logistic_biplot_script

Camilo Estrella

2024-02-20

Loading libraries

We will start by loading the libraries that will be used, whose function is explained below:

```
library(MultBiplotR)
library(knitr)
library(dplyr)
library(ggplot2)
library(scales)
```

- **MultBiplotR**: It allows multivariate analysis using logistic regression and its graphical representation employing a Biplot.
- **knitr**: The *kable* function is used to obtain a better visualization of the tables.
- **dplyr**: It has multiple functions that allow data management, some of the most interesting are those that allow you to manage tables as if it were a relational database, as is done with SQL.
- **ggplot2**: Perhaps the most important library for the creation of statistical graphics in R.
- **scales**: It is used in conjunction with ggplot2 to scale axes within the plot.

Loading datasets

The original datasets are 5, named as in the code. In addition, there is a sixth one called "cruzada" (a cross of all 5 tables), which has binary information of aisles of a supermarket in the columns and rows representing the customers. The value 1 means that the customer has shopped in that aisle and 0 the opposite.

The calculation of the "cross" dimension of the dataset is included first.

```
order <- read.csv("Order.csv")
order_product <- read.csv("Order_Product.csv")
product <- read.csv("Product.csv")
aisle <- read.csv("Aisle.csv")
department <- read.csv("Department.csv")

cruzada <- as.data.frame(read.csv("cruzada_df.csv"))
```

```
dimensions <- as.data.frame(matrix(dim(cruzada), ncol = 2))
names(dimensions) <- c("customers (rows)", "aisles (columns)")
kable(dimensions, row.names = FALSE, align = "cc")
```

customers (rows)	aisles (columns)
206209	134

The datasets are initially from a supermarket and cannot be shared due to source privacy issues.

Extract the first 30 aisles with the most sales

Why only take the first 30 aisles into account? The idea is to reduce the dimensionality of the "cruzada" dataset, so the first 30 aisles with the most sales are taken.

This is possible through the following 4 steps:

- **Join tables to map each product_id to its aisle_id:**

```
order_product_info <- order_product %>%
  inner_join(product, by = "product_id")
```

- **Counting sales per aisle:**

```
aisle_counts <- order_product_info %>%
  group_by(aisle_id) %>%
  summarise(sales_count = n()) %>%
  ungroup()
```

- **Sort and select the first 30 aisles:**

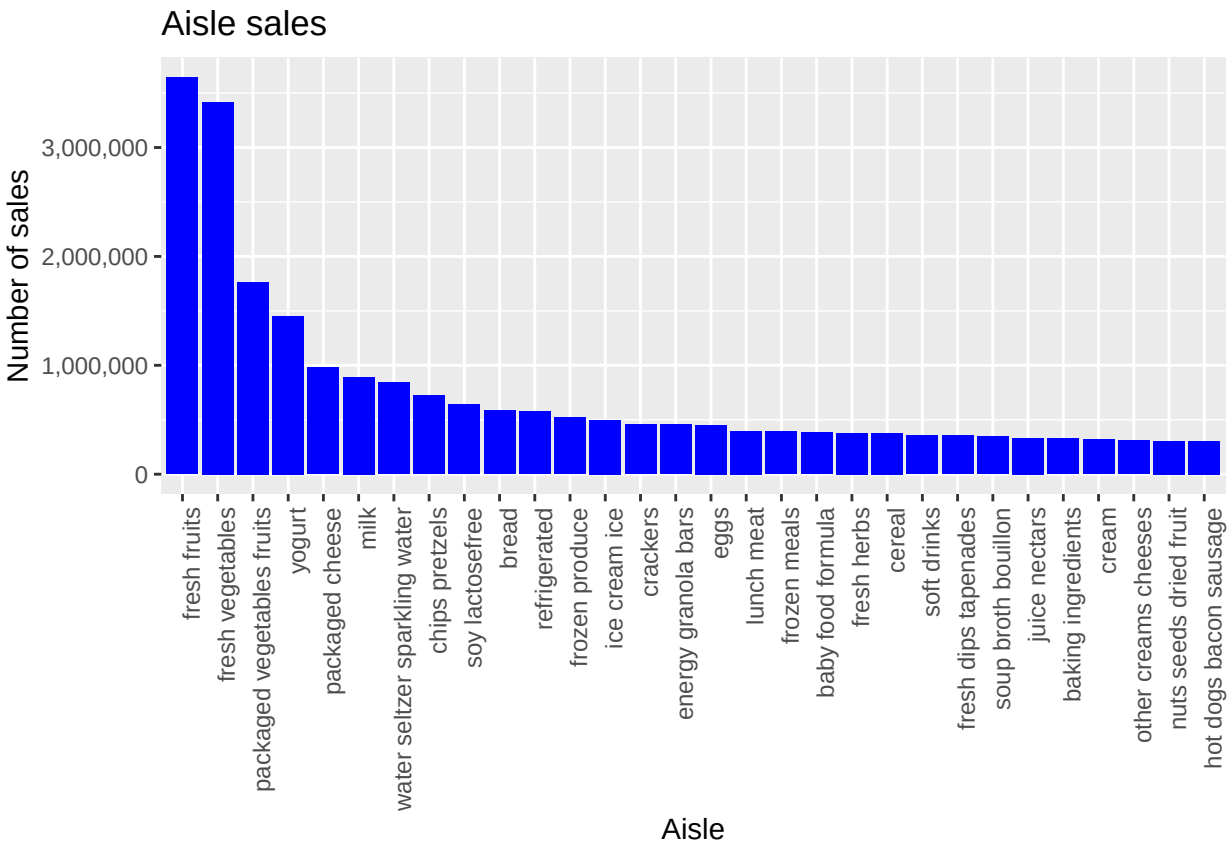
```
top_aisles <- aisle_counts %>%
  arrange(desc(sales_count)) %>%
  slice(1:30)
```

- **Relate 'aisle_id' to aisle names:**

```
top_aisles <- top_aisles %>%
  inner_join(aisle, by = "aisle_id")
```

And finally, we can create a plot that contains the distribution of this data in descending order.

```
ggplot(top_aisles, aes(x = reorder(aisle, -sales_count), y = sales_count)) +
  geom_bar(stat = "identity", fill = "blue") +
  scale_y_continuous(labels = comma) + # This will change the format of the numbers on the y-axis.
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  labs(x = "Aisle", y = "Number of sales", title = "Aisle sales")
```



It is possible to see that the aisles with the most purchases are *fresh fruits*, *fresh vegetables*, *packaged vegetables fruits*, and *yogurt*, all with more than 1 million sales.

