

Unsupervised Learning

by Crystal (Yunan) Zhu, Anupama r.k, Queenie Tsang

18/02/2021

```
library(tidyverse)
library("readxl")
```

```
#get the names of the columns
nms <- names(read_excel("US_Superstore_data.xls"))

#if the column name has "Date" in it, read the column as date data type, otherwise guess the type
ct <- ifelse(grepl("^Date", nms), "date", "guess")
data <- read_excel("US_Superstore_data.xls", col_types = ct)
```

To look at some basic statistics for this dataset:

```
summary(data)
```

```
##      Row ID      Order ID      Order Date
## Min.      : 1      Length:9994      Min.      :2014-01-03 00:00:00
## 1st Qu.:2499      Class :character  1st Qu.:2015-05-23 00:00:00
## Median :4998      Mode  :character  Median :2016-06-26 00:00:00
## Mean    :4998                                Mean    :2016-04-30 00:07:12
## 3rd Qu.:7496                                3rd Qu.:2017-05-14 00:00:00
## Max.     :9994                                Max.     :2017-12-30 00:00:00
##      Ship Date      Ship Mode      Customer ID
## Min.      :2014-01-07 00:00:00      Length:9994      Length:9994
## 1st Qu.:2015-05-27 00:00:00      Class :character  Class :character
## Median :2016-06-29 00:00:00      Mode  :character  Mode  :character
## Mean     :2016-05-03 23:06:58
## 3rd Qu.:2017-05-18 00:00:00
## Max.     :2018-01-05 00:00:00
## Customer Name      Segment      Country      City
## Length:9994      Length:9994      Length:9994      Length:9994
## Class :character  Class :character  Class :character  Class :character
## Mode  :character  Mode  :character  Mode  :character  Mode  :character
##
##
##
##      State      Postal Code      Region      Product ID
## Length:9994      Min.      : 1040      Length:9994      Length:9994
## Class :character  1st Qu.:23223      Class :character  Class :character
## Mode  :character  Median :56431      Mode  :character  Mode  :character
##
##      Mean      :55190
```

```
##           3rd Qu.:90008
##           Max.      :99301
##   Category   Sub-Category   Product Name       Sales
## Length:9994   Length:9994   Length:9994       Min.    :    0.444
## Class :character Class :character Class :character 1st Qu.:   17.280
## Mode  :character Mode  :character Mode  :character Median :    54.490
##                                           Mean  :   229.858
##                                           3rd Qu.:  209.940
##                                           Max.   :22638.480
##   Quantity      Discount      Profit
## Min.    : 1.00   Min.    :0.0000   Min.    : -6599.978
## 1st Qu.: 2.00   1st Qu.:0.0000   1st Qu.:    1.729
## Median : 3.00   Median :0.2000   Median :    8.666
## Mean   : 3.79   Mean   :0.1562   Mean   :   28.657
## 3rd Qu.: 5.00   3rd Qu.:0.2000   3rd Qu.:   29.364
## Max.   :14.00   Max.   :0.8000   Max.   :  8399.976
```

To look at the dimensions of the data:

```
dim(data)
```

```
## [1] 9994  21
```

The dimensions of the dataset are 9994 by 21.

Check that the Order Date and Ship Date column type is POSIXct which is a date data type:

```
data
```

```
## # A tibble: 9,994 x 21
##   'Row ID' 'Order ID' 'Order Date'      'Ship Date'      'Ship Mode'
##   <dbl> <chr>      <dtm>      <dtm>      <chr>
## 1      1 CA-2016-1~ 2016-11-08 00:00:00 2016-11-11 00:00:00 Second Cla~
## 2      2 CA-2016-1~ 2016-11-08 00:00:00 2016-11-11 00:00:00 Second Cla~
## 3      3 CA-2016-1~ 2016-06-12 00:00:00 2016-06-16 00:00:00 Second Cla~
## 4      4 US-2015-1~ 2015-10-11 00:00:00 2015-10-18 00:00:00 Standard C~
## 5      5 US-2015-1~ 2015-10-11 00:00:00 2015-10-18 00:00:00 Standard C~
## 6      6 CA-2014-1~ 2014-06-09 00:00:00 2014-06-14 00:00:00 Standard C~
## 7      7 CA-2014-1~ 2014-06-09 00:00:00 2014-06-14 00:00:00 Standard C~
## 8      8 CA-2014-1~ 2014-06-09 00:00:00 2014-06-14 00:00:00 Standard C~
## 9      9 CA-2014-1~ 2014-06-09 00:00:00 2014-06-14 00:00:00 Standard C~
## 10    10 CA-2014-1~ 2014-06-09 00:00:00 2014-06-14 00:00:00 Standard C~
## # ... with 9,984 more rows, and 16 more variables: 'Customer ID' <chr>,
## #   'Customer Name' <chr>, Segment <chr>, Country <chr>, City <chr>,
## #   State <chr>, 'Postal Code' <dbl>, Region <chr>, 'Product ID' <chr>,
## #   Category <chr>, 'Sub-Category' <chr>, 'Product Name' <chr>, Sales <dbl>,
## #   Quantity <dbl>, Discount <dbl>, Profit <dbl>
```

