Analysis of Sales

Floriano Peixoto

Here we are going to treat the analysis of the sales data set and discover what it has to give us.

The data is described as bellow:

- PROD_ID: Product ID. the values varies between P1 to P9;
- DATE_ORDER: Sales Date, under YYYY-MM-DD format;
- QTY_ORDER: Quantity Sold;
- REVENUE: Sale revenue. There can be variations of the price for the same product, depending on the sales channel or discounts, which are applied to the base price

```
sales <-  read.csv("sales.csv", sep=",", stringsAsFactors = T)

tail(sales)
```

```
##        PROD_ID DATE_ORDER QTY_ORDER REVENUE
## 351086      P3 2015-10-04         1 1166.96
## 351087      P3 2015-09-24         1 1008.83
## 351088      P3 2015-10-13         2 2333.92
## 351089      P3 2015-09-24         1 1311.81
## 351090      P3 2015-10-13         1 1166.96
## 351091      P3 2015-10-10         1 1166.96
```

Now let´s prepare the date to make it easy to explore its features. The first thing we could do is sum the QTY_ORDER and the REVENUE by PROD_ID and DATE_ORDER to understand how much was sold in a date per product, than the DATE_ORDER field could be extracted into three: YEAR, MONTH and DAY, so we could use it later to understand more features of the data set. The product will be turned into a numeric representation as well.

```
sales_data_split <- mutate( sales, YEAR = lubridate::year(DATE_ORDER),
                            MONTH = lubridate::month(DATE_ORDER),
                            DAY = lubridate::day(DATE_ORDER),
                            PROD_ID = as.factor(PROD_ID)) %>%
                    group_by(PROD_ID, YEAR, MONTH, DAY) %>%
                    summarise(QTY_ORDER = sum(QTY_ORDER),
                           REVENUE = sum(REVENUE)) %>%
                    arrange(PROD_ID, YEAR, MONTH, DAY)
sales_data_split
```

```
## Source: local data frame [2,162 x 6]
## Groups: PROD_ID, YEAR, MONTH [79]
##
##    PROD_ID  YEAR MONTH   DAY QTY_ORDER  REVENUE
##     <fctr> <dbl> <dbl> <int>     <dbl>    <dbl>
## 1       P1  2015     2     4        10 14990.00
## 2       P1  2015     2     5        12 17688.20
## 3       P1  2015     2     6        21 31254.15
## 4       P1  2015     2     7         4  5996.00
## 5       P1  2015     2     8         7 10493.00
## 6       P1  2015     2     9         5  7420.05
## 7       P1  2015     2    10        10 13940.70
## 8       P1  2015     2    11         11 15659.30
## 9       P1  2015     2    12         16 22914.40
## 10      P1  2015     2    13          7 10318.05
## # ... with 2,152 more rows
```

Now we can see if the data has any correlation.

```
sales_data_corr <- mutate(sales_data_split, ID = as.numeric(PROD_ID)) %>%
                   group_by(ID, YEAR, MONTH, DAY) %>%
                   select(-PROD_ID)

cor.wt(sales_data_corr)
```

```
## Weighted Correlations
## Call:cor.wt(data = sales_data_corr)
##           YEAR  MONTH DAY   QTY_O REVEN ID
## YEAR       1.00
## MONTH      0.00  1.00
## DAY        0.00 -0.11  1.00
## QTY_ORDER  0.00  0.01  0.00  1.00
## REVENUE    0.00 -0.04 -0.01  0.99  1.00
## ID         0.00  0.07  0.01  0.18  0.16  1.00
```

The most relevant correlation between the fields is the QTY_ORDER and REVENUE, it is almost a direct relationship, we can think that happens because the REVENUE field indicates that total REVENUA not per item, so we could use another field that shows a ratio of REVENUE by WTY_ORDER.

```
sales_processed <- mutate(sales_data_split, REV_PER_ITEM = REVENUE / QTY_ORDER)

sales_processed
```

```
## Source: local data frame [2,162 x 7]
## Groups: PROD_ID, YEAR, MONTH [79]
##
##    PROD_ID  YEAR MONTH   DAY QTY_ORDER  REVENUE REV_PER_ITEM
##     <fctr> <dbl> <dbl> <int>     <dbl>    <dbl>        <dbl>
## 1       P1  2015     2     4        10 14990.00     1499.000
## 2       P1  2015     2     5        12 17688.20     1474.017
## 3       P1  2015     2     6        21 31254.15     1488.293
## 4       P1  2015     2     7         4  5996.00     1499.000
## 5       P1  2015     2     8         7 10493.00     1499.000
## 6       P1  2015     2     9         5  7420.05     1484.010
## 7       P1  2015     2    10        10 13940.70     1394.070
## 8       P1  2015     2    11        11 15659.30     1423.573
## 9       P1  2015     2    12        16 22914.40     1432.150
## 10      P1  2015     2    13         7 10318.05     1474.007
## # ... with 2,152 more rows
```

The most lucrative product

Now we´re ready to study the lucrative factor of the products.