Reduction of variables in "cruzada" table

It is necessary to reduce the dimensionality of the "cruzada" table from 134 variables to only 30, the 30 we found in the immediate previous step. This can be done in 3 steps:

- **Replace spaces with periods in corridor names:**

```
top_aisles$aisle <- gsub(" ", ".", top_aisles$aisle)
```

- **Extract aisle names from top_aisles:**

```
top_aisle_names <- top_aisles$aisle
```

- **Filter 'cruzada' variables to include only those of top_aisle_names:**

```
reduced_cruzada <- cruzada %>%
  select(all_of(top_aisle_names))
```

In this way, the new table has 30 variables and 206209 elements (rows).

Generate a random sample

Generating a simple random sampling is vital to reduce the dimensionality corresponding to the number of rows since biplot methods present the difficulty that the more rows and columns there are, the more difficult it is to interpret the results since everything becomes difficult to read. graphically.

Consequently, five random samples were created from reduced_cross. The samples have 300, 400, 500, 700, and 1000 elements and were created using the following code:

```
# 300 sample
set.seed(300)
random_dataset_300 <- reduced_cruzada[sample(nrow(reduced_cruzada), size=300),]

# 400 sample
set.seed(400)
random_dataset_400 <- reduced_cruzada[sample(nrow(reduced_cruzada), size=400),]

# 500 sample
set.seed(500)
random_dataset_500 <- reduced_cruzada[sample(nrow(reduced_cruzada), size=500),]

# 700 sample
set.seed(700)
random_dataset_700 <- reduced_cruzada[sample(nrow(reduced_cruzada), size=700),]

# 1000 sample
set.seed(1000)
random_dataset_1000 <- reduced_cruzada[sample(nrow(reduced_cruzada), size=1000),]
```

It is important to remember the use of the seed since it is what allows the replicability of the random sample, permanently storing the same dataframe that has been defined within the seed and avoiding the generation of random sampling that we will not be able to replicate later.

Validation of samples

It is not advisable to use random samples directly without first performing some validation that these are representative samples of the dataset that we have chosen as the population.

Below is the code where it is seen that a graphic technique was applied and another chi-square test that yields a p-value to check whether the samples meet the conditions of representativeness statistically.

- **Function to calculate proportions and create a dataframe:**

```
create_proportions_df <- function(complete_dataset, sample_dataset, suffix) {
  proportions_complete <- colSums(complete_dataset) / nrow(complete_dataset)
  proportions_sample <- colSums(sample_dataset) / nrow(sample_dataset)

  # Create a dataframe with the proportions of the sample and the complete set
  proportions_df <- data.frame(
    Aisle = rep(names(proportions_complete), 2),
    Proportion = c(proportions_complete, proportions_sample),
    Type = rep(c("Complete", "Sample"), each = length(proportions_complete))
  )
```

```
  # Convert 'Type' to a factor with consistent levels for the legend
  proportions_df$Type <- factor(proportions_df$Type, levels = c("Complete", "Sample"))

  return(proportions_df)
}
```

As can be seen, since this is binary data, we are working on proportions, and these are compared with each other.

- **Function to plot proportions:**

```
plot_proportions <- function(proportions_df, suffix) {
  ggplot(proportions_df, aes(x = Aisle, y = Proportion, fill = Type)) +
    geom_bar(stat = "identity", position = position_dodge(width = 0.7)) +
    scale_fill_manual(
      values = c("Complete" = "blue", "Sample" = "red"),
      labels = c("Complete dataset", paste("Sample", suffix))
    ) +
    theme_minimal() +
    theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
    labs(y = "Proportion of Sales", x = "Aisles", fill = "Type",
         title = paste("Comparison of Proportions of Sales between Full and Sample Datasets", suffix))
}
```

- **Function to perform chi-square test and return adjusted and unadjusted p-values:**

The chi-square test is run with and without adjustment. This adjustment uses the Bonferroni technique to prevent type 1 errors when multiple comparisons are made simultaneously. Both types are presented because it can be helpful to see how the result changes.

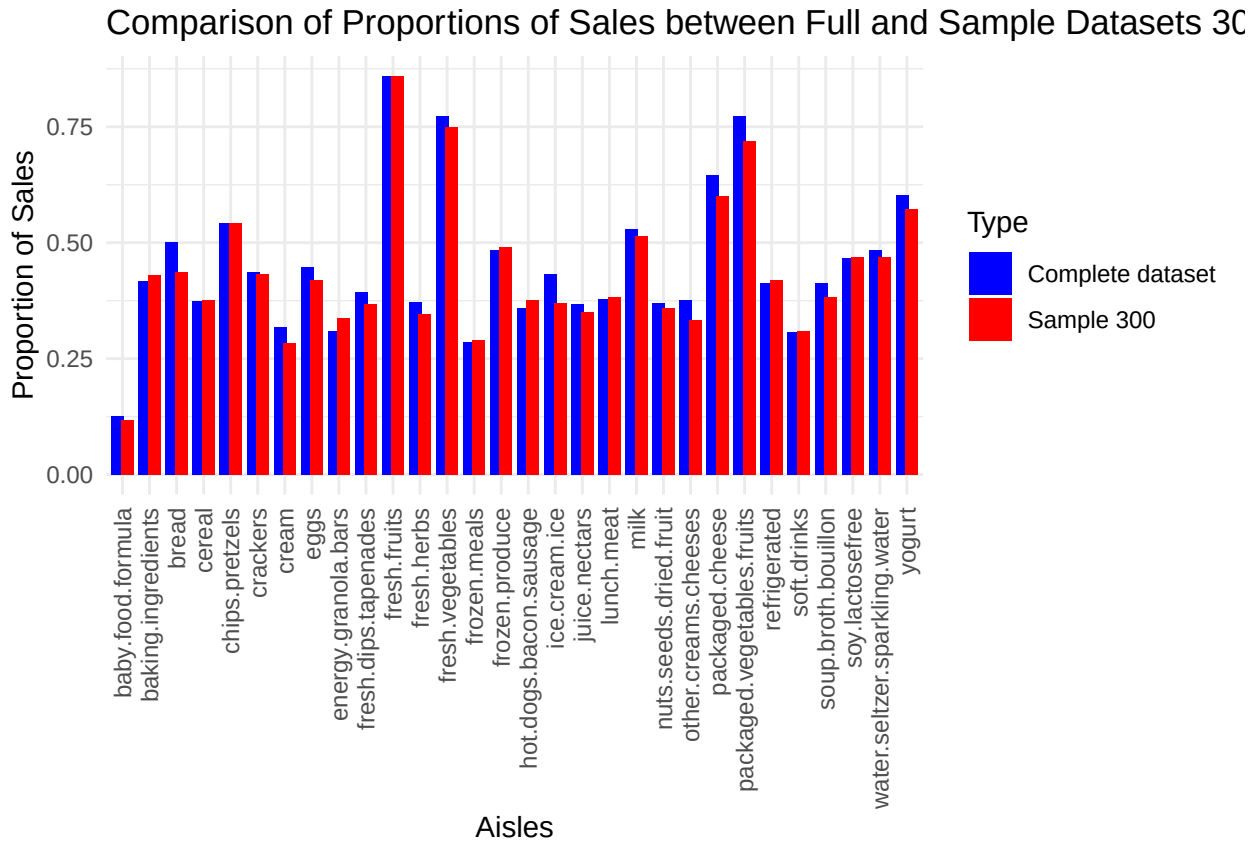
```
perform_chi_square_test <- function(complete_dataset, sample_dataset) {
  successes_complete <- colSums(complete_dataset)
  failures_complete <- nrow(complete_dataset) - successes_complete
  successes_sample <- colSums(sample_dataset)
  failures_sample <- nrow(sample_dataset) - successes_sample

  results_chi_square <- sapply(names(successes_complete), function(name) {
    contingency_table <- matrix(c(successes_complete[name], successes_sample[name],
                                  failures_complete[name], failures_sample[name]),
                                nrow = 2)
    chisq.test(contingency_table)$p.value
  })

  adjusted_p_values <- p.adjust(results_chi_square, method = "bonferroni")

  # Return a list with adjusted and unadjusted p-values
  return(list(unadjusted_p_values = results_chi_square, adjusted_p_values = adjusted_p_values))
}
```