To check the data type for each column:

```
## tibble [9,994 x 21] (S3: tbl_df/tbl/data.frame)
## $ Row ID      : num [1:9994] 1 2 3 4 5 6 7 8 9 10 ...
## $ Order ID    : chr [1:9994] "CA-2016-152156" "CA-2016-152156" "CA-2016-138688" "US-2015-108966" .
```

```
## $ Order Date : POSIXct[1:9994], format: "2016-11-08" "2016-11-08" ...
## $ Ship Date : POSIXct[1:9994], format: "2016-11-11" "2016-11-11" ...
## $ Ship Mode : chr [1:9994] "Second Class" "Second Class" "Second Class" "Standard Class" ...
## $ Customer ID : chr [1:9994] "CG-12520" "CG-12520" "DV-13045" "SO-20335" ...
## $ Customer Name: chr [1:9994] "Claire Gute" "Claire Gute" "Darrin Van Huff" "Sean O'Donnell" ...
## $ Segment : chr [1:9994] "Consumer" "Consumer" "Corporate" "Consumer" ...
## $ Country : chr [1:9994] "United States" "United States" "United States" "United States" ...
## $ City : chr [1:9994] "Henderson" "Henderson" "Los Angeles" "Fort Lauderdale" ...
## $ State : chr [1:9994] "Kentucky" "Kentucky" "California" "Florida" ...
## $ Postal Code : num [1:9994] 42420 42420 90036 33311 33311 ...
## $ Region : chr [1:9994] "South" "South" "West" "South" ...
## $ Product ID : chr [1:9994] "FUR-BO-10001798" "FUR-CH-10000454" "OFF-LA-10000240" "FUR-TA-1000057" ...
## $ Category : chr [1:9994] "Furniture" "Furniture" "Office Supplies" "Furniture" ...
## $ Sub-Category : chr [1:9994] "Bookcases" "Chairs" "Labels" "Tables" ...
## $ Product Name : chr [1:9994] "Bush Somerset Collection Bookcase" "Hon Deluxe Fabric Upholstered St..."
## $ Sales : num [1:9994] 262 731.9 14.6 957.6 22.4 ...
## $ Quantity : num [1:9994] 2 3 2 5 2 7 4 6 3 5 ...
## $ Discount : num [1:9994] 0 0 0 0.45 0.2 0 0 0.2 0.2 0 ...
## $ Profit : num [1:9994] 41.91 219.58 6.87 -383.03 2.52 ...
```

There are 6 numeric variables, 2 date variables, and 13 character variables.

```
glimpse(data)
```

```
## Rows: 9,994
## Columns: 21
## $ 'Row ID' <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, ...
## $ 'Order ID' <chr> "CA-2016-152156", "CA-2016-152156", "CA-2016-138688..."
## $ 'Order Date' <dtm> 2016-11-08, 2016-11-08, 2016-06-12, 2015-10-11, 20...
## $ 'Ship Date' <dtm> 2016-11-11, 2016-11-11, 2016-06-16, 2015-10-18, 20...
## $ 'Ship Mode' <chr> "Second Class", "Second Class", "Second Class", "St...
## $ 'Customer ID' <chr> "CG-12520", "CG-12520", "DV-13045", "SO-20335", "SO...
## $ 'Customer Name' <chr> "Claire Gute", "Claire Gute", "Darrin Van Huff", "S...
## $ Segment <chr> "Consumer", "Consumer", "Corporate", "Consumer", "C...
## $ Country <chr> "United States", "United States", "United States", ...
## $ City <chr> "Henderson", "Henderson", "Los Angeles", "Fort Laud...
## $ State <chr> "Kentucky", "Kentucky", "California", "Florida", "F...
## $ 'Postal Code' <dbl> 42420, 42420, 90036, 33311, 33311, 90032, 90032, 90...
## $ Region <chr> "South", "South", "West", "South", "South", "West",...
## $ 'Product ID' <chr> "FUR-BO-10001798", "FUR-CH-10000454", "OFF-LA-10000...
## $ Category <chr> "Furniture", "Furniture", "Office Supplies", "Furni...
## $ 'Sub-Category' <chr> "Bookcases", "Chairs", "Labels", "Tables", "Storage...
## $ 'Product Name' <chr> "Bush Somerset Collection Bookcase", "Hon Deluxe Fa...
## $ Sales <dbl> 261.9600, 731.9400, 14.6200, 957.5775, 22.3680, 48....
## $ Quantity <dbl> 2, 3, 2, 5, 2, 7, 4, 6, 3, 5, 9, 4, 3, 3, 5, 3, 6, ...
## $ Discount <dbl> 0.00, 0.00, 0.00, 0.45, 0.20, 0.00, 0.00, 0.20, 0.2...
## $ Profit <dbl> 41.9136, 219.5820, 6.8714, -383.0310, 2.5164, 14.16...
```

