Quantium Virtual Internship - Retail Strategy and Analytics _ Task 1

Manon Hernandez Angles

2025-01-09

Setup CRAN Mirror

R Markdown

```
##
## The downloaded binary packages are in
##  /var/folders/gz/t14s5qm50634f62z31q2t7x80000gn/T//RtmpiKR7tL/downloaded_packages

##
## The downloaded binary packages are in
##  /var/folders/gz/t14s5qm50634f62z31q2t7x80000gn/T//RtmpiKR7tL/downloaded_packages

##
## The downloaded binary packages are in
##  /var/folders/gz/t14s5qm50634f62z31q2t7x80000gn/T//RtmpiKR7tL/downloaded_packages

##
## The downloaded binary packages are in
##  /var/folders/gz/t14s5qm50634f62z31q2t7x80000gn/T//RtmpiKR7tL/downloaded_packages
```

```r
#Load files
transactionData <- read_excel("~/Downloads/QVI_transaction_data.xlsx")
customerData <- read_csv("~/Downloads/QVI_purchase_behaviour.csv")
```

```
## Rows: 72637 Columns: 3
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): LIFESTAGE, PREMIUM_CUSTOMER
## dbl (1): LYLTY_CARD_NBR
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

##Exploratory data analysis

###Examining transaction data

```r
head(transactionData)
```

```
## # A tibble: 6 x 8
##    DATE STORE_NBR LYLTY_CARD_NBR TXN_ID PROD_NBR PROD_NAME    PROD_QTY TOT_SALES
##   <dbl>     <dbl>          <dbl>  <dbl>    <dbl> <chr>           <dbl>     <dbl>
## 1 43390         1           1000      1        5 Natural Chi~        2       6
## 2 43599         1           1307    348       66 CCs Nacho C~        3       6.3
## 3 43605         1           1343    383       61 Smiths Crin~        2       2.9
```

```
## 4 43329         2            2373    974       69 Smiths Chip~        5       15
## 5 43330         2            2426   1038      108 Kettle Tort~        3       13.8
## 6 43604         4            4074   2982       57 Old El Paso~        1        5.1
```

```r
head(customerData)
```

```
## # A tibble: 6 x 3
##   LYLTY_CARD_NBR LIFESTAGE              PREMIUM_CUSTOMER
##            <dbl> <chr>                  <chr>
## 1           1000 YOUNG SINGLES/COUPLES  Premium
## 2           1002 YOUNG SINGLES/COUPLES  Mainstream
## 3           1003 YOUNG FAMILIES         Budget
## 4           1004 OLDER SINGLES/COUPLES  Mainstream
## 5           1005 MIDAGE SINGLES/COUPLES Mainstream
## 6           1007 YOUNG SINGLES/COUPLES  Budget
```

```r
str(transactionData)
```

```
## 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
##  $ 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 ...
```

```r
str(customerData)
```

```
## spc_tbl_ [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"
##  $ PREMIUM_CUSTOMER: chr [1:72637] "Premium" "Mainstream" "Budget" "Mainstream" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   LYLTY_CARD_NBR = col_double(),
##   ..   LIFESTAGE = col_character(),
##   ..   PREMIUM_CUSTOMER = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```r
names(transactionData)
```

```
## [1] "DATE"           "STORE_NBR"      "LYLTY_CARD_NBR" "TXN_ID"
## [5] "PROD_NBR"       "PROD_NAME"      "PROD_QTY"       "TOT_SALES"
```

We can see that the date column is in an integer format. Let's change this to a date format.

```r
##Convert DATE column to a date format
transactionData$DATE <- as.Date(transactionData$DATE, origin = "1899-12-30")
## Examine PROD_NAME
summary(transactionData$PROD_NAME)
```

```
##    Length     Class      Mode
##    264836 character character
```

```r
table(transactionData$PROD_NAME)
```

```
## 
##                         Burger Rings 220g 
##                                     1564 
##                  CCs Nacho Cheese    175g 
##                                     1498 
##                         CCs Original 175g 
##                                     1514 
##                  CCs Tasty Cheese    175g 
##                                     1539 
##            Cheetos Chs & Bacon Balls 190g 
##                                     1479 
##                        Cheetos Puffs 165g 
##                                     1448 
##                      Cheezels Cheese 330g 
##                                     3149 
##                  Cheezels Cheese Box 125g 
##                                     1454 
##            Cobs Popd Sea Salt  Chips 110g 
##                                     3265 
##    Cobs Popd Sour Crm  &Chives Chips 110g 
##                                     3159 
## Cobs Popd Swt/Chlli &Sr/Cream Chips 110g 
##                                     3269 
##          Dorito Corn Chp     Supreme 380g 
##                                     3185 
##          Doritos Cheese      Supreme 330g 
##                                     3052 
##   Doritos Corn Chip Mexican Jalapeno 150g 
##                                     3204 
##   Doritos Corn Chip Southern Chicken 150g 
##                                     3172 
##   Doritos Corn Chips  Cheese Supreme 170g 
##                                     3217 
##     Doritos Corn Chips  Nacho Cheese 170g 
##                                     3160 
##        Doritos Corn Chips  Original 170g 
##                                     3121 
##                 Doritos Mexicana    170g 
##                                     3115 
##           Doritos Salsa       Medium 300g 
##                                     1449 
##                 Doritos Salsa Mild  300g 
##                                     1472 
##            French Fries Potato Chips 175g 
##                                     1418 
##     Grain Waves         Sweet Chilli 210g 
##                                     3167 
##     Grain Waves Sour    Cream&Chives 210G 
##                                     3105 
##     GrnWves Plus Btroot & Chilli Jam 180g 
##                                     1468 
##   Infuzions BBQ Rib   Prawn Crackers 110g 
##                                     3174 
##   Infuzions Mango     Chutny Papadums 70g
```

```
##                                     1507
## Infuzions SourCream&Herbs Veg Strws 110g
##                                     3134
## Infuzions Thai SweetChili PotatoMix 110g
##                                     3242
##   Infzns Crn Crnchers Tangy Gcamole 110g
##                                     3144
##              Kettle 135g Swt Pot Sea Salt
##                                     3257
##                        Kettle Chilli 175g
##                                     3038
##         Kettle Honey Soy    Chicken 175g
##                                     3148
##   Kettle Mozzarella   Basil & Pesto 175g
##                                     3304
##                      Kettle Original 175g
##                                     3159
##     Kettle Sea Salt     And Vinegar 175g
##                                     3173
##       Kettle Sensations   BBQ&Maple 150g
##                                     3083
## Kettle Sensations   Camembert & Fig 150g
##                                     3219
##    Kettle Sensations   Siracha Lime 150g
##                                     3127
##  Kettle Sweet Chilli And Sour Cream 175g
##                                     3200
##  Kettle Tortilla ChpsBtroot&Ricotta 150g
##                                     3146
##     Kettle Tortilla ChpsFeta&Garlic 150g
##                                     3138
## Kettle Tortilla ChpsHny&Jlpno Chili 150g
##                                     3296
##   Natural Chip        Compny SeaSalt175g
##                                     1468
##  Natural Chip Co     Tmato Hrb&Spce 175g
##                                     1572
##   Natural ChipCo      Hony Soy Chckn175g
##                                     1460
##   Natural ChipCo Sea  Salt & Vinegr 175g
##                                     1550
##   NCC Sour Cream &    Garden Chives 175g
##                                     1419
## Old El Paso Salsa   Dip Chnky Tom Ht300g
##                                     3125
##  Old El Paso Salsa   Dip Tomato Med 300g
##                                     3114
## Old El Paso Salsa   Dip Tomato Mild 300g
##                                     3085
##                 Pringles Barbeque   134g
##                                     3210
##      Pringles Chicken    Salt Crips 134g
##                                     3104
##         Pringles Mystery    Flavour 134g
```

```
##                                   3114
##          Pringles Original   Crisps 134g
##                                   3157
##                 Pringles Slt Vingar 134g
##                                   3095
##           Pringles SourCream  Onion 134g
##                                   3162
##         Pringles Sthrn FriedChicken 134g
##                                   3083
##             Pringles Sweet&Spcy BBQ 134g
##                                   3177
##    Red Rock Deli Chikn&Garlic Aioli 150g
##                                   1434
##  Red Rock Deli Sp    Salt & Truffle 150G
##                                   1498
## Red Rock Deli SR    Salsa & Mzzrlla 150g
##                                   1458
##     Red Rock Deli Thai  Chilli&Lime 150g
##                                   1495
##         RRD Chilli&         Coconut 150g
##                                   1506
##         RRD Honey Soy       Chicken 165g
##                                   1513
##                 RRD Lime & Pepper   165g
##                                   1473
##                 RRD Pc Sea Salt     165g
##                                   1431
##                 RRD Salt & Vinegar  165g
##                                   1474
##      RRD SR Slow Rst     Pork Belly 150g
##                                   1526
##     RRD Steak &         Chimuchurri 150g
##                                   1455
##      RRD Sweet Chilli &  Sour Cream 165g
##                                   1516
##       Smith Crinkle Cut   Bolognese 150g
##                                   1451
##    Smith Crinkle Cut   Mac N Cheese 150g
##                                   1512
##    Smiths Chip Thinly  Cut Original 175g
##                                   1614
##   Smiths Chip Thinly  CutSalt/Vinegr175g
##                                   1440
##   Smiths Chip Thinly  S/Cream&Onion 175g
##                                   1473
##        Smiths Crinkle      Original 330g
##                                   3142
## Smiths Crinkle Chips Salt & Vinegar 330g
##                                   3197
##  Smiths Crinkle Cut  Chips Barbecue 170g
##                                   1489
##   Smiths Crinkle Cut  Chips Chicken 170g
##                                   1484
##  Smiths Crinkle Cut  Chips Chs&Onion170g
```