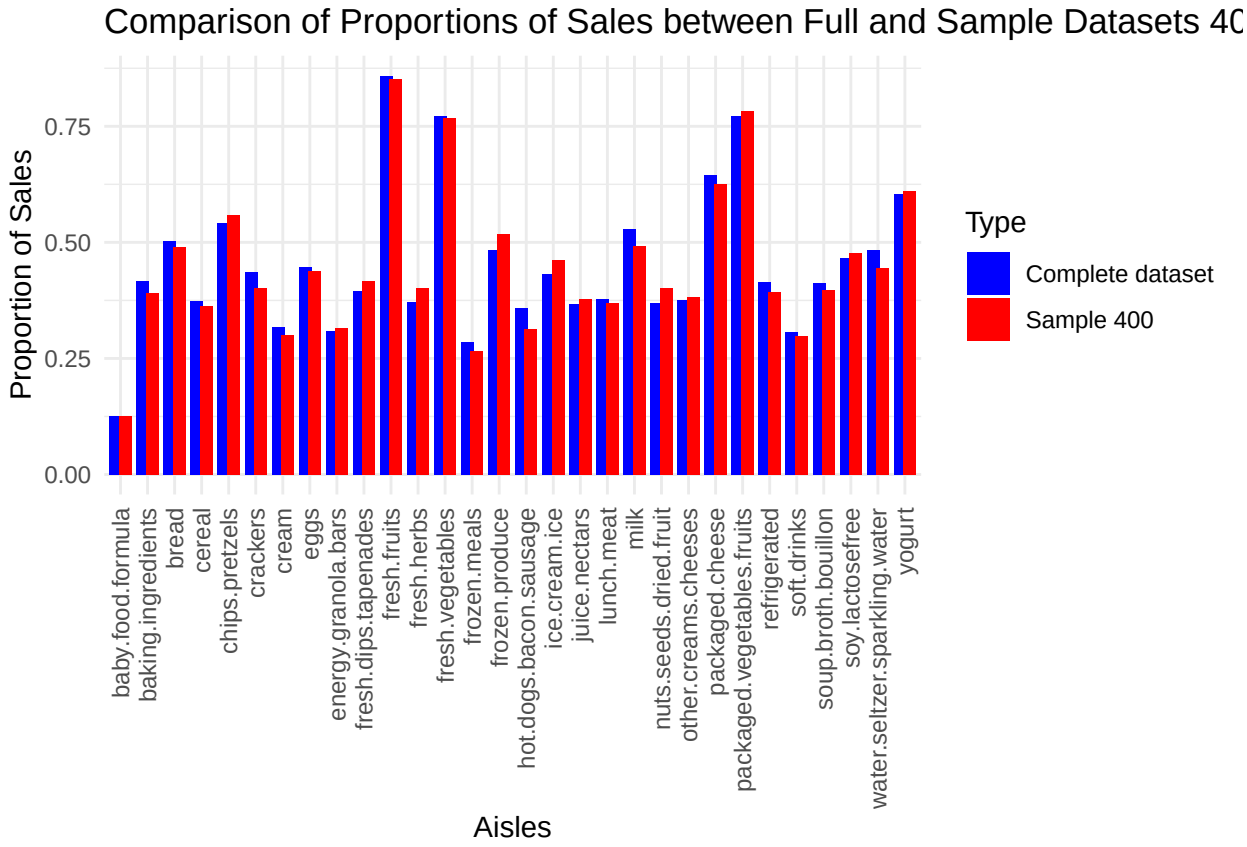
Some other complementary variables to continue with the execution of the code are the following:

