R Notebook for InstaCart Product Analysis

Assignment: Choose a problem domain and a question you are interested to answer. It could be in the public policy, sports, business, politics, art etc - any topic that you are interested to explore.Then apply the problem-solving framework to answer that question. • Frame the Problem: Identify the key question are you trying to answer. • Acquire the Data: Get the dataset to answer the question. • Refine the Data: Do the basic refinement to clean up the dataset. • Transform the Data: Do the transformation needed for the dataset. • Explore the Data: Create the 3 - 4 individual visualisation that explore the dataset. • Model the Data: Do the basic modelling (as needed) to answer the question • Communicate the insight: Create final visualisations to share the insight. Please ensure you create .rmd notebook to communicate your thought process as well as the code. Create a github account and repo for your code and data. Potential Data Sources • Data is Plural • Data.gov.in • Kaggle Datasets • Awesome Public Dataset Please submit the following - 1. A working R code in .Rmd file format. 2. The Question you framed and Output/Insights of what you have done.

Frame the Problem: Identify the key question are you trying to answer

InstaCart wants to find more insights about their store sales, best sellers and want to learn about the product portfolio. There are 6 csv data files downloaded from Kaggle competitions.

Questions to be answered: 1. What is the most prominent day/time for the sales? 2. How many items do people buy ? 3. What is the best seller of the entire product portfolio? 4. How often do people order the same items again ? 5. What are the most often reordered products? 6. Which item do people put into the cart first? 7. Is there an association between time of last order and probability of reorder ? 8. Is there an association between number of orders and probability of reordering ? 9. Visualize the product portfolio 10. How many unique products are offered in each department/aisle? 11. How often are products from the department/aisle sold?

Acquire the Data: Get the dataset to answer the question.

```
# install.packages("readr")
# install.packages("dplyr")
# install.packages("ggplot2")
# install.packages("knitr")
# install.packages("stringr")
# install.packages("DT")
# install.packages("data.table")


library(readr)
library(dplyr)
library(ggplot2)
library(knitr)
library(stringr)
library(DT)
library(data.table)

orders <- fread('Data/orders.csv', sep = ',')
```

```
##
Read 74.8% of 3421083 rows
Read 3421083 rows and 7 (of 7) columns from 0.101 GB file in 00:00:03
```

```
products <- fread('Data/products.csv')
order_products <- fread('Data/order_products__train.csv')
order_products_prior <- fread('Data/order_products__prior.csv')
```

```
##
Read 19.9% of 32434489 rows
Read 40.9% of 32434489 rows
Read 61.9% of 32434489 rows
Read 82.8% of 32434489 rows
Read 32434489 rows and 4 (of 4) columns from 0.538 GB file in 00:00:06
```

```
aisles <- fread('Data/aisles.csv')
departments <- fread('Data/departments.csv')
```

Lets first have a look at these files:

Refine the Data: Do the basic refinement to clean up the dataset.

orders

This file gives a list of all orders we have in the dataset. 1 row per order. For example, we can see that user 1 has 11 orders, 1 of which is in the train set, and 10 of which are prior orders. The orders.csv doesn’t tell us about which products were ordered. This is contained in the order_products.csv

```
kable(head(orders,12))
```

order_id	user_id	eval_set	order_number	order_dow	order_hour_of_day	days_since_prior_order
2539329	1	prior	1	2	8	NA
2398795	1	prior	2	3	7	15
473747	1	prior	3	3	12	21
2254736	1	prior	4	4	7	29
431534	1	prior	5	4	15	28
3367565	1	prior	6	2	7	19
550135	1	prior	7	1	9	20
3108588	1	prior	8	1	14	14
2295261	1	prior	9	1	16	0
2550362	1	prior	10	4	8	30
1187899	1	train	11	4	8	14
2168274	2	prior	1	2	11	NA

```
str(orders, max.level=1)
```

```
## Classes 'data.table' and 'data.frame':   3421083 obs. of  7 variables:
##  $ order_id              : int  2539329 2398795 473747 2254736 431534 3367565 550135 3108588
 2295261 2550362 ...
##  $ user_id               : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ eval_set              : chr  "prior" "prior" "prior" "prior" ...
##  $ order_number          : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ order_dow             : int  2 3 3 4 4 2 1 1 1 4 ...
##  $ order_hour_of_day     : int  8 7 12 7 15 7 9 14 16 8 ...
##  $ days_since_prior_order: num  NA 15 21 29 28 19 20 14 0 30 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

order_products_train

This file gives us information about which products (product_id) were ordered. It also contains information of the order (add_to_cart_order) in which the products were put into the cart and information of whether this product is a re-order(1) or not(0).

For example, we see below that order_id 1 had 8 products, 4 of which are reorders.

Still we don’t know what these products are. This information is in the products.csv

```
kable(head(order_products,10))
```

order_id	product_id	add_to_cart_order	reordered

1	49302	1	1
1	11109	2	1
1	10246	3	0
1	49683	4	0
1	43633	5	1
1	13176	6	0
1	47209	7	0
1	22035	8	1
36	39612	1	0
36	19660	2	1

```
str(order_products, max.level=1)
```

```
## Classes 'data.table' and 'data.frame':   1384617 obs. of  4 variables:
##  $ order_id         : int  1 1 1 1 1 1 1 1 36 36 ...
##  $ product_id       : int  49302 11109 10246 49683 43633 13176 47209 22035 39612 19660 ...
##  $ add_to_cart_order: int  1 2 3 4 5 6 7 8 1 2 ...
##  $ reordered        : int  1 1 0 0 1 0 0 1 0 1 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

products

This file contains the names of the products with their corresponding product_id. Furthermore the aisle and deparment are included.

```
kable(head(products,10))
```

product_id	product_name	aisle_id	department_id
1	Chocolate Sandwich Cookies	61	19
2	All-Seasons Salt	104	13
3	Robust Golden Unsweetened Oolong Tea	94	7
4	Smart Ones Classic Favorites Mini Rigatoni With Vodka Cream Sauce	38	1
5	Green Chile Anytime Sauce	5	13
6	Dry Nose Oil	11	11
7	Pure Coconut Water With Orange	98	7
8	Cut Russet Potatoes Steam N’ Mash	116	1
9	Light Strawberry Blueberry Yogurt	120	16

product_id	product_name	aisle_id	department_id
10	Sparkling Orange Juice & Prickly Pear Beverage	115	7

```
str(products, max.level=1)
```

```
## Classes 'data.table' and 'data.frame':   49688 obs. of  4 variables:
##  $ product_id   : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ product_name : chr  "Chocolate Sandwich Cookies" "All-Seasons Salt" "Robust Golden Unsweet
ened Oolong Tea" "Smart Ones Classic Favorites Mini Rigatoni With Vodka Cream Sauce" ...
##  $ aisle_id     : int  61 104 94 38 5 11 98 116 120 115 ...
##  $ department_id: int  19 13 7 1 13 11 7 1 16 7 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

order_products_prior

This file is structurally the same as the other_products_train.csv.

```
kable(head(order_products_prior,10))
```

order_id	product_id	add_to_cart_order	reordered
2	33120	1	1
2	28985	2	1
2	9327	3	0
2	45918	4	1
2	30035	5	0
2	17794	6	1
2	40141	7	1
2	1819	8	1
2	43668	9	0
3	33754	1	1

```
str(order_products_prior, max.level=1)
```

```
## Classes 'data.table' and 'data.frame':   32434489 obs. of  4 variables:
##  $ order_id         : int  2 2 2 2 2 2 2 2 2 3 ...
##  $ product_id       : int  33120 28985 9327 45918 30035 17794 40141 1819 43668 33754 ...
##  $ add_to_cart_order: int  1 2 3 4 5 6 7 8 9 1 ...
##  $ reordered        : int  1 1 0 1 0 1 1 1 0 1 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

aisles

This file contains the different aisles.

```
kable(head(aisles,10))
```

aisle_id	aisle
1	prepared soups salads
2	specialty cheeses
3	energy granola bars
4	instant foods
5	marinades meat preparation
6	other
7	packaged meat
8	bakery desserts
9	pasta sauce
10	kitchen supplies

```
str(aisles, max.level=1)
```

```
## Classes 'data.table' and 'data.frame':   134 obs. of  2 variables:
##  $ aisle_id: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ aisle   : chr  "prepared soups salads" "specialty cheeses" "energy granola bars" "instant
 foods" ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

departments

