Data Science Problem

Siddhartha Jetti

August 18, 2017

Initialize and pre-process

Load all the required libraries to analyse the provided sales data.

```
library(dplyr)   # functions to process and manipulate data
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2) # functions to visualize and plot data
```

Read input files.

```
# List all the input files with names starting 'Sales_week_starting' and ending with '.csv'
setwd("./data")
sales_files <- list.files(pattern="^sales_week_starting_[[:print:]]*.csv$")

# Use lapply to read files sequentially and bind them into a data frame
sales_input <- sales_files %>%
               lapply(read.csv,stringsAsFactors = F) %>%
               bind_rows()
```

Process Input data. Create new fields to hold time, week and part of day when each of the sales happened.

```
sales_df <- sales_input %>%
  mutate(sale_date = as.Date(substr(sale_time,1,10),format="%Y-%m-%d"),
         time = format(strptime(sale_time,"%Y-%m-%d %H:%M:%S"), "%H:%M:%S"),
         week = strftime(as.POSIXlt(sale_time),format="%Y-%U"),
         dayparts = ifelse(as.POSIXct(time,format = '%T') >= as.POSIXct("00:00:00",format =
'%T') & as.POSIXct(time,format = '%T') < as.POSIXct("06:00:00",format = '%T'),"night",
                       ifelse(as.POSIXct(time,format = '%T') >= as.POSIXct("06:00:00",format = '%
T') & as.POSIXct(time,format = '%T') < as.POSIXct("12:00:00",format = '%T'),"morning",
                         ifelse(as.POSIXct(time,format = '%T') >= as.POSIXct("12:00:00",format =
'%T') & as.POSIXct(time,format = '%T') < as.POSIXct("18:00:00",format = '%T'),"afternoon","eveni
ng"))))
```

Check the processed data if everything looks fine.

```
names(sales_df)
```

```
## [1] "sale_time"        "purchaser_gender" "sale_date"
## [4] "time"             "week"             "dayparts"
```

```
head(sales_df,10)
```

```
##              sale_time purchaser_gender  sale_date     time    week
## 1  2012-10-01 01:42:22           female 2012-10-01 01:42:22 2012-40
## 2  2012-10-01 02:24:53           female 2012-10-01 02:24:53 2012-40
## 3  2012-10-01 02:25:40           female 2012-10-01 02:25:40 2012-40
## 4  2012-10-01 02:30:42           female 2012-10-01 02:30:42 2012-40
## 5  2012-10-01 02:51:32             male 2012-10-01 02:51:32 2012-40
## 6  2012-10-01 03:03:00           female 2012-10-01 03:03:00 2012-40
## 7  2012-10-01 03:09:10           female 2012-10-01 03:09:10 2012-40
## 8  2012-10-01 03:09:40             male 2012-10-01 03:09:40 2012-40
## 9  2012-10-01 03:16:08           female 2012-10-01 03:16:08 2012-40
## 10 2012-10-01 03:43:50           female 2012-10-01 03:43:50 2012-40
##    dayparts
## 1     night
## 2     night
## 3     night
## 4     night
## 5     night
## 6     night
## 7     night
## 8     night
## 9     night
## 10    night
```

Problem could arise in extracting the ‘week’ of sale if subsequent year doesn’t start on Monday. Check and resolve issues if any.

```
# filter out records from last week of 2012 and first week of 2013 to inspect.
check_year_start <-sales_df %>%
  filter(week %in% c("2012-53","2013-00"))

# two way frequency table
table(check_year_start$week,check_year_start$sale_date)
```

```
##
##           2012-12-30 2012-12-31 2013-01-01 2013-01-02 2013-01-03
##   2012-53        549        538          0          0          0
##   2013-00          0          0        458        532        510
##
##           2013-01-04 2013-01-05
##   2012-53          0          0
##   2013-00        536        497
```

The week starting 30th December 2012 is split into two and needs to be fixed.

```
sales_df <- sales_df %>%
  mutate(week= ifelse(week=="2013-00","2012-53",week))
```

Assumptions

Since the value ($) of each sale is not provided, all the sales recorded are assumed to be equal in value and importance.

