Grocery Sales Forecast

Christine Vu[1], Dave Friesen[2]

12/12/2022

Define Goal

Get Data

```
# Load dataset(s); assumes folder structure with data parallel to src
sales_df <- read.csv("../data/train.csv", header = TRUE)
sales_test_df <- read.csv("../data/test.csv", header = TRUE)
stores_df <- read.csv("../data/stores.csv", header = TRUE)
oil_df <- read.csv("../data/oil.csv", header = TRUE)
events_df <- read.csv("../data/holidays_events.csv", header = TRUE)

# Data validation and understanding, including structure, content, and statistical characteristics covered below
```

Explore & Visualize Series

```
# e.g., statistical characteristics (including distribution, skewness, outliers)
#   +[optionally] review sample observations
univariate(sales_df); head(sales_df, 3); #str(sales_df)
```

```
Summary Univariate Analysis for (sales_df) (3,000,888 observations)
            Type         NA% Blank%    Unique       Min       Max      Mean    Median Outlier< >Outlier Skewness nZV ACF1
id          integer                   3000888             3000887             1500444    No       No              N   1.0
date        character                    1684                                                                     N   1.0
store_nbr   integer                        54         1        54                  28    No      Yes              N   1.0
family      character                      33                                                                     N   0.8
sales       numeric             31%    379610            124717.0     357.8      11.0    No      Yes         7.4  N
onpromotion integer             79%       362                 741                        No       No        11.2  N
```

```
  id       date store_nbr     family sales onpromotion
1  0 2013-01-01         1 AUTOMOTIVE     0           0
2  1 2013-01-01         1  BABY CARE     0           0
3  2 2013-01-01         1     BEAUTY     0           0
```

```
univariate(sales_test_df); head(sales_test_df, 3); #str(sales_test_df)
```

```
Summary Univariate Analysis for (sales_test_df) (28,512 observations)
            Type         NA% Blank%    Unique       Min       Max      Mean    Median Outlier< >Outlier Skewness nZV ACF1
id          integer                     28512   3000888   3029399             3015144    No       No              N   1.0
date        character                      16                                                                     N   1.0
store_nbr   integer                        54         1        54                  28    No      Yes              N   1.0
family      character                      33                                                                     N   0.8
onpromotion integer             55%       212                 646                        No      Yes         8.5  N
```

```
       id       date store_nbr     family onpromotion
1 3000888 2017-08-16         1 AUTOMOTIVE           0
2 3000889 2017-08-16         1  BABY CARE           0
3 3000890 2017-08-16         1     BEAUTY           2
```

```
univariate(stores_df); head(stores_df, 3); #str(stores_df)
```

```
Summary Univariate Analysis for (stores_df) (54 observations)
          Type         NA% Blank%    Unique       Min       Max      Mean    Median Outlier< >Outlier Skewness nZV ACF1
store_nbr integer                        54         1        54                  28    No       No              N   0.9
city      character                      22                                                                     N   0.3
state     character                      16                                                                     N   0.4
type      character                       5                                                                     N   0.6
cluster   integer                        17         1        17                   8    No      Yes              N   0.2
```

```
  store_nbr  city     state type cluster
1         1 Quito Pichincha    D      13
2         2 Quito Pichincha    D      13
3         3 Quito Pichincha    D       8
```

```
univariate(oil_df); head(oil_df, 3); #str(oil_df)
```

```
Summary Univariate Analysis for (oil_df) (1,218 observations)
           Type         NA% Blank%    Unique       Min       Max      Mean    Median Outlier< >Outlier Skewness nZV ACF1
date       character                    1218                                                                     N   1.0
dcoilwtico numeric       3               998      26.2     110.6      67.7      53.2    No      Yes         0.3  N   0.9
```

```
        date dcoilwtico
1 2013-01-01         NA
2 2013-01-02         93
3 2013-01-03         93
```

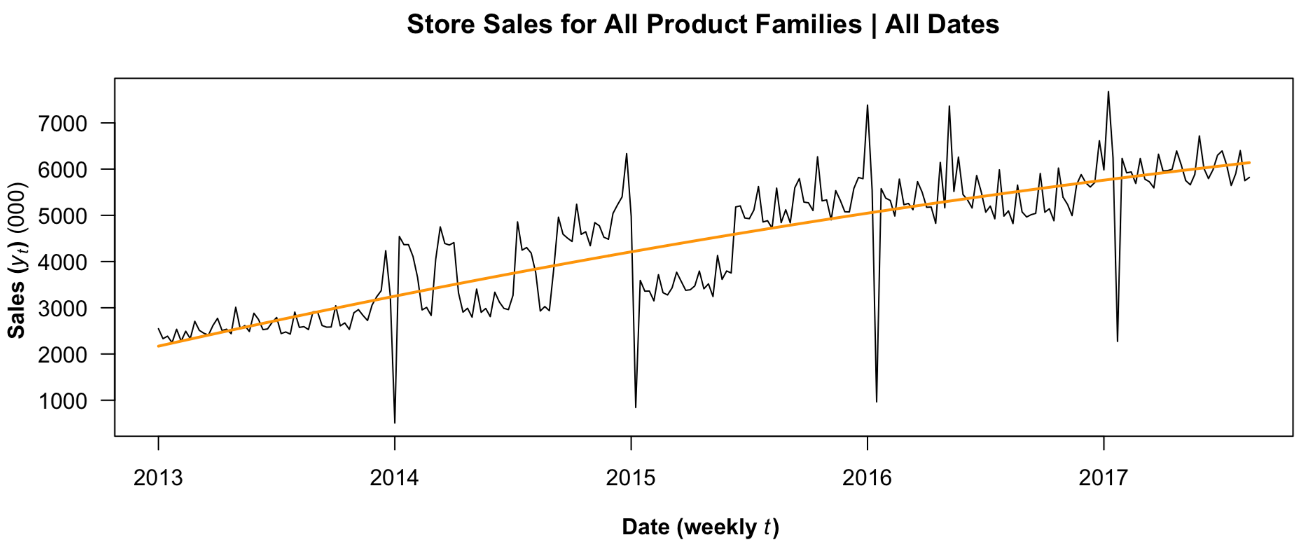
```
univariate(events_df); head(events_df, 3); #str(events_df)
```

```
Summary Univariate Analysis for (events_df) (350 observations)
            Type         NA% Blank%    Unique       Min       Max      Mean    Median Outlier< >Outlier Skewness nZV ACF1
date        character                     312                                                                     N   1.0
type        character                       6                                                                     N   0.3
locale      character                       3                                                                     N   0.3
locale_name character                      24                                                                     N
description character                     103                                                                     N   0.2
transferred character                       2                                                                     Y
```