```
kable(head(departments,10))
```

department_id	department
1	frozen
2	other
3	bakery
4	produce
5	alcohol
6	international
7	beverages
8	pets

department_id	department
9	dry goods pasta
10	bulk

```
str(departments, max.level=1)
```

```
## Classes 'data.table' and 'data.frame':   21 obs. of  2 variables:
##  $ department_id: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ department   : chr  "frozen" "other" "bakery" "produce" ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

Transform the Data: Do the transformation needed for the dataset. Record Variables

We should do some recoding and convert character variables to factors.

```
head(orders)
```

```
##    order_id user_id eval_set order_number order_dow order_hour_of_day
## 1:  2539329       1    prior            1         2                 8
## 2:  2398795       1    prior            2         3                 7
## 3:   473747       1    prior            3         3                12
## 4:  2254736       1    prior            4         4                 7
## 5:   431534       1    prior            5         4                15
## 6:  3367565       1    prior            6         2                 7
##    days_since_prior_order
## 1:                     NA
## 2:                     15
## 3:                     21
## 4:                     29
## 5:                     28
## 6:                     19
```

```
orders <- orders %>% mutate(order_hour_of_day = as.numeric(order_hour_of_day), eval_set = as.fac
tor(eval_set))
products <- products %>% mutate(product_name = as.factor(product_name))
aisles <- aisles %>% mutate(aisle = as.factor(aisle))
departments <- departments %>% mutate(department = as.factor(department))
```

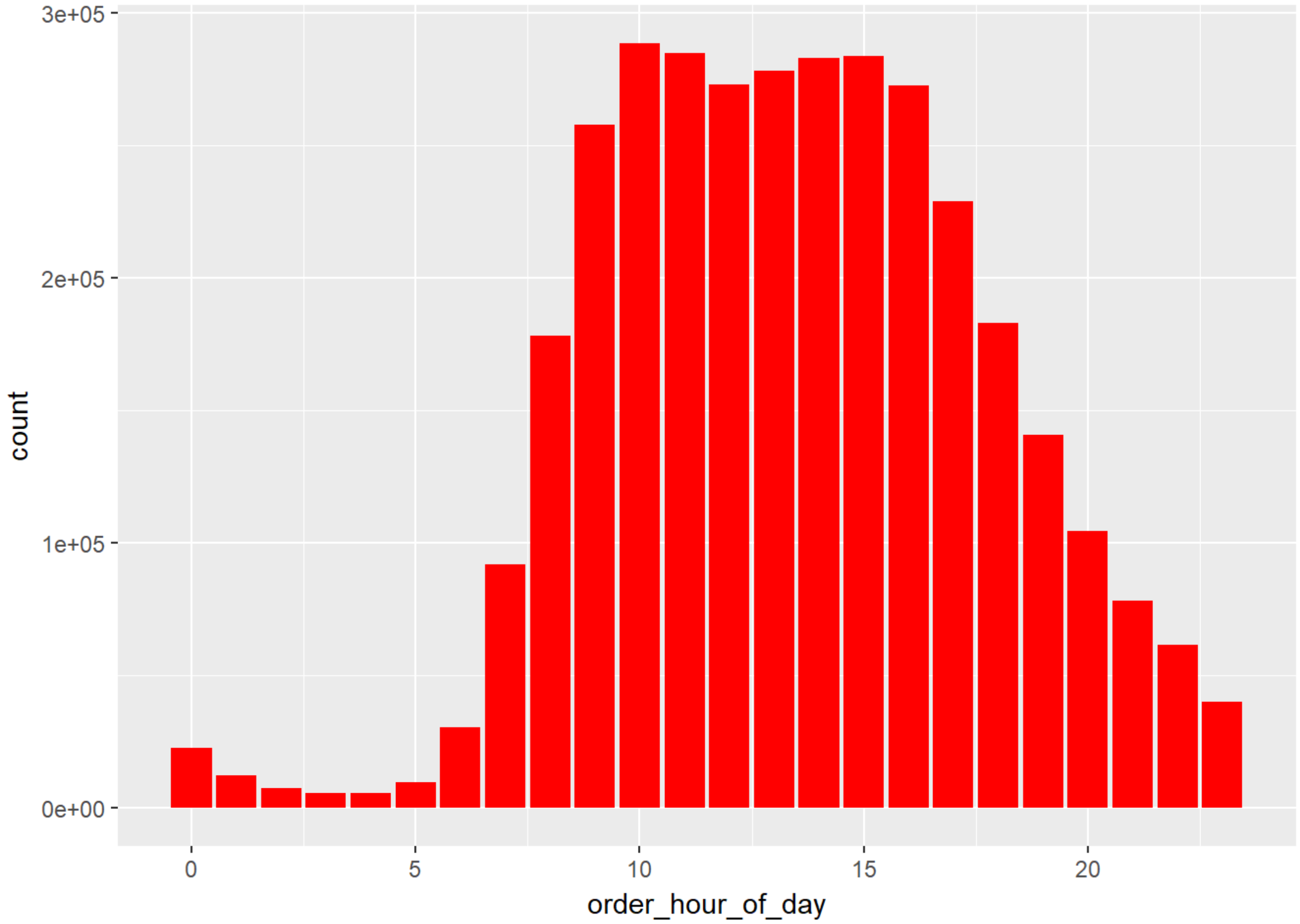
Explore the Data, Model the data, and Communicate the insights

When do people order? Let’s have a look when people buy groceries online.

Hour of Day

There is a clear effect of hour of day on order volume. Most orders are between 8.00-18.00

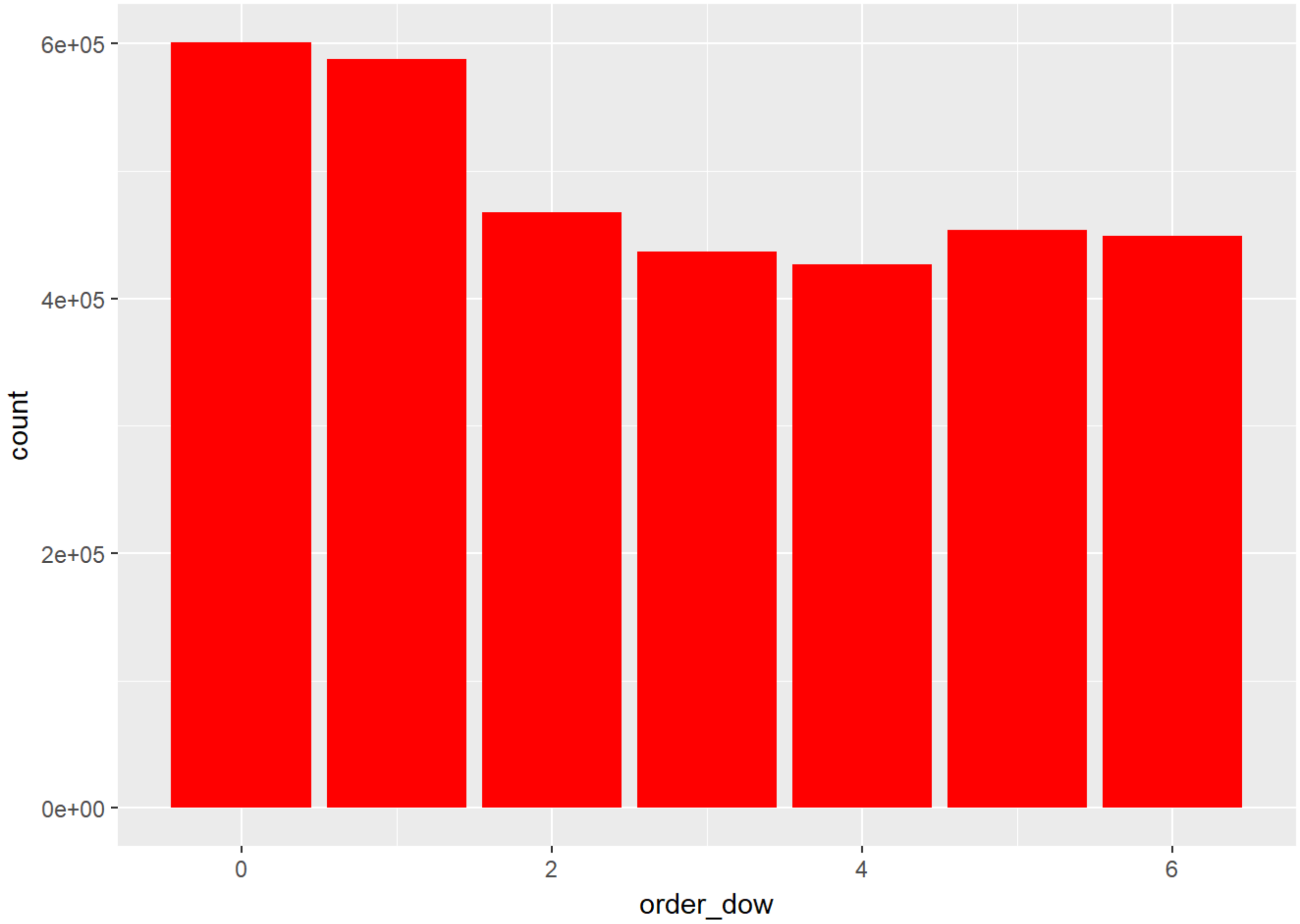
```
orders %>%
  ggplot(aes(x=order_hour_of_day)) +
  geom_histogram(stat="count",fill="red")
```



Day of Week

There is a clear effect of day of the week. Most orders are on days 0 and 1. Unfortunately there is no info regarding which values represent which day, but one would assume that this is the weekend.