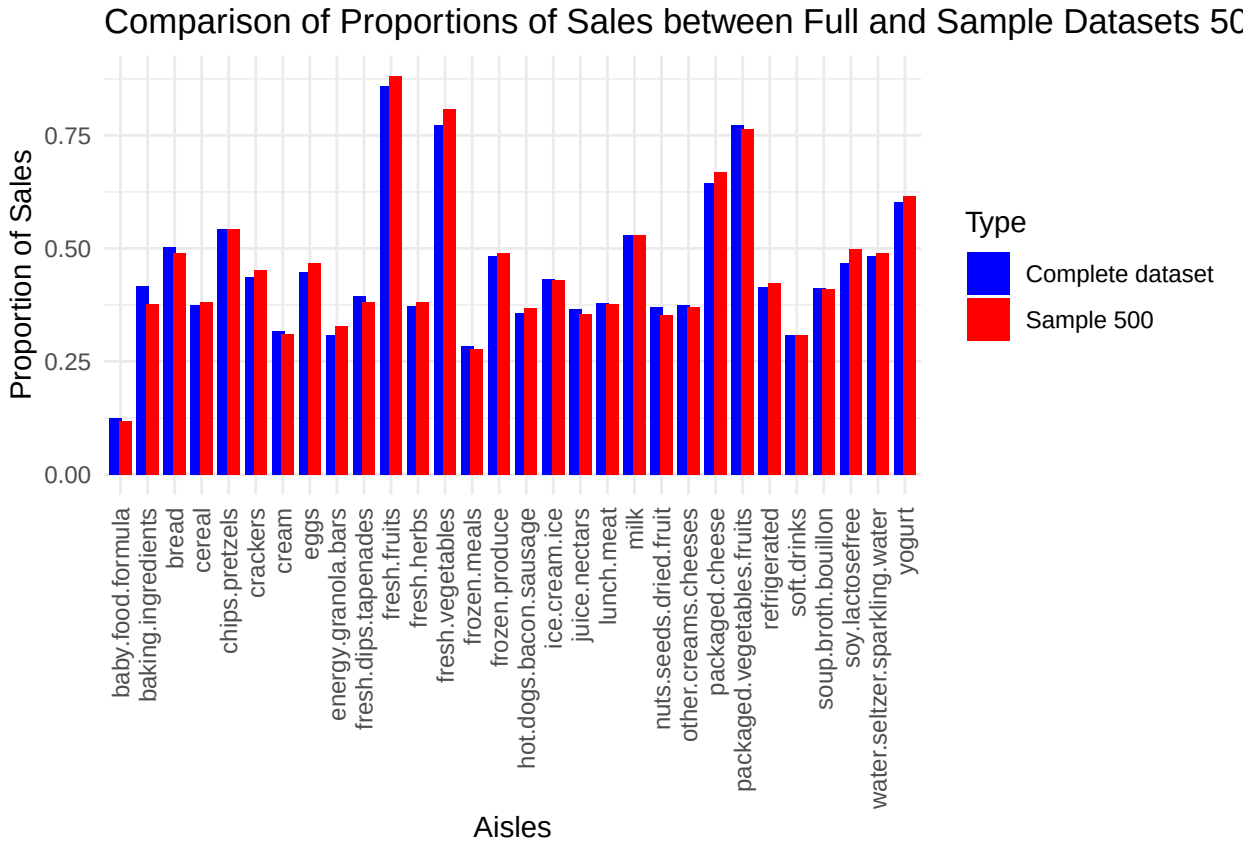
```r
# List for storing charts
chart_list <- list()

# Dictionary for storing chi-square test results (adjusted and unadjusted)
results_p_values <- list()

# Process samples, plot and perform chi-square tests
suffixes <- c("300", "400", "500", "700", "1000")

for(suffix in suffixes) {
  sample_dataset <- get(paste("random_dataset_", suffix, sep = ""))
  proportions_df <- create_proportions_df(reduced_cruzada, sample_dataset, suffix)
  chart <- plot_proportions(proportions_df, suffix)
  chart_list[[suffix]] <- chart # Store the chart in the list

  results <- perform_chi_square_test(reduced_cruzada, sample_dataset)
  results_p_values[[suffix]] <- results
}
```

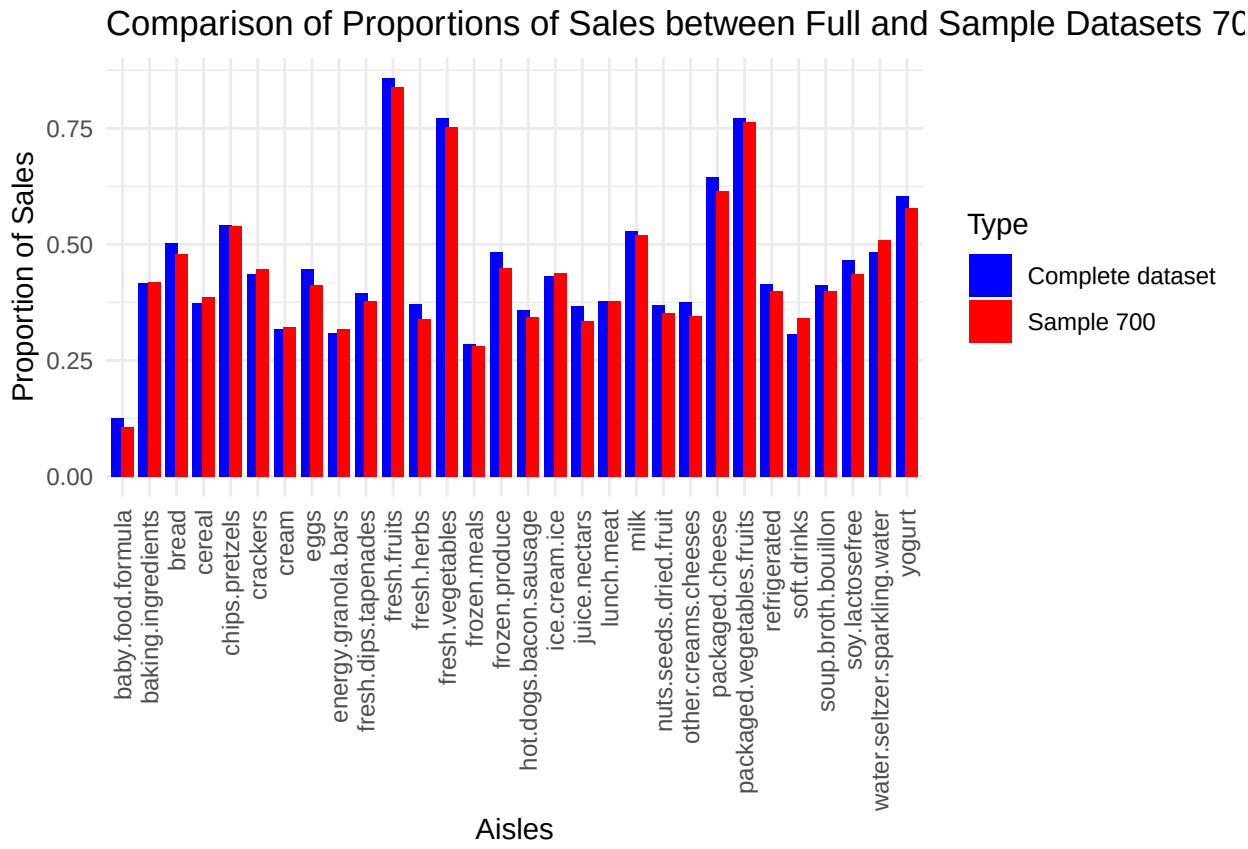
The sample validation graphs are printed below, where the proportions are compared:

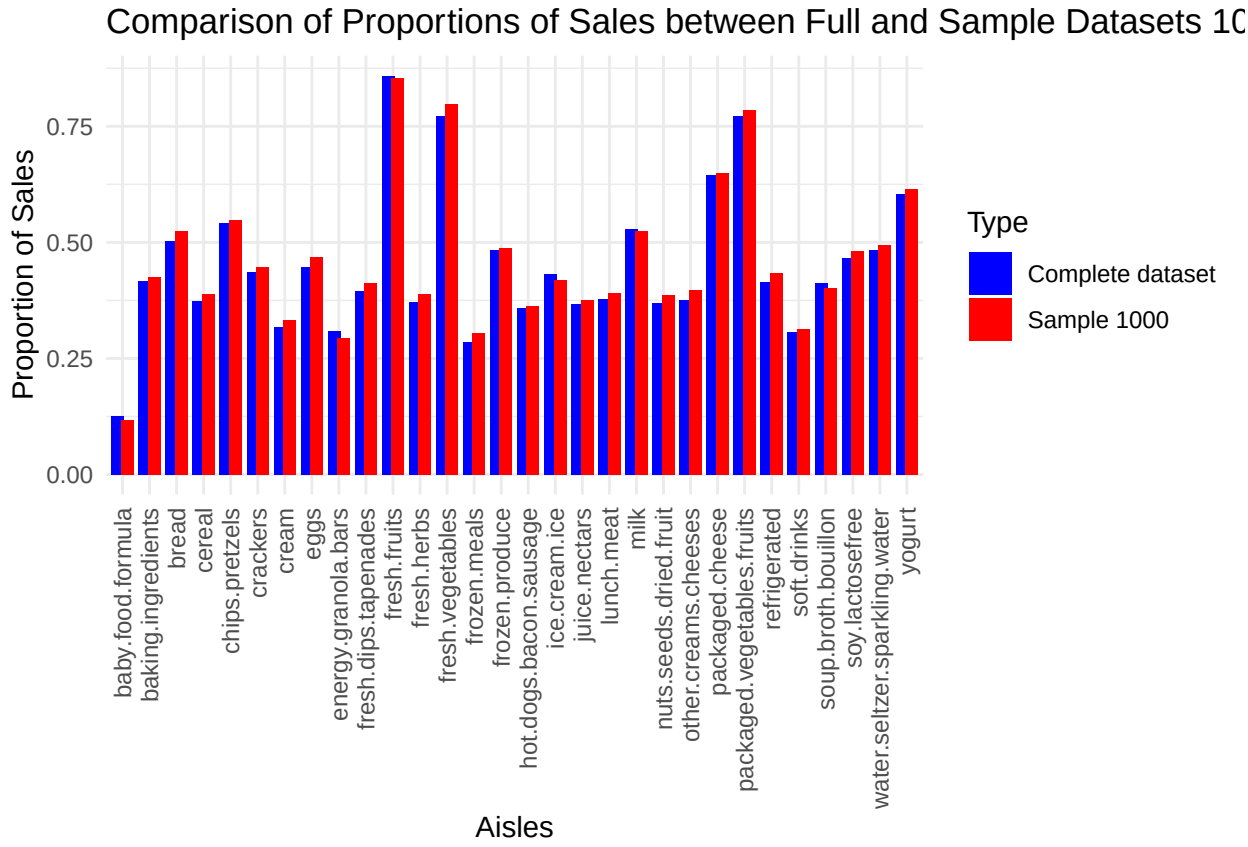
```r
lapply(chart_list, print)
```



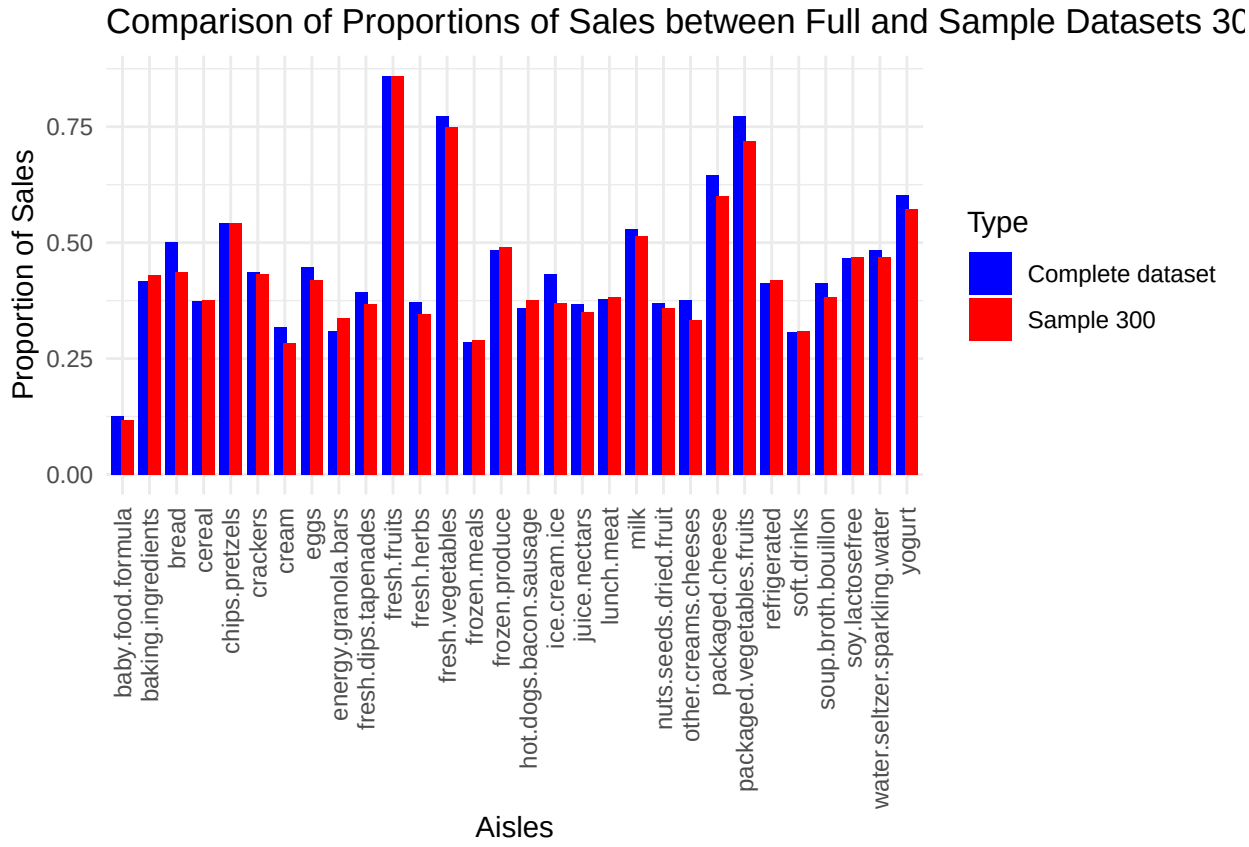

Comparison of Proportions of Sales between Full and Sample Datasets 40
Proportion of Sales
0.00
0.25
0.50
0.75
baby.food.formula
baking.ingredients
bread
cereal
chips.pretzels
crackers
cream
eggs
energy.granola.bars
fresh.dips.tapenades
fresh.fruits
fresh.herbs
fresh.vegetables
frozen.meals
frozen.produce
hot.dogs.bacon.sausage
ice.cream.ice
juice.nectars
lunch.meat
milk
nuts.seeds.dried.fruit
other.creams.cheeses
packaged.cheese
packaged.vegetables.fruits
refrigerated
soft.drinks
soup.broth.bouillon
soy.lactosefree
water.seltzer.sparkling.water
yogurt
Aisles
Type
Complete dataset
Sample 400