```
##                                    1481
##   Smiths Crinkle Cut  Chips Original 170g
##                                    1461
##  Smiths Crinkle Cut  French OnionDip 150g
##                                    1438
##   Smiths Crinkle Cut  Salt & Vinegar 170g
##                                    1455
##       Smiths Crinkle Cut  Snag&Sauce 150g
##                                    1503
##     Smiths Crinkle Cut  Tomato Salsa 150g
##                                    1470
##    Smiths Crnkle Chip  Orgnl Big Bag 380g
##                                    3233
##  Smiths Thinly       Swt Chli&S/Cream175G
##                                    1461
##    Smiths Thinly Cut   Roast Chicken 175g
##                                    1519
##      Snbts Whlgrn Crisps Cheddr&Mstrd 90g
##                                    1576
##  Sunbites Whlegrn    Crisps Frch/Onin 90g
##                                    1432
##    Thins Chips         Originl saltd 175g
##                                    1441
##            Thins Chips Light&  Tangy 175g
##                                    3188
##          Thins Chips Salt &  Vinegar 175g
##                                    3103
##          Thins Chips Seasonedchicken 175g
##                                    3114
##      Thins Potato Chips  Hot & Spicy 175g
##                                    3229
##           Tostitos Lightly    Salted 175g
##                                    3074
##         Tostitos Smoked     Chipotle 175g
##                                    3145
##             Tostitos Splash Of  Lime 175g
##                                    3252
##                  Twisties Cheese     270g
##                                    3115
##           Twisties Cheese     Burger 250g
##                                    3169
##                      Twisties Chicken270g
##                                    3170
##    Tyrrells Crisps     Ched & Chives 165g
##                                    3268
##   Tyrrells Crisps     Lightly Salted 165g
##                                    3174
##            Woolworths Cheese   Rings 190g
##                                    1516
##            Woolworths Medium   Salsa 300g
##                                    1430
##            Woolworths Mild     Salsa 300g
##                                    1491
##          WW Crinkle Cut      Chicken 175g
```

```
##                                    1467
##         WW Crinkle Cut      Original 175g
##                                    1410
##         WW D/Style Chip     Sea Salt 200g
##                                    1469
##            WW Original Corn    Chips 200g
##                                    1495
##            WW Original Stacked Chips 160g
##                                    1487
##    WW Sour Cream &OnionStacked Chips 160g
##                                    1483
##       WW Supreme Cheese   Corn Chips 200g
##                                    1509
```

Looks like we are definitely looking at potato chips but how can we check that these are all chips? We can do some basic text analysis by summarising the individual words in the product name.

```
###Examine the words in PROD_NAME
productWords <- data.table(unlist(strsplit(unique(transactionData[['PROD_NAME']]), " ")))
setnames(productWords, 'words')
####Remove digits
productWords$words <- gsub("[0-9]", "", productWords$words)
####Remove the special characters
productWords$words <- gsub("[&]", "", productWords$words)
productWords$words <- gsub("[[:punct:]]", "", productWords$words)
####Sort by frequency
word_freq <- table(productWords$words)
sorted_word_freq <- sort(word_freq, decreasing = TRUE)
sorted_word_freq_df <- data.frame(word = names(sorted_word_freq), frequency = as.vector(sorted_word_fre
###Remove the SALSA
transactionData <- as.data.table(transactionData)
transactionData[, SALSA := grepl("salsa", tolower(PROD_NAME))]
transactionData <- transactionData[SALSA == FALSE, ][, SALSA := NULL]
```

Next, we can use summary() to check summary statistics such as mean, min and max values for each feature to see if there are any obvious outliers in the data and if there are any nulls in any of the columns (NA's : number of nulls will appear in the output if there are any nulls).

```
#Find outliers and null values
summary(transactionData)
```

```
##       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
```

There are no nulls in the columns but product quantity appears to have an outlier which we should investigate further. Let's investigate further the case where 200 packets of chips are bought in one transaction.

```
transactionData[PROD_QTY == 200]
```

```
##          DATE STORE_NBR LYLTY_CARD_NBR TXN_ID PROD_NBR
##        <Date>     <num>          <num>  <num>    <num>
## 1: 2018-08-19       226         226000 226201        4
## 2: 2019-05-20       226         226000 226210        4
##                           PROD_NAME PROD_QTY TOT_SALES
##                              <char>    <num>     <num>
## 1: Dorito Corn Chp     Supreme 380g      200       650
## 2: Dorito Corn Chp     Supreme 380g      200       650
```

```
transactionData[LYLTY_CARD_NBR == 226000]
```

```
##          DATE STORE_NBR LYLTY_CARD_NBR TXN_ID PROD_NBR
##        <Date>     <num>          <num>  <num>    <num>
## 1: 2018-08-19       226         226000 226201        4
## 2: 2019-05-20       226         226000 226210        4
##                           PROD_NAME PROD_QTY TOT_SALES
##                              <char>    <num>     <num>
## 1: Dorito Corn Chp     Supreme 380g      200       650
## 2: Dorito Corn Chp     Supreme 380g      200       650
```

It looks like this customer has only had the two transactions over the year and is not an ordinary retail customer. The customer might be buying chips for commercial purposes instead. We'll remove this loyalty card number from further analysis.

```
#Remove outliers
transactionData <- transactionData[LYLTY_CARD_NBR != 226000]
summary(transactionData)
```

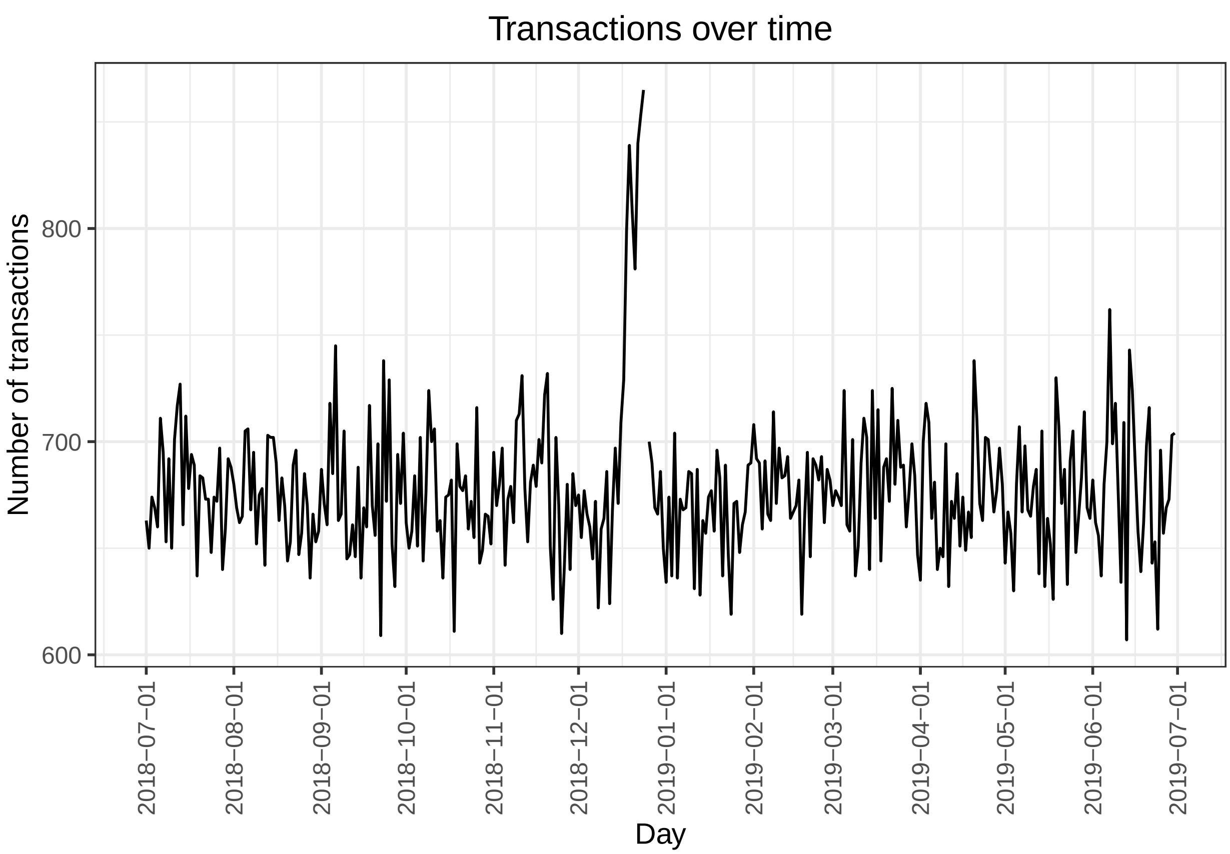
```
##       DATE              STORE_NBR     LYLTY_CARD_NBR        TXN_ID
##  Min.   :2018-07-01   Min.   :  1.0   Min.   :   1000   Min.   :      1
##  1st Qu.:2018-09-30   1st Qu.: 70.0   1st Qu.:  70015   1st Qu.:  67569
##  Median :2018-12-30   Median :130.0   Median : 130367   Median : 135182
##  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
```

That's better. Now, let's look at the number of transaction lines over time to see if there are any obvious data issues such as missing data.