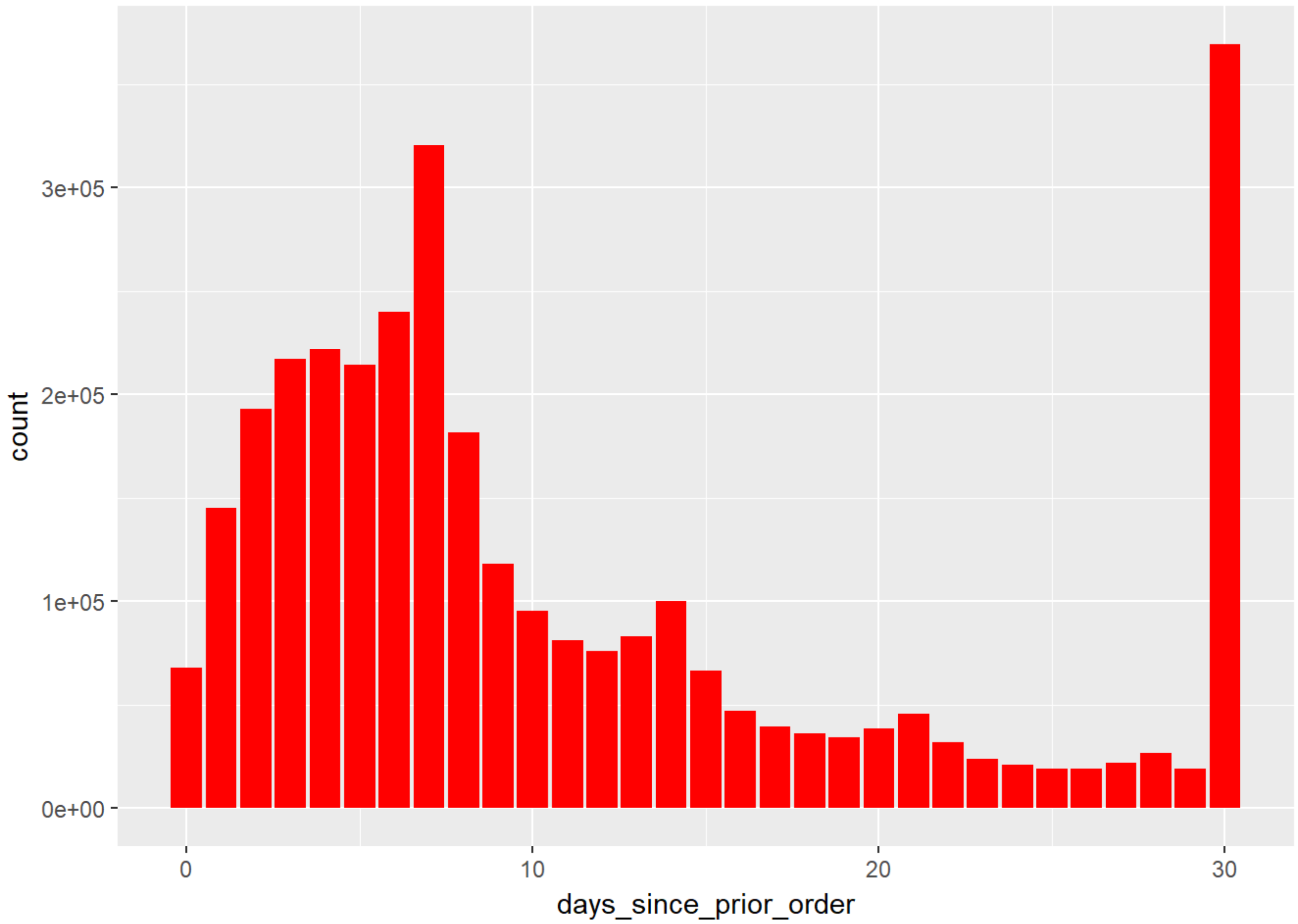
```
orders %>%
  ggplot(aes(x=order_dow)) +
  geom_histogram(stat="count",fill="red")
```



When do they order again?

People seem to order more often after exactly 1 week.

```
orders %>%
  ggplot(aes(x=days_since_prior_order)) +
  geom_histogram(stat="count",fill="red")
```



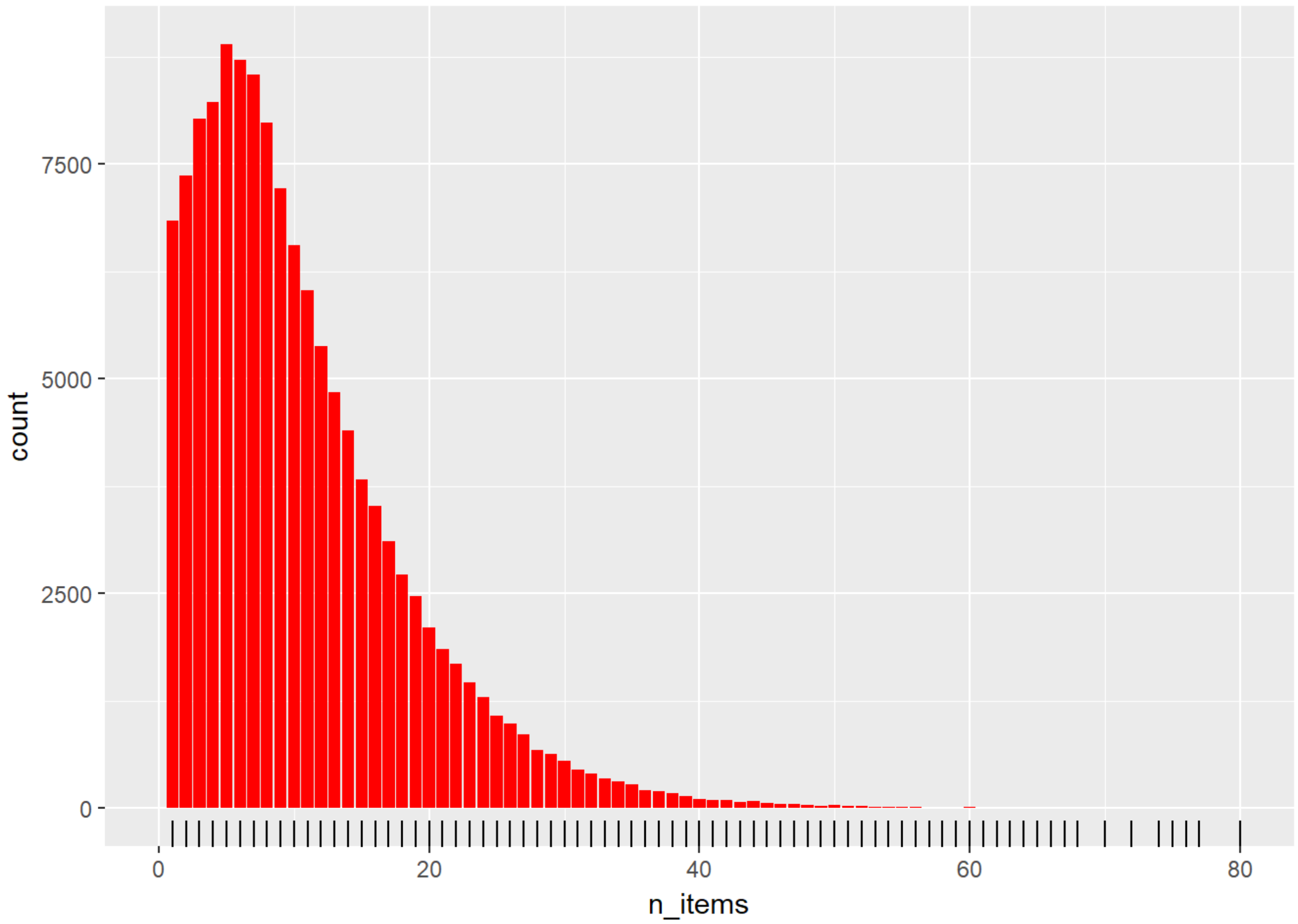
How many items do people buy?