Comparison of Proportions of Sales between Full and Sample Datasets 50
Proportion of Sales
0.00
0.25
0.50
0.75
baby.food.formula
baking.ingredients
bread
cereal
chips.pretzels
crackers
cream
eggs
energy.granola.bars
fresh.dips.tapenades
fresh.fruits
fresh.herbs
fresh.vegetables
frozen.meals
frozen.produce
hot.dogs.bacon.sausage
ice.cream.ice
juice.nectars
lunch.meat
milk
nuts.seeds.dried.fruit
other.creams.cheeses
packaged.cheese
packaged.vegetables.fruits
refrigerated
soft.drinks
soup.broth.bouillon
soy.lactosefree
water.seltzer.sparkling.water
yogurt
Aisles
Type
Complete dataset
Sample 500

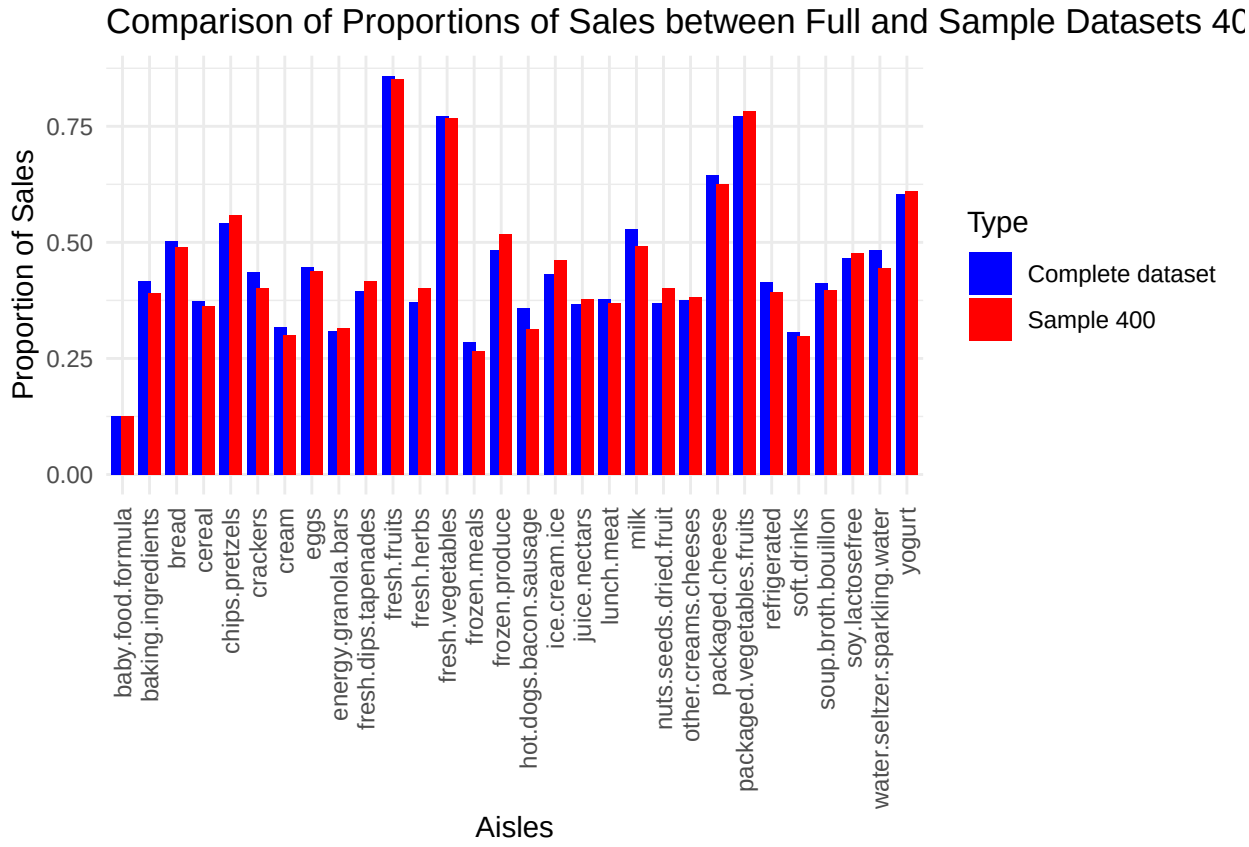

Comparison of Proportions of Sales between Full and Sample Datasets 70
Proportion of Sales
0.00
0.25
0.50
0.75
baby.food.formula
baking.ingredients
bread
cereal
chips.pretzels
crackers
cream
eggs
energy.granola.bars
fresh.dips.tapenades
fresh.fruits
fresh.herbs
fresh.vegetables
frozen.meals
frozen.produce
hot.dogs.bacon.sausage
ice.cream.ice
juice.nectars
lunch.meat
milk
nuts.seeds.dried.fruit
other.creams.cheeses
packaged.cheese
packaged.vegetables.fruits
refrigerated
soft.drinks
soup.broth.bouillon
soy.lactosefree
water.seltzer.sparkling.water
yogurt
Aisles
Type
Complete dataset
Sample 700


Comparison of Proportions of Sales between Full and Sample Datasets 10
Proportion of Sales
0.75
0.50
0.25
0.00
baby.food.formula
baking.ingredients
bread
cereal
chips.pretzels
crackers
cream
eggs
energy.granola.bars
fresh.dips.tapenades
fresh.fruits
fresh.herbs
fresh.vegetables
frozen.meals
frozen.produce
hot.dogs.bacon.sausage
ice.cream.ice
juice.nectars
lunch.meat
milk
nuts.seeds.dried.fruit
other.creams.cheeses
packaged.cheese
packaged.vegetables.fruits
refrigerated
soft.drinks
soup.broth.bouillon
soy.lactosefree
water.seltzer.sparkling.water
yogurt
Aisles
Type
Complete dataset
Sample 1000

