SalesforEconomyDataAnalysis

Malay Raj

2023-09-19

## SALES FOR ECONOMY DATA ANALYSIS :

The dataset contains information about sales transactions, including details such as the customer’s age, gender, location, and the products sold.

## Loading Required Packages.

library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.2.3

## Warning: package 'ggplot2' was built under R version 4.2.3

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.0 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.4.2 ✔ tibble 3.1.8  
## ✔ lubridate 1.9.2 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.1   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the ]8;;http://conflicted.r-lib.org/conflicted package]8;; to force all conflicts to become errors

library(janitor)

## Warning: package 'janitor' was built under R version 4.2.3

##   
## Attaching package: 'janitor'  
##   
## The following objects are masked from 'package:stats':  
##   
## chisq.test, fisher.test

library(dplyr)  
library(ggplot2)  
library(skimr)

## Warning: package 'skimr' was built under R version 4.2.3

library(plyr)

## ------------------------------------------------------------------------------  
## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)  
## ------------------------------------------------------------------------------  
##   
## Attaching package: 'plyr'  
##   
## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize  
##   
## The following object is masked from 'package:purrr':  
##   
## compact

library(lubridate)  
library(scales)

##   
## Attaching package: 'scales'  
##   
## The following object is masked from 'package:purrr':  
##   
## discard  
##   
## The following object is masked from 'package:readr':  
##   
## col\_factor

library(plotly)

## Warning: package 'plotly' was built under R version 4.2.3

##   
## Attaching package: 'plotly'  
##   
## The following objects are masked from 'package:plyr':  
##   
## arrange, mutate, rename, summarise  
##   
## The following object is masked from 'package:ggplot2':  
##   
## last\_plot  
##   
## The following object is masked from 'package:stats':  
##   
## filter  
##   
## The following object is masked from 'package:graphics':  
##   
## layout

library(data.table)

##   
## Attaching package: 'data.table'  
##   
## The following objects are masked from 'package:lubridate':  
##   
## hour, isoweek, mday, minute, month, quarter, second, wday, week,  
## yday, year  
##   
## The following objects are masked from 'package:dplyr':  
##   
## between, first, last  
##   
## The following object is masked from 'package:purrr':  
##   
## transpose

## Displaying my current working directory.

getwd()

## [1] "C:/Sales Data for Economy Data Analysis"

## Collecting the data.

sales = read.csv("C:/Sales Data for Economy Data Analysis/salesforcourse-4fe2kehu.csv")

## Displaying the first six rows of dataset.

head(sales)

## index Date Year Month Customer.Age Customer.Gender Country  
## 1 0 2/19/2016 2016 February 29 F United States  
## 2 1 2/20/2016 2016 February 29 F United States  
## 3 2 2/27/2016 2016 February 29 F United States  
## 4 3 3/12/2016 2016 March 29 F United States  
## 5 4 3/12/2016 2016 March 29 F United States  
## 6 5 4/8/2016 2016 April 29 F United States  
## State Product.Category Sub.Category Quantity Unit.Cost Unit.Price  
## 1 Washington Accessories Tires and Tubes 1 80.00 109.00000  
## 2 Washington Clothing Gloves 2 24.50 28.50000  
## 3 Washington Accessories Tires and Tubes 3 3.67 5.00000  
## 4 Washington Accessories Tires and Tubes 2 87.50 116.50000  
## 5 Washington Accessories Tires and Tubes 3 35.00 41.66667  
## 6 Washington Accessories Tires and Tubes 1 66.00 78.00000  
## Cost Revenue Column1  
## 1 80 109 NA  
## 2 49 57 NA  
## 3 11 15 NA  
## 4 175 233 NA  
## 5 105 125 NA  
## 6 66 78 NA

## Checking the dimension of the dataset.

dim(sales)

## [1] 34867 16

## Displaying the column names of our dataframe.

colnames(sales)

## [1] "index" "Date" "Year" "Month"   
## [5] "Customer.Age" "Customer.Gender" "Country" "State"   
## [9] "Product.Category" "Sub.Category" "Quantity" "Unit.Cost"   
## [13] "Unit.Price" "Cost" "Revenue" "Column1"

## Inspecting the dataframe and look for inconguencies.

str(sales)

## 'data.frame': 34867 obs. of 16 variables:  
## $ index : int 0 1 2 3 4 5 6 7 8 9 ...  
## $ Date : chr "2/19/2016" "2/20/2016" "2/27/2016" "3/12/2016" ...  
## $ Year : int 2016 2016 2016 2016 2016 2016 2016 2016 2016 2016 ...  
## $ Month : chr "February" "February" "February" "March" ...  
## $ Customer.Age : int 29 29 29 29 29 29 29 29 29 29 ...  
## $ Customer.Gender : chr "F" "F" "F" "F" ...  
## $ Country : chr "United States" "United States" "United States" "United States" ...  
## $ State : chr "Washington" "Washington" "Washington" "Washington" ...  
## $ Product.Category: chr "Accessories" "Clothing" "Accessories" "Accessories" ...  
## $ Sub.Category : chr "Tires and Tubes" "Gloves" "Tires and Tubes" "Tires and Tubes" ...  
## $ Quantity : int 1 2 3 2 3 1 2 1 2 2 ...  
## $ Unit.Cost : num 80 24.5 3.67 87.5 35 66 52 60 8 2.5 ...  
## $ Unit.Price : num 109 28.5 5 116.5 41.7 ...  
## $ Cost : int 80 49 11 175 105 66 104 60 16 5 ...  
## $ Revenue : num 109 57 15 233 125 78 120 68 20 6 ...  
## $ Column1 : num NA NA NA NA NA NA NA NA NA NA ...