Let’s have a look how many items are in the orders. We can see that people most often order around 5 items. The distributions are comparable between the train and prior order set.

Train set

Prior orders set

```
order_products %>%
  group_by(order_id) %>%
  summarize(n_items = last(add_to_cart_order)) %>%
  ggplot(aes(x=n_items))+
  geom_histogram(stat="count",fill="red") +
  geom_rug()+
  coord_cartesian(xlim=c(0,80))
```



Bestsellers

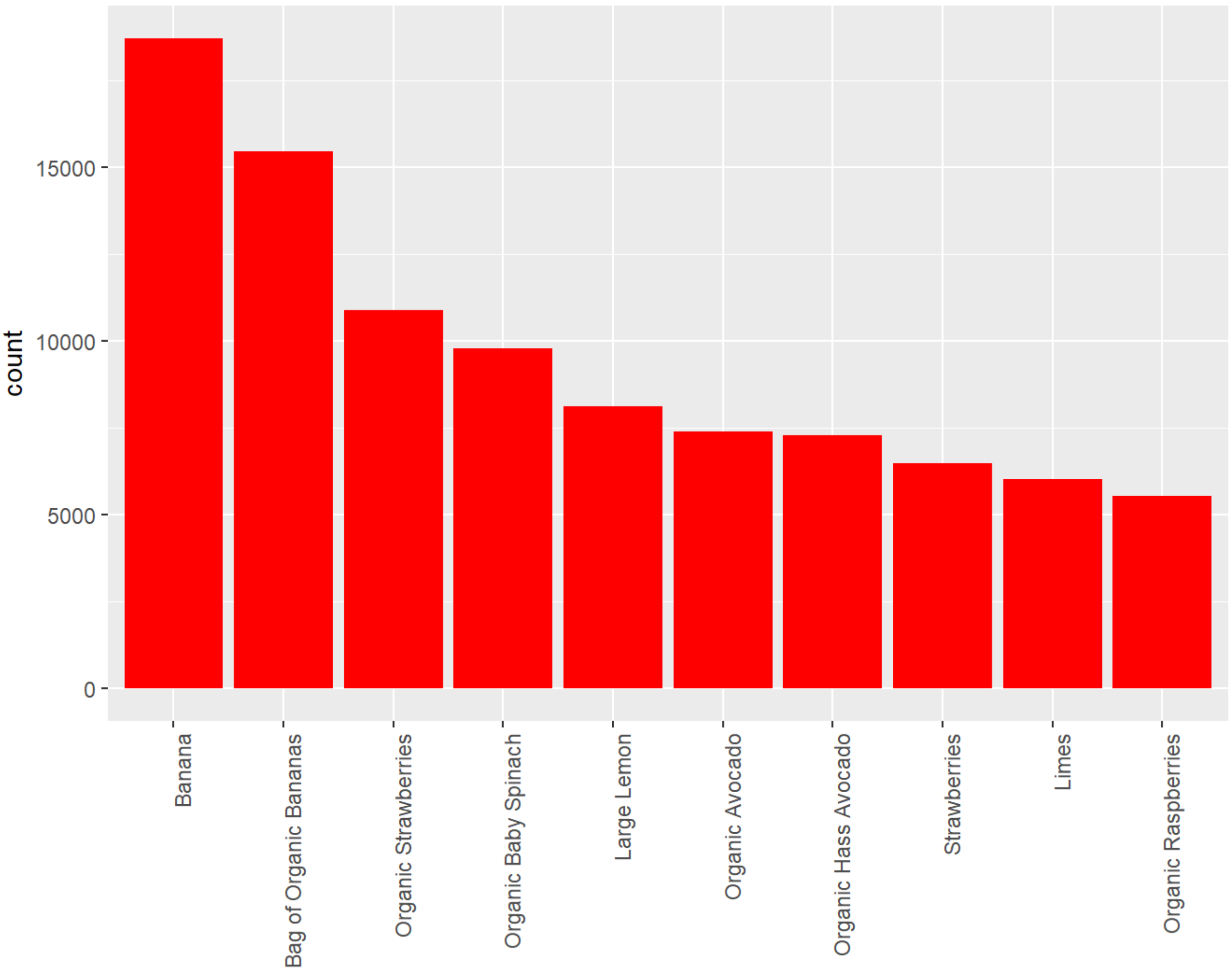
Let's have a look which products are sold most often (top10). And the clear winner is: **Bananas**

```
tmp <- order_products %>%
  group_by(product_id) %>%
  summarize(count = n()) %>%
  top_n(10, wt = count) %>%
  left_join(select(products,product_id,product_name),by="product_id") %>%
  arrange(desc(count))
kable(tmp)
```

product_id	count	product_name
24852	18726	Banana
13176	15480	Bag of Organic Bananas
21137	10894	Organic Strawberries
21903	9784	Organic Baby Spinach
47626	8135	Large Lemon
47766	7409	Organic Avocado
47209	7293	Organic Hass Avocado

product_id	count	product_name
16797	6494	Strawberries
26209	6033	Limes
27966	5546	Organic Raspberries

```
tmp %>%
  ggplot(aes(x=reorder(product_name,-count), y=count))+
  geom_bar(stat="identity",fill="red")+
  theme(axis.text.x=element_text(angle=90, hjust=1),axis.title.x = element_blank())
```



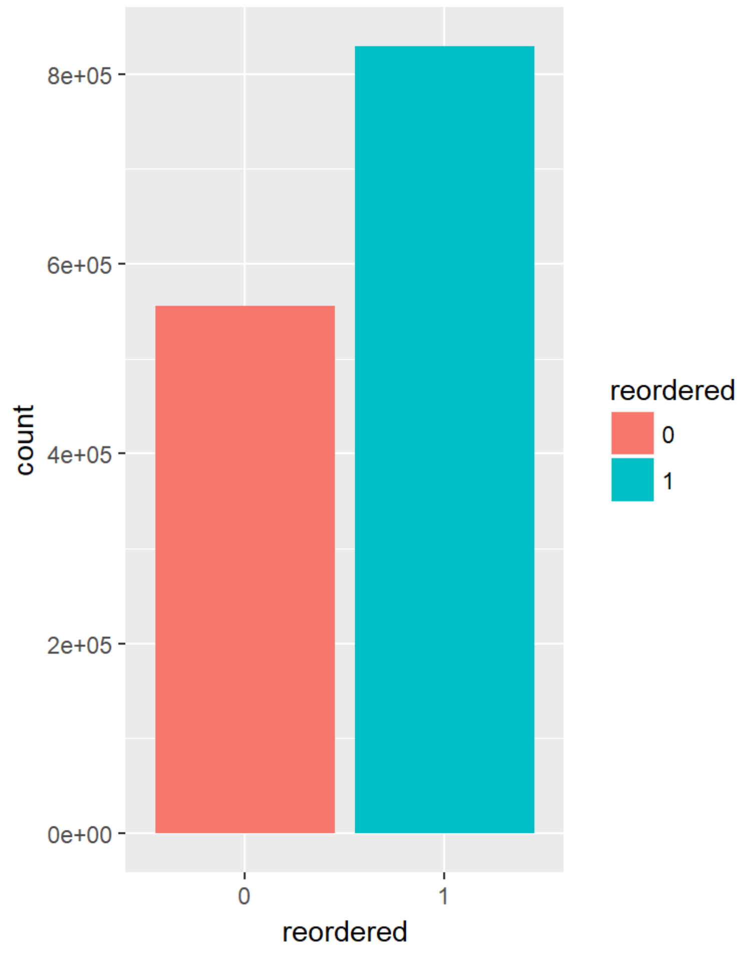
How often do people order the same items again?

