# Product Recommendation System with Clustering and Principal Component Analysis

## Contents

# 1 Introduction

In this project we tried to create a system to predict best items to offer when someone buys an item or items. The main issue was to find solutions to a recommendation system for an e-grocery. We used data from Hunter's e-grocery from Kaggle and its contents and limitations directed the paths we have taken in the project. We made use of machine learning techniques, such as clustering and principal component analysis to make predictions. We analyzed the co-occurrence of the products in the orders and based our predictions on this analysis. We evaluated the models by applying them to the test data and comparing the predictions with the actual values in the test data.

Our baseline model used departments that the products belong to, for understanding if simply suggesting other products from the same department would be a good recommendation. We then made predictions with a simple clustering model and finally upon building on this method we applied principal component analysis before clustering to see to what extent it improves the predictions. Then we decreased the number clusters to 10 from 21, which is the number of departments in the data, to see how the predictions would change. In this final model we were able to predict 15.6 percent of the co-occurrences in the test data, which is substantially higher than the previous models, which have given results of 5, 5.5 and 8.4 percent respectively. However, it should be noted that the improvement in the final model is solely due to the reduction in the number of clusters, but it can provide a basis for future work, which would consider tuning the number of clusters and their sizes.

The project didn't only assist in developing an improved understanding with machine learning methods for the question at hand, but it also helped to build knowledge upon data limitations, what to look for in a dataset for such analysis and how to deal with the limitations for the best results.

We start with an explanation on the information about the data to be used in the project and how to retrieve it. We continue with exploratory data analysis and expand on it with the insights derived from this analysis. This is followed by more in-depth analysis with machine learning methods and the results pertaining to the outputs, predictions and evaluations of the models that have been used. We conclude by emphasizing the limitations of the work, summarizing the results and providing suggestions for future work.

## 1.1 Data

We retrieve the data from Kaggle. The dataset is from Hunter's e-grocery. We selected this dataset as it was a fairly large dataset that would allow analysis and testing of different models. The dataset is accessible through this link.

It's also included in the data folder of this project and can be loaded directly from there. The entire project can be accessed from Github.

Since Github doesn't allow large data files to be uploaded, the data file is provided in a zip file within the data folder. To correctly execute the code and to knit the Rmd the file the user needs to unzip the csv file. Only after this action, the following code will run to load the data.

```
# Load the data. The data is in the data folder in a zipper format.
# If you want to run the code, you need to unzip the file first.
# Keep the unzipped file also in the data folder.
hunters_data <-
  read.csv("data/ECommerce_consumer behaviour.csv")
```

In the project, we mostly prefer to refer to the basket contents as products instead of items to be in line with the data in context. The basket is referred to as the order. The customers are referred to a users. In any case, these terms can be used interchangeably within the text and the reader should know they refer to the same thing, unless explicitly stated otherwise.

# 2 Analysis

## 2.1 Description of the dataset

The dataset is not described in detail in the source. We will make assumptions wherever necessary, based on the column names and the data itself.

First, we have a general look at the dataset with the glimpse function to understand the structure of the data. We see that it is comprised of 2,019,501 rows and 12 columns.

```
## Rows: 2,019,501
## Columns: 12
## $ order_id               <int> 2425083, 2425083, 2425083, 2425083, 2425083, 24~
## $ user_id                <int> 49125, 49125, 49125, 49125, 49125, 49125, 49125~
## $ order_number           <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
## $ order_dow              <int> 2, 2, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3,~
## $ order_hour_of_day      <int> 18, 18, 18, 18, 18, 18, 18, 18, 18, 17, 17, 17,~
## $ days_since_prior_order <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ product_id             <int> 17, 91, 36, 83, 83, 91, 120, 59, 35, 37, 24, 83~
## $ add_to_cart_order      <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 1, 2, 3, 4, 5, 6, 7,~
## $ reordered              <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ department_id          <int> 13, 16, 16, 4, 4, 16, 16, 15, 12, 1, 4, 4, 16, ~
## $ department             <chr> "pantry", "dairy eggs", "dairy eggs", "produce"~
## $ product_name           <chr> "baking ingredients", "soy lactosefree", "butte~
```

Then we have a look at each variable for missing values and number of unique values. For each variable, the first line of the outputs shows the number of missing values and the second line shows the number of unique values. We see that there are no missing values in the dataset except for one variable, where the missing observations actually refer to a specific situation.

### 2.1.1 order_id

```
## Number of missing values: 0
```

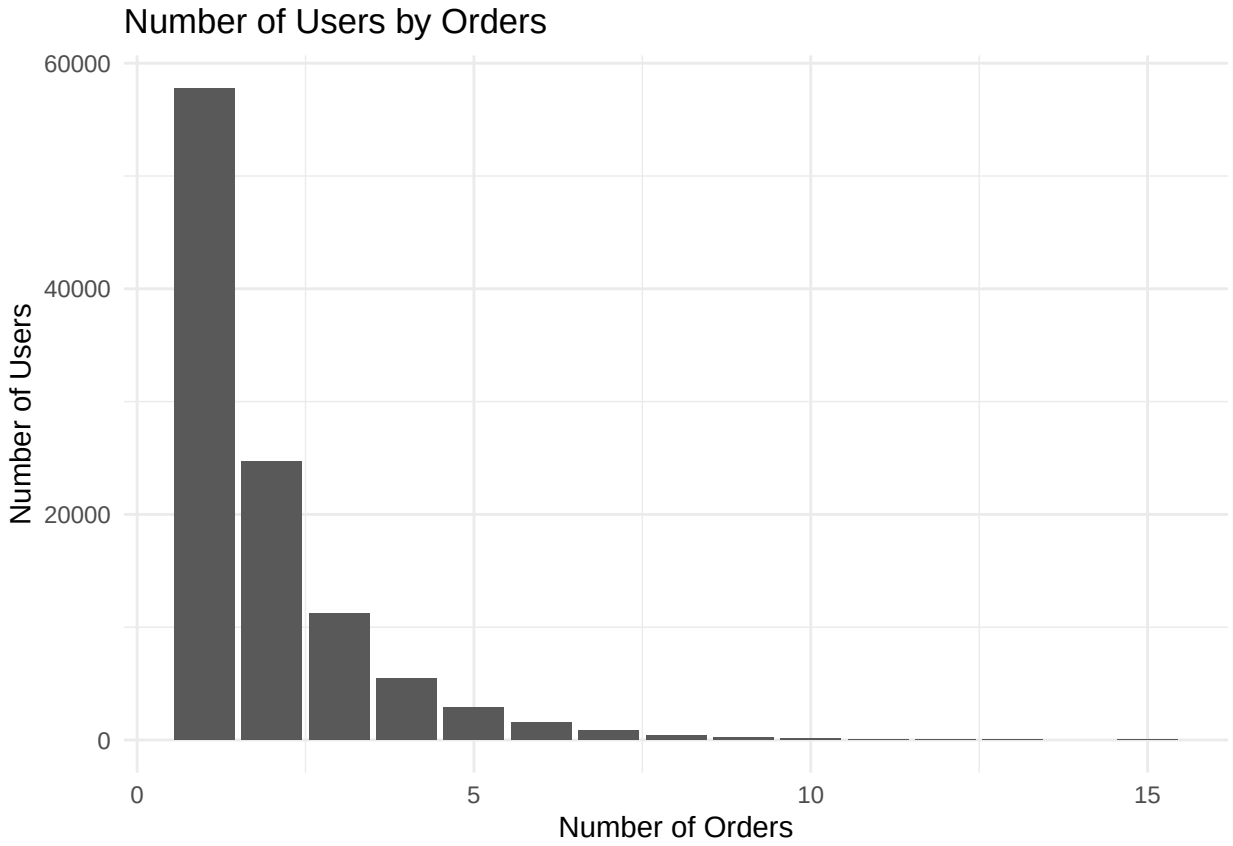
```
## Number of unique values: 200000
```

### 2.1.2 user_id

```
## Number of missing values: 0
```

```
## Number of unique values: 105273
```

There are different number of orders in the data for each user. The data includes a maximum of 15 orders for one user. A major portion of the users have only one order in the dataset. This would be limiting any temporal analysis and evaluation of the predictions from within the data, such as predicting the users' next orders and evaluating with the actual next orders. One way to deal with this limitation could be to make use of cross-validation by splitting the data into a training and test set. Another option could be to disregard one order users in the data, which would substantially reduce the size of the available data for such analysis. It would also cause loss of variability in the data.




