Data Analysis and Visualization - Assignment 2

Ma Jingchun, 2020111235

利用tidyverse的框架，基于数据dataset-cac-ma.xlsx 进行如下内容

（1）将数据读入R，并保存成tibble 类型的数据。展示该数据框的前6行。

```
library(tidyverse)
library(readxl)
library(tibble)
library(dplyr)
library(lubridate)
```

```
########## Please write your R code in this chunk ##########
### Solution to Q1
d <- as_tibble(read_excel("dataset-cac-ma.xlsx",1))
head(d)
```

（2）展示该数据的所有变量的变量类型和描述性统计。

```
########## Please write your R code in this chunk ##########
### Solution to Q2
# 变量类型
str(d)
```

```
## tibble [49,839 × 13] (S3: tbl_df/tbl/data.frame)
##  $ customer_number    : num [1:49839] 20943 126101 161675 175749 216582 ...
##  $ region             : chr [1:49839] "Midwest" "Northwest" "West" "West" ...
##  $ date_of_sale       : POSIXct[1:49839], format: "2015-01-01" "2015-01-01" ...
##  $ item               : chr [1:49839] "918DP" "2981AB" "910-128PC" "351-128PC" ...
##  $ brand              : chr [1:49839] "Jeffrey Alexander" "Elements" "Jeffrey Alexander" "El
ements" ...
##  $ collection         : chr [1:49839] "Prestige" "Florence" "Modena" "Calloway" ...
##  $ description        : chr [1:49839] "Knob" "3\" pull" "128 mm CC pull" "128\" CC pull" ...
##  $ list_price         : num [1:49839] 14.14 6.83 17.68 7.63 2.52 ...
##  $ cost               : num [1:49839] 8.62 4.27 11.08 4.85 1.6 ...
##  $ quantity_sold      : num [1:49839] 434 54 450 467 380 689 85 649 100 762 ...
##  $ sales revenue      : logi [1:49839] NA NA NA NA NA NA ...
##  $ variable cost      : logi [1:49839] NA NA NA NA NA NA ...
##  $ contribution margin: logi [1:49839] NA NA NA NA NA NA ...
```

```
# 描述性统计
summary(d)
```

```
##  customer_number     region           date_of_sale
##  Min.   :    32   Length:49839       Min.   :2015-01-01 00:00:00.00
##  1st Qu.:240110   Class :character   1st Qu.:2016-03-21 00:00:00.00
##  Median :480402   Mode  :character   Median :2017-04-11 00:00:00.00
##  Mean   :491308                      Mean   :2017-03-07 04:43:19.50
##  3rd Qu.:746519                      3rd Qu.:2018-03-12 00:00:00.00
##  Max.   :999853                      Max.   :2018-12-31 00:00:00.00
##      item              brand            collection        description
##  Length:49839       Length:49839       Length:49839       Length:49839
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##    list_price            cost        quantity_sold    sales revenue
##  Min.   :  0.7237   Min.   :  0.77   Min.   :   7.0   Mode:logical
##  1st Qu.:  5.2000   1st Qu.:  3.29   1st Qu.: 263.0   NA's:49839
##  Median :  8.7900   Median :  5.50   Median : 516.0
##  Mean   : 18.3041   Mean   : 11.17   Mean   : 531.5
##  3rd Qu.: 14.9900   3rd Qu.:  9.14   3rd Qu.: 769.0
##  Max.   :167.5000   Max.   :108.40   Max.   :2006.0
##  variable cost  contribution margin
##  Mode:logical   Mode:logical
##  NA's:49839     NA's:49839
##
##
##
##
```

(3) 选择第 5-10 行在 list_price 和 cost 这两列上面的数据。

```
########## Please write your R code in this chunk #########
### Solution to Q3
d1 <- select(d, list_price, cost)[5:10,]
d1
```

(4) 分别选择 customer 为 175749 的所有数据, 以及 region 上取值为 Midwest 的所有数据

```
########## Please write your R code in this chunk #########
### Solution to Q4
d2 <- filter(d, customer_number == 175749)
d2
```

```
d3 <- filter(d, region == "Midwest")
d3
```

(5) 该数据中有多少个不同的region？其中是否有错误的数据? 如果有，请改正。

```
########## Please write your R code in this chunk #########
### Solution to Q5
# region 不同取值个数
length(unique(d$region))
```

```
## [1] 10
```

```
# 修改前，有错误数据
aggregate(d$item,by=list(region=d$region),length)
```

```
d[d$region=="Centrall","region"] <- "Central"
d[d$region=="Soouth","region"] <- "South"
# 修改后
length(unique(d$region))
```

```
## [1] 8
```

```
aggregate(d$item,by=list(region=d$region),length)
```

实际上共有8个region

(6) 计算 sales revenue， variable cost 和 contribution margin 的数值，其中：

- sales revenue = quantity sold * list_price
- variable cost = quantity sold * cost
- contribution margin = (sales revenue - variable cost) / sales revenue

```
######### Please write your R code in this chunk #########
### Solution to Q6
d <- mutate(d,
  "sales revenue" = quantity_sold * list_price,
  "variable cost" = quantity_sold * cost,
  "contribution margin" = (`sales revenue` - `variable cost`) / `sales revenue`
  )
```

(7) 根据 year 和 quarter 的信息，计算每个地区的 contribution margin 的平均值。

```
######### Please write your R code in this chunk #########
### Solution to Q7
d <- mutate(d,
  year_quarter = paste(year(date_of_sale),quarter(date_of_sale),sep="_")
  )
d %>%
  group_by( region, year_quarter) %>%
  summarise( ave_contribution_margin=mean(`contribution margin`, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'region'. You can override using the
## `.groups` argument.
```

(8) 展示每年中，contribution margin 平均值最高的前 3 个 region。这个名单随着年份变化而变化吗？

```
######### Please write your R code in this chunk #########
### Solution to Q8
d <- mutate(d,
  year = year(date_of_sale)
  )
d %>%
  group_by( year,region) %>%
  summarise( ave_contribution_margin=mean(`contribution margin`, na.rm = TRUE)) %>%
  arrange( desc(ave_contribution_margin),.by_group = TRUE) %>%
  filter(row_number() <= 3)
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

名单随年份变化

(9) 每年中，最赚钱的 collection 前 3 名分别是什么?

```
########## Please write your R code in this chunk #########
### Solution to Q9
d %>%
  group_by( year, collection) %>%
  summarise( profit = sum(`sales revenue` - `variable cost`, na.rm = TRUE)) %>%
  arrange( desc(profit), .by_group = TRUE) %>%
  filter(row_number() <= 3)
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

注意：这里profit = sales revenue - variable cost

(10) 2018 年中，每个 brand 最赚钱的 collection 是什么?

```
########## Please write your R code in this chunk #########
### Solution to Q10
d %>%
  filter( year == 2018) %>%
  group_by( brand, collection) %>%
  summarise( profit = sum(`sales revenue` - `variable cost`, na.rm = TRUE)) %>%
  arrange( desc(profit), .by_group = TRUE) %>%
  filter(row_number() <= 1)
```

```
## `summarise()` has grouped output by 'brand'. You can override using the
## `.groups` argument.
```