Solution to the Quantium Data Analytics Virtual Internship Program - Task 1

Chinonso Emezie

15/09/2021

#Load required libraries

library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5 v purrr 0.3.4  
## v tibble 3.1.2 v dplyr 1.0.6  
## v tidyr 1.1.3 v stringr 1.4.0  
## v readr 1.4.0 v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(data.table)

##   
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':  
##   
## between, first, last

## The following object is masked from 'package:purrr':  
##   
## transpose

library(ggplot2)  
library(ggmosaic)

## Warning: package 'ggmosaic' was built under R version 4.1.1

library(readr)  
library(stringr)  
library(lessR)

## Warning: package 'lessR' was built under R version 4.1.1

##   
## lessR 4.0.2 feedback: gerbing@pdx.edu web: lessRstats.com/new  
## ---------------------------------------------------------------  
## > d <- Read("") Read text, Excel, SPSS, SAS, or R data file  
## d is default data frame, data= in analysis routines optional  
##   
## Learn about reading, writing, and manipulating data, graphics,  
## testing means and proportions, regression, factor analysis,  
## customization, and descriptive statistics from pivot tables.  
## Enter: browseVignettes("lessR")  
##   
## View changes in this new version of lessR.  
## Enter: help(package=lessR) Click: Package NEWS

##   
## Attaching package: 'lessR'

## The following object is masked from 'package:data.table':  
##   
## set

## The following object is masked from 'package:dplyr':  
##   
## recode

library(scales)

##   
## Attaching package: 'scales'

## The following object is masked from 'package:lessR':  
##   
## rescale

## The following object is masked from 'package:purrr':  
##   
## discard

## The following object is masked from 'package:readr':  
##   
## col\_factor

#Import transaction data set and purchase behavior data set

library(readxl)  
QVI\_transaction\_data <- read\_excel("~/Desktop/Quantium/QVI\_transaction\_data.xlsx")  
View(QVI\_transaction\_data)

library(readr)  
QVI\_purchase\_behaviour <- read\_csv("C:/Users/abdul/Downloads/QVI\_purchase\_behaviour.csv")

##   
## -- Column specification --------------------------------------------------------  
## cols(  
## LYLTY\_CARD\_NBR = col\_double(),  
## LIFESTAGE = col\_character(),  
## PREMIUM\_CUSTOMER = col\_character()  
## )

View(QVI\_purchase\_behaviour)

#Examining closely the transaction data set

str(QVI\_transaction\_data)

## tibble [264,836 x 8] (S3: tbl\_df/tbl/data.frame)  
## $ DATE : num [1:264836] 43390 43599 43605 43329 43330 ...  
## $ STORE\_NBR : num [1:264836] 1 1 1 2 2 4 4 4 5 7 ...  
## $ LYLTY\_CARD\_NBR: num [1:264836] 1000 1307 1343 2373 2426 ...  
## $ TXN\_ID : num [1:264836] 1 348 383 974 1038 ...  
## $ PROD\_NBR : num [1:264836] 5 66 61 69 108 57 16 24 42 52 ...  
## $ PROD\_NAME : chr [1:264836] "Natural Chip Compny SeaSalt175g" "CCs Nacho Cheese 175g" "Smiths Crinkle Cut Chips Chicken 170g" "Smiths Chip Thinly S/Cream&Onion 175g" ...  
## $ PROD\_QTY : num [1:264836] 2 3 2 5 3 1 1 1 1 2 ...  
## $ TOT\_SALES : num [1:264836] 6 6.3 2.9 15 13.8 5.1 5.7 3.6 3.9 7.2 ...