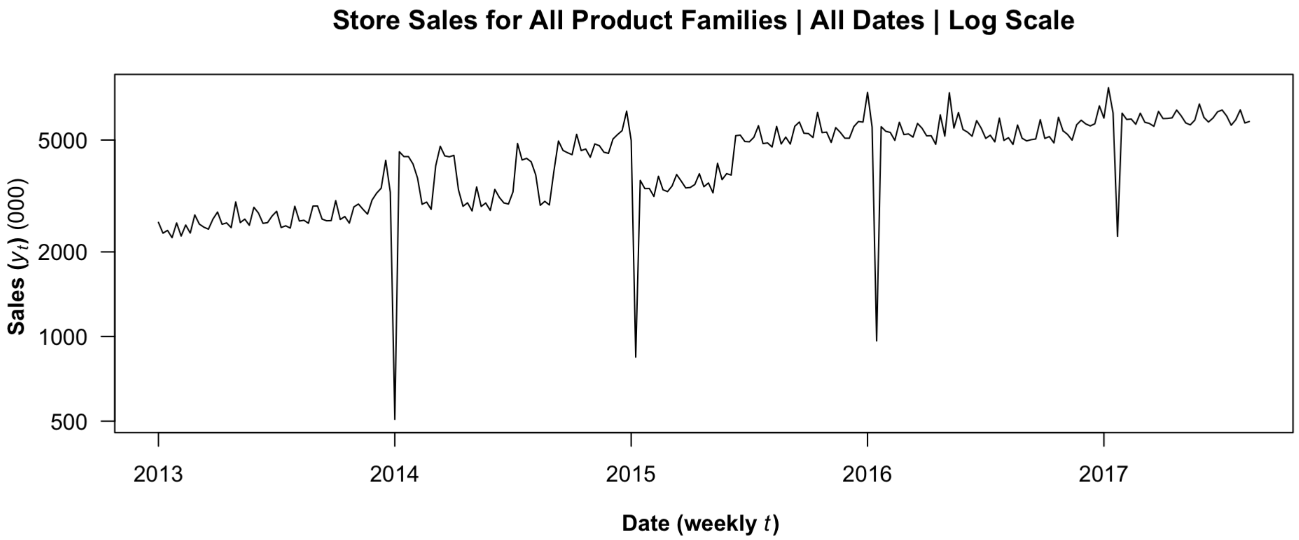
```
        date    type   locale locale_name                    description transferred
1 2012-03-02 Holiday    Local       Manta             Fundacion de Manta       False
2 2012-04-01 Holiday Regional    Cotopaxi Provincializacion de Cotopaxi       False
3 2012-04-12 Holiday    Local      Cuenca            Fundacion de Cuenca       False
```

```
# Convert string mm/dd/yyyy to Date values and confirm sort
sales_df <- (sales_df %>%
                mutate(date = as.Date(date, format = "%Y-%m-%d"),) %>%
                arrange(date))
```