59% of the ordered items are reorders.

```
tmp <- order_products %>%
  group_by(reordered) %>%
  summarize(count = n()) %>%
  mutate(reordered = as.factor(reordered)) %>%
  mutate(proportion = count/sum(count))
kable(tmp)
```

reordered	count	proportion
0	555793	0.4014056
1	828824	0.5985944

```
tmp %>%
  ggplot(aes(x=reordered,y=count,fill=reordered))+
  geom_bar(stat="identity")
```



Most often reordered

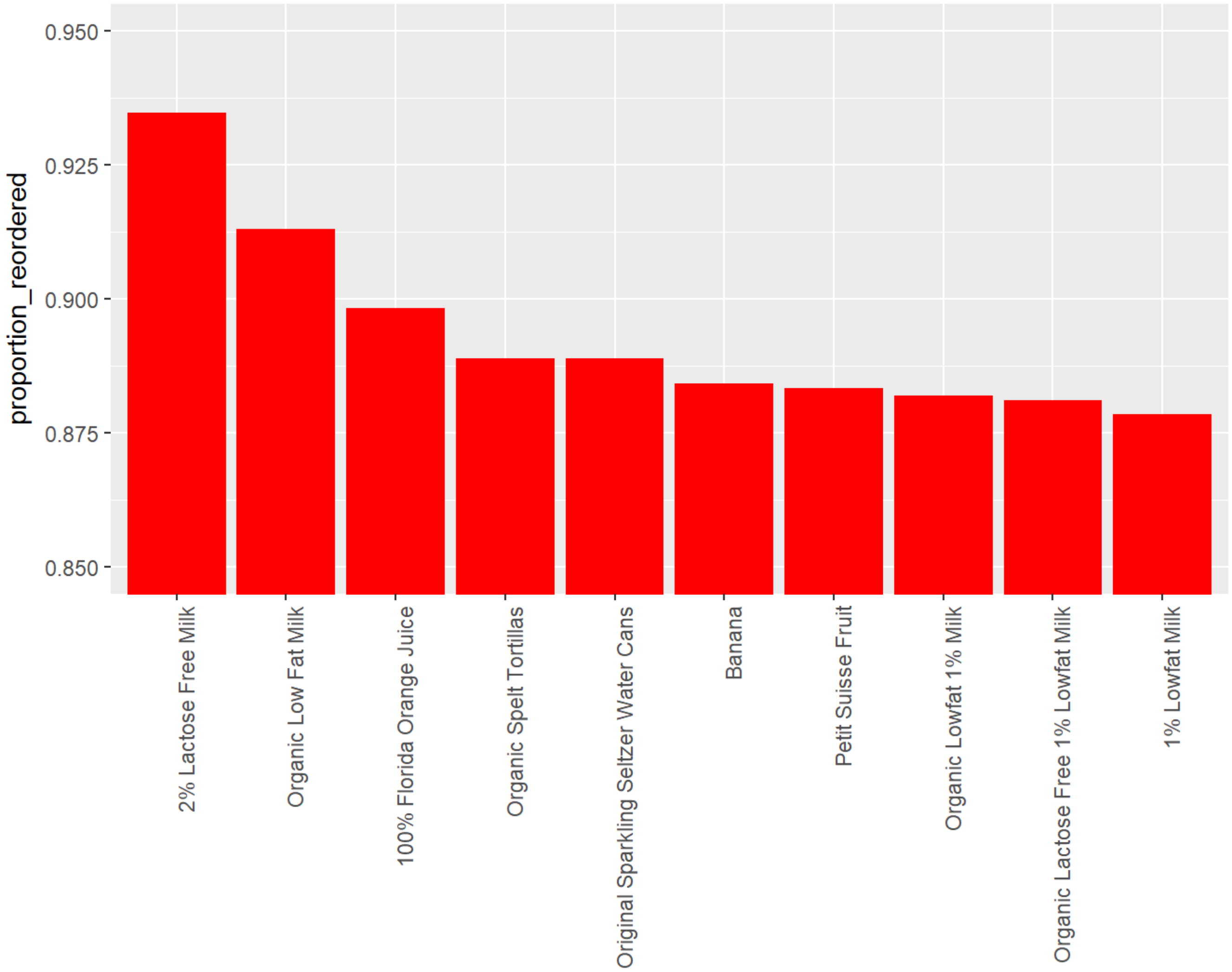
Now here it becomes really interesting. These 10 products have the highest probability of being reordered.

```
tmp <-order_products %>%
  group_by(product_id) %>%
  summarize(proportion_reordered = mean(reordered), n=n()) %>%
  filter(n>40) %>%
  top_n(10,wt=proportion_reordered) %>%
  arrange(desc(proportion_reordered)) %>%
  left_join(products,by="product_id")

kable(tmp)
```

product_id	proportion_reordered	n	product_name	aisle_id	department_id
1729	0.9347826	92	2% Lactose Free Milk	84	16
20940	0.9130435	368	Organic Low Fat Milk	84	16
12193	0.8983051	59	100% Florida Orange Juice	98	7
21038	0.8888889	81	Organic Spelt Tortillas	128	3
31764	0.8888889	45	Original Sparkling Seltzer Water Cans	115	7
24852	0.8841717	18726	Banana	24	4
117	0.8833333	120	Petit Suisse Fruit	2	16
39180	0.8819876	483	Organic Lowfat 1% Milk	84	16
12384	0.8810409	269	Organic Lactose Free 1% Lowfat Milk	91	16
24024	0.8785249	461	1% Lowfat Milk	84	16

```
tmp %>%
  ggplot(aes(x=reorder(product_name,-proportion_reordered), y=proportion_reordered))+
  geom_bar(stat="identity",fill="red")+
  theme(axis.text.x=element_text(angle=90, hjust=1),axis.title.x = element_blank())+coord_cartes
ian(ylim=c(0.85,0.95))
```



Which item do people put into the cart first?

People seem to be quite certain about Multifold Towels and if they buy them, put them into their cart first in 66% of the time.

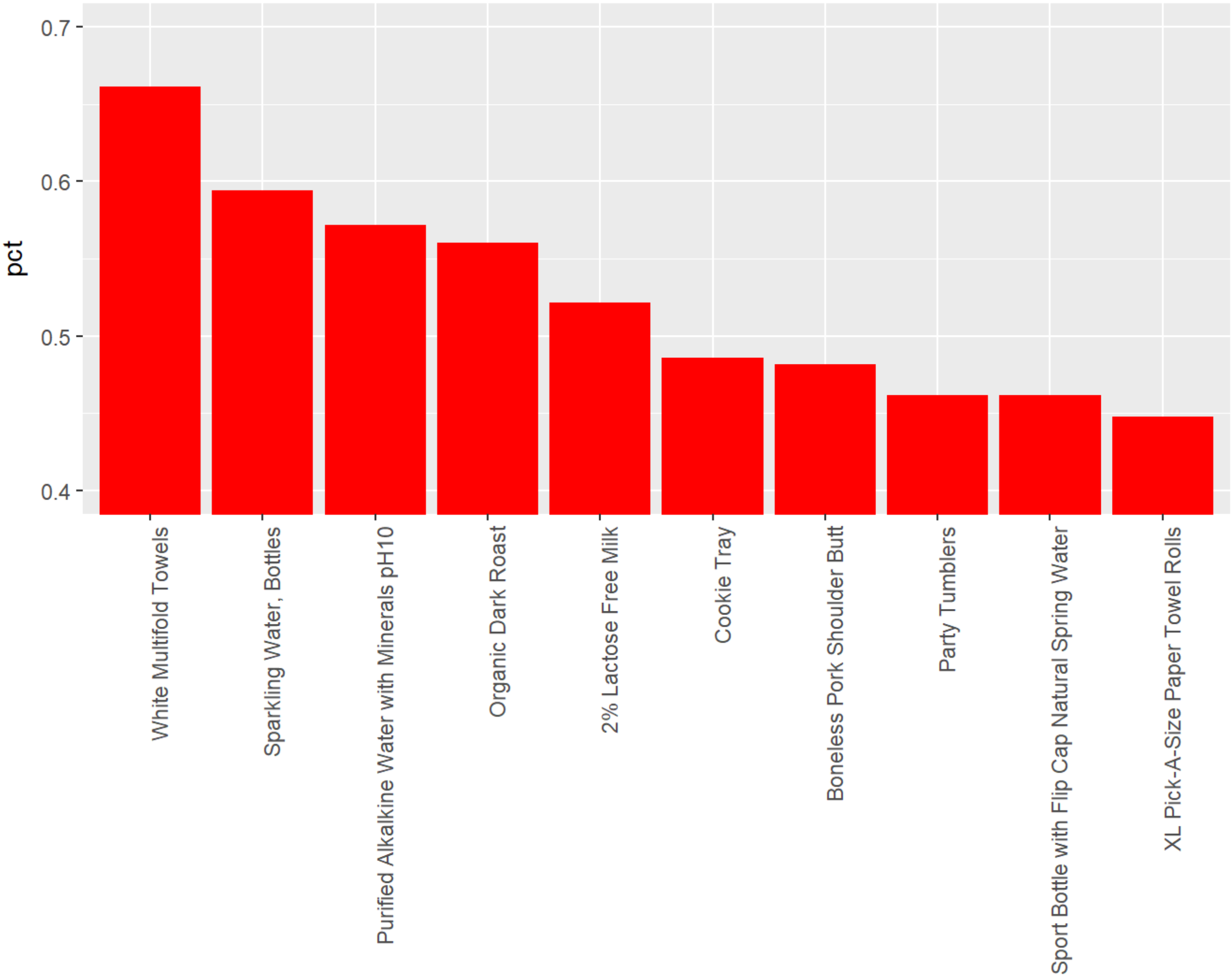
```
tmp <- order_products %>%
  group_by(product_id, add_to_cart_order) %>%
  summarize(count = n()) %>% mutate(pct=count/sum(count)) %>%
  filter(add_to_cart_order == 1, count>10) %>%
  arrange(desc(pct)) %>%
  left_join(products,by="product_id") %>%
  select(product_name, pct, count) %>%
  ungroup() %>%
  top_n(10, wt=pct)

kable(tmp)
```