#Converting DATE column to a date format

QVI\_transaction\_data$DATE <- as.Date(QVI\_transaction\_data$DATE, origin = "1899-12-30")

#Examining PROD\_NAME column

head(QVI\_transaction\_data$PROD\_NAME)

## [1] "Natural Chip Compny SeaSalt175g"   
## [2] "CCs Nacho Cheese 175g"   
## [3] "Smiths Crinkle Cut Chips Chicken 170g"   
## [4] "Smiths Chip Thinly S/Cream&Onion 175g"   
## [5] "Kettle Tortilla ChpsHny&Jlpno Chili 150g"  
## [6] "Old El Paso Salsa Dip Tomato Mild 300g"

#Examine the words in PROD\_NAME to see if there are any incorrect entries such as products that are not chips

productWords <- data.table(unlist(strsplit(unique(QVI\_transaction\_data$PROD\_NAME), " ")))  
setnames(productWords, 'words')

#Removing digits

productWords <- productWords[grepl("\\d", words) == FALSE, ]  
View(productWords)

#Removing special characters.

productWords <- productWords[grepl("[:alpha:]", words), ]  
View(productWords)

#Let’s look at the words and the frequency of each word.

productWords %>% count(productWords$words) %>% arrange(n, desc())

## productWords$words n  
## 1: &Chives 1  
## 2: &OnionStacked 1  
## 3: &Sr/Cream 1  
## 4: Aioli 1  
## 5: Bacon 1  
## ---   
## 127: Salt 12  
## 128: Kettle 13  
## 129: Crinkle 14  
## 130: Smiths 16  
## 131: Chips 21

#Removing salsa products.

QVI\_transaction\_data = QVI\_transaction\_data %>% filter(!str\_detect(PROD\_NAME, "Salsa"))

#Summarize the data to check for nulls and possible outliers.

summary(QVI\_transaction\_data)

## DATE STORE\_NBR LYLTY\_CARD\_NBR TXN\_ID   
## Min. :2018-07-01 Min. : 1.0 Min. : 1000 Min. : 1   
## 1st Qu.:2018-09-30 1st Qu.: 70.0 1st Qu.: 70015 1st Qu.: 67569   
## Median :2018-12-30 Median :130.0 Median : 130367 Median : 135183   
## Mean :2018-12-30 Mean :135.1 Mean : 135531 Mean : 135131   
## 3rd Qu.:2019-03-31 3rd Qu.:203.0 3rd Qu.: 203084 3rd Qu.: 202654   
## Max. :2019-06-30 Max. :272.0 Max. :2373711 Max. :2415841   
## PROD\_NBR PROD\_NAME PROD\_QTY TOT\_SALES   
## Min. : 1.00 Length:246742 Min. : 1.000 Min. : 1.700   
## 1st Qu.: 26.00 Class :character 1st Qu.: 2.000 1st Qu.: 5.800   
## Median : 53.00 Mode :character Median : 2.000 Median : 7.400   
## Mean : 56.35 Mean : 1.908 Mean : 7.321   
## 3rd Qu.: 87.00 3rd Qu.: 2.000 3rd Qu.: 8.800   
## Max. :114.00 Max. :200.000 Max. :650.000

#Filter the data set to find the outlier because the mean of the PROD\_QTY is 1.908 and the max PROD\_QTY is 200.

QVI\_transaction\_data %>% filter(PROD\_QTY == 200)

## # A tibble: 2 x 8  
## DATE STORE\_NBR LYLTY\_CARD\_NBR TXN\_ID PROD\_NBR PROD\_NAME PROD\_QTY  
## <date> <dbl> <dbl> <dbl> <dbl> <chr> <dbl>  
## 1 2018-08-19 226 226000 226201 4 Dorito Corn Chp ~ 200  
## 2 2019-05-20 226 226000 226210 4 Dorito Corn Chp ~ 200  
## # ... with 1 more variable: TOT\_SALES <dbl>

#Filtering out the outlier based on the PROD\_QTY

QVI\_transaction\_data <- QVI\_transaction\_data[!(QVI\_transaction\_data$PROD\_QTY==200),]  
QVI\_transaction\_data %>% filter(PROD\_QTY == 200)