See if there are duplicates in the data and extract them:

```
data %>% distinct()
```

```
## # A tibble: 9,994 x 21
```

```
##      'Row ID' 'Order ID' 'Order Date'      'Ship Date'      'Ship Mode'
##      <dbl> <chr>      <dtm>      <dtm>      <chr>
## 1      1 CA-2016-1~ 2016-11-08 00:00:00 2016-11-11 00:00:00 Second Cla~
## 2      2 CA-2016-1~ 2016-11-08 00:00:00 2016-11-11 00:00:00 Second Cla~
## 3      3 CA-2016-1~ 2016-06-12 00:00:00 2016-06-16 00:00:00 Second Cla~
## 4      4 US-2015-1~ 2015-10-11 00:00:00 2015-10-18 00:00:00 Standard C~
## 5      5 US-2015-1~ 2015-10-11 00:00:00 2015-10-18 00:00:00 Standard C~
## 6      6 CA-2014-1~ 2014-06-09 00:00:00 2014-06-14 00:00:00 Standard C~
## 7      7 CA-2014-1~ 2014-06-09 00:00:00 2014-06-14 00:00:00 Standard C~
## 8      8 CA-2014-1~ 2014-06-09 00:00:00 2014-06-14 00:00:00 Standard C~
## 9      9 CA-2014-1~ 2014-06-09 00:00:00 2014-06-14 00:00:00 Standard C~
## 10     10 CA-2014-1~ 2014-06-09 00:00:00 2014-06-14 00:00:00 Standard C~
## # ... with 9,984 more rows, and 16 more variables: 'Customer ID' <chr>,
## #   'Customer Name' <chr>, Segment <chr>, Country <chr>, City <chr>,
## #   State <chr>, 'Postal Code' <dbl>, Region <chr>, 'Product ID' <chr>,
## #   Category <chr>, 'Sub-Category' <chr>, 'Product Name' <chr>, Sales <dbl>,
## #   Quantity <dbl>, Discount <dbl>, Profit <dbl>
```

Another way to extract only the unique columns of the dataset:

```
data_unique <- unique(data)
```

```
dim(data_unique)
```

```
## [1] 9994    21
```

The dimensions of the dataset with only the unique rows are still 9994 by 21, so it appears there are no duplicated rows in the original dataset.

Check for missing values in data:

```
## [1] "Row ID-1 missing values"      "Order ID-1 missing values"
## [3] "Order Date-1 missing values"  "Ship Date-1 missing values"
## [5] "Ship Mode-1 missing values"   "Customer ID-1 missing values"
## [7] "Customer Name-1 missing values" "Segment-1 missing values"
## [9] "Country-1 missing values"     "City-1 missing values"
## [11] "State-1 missing values"       "Postal Code-1 missing values"
## [13] "Region-1 missing values"      "Product ID-1 missing values"
## [15] "Category-1 missing values"    "Sub-Category-1 missing values"
## [17] "Product Name-1 missing values" "Sales-1 missing values"
## [19] "Quantity-1 missing values"    "Discount-1 missing values"
## [21] "Profit-1 missing values"
```