product_id	product_name	pct	count
45004	White Multifold Towels	0.6610169	39
11885	Sparkling Water, Bottles	0.5942029	41

product_id	product_name	pct	count
13128	Purified Alkalkine Water with Minerals pH10	0.5714286	12
4100	Organic Dark Roast	0.5600000	14
1729	2% Lactose Free Milk	0.5217391	48
6729	Cookie Tray	0.4861111	35
9285	Boneless Pork Shoulder Butt	0.4814815	13
6848	Party Tumblers	0.4615385	12
12640	Sport Bottle with Flip Cap Natural Spring Water	0.4615385	12
26405	XL Pick-A-Size Paper Towel Rolls	0.4476190	47

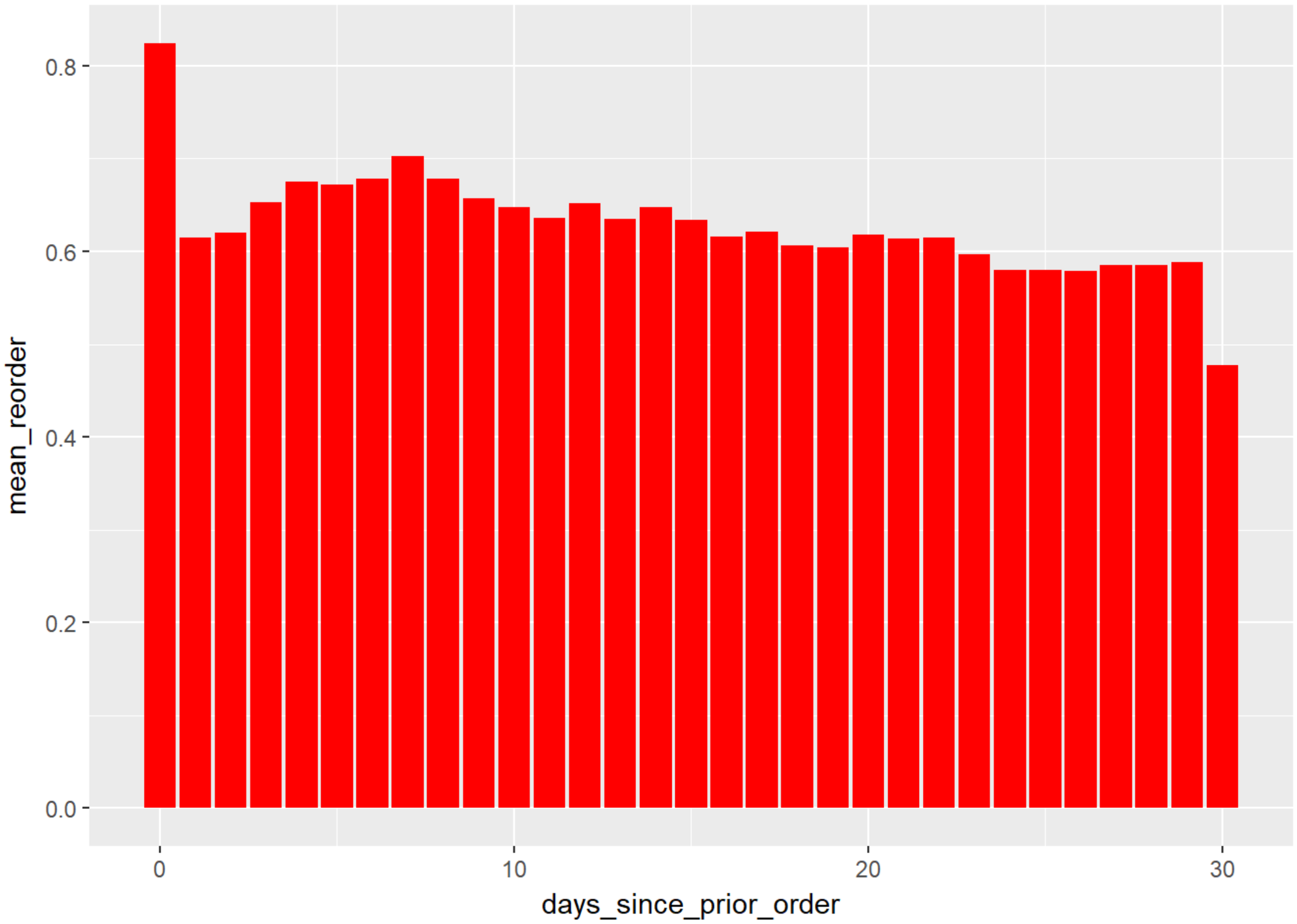
```
tmp %>%
  ggplot(aes(x=reorder(product_name,-pct), y=pct))+
  geom_bar(stat="identity",fill="red")+
  theme(axis.text.x=element_text(angle=90, hjust=1),axis.title.x = element_blank())+coord_cartesian(ylim=c(0.4,0.7))
```



Association between time of last order and probability of reorder

This is interesting: We can see that if people order again on the same day, they order the same product more often. Whereas when 30 days have passed, they tend to try out new things in their order.