```
#Count the number of transaction by date
transactionCountByDate <- transactionData[, .N, by = DATE]
```

There's only 364 rows, meaning only 364 dates which indicates a missing date. Let's create a sequence of dates from 1 Jul 2018 to 30 Jun 2019 and use this to create a chart of number of transactions over time to find the missing date.

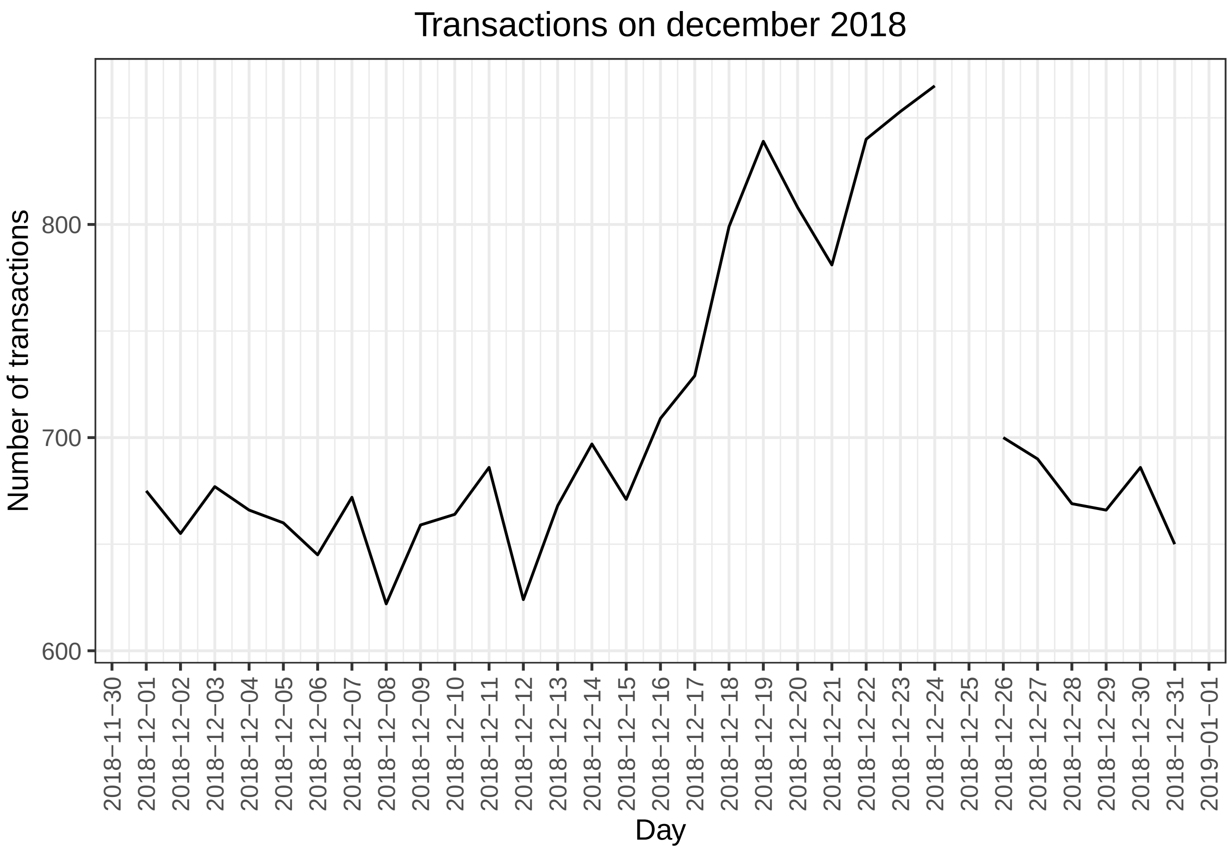
```r
#Find the missing date
dateSequence <- data.table(DATE = seq(as.Date("2018-07-01"), as.Date("2019-06-30"), by = "day"))
transactionCountByDate <- merge(dateSequence, transactionCountByDate, by = "DATE", all.x = TRUE)
transactionData <- merge(dateSequence, transactionData, by = "DATE", all.x = TRUE)
#Setting plot themes to format graphs
theme_set(theme_bw())
theme_update(plot.title = element_text(hjust = 0.5))
#Plot transactions over times
ggplot(transactionCountByDate, aes(x = DATE, y = N)) +
  geom_line() +
  labs(x = "Day", y = "Number of transactions", title = "Transactions over time") +
  scale_x_date(breaks = "1 month") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```



We can see that there is an increase in purchases in December and a break in late December. Let's zoom in on this.

```r
#Plot transaction in december
ggplot(transactionCountByDate, aes(x = DATE, y = N)) +
  geom_line() +
  labs(x = "Day", y = "Number of transactions", title = "Transactions on december 2018") +
  scale_x_date(breaks = "1 day", limits = as.Date(c("2018-12-01", "2018-12-31"))) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```

```
## Warning: Removed 334 rows containing missing values or values outside the scale range
## (`geom_line()`).
```



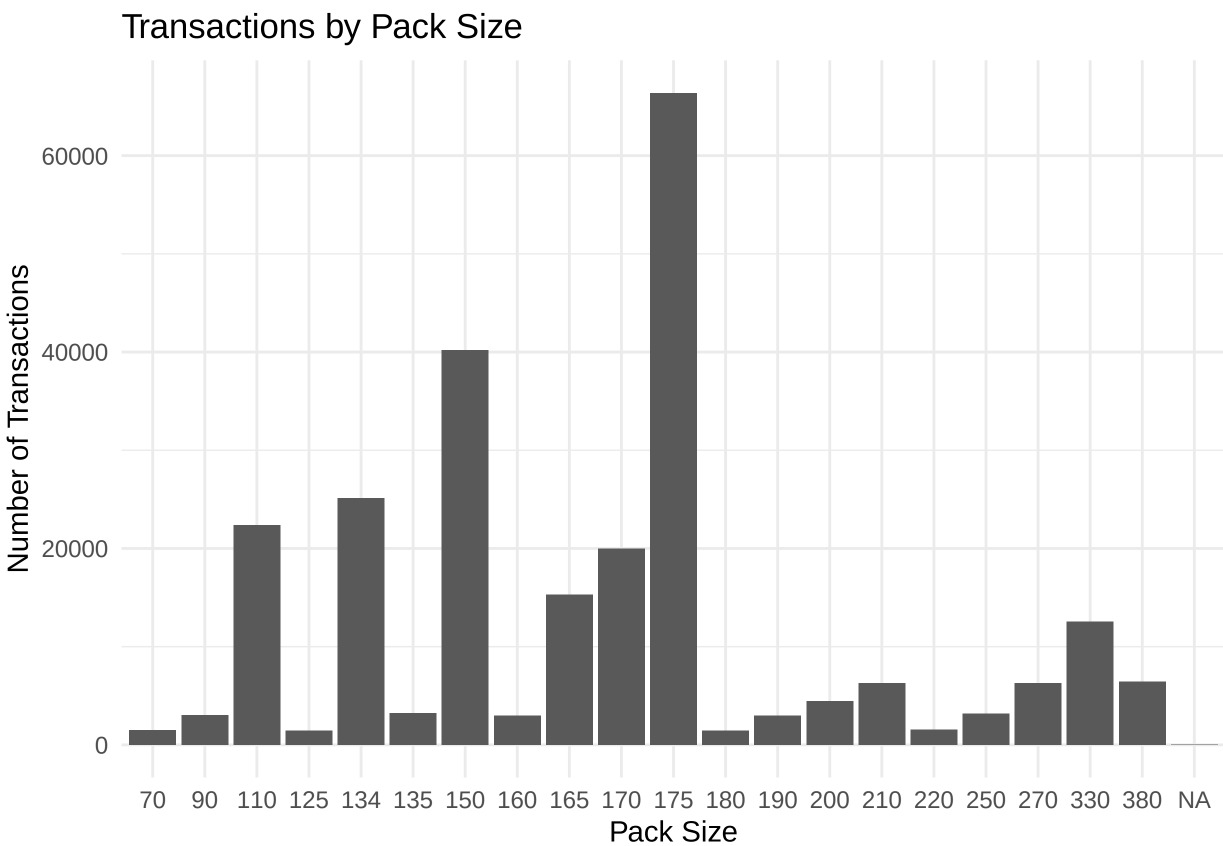
We can see that the increase in sales occurs in the lead-up to Christmas and that there are zero sales on Christmas day itself. This is due to shops being closed on Christmas day. Now that we are satisfied that the data no longer has outliers, we can move on to creating other features such as brand of chips or pack size from PROD_NAME. We will start with pack size.

