Forest

4   References

[1] Badi H Baltagi and Seuck Heun Song. Unbalanced panel data: A survey. *Statistical Papers*, 47(4): 493–523, 2006.

[2] Yves Croissant, Giovanni Millo, et al. Panel data econometrics in r: The plm package. *Journal of statistical software*, 27(2):1–43, 2008.

[3] G Grothendieck. sqldf: manipulate r data frames using sql. *R package version 0.4-11*, 2017.

[4] Marek Hlavac. Stargazer: Well-formatted regression and summary statistics tables. *R package version*, 5(1), 2015.

[5] Ben Jann. Estout: Stata module to export estimation results from estimates table. 2004.

[6] Allen McDowell. From the help desk: Seemingly unrelated regression with unbalanced equations. *The STATA journal*, 4(4):442–448, 2004.

[7] Hadley Wickham and Evan Miller. haven: Import and export "spss","stata" and "sas" files. *R package version*, 1(0), 2018.

[8] Hadley Wickham, Romain Francois, Lionel Henry, Kirill Müller, et al. dplyr: A grammar of data manipulation. *R package version 0.4*, 3, 2015.

[9] Achim Zeileis and Gabor Grothendieck. zoo: S3 infrastructure for regular and irregular time series. *arXiv preprint math/0505527*, 2005.