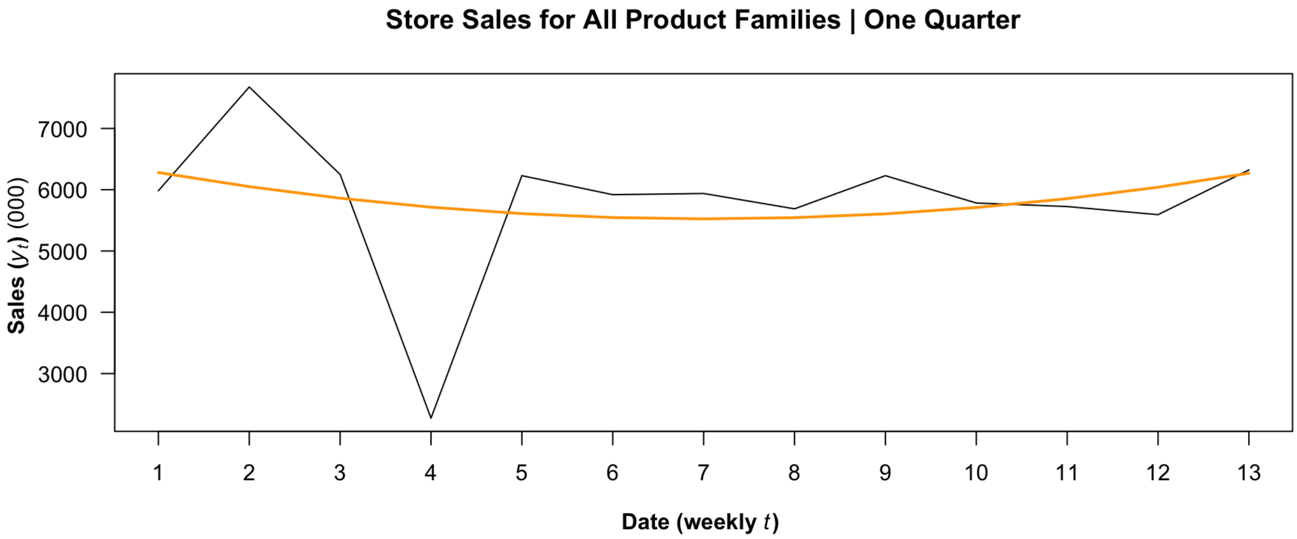
```
# Aggregate base dataframe from daily to weekly for all product families
sales_wk_df <- as.data.frame(sales_df %>%
                                mutate(year = year(date), week = week(date)) %>%
                                group_by(year, week) %>%
                                summarize(sales = sum(sales / 1000.0)))

# Create overall time series
sales_begin_year <- head(sales_wk_df$year, 1)
sales_begin_week <- head(sales_wk_df$week, 1)
sales_end_year <- tail(sales_wk_df$year, 1)
sales_end_week <- tail(sales_wk_df$week, 1)
sales_ts <- ts(sales_wk_df$sales,
               start = c(sales_begin_year, sales_begin_week),
               end = c(sales_end_year, sales_end_week), freq = 52)

# Plot overall time series with trend line
plot(sales_ts, type = "l",
     main = "Store Sales for All Product Families | All Dates",
     xlab = TeX(r"(\textbf{Date (weekly \textit{$t$})} )"), ylab = TeX(r"(\textbf{Sales (\textit{$y_t$})} (000) )"),
     las = 1, cex.axis = 0.7)

sales_lm <- tslm(sales_ts ~ trend + I(trend^2))
lines(sales_lm$fitted, lwd = 2, lty = 1, col = "orange")
```



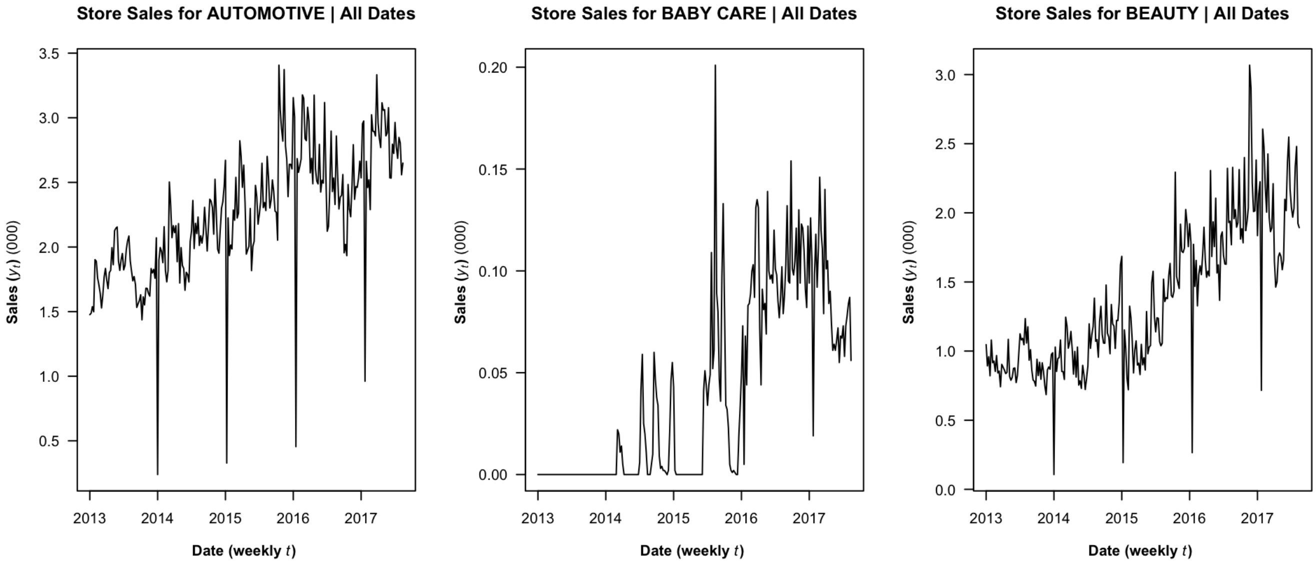
```
# Plot overall time series w/log scale
plot(sales_ts, type = "l",
     main = "Store Sales for All Product Families | All Dates | Log Scale",
     xlab = TeX(r"(\textbf{Date (weekly \textit{$t$})} )"), ylab = TeX(r"(\textbf{Sales (\textit{$y_t$})} (000) )"),
     las = 1, cex.axis = 0.7,
     log = "y")
```