### 2.1.3 order_number

This is explained as the number of the order made by the individual.

```
## Number of missing values: 0
```

```
## Number of unique values: 100
```

When we look at the data for a user, we see that the order numbers do not follow a sequential order and increase by 1. For example, when we select user_id = 23986 and sort by the order_number, we see that order_number 4 is followed by order_number 23.

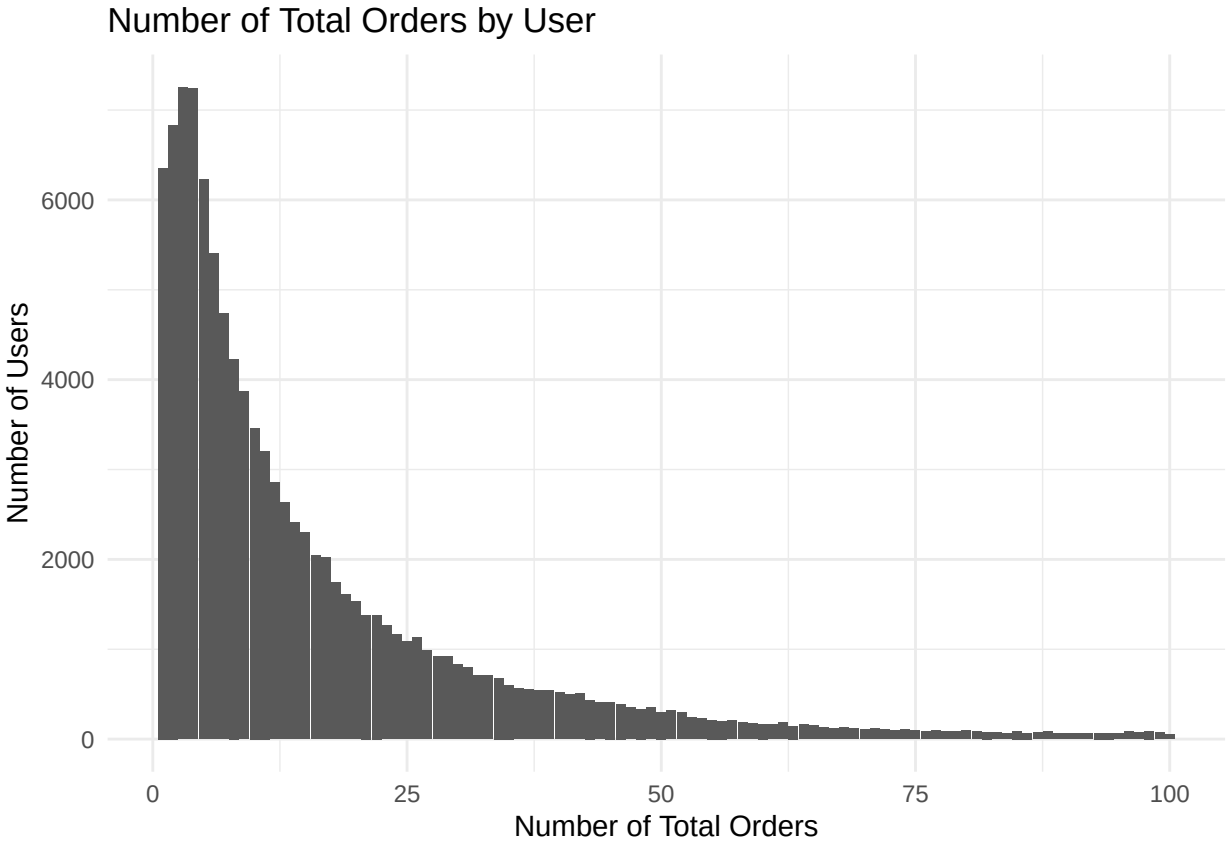
```
# Look at the data to check if the order_number is sequential for a user
hunters_data |>
  filter(user_id == 23986) |>
  arrange(order_number) |>
  select(1:4, 6) |>
  head(10)
```

```
##    order_id user_id order_number order_dow days_since_prior_order
## 1   2982740   23986            4         2                     11
## 2   2982740   23986            4         2                     11
## 3   1620643   23986           23         5                      1
## 4   1620643   23986           23         5                      1
## 5   2506971   23986           30         5                      2
## 6   2506971   23986           30         5                      2
## 7   2506971   23986           30         5                      2
## 8   2506971   23986           30         5                      2
## 9   2506971   23986           30         5                      2
```

```
## 10  2506971    23986           30         5                      2
```

This shows us that the data is not comprehensive with regards to any user and period. It's a selection of orders and it is not possible to understand the exact time of the orders, but only we can understand which order is followed by which order for a user. As it's not comprehensive and the sampling method is not clearly stated, we can mention this as a shortcoming of the predictions to be produced from this dataset.

The following plot is based on the maximum value of order_number variable for each user. It show the total orders that were made by a user. The maximum is 100, which is probably a deliberately set cut off point. There are still many users with one order observed in the plot, however it's not proportionally as high as the number of users with one order in the dataset. This is more likely to be closer to the real distribution of the orders by user.




### 2.1.4 order_dow

This variable corresponds to the day of the week that the order was placed. The variable have values from 0 to 6 and based on the discussion on the Kaggle page, we understand that 0 is Monday.

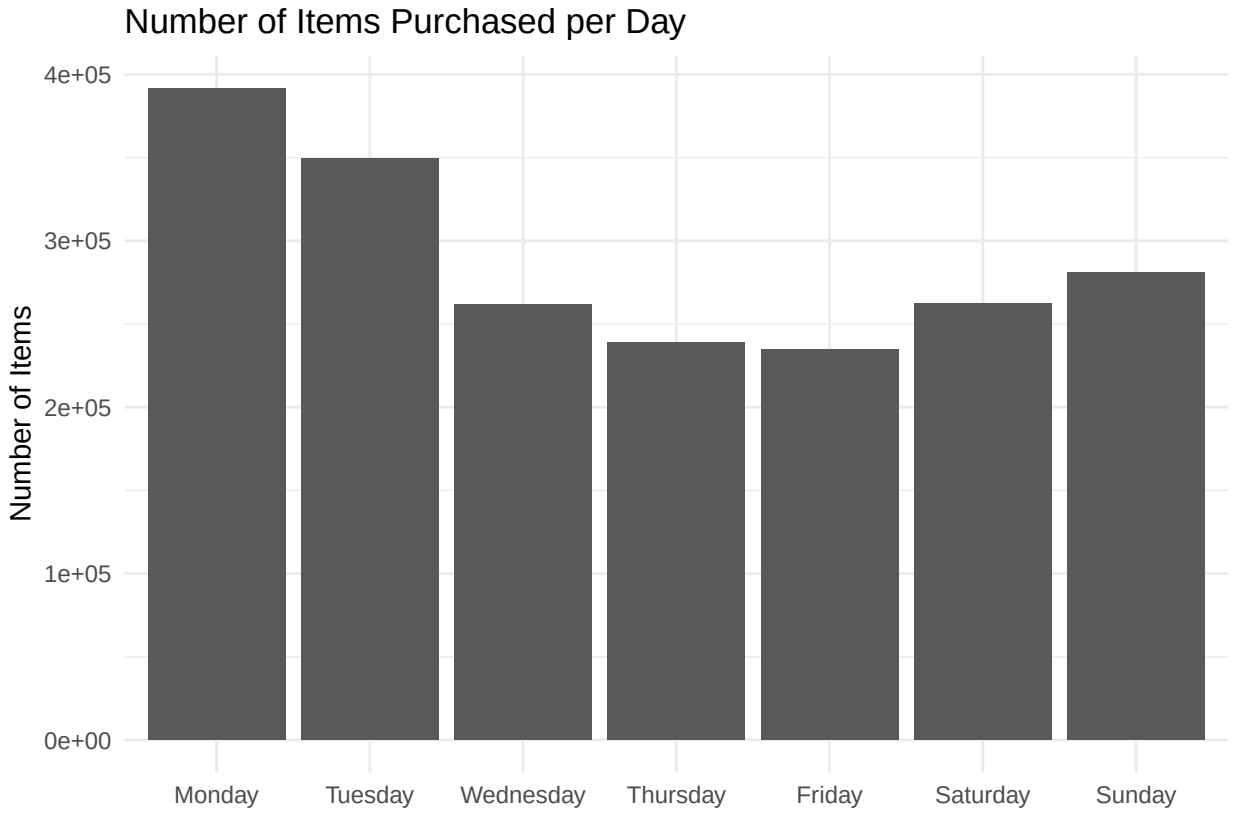
```
## Number of missing values: 0
```

```
## Number of unique values: 7
```