```
#Check the Pack size
transactionData[, PACK_SIZE := parse_number(PROD_NAME)]
transactionData[, .N, PACK_SIZE][order(PACK_SIZE)]
```

```
##     PACK_SIZE     N
##         <num> <int>
##  1:        70  1507
##  2:        90  3008
##  3:       110 22387
##  4:       125  1454
##  5:       134 25102
##  6:       135  3257
##  7:       150 40203
##  8:       160  2970
##  9:       165 15297
## 10:       170 19983
## 11:       175 66390
## 12:       180  1468
## 13:       190  2995
## 14:       200  4473
## 15:       210  6272
## 16:       220  1564
## 17:       250  3169
```

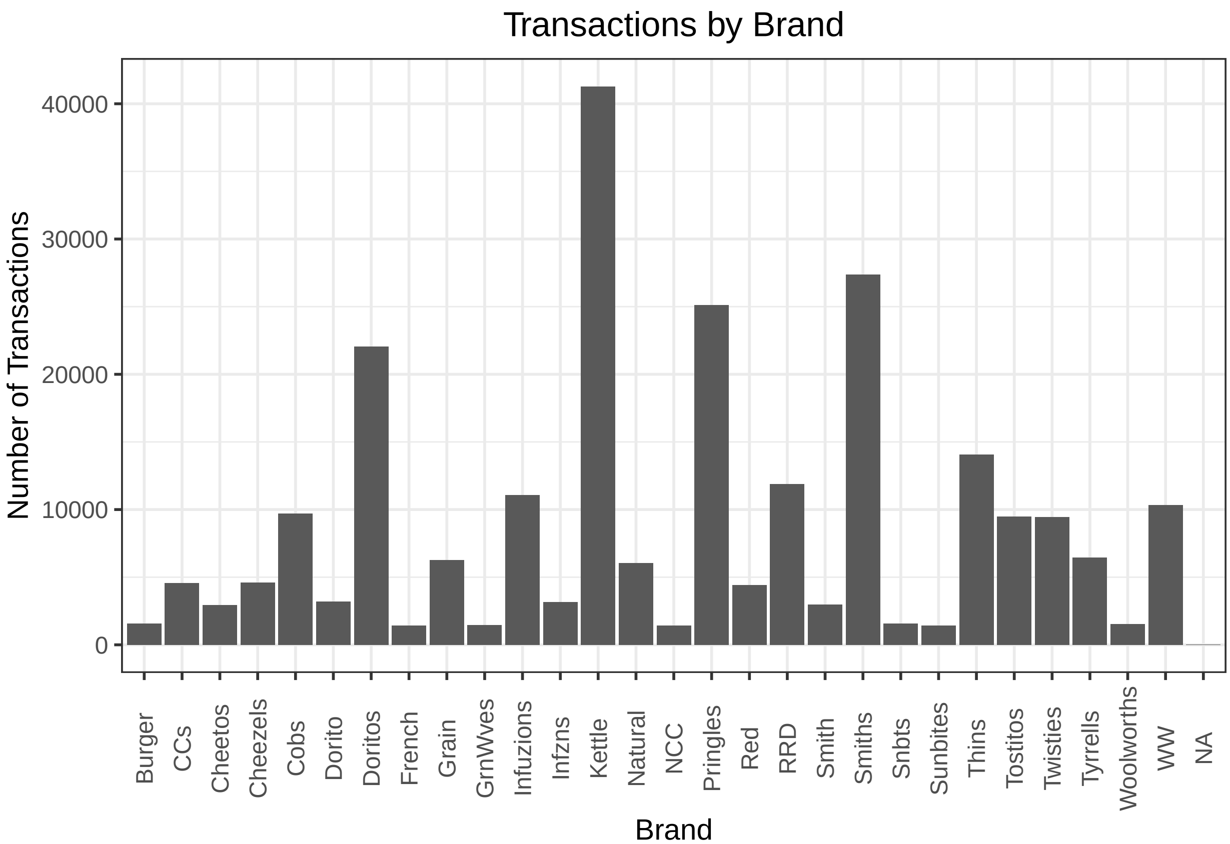
```
## 18:       270  6285
## 19:       330 12540
## 20:       380  6416
## 21:        NA     1
##     PACK_SIZE     N
```

```
ggplot(transactionData, aes(x = as.factor(PACK_SIZE))) +
  geom_bar() +
  labs(x = "Pack Size", y = "Number of Transactions", title = "Transactions by Pack Size") +
  theme_minimal()
```



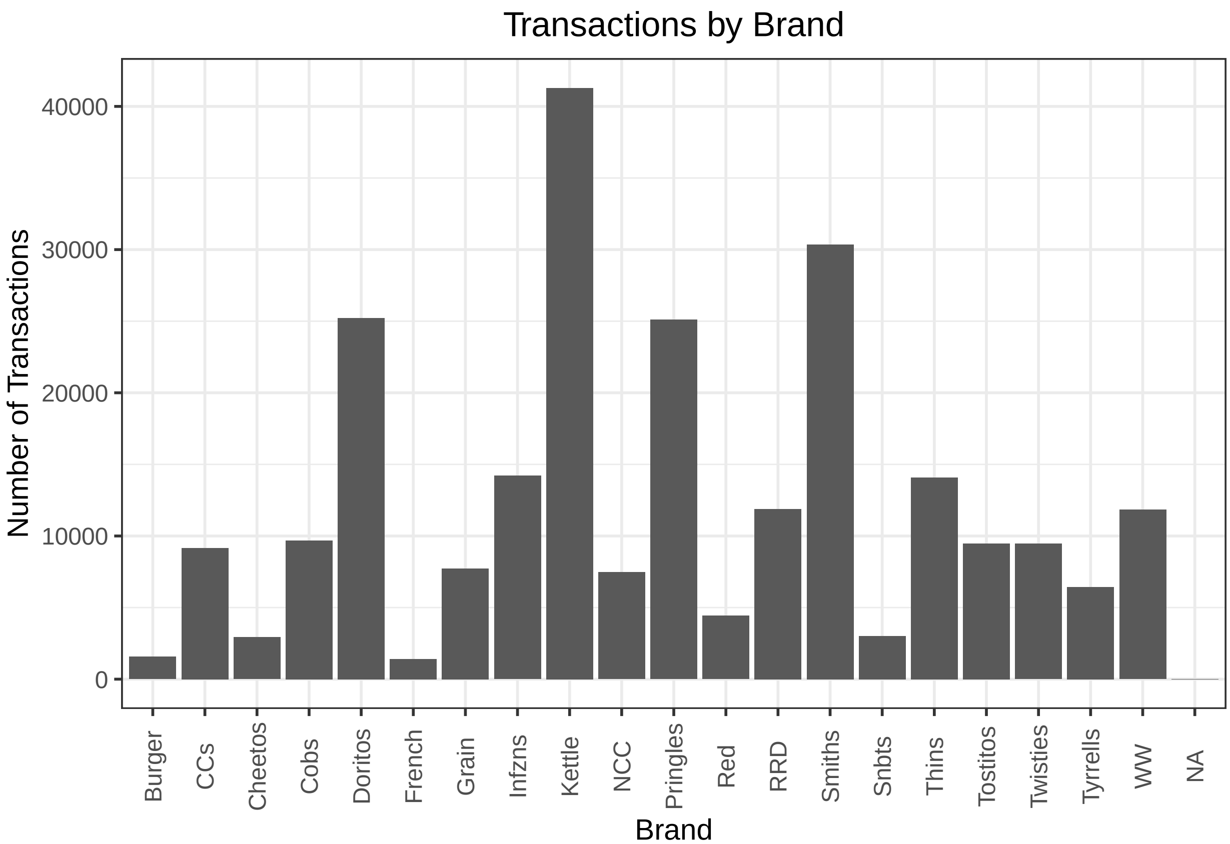
The largest size is 380g and the smallest size is 70g - seems sensible! Pack sizes created look reasonable and now to create brands, we can use the first word in PROD_NAME to work out the brand name.