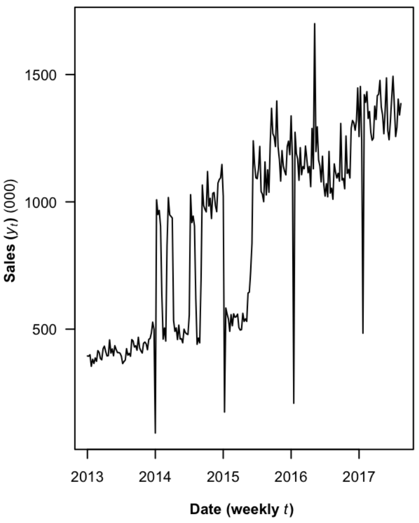


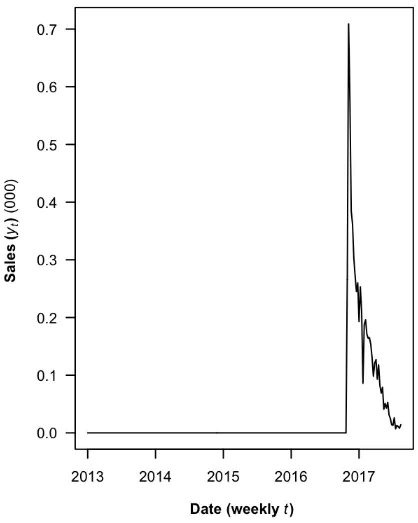
```
# Plot zoomed time series with trend line
sales_zoom_ts <- window(sales_ts, start = c(sales_end_year, 1), end = c(sales_end_year, 13))
plot(sales_zoom_ts, type = "l",
     main = "Store Sales for All Product Families | One Quarter",
     xlab = TeX(r"(\textbf{Date (weekly \textit{$t$})} )"), ylab = TeX(r"(\textbf{Sales (\textit{$y_t$})} (000) )"),
     xaxt = "n",
     las = 1, cex = 0.7)

sales_zoom_lm <- tslm(sales_zoom_ts ~ trend + I(trend^2))
lines(sales_zoom_lm$fitted, lwd = 2, lty = 1, col = "orange")

axis(1, at = as.numeric(time(sales_zoom_ts)), labels = seq(sales_zoom_ts))
```

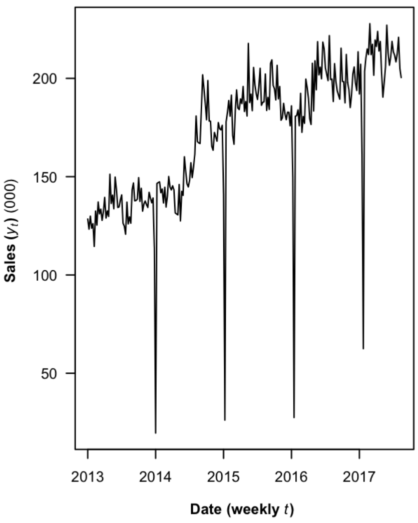


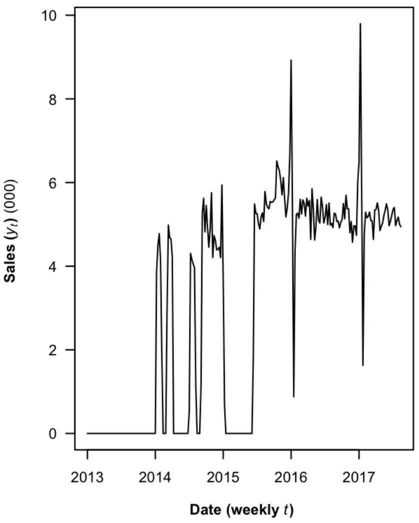
```
# Aggregate base dataframe from daily to weekly by product family
sales_wk_df <- as.data.frame(sales_df %>%
                               mutate(year = year(date), week = week(date)) %>%
                               group_by(family, year, week) %>%
                               summarize(sales = sum(sales / 1000.0)))

opar = par()
par(mfrow = c(1, 3))

for (f in unique(sales_wk_df$family)) {
  # Subset data by product family and create time series
  df <- filter(sales_wk_df, family == f)
  df_ts <- ts(df$sales,
              start = c(sales_begin_year, sales_begin_week),
              end = c(sales_end_year, sales_end_week), freq = 52)

  # Plot time series
  plot(df_ts, type = "l",
       main = paste("Store Sales for ", f, " | All Dates", sep = ""),
       xlab = TeX(r"(\textbf{Date (weekly \textit{$t$})} )"), ylab = TeX(r"(\textbf{Sales (\textit{$y_t$})} (000) )"),
       las = 1, cex.axis = 0.7)
}
```

