Assignment 1 Solution: Drizly Case

Sina Bahrami

2023-01-20

Question 1

Part 1.1

First, the csv file is assigned to the variable "drizly_df", then Weekdays field is calculated and added to it based on delivery dates, then N/A values are removed from Delivery.Time column.

```
drizly_df = read.csv("Case2.csv")
drizly_df$Order.Date <- as.Date(drizly_df$Order.Date, tryFormats = "%m/%d/%Y")
drizly_df$Weekday <- weekdays(drizly_df$Order.Date)
drizly_df$Delivery.Time = as.numeric(sub("N/A", "", drizly_df$Delivery.Time))
drizly_df$Retailer.ID <- as.character(drizly_df$Retailer.ID)
```

To obtain the average delivery time of each retailer on each weekday, first those records with empty delivery time are filtered out, then the data frame is grouped based on retailer ID and weekday. Finally, delivery time is averaged over each retailer and weekday. The results are given in the table below.

```
options(dplyr.summarise.inform = FALSE)
avg_del_time_day <- drizly_df %>%
  group_by(Retailer.ID,Weekday) %>%
  summarize(Avg.Del.Time = mean(Delivery.Time, na.rm = TRUE))
avg_del_time_day
```

```
## # A tibble: 21 x 3
## # Groups:   Retailer.ID [3]
##    Retailer.ID Weekday   Avg.Del.Time
##    <chr>       <chr>            <dbl>
##  1 1           Friday            97.3
##  2 1           Monday            47.3
##  3 1           Saturday         119.
##  4 1           Sunday            79
##  5 1           Thursday          51.6
##  6 1           Tuesday           48.9
##  7 1           Wednesday         34.7
##  8 2           Friday            91.7
##  9 2           Monday            40.6
## 10 2           Saturday         112.
## 11 2           Sunday            81.9
## 12 2           Thursday          35.0
## 13 2           Tuesday           34.9
```

```
## 14 2           Wednesday         41.1
## 15 3           Friday            89.8
## 16 3           Monday            35.9
## 17 3           Saturday         102.
## 18 3           Sunday            71.3
## 19 3           Thursday          41.2
## 20 3           Tuesday           35.7
## 21 3           Wednesday         42.6
```

With the obtained average values for different retailers and weekdays, two-way ANOVA is applied to do hypothesis test whether the average delivery time over the week is different for each retailer. The hypothesis is formulated as:

H0: Retailers have equal average of delivery time on each day H1: There are at least two retailers with different delivery time on each day

P-value(retailers) $= 0.032 < 0.05$ in the ANOVA table below, so H0 is rejected i.e. at least two retailers have different average delivery time on each day at level of significance of 0.05. However, at alpha < 0.032 there is no significant difference between them.

```
aov_result <- avg_del_time_day %>%
  aov(formula = Avg.Del.Time ~ Weekday + Retailer.ID)
summary(aov_result)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## Weekday      6  16109  2684.8  96.262 1.92e-09 ***
## Retailer.ID  2    259   129.3   4.635   0.0323 *
## Residuals   12    335    27.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Part 1.2

The void rate is obtained by counting the total number of voids and delivered orders for each retailer on each day then dividing the number of voids by corresponding total number of orders. The results are shown in the table below.

```
status_days <- drizly_df %>%
  group_by(Retailer.ID, Weekday) %>%
  summarise(Void.Count = sum(Order.Status == "Void"),
            Delivered.Count = sum(Order.Status == "Delivered"),
            Void.Rate = Void.Count/(Void.Count+Delivered.Count))
status_days
```

```
## # A tibble: 21 x 5
## # Groups:   Retailer.ID [3]
##    Retailer.ID Weekday   Void.Count Delivered.Count Void.Rate
##    <chr>       <chr>          <int>           <int>     <dbl>
##  1 1           Friday            37             164    0.184
##  2 1           Monday             5              70    0.0667
##  3 1           Saturday          57             218    0.207
##  4 1           Sunday            18             126    0.125
##  5 1           Thursday           4              94    0.0408
```