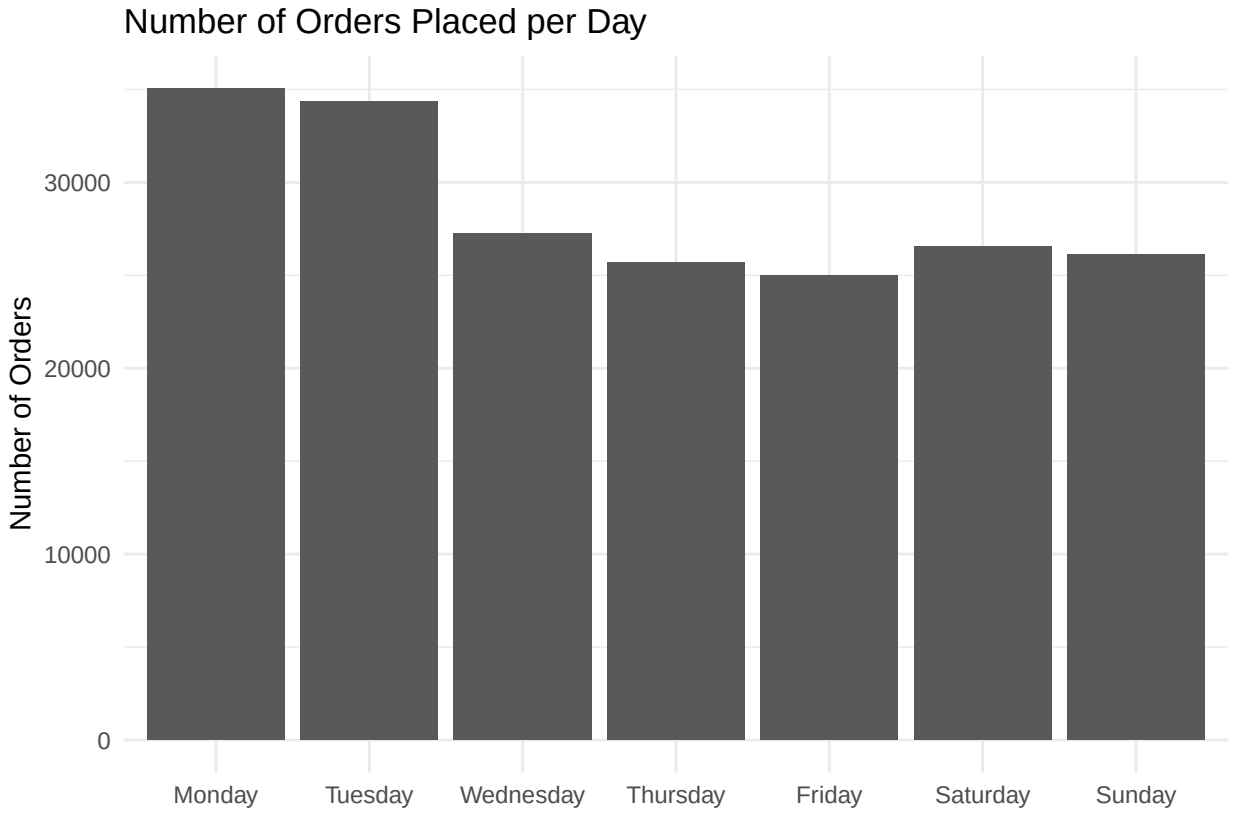
```
## [1] 0 1 2 3 4 5 6
```

We replace the numbers with the corresponding days of the week.

Then we visualize the distribution of the data by day of the week. First we do this for items, showing number of items purchased on each day of the week. We see that most items are bought on Monday, followed by Tuesday and Sunday.




Then we have a look by grouping by orders to see the number of orders placed per day. The main pattern is similar to the number of items purchased per day for Monday and Tuesday. However, Sunday is no more in the top 3 days for number of orders placed, and comes after Wednesday and Saturday. This depicts that the number of items purchased on Sunday is generally higher compared to Wednesday and Saturday.




#### 2.1.5 order_hour_of_day

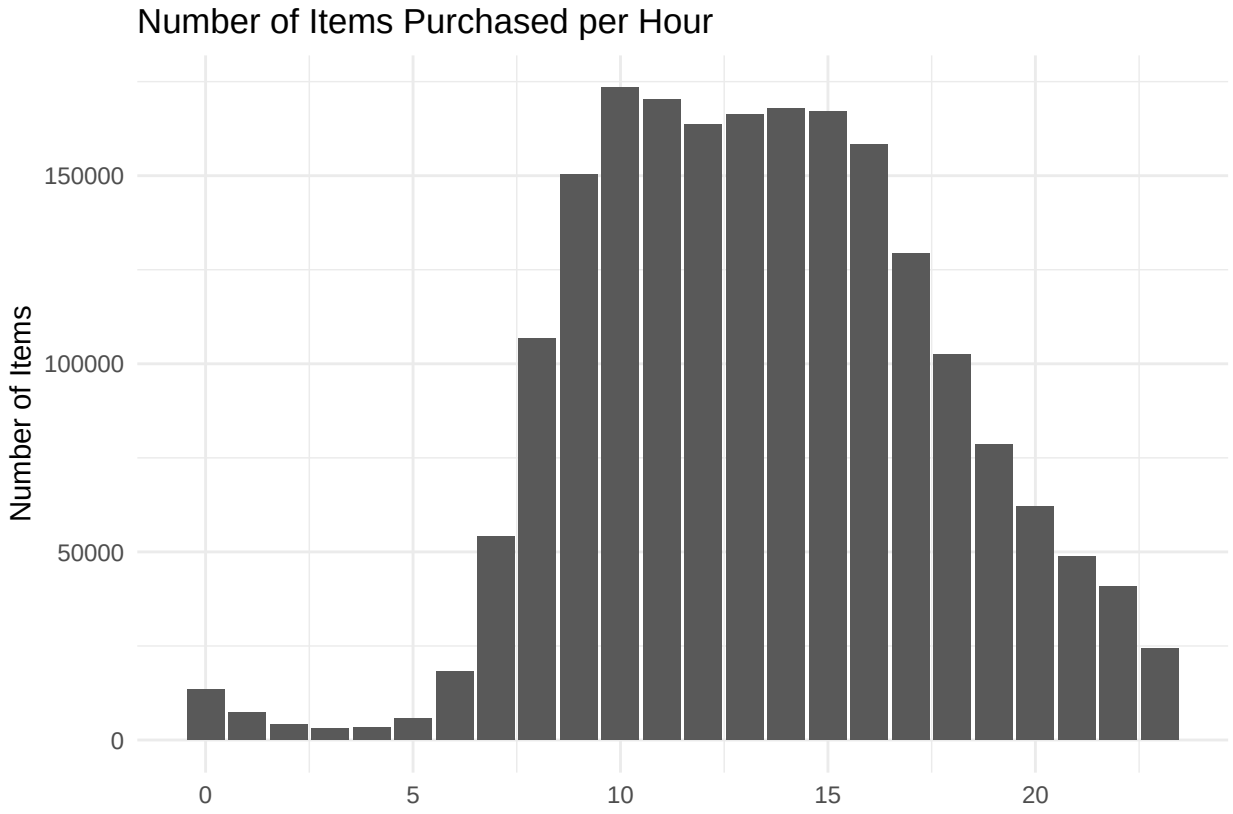
The hour when the order was placed also provides valuable information. This information also can be combined with the day of the order to understand specifics of order timing for each day.

```
## Number of missing values: 0
```