```
#Check the Brand
transactionData[, BRAND := sub(" .*", "", PROD_NAME)]
ggplot(transactionData, aes(x = as.factor(BRAND))) +
  geom_bar() +
  labs(x = "Brand", y = "Number of Transactions", title = "Transactions by Brand") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```



Some of the brand names look like they are of the same brands - such as RED and RRD, which are both Red Rock Deli chips. Let's combine these together.

```
transactionData[BRAND == "RED", BRAND := "RRD"]
transactionData[BRAND == "Infuzions", BRAND := "Infzns"]
transactionData[BRAND == "Woolworths", BRAND := "WW"]
transactionData[BRAND == "Cheezels", BRAND := "CCs"]
transactionData[BRAND == "Dorito", BRAND := "Doritos"]
transactionData[BRAND == "GrnWves", BRAND := "Grain"]
transactionData[BRAND == "Sunbites", BRAND := "Snbts"]
transactionData[BRAND == "Smith", BRAND := "Smiths"]
transactionData[BRAND == "Natural", BRAND := "NCC"]
ggplot(transactionData, aes(x = as.factor(BRAND))) +
  geom_bar() +
  labs(x = "Brand", y = "Number of Transactions", title = "Transactions by Brand") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```

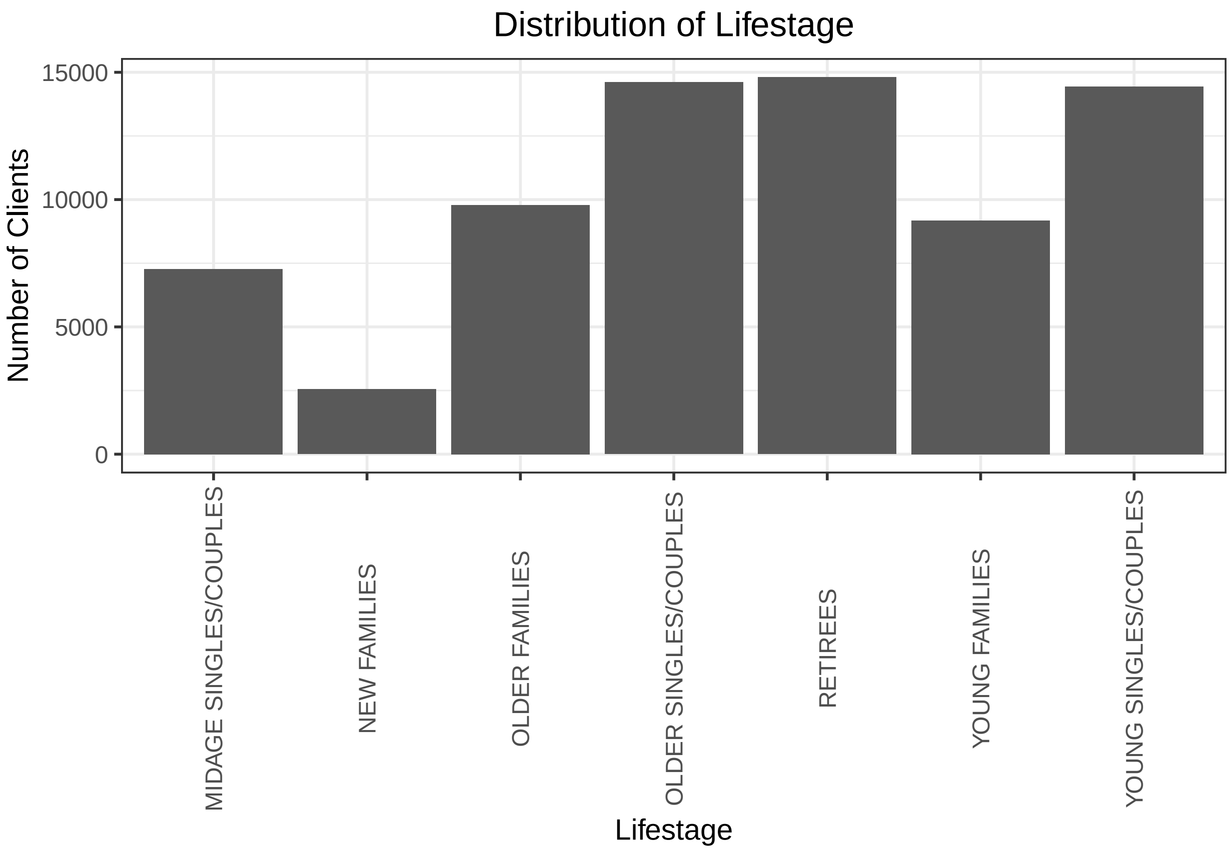


Examining customer data Now that we are happy with the transaction dataset, let's have a look at the customer dataset.

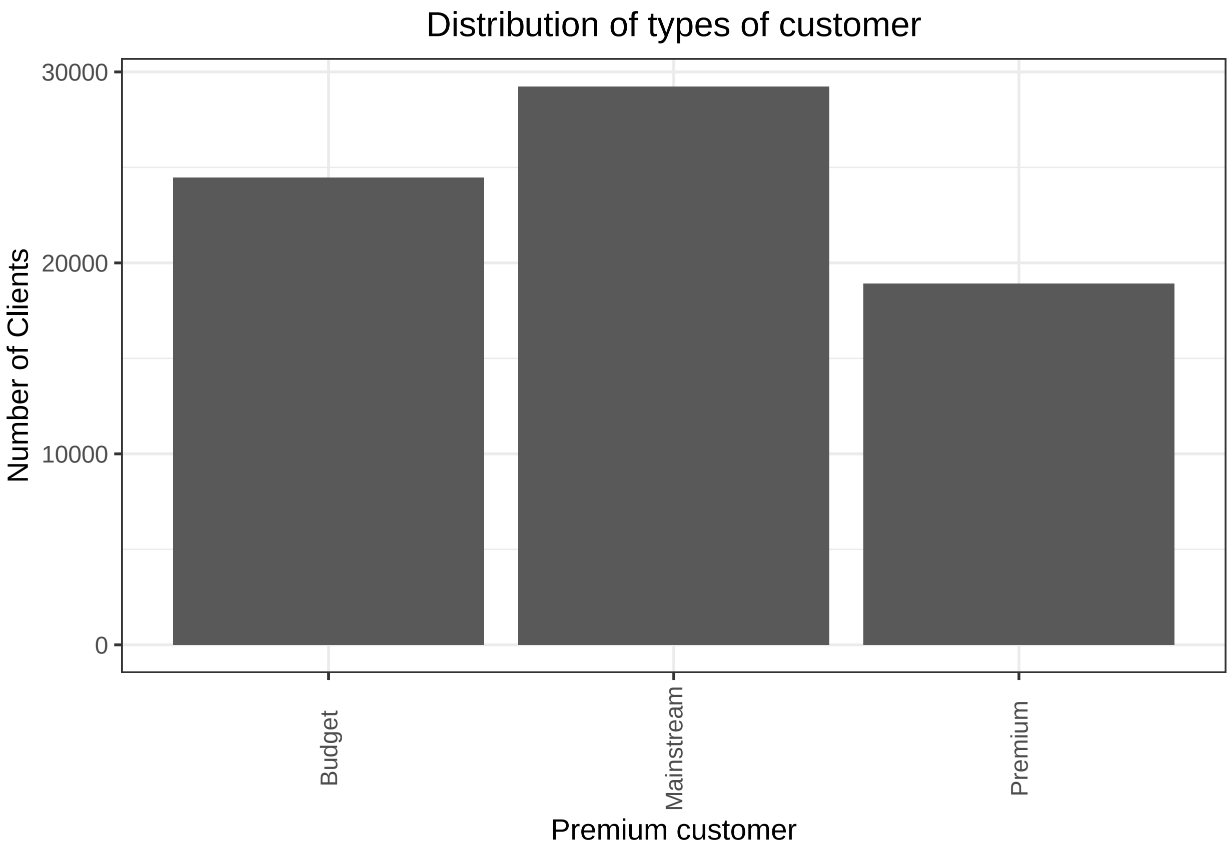
```
#Examine customer data
summary(customerData)
```

```
##  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
```

```
ggplot(customerData, aes(x = LIFESTAGE)) +
  geom_bar() +
  labs(x = "Lifestage", y = "Number of Clients", title = "Distribution of Lifestage") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```




```r
ggplot(customerData, aes(x = PREMIUM_CUSTOMER)) +
  geom_bar() +
  labs(x = "Premium customer", y = "Number of Clients", title = "Distribution of types of customer") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```



As there do not seem to be any issues with the customer data, we can now go ahead and join the transaction and customer data sets together