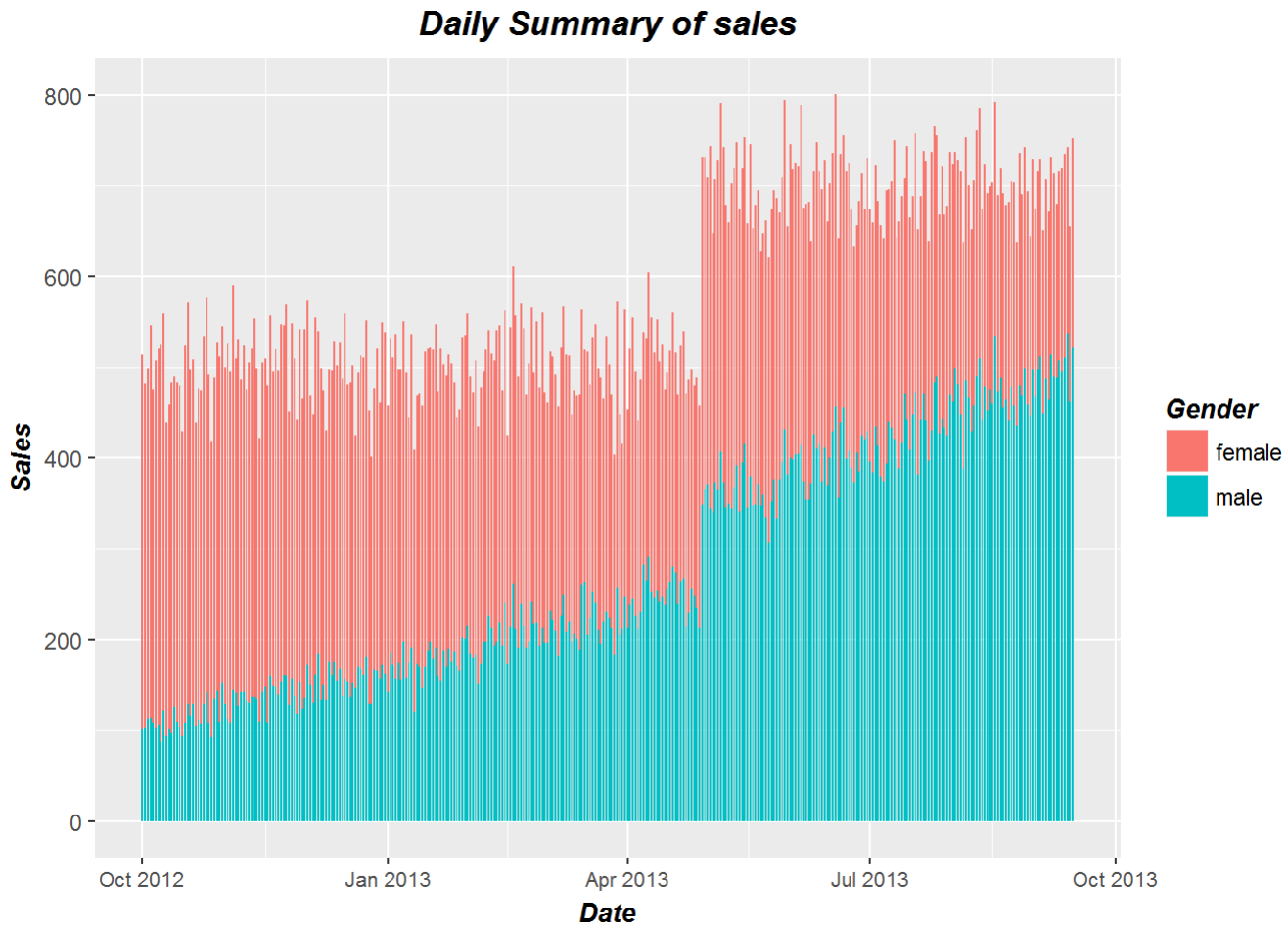
Question 1

Plotting daily sales

```
# Summarize daily sales
daily_summary_by_gender <- sales_df %>%
  group_by(sale_date) %>%
  mutate(total_by_day=n()) %>%
  ungroup()%>%
  group_by(sale_date,purchaser_gender) %>%
  mutate(percent_by_gender=round(n()*100/total_by_day)) %>%
  summarise(counts=n(),percent_by_gender=unique(percent_by_gender)) %>%
  rename(Sales=counts,Date=sale_date,Gender=purchaser_gender)

# Plotting daily sales
daily_plot <-  ggplot(daily_summary_by_gender, aes(x = Date, y = Sales, fill = Gender)) +
  geom_bar(position = position_stack(), stat = "identity", width = .7) +
  ggtitle("Daily Summary of sales")+
  theme(axis.title.x = element_text(face="bold.italic",size=10),axis.text.x  = element_text(vjus
t=0.5, size=8), axis.title.y = element_text(face="bold.italic",size=10),legend.title = element_t
ext(face="bold.italic",size=10),plot.title = element_text(face="bold.italic",hjust = 0.5))+
  labs(y="Sales", x = "Date")

print(daily_plot)
```



Question 2

Looking at the previous plot, it is clear that there is a sudden change in daily sales on a day in April 2013. Now, lets investigate to find the exact date. The day over day sales change would be maximum on the required day.

```
daily_summary <- daily_summary_by_gender %>%
  group_by(Date) %>%
  summarise(Sales=sum(Sales)) %>%
  mutate(dod_change = Sales-lag(Sales,1))

max(abs(daily_summary$dod_change),na.rm=T)
```

```
## [1] 274
```

```
daily_summary$Date[which(abs(daily_summary$dod_change)==max(abs(daily_summary$dod_change),na.rm=T)
```