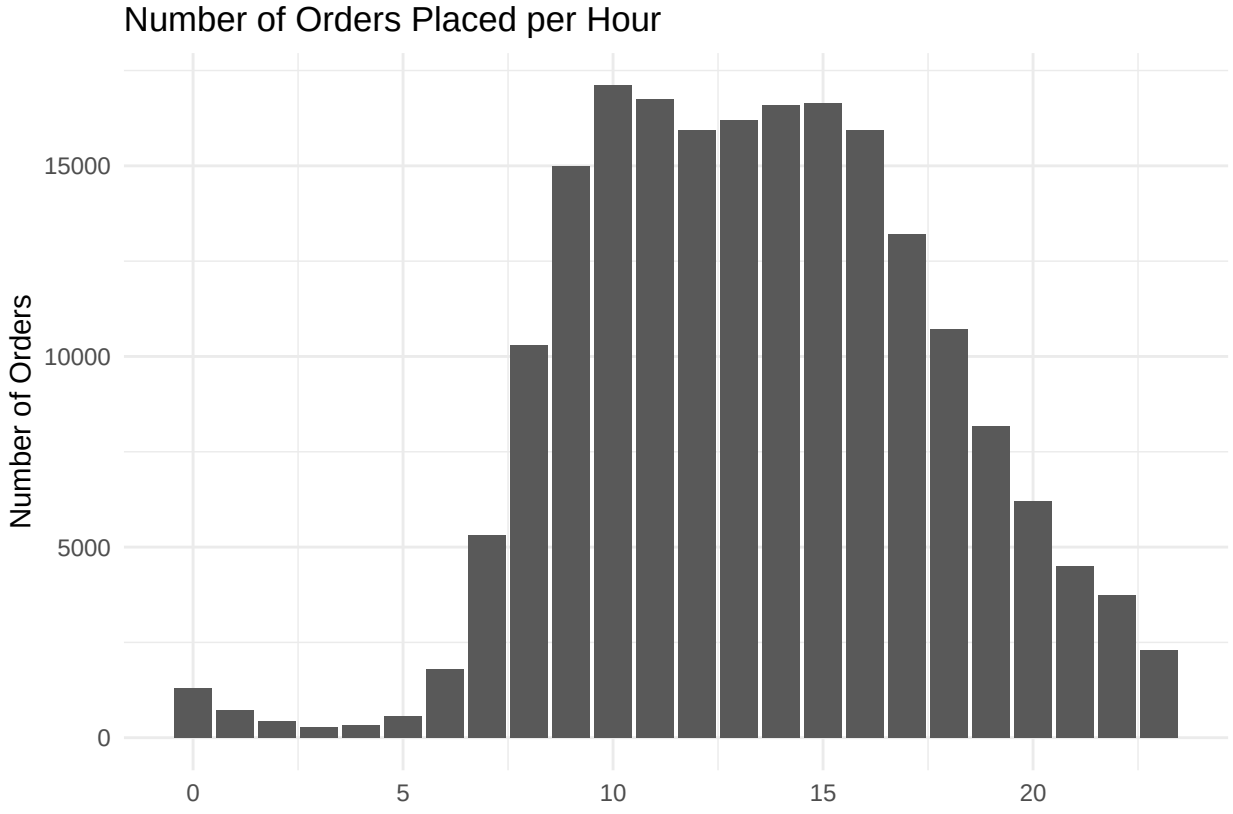
```
## Number of unique values: 24
```

```
##  [1]  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
```

Once again, we visualize the distribution of the data by hour of the day. The number of items purchased peak at 11 am, followed by 12. We see that peak hours are between 11 am and 4 pm. The hours can also be group into categories.




The pattern is slightly different when we look at the number of orders placed per hour, where the peak hour becomes 10 am.



### 2.1.6 days_since_prior_order

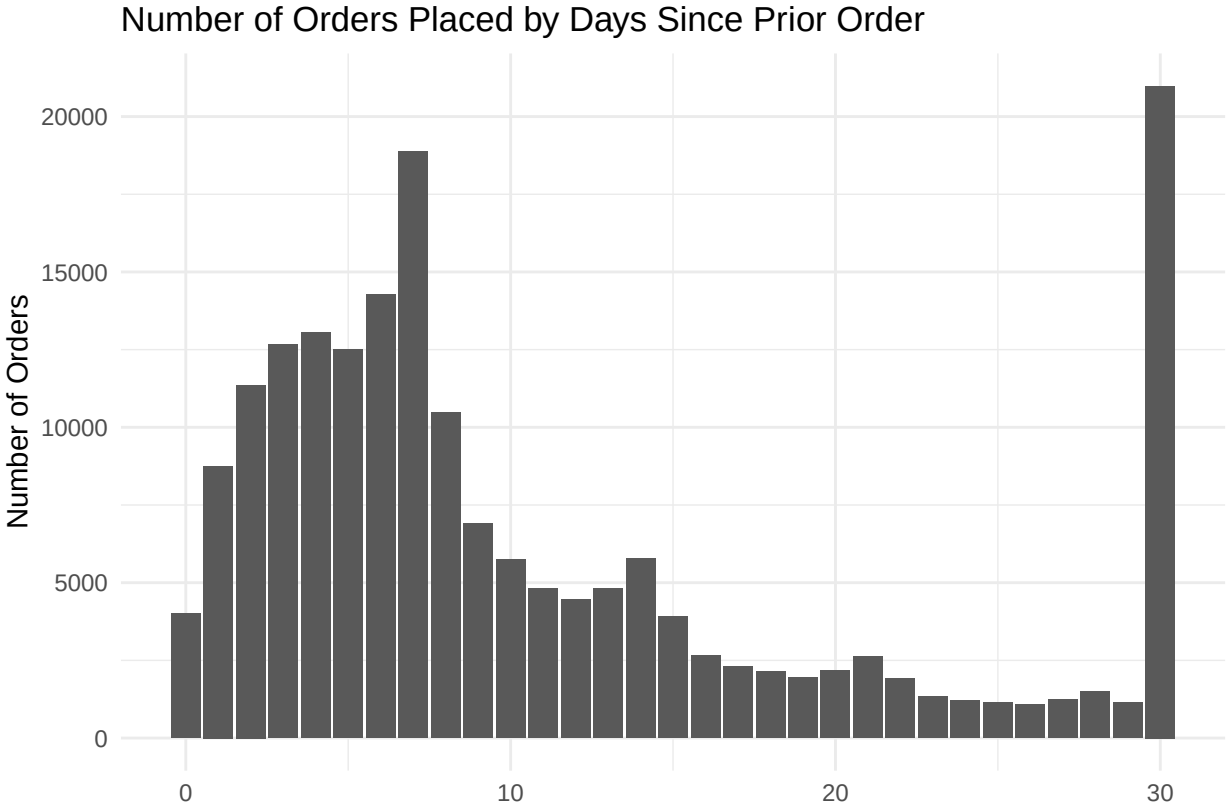
There are missing values in this variable, which would mean that it's the first order of the user. The unique values are from 0 to 30. From this we understand that the variable shows the number of days since the last order within a month.

```
## Number of missing values: 124342
```

```
## Number of unique values: 32
```

```
##  [1]  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
## [26] 25 26 27 28 29 30
```

Here we focus on the orders only, disregaring the number of items included. Based on the visual we can assume that the orders are repeated on the 30th and 7th days. It is understood that the tendency is monthly and weekly repeats in orders.