## Changing the datatypes.

sales$Date <- as.Date(sales$Date, format = "%m/%d/%Y")

## Checking if it’s changed or not.

str(sales)

## 'data.frame': 34867 obs. of 16 variables:  
## $ index : int 0 1 2 3 4 5 6 7 8 9 ...  
## $ Date : Date, format: "2016-02-19" "2016-02-20" ...  
## $ Year : int 2016 2016 2016 2016 2016 2016 2016 2016 2016 2016 ...  
## $ Month : chr "February" "February" "February" "March" ...  
## $ Customer.Age : int 29 29 29 29 29 29 29 29 29 29 ...  
## $ Customer.Gender : chr "F" "F" "F" "F" ...  
## $ Country : chr "United States" "United States" "United States" "United States" ...  
## $ State : chr "Washington" "Washington" "Washington" "Washington" ...  
## $ Product.Category: chr "Accessories" "Clothing" "Accessories" "Accessories" ...  
## $ Sub.Category : chr "Tires and Tubes" "Gloves" "Tires and Tubes" "Tires and Tubes" ...  
## $ Quantity : int 1 2 3 2 3 1 2 1 2 2 ...  
## $ Unit.Cost : num 80 24.5 3.67 87.5 35 66 52 60 8 2.5 ...  
## $ Unit.Price : num 109 28.5 5 116.5 41.7 ...  
## $ Cost : int 80 49 11 175 105 66 104 60 16 5 ...  
## $ Revenue : num 109 57 15 233 125 78 120 68 20 6 ...  
## $ Column1 : num NA NA NA NA NA NA NA NA NA NA ...

## Removing certain columns from each of the datasets because we don’t need them in our analysis.

sales = subset(sales, select = -c(index, Year, Month, Column1))

## Checking the column names which are left for analysis.

colnames(sales)

## [1] "Date" "Customer.Age" "Customer.Gender" "Country"   
## [5] "State" "Product.Category" "Sub.Category" "Quantity"   
## [9] "Unit.Cost" "Unit.Price" "Cost" "Revenue"

## Creating a new column of Year\_Month from Date column.

sales$Year\_Month <- as.Date(paste(year(sales$Date), month(sales$Date), "01", sep = "-"))

## Checking for null values in our dataset.

null\_sum <- colSums(is.na(sales))  
null\_sum

## Date Customer.Age Customer.Gender Country   
## 1 1 0 0   
## State Product.Category Sub.Category Quantity   
## 0 0 0 1   
## Unit.Cost Unit.Price Cost Revenue   
## 1 1 1 0   
## Year\_Month   
## 1

## Removing the null values.

sales <- na.omit(sales)

## Checking if there is any null value left or not.

sum(is.na(sales))

## [1] 0

## Creating four more columns that we need in our analysis.

sales$Unit\_Margin <- sales$Unit.Price - sales$Unit.Cost  
sales$Unit\_Margin\_percent <- round((sales$Unit\_Margin / sales$Unit.Price) \* 100, 2)  
  
sales$Margin <- sales$Revenue - sales$Cost  
sales$Margin\_percent <- round((sales$Margin / sales$Revenue) \* 100, 2)

## Checking for any duplicate values in our dataset.

anyDuplicated(sales)

## [1] 870

## Removing those duplicate values.

sales <- unique(sales)  
anyDuplicated(sales)

## [1] 0

## Checking for the current dimensions of our dataset.

dim(sales)

## [1] 34865 17

## Seeing how many observations fall under each of these column.

table(sales$Customer.Age)

##   
## 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32   
## 357 451 605 596 690 722 810 926 919 1016 1034 1277 1234 1204 1307 1199   
## 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48   
## 1156 1260 1139 1054 1070 954 983 1128 999 932 904 911 752 723 609 536   
## 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64   
## 617 523 546 544 496 384 376 296 272 185 186 179 172 142 117 105   
## 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80   
## 50 23 22 29 19 24 14 10 4 2 10 2 8 11 5 3   
## 81 82 84 85 86 87   
## 6 2 9 8 4 3

table(sales$Quantity)

##   
## 1 2 3   
## 11505 11767 11593

table(sales$Date)