The most lucrative of all times

We could find out the most lucrative product by the total REVENUE

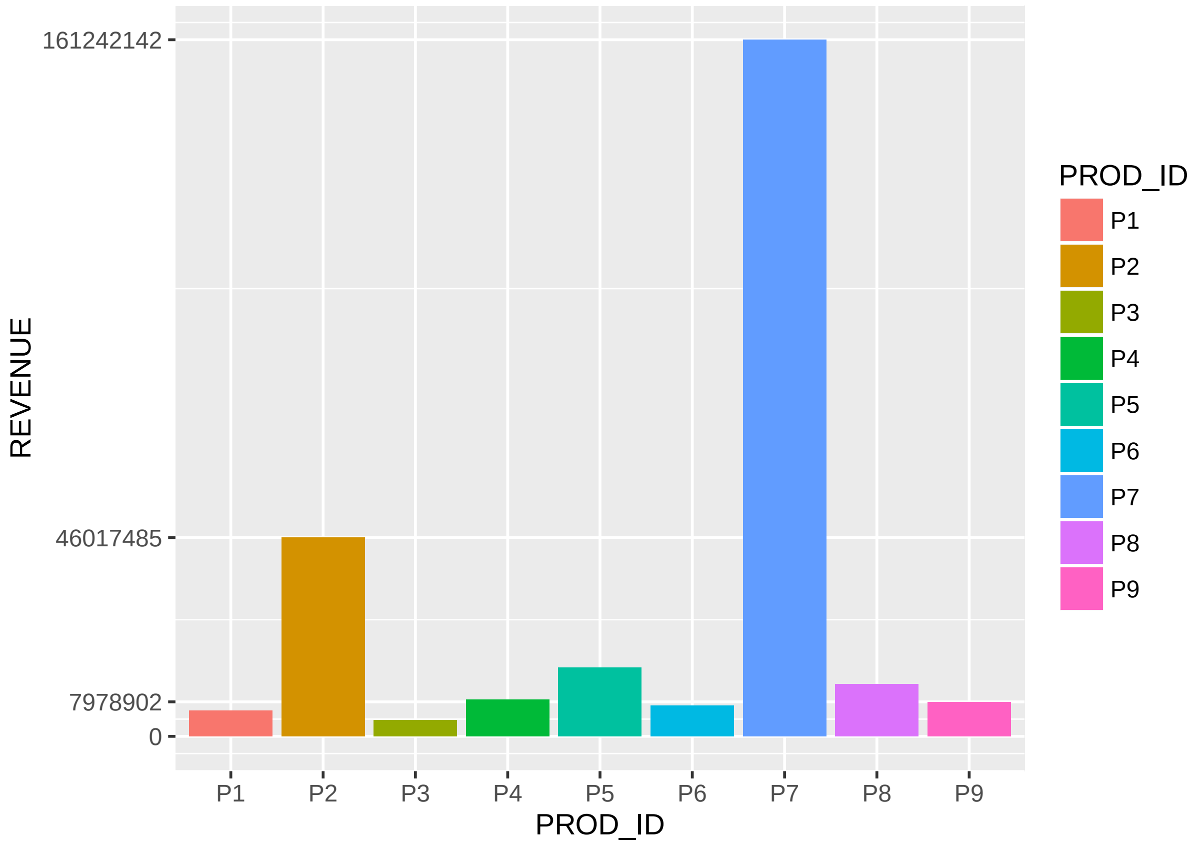
```
sales_most_lucrative <- group_by(sales_processed, PROD_ID) %>%
                        summarise(REVENUE = sum(REVENUE)) %>%
                        arrange(desc(REVENUE))

sales_most_lucrative
```

```
## # A tibble: 9 × 2
##   PROD_ID   REVENUE
##    <fctr>     <dbl>
## 1      P7 161242142
## 2      P2  46017485
## 3      P5  15907636
## 4      P8  12157680
## 5      P4   8456698
## 6      P9   7978902
## 7      P6   7079520
## 8      P1   6014097
## 9      P3   3714170
```

```
g <- ggplot(sales_most_lucrative, aes(y = REVENUE, x = PROD_ID))

g +  geom_bar(stat = "identity", aes(fill = PROD_ID, PROD_ID), position = "dodge") +
   scale_y_continuous(breaks = c(0,7978902,46017485,161242142))
```



As we can see the P7 is the most lucrative product of all time by far, follwed by P2. The others products seem to be very close to each other in perspective.