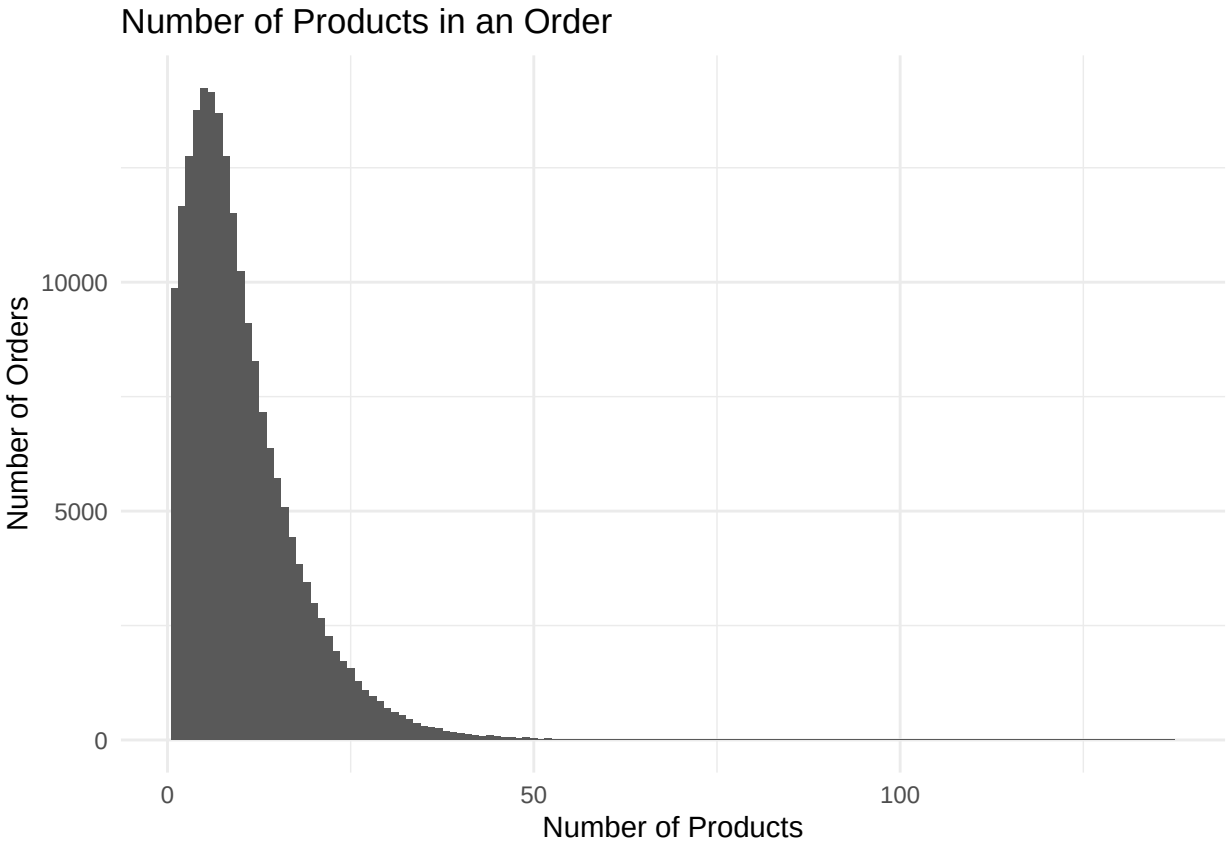



### 2.1.7 product_id

```
## Number of missing values: 0
```

```
## Number of unique values: 134
```

We look at the number of items in an order. This would give the same results with the add_to_cart_order variable, but here we go by counting the number products in an order where in the latter we only check for the maximum value of add_to_cart_order of in an order. Also in this plot we use a histogram with a bandwidth of 1 instead of a bar plot. However, as mentioned the outputs are the same.




#### 2.1.8 add_to_cart_order

This variable shows the order of the items added to the cart. There is a maximum of 137 items added to the cart. The maximum number in each order is the also the number of total items in the cart.

```
## Number of missing values: 0
```

```
## Number of unique values: 137
```

```
##   [1]   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
##  [19]  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36
##  [37]  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54
##  [55]  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72
##  [73]  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90
##  [91]  91  92  93  94  95  96  97  98  99 100 101 102 103 104 105 106 107 108
## [109] 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126
## [127] 127 128 129 130 131 132 133 134 135 136 137
```

Mostly, the number of items added to the cart is 5, followed by 6 and 4, and decreasing as the number of items added to the cart increases. It is a highly right-skewed distribution, as would be expected.


#### 2.1.9 reordered

This variable has two values, 0 and 1. It appears to show that if the value is 1, the product was ordered by this user before and if it is 0, this is the first time the product is ordered by this user. But what the first order of the user is not clear from the data, it could be referring to the first order of the product in that month. Even so, in the data there are products with the same id and name, but they have different values (some 0 and some 1) for the same user and order. So, it is not clear how this variable is generated.

```
## Number of missing values: 0
```

```
## Number of unique values: 2
```

```
## [1] 0 1
```

#### 2.1.10 department_id

There are 21 departments in the dataset.

```
## Number of missing values: 0
```

```
## Number of unique values: 21
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21
```

#### 2.1.11 department

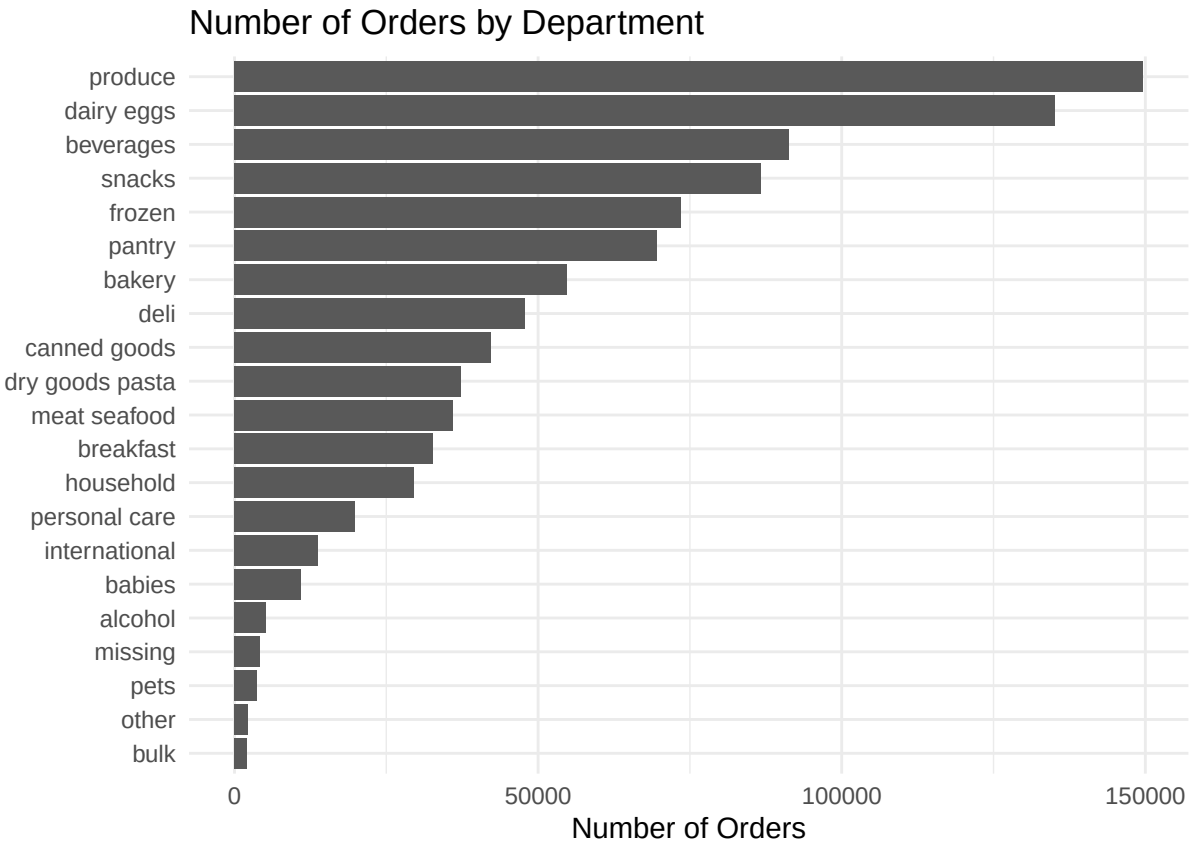
The departments are listed below.

```
## Number of missing values: 0
```

```
## Number of unique values: 21
```

```
##  [1] "alcohol"         "babies"          "bakery"          "beverages"
##  [5] "breakfast"       "bulk"            "canned goods"    "dairy eggs"
##  [9] "deli"            "dry goods pasta" "frozen"          "household"
## [13] "international"   "meat seafood"    "missing"         "other"
## [17] "pantry"          "personal care"   "pets"            "produce"
## [21] "snacks"
```

The highest number of orders are from the produce department, followed by dairy/eggs.




#### 2.1.12 product_name

Here, we list the 134 unique product names in the dataset.

```
## Number of missing values: 0
```