```
##  6 1           Tuesday            3              69    0.0417
##  7 1           Wednesday          0              56    0
##  8 2           Friday            25             132    0.159
##  9 2           Monday             4              58    0.0645
## 10 2           Saturday          49             162    0.232
## 11 2           Sunday            19             127    0.130
## 12 2           Thursday           2              53    0.0364
## 13 2           Tuesday            0              50    0
## 14 2           Wednesday          3              62    0.0462
## 15 3           Friday            22             101    0.179
## 16 3           Monday             2              44    0.0435
## 17 3           Saturday          39             151    0.205
## 18 3           Sunday            20             102    0.164
## 19 3           Thursday           3              63    0.0455
## 20 3           Tuesday            2              46    0.0417
## 21 3           Wednesday          4              54    0.0690
```

Two-way ANOVA is performed on the data frame obtained in the previous part for the hypothesis test:

H0: Retailers have equal void rate H1: There are at least two retailers with different void rate

P-value(retailers) $= 0.501 > 0.05$ in the ANOVA table below, so H0 is not rejected i.e. there is not enough evidence that retailers have different void rates at level of significance of 0.05.

```
status_days$Retailer.ID <- as.character(status_days$Retailer.ID)

aov_result <- status_days %>%
  aov(formula = Void.Rate ~ Weekday + Retailer.ID)
summary(aov_result)
```

```
##             Df  Sum Sq  Mean Sq F value   Pr(>F)
## Weekday      6 0.10355 0.017258  40.875 2.66e-07 ***
## Retailer.ID  2 0.00062 0.000309   0.733    0.501
## Residuals   12 0.00507 0.000422
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Part 1.3

The total GMV of each retailer is calculated below.

```
GMV_day <- drizly_df %>%
  group_by(Weekday, Retailer.ID) %>%
  summarize(GMV.Day = sum(GMV))
GMV_day
```

```
## # A tibble: 21 x 3
## # Groups:   Weekday [7]
##    Weekday   Retailer.ID GMV.Day
##    <chr>     <chr>         <int>
##  1 Friday    1             12832
##  2 Friday    2             10899
##  3 Friday    3              7887
```

```
##  4 Monday    1              6261
##  5 Monday    2              4021
##  6 Monday    3              2989
##  7 Saturday  1             18100
##  8 Saturday  2             13515
##  9 Saturday  3             12651
## 10 Sunday    1              9344
## 11 Sunday    2              9214
## 12 Sunday    3              7368
## 13 Thursday  1              6574
## 14 Thursday  2              3892
## 15 Thursday  3              4338
## 16 Tuesday   1              5223
## 17 Tuesday   2              3225
## 18 Tuesday   3              3234
## 19 Wednesday 1              3663
## 20 Wednesday 2              4435
## 21 Wednesday 3              4287
```

The result of two-way ANOVA shows that at least two retailers are different in terms of GMV on each day at 0.05 level of significance.

H0: Retailers have equal total GMV on each day H1: There are at least two retailers with different total GMV