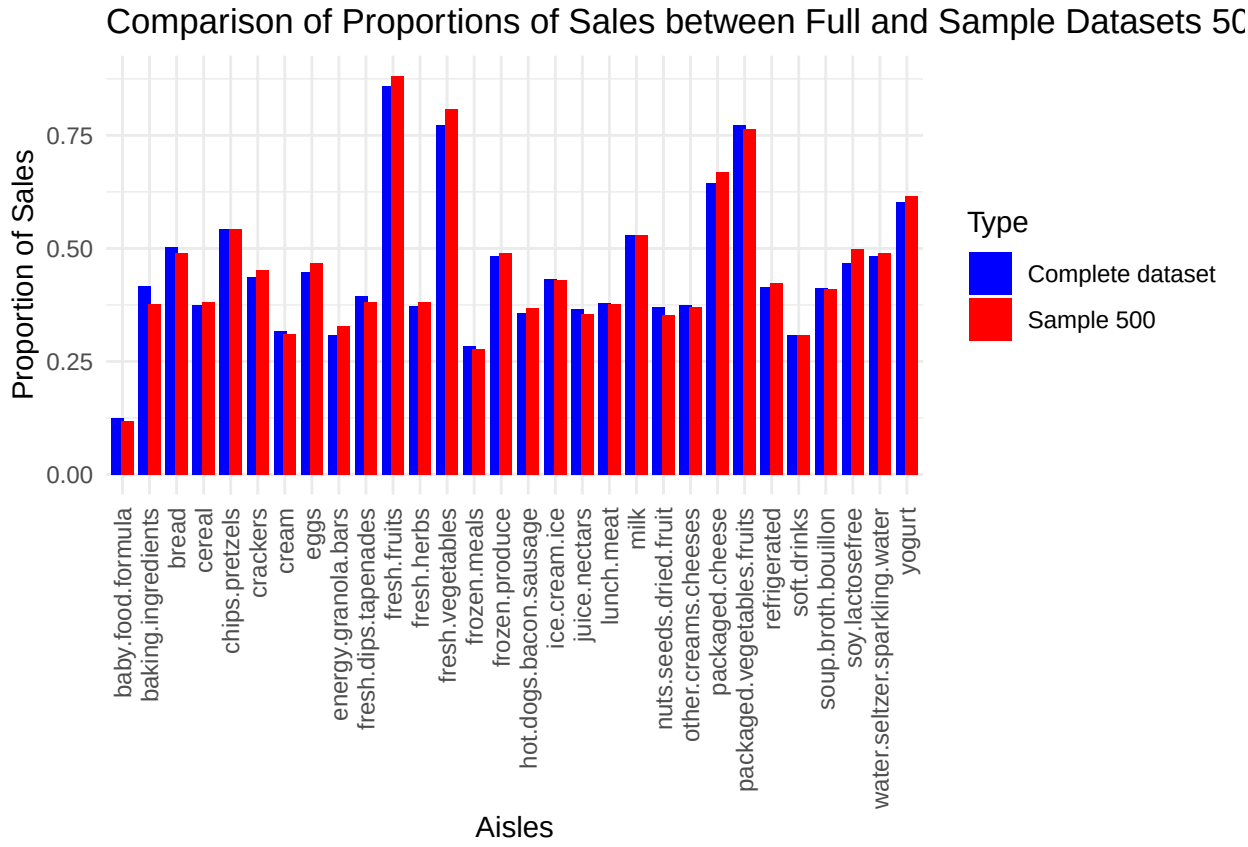
```
## $`300`
```


Comparison of Proportions of Sales between Full and Sample Datasets 30
Proportion of Sales
0.00
0.25
0.50
0.75
baby.food.formula
baking.ingredients
bread
cereal
chips.pretzels
crackers
cream
eggs
energy.granola.bars
fresh.dips.tapenades
fresh.fruits
fresh.herbs
fresh.vegetables
frozen.meals
frozen.produce
hot.dogs.bacon.sausage
ice.cream.ice
juice.nectars
lunch.meat
milk
nuts.seeds.dried.fruit
other.creams.cheeses
packaged.cheese
packaged.vegetables.fruits
refrigerated
soft.drinks
soup.broth.bouillon
soy.lactosefree
water.seltzer.sparkling.water
yogurt
Aisles
Type
Complete dataset
Sample 300

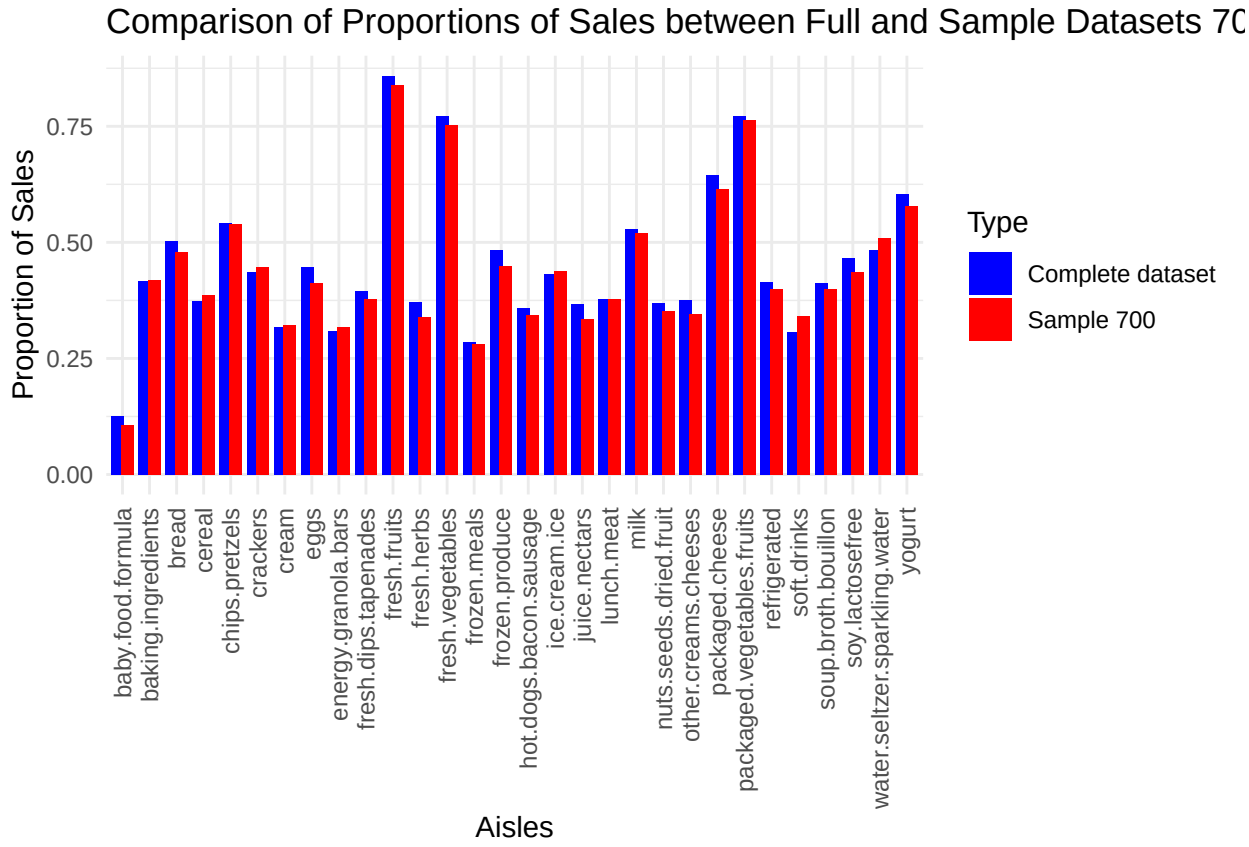
```
##
## $`400`
```


Comparison of Proportions of Sales between Full and Sample Datasets 40
Proportion of Sales
0.00
0.25
0.50
0.75
baby.food.formula
baking.ingredients
bread
cereal
chips.pretzels
crackers
cream
eggs
energy.granola.bars
fresh.dips.tapenades
fresh.fruits
fresh.herbs
fresh.vegetables
frozen.meals
frozen.produce
hot.dogs.bacon.sausage
ice.cream.ice
juice.nectars
lunch.meat
milk
nuts.seeds.dried.fruit
other.creams.cheeses
packaged.cheese
packaged.vegetables.fruits
refrigerated
soft.drinks
soup.broth.bouillon
soy.lactosefree
water.seltzer.sparkling.water
yogurt
Aisles
Type
Complete dataset
Sample 400