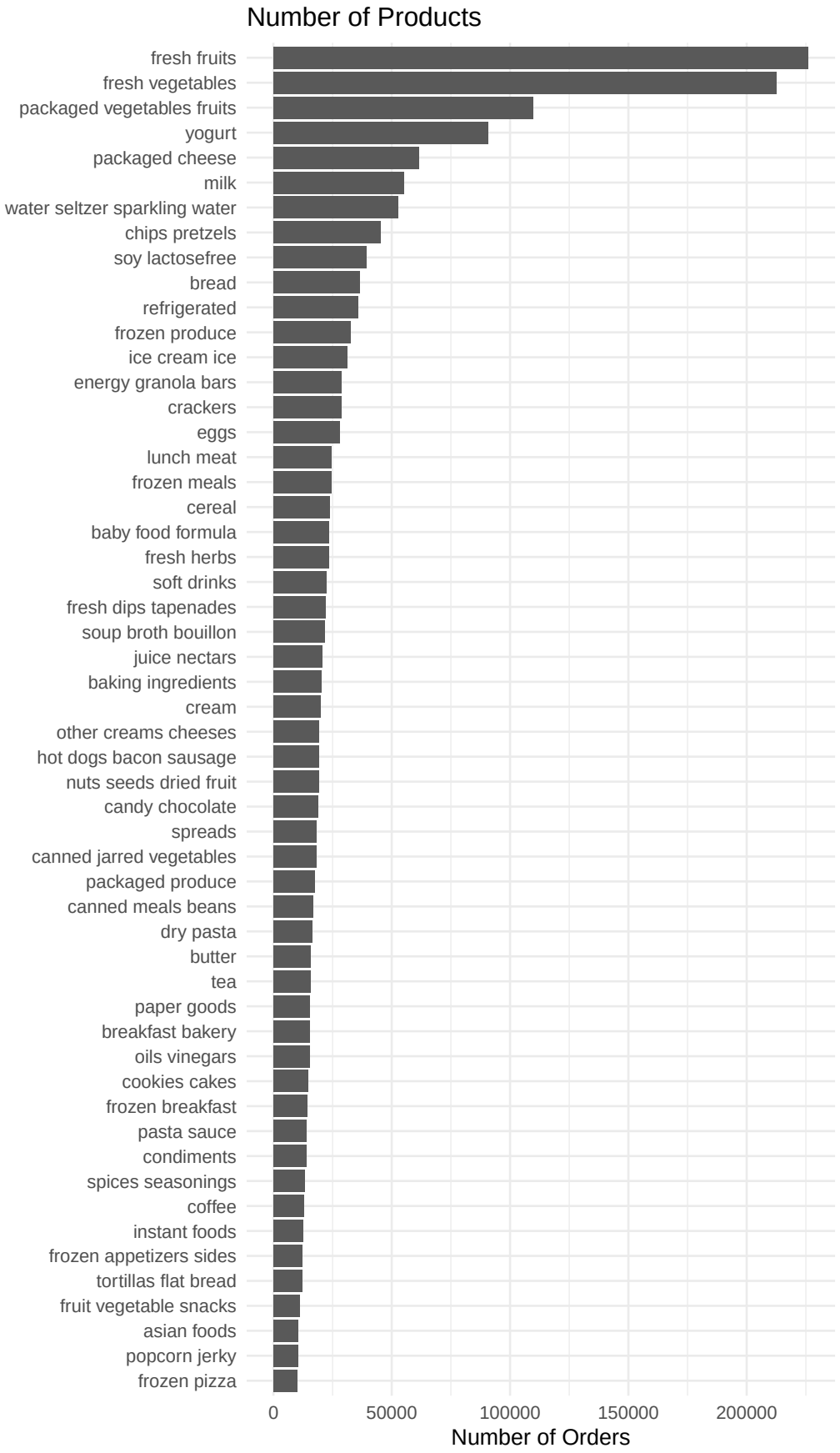
```
## Number of unique values: 134
```

```
##   [1] "air fresheners candles"        "asian foods"
##   [3] "baby accessories"              "baby bath body care"
##   [5] "baby food formula"             "bakery desserts"
##   [7] "baking ingredients"            "baking supplies decor"
##   [9] "beauty"                        "beers coolers"
##  [11] "body lotions soap"             "bread"
##  [13] "breakfast bakery"              "breakfast bars pastries"
##  [15] "bulk dried fruits vegetables"  "bulk grains rice dried goods"
##  [17] "buns rolls"                    "butter"
##  [19] "candy chocolate"               "canned fruit applesauce"
##  [21] "canned jarred vegetables"      "canned meals beans"
##  [23] "canned meat seafood"           "cat food care"
```

```
##  [25] "cereal"                          "chips pretzels"
##  [27] "cleaning products"               "cocoa drink mixes"
##  [29] "coffee"                          "cold flu allergy"
##  [31] "condiments"                      "cookies cakes"
##  [33] "crackers"                        "cream"
##  [35] "deodorants"                      "diapers wipes"
##  [37] "digestion"                       "dish detergents"
##  [39] "dog food care"                   "doughs gelatins bake mixes"
##  [41] "dry pasta"                       "eggs"
##  [43] "energy granola bars"             "energy sports drinks"
##  [45] "eye ear care"                    "facial care"
##  [47] "feminine care"                   "first aid"
##  [49] "food storage"                    "fresh dips tapenades"
##  [51] "fresh fruits"                    "fresh herbs"
##  [53] "fresh pasta"                     "fresh vegetables"
##  [55] "frozen appetizers sides"         "frozen breads doughs"
##  [57] "frozen breakfast"                "frozen dessert"
##  [59] "frozen juice"                    "frozen meals"
##  [61] "frozen meat seafood"             "frozen pizza"
##  [63] "frozen produce"                  "frozen vegan vegetarian"
##  [65] "fruit vegetable snacks"          "grains rice dried goods"
##  [67] "granola"                         "hair care"
##  [69] "honeys syrups nectars"           "hot cereal pancake mixes"
##  [71] "hot dogs bacon sausage"          "ice cream ice"
##  [73] "ice cream toppings"              "indian foods"
##  [75] "instant foods"                   "juice nectars"
##  [77] "kitchen supplies"                "kosher foods"
##  [79] "latino foods"                    "laundry"
##  [81] "lunch meat"                      "marinades meat preparation"
##  [83] "meat counter"                    "milk"
##  [85] "mint gum"                        "missing"
##  [87] "more household"                  "muscles joints pain relief"
##  [89] "nuts seeds dried fruit"          "oils vinegars"
##  [91] "oral hygiene"                    "other"
##  [93] "other creams cheeses"            "packaged cheese"
##  [95] "packaged meat"                   "packaged poultry"
##  [97] "packaged produce"                "packaged seafood"
##  [99] "packaged vegetables fruits"      "paper goods"
## [101] "pasta sauce"                     "pickled goods olives"
## [103] "plates bowls cups flatware"      "popcorn jerky"
## [105] "poultry counter"                 "prepared meals"
## [107] "prepared soups salads"           "preserved dips spreads"
## [109] "protein meal replacements"       "red wines"
## [111] "refrigerated"                    "refrigerated pudding desserts"
## [113] "salad dressing toppings"         "seafood counter"
## [115] "shave needs"                     "skin care"
## [117] "soap"                            "soft drinks"
## [119] "soup broth bouillon"             "soy lactosefree"
## [121] "specialty cheeses"               "specialty wines champagnes"
## [123] "spices seasonings"               "spirits"
## [125] "spreads"                         "tea"
## [127] "tofu meat alternatives"          "tortillas flat bread"
## [129] "trail mix snack mix"             "trash bags liners"
## [131] "vitamins supplements"            "water seltzer sparkling water"
```

```
## [133] "white wines"                    "yogurt"
```

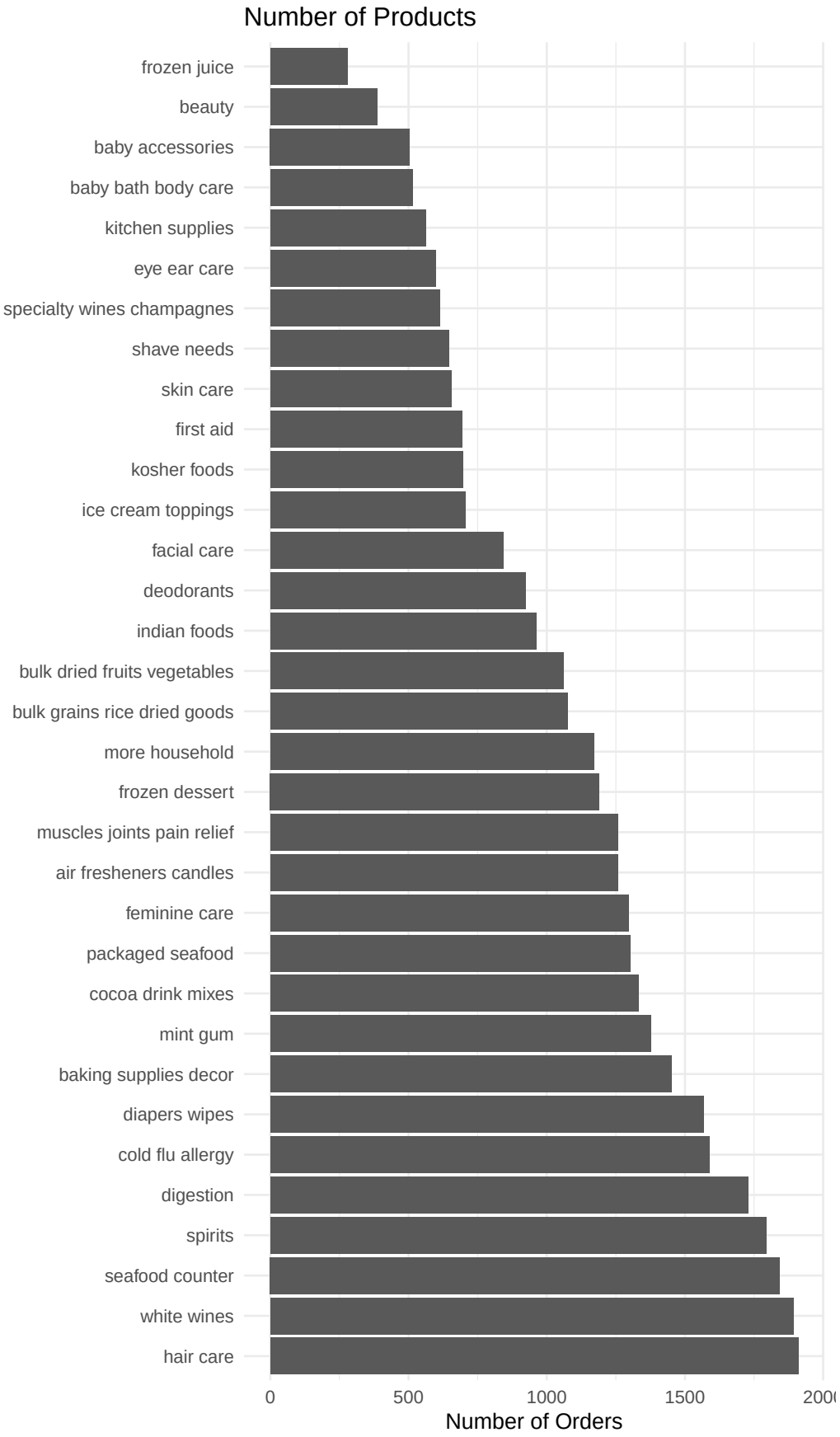
Fresh fruits and fresh vegetables are the most ordered products followed by packaged vegetables/fruits, yogurt and milk. In this plot we take a count of the products regardless of the order, meaning that a product can appear multiple times in one order. However we are interested in the preferences in general first. As there are more than 100 products, we filter out the products that have been ordered less than 10,000 for better visualisation of the remaining products.