P-value(retailers) = 0.0034 < 0.05 in the ANOVA table below, so H0 is rejected i.e. at least two retailers have different total GMV at level of significance 0.05.

```
aov_result <- GMV_day %>%
  aov(formula = GMV.Day ~ Weekday + Retailer.ID)
summary(aov_result)
```

```
##             Df    Sum Sq  Mean Sq F value   Pr(>F)
## Weekday      6 310193678 51698946  35.820 5.58e-07 ***
## Retailer.ID  2  27409261 13704630   9.495  0.00337 **
## Residuals   12  17319735  1443311
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The results obtained in part 1.1, 1.2 and 1.3 shows that at least two retailers are different in terms of Delivery time and GMV but there is not enough evidence that void rates are different at alpha=0.05.

Question 2

Part 2.1

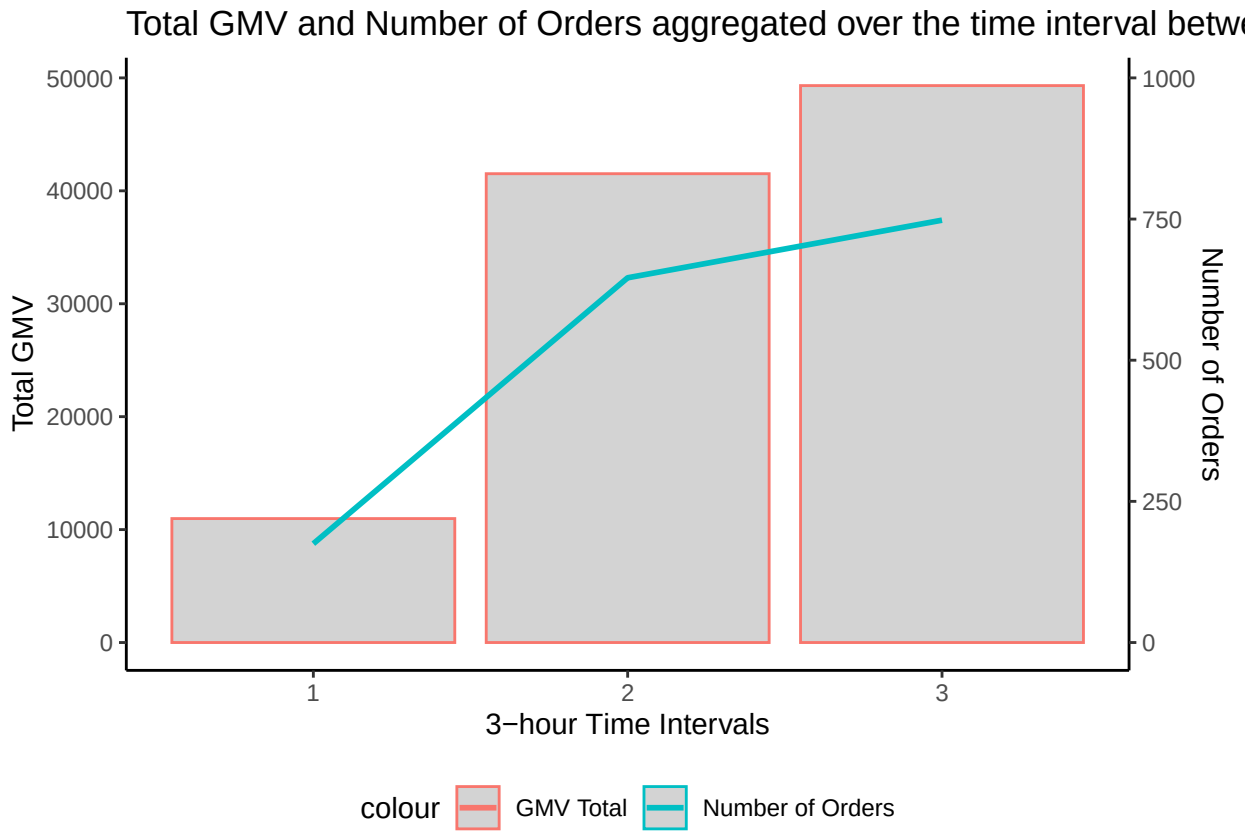
In the following , first the format of order date and time are corrected, then a new column is created to represent 3-hour time intervals called interval 1, 2 and 3 with interval 1 starting from 12:00 PM and so on. The limits of time intervals are stored in a variable called "tms", then order times are compared to each interval limit to populate interval column with corresponding interval number. The results are obtained first as aggregate value over all the three days and then over each separate day. A new data frame is created based on the intervals with corresponding total GVMs and number of orders of all Retailers aggregated over dates between 3rd and 5th April.