## # A tibble: 0 x 8  
## # ... with 8 variables: DATE <date>, STORE\_NBR <dbl>, LYLTY\_CARD\_NBR <dbl>,  
## # TXN\_ID <dbl>, PROD\_NBR <dbl>, PROD\_NAME <chr>, PROD\_QTY <dbl>,  
## # TOT\_SALES <dbl>

#Re-examine transaction data

summary(QVI\_transaction\_data)

## DATE STORE\_NBR LYLTY\_CARD\_NBR TXN\_ID   
## Min. :2018-07-01 Min. : 1.0 Min. : 1000 Min. : 1   
## 1st Qu.:2018-09-30 1st Qu.: 70.0 1st Qu.: 70015 1st Qu.: 67569   
## Median :2018-12-30 Median :130.0 Median : 130367 Median : 135182   
## Mean :2018-12-30 Mean :135.1 Mean : 135530 Mean : 135130   
## 3rd Qu.:2019-03-31 3rd Qu.:203.0 3rd Qu.: 203083 3rd Qu.: 202652   
## Max. :2019-06-30 Max. :272.0 Max. :2373711 Max. :2415841   
## PROD\_NBR PROD\_NAME PROD\_QTY TOT\_SALES   
## Min. : 1.00 Length:246740 Min. :1.000 Min. : 1.700   
## 1st Qu.: 26.00 Class :character 1st Qu.:2.000 1st Qu.: 5.800   
## Median : 53.00 Mode :character Median :2.000 Median : 7.400   
## Mean : 56.35 Mean :1.906 Mean : 7.316   
## 3rd Qu.: 87.00 3rd Qu.:2.000 3rd Qu.: 8.800   
## Max. :114.00 Max. :5.000 Max. :29.500

# Checking for missing DATE in the DATE column

FullSeq <- seq.Date(from = min(QVI\_transaction\_data$DATE), to = max(QVI\_transaction\_data$DATE), by = 1)  
Missing <- FullSeq[!FullSeq %in% QVI\_transaction\_data$DATE]

#Calculate the total sales for each day

Total\_Sales\_D = QVI\_transaction\_data %>% group\_by(DATE) %>% summarise(TOTAL\_SALES = sum(TOT\_SALES)) %>%  
 arrange(DATE, desc())  
View(Total\_Sales\_D)

#Check the total number of days transactions occurred in our dataset.

length(Total\_Sales\_D$DATE)

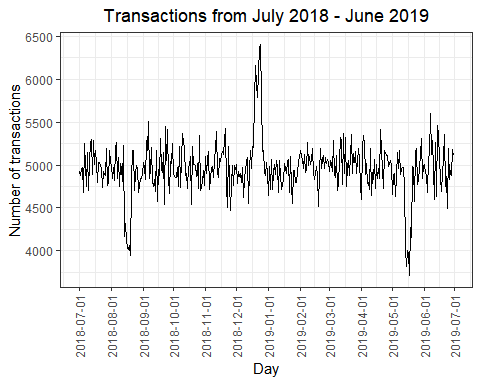
## [1] 364

#Set plot themes to format graphs

theme\_set(theme\_bw())  
theme\_update(plot.title = element\_text(hjust = 0.5))

#Plot transactions from July 218 - June 2019

ggplot(Total\_Sales\_D, aes(x = DATE, y = TOTAL\_SALES)) +  
 geom\_line() +  
 labs(x = "Day", y = "Number of transactions", title = "Transactions from July 2018 - June 2019") +  
 scale\_x\_date(breaks = "1 month") +  
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5))

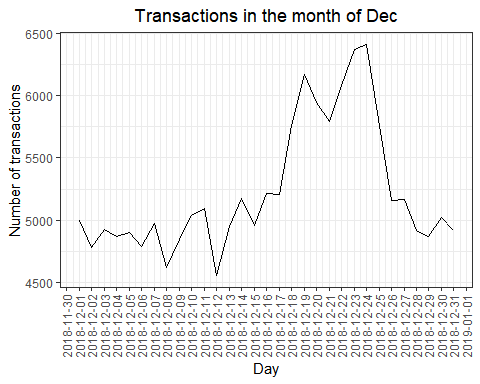


#Filter out the transactions that occurred in the month of December and look at individual days