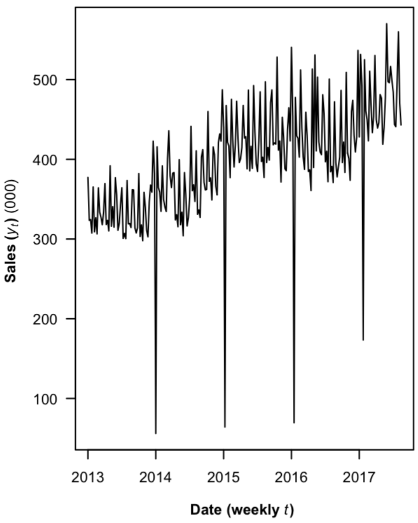


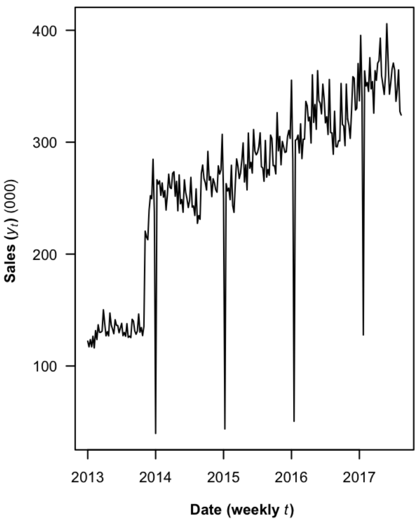

Store Sales for BEVERAGES | All Dates

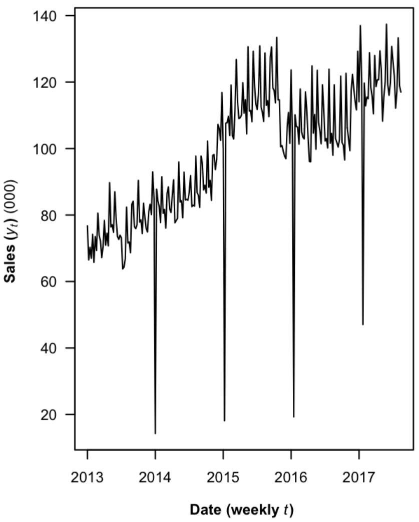

Store Sales for BOOKS | All Dates

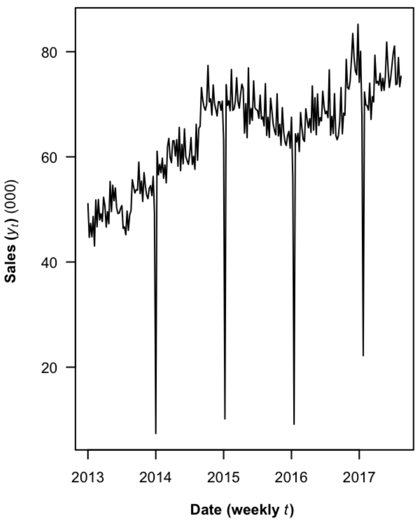

Store Sales for BREAD/BAKERY | All Dates

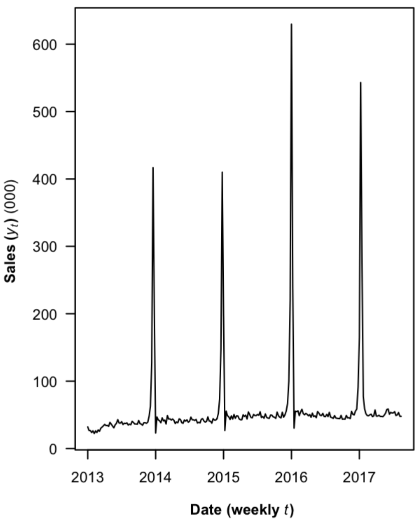

Store Sales for CELEBRATION | All Dates


Store Sales for CLEANING | All Dates

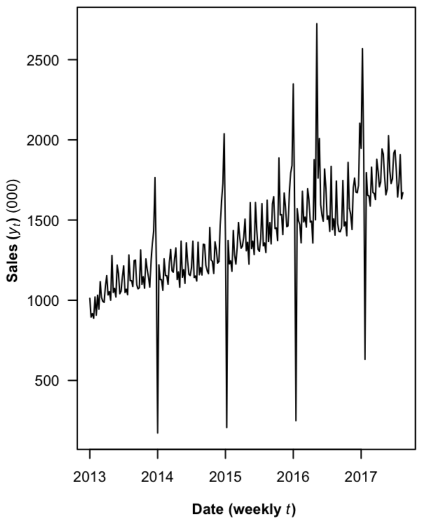

Store Sales for DAIRY | All Dates


Store Sales for DELI | All Dates

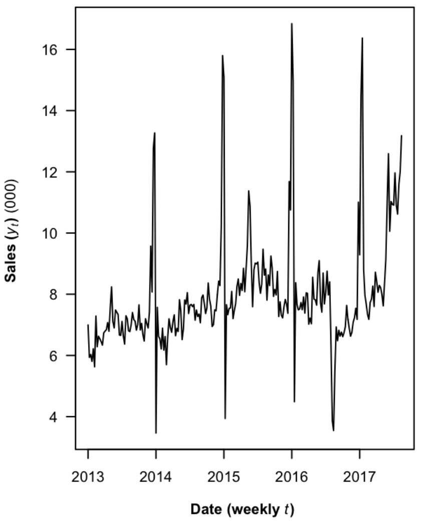