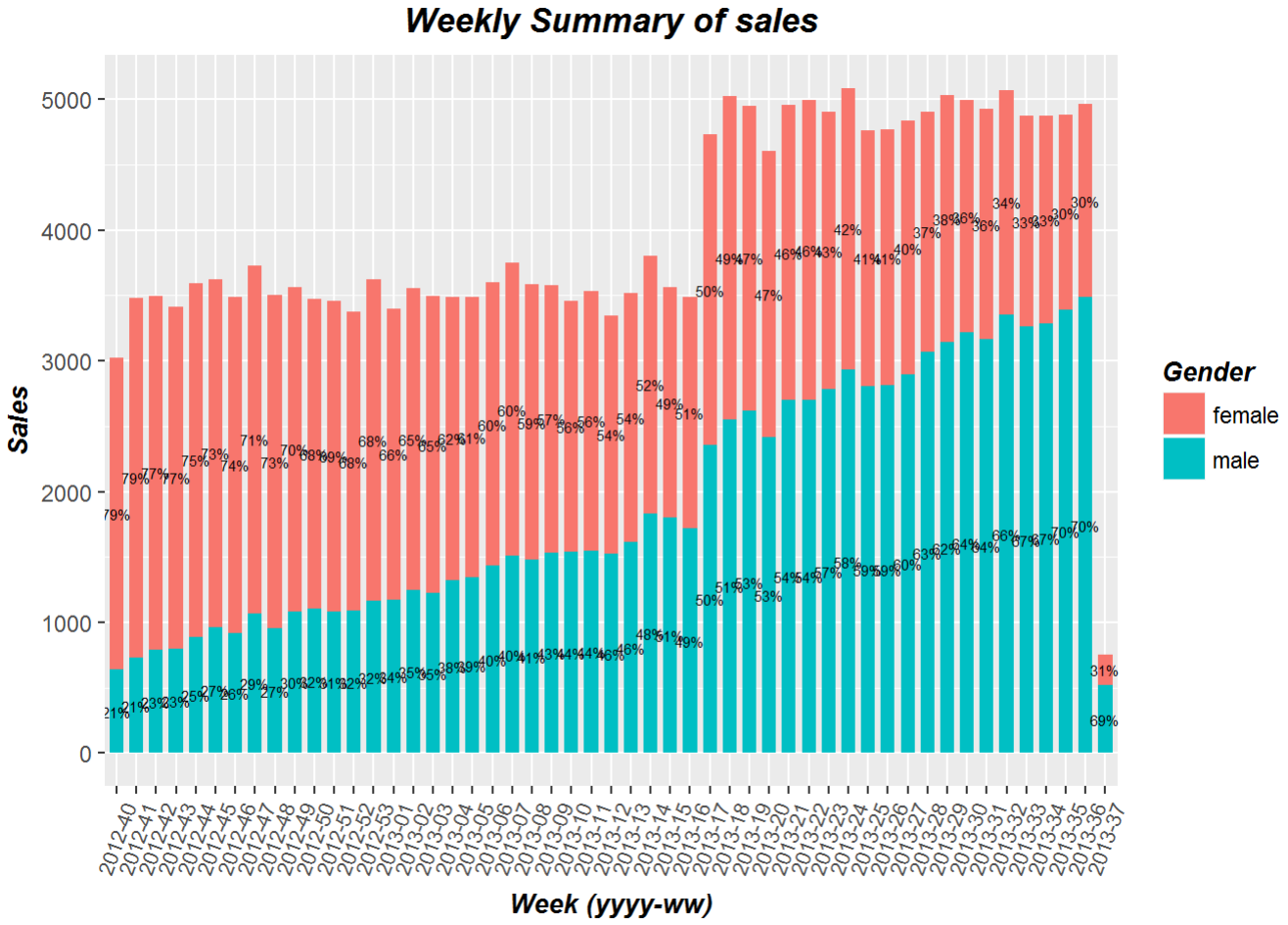
```
## [1] "2013-04-29"
```

The sudden increase in daily sales happened on 29th April 2013 and sales maintained that level from that day. This can be visualized graphically from the following weekly summary plots as well.

```
weekly_summary <- sales_df %>%
  group_by(week) %>%
  mutate(total_by_week=n()) %>%
  ungroup()%>%
  group_by(week,purchaser_gender) %>%
  mutate(percent_by_gender=round(n()*100/total_by_week)) %>%
  summarise(counts=n(),percent_by_gender=unique(percent_by_gender)) %>%
  rename(Sales=counts,Gender=purchaser_gender)

weekly_plot <- ggplot(weekly_summary, aes(x = week, y = Sales, fill = Gender)) +
  geom_bar(position = position_stack(), stat = "identity", width = .7) +
  ggtitle("Weekly Summary of sales") +
  geom_text(aes(label = paste0(percent_by_gender,"%")), position = position_stack(vjust = 0.5),
size = 2) +
  theme(axis.title.x = element_text(face="bold.italic",size=10),axis.text.x  = element_text(angl
e=70, vjust=0.5, size=8),
        axis.title.y = element_text(face="bold.italic",size=10),legend.title =
element_text(face="bold.italic",size=10),
        plot.title = element_text(face="bold.italic",hjust = 0.5))+
  labs(y="Sales", x = "Week (yyyy-ww)")

weekly_plot
```