Total\_Sales\_Dec = Total\_Sales\_D %>% filter(DATE > "2018-11-30" & DATE < "2019-01-01")  
View(Total\_Sales\_Dec)

#Plot the daily transactions in the month of December.

ggplot(Total\_Sales\_Dec, aes(x = DATE, y = TOTAL\_SALES)) +  
 geom\_line() +  
 labs(x = "Day", y = "Number of transactions", title = "Transactions in the month of Dec") +  
 scale\_x\_date(breaks = "1 day") +  
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5))



#Creating another Pack size column from the Product name column

QVI\_transaction\_data$PACK\_SIZE <- str\_extract(QVI\_transaction\_data$PROD\_NAME, "[-+.e0-9]\*\\d")

#Convert PACK\_SIZE to numeric and sorting in ascending

QVI\_transaction\_data$PACK\_SIZE <- as.numeric(as.character(QVI\_transaction\_data$PACK\_SIZE))  
class(QVI\_transaction\_data$PACK\_SIZE)

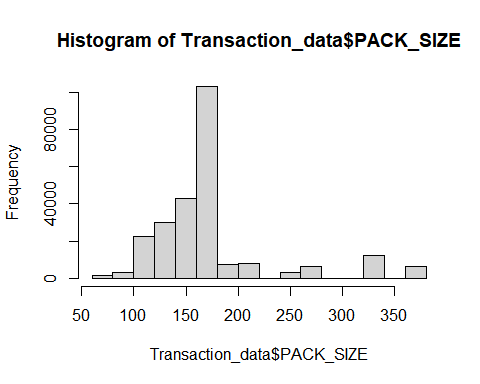
## [1] "numeric"

Transaction\_data = QVI\_transaction\_data %>% arrange(PACK\_SIZE)  
summary(Transaction\_data$PACK\_SIZE)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 70.0 150.0 170.0 175.6 175.0 380.0

#Plot a histogram showing the distribution of the PACk SIZE

hist(Transaction\_data$PACK\_SIZE)



#Creating brand using the first word from the product name column

Transaction\_data$BRAND <- str\_extract(Transaction\_data$PROD\_NAME,"(\\w+)")   
View(Transaction\_data)

#Get the frequency of each brand name

Brand\_Name = Transaction\_data %>% group\_by(BRAND) %>% count(BRAND)

#Sort the frequency of the brand name in descending.

Brand\_Name = Brand\_Name %>% arrange(n, desc(n))  
View(Brand\_Name)

#Clean Brand name to make it consistent

Transaction\_data$BRAND[Transaction\_data$BRAND == "Red"]<-"RRD"  
Transaction\_data$BRAND[Transaction\_data$BRAND == "Snbts"]<-"Sunbites"  
Transaction\_data$BRAND[Transaction\_data$BRAND == "Infzns"]<-"Infuzions"  
Transaction\_data$BRAND[Transaction\_data$BRAND == "Ww"]<-"Woolworths"  
Transaction\_data$BRAND[Transaction\_data$BRAND == "Smith"]<-"Smiths"  
Transaction\_data$BRAND[Transaction\_data$BRAND == "NATURAL"]<-"Natural"  
Transaction\_data$BRAND[Transaction\_data$BRAND == "Dorito"]<-"Doritos"  
Transaction\_data$BRAND[Transaction\_data$BRAND == "Grain"]<-"Grnwves"

#Get the frequency of the brand name after regularizing it.