Store Sales for EGGS | All Dates


Store Sales for FROZEN FOODS | All Dates

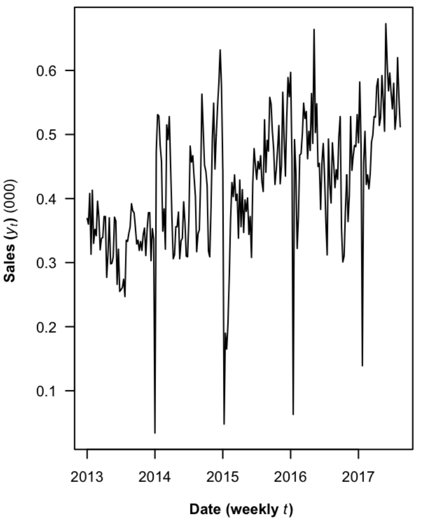
Store Sales for GROCERY I | All Dates



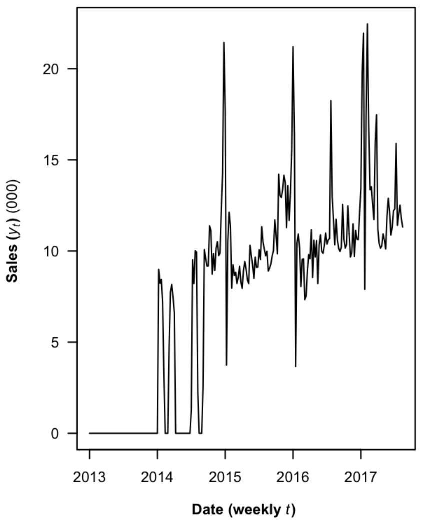
Store Sales for GROCERY II | All Dates



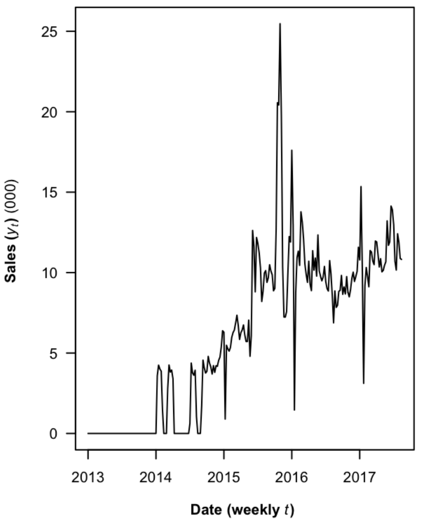
Store Sales for HARDWARE | All Dates



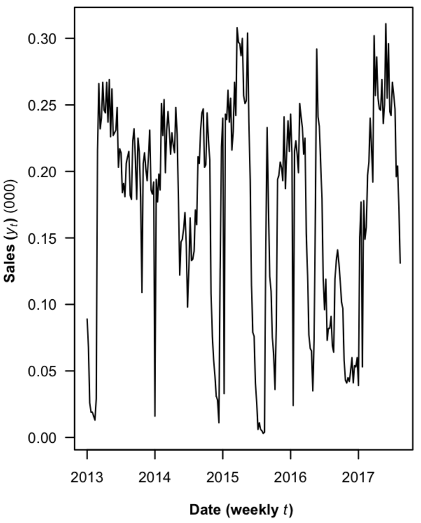
Store Sales for HOME AND KITCHEN I | All Dates



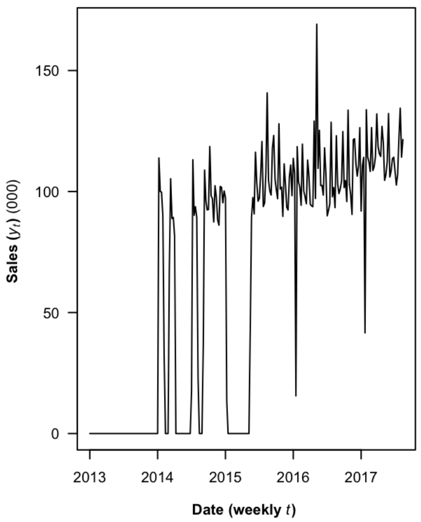
Store Sales for HOME AND KITCHEN II | All Dates



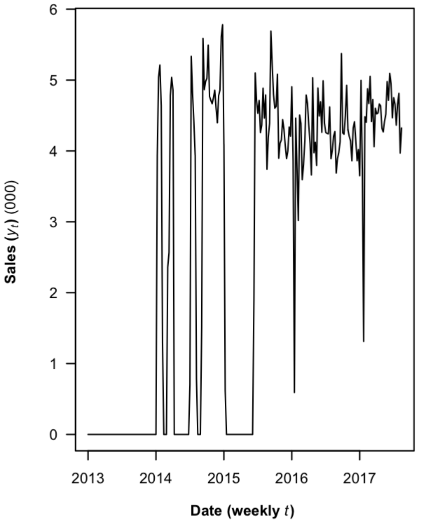
Store Sales for HOME APPLIANCES | All Dates