```
tms <- as.POSIXct(c("2020-01-01 12:00:00", "2020-01-01 15:00:00", "2020-01-01 18:00:00", "2020-01-01 21
tms <- format(tms, format = "%H:%M %p")
drizly_df$Order.Date <- as.Date(drizly_df$Order.Date)
drizly_df$Order.Time <- as.POSIXct(drizly_df$Order.Time, format = "%I:%M:%S %p")
drizly_df$Order.Time <- format(drizly_df$Order.Time, format = "%H:%M %p")

drizly_df$Interval <-
  1*(drizly_df$Order.Time >= tms[1] & drizly_df$Order.Time < tms[2]) +
  2*(drizly_df$Order.Time >= tms[2] & drizly_df$Order.Time < tms[3]) +
  3*(drizly_df$Order.Time >= tms[3] & drizly_df$Order.Time < tms[4]) +
  4*(drizly_df$Order.Time >= tms[4] & drizly_df$Order.Time < tms[5])

drizly_df_int <- drizly_df
drizly_df_int <- drizly_df_int %>%
  filter(Order.Date >= "2020-04-03" & Order.Date <= "2020-04-05") %>%
  group_by(Interval) %>%
  summarize(GMV.Total = sum(GMV), Order.Count = n())
drizly_df_int
```

```
## # A tibble: 3 x 3
##   Interval GMV.Total Order.Count
##      <dbl>     <int>       <int>
## 1        1     10975         175
## 2        2     41516         646
## 3        3     49319         748
```

```
coeff <- 50
ggplot(data = drizly_df_int, aes(x=Interval)) +
  geom_col(aes(y=GMV.Total, color = "GMV Total"), fill="light gray") +
  geom_line(aes(y=Order.Count*coeff, color = "Number of Orders"), size=1) +
  scale_y_continuous(
    name = "Total GMV",
    sec.axis = sec_axis(~./coeff, name="Number of Orders")
  ) +
  labs(x="3-hour Time Intervals", title="Total GMV and Number of Orders aggregated over the time interva
  theme_classic() +
  theme(legend.position="bottom")
```

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



Total GMV and number of orders between 3rd April and 5th April are provided in the figure above.

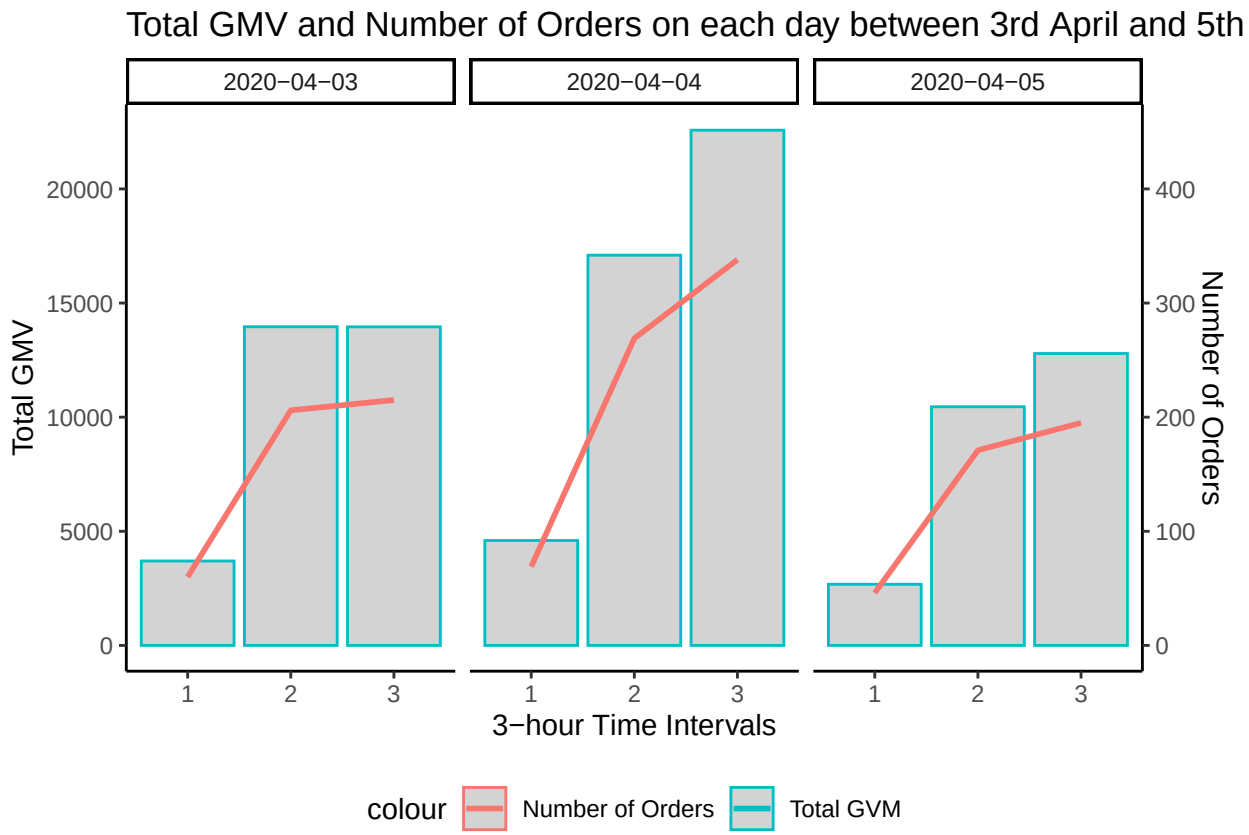
To compare the results day by day from 3rd to 5th April, similar calculations are carried out below:

```
drizly_df$Interval <-
  1*(drizly_df$Order.Time >= tms[1] & drizly_df$Order.Time < tms[2]) +
  2*(drizly_df$Order.Time >= tms[2] & drizly_df$Order.Time < tms[3]) +
  3*(drizly_df$Order.Time >= tms[3] & drizly_df$Order.Time < tms[4]) +
  4*(drizly_df$Order.Time >= tms[4] & drizly_df$Order.Time < tms[5])

drizly_df_int <- drizly_df
drizly_df_int <- drizly_df_int %>%
  filter(Order.Date >= "2020-04-03" & Order.Date <= "2020-04-05") %>%
  group_by(Interval, Order.Date) %>%
  summarize(GMV.Total = sum(GMV), Order.Count = n())
drizly_df_int
```

```
## # A tibble: 9 x 4
## # Groups:   Interval [3]
##   Interval Order.Date GMV.Total Order.Count
##      <dbl> <date>         <int>       <int>
## 1        1 2020-04-03      3699          60
## 2        1 2020-04-04      4597          69
## 3        1 2020-04-05      2679          46
## 4        2 2020-04-03     13962         206
## 5        2 2020-04-04     17097         269
```

```
## 6         2 2020-04-05      10457          171
## 7         3 2020-04-03      13957          215
## 8         3 2020-04-04      22572          338
## 9         3 2020-04-05      12790          195
```

```
coeff <- 50
ggplot(data = drizly_df_int, aes(x=Interval)) +
  geom_col(aes(y=GMV.Total, color="Total GVM"), fill="light gray") +
  geom_line(aes(y=Order.Count*coeff, color="Number of Orders"), size=1) +
  facet_grid(cols = vars(Order.Date)) +
  scale_y_continuous(
    name = "Total GMV",
    sec.axis = sec_axis(~./coeff, name="Number of Orders")
  ) +
  labs(x="3-hour Time Intervals", title="Total GMV and Number of Orders on each day between 3rd April a
  theme_classic() +
  theme(legend.position="bottom")
```



Part 2.2

As shown in the figures, number of orders and total GMV increase as it goes to 2nd and 3rd time interval with maximum at the 3rd time interval. However, between Interval 1 and 2 the growth rate is the highest. On 3rd April, the total GVM for interval 2 and 3 is almost the same and number of orders sees only a little increase compared to the other days. Also the results show that variations of number of orders and total GVM follow a similar pattern.

Part 2.3

Management can use these data to improve GMV in several ways, which can be categorized based on speed and impact:

- Slow and low impact
  - Forward deliveries that are cancelled by retailers to other retailers particularly at the 2nd and 3rd time.
  - Find more retailers and encourage them to use their services
- Slow but high impact
  - Adjust pricing of the products: considering higher pricing at the 1st interval and lower for the 2nd and 3rd.
  - Hire more staff who can work at the 2nd and 3rd interval
- Fast but low impact
  - Increase delivery fee for the 1st interval but decrease it for the 2nd and 3rd.
- Fast and high impact
  - Set a minimum on the price of products sold at the 2nd and 3rd interval
  - Accept scheduled order during the 2nd and 3rd interval