Brand\_Name\_C = Transaction\_data %>% group\_by(BRAND) %>% count(BRAND)

#Sorting the brand name in descending order after making regularizing it

Brand\_Name\_C = Brand\_Name\_C %>% arrange(desc(n))  
View(Brand\_Name\_C)

#Examine the customer dataset

str(QVI\_purchase\_behaviour)

## spec\_tbl\_df [72,637 x 3] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ LYLTY\_CARD\_NBR : num [1:72637] 1000 1002 1003 1004 1005 ...  
## $ LIFESTAGE : chr [1:72637] "YOUNG SINGLES/COUPLES" "YOUNG SINGLES/COUPLES" "YOUNG FAMILIES" "OLDER SINGLES/COUPLES" ...  
## $ PREMIUM\_CUSTOMER: chr [1:72637] "Premium" "Mainstream" "Budget" "Mainstream" ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. LYLTY\_CARD\_NBR = col\_double(),  
## .. LIFESTAGE = col\_character(),  
## .. PREMIUM\_CUSTOMER = col\_character()  
## .. )

summary(QVI\_purchase\_behaviour)

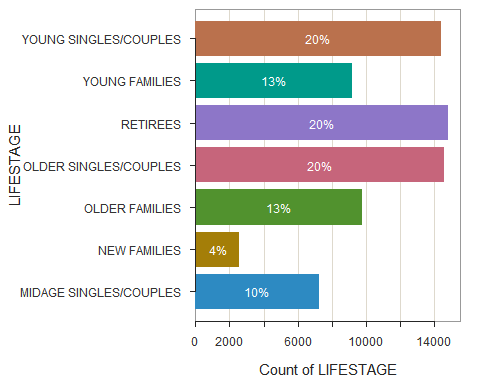
## LYLTY\_CARD\_NBR LIFESTAGE PREMIUM\_CUSTOMER   
## Min. : 1000 Length:72637 Length:72637   
## 1st Qu.: 66202 Class :character Class :character   
## Median : 134040 Mode :character Mode :character   
## Mean : 136186   
## 3rd Qu.: 203375   
## Max. :2373711

#Group the Lifestage column by the frequency of occurrence and plot a bar chart

QVI\_purchase\_behaviour %>% group\_by(LIFESTAGE) %>% count(LIFESTAGE) %>% arrange(desc(n))

## # A tibble: 7 x 2  
## # Groups: LIFESTAGE [7]  
## LIFESTAGE n  
## <chr> <int>  
## 1 RETIREES 14805  
## 2 OLDER SINGLES/COUPLES 14609  
## 3 YOUNG SINGLES/COUPLES 14441  
## 4 OLDER FAMILIES 9780  
## 5 YOUNG FAMILIES 9178  
## 6 MIDAGE SINGLES/COUPLES 7275  
## 7 NEW FAMILIES 2549

BarChart(LIFESTAGE, data=QVI\_purchase\_behaviour, horiz=TRUE)



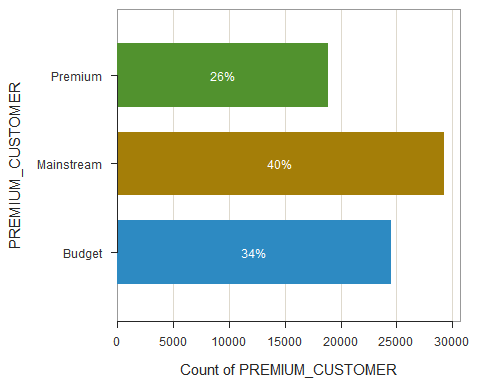
## >>> Suggestions  
## BarChart(LIFESTAGE, horiz=TRUE) # horizontal bar chart  
## BarChart(LIFESTAGE, fill="greens") # sequential green bars  
## PieChart(LIFESTAGE) # doughnut (ring) chart  
## Plot(LIFESTAGE) # bubble plot  
## Plot(LIFESTAGE, stat="count") # lollipop plot   
##   
##   
## --- LIFESTAGE ---  
##   
##   
## Missing Values of LIFESTAGE: 0   
##   
##   
## LIFESTAGE Count Prop   
## --------------------------------------   
## MIDAGE SINGLES/COUPLES 7275 0.100   
## NEW FAMILIES 2549 0.035   
## OLDER FAMILIES 9780 0.135   
## OLDER SINGLES/COUPLES 14609 0.201   
## RETIREES 14805 0.204   
## YOUNG FAMILIES 9178 0.126   
## YOUNG SINGLES/COUPLES 14441 0.199   
## --------------------------------------   
## Total 72637 1.000   
##   
##   
## Chi-squared test of null hypothesis of equal probabilities   
## Chisq = 12212.642, df = 6, p-value = 0.000