The most lucrative month

Let´s discover what is the most lucrative month of the year in the data set history, for that we´ll calculate the median of the REVENUE by month.

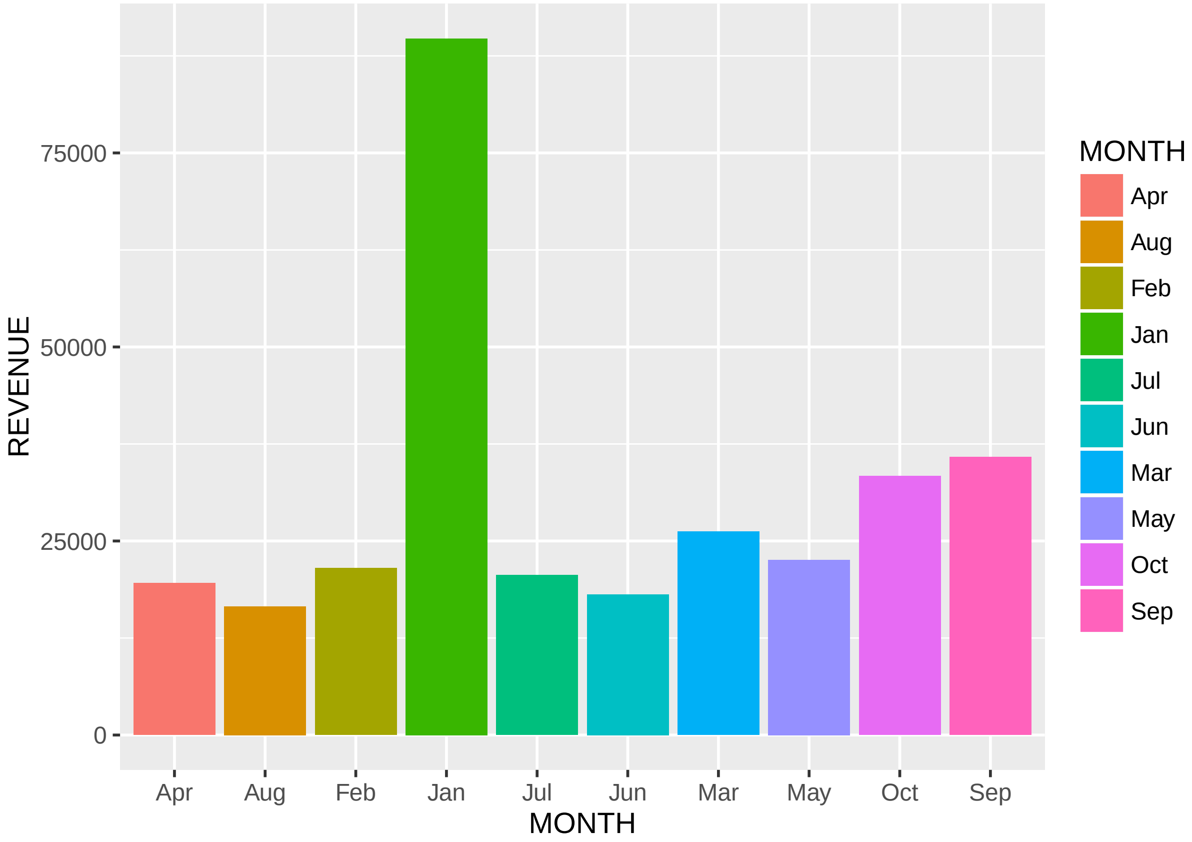
```
sales_month <- group_by(sales_processed, MONTH) %>%
               summarise(REVENUE = median(REVENUE)) %>%
               mutate(MONTH = month.abb[MONTH]) %>%
               arrange(desc(REVENUE))

sales_month
```

```
## # A tibble: 10 × 2
##    MONTH  REVENUE
##    <chr>    <dbl>
## 1    Jan 89751.35
## 2    Sep 35826.10
## 3    Oct 33383.78
## 4    Mar 26229.00
## 5    May 22584.28
## 6    Feb 21510.55
## 7    Jul 20614.13
## 8    Apr 19596.29
## 9    Jun 18121.25
## 10   Aug 16583.17
```

```
g <- ggplot(sales_month, aes(y = REVENUE, x = MONTH))

g +  geom_bar(stat = "identity", aes(fill = MONTH), position = "dodge")
```



The first thing we notice is that the data does not register sales for november and december. But January is the month with more sales by far, followed by September and October.