##   
## 2015-01-01 2015-01-02 2015-01-03 2015-01-04 2015-01-05 2015-01-07 2015-01-08   
## 6 7 3 3 3 7 6   
## 2015-01-09 2015-01-10 2015-01-11 2015-01-12 2015-01-13 2015-01-14 2015-01-15   
## 6 4 4 5 8 4 5   
## 2015-01-16 2015-01-17 2015-01-18 2015-01-19 2015-01-20 2015-01-21 2015-01-22   
## 8 5 4 2 4 2 5   
## 2015-01-23 2015-01-24 2015-01-25 2015-01-26 2015-01-27 2015-01-28 2015-01-29   
## 8 6 3 4 2 2 5   
## 2015-01-30 2015-01-31 2015-02-01 2015-02-02 2015-02-03 2015-02-04 2015-02-05   
## 3 4 6 5 7 7 5   
## 2015-02-06 2015-02-07 2015-02-08 2015-02-09 2015-02-10 2015-02-11 2015-02-12   
## 10 1 7 7 3 2 4   
## 2015-02-13 2015-02-14 2015-02-15 2015-02-16 2015-02-17 2015-02-18 2015-02-19   
## 7 8 3 2 6 6 7   
## 2015-02-20 2015-02-21 2015-02-22 2015-02-23 2015-02-24 2015-02-25 2015-02-26   
## 4 9 7 1 8 9 6   
## 2015-02-27 2015-02-28 2015-03-01 2015-03-02 2015-03-03 2015-03-04 2015-03-05   
## 2 5 5 5 2 4 9   
## 2015-03-06 2015-03-07 2015-03-08 2015-03-09 2015-03-10 2015-03-11 2015-03-12   
## 5 8 6 2 2 4 10   
## 2015-03-13 2015-03-14 2015-03-15 2015-03-16 2015-03-17 2015-03-18 2015-03-19   
## 3 7 3 5 4 9 8   
## 2015-03-20 2015-03-21 2015-03-22 2015-03-23 2015-03-24 2015-03-25 2015-03-26   
## 7 2 5 2 3 2 4   
## 2015-03-27 2015-03-28 2015-03-29 2015-03-30 2015-03-31 2015-04-01 2015-04-02   
## 1 7 5 3 7 10 5   
## 2015-04-03 2015-04-04 2015-04-05 2015-04-06 2015-04-07 2015-04-08 2015-04-09   
## 3 8 7 3 2 8 6   
## 2015-04-10 2015-04-11 2015-04-12 2015-04-13 2015-04-14 2015-04-15 2015-04-16   
## 1 10 3 13 5 7 5   
## 2015-04-17 2015-04-18 2015-04-19 2015-04-20 2015-04-21 2015-04-22 2015-04-23   
## 7 5 5 6 5 4 4   
## 2015-04-24 2015-04-25 2015-04-26 2015-04-27 2015-04-28 2015-04-29 2015-04-30   
## 10 9 5 4 5 3 9   
## 2015-05-01 2015-05-02 2015-05-03 2015-05-04 2015-05-05 2015-05-06 2015-05-07   
## 8 7 9 5 10 10 5   
## 2015-05-08 2015-05-09 2015-05-10 2015-05-11 2015-05-12 2015-05-13 2015-05-14   
## 11 5 4 6 6 5 8   
## 2015-05-15 2015-05-16 2015-05-17 2015-05-18 2015-05-19 2015-05-20 2015-05-21   
## 3 6 5 1 5 8 7   
## 2015-05-22 2015-05-23 2015-05-24 2015-05-25 2015-05-26 2015-05-27 2015-05-28   
## 4 9 11 3 7 6 5   
## 2015-05-29 2015-05-30 2015-05-31 2015-06-01 2015-06-02 2015-06-03 2015-06-04   
## 5 8 12 9 10 6 7   
## 2015-06-05 2015-06-06 2015-06-07 2015-06-08 2015-06-09 2015-06-10 2015-06-11   
## 5 4 7 13 4 9 5   
## 2015-06-12 2015-06-13 2015-06-14 2015-06-15 2015-06-16 2015-06-17 2015-06-18   
## 8 5 7 8 6 6 7   
## 2015-06-19 2015-06-20 2015-06-21 2015-06-22 2015-06-23 2015-06-24 2015-06-25   
## 14 4 9 5 6 8 4   
## 2015-06-26 2015-06-27 2015-06-28 2015-06-29 2015-06-30 2015-07-01 2015-07-02   
## 4 9 8 4 5 29 21   
## 2015-07-03 2015-07-04 2015-07-05 2015-07-06 2015-07-07 2015-07-08 2015-07-09   
## 35 43 30 15 24 26 36   
## 2015-07-10 2015-07-11 2015-07-12 2015-07-13 2015-07-14 2015-07-15 2015-07-16   
## 27 48 37 15 34 25 31   
## 2015-07-17 2015-07-18 2015-07-19 2015-07-20 2015-07-21 2015-07-22 2015-07-23   
## 25 18 25 53 22 33 21   
## 2015-07-24 2015-07-25 2015-07-26 2015-07-27 2015-07-28 2015-07-29 2015-07-30   
## 52 16 29 30 28 39 24   
## 2015-07-31 2015-08-01 2015-08-02 2015-08-03 2015-08-04 2015-08-05 2015-08-06   
## 37 68 54 68 53 59 71   
## 2015-08-07 2015-08-08 2015-08-09 2015-08-10 2015-08-11 2015-08-12 2015-08-13   
## 75 129 71 60 57 97 74   
## 2015-08-14 2015-08-15 2015-08-16 2015-08-17 2015-08-18 2015-08-19 2015-08-20   
## 68 60 87 77 94 91 61   
## 2015-08-21 2015-08-22 2015-08-23 2015-08-24 2015-08-25 2015-08-26 2015-08-27   
## 92 52 51 82 86 74 79   
## 2015-08-28 2015-08-29 2015-08-30 2015-08-31 2015-09-01 2015-09-02 2015-09-03   
## 82 89 75 67 53 81 69   
## 2015-09-04 2015-09-05 2015-09-06 2015-09-07 2015-09-08 2015-09-09 2015-09-10   
## 92 74 65 75 84 94 77   
## 2015-09-11 2015-09-12 2015-09-13 2015-09-14 2015-09-15 2015-09-16 2015-09-17   
## 68 75 99 84 80 63 86   
## 2015-09-18 2015-09-19 2015-09-20 2015-09-21 2015-09-22 2015-09-23 2015-09-24   
## 89 97 78 74 80 105 59   
## 2015-09-25 2015-09-26 2015-09-27 2015-09-28 2015-09-29 2015-09-30 2015-10-01   
## 68 74 84 61 51 64 93   
## 2015-10-02 2015-10-03 2015-10-04 2015-10-05 2015-10-06 2015-10-07 2015-10-08   
## 97 85 80 91 110 87 106   
## 2015-10-09 2015-10-10 2015-10-11 2015-10-12 2015-10-13 2015-10-14 2015-10-15   
## 101 80 78 85 68 86 112   
## 2015-10-16 2015-10-17 2015-10-18 2015-10-19 2015-10-20 2015-10-21 2015-10-22   
## 63 60 78 59 84 66 89   
## 2015-10-23 2015-10-24 2015-10-25 2015-10-26 