#Group the premium customer by the frequency of occurrence and plot a bar chart

QVI\_purchase\_behaviour %>% group\_by(PREMIUM\_CUSTOMER) %>% count(PREMIUM\_CUSTOMER) %>% arrange(desc(n))

## # A tibble: 3 x 2  
## # Groups: PREMIUM\_CUSTOMER [3]  
## PREMIUM\_CUSTOMER n  
## <chr> <int>  
## 1 Mainstream 29245  
## 2 Budget 24470  
## 3 Premium 18922

BarChart(PREMIUM\_CUSTOMER, data=QVI\_purchase\_behaviour, horiz=TRUE)



## >>> Suggestions  
## BarChart(PREMIUM\_CUSTOMER, horiz=TRUE) # horizontal bar chart  
## BarChart(PREMIUM\_CUSTOMER, fill="greens") # sequential green bars  
## PieChart(PREMIUM\_CUSTOMER) # doughnut (ring) chart  
## Plot(PREMIUM\_CUSTOMER) # bubble plot  
## Plot(PREMIUM\_CUSTOMER, stat="count") # lollipop plot   
##   
##   
## --- PREMIUM\_CUSTOMER ---  
##   
##   
## Missing Values of PREMIUM\_CUSTOMER: 0   
##   
##   
## Budget Mainstream Premium Total   
## Frequencies: 24470 29245 18922 72637   
## Proportions: 0.337 0.403 0.261 1.000   
##   
##   
## Chi-squared test of null hypothesis of equal probabilities   
## Chisq = 2204.734, df = 2, p-value = 0.000

#Merge transaction data to purchase behavior data

Data\_M <- merge(Transaction\_data, QVI\_purchase\_behaviour, by = "LYLTY\_CARD\_NBR")  
View(Data\_M)

#Examine the merged dataset

str(Data\_M)

## 'data.frame': 246740 obs. of 12 variables:  
## $ LYLTY\_CARD\_NBR : num 1000 1002 1003 1003 1004 ...  
## $ DATE : Date, format: "2018-10-17" "2018-09-16" ...  
## $ STORE\_NBR : num 1 1 1 1 1 1 1 1 1 1 ...  
## $ TXN\_ID : num 1 2 3 4 5 6 8 7 9 10 ...  
## $ PROD\_NBR : num 5 58 52 106 96 86 10 49 20 51 ...  
## $ PROD\_NAME : chr "Natural Chip Compny SeaSalt175g" "Red Rock Deli Chikn&Garlic Aioli 150g" "Grain Waves Sour Cream&Chives 210G" "Natural ChipCo Hony Soy Chckn175g" ...  
## $ PROD\_QTY : num 2 1 1 1 1 1 1 1 1 2 ...  
## $ TOT\_SALES : num 6 2.7 3.6 3 1.9 2.8 2.7 3.8 5.7 8.8 ...  
## $ PACK\_SIZE : num 175 150 210 175 160 165 150 110 330 170 ...  
## $ BRAND : chr "Natural" "RRD" "Grnwves" "Natural" ...  
## $ LIFESTAGE : chr "YOUNG SINGLES/COUPLES" "YOUNG SINGLES/COUPLES" "YOUNG FAMILIES" "YOUNG FAMILIES" ...  
## $ PREMIUM\_CUSTOMER: chr "Premium" "Mainstream" "Budget" "Budget" ...

summary(Data\_M)