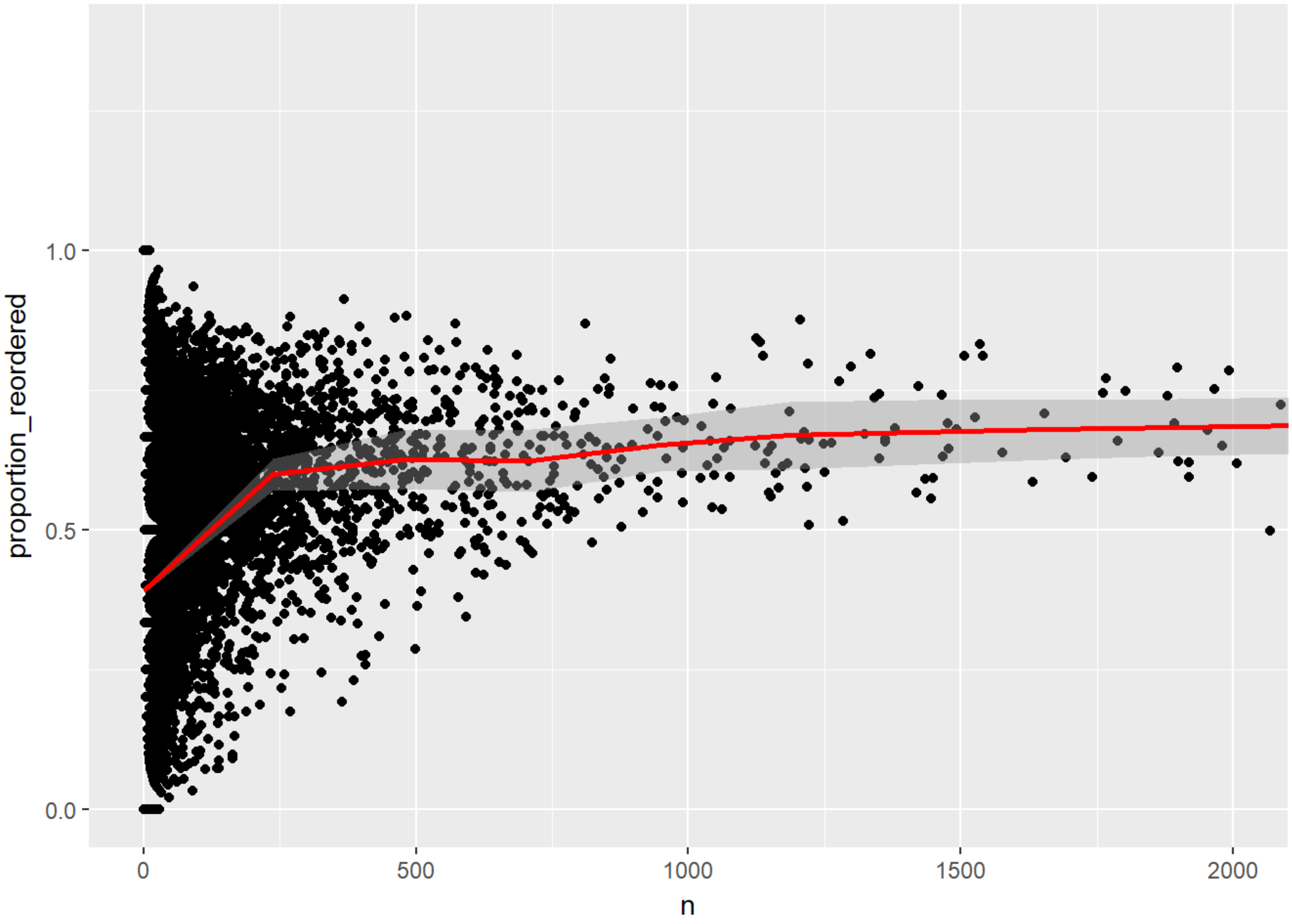
```
order_products %>%
  left_join(orders,by="order_id") %>%
  group_by(days_since_prior_order) %>%
  summarize(mean_reorder = mean(reordered)) %>%
  ggplot(aes(x=days_since_prior_order,y=mean_reorder))+
  geom_bar(stat="identity",fill="red")
```



Association between number of orders and probability of reordering

Products with a high number of orders are naturally more likely to be reordered. However, there seems to be a ceiling effect.

```
order_products %>%
  group_by(product_id) %>%
  summarize(proportion_reordered = mean(reordered), n=n()) %>%
  ggplot(aes(x=n,y=proportion_reordered))+
  geom_point()+
  geom_smooth(color="red")+
  coord_cartesian(xlim=c(0,2000))
```



Visualizing the Product Portfolio

Here is use to treemap package to visualize the structure of instacarts product portfolio. In total there are 21 departments containing 134 aisles.

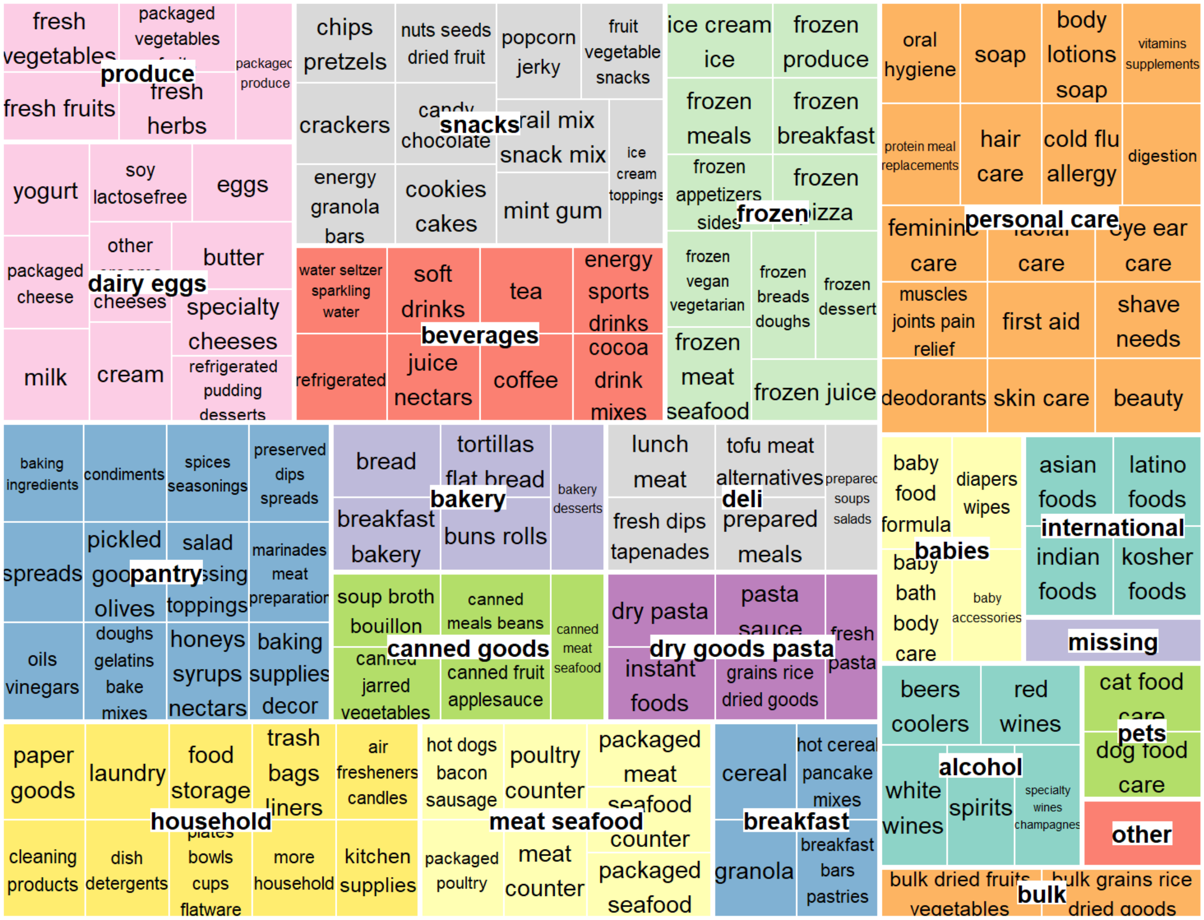
```
# install.packages("treemap")
library(treemap)

tmp <- products %>% group_by(department_id, aisle_id) %>% summarize(n=n())
tmp <- tmp %>% left_join(departments,by="department_id")
tmp <- tmp %>% left_join(aisles,by="aisle_id")

tmp2<-order_products %>%
  group_by(product_id) %>%
  summarize(count=n()) %>%
  left_join(products,by="product_id") %>%
  ungroup() %>%
  group_by(department_id,aisle_id) %>%
  summarize(sumcount = sum(count)) %>%
  left_join(tmp, by = c("department_id", "aisle_id")) %>%
  mutate(onesize = 1)
```

How are aisles organized within departments?