2015-10-27 2015-10-28 2015-10-29   
## 105 57 78 96 60 68 73   
## 2015-10-30 2015-10-31 2015-11-01 2015-11-02 2015-11-03 2015-11-04 2015-11-05   
## 81 78 61 86 85 69 83   
## 2015-11-06 2015-11-07 2015-11-08 2015-11-09 2015-11-10 2015-11-11 2015-11-12   
## 72 80 72 71 90 49 85   
## 2015-11-13 2015-11-14 2015-11-15 2015-11-16 2015-11-17 2015-11-18 2015-11-19   
## 92 89 94 68 93 96 88   
## 2015-11-20 2015-11-21 2015-11-22 2015-11-23 2015-11-24 2015-11-25 2015-11-26   
## 55 73 97 75 113 57 87   
## 2015-11-27 2015-11-28 2015-11-29 2015-11-30 2015-12-01 2015-12-02 2015-12-03   
## 95 96 106 111 112 118 134   
## 2015-12-04 2015-12-05 2015-12-06 2015-12-07 2015-12-08 2015-12-09 2015-12-10   
## 124 108 123 147 97 98 67   
## 2015-12-11 2015-12-12 2015-12-13 2015-12-14 2015-12-15 2015-12-16 2015-12-17   
## 126 115 110 107 141 75 114   
## 2015-12-18 2015-12-19 2015-12-20 2015-12-21 2015-12-22 2015-12-23 2015-12-24   
## 131 95 93 116 63 103 102   
## 2015-12-25 2015-12-26 2015-12-27 2015-12-28 2015-12-29 2015-12-30 2015-12-31   
## 120 131 78 120 132 103 111   
## 2016-01-01 2016-01-02 2016-01-03 2016-01-04 2016-01-05 2016-01-06 2016-01-07   
## 60 89 85 61 83 96 89   
## 2016-01-08 2016-01-09 2016-01-10 2016-01-11 2016-01-12 2016-01-13 2016-01-14   
## 116 80 129 104 57 59 94   
## 2016-01-15 2016-01-16 2016-01-17 2016-01-18 2016-01-19 2016-01-20 2016-01-21   
## 96 99 58 110 86 143 74   
## 2016-01-22 2016-01-23 2016-01-24 2016-01-25 2016-01-26 2016-01-27 2016-01-28   
## 84 72 102 90 83 88 117   
## 2016-01-29 2016-01-30 2016-01-31 2016-02-01 2016-02-02 2016-02-03 2016-02-04   
## 86 101 78 122 93 97 81   
## 2016-02-05 2016-02-06 2016-02-07 2016-02-08 2016-02-09 2016-02-10 2016-02-11   
## 116 102 103 104 85 83 65   
## 2016-02-12 2016-02-13 2016-02-14 2016-02-15 2016-02-16 2016-02-17 2016-02-18   
## 96 95 96 92 82 107 117   
## 2016-02-19 2016-02-20 2016-02-21 2016-02-22 2016-02-23 2016-02-24 2016-02-25   
## 103 162 97 78 98 117 64   
## 2016-02-26 2016-02-27 2016-02-28 2016-03-01 2016-03-02 2016-03-03 2016-03-04   
## 81 91 106 196 92 81 95   
## 2016-03-05 2016-03-06 2016-03-07 2016-03-08 2016-03-09 2016-03-10 2016-03-11   
## 85 83 100 82 120 96 103   
## 2016-03-12 2016-03-13 2016-03-14 2016-03-15 2016-03-16 2016-03-17 2016-03-18   
## 106 117 87 81 62 73 89   
## 2016-03-19 2016-03-20 2016-03-21 2016-03-22 2016-03-23 2016-03-24 2016-03-25   
## 102 87 113 84 105 95 88   
## 2016-03-26 2016-03-27 2016-03-28 2016-03-29 2016-03-30 2016-03-31 2016-04-01   
## 74 137 105 95 95 78 97   
## 2016-04-02 2016-04-03 2016-04-04 2016-04-05 2016-04-06 2016-04-07 2016-04-08   
## 112 125 99 68 108 117 109   
## 2016-04-09 2016-04-10 2016-04-11 2016-04-12 2016-04-13 2016-04-14 2016-04-15   
## 100 108 85 85 83 94 135   
## 2016-04-16 2016-04-17 2016-04-18 2016-04-19 2016-04-20 2016-04-21 2016-04-22   
## 91 98 114 171 90 74 99   
## 2016-04-23 2016-04-24 2016-04-25 2016-04-26 2016-04-27 2016-04-28 2016-04-29   
## 110 113 102 89 104 117 98   
## 2016-04-30 2016-05-01 2016-05-02 2016-05-03 2016-05-04 2016-05-05 2016-05-06   
## 111 142 86 118 142 126 114   
## 2016-05-07 2016-05-08 2016-05-09 2016-05-10 2016-05-11 2016-05-12 2016-05-13   
## 121 119 122 106 96 130 99   
## 2016-05-14 2016-05-15 2016-05-16 2016-05-17 2016-05-18 2016-05-19 2016-05-20   
## 113 122 126 108 89 115 129   
## 2016-05-21 2016-05-22 2016-05-23 2016-05-24 2016-05-25 2016-05-26 2016-05-27   
## 140 85 91 90 114 110 95   
## 2016-05-28 2016-05-29 2016-05-30 2016-05-31 2016-06-01 2016-06-02 2016-06-03   
## 113 94 112 105 77 84 99   
## 2016-06-04 2016-06-05 2016-06-06 2016-06-07 2016-06-08 2016-06-09 2016-06-10   
## 86 132 136 143 106 112 102   
## 2016-06-11 2016-06-12 2016-06-13 2016-06-14 2016-06-15 2016-06-16 2016-06-17   
## 136 93 88 172 133 96 146   
## 2016-06-18 2016-06-19 2016-06-20 2016-06-21 2016-06-22 2016-06-23 2016-06-24   
## 109 138 114 136 146 100 120   
## 2016-06-25 2016-06-26 2016-06-27 2016-06-28 2016-06-29 2016-06-30 2016-07-01   
## 114 106 109 149 113 79 40   
## 2016-07-02 2016-07-03 2016-07-04 2016-07-05 2016-07-06 2016-07-07 2016-07-08   
## 47 39 30 41 37 39 72   
## 2016-07-09 2016-07-10 2016-07-11 2016-07-12 2016-07-13 2016-07-14 2016-07-15   
## 36 39 33 35 22 38 56   
## 2016-07-16 2016-07-17 2016-07-18 2016-07-19 2016-07-20 2016-07-21 2016-07-22   
## 32 58 42 59 55 23 49   
## 2016-07-23 2016-07-24 2016-07-25 2016-07-26 2016-07-27 2016-07-28 2016-07-29   
## 35 52 24 55 35 51 36   
## 2016-07-30 2016-07-31   
## 26 51