## LYLTY\_CARD\_NBR DATE STORE\_NBR TXN\_ID   
## Min. : 1000 Min. :2018-07-01 Min. : 1.0 Min. : 1   
## 1st Qu.: 70015 1st Qu.:2018-09-30 1st Qu.: 70.0 1st Qu.: 67569   
## Median : 130367 Median :2018-12-30 Median :130.0 Median : 135182   
## Mean : 135530 Mean :2018-12-30 Mean :135.1 Mean : 135130   
## 3rd Qu.: 203083 3rd Qu.:2019-03-31 3rd Qu.:203.0 3rd Qu.: 202652   
## Max. :2373711 Max. :2019-06-30 Max. :272.0 Max. :2415841   
## PROD\_NBR PROD\_NAME PROD\_QTY TOT\_SALES   
## Min. : 1.00 Length:246740 Min. :1.000 Min. : 1.700   
## 1st Qu.: 26.00 Class :character 1st Qu.:2.000 1st Qu.: 5.800   
## Median : 53.00 Mode :character Median :2.000 Median : 7.400   
## Mean : 56.35 Mean :1.906 Mean : 7.316   
## 3rd Qu.: 87.00 3rd Qu.:2.000 3rd Qu.: 8.800   
## Max. :114.00 Max. :5.000 Max. :29.500   
## PACK\_SIZE BRAND LIFESTAGE PREMIUM\_CUSTOMER   
## Min. : 70.0 Length:246740 Length:246740 Length:246740   
## 1st Qu.:150.0 Class :character Class :character Class :character   
## Median :170.0 Mode :character Mode :character Mode :character   
## Mean :175.6   
## 3rd Qu.:175.0   
## Max. :380.0

#Total sales by LIFESTAGE and PREMIUM\_CUSTOMER

Total\_Sales = Data\_M %>% group\_by(LIFESTAGE, PREMIUM\_CUSTOMER) %>% summarise(TOTAL\_SALES = sum(TOT\_SALES))

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

View(Total\_Sales)

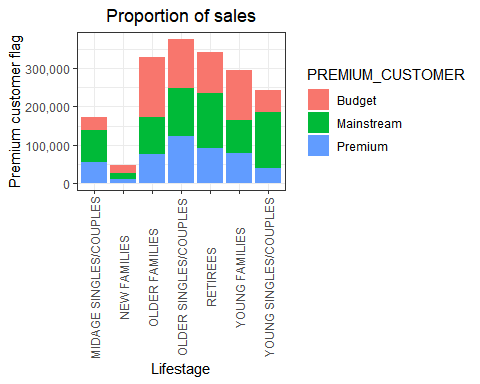
#Convert the premium customer and lifestage columns from character to a factor

Total\_Sales$LIFESTAGE <- as.factor(Total\_Sales$LIFESTAGE)  
Total\_Sales$PREMIUM\_CUSTOMER <- as.factor(Total\_Sales$PREMIUM\_CUSTOMER)  
is.factor(Total\_Sales$PREMIUM\_CUSTOMER)

## [1] TRUE

#Visualize total sale with life stages and premium customer

Total\_Sales %>% ggplot(aes(x = LIFESTAGE, y = TOTAL\_SALES, fill = PREMIUM\_CUSTOMER)) +  
 geom\_col() +   
 scale\_y\_continuous(labels = comma) +   
 labs(x = "Lifestage",  
 y = "Premium customer flag",   
 title = "Proportion of sales") + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5))

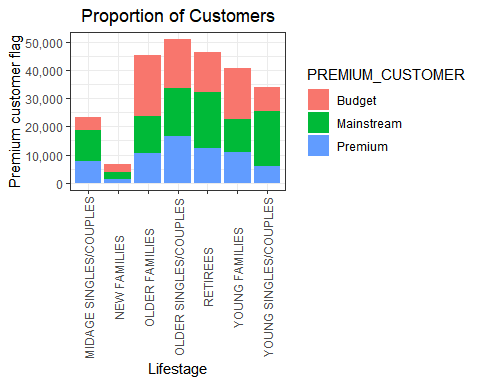


#Group premium customer and lifestage by the frequency of premium customer

No\_of\_Premium\_c = Data\_M %>% group\_by(PREMIUM\_CUSTOMER, LIFESTAGE) %>% count(PREMIUM\_CUSTOMER)   
View(No\_of\_Premium\_c)