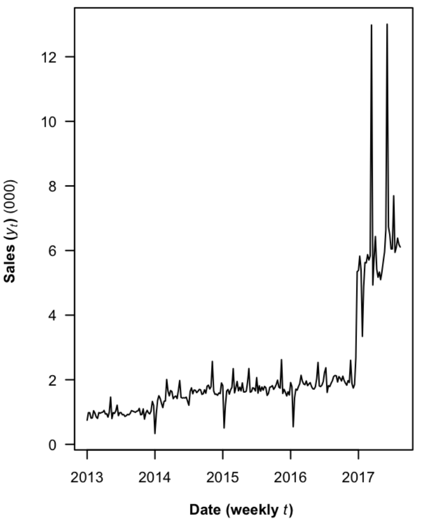
Store Sales for HOME CARE | All Dates



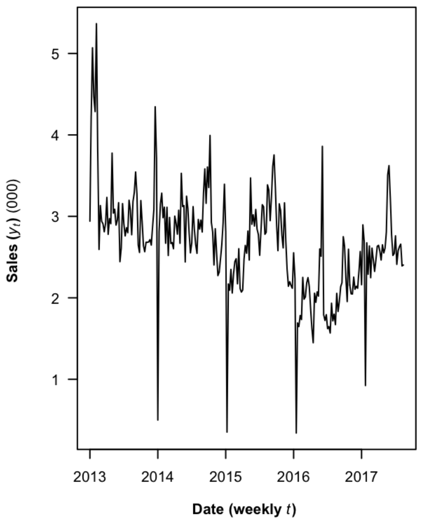
Store Sales for LADIESWEAR | All Dates



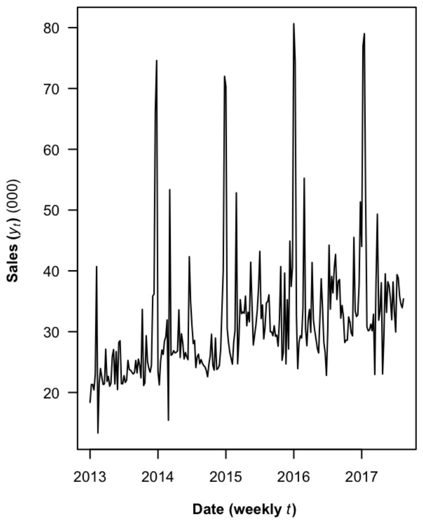
Store Sales for LAWN AND GARDEN | All Dates



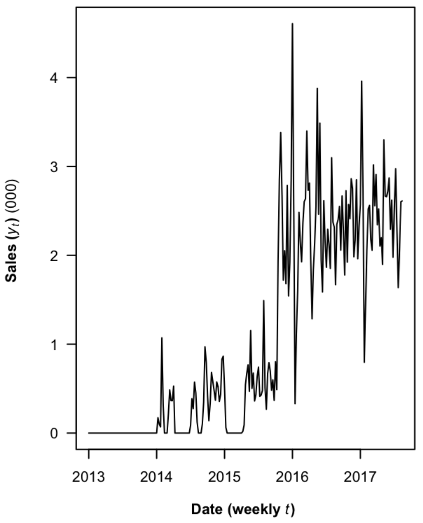
Store Sales for LINGERIE | All Dates



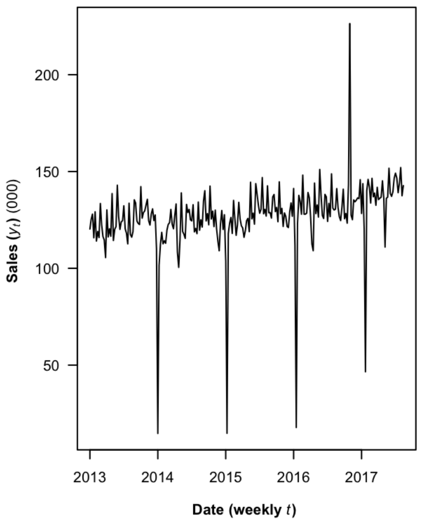
Store Sales for LIQUOR,WINE,BEER | All Dates



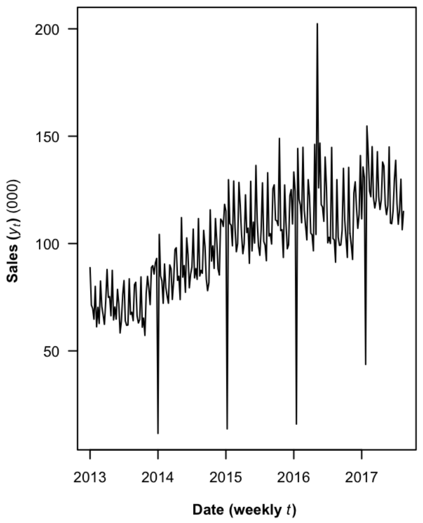
Store Sales for MAGAZINES | All Dates



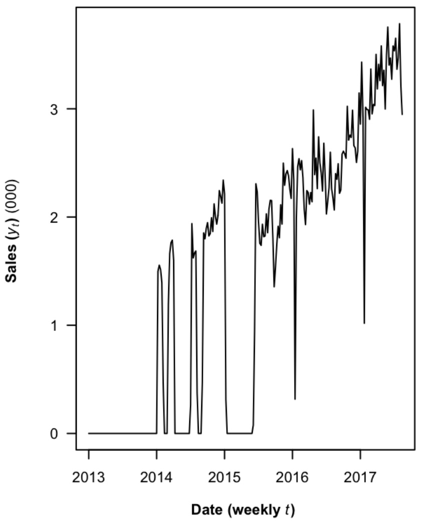
Store Sales for MEATS | All Dates



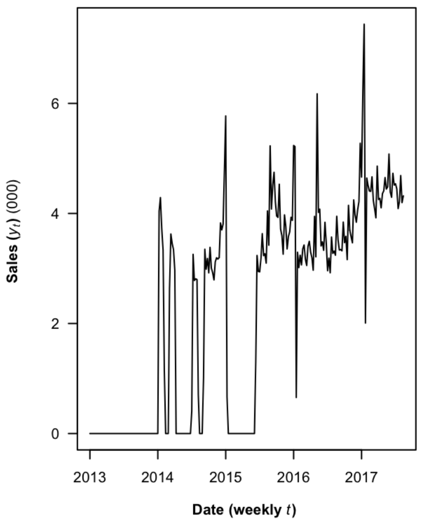
Store Sales for PERSONAL CARE | All Dates