table(sales$Year\_Month)

##   
## 2015-01-01 2015-02-01 2015-03-01 2015-04-01 2015-05-01 2015-06-01 2015-07-01   
## 138 154 149 177 204 206 928   
## 2015-08-01 2015-09-01 2015-10-01 2015-11-01 2015-12-01 2016-01-01 2016-02-01   
## 2303 2303 2554 2488 3414 2769 2733   
## 2016-03-01 2016-04-01 2016-05-01 2016-06-01 2016-07-01   
## 3006 3106 3472 3474 1287

## CONDUCTNG DESCRIPTIVE ANALYSIS:

Descriptive analysis on few columns of our dataset.

summary(sales$Quantity)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.000 1.000 2.000 2.003 3.000 3.000

summary(sales$Unit.Cost)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.67 45.00 150.00 349.89 455.00 3240.00

summary(sales$Unit.Price)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.667 53.667 179.000 389.243 521.000 5082.000

summary(sales$Unit\_Margin)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -937.00 3.00 14.50 39.35 53.00 1842.00

summary(sales$Unit\_Margin\_percent)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -68.67 6.17 14.79 13.41 22.57 49.75

summary(sales$Margin)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -937.00 5.00 27.00 64.87 96.00 1842.00

summary(sales$Margin\_percent)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -68.67 6.17 14.80 13.41 22.57 50.00

Comparing different columns with each other from each of our dataset.

aggregate(sales$Quantity ~ sales$Customer.Age, data = sales, FUN = mean)

## sales$Customer.Age sales$Quantity  
## 1 17 2.047619  
## 2 18 1.966741  
## 3 19 2.000000  
## 4 20 1.951342  
## 5 21 1.988406  
## 6 22 2.013850  
## 7 23 1.972840  
## 8 24 1.987041  
## 9 25 1.991295  
## 10 26 2.048228  
## 11 27 1.998066  
## 12 28 1.992952  
## 13 29 2.033225  
## 14 30 1.990033  
## 15 31 2.012242  
## 16 32 1.984987  
## 17 33 1.964533  
## 18 34 2.015079  
## 19 35 1.963126  
## 20 36 2.012334  
## 21 37 2.014019  
## 22 38 1.986373  
## 23 39 2.013225  
## 24 40 1.940603  
## 25 41 2.004004  
## 26 42 2.049356  
## 27 43 2.034292  
## 28 44 2.032931  
## 29 45 2.057181  
## 30 46 2.045643  
## 31 47 2.009852  
## 32 48 2.007463  
## 33 49 2.035656  
## 34 50 1.992352  
## 35 51 1.959707  
## 36 52 2.029412  
## 37 53 1.973790  
## 38 54 1.955729  
## 39 55 1.957447  
## 40 56 2.027027  
## 41 57 2.125000  
## 42 58 2.054054  
## 43 59 2.026882  
## 44 60 1.932961  
## 45 61 1.976744  
## 46 62 2.091549  
## 47 63 1.991453  
## 48 64 1.876190  
## 49 65 1.940000  
## 50 66 1.826087  
## 51 67 2.136364  
## 52 68 2.103448  
## 53 69 2.000000  
## 54 70 2.166667  
## 55 71 1.857143  
## 56 72 1.800000  
## 57 73 2.250000  
## 58 74 2.000000  
## 59 75 2.200000  
## 60 76 2.000000  
## 61 77 1.750000  
## 62 78 1.727273  
## 63 79 2.000000  
## 64 80 1.333333  
## 65 81 1.333333  
## 66 82 2.500000  
## 67 84 2.111111  
## 68 85 1.375000  
## 69 86 2.000000  
## 70 87 2.666667

aggregate(sales$Quantity ~ sales$Customer.Age, data = sales, FUN = median)

## sales$Customer.Age sales$Quantity  
## 1 17 2.0  
## 2 18 2.0  
## 3 19 2.0  
## 4 20 2.0  
## 5 21 2.0  
## 6 22 2.0  
## 7 23 2.0  
## 8 24 2.0  
## 9 25 2.0  
## 10 26 2.0  
## 11 27 2.0  
## 12 28 2.0  
## 13 29 2.0  
## 14 30 2.0  
## 15 31 2.0  
## 16 32 2.0  
## 17 33 2.0  
## 18 34 2.0  
## 19 35 2.0  
## 20 36 2.0  
## 21 37 2.0  
## 22 38 2.0  
## 23 39 2.0  
## 24 40 2.0  
## 25 41 2.0  
## 26 42 2.0  
## 27 43 2.0  
## 28 44 2.0  
## 29 45 2.0  
## 30 46 2.0  
## 31 47 2.0  
## 32 48 2.0  
## 33 49 2.0  
## 34 50 2.0  
## 35 51 2.0  
## 36 52 2.0  
## 37 53 2.0  
## 38 54 2.0  
## 39 55 2.0  
## 40 56 2.0  
## 41 57 2.0  
## 42 58 2.0  
## 43 59 2.0  
## 44 60 2.0  
## 45 61 2.0  
## 46 62 2.0  
## 47 63 2.0  
## 48 64 2.0  
## 49 65 2.0  
## 50 66 2.0  
## 51 67 2.0  
## 52 68 2.0  
## 53 69 2.0  
## 54 70 2.0  
## 55 71 2.0  
## 56 72 1.5  
## 57 73 2.5  
## 58 74 2.0  
## 59 75 2.0  
## 60 76 2.0  
## 61 77 1.5  
## 62 78 2.0  
## 63 79 2.0  
## 64 80 1.0  
## 65 81 1.0  
## 66 82 2.5  
## 67 84 2.0  
## 68 85 1.0  
## 69 86 2.0  
## 70 87 3.0

aggregate(sales$Quantity ~ sales$Customer.Age, data = sales, FUN = max)