```
#Merge transaction data to customer data
data <- merge(transactionData, customerData, by = 'LYLTY_CARD_NBR', all.x = TRUE)
summary(data)
```

```
##  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
##  NA's   :1                              NA's   :1       NA's   :1
##     PROD_NBR       PROD_NAME            PROD_QTY        TOT_SALES
##  Min.   :  1.00   Length:246741      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
##  NA's   :1                           NA's   :1       NA's   :1
##    PACK_SIZE        BRAND             LIFESTAGE          PREMIUM_CUSTOMER
##  Min.   : 70.0   Length:246741      Length:246741      Length:246741
##  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
##  NA's   :1
```

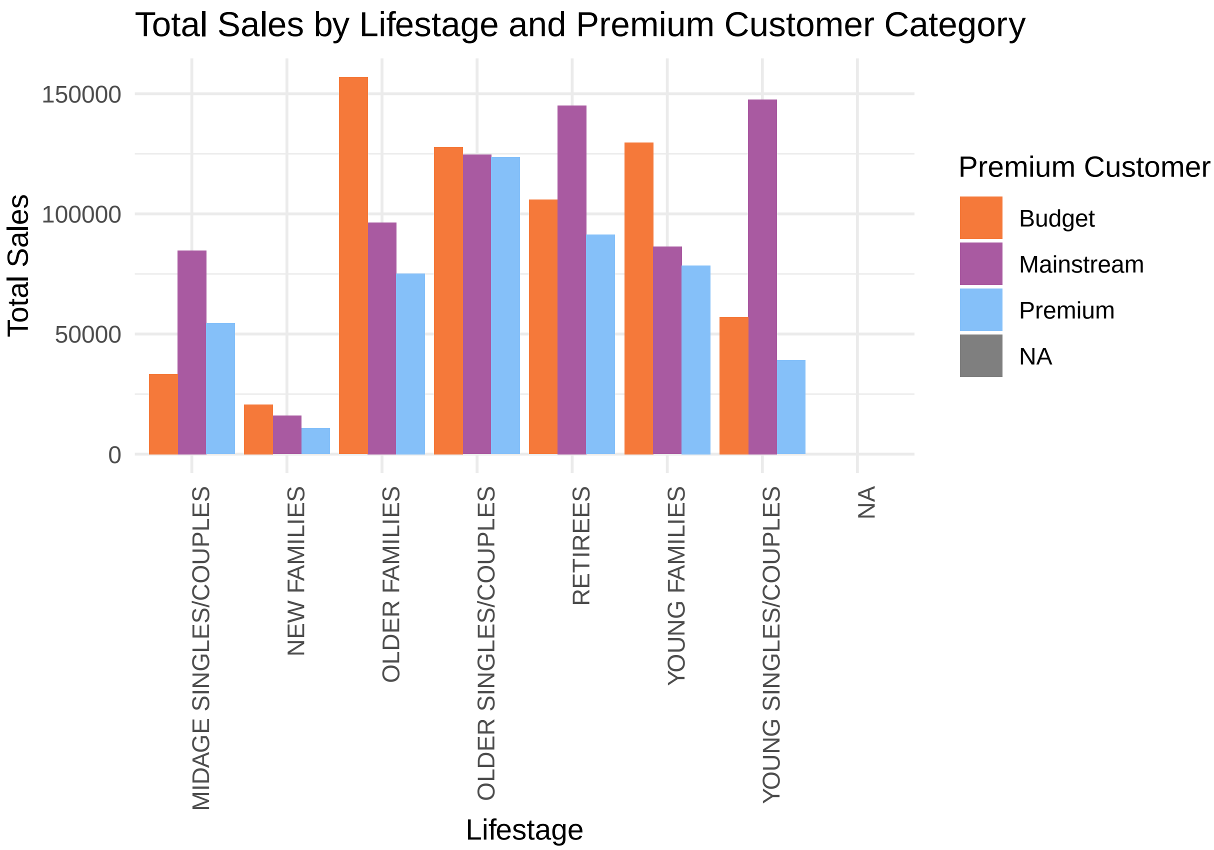
```
fwrite(data, paste0("~/Downloads/QVI_data.csv"))
```

Data analysis on customer segments

Now that the data is ready for analysis, we can define some metrics of interest to the client: * Who spends the most on chips (total sales), describing customers by lifestage and how premium their general purchasing behaviour is * How many customers are in each segment * How many chips are bought per customer by segment * What's the average chip price by customer segment We could also ask our data team for more information. Examples are: * The customer's total spend over the period and total spend for each transaction to understand what proportion of their grocery spend is on chips * Proportion of customers in each customer segment overall to compare against the mix of customers who purchase chips Let's start with calculating total sales by LIFESTAGE and PREMIUM_CUSTOMER and plotting the split by these segments to describe which customer segment contribute most to chip sales.

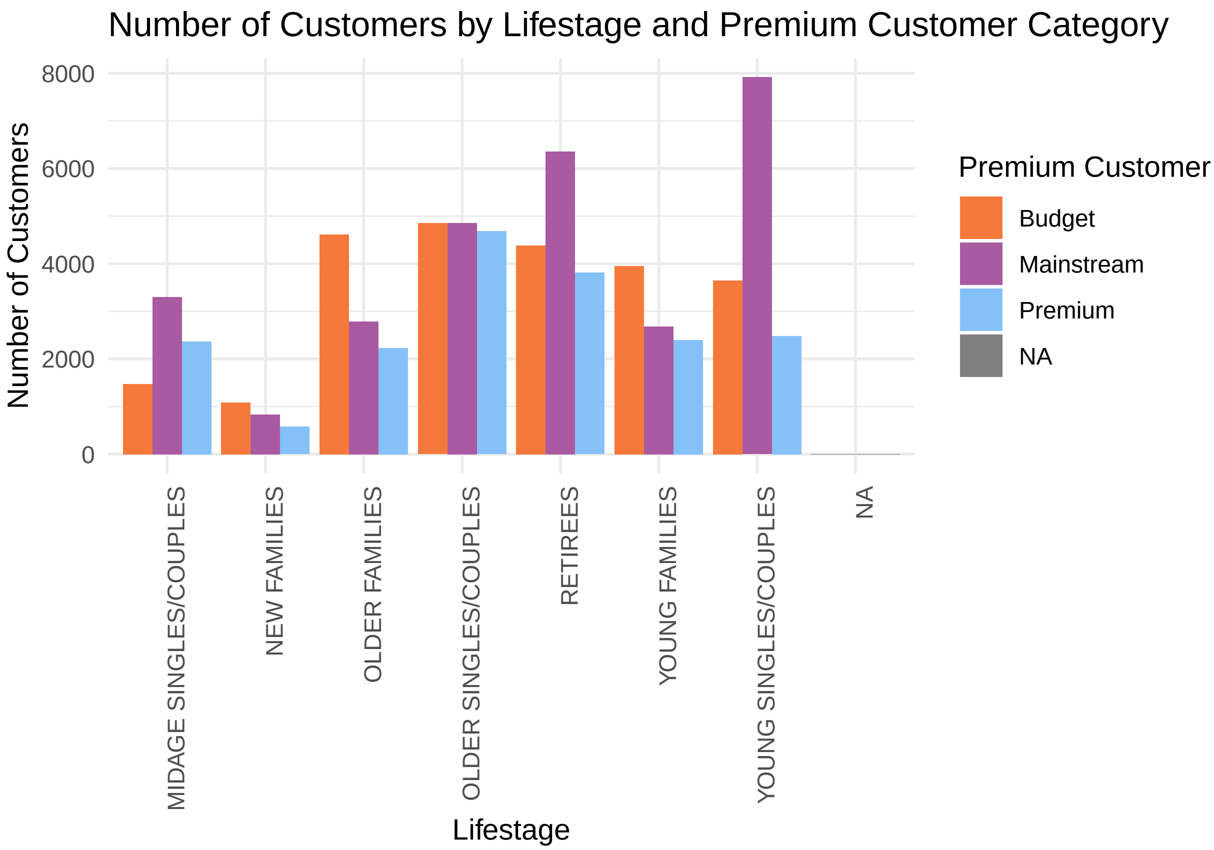
```
#Data Analysis
#Sales by lifestage and premium customer
salesSummary <- data[, .( total_sales = sum(TOT_SALES),
                                      average_sales = mean(TOT_SALES),
                                      min_sales = min(TOT_SALES),
                                      max_sales = max(TOT_SALES) ),
                                  by = .(LIFESTAGE, PREMIUM_CUSTOMER)]
ggplot(salesSummary, aes(x = LIFESTAGE, y = total_sales, fill = PREMIUM_CUSTOMER)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Lifestage", y = "Total Sales", title = "Total Sales by Lifestage and Premium Customer Categ
  theme_minimal() +
  scale_fill_manual(values = c("Premium" = "#85C0F9", "Mainstream" = "#A95AA1", "Budget" = "#F5793A"), n
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_bar()`).
```



Sales are coming mainly from Budget - older families, Mainstream - young singles/couples, and Mainstream - retirees Let's see if the higher sales are due to there being more customers who buy chips.

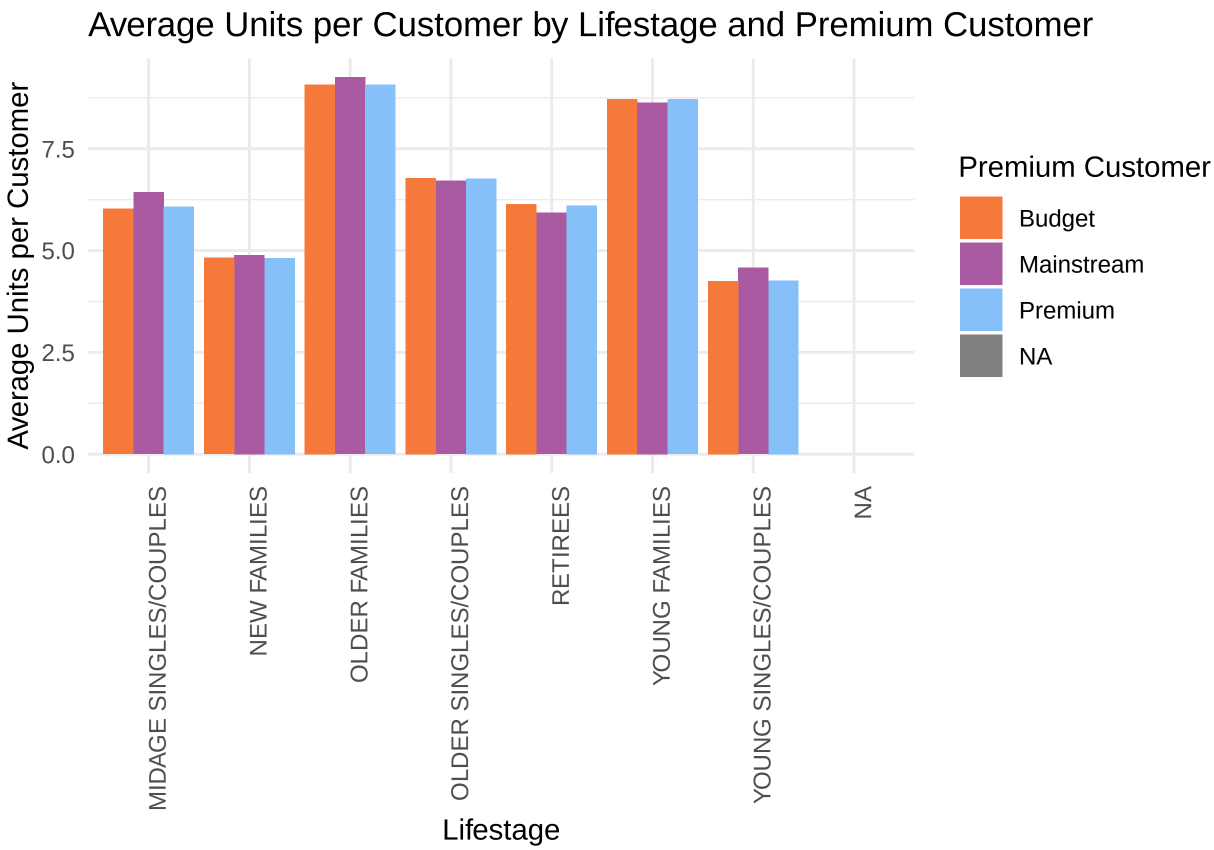
```
#Number of customer by lifestage and premium customer
uniqueCustomers <- unique(data, by = c("LYLTY_CARD_NBR", "LIFESTAGE", "PREMIUM_CUSTOMER"))
customerSummary <- uniqueCustomers[, .N, by = .(LIFESTAGE, PREMIUM_CUSTOMER)]
ggplot(customerSummary, aes(x = LIFESTAGE, y = N, fill = PREMIUM_CUSTOMER)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Lifestage", y = "Number of Customers", title = "Number of Customers by Lifestage and Premiu
  theme_minimal() +
  scale_fill_manual(values = c("Premium" = "#85C0F9", "Mainstream" = "#A95AA1", "Budget" = "#F5793A"), n
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