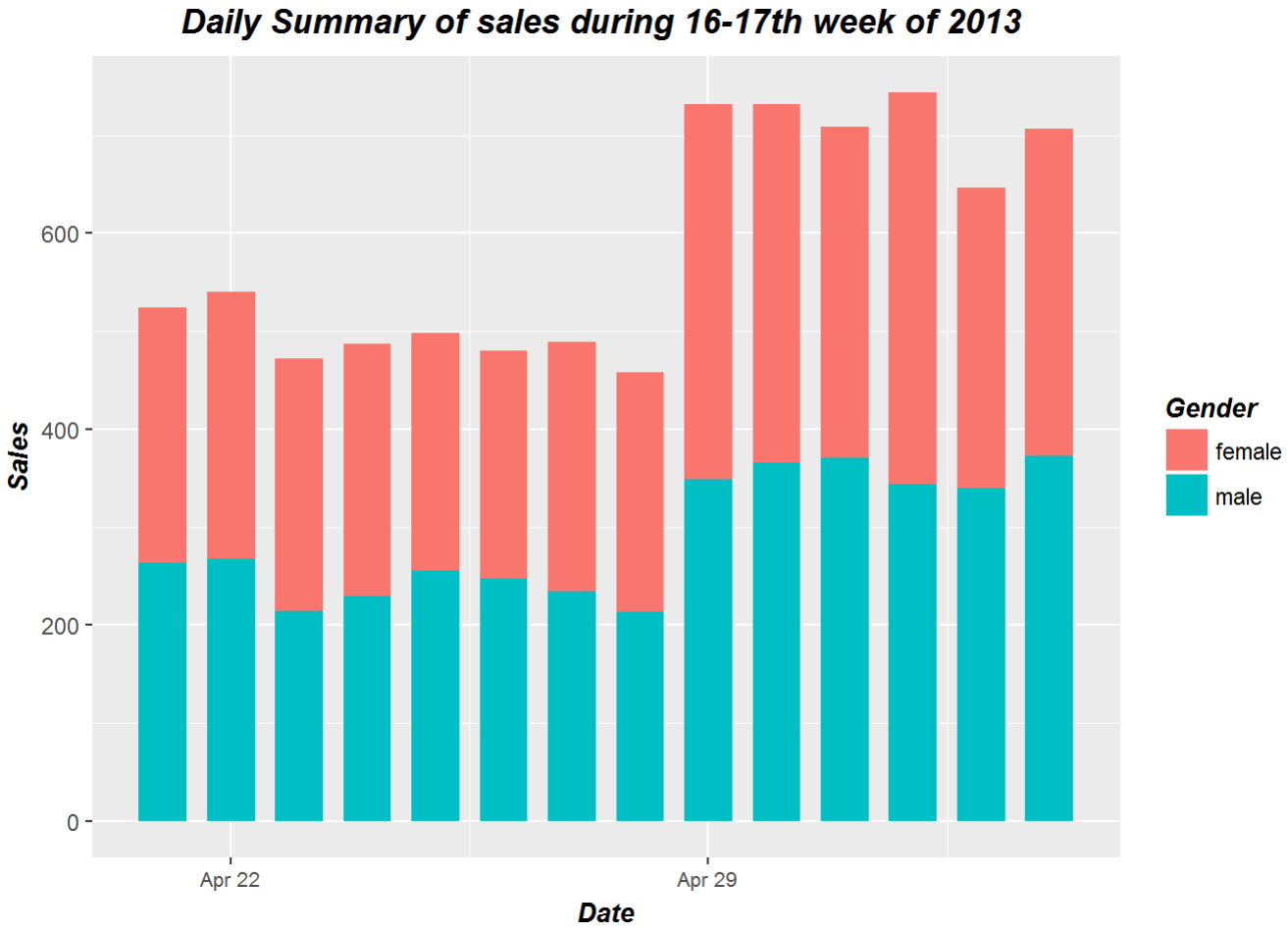


The plot shows that sudden increase happened during 16th week of 2013.

```
sales_spike <- sales_df %>%
  filter(week %in% c("2013-16","2013-17")) %>%
  group_by(sale_date,purchaser_gender) %>%
  summarise(counts=n()) %>%
  rename(Gender=purchaser_gender)

zoom_plot <- ggplot(sales_spike, aes(x = sale_date, y = counts, fill = Gender)) +
  geom_bar(position = position_stack(), stat = "identity", width = .7) +
  ggtitle("Daily Summary of sales during 16-17th week of 2013 ")+
  theme(axis.title.x = element_text(face="bold.italic",size=10),axis.text.x  = element_text(vjus
t=0.5, size=8),
        axis.title.y = element_text(face="bold.italic",size=10),legend.title =
element_text(face="bold.italic",size=10),
        plot.title = element_text(face="bold.italic",hjust = 0.5))+
  labs(y="Sales", x = "Date")

zoom_plot
```



The above plot clearly illustrates that sudden change in sales is happening on 29th April 2013.

Question 3

To check the statistical significance of the increase in sales, we can use two sample t-test. First divide the daily sales population into two samples. One sample containing the daily sales numbers before 29th April 2013 and other containing sales numbers from and after 29th April 2013.