## sales$Customer.Age sales$Quantity  
## 1 17 3  
## 2 18 3  
## 3 19 3  
## 4 20 3  
## 5 21 3  
## 6 22 3  
## 7 23 3  
## 8 24 3  
## 9 25 3  
## 10 26 3  
## 11 27 3  
## 12 28 3  
## 13 29 3  
## 14 30 3  
## 15 31 3  
## 16 32 3  
## 17 33 3  
## 18 34 3  
## 19 35 3  
## 20 36 3  
## 21 37 3  
## 22 38 3  
## 23 39 3  
## 24 40 3  
## 25 41 3  
## 26 42 3  
## 27 43 3  
## 28 44 3  
## 29 45 3  
## 30 46 3  
## 31 47 3  
## 32 48 3  
## 33 49 3  
## 34 50 3  
## 35 51 3  
## 36 52 3  
## 37 53 3  
## 38 54 3  
## 39 55 3  
## 40 56 3  
## 41 57 3  
## 42 58 3  
## 43 59 3  
## 44 60 3  
## 45 61 3  
## 46 62 3  
## 47 63 3  
## 48 64 3  
## 49 65 3  
## 50 66 3  
## 51 67 3  
## 52 68 3  
## 53 69 3  
## 54 70 3  
## 55 71 3  
## 56 72 3  
## 57 73 3  
## 58 74 3  
## 59 75 3  
## 60 76 3  
## 61 77 3  
## 62 78 3  
## 63 79 3  
## 64 80 2  
## 65 81 2  
## 66 82 3  
## 67 84 3  
## 68 85 2  
## 69 86 3  
## 70 87 3

aggregate(sales$Quantity ~ sales$Customer.Age, data = sales, FUN = min)

## sales$Customer.Age sales$Quantity  
## 1 17 1  
## 2 18 1  
## 3 19 1  
## 4 20 1  
## 5 21 1  
## 6 22 1  
## 7 23 1  
## 8 24 1  
## 9 25 1  
## 10 26 1  
## 11 27 1  
## 12 28 1  
## 13 29 1  
## 14 30 1  
## 15 31 1  
## 16 32 1  
## 17 33 1  
## 18 34 1  
## 19 35 1  
## 20 36 1  
## 21 37 1  
## 22 38 1  
## 23 39 1  
## 24 40 1  
## 25 41 1  
## 26 42 1  
## 27 43 1  
## 28 44 1  
## 29 45 1  
## 30 46 1  
## 31 47 1  
## 32 48 1  
## 33 49 1  
## 34 50 1  
## 35 51 1  
## 36 52 1  
## 37 53 1  
## 38 54 1  
## 39 55 1  
## 40 56 1  
## 41 57 1  
## 42 58 1  
## 43 59 1  
## 44 60 1  
## 45 61 1  
## 46 62 1  
## 47 63 1  
## 48 64 1  
## 49 65 1  
## 50 66 1  
## 51 67 1  
## 52 68 1  
## 53 69 1  
## 54 70 1  
## 55 71 1  
## 56 72 1  
## 57 73 1  
## 58 74 1  
## 59 75 1  
## 60 76 1  
## 61 77 1  
## 62 78 1  
## 63 79 1  
## 64 80 1  
## 65 81 1  
## 66 82 2  
## 67 84 1  
## 68 85 1  
## 69 86 1  
## 70 87 2

aggregate(sales$Unit.Cost ~ sales$Customer.Age, data = sales, FUN = mean)

## sales$Customer.Age sales$Unit.Cost  
## 1 17 207.13829  
## 2 18 263.58089  
## 3 19 263.65169  
## 4 20 222.64819  
## 5 21 228.96787  
## 6 22 292.94273  
## 7 23 248.67567  
## 8 24 341.99419  
## 9 25 343.17820  
## 10 26 313.56961  
## 11 27 351.08079  
## 12 28 415.33204  
## 13 29 391.22041  
## 14 30 358.24129  
## 15 31 398.01506  
## 16 32 352.43776  
## 17 33 367.65518  
## 18 34 369.58452  
## 19 35 401.60115  
## 20 36 344.19138  
## 21 37 354.63681  
## 22 38 378.86579  
## 23 39 411.16966  
## 24 40 390.32935  
## 25 41 371.99702  
## 26 42 397.54374  
## 27 43 392.27491  
## 28 44 352.13392  
## 29 45 392.19085  
## 30 46 341.13827  
## 31 47 323.16302  
## 32 48 367.28069  
## 33 49 304.84919  
## 34 50 327.96205  
## 35 51 382.52526  
## 36 52 347.21029  
## 37 53 403.92196  
## 38 54 339.49164  
## 39 55 303.41750  
## 40 56 353.80953  
## 41 57 289.14154  
## 42 58 341.00449  
## 43 59 297.16204  
## 44 60 282.02520  
## 45 61 374.56035  
## 46 62 330.22394  
## 47 63 289.29188  
## 48 64 309.76829  
## 49 65 310.78960  
## 50 66 332.26783  
## 51 67 190.60636  
## 52 68 116.10931  
## 53 69 251.31526  
## 54 70 118.90958  
## 55 71 172.40429  
## 56 72 565.63400  
## 57 73 404.66750  
## 58 74 321.66500  
## 59 75 274.16700  
## 60 76 85.83500  
## 61 77 158.72875  
## 62 78 139.13636  
## 63 79 84.13400  
## 64 80 179.33333  
## 65 81 180.75000  
## 66 82 159.41500  
## 67 84 78.72222  
## 68 85 140.18750  
## 69 86 559.33250  
## 70 87 37.33333

aggregate(sales$Unit.Cost ~ sales$Customer.Age, data = sales, FUN = median)