This is 1 missing value in this dataset.

```
## [1] "Row ID-0 missing values"      "Order ID-0 missing values"
## [3] "Order Date-0 missing values"  "Ship Date-0 missing values"
## [5] "Ship Mode-0 missing values"   "Customer ID-0 missing values"
## [7] "Customer Name-0 missing values" "Segment-0 missing values"
## [9] "Country-0 missing values"     "City-0 missing values"
## [11] "State-0 missing values"       "Postal Code-0 missing values"
```

```
## [13] "Region-0 missing values"      "Product ID-0 missing values"
## [15] "Category-0 missing values"    "Sub-Category-0 missing values"
## [17] "Product Name-0 missing values" "Sales-0 missing values"
## [19] "Quantity-0 missing values"    "Discount-0 missing values"
## [21] "Profit-0 missing values"
```

list rows of data that have missing values

```
data[!complete.cases(data),]
```

```
## # A tibble: 0 x 21
## # ... with 21 variables: 'Row ID' <dbl>, 'Order ID' <chr>, 'Order Date' <dtm>,
## #   'Ship Date' <dtm>, 'Ship Mode' <chr>, 'Customer ID' <chr>, 'Customer
## #   Name' <chr>, Segment <chr>, Country <chr>, City <chr>, State <chr>, 'Postal
## #   Code' <dbl>, Region <chr>, 'Product ID' <chr>, Category <chr>,
## #   'Sub-Category' <chr>, 'Product Name' <chr>, Sales <dbl>, Quantity <dbl>,
## #   Discount <dbl>, Profit <dbl>
```

Percentage of missing values

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

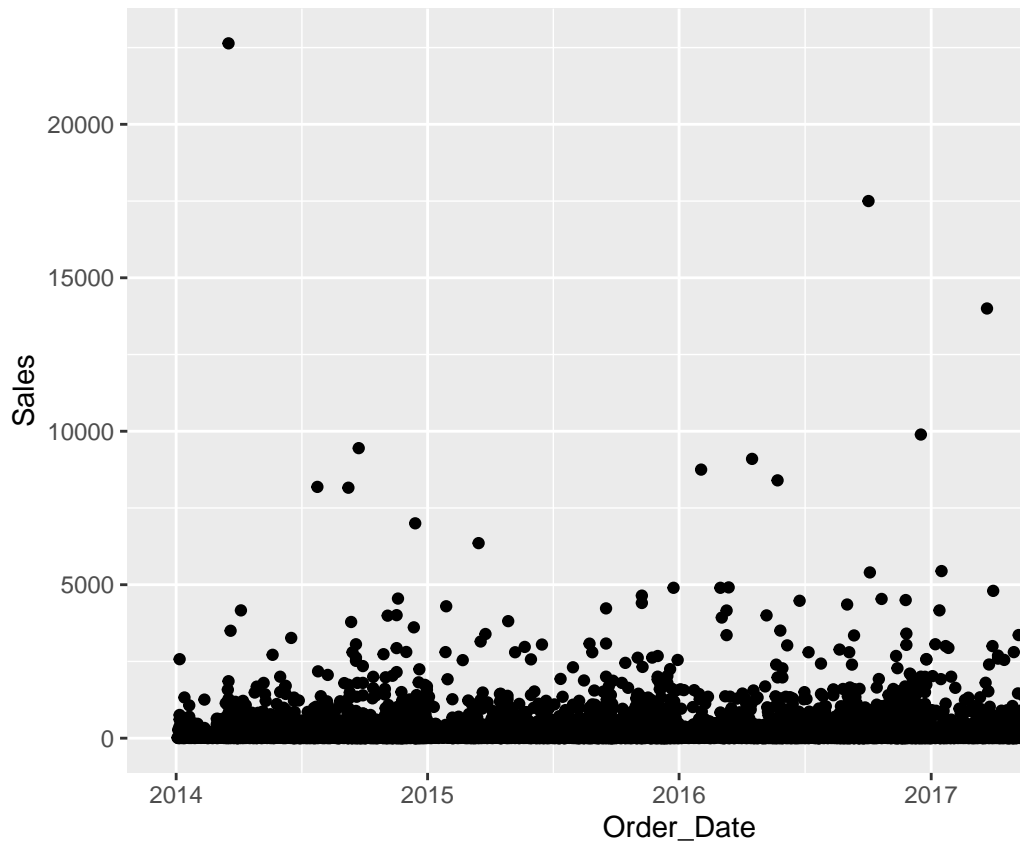
Remove the spaces in the column names and replace with "_" to make variable names easier to handle:

```
## [1] "Row_ID"      "Order_ID"    "Order_Date"  "Ship_Date"
## [5] "Ship_Mode"   "Customer_ID" "Customer_Name" "Segment"
## [9] "Country"     "City"        "State"       "Postal_Code"
## [13] "Region"      "Product_ID"  "Category"    "Sub-Category"
## [17] "Product_Name" "Sales"       "Quantity"    "Discount"
## [21] "Profit"
```