#Plot a bar chart showing the premium customers and the number of customers

No\_of\_Premium\_c %>% ggplot(aes(x = LIFESTAGE, y = n, fill = PREMIUM\_CUSTOMER)) +  
 geom\_col() +   
 scale\_y\_continuous(labels = comma) +   
 labs(x = "Lifestage",  
 y = "Premium customer flag",   
 title = "Proportion of Customers") + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5))



#Average number of units per customer by LIFESTAGE and PREMIUM\_CUSTOMER

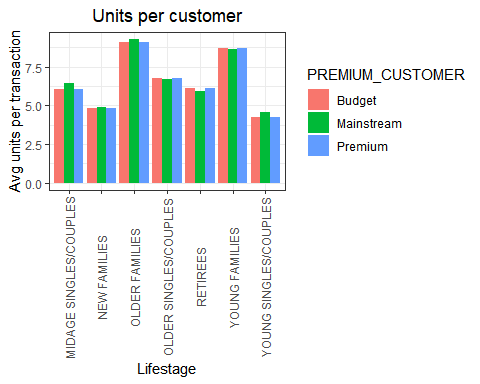
Avg\_Units <- Data\_M %>% group\_by(LIFESTAGE, PREMIUM\_CUSTOMER) %>% summarise(AVG = sum(PROD\_QTY)/uniqueN(LYLTY\_CARD\_NBR)) %>% arrange(desc(AVG))

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

View(Avg\_Units)

#Plot a bar chart for Avg units per customer by lifestage and premium customer

ggplot(data = Avg\_Units, aes(weight = AVG, x = LIFESTAGE, fill = PREMIUM\_CUSTOMER)) +  
 geom\_bar(position = position\_dodge()) +  
 labs(x = "Lifestage", y = "Avg units per transaction", title = "Units per customer") +  
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5))



#Exame average price by per unit by LIFESTAGE and PREMIUM\_CUSTOMER

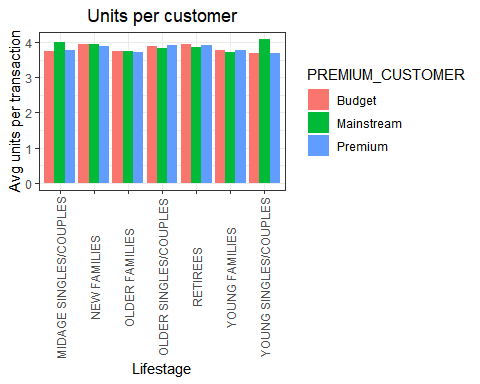
AVG\_PRICE <- Data\_M %>% group\_by(LIFESTAGE, PREMIUM\_CUSTOMER) %>% summarise(Avg = sum(TOT\_SALES)/sum(PROD\_QTY)) %>% arrange(desc(Avg))

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

View(AVG\_PRICE)

#Plot a chart showing the average price by Lifestage and premium customer

ggplot(data = AVG\_PRICE, aes(weight = Avg, x = LIFESTAGE, fill = PREMIUM\_CUSTOMER)) +  
 geom\_bar(position = position\_dodge()) +  
 labs(x = "Lifestage", y = "Avg units per transaction", title = "Units per customer") +  
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5))



Findings

2018-12-25 is missing in the DATE column and it’s assumed that the store was closed on Christmas day.

Sales increased in the days before Christmas day.

Sales are mainly coming from older families (budget), retirees (mainstream) and young singles/couples (mainstream).

Mainstream retirees and mainstream young singles/couples contribute more to the sales and it isn’t the case with older families budget.

Each of the premium segment in older families and young families buy more chips.

Mainstream mid-age single/couples and mainstream young singles/couples spend more per chips compared to other premium customers in their category.