## sales$Customer.Age sales$Unit.Cost  
## 1 17 80.000  
## 2 18 114.670  
## 3 19 100.000  
## 4 20 84.500  
## 5 21 90.000  
## 6 22 99.000  
## 7 23 107.500  
## 8 24 145.000  
## 9 25 150.000  
## 10 26 127.665  
## 11 27 180.000  
## 12 28 200.670  
## 13 29 180.000  
## 14 30 180.000  
## 15 31 203.000  
## 16 32 174.000  
## 17 33 175.000  
## 18 34 180.000  
## 19 35 180.000  
## 20 36 159.500  
## 21 37 183.165  
## 22 38 183.330  
## 23 39 190.000  
## 24 40 146.750  
## 25 41 145.000  
## 26 42 180.000  
## 27 43 175.000  
## 28 44 144.000  
## 29 45 175.000  
## 30 46 130.000  
## 31 47 121.500  
## 32 48 140.000  
## 33 49 120.000  
## 34 50 126.000  
## 35 51 161.165  
## 36 52 140.000  
## 37 53 180.000  
## 38 54 150.000  
## 39 55 112.500  
## 40 56 140.665  
## 41 57 129.165  
## 42 58 153.000  
## 43 59 127.665  
## 44 60 100.000  
## 45 61 150.835  
## 46 62 107.500  
## 47 63 80.000  
## 48 64 105.000  
## 49 65 123.500  
## 50 66 62.500  
## 51 67 132.500  
## 52 68 66.670  
## 53 69 110.000  
## 54 70 47.500  
## 55 71 51.665  
## 56 72 396.500  
## 57 73 181.335  
## 58 74 321.665  
## 59 75 173.750  
## 60 76 85.835  
## 61 77 127.000  
## 62 78 67.500  
## 63 79 93.000  
## 64 80 35.000  
## 65 81 167.500  
## 66 82 159.415  
## 67 84 39.000  
## 68 85 70.000  
## 69 86 160.165  
## 70 87 25.000

aggregate(sales$Unit.Cost ~ sales$Customer.Age, data = sales, FUN = max)

## sales$Customer.Age sales$Unit.Cost  
## 1 17 1701.00  
## 2 18 2295.00  
## 3 19 2443.00  
## 4 20 2443.00  
## 5 21 2384.00  
## 6 22 2443.00  
## 7 23 2443.00  
## 8 24 2443.00  
## 9 25 2443.00  
## 10 26 2760.00  
## 11 27 2443.00  
## 12 28 2443.00  
## 13 29 2443.00  
## 14 30 2443.00  
## 15 31 2443.00  
## 16 32 3120.00  
## 17 33 2443.00  
## 18 34 2443.00  
## 19 35 3120.00  
## 20 36 2384.00  
## 21 37 2443.00  
## 22 38 2443.00  
## 23 39 2443.00  
## 24 40 2443.00  
## 25 41 2443.00  
## 26 42 2443.00  
## 27 43 2443.00  
## 28 44 2443.00  
## 29 45 2443.00  
## 30 46 2443.00  
## 31 47 2443.00  
## 32 48 2443.00  
## 33 49 2443.00  
## 34 50 3240.00  
## 35 51 2443.00  
## 36 52 2443.00  
## 37 53 2443.00  
## 38 54 2443.00  
## 39 55 2384.00  
## 40 56 2640.00  
## 41 57 2320.00  
## 42 58 2320.00  
## 43 59 2295.00  
## 44 60 2295.00  
## 45 61 2320.00  
## 46 62 3000.00  
## 47 63 2443.00  
## 48 64 2320.00  
## 49 65 2320.00  
## 50 66 2384.00  
## 51 67 592.67  
## 52 68 621.00  
## 53 69 1147.50  
## 54 70 550.00  
## 55 71 685.00  
## 56 72 2320.00  
## 57 73 1150.00  
## 58 74 513.33  
## 59 75 773.33  
## 60 76 140.00  
## 61 77 423.33  
## 62 78 595.00  
## 63 79 159.00  
## 64 80 473.00  
## 65 81 350.00  
## 66 82 233.33  
## 67 84 265.00  
## 68 85 455.00  
## 69 86 1842.00  
## 70 87 80.00

aggregate(sales$Unit.Cost ~ sales$Customer.Age, data = sales, FUN = min)

## sales$Customer.Age sales$Unit.Cost  
## 1 17 1.00  
## 2 18 0.67  
## 3 19 0.67  
## 4 20 1.67  
## 5 21 2.00  
## 6 22 1.67  
## 7 23 0.67  
## 8 24 0.67  
## 9 25 0.67  
## 10 26 0.67  
## 11 27 0.67  
## 12 28 1.67  
## 13 29 1.67  
## 14 30 1.00  
## 15 31 0.67  
## 16 32 0.67  
## 17 33 1.00  
## 18 34 1.00  
## 19 35 0.67  
## 20 36 0.67  
## 21 37 1.00  
## 22 38 0.67  
## 23 39 1.67  
## 24 40 1.00  
## 25 41 0.67  
## 26 42 1.00  
## 27 43 0.67  
## 28 44 0.67  
## 29 45 1.00  
## 30 46 1.67  
## 31 47 1.00  
## 32 48 0.67  
## 33 49 1.00  
## 34 50 0.67  
## 35 51 1.67  
## 36 52 0.67  
## 37 53 1.33  
## 38 54 1.00  
## 39 55 1.00  
## 40 56 1.67  
## 41 57 2.50  
## 42 58 0.67  
## 43 59 2.50  
## 44 60 0.67  
## 45 61 0.67  
## 46 62 1.67  
## 47 63 2.50  
## 48 64 1.33  
## 49 65 11.00  
## 50 66 2.50  
## 51 67 10.00  
## 52 68 7.00  
## 53 69 2.50  
## 54 70 5.00  
## 55 71 7.00  
## 56 72 21.67  
## 57 73 106.00  
## 58 74 130.00  
## 59 75 6.67  
## 60 76 31.67  
## 61 77 35.00  
## 62 78 4.50  
## 63 79 20.67  
## 64 80 30.00  
## 65 81 68.00  
## 66 82 85.50  
## 67 84 0.67  
## 68 85 2.50  
## 69 86 75.00  
## 70 87 7.00