There are more Mainstream - young singles/couples and Mainstream - retirees who buy chips. This contributes to there being more sales to these customer segments but this is not a major driver for the Budget - Older families segment. Higher sales may also be driven by more units of chips being bought per customer. Let's have a look at this next.

```
#Average number of unit per customer by lifestage and premium customer
unitsSummary <- data[, .( total_units = sum(PROD_QTY),
                          unique_customers = uniqueN(LYLTY_CARD_NBR) ),
                     by = .(LIFESTAGE, PREMIUM_CUSTOMER)]
unitsSummary[, avg_units_per_customer := total_units / unique_customers]
ggplot(unitsSummary, aes(x = LIFESTAGE, y = avg_units_per_customer, fill = PREMIUM_CUSTOMER)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Lifestage", y = "Average Units per Customer", title = "Average Units per Customer by Lifesta
  theme_minimal() +
  scale_fill_manual(values = c("Premium" = "#85C0F9", "Mainstream" = "#A95AA1", "Budget" = "#F5793A"), n
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

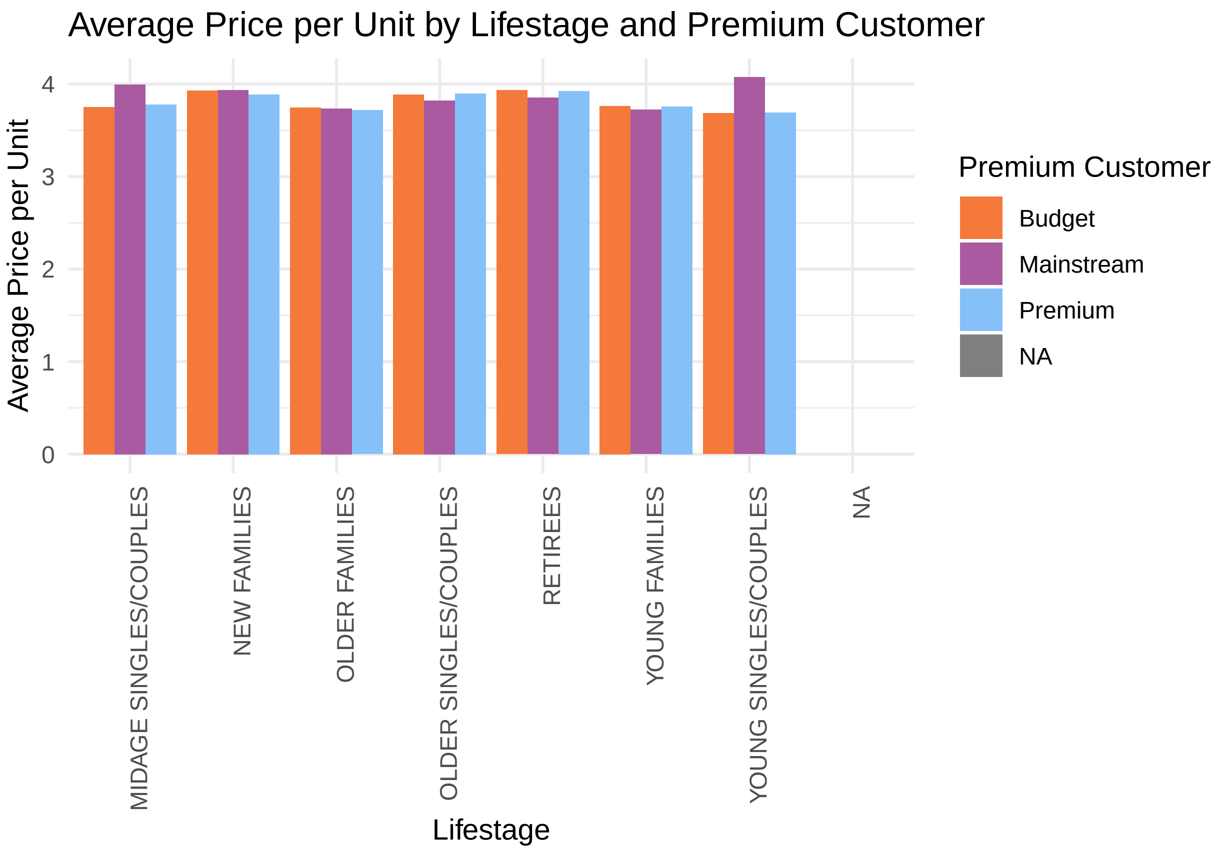
```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_bar()`).
```



Older families and young families in general buy more chips per customer Let's also investigate the average price per unit chips bought for each customer segment as this is also a driver of total sales.