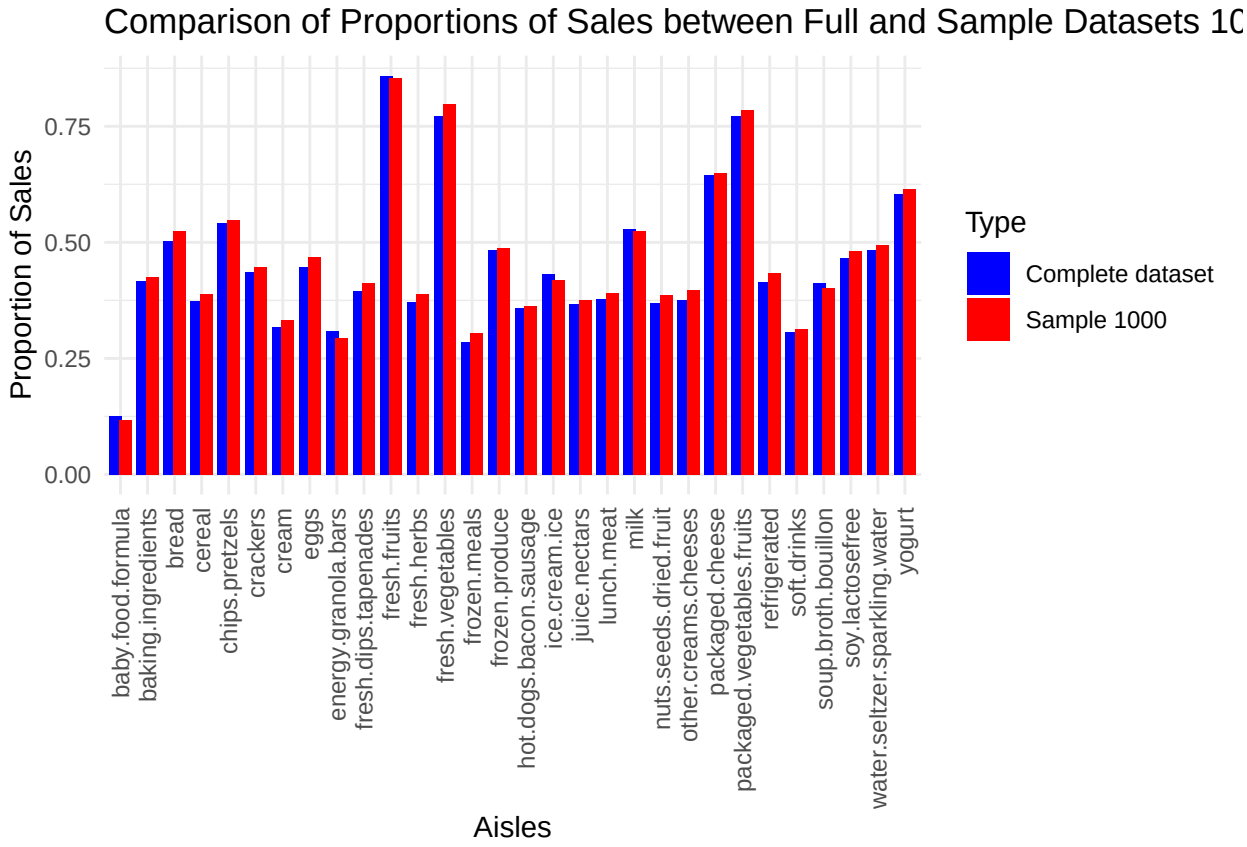
```
##
## $`500`
```


Comparison of Proportions of Sales between Full and Sample Datasets 50
Proportion of Sales
0.00
0.25
0.50
0.75
baby.food.formula
baking.ingredients
bread
cereal
chips.pretzels
crackers
cream
eggs
energy.granola.bars
fresh.dips.tapenades
fresh.fruits
fresh.herbs
fresh.vegetables
frozen.meals
frozen.produce
hot.dogs.bacon.sausage
ice.cream.ice
juice.nectars
lunch.meat
milk
nuts.seeds.dried.fruit
other.creams.cheeses
packaged.cheese
packaged.vegetables.fruits
refrigerated
soft.drinks
soup.broth.bouillon
soy.lactosefree
water.seltzer.sparkling.water
yogurt
Aisles
Type
Complete dataset
Sample 500

```
##
## $`700`
```


Comparison of Proportions of Sales between Full and Sample Datasets 70
Proportion of Sales
0.00
0.25
0.50
0.75
baby.food.formula
baking.ingredients
bread
cereal
chips.pretzels
crackers
cream
eggs
energy.granola.bars
fresh.dips.tapenades
fresh.fruits
fresh.herbs
fresh.vegetables
frozen.meals
frozen.produce
hot.dogs.bacon.sausage
ice.cream.ice
juice.nectars
lunch.meat
milk
nuts.seeds.dried.fruit
other.creams.cheeses
packaged.cheese
packaged.vegetables.fruits
refrigerated
soft.drinks
soup.broth.bouillon
soy.lactosefree
water.seltzer.sparkling.water
yogurt
Aisles
Type
Complete dataset
Sample 700

```
##
## $`1000`
```


Comparison of Proportions of Sales between Full and Sample Datasets 1(
Proportion of Sales
0.00
0.25
0.50
0.75
baby.food.formula
baking.ingredients
bread
cereal
chips.pretzels
crackers
cream
eggs
energy.granola.bars
fresh.dips.tapenades
fresh.fruits
fresh.herbs
fresh.vegetables
frozen.meals
frozen.produce
hot.dogs.bacon.sausage
ice.cream.ice
juice.nectars
lunch.meat
milk
nuts.seeds.dried.fruit
other.creams.cheeses
packaged.cheese
packaged.vegetables.fruits
refrigerated
soft.drinks
soup.broth.bouillon
soy.lactosefree
water.seltzer.sparkling.water
yogurt
Aisles
Type
Complete dataset
Sample 1000