Perform hypothesis testing on the following hypotheses. H0 : the two sample means are equal H1: mean(sample 1) < mean(sample 2)

```
sample1 <- daily_summary %>%
  filter(Date < as.Date("2013-04-29",format="%Y-%m-%d"))
sample1 <- sample1$Sales

sample2 <- daily_summary %>%
  filter(Date >= as.Date("2013-04-29",format="%Y-%m-%d"))
sample2 <- sample2$Sales

# Perform t-test
t.test(sample1,sample2,alternative="less",paired=FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  sample1 and sample2
## t = -45.944, df = 301.5, p-value < 2.2e-16
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##       -Inf -191.3646
## sample estimates:
## mean of x mean of y
##  504.4000  702.8929
```

Clearly `p-value < 2.2e-16` abtained above is less than alpha=0.05 and null hypothesis can be rejected. There is a strong evidence to state that sales increased on April 29th 2013 and after.

The sudden rise in daily sales could be due to a new acquisition or opening of new online store/outlet.

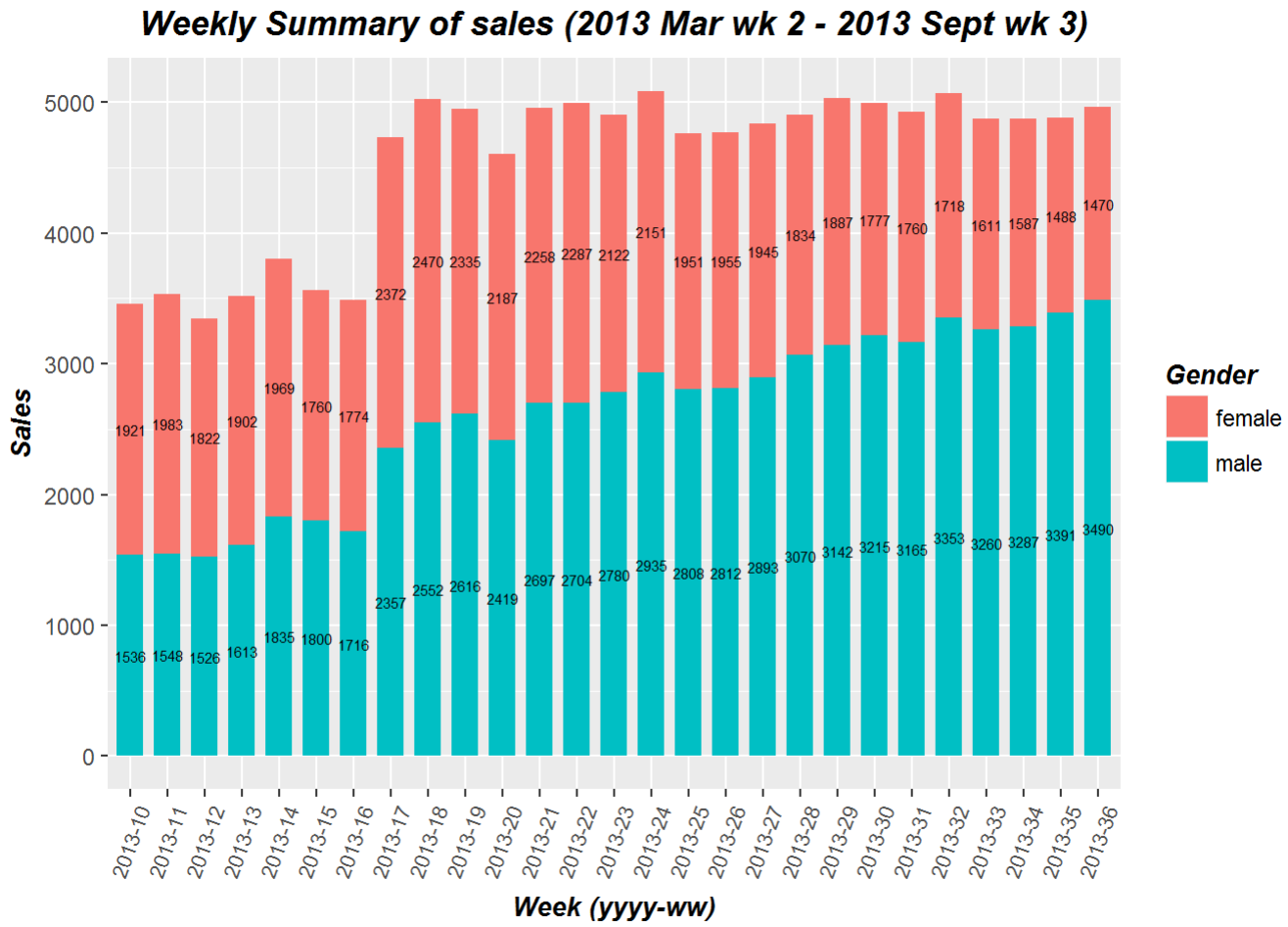
Question 4

Let's take a look at weekly summary to answer if shift in male vs female customer is driving sales.

```
required_weeks <- paste0("2013-",seq(10,36))
weekly_summary_part <- weekly_summary %>%
  filter(week %in% required_weeks)

weekly_plot <- ggplot(weekly_summary_part, aes(x = week, y = Sales, fill = Gender)) +
  geom_bar(position = position_stack(), stat = "identity", width = .7) +
  ggtitle("Weekly Summary of sales (2013 Mar wk 2 - 2013 Sept wk 3)") +
  geom_text(aes(label = Sales), position = position_stack(vjust = 0.5), size = 2) +
  theme(axis.title.x = element_text(face="bold.italic",size=10),axis.text.x  = element_text(angl
e=70, vjust=0.5, size=8),
        axis.title.y = element_text(face="bold.italic",size=10),legend.title =
element_text(face="bold.italic",size=10),
        plot.title = element_text(face="bold.italic",hjust = 0.5))+
  labs(y="Sales", x = "Week (yyyy-ww)")

weekly_plot
```