Exploratory Data Analysis

Find the difference between Order Date and Ship Date, and store into a new column called diff_in_days:

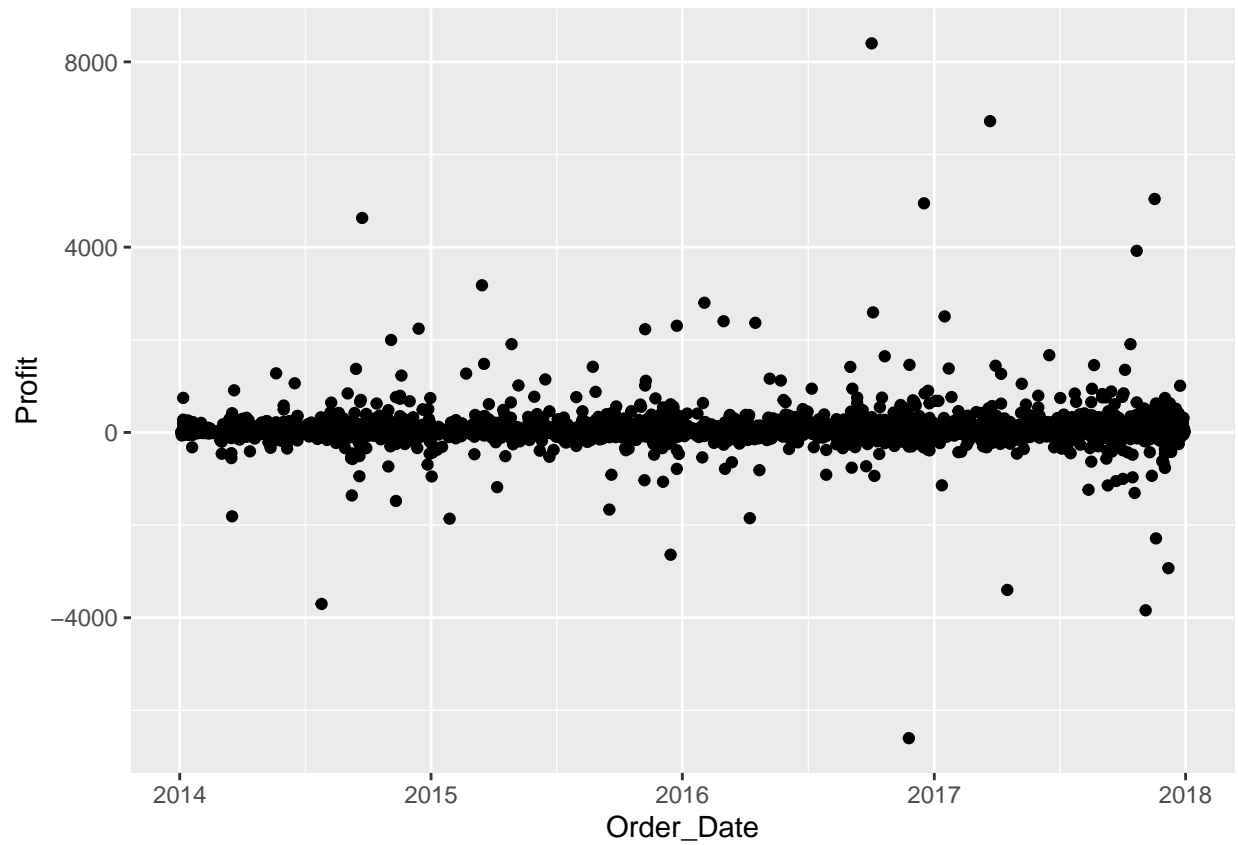
```
data$diff_in_days<- difftime(data$Ship_Date, data$Order_Date, units = c("days"))
```



Plot Sales in relation to Order Date:
Plot Profit in relation to Order Date:

```
ggplot(data = data) +  
  geom_point(mapping = aes(x = Order_Date, y = Profit), xlab="Order Date", ylab="Profit")
```

```
## Warning: Ignoring unknown parameters: xlab, ylab
```



Some outliers for certain days