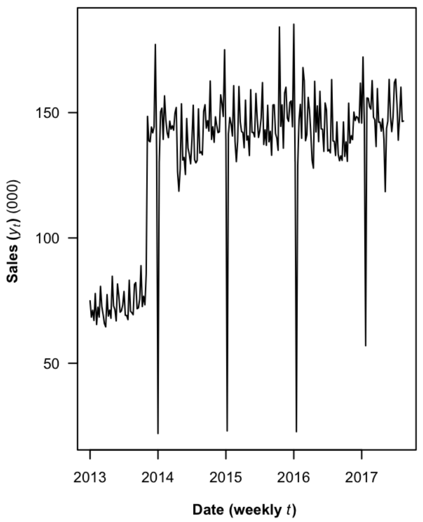
Store Sales for PET SUPPLIES | All Dates



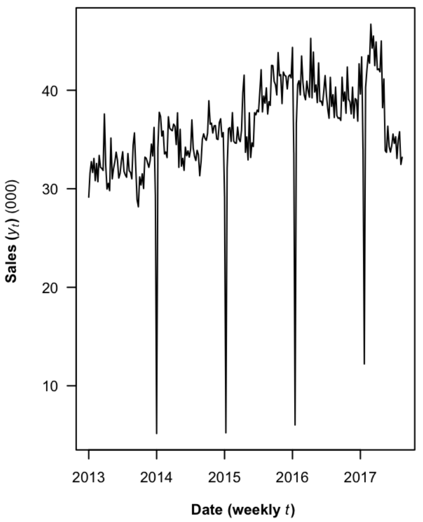
Store Sales for PLAYERS AND ELECTRONICS | All Da

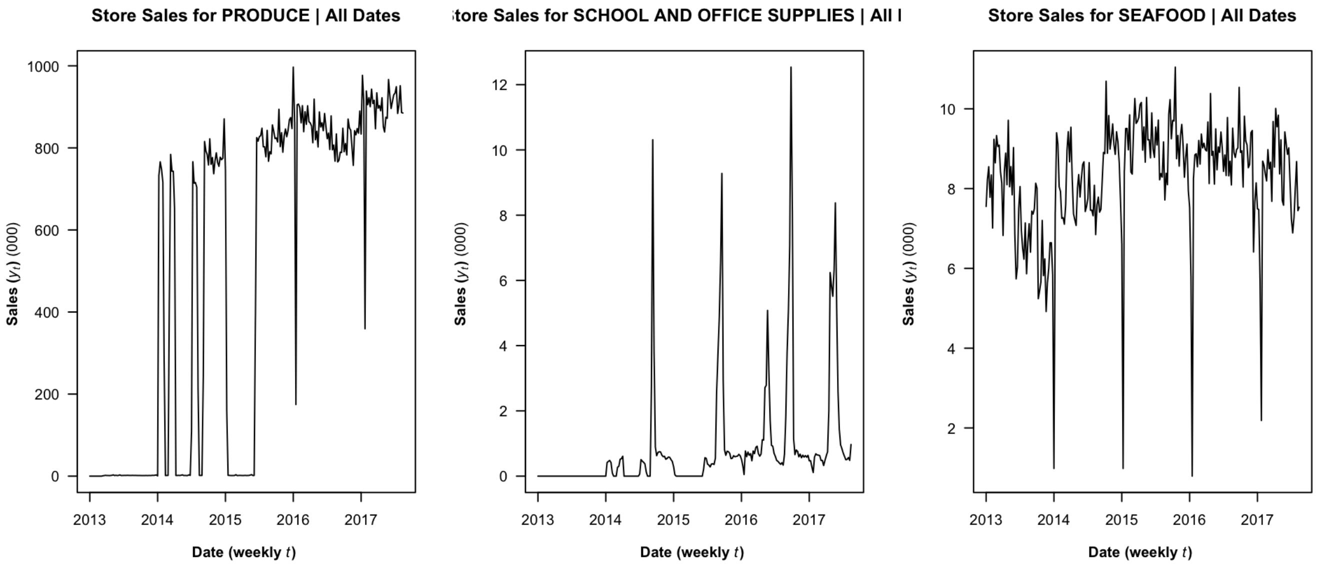


Store Sales for POULTRY | All Dates



Store Sales for PREPARED FOODS | All Dates





```
par(opar)
```

Pre-Process Data

Partition Series

```
# Use one year (52 weeks) as validation period (representative set of quarters, seasons)
sales_n_valid <- 52
sales_n_train <- length(sales_ts) - sales_n_valid

# Split data into training and validation periods
sales_train_ts <- window(sales_ts, start = c(sales_begin_year, 1), end = c(sales_begin_year, sales_n_train))
sales_valid_ts <- window(sales_ts, start = c(sales_begin_year, sales_n_train + 1), end = c(sales_begin_year, sales_n_train +
sales_n_valid))
```

Apply Forecasting Method(s)

Evaluate & Compare Performance

Implement Forecasts/System

1. University of San Diego, cvu@sandiego.edu (mailto:cvu@sandiego.edu)↩
2. University of San Diego, dfriesen@sandiego.edu (mailto:dfriesen@sandiego.edu)↩