```
#Average price per unit by lifestage and premium customer
priceSummary <- data[, .(
  total_sales = sum(TOT_SALES),
  total_units = sum(PROD_QTY) ),
  by = .(LIFESTAGE, PREMIUM_CUSTOMER)]
priceSummary[, avg_price_per_unit := total_sales / total_units]
ggplot(priceSummary, aes(x = LIFESTAGE, y = avg_price_per_unit, fill = PREMIUM_CUSTOMER)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Lifestage", y = "Average Price per Unit", title = "Average Price per Unit by Lifestage and I
  theme_minimal() +
  scale_fill_manual(values = c("Premium" = "#85C0F9", "Mainstream" = "#A95AA1", "Budget" = "#F5793A"), r
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_bar()`).
```



Mainstream midage and young singles and couples are more willing to pay more per packet of chips compared to their budget and premium counterparts. This may be due to premium shoppers being more likely to buy healthy snacks and when they buy chips, this is mainly for entertainment purposes rather than their own consumption. This is also supported by there being fewer premium midage and young singles and couples buying chips compared to their mainstream counterparts. As the difference in average price per unit isn't large, we can check if this difference is statistically different.

```
#T-test between mainstream vs premium and budget midage ans young singles and couples
mainstream_midage_young_sales <- data[
  PREMIUM_CUSTOMER == "Mainstream" & (
  LIFESTAGE == "MIDAGE SINGLES/COUPLES" | LIFESTAGE == "YOUNG SINGLES/COUPLES"),
  TOT_SALES]
budget_premium_midage_young_sales <- data[
  (PREMIUM_CUSTOMER == "Budget" | PREMIUM_CUSTOMER == "Premium") &
  (LIFESTAGE == "MIDAGE SINGLES/COUPLES" | LIFESTAGE == "YOUNG SINGLES/COUPLES"),
  TOT_SALES]
t_test <- t.test(
  mainstream_midage_young_sales, budget_premium_midage_young_sales)
print(t_test)
```

```
##
##  Welch Two Sample t-test
##
## data:  mainstream_midage_young_sales and budget_premium_midage_young_sales
## t = 33.067, df = 55260, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.6580945 0.7410243
```

```
## sample estimates:
## mean of x mean of y
##  7.582377  6.882818
```

The t-test results in a p-value < 2.2e-16, i.e. the unit price for mainstream, young and mid-age singles and couples are significantly higher than that of budget or premium, young and midage singles and couples.

Deep dive into specific customer segments for insights

We have found quite a few interesting insights that we can dive deeper into. We might want to target customer segments that contribute the most to sales to retain them or further increase sales. Let's look at Mainstream - young singles/couples. For instance, let's find out if they tend to buy a particular brand of chips.

```r
#Deep dive into Mainstream, young singles/couples
##Preferred brand
segment1 <- data[LIFESTAGE == "YOUNG SINGLES/COUPLES" & PREMIUM_CUSTOMER == "Mainstream"]
other <- data[!(LIFESTAGE == "YOUNG SINGLES/COUPLES" & PREMIUM_CUSTOMER == "Mainstream")]
quantity_segment1 <- segment1[, sum(PROD_QTY)]
quantity_other <- other[, sum(PROD_QTY)]
quantity_segment1_by_brand <- segment1[, .(targetSegment = sum(PROD_QTY) / quantity_segment1), by = BRA
quantity_other_by_brand <- other[, .(other = sum(PROD_QTY) / quantity_other), by = BRAND]
brand_proportions <- merge(quantity_segment1_by_brand, quantity_other_by_brand, by = "BRAND")[, affinit
brand_proportions_ordered <- brand_proportions[order(-affinityToBrand)]
print(brand_proportions_ordered)
```

```
##        BRAND targetSegment       other affinityToBrand
##       <char>         <num>       <num>           <num>
##  1: Tyrrells   0.031552795 0.025692464       1.2280953
##  2: Twisties   0.046183575 0.037876520       1.2193194
##  3:  Doritos   0.122760524 0.101074684       1.2145526
##  4:   Kettle   0.197984817 0.165553442       1.1958967
##  5: Tostitos   0.045410628 0.037977861       1.1957131
##  6: Pringles   0.119420290 0.100634769       1.1866703
##  7:     Cobs   0.044637681 0.039048861       1.1431238
##  8:   Infzns   0.064679089 0.057064679       1.1334347
##  9:    Thins   0.060372671 0.056986370       1.0594230
## 10:    Grain   0.032712215 0.031187957       1.0488733
## 11:      CCs   0.029151139 0.037542552       0.7764826
## 12:   Smiths   0.096369910 0.124583692       0.7735355
## 13:   French   0.003947550 0.005758060       0.6855694
## 14:  Cheetos   0.008033126 0.012066591       0.6657329
## 15:      RRD   0.032022084 0.049150801       0.6515069
## 16:      Red   0.011787440 0.018342876       0.6426168
## 17:      NCC   0.019599724 0.030853989       0.6352412
## 18:    Snbts   0.006349206 0.012580210       0.5046980
## 19:       WW   0.024099379 0.049427188       0.4875733
## 20:   Burger   0.002926156 0.006596434       0.4435967
##        BRAND targetSegment       other affinityToBrand
```

We can see that : * Mainstream young singles/couples are 23% more likely to purchase Tyrrells chips compared to the rest of the population * Mainstream young singles/couples are 56% less likely to purchase Burger Rings compared to the rest of the population Let's also find out if our target segment tends to buy larger packs of chips.

```r
##Preferred pack size
quantity_segment1_by_pack <- segment1[, .(targetSegment = sum(PROD_QTY)/quantity_segment1), by = PACK_S
```

```r
quantity_other_by_pack <- other[, .(other = sum(PROD_QTY)/quantity_other), by = PACK_SIZE]
pack_proportions <- merge(quantity_segment1_by_pack, quantity_other_by_pack)[, affinityToPack := targetS
pack_proportions[order(-affinityToPack)]
```

```
##     PACK_SIZE targetSegment       other affinityToPack
##         <num>         <num>       <num>          <num>
##  1:       270   0.031828847 0.025095929      1.2682873
##  2:       380   0.032160110 0.025584213      1.2570295
##  3:       330   0.061283644 0.050161917      1.2217166
##  4:       134   0.119420290 0.100634769      1.1866703
##  5:       110   0.106280193 0.089791190      1.1836372
##  6:       210   0.029123533 0.025121265      1.1593180
##  7:       135   0.014768806 0.013075403      1.1295106
##  8:       250   0.014354727 0.012780590      1.1231662
##  9:       170   0.080772947 0.080985964      0.9973697
## 10:       150   0.157598344 0.163420656      0.9643722
## 11:       175   0.254989648 0.270006956      0.9443818
## 12:       165   0.055652174 0.062267662      0.8937572
## 13:       190   0.007481021 0.012442016      0.6012708
## 14:       180   0.003588682 0.006066692      0.5915385
## 15:       160   0.006404417 0.012372920      0.5176157
## 16:        90   0.006349206 0.012580210      0.5046980
## 17:       125   0.003008972 0.006036750      0.4984423
## 18:       200   0.008971705 0.018656115      0.4808989
## 19:        70   0.003036577 0.006322350      0.4802924
## 20:       220   0.002926156 0.006596434      0.4435967
##     PACK_SIZE targetSegment       other affinityToPack
```

It looks like Mainstream young singles/couples are 27% more likely to purchase a 270g pack of chips compared to the rest of the population but let's dive into what brands sell this pack size.

```r
##Preferred pack size
data[PACK_SIZE== 270, unique(PROD_NAME)]
```

```
## [1] "Twisties Cheese     270g" "Twisties Chicken270g"
```

Twisties are the only brand offering 270g packs and so this may instead be reflecting a higher likelihood of purchasing Twisties.

Conclusion

Let's recap what we've found! Sales have mainly been due to Budget - older families, Mainstream - young singles/couples, and Mainstream - retirees shoppers. We found that the high spend in chips for mainstream young singles/couples and retirees is due to there being more of them than other buyers. Mainstream, midage and young singles and couples are also more likely to pay more per packet of chips. This is indicative of impulse buying behaviour. We've also found that Mainstream young singles and couples are 23% more likely to purchase Tyrrells chips compared to the rest of the population. The Category Manager may want to increase the category's performance by off-locating some Tyrrells and smaller packs of chips in discretionary space near segments where young singles and couples frequent more often to increase visibilty and impulse behaviour. Quantium can help the Category Manager with recommendations of where these segments are and further help them with measuring the impact of the changed placement. We'll work on measuring the impact of trials in the next task and putting all these together in the third task.