Number of Products
fresh fruits
fresh vegetables
packaged vegetables fruits
yogurt
packaged cheese
milk
water seltzer sparkling water
chips pretzels
soy lactosefree
bread
refrigerated
frozen produce
ice cream ice
energy granola bars
crackers
eggs
lunch meat
frozen meals
cereal
baby food formula
fresh herbs
soft drinks
fresh dips tapenades
soup broth bouillon
juice nectars
baking ingredients
cream
other creams cheeses
hot dogs bacon sausage
nuts seeds dried fruit
candy chocolate
spreads
canned jarred vegetables
packaged produce
canned meals beans
dry pasta
butter
tea
paper goods
breakfast bakery
oils vinegars
cookies cakes
frozen breakfast
pasta sauce
condiments
spices seasonings
coffee
instant foods
frozen appetizers sides
tortillas flat bread
fruit vegetable snacks
asian foods
popcorn jerky
frozen pizza
0
50000
100000
150000
200000
Number of Orders

From the categories of this variable, we also understand that at least some of the products actually refer to a more general category, such as fresh fruits, where there are actually multiple products under the category of which the details are not provided in the dataset. This is also a limitation of the dataset, when this is especially considered together with the reordered variable, showing that some of the subcategories are reordered and some are not, but we don't have an insight on the specifics of these products.

It should also be known that some products are consumed less frequently and therefore are not ordered as much as others. One would expect food products to be ordered in most of the orders. Also as expected there are multiple types of the same product because food products are available in various types.

The plot provided us most of the frequently ordered products, however it would be useful to least frequently ordered products as well. For this puspose we plot products that are ordered less than 2,000 times in the dataset. It's seen that the products are mostly utensils and long-lasting products. There are also some food products, which are very specific.


Number of Products
frozen juice
beauty
baby accessories
baby bath body care
kitchen supplies
eye ear care
specialty wines champagnes
shave needs
skin care
first aid
kosher foods
ice cream toppings
facial care
deodorants
indian foods
bulk dried fruits vegetables
bulk grains rice dried goods
more household
frozen dessert
muscles joints pain relief
air fresheners candles
feminine care
packaged seafood
cocoa drink mixes
mint gum
baking supplies decor
diapers wipes
cold flu allergy
digestion
spirits
seafood counter
white wines
hair care
0
500
1000
1500
Number of Orders

## 2.2 Insights from the exploratory data analysis

So far, we made the basic exploratory data analysis and gained valuable insights from the dataset.

The data doesn't clarify how it was sampled from its source and the information on the source and the dataset itself are not clearly stated.

The data is structured as such that the orders of users (customers) are registered consisting of the products included in the cart (basket). We have the names of the products along with the department they belong to. We know have many products are there in each order and the order of the products in the cart, so that we can see which product was followed by which product. We also know if the product was reordered by the user or not, although it's not clear how this variable is generated.

We also have temporal information, as the days and the hours of the orders, along with the days since the prior order. These can be providing some patterns with regards to users (customers) and products.

We understood from the data that within one order there can be multiple products of the same category.

- A user gives an order.
- An order includes products in it.
- The products are related to a department.
- The products within an order may repeat, but this is because the subcategories of the products are not provided.
- A product is reordered or not.

Temporal characteristics:

- The order is placed on a day of the week.
- The order is placed on an hour of the day.
- The order is placed after a certain number of days since the prior order.
- A product is added to the cart in a certain order.
- The data does not have a timestamp for the orders.
- Most of the users have only one order in the dataset.

## 2.3 Models

There are many approaches that can be taken to make the predictions for the intended recommendation system. We will rely on the availability of the products in the orders mainly. We will model the co-occurrence of the products in the orders and evaluate the validity of the model with the test data. The question to be answered by this analysis can be summarized as:

"If there is one product in the order of the user, how likely is it that the other product from a group (cluster) of product suggestions will be in the same order?"

If the likelihood is high, then the two products are likely to be purchased together and a recommendation system can be built based on this information. We will create clusters of products and when there is one product in the order we will be suggesting the other products in the same cluster. As the products may repeat in the orders, we will consider only distinct cases. By taking distinct values of products in each order, we are ignoring the purchase of multiple products of the same kind in one order. The repeating products are mostly higher level categories of products in essence, as the subcategories are not provided in the dataset. This will be a limitation in the analysis.

The available data will also pose limitations in this approach as there are many orders with only one product in it. This will be a restricting factor. Another shortcoming in our approach is that we suggest products from a cluster, meaning that there will be many suggestions for one product in some cases.

As a baseline we will use the department information as a basis for the clusters and suggest products from the same department. This itself would be expected to provide a good prediction and we will see if other models will provide better predictions.

We will use two main methods to model and make predictions. The first will be the simple clustering method. The second will follow a path of dimension reduction with principal component analysis (PCA) before the clustering operation, which will allow us to see the effectiveness of the dimension reduction in the predictions.

Other variables in the data set can make significant contributions to our analysis, however we will not employ them in this project to avoid further complexity. We will look into the co-occurrence of the products in the orders, also disregarding any temporal information.

We will evaluate the validity of the models by creating the model on the training data and testing it by applying the models to the test data and comparing the predictions with the actual values in the test data.

### 2.3.1 Training and Test Data

We start by splitting the data into training and test data. We will use the training data to create the models and the test data to evaluate the models. We will use the caret package for this purpose. The training data consist of 90 percent of the data and the test data will consist of 10 percent of the data.

### 2.3.2 Model with Departments

We start with the model where we make our suggestions based on the departments of the products. When a product is in the order we will suggest any other product from the same department. We will evaluate the model with the test data.

We generate a vector of the products in the orders.

We would like to cluster the products to check their co-occurrence in the orders. By taking distinct values of products in each order, we are ignoring the purchase of multiple products of the same kind in one order. In corrrespondence with the departments, we created 21 clusters.