The distribuition of product sale per month

Let´s try to discover how the sales by product are distribuited over the months.

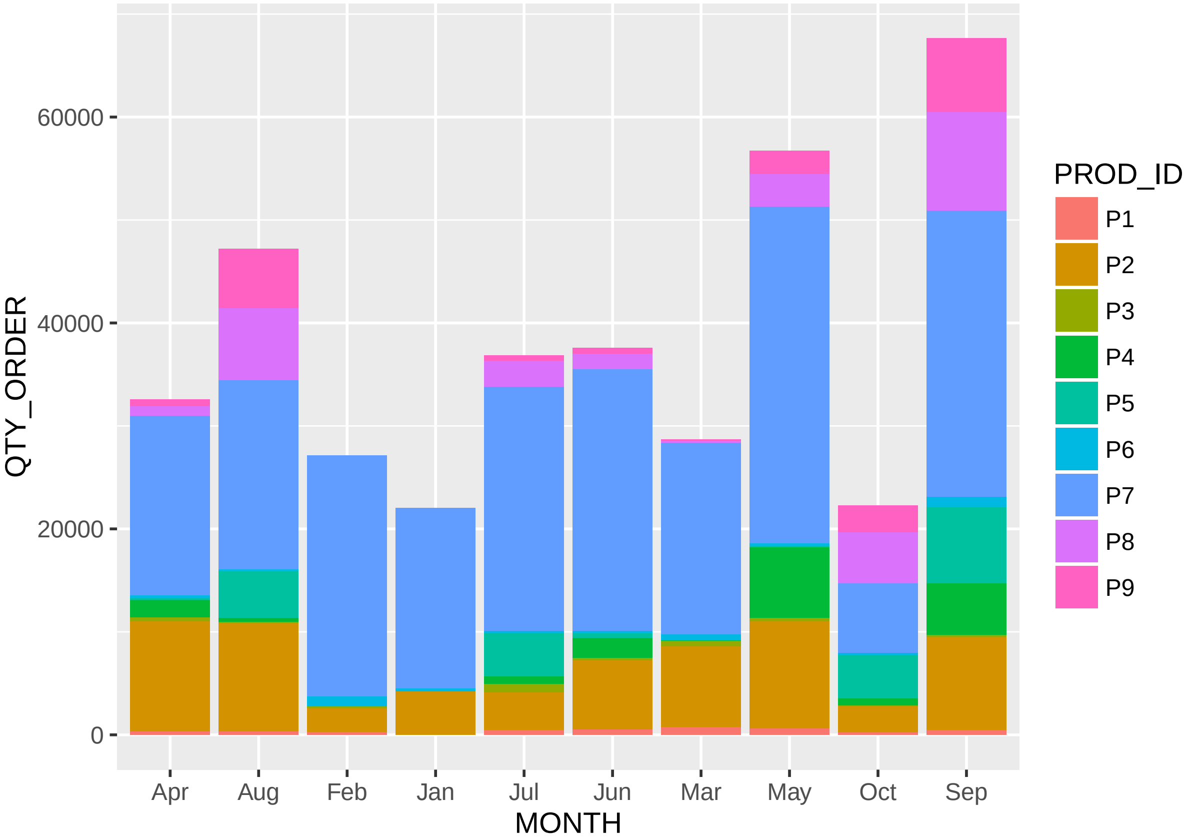
```
sales_product_month <- group_by(sales_processed,PROD_ID, MONTH) %>%
                       summarise(QTY_ORDER = sum(QTY_ORDER)) %>%
                       mutate( MONTH = month.abb[MONTH]) %>%
                       arrange(PROD_ID, MONTH, QTY_ORDER)

sales_product_month
```

```
## Source: local data frame [79 x 3]
## Groups: PROD_ID [9]
##
##    PROD_ID MONTH QTY_ORDER
##     <fctr> <chr>     <dbl>
## 1       P1   Apr       354
## 2       P1   Aug       341
## 3       P1   Feb       307
## 4       P1   Jul       478
## 5       P1   Jun       503
## 6       P1   Mar       791
## 7       P1   May       666
## 8       P1    Oct       229
## 9       P1    Sep       504
## 10      P2    Apr     10664
## # ... with 69 more rows
```

```
g <- ggplot(sales_product_month, aes(y = QTY_ORDER, x = MONTH))

g +  geom_bar(stat = "identity", aes(fill = PROD_ID))
```



September is the month with more sales and the product P6 sells much more than the others products. Some products seem to have no sales in some months like P8 and P9 in Feb and Jan.