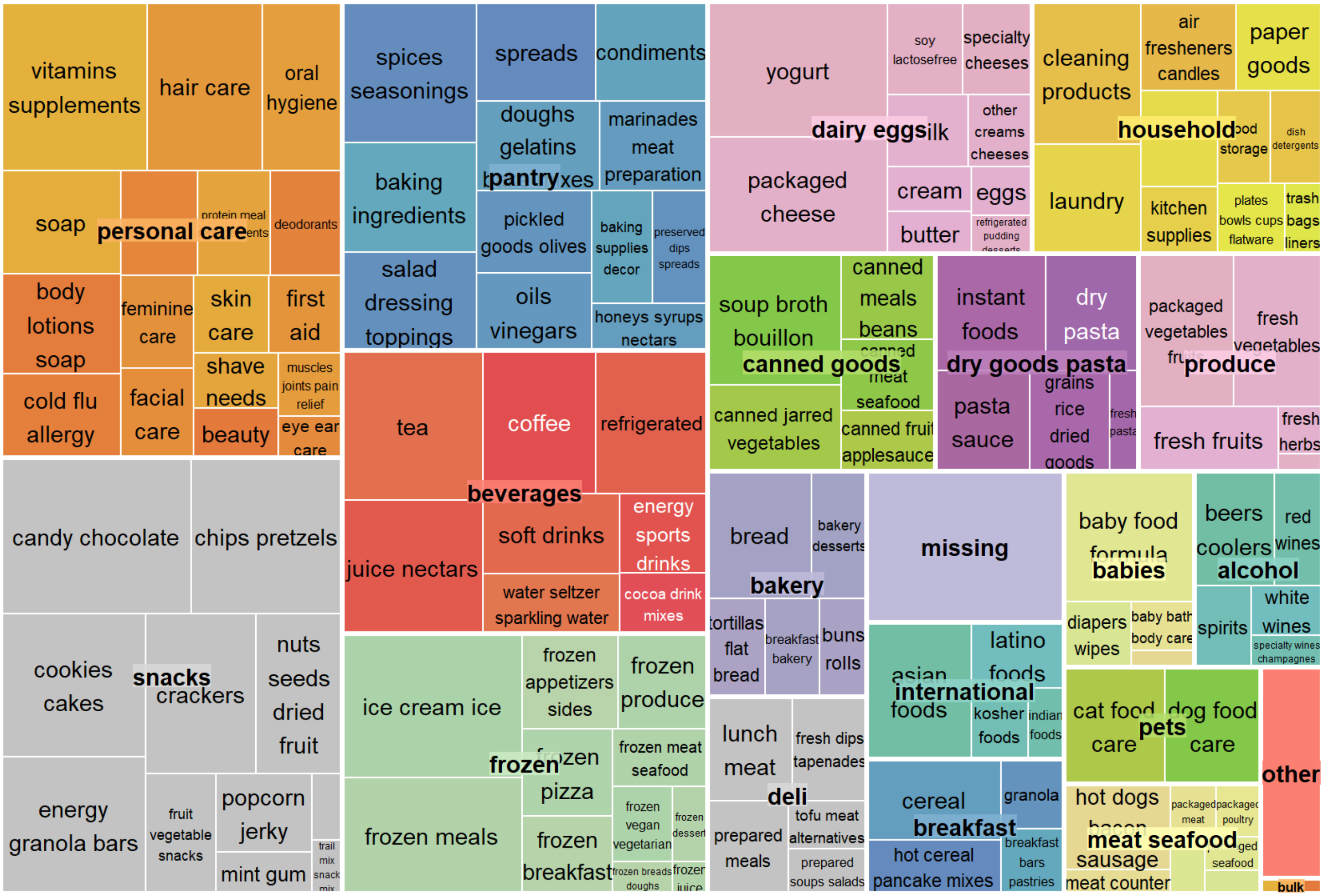
```
treemap(tmp2,index=c("department","aisle"),vSize="onesize",vColor="department",palette="Set3",ti
tle="",sortID="-sumcount", border.col="#FFFFFF",type="categorical", fontsize.legend = 0,bg.label
s = "#FFFFFF")
```



How many unique products are offered in each department/aisle?

The size of the boxes shows the number of products in each category.

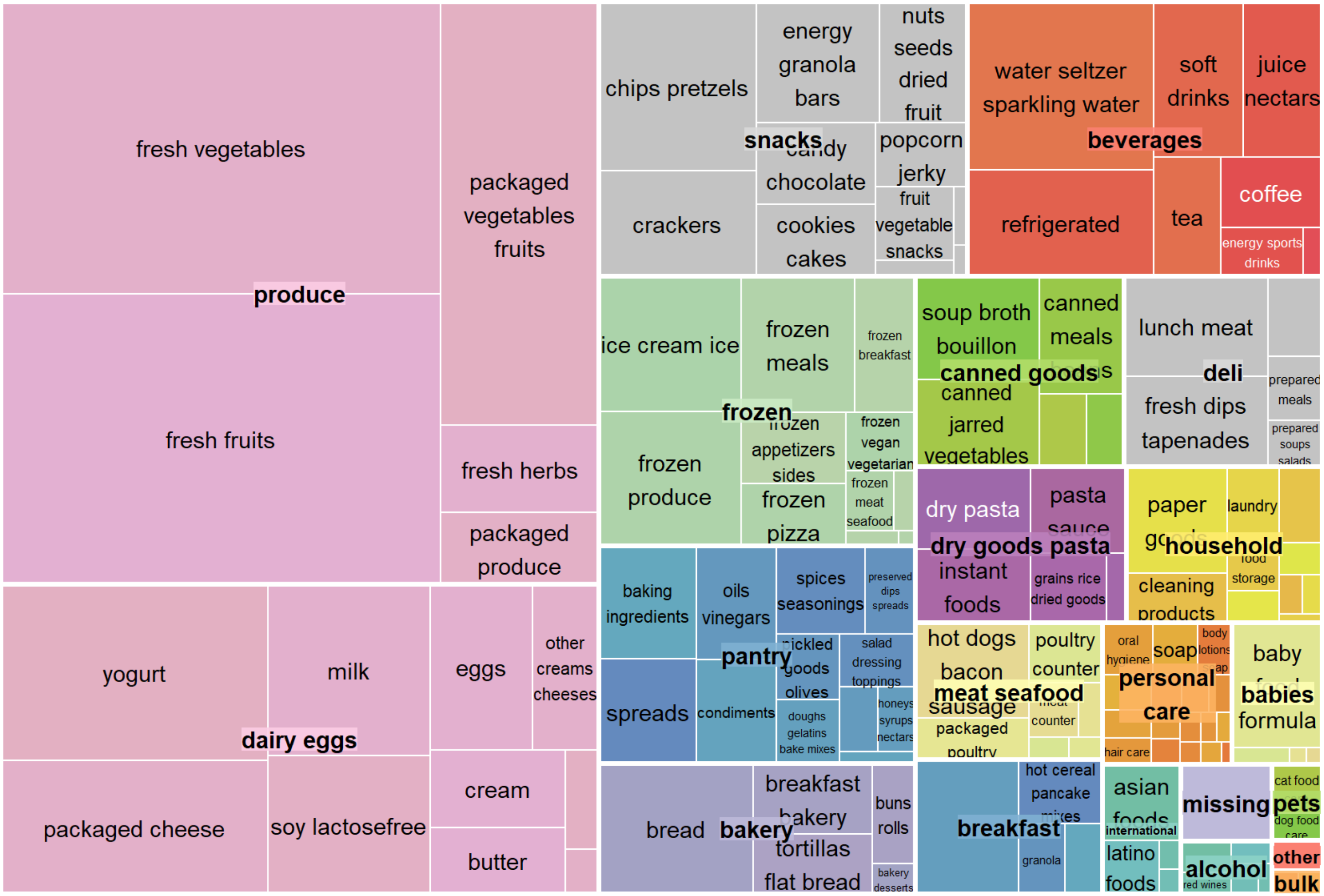
```
treemap(tmp,index=c("department","aisle"),vSize="n",title="",palette="Set3",border.col="#FFFFFF")
```



How often are products from the department/aisle sold?

The size of the boxes shows the number of sales.

```
treemap(tmp2,index=c("department","aisle"),vSize="sumcount",title="",palette="Set3",border.col="#F
FFFF")
```


fresh vegetables
packaged vegetables fruits
produce
fresh fruits
fresh herbs
packaged produce
yogurt
milk
eggs
other creams cheeses
dairy eggs
packaged cheese
soy lactosefree
cream
butter
chips pretzels
energy granola bars
nuts seeds dried fruit
snacks
candy chocolate
popcorn jerky
fruit vegetable snacks
crackers
cookies cakes
water seltzer sparkling water
soft drinks
juice nectars
beverages
coffee
refrigerated
tea
energy sports drinks
ice cream ice
frozen meals
frozen breakfast
frozen
frozen appetizers sides
frozen vegan vegetarian
frozen produce
frozen pizza
frozen meat seafood
soup broth bouillon
canned meals beans
canned goods
canned jarred vegetables
lunch meat
deli
prepared meals
fresh dips tapenades
prepared soups salads
dry pasta
pasta sauce
dry goods pasta
instant foods
grains rice dried goods
paper goods
laundry
household
food storage
cleaning products
baking ingredients
oils vinegars
spices seasonings
preserved dips spreads
pantry
pickled goods olives
salad dressing toppings
spreads
condiments
doughs gelatins bake mixes
honeys syrups nectars
hot dogs bacon sausage
poultry counter
meat seafood
meat counter
packaged poultry
oral hygiene
soap
body lotions soap
personal care
hair care
baby food formula
babies
bread
breakfast bakery
bakery
buns rolls
tortillas flat bread
bakery desserts
hot cereal pancake mixes
breakfast
granola
asian foods
international
latino foods
missing
cat food
pets
dog food care
alcohol
red wines
other
bulk