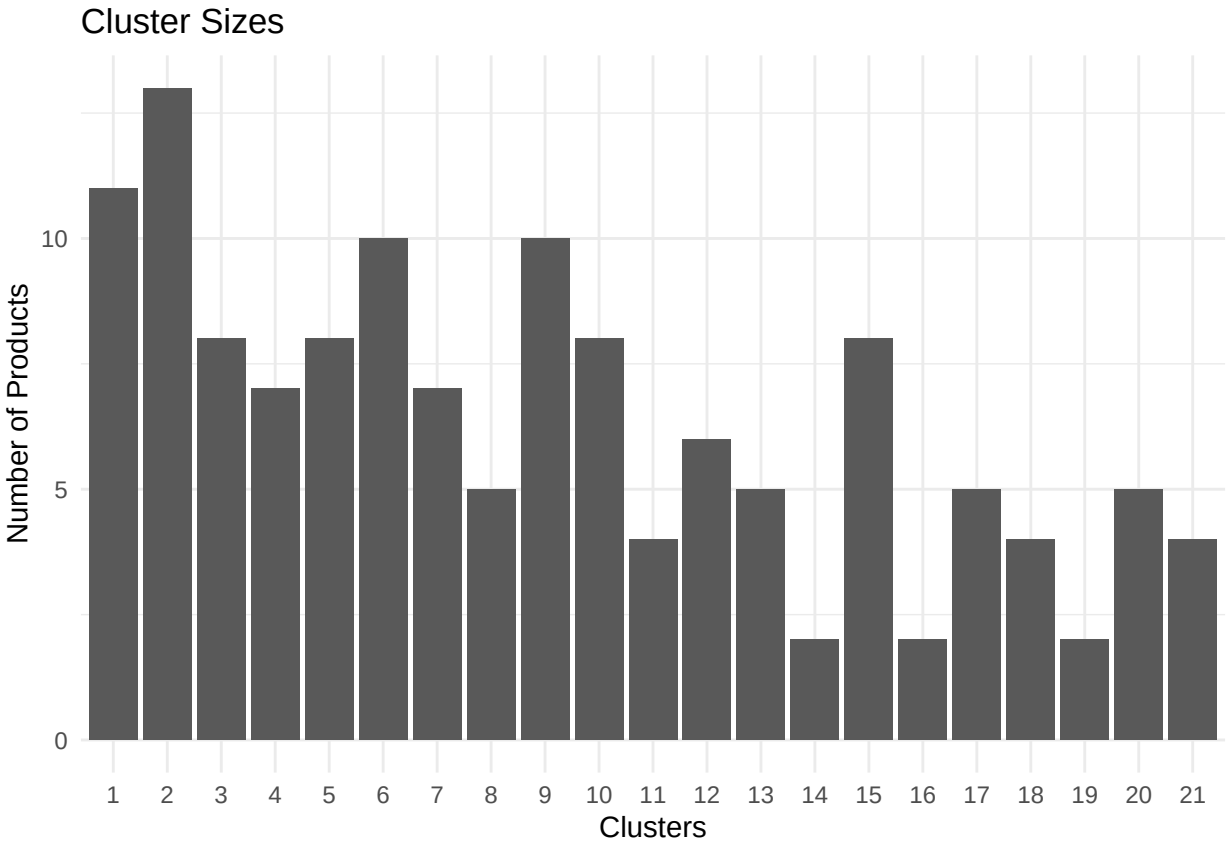
**2.3.2.1 Evaluation** We evaluate the model with the test data. The baseline model gives a value of 0.04995, meaning that 5 percent of the co-occurrences between two products in the test data are aligned with the clusters. Only by suggesting products from the same department, we can predict 5 percent of the co-occurrences in the test data.

```
## Proportion of test co-occurrences aligned with clusters: 0.04995302
```

### 2.3.3 Model with Simple Clustering

Then we will do clustering of to detect commonly ordered products together. We will use Jaccard distance to measure the similarity between two products. We would like to cluster the products to check their co-occurrence in the orders. We create the model for the training data.

We create a 21 cluster division for the products to make it comparable to the departments and see the number of products in each cluster. We provide a plot for the cluster sizes.




**2.3.3.1 Evaluation** We finally evaluate the model with the test data. We use 21 clusters for our predictions. The alignment score is 0.548, indicating a 5.5 percent match. The clustering technique provided a slight improvement over only using the departments.

```
## Proportion of test co-occurrences aligned with clusters: 0.05480739
```

### 2.3.4 Model with Principal Component Analysis

Now we would like to see the effect of dimension reduction in forming the clusters. We use PCA method for this application. With PCA a more effective clustering was intended to be achieved. The data was reduced to 90 percent of the variance. The reduced data was then used for k-means clustering, dividing the products into 21 clusters.

**2.3.4.1 Evaluation** Using PCA before clustering in a model with 21 clusters, improved the evaluation score to 0.08362, imdicating that 8.4 percent of the co-occurrences in the test data are aligned with the clusters, which is well over the simple clustering model.

```
## Proportion of test co-occurrences aligned with PCA clusters: 0.08362042
```

### 2.3.5 Model with 10 Clusters

Then we limit the number of clusters to 10, meaning that we will be suggesting more products at once. It is obvious that as we decrease the number of clusters, we will be getting higher co-occurences. Although it might not be ideal for a suggestion system to provide too many options, we would like to see how the co-occurrence would change. In another study the clusters can be set to an ideal size and the models can be generated by utilizing other methods to make it more effective. Or the optimal cluster size can be looked for depending on the targeted evaluation score.

In this case, the evaluation score indicates that the likelihood increases to 0.1555, with 10 clusters. With this model we are able to predict 15.6 percent of the co-occurrences in the test data.

```
## Proportion of test co-occurrences aligned with PCA clusters: 0.1554964
```

# 3 Results

Creating a recommendation system for a customer is not straightforward, but we have utilized two models and evaluated their validity. The models can be improved and developed further.

With the data at hand we found that simply clustering is better than using the departments for the predictions. The simple clustering model provided a 5.5 percent alignment score, which was over the 5 percent alignment score of the department model. The PCA model provided better results with 21 clusters, with an alignment score of 8.8 percent. The model with 10 clusters provided an alignment score of 15.6 percent. The improvement with smaller number of clsuters don't indicate an improvement in the prediction, but rather an increase in the likelihood of the co-occurrences. In another study, the ideal number of clusters can also be looked for, depending on the targeted evaluation score.

| **Model** | **Alignment Score** |
|---|---|
| Model with Departments as Clusters | 0.04995 |
| Model with Simple Clustering | 0.0548 |
| Model with PCA, 21 Clusters | 0.0836 |
| Model with PCA, 10 Clusters | 0.1555 |

# 4 Conclusion

In this project we have analysed the dataset from Hunter's e-grocery in detail and using various methods tried to predict the best items to offer when a customer buys an item or items. The project allowed us to explore methods to use for such predictions.

There were some limitations to the data at hand, which prevented better, more accurate and reliable predictions. First limitation of the project was regarding the metadata of the dataset, which was not provided in detail. This required making assumptions where it was not clear what the variables represented.

The dataset is not comprehensive for a period of time or a user, and the sampling method is not clearly stated. This is a major shortcoming that would affect any insights from the dataset, as the sampling method would definitely be affective on the outputs. In relation to this, it's also not possible to make accurate temporal analysis, as no exact timestamp is provided for the orders, except for generalized timing information. Comprehensivity of the data is important because without a comprehesive data the found patterns would be biased and misleading. One can easily see that some products are normally purchased less frequently and when the data is not comprehensive, it's not possible to detect the average frequency for the purchase of these products. In addition, the lack of information on the amount purchased, its price and the demographics of the customers highly limit the accurary of the predictions.

There is only one order for pertaining to some users in the dataset. Another limitation was that not all orders had more than one item, as such orders were not useful for the predictions.

In short, lack of comprehensivity of the dataset and lack of many potential covariates that could be effective on the predictions are the main limitations for such analysis and predictions.

Baring these limitations in mind, we tried to exploit the data at hand to make predictions. We started with an exploratory data analysis, where we looked at each variable in detail. This allowed us to gain insights from the data at hand, its above-mentioned limitations for analysis and potential inferences that can be made from the data.

We used a clustering model and PCA. Our models improved with clustering and then with PCA. Decreasing the number of clusters to 10 enabled to predict 15.6 percent of the co-occurrences in the test data. The final model with smaller number of clusters doesn't mean an improvement in the model, but it shows the improvement in the likelihood of the co-occurrences. This can provide a basis for further studies to find the optimal number of clusters for the predictions.

We didn't analyze the effect of different clustering methods. And our evaluation metric was a simple one, not taking into consideration any ranking of the order products. The model and the evaluation metric can be improved by taking into consideration by predicting the next product that would be added to the order.

As future work, first a larger dataset would be useful. It should also be comprehensive to grasp the patterns correctly. If it's sampled from a data, the sampling method should be valid and well reported.

The data does not include any information on the demographics of the customers. Such information would be very useful to undestand the patterns for different groups of customers and make suggestions accordingly. An ideal data for such analysis should include information on the subcategories of the products, timestamp of the orders, amount purchased and its price. A more clearly defined data would definitely help in making more accurate predictions.

# 5 References


- Irizzary, R. A., 2019, "Introduction to Data Science", https://leanpub.com/datasciencebook.
- Supermarket dataset for predictive marketing 2023, https://www.kaggle.com/datasets/hunter0007/ecommerce-dataset-for-predictive-marketing-2023/data.