From the plot, it is clear that overall sales after 29th April increased compared to sales that existed before april 29th and increase in overall sales is driven by the increase in sales by both male and female customers. Further examining reveals that from 29th April to end of September, the sales by female customers have steadily decreased and sales by male customers have increased with overall sales almost at same level.

Question 5

Summarizing the daily sales by day parts

```
sales_by_dayparts <- sales_df %>%
  group_by(dayparts) %>%
  summarize(total_sales = n()) %>%
  ungroup() %>%
  mutate(percentage=paste0(round(total_sales*100/sum(total_sales)),"%"))
sales_by_dayparts
```

```
## # A tibble: 4 × 3
##    dayparts total_sales percentage
##       <chr>       <int>      <chr>
## 1 afternoon       80533        39%
## 2   evening       42620        21%
## 3   morning       62870        31%
## 4     night       18306         9%
```

In the given 54 week period, About 39% of sales happened during afternoon, 31% during morning, 21% during evening and only 9% during night.

Recommendation

Although lesser proportion of sales during night is expected, Implementing night time (12:00AM - 6:00AM) only promotions and offers could help drive the sales. However more analysis needs to be done to ensure night only promotions don't cannibalize sales in other parts of day before implementing them.

Session Info

```
sessionInfo()
```

```
## R version 3.3.2 (2016-10-31)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 7 x64 (build 7601) Service Pack 1
##
## locale:
## [1] LC_COLLATE=English_United States.1252
## [2] LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] ggplot2_2.2.1 dplyr_0.5.0
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.9      digest_0.6.11    rprojroot_1.2    assertthat_0.1
##  [5] plyr_1.8.4       grid_3.3.2       R6_2.2.0         gtable_0.2.0
##  [9] DBI_0.5-1        backports_1.1.0  magrittr_1.5     scales_0.4.1
## [13] evaluate_0.10    stringi_1.1.2    lazyeval_0.2.0   rmarkdown_1.3
## [17] labeling_0.3     tools_3.3.2      stringr_1.1.0    munsell_0.4.3
## [21] yaml_2.1.14      colorspace_1.3-2 htmltools_0.3.5  knitr_1.15.1
## [25] tibble_1.2
```