aggregate(sales$Unit.Price ~ sales$Customer.Age, data = sales, FUN = mean)

## sales$Customer.Age sales$Unit.Price  
## 1 17 234.87675  
## 2 18 302.82040  
## 3 19 294.68457  
## 4 20 255.90101  
## 5 21 265.70266  
## 6 22 329.25231  
## 7 23 277.84568  
## 8 24 384.91901  
## 9 25 385.90606  
## 10 26 348.50312  
## 11 27 390.62911  
## 12 28 451.23036  
## 13 29 426.32226  
## 14 30 396.22951  
## 15 31 436.43369  
## 16 32 393.46914  
## 17 33 405.59703  
## 18 34 410.24841  
## 19 35 443.81197  
## 20 36 379.39706  
## 21 37 397.19424  
## 22 38 419.22676  
## 23 39 455.94981  
## 24 40 436.95937  
## 25 41 419.51768  
## 26 42 443.19778  
## 27 43 429.98562  
## 28 44 397.92554  
## 29 45 432.53701  
## 30 46 380.81904  
## 31 47 360.30487  
## 32 48 409.82058  
## 33 49 336.41788  
## 34 50 374.24315  
## 35 51 431.77686  
## 36 52 391.94577  
## 37 53 446.59039  
## 38 54 380.24826  
## 39 55 340.39672  
## 40 56 387.81194  
## 41 57 322.70221  
## 42 58 369.69189  
## 43 59 328.91219  
## 44 60 311.87244  
## 45 61 406.17636  
## 46 62 366.23709  
## 47 63 318.34188  
## 48 64 341.16667  
## 49 65 328.01333  
## 50 66 347.43478  
## 51 67 231.16667  
## 52 68 139.98851  
## 53 69 297.65789  
## 54 70 157.29167  
## 55 71 205.94048  
## 56 72 676.96667  
## 57 73 554.45833  
## 58 74 373.66667  
## 59 75 356.56667  
## 60 76 115.33333  
## 61 77 206.85417  
## 62 78 169.25758  
## 63 79 96.36667  
## 64 80 272.50000  
## 65 81 250.16667  
## 66 82 186.00000  
## 67 84 96.68519  
## 68 85 170.31250  
## 69 86 896.91667  
## 70 87 57.44444

aggregate(sales$Unit.Price ~ sales$Customer.Age, data = sales, FUN = median)

## sales$Customer.Age sales$Unit.Price  
## 1 17 95.66667  
## 2 18 135.00000  
## 3 19 122.00000  
## 4 20 98.50000  
## 5 21 104.75000  
## 6 22 120.00000  
## 7 23 123.00000  
## 8 24 174.50000  
## 9 25 189.00000  
## 10 26 153.25000  
## 11 27 213.33333  
## 12 28 239.00000  
## 13 29 225.33333  
## 14 30 211.00000  
## 15 31 229.66667  
## 16 32 193.66667  
## 17 33 200.00000  
## 18 34 220.00000  
## 19 35 209.00000  
## 20 36 183.33333  
## 21 37 222.00000  
## 22 38 223.16667  
## 23 39 225.66667  
## 24 40 179.75000  
## 25 41 168.00000  
## 26 42 205.91667  
## 27 43 199.50000  
## 28 44 175.33333  
## 29 45 201.25000  
## 30 46 160.00000  
## 31 47 138.00000  
## 32 48 169.33333  
## 33 49 139.00000  
## 34 50 152.00000  
## 35 51 182.50000  
## 36 52 175.25000  
## 37 53 224.50000  
## 38 54 188.50000  
## 39 55 140.75000  
## 40 56 161.83333  
## 41 57 150.00000  
## 42 58 179.00000  
## 43 59 154.75000  
## 44 60 121.00000  
## 45 61 188.50000  
## 46 62 133.25000  
## 47 63 99.00000  
## 48 64 134.00000  
## 49 65 140.75000  
## 50 66 67.00000  
## 51 67 149.75000  
## 52 68 75.00000  
## 53 69 135.00000  
## 54 70 63.66667  
## 55 71 65.50000  
## 56 72 422.83333  
## 57 73 239.25000  
## 58 74 373.66667  
## 59 75 245.00000  
## 60 76 115.33333  
## 61 77 162.00000  
## 62 78 84.00000  
## 63 79 102.00000  
## 64 80 46.50000  
## 65 81 232.00000  
## 66 82 186.00000  
## 67 84 53.00000  
## 68 85 92.00000  
## 69 86 201.66667  
## 70 87 33.66667

aggregate(sales$Unit.Price ~ sales$Customer.Age, data = sales, FUN = max)

## sales$Customer.Age sales$Unit.Price  
## 1 17 1618.0000  
## 2 18 2379.0000  
## 3 19 2587.0000  
## 4 20 3397.0000  
## 5 21 2592.0000  
## 6 22 2837.0000  
## 7 23 3242.0000  
## 8 24 3423.0000  
## 9 25 3378.0000  
## 10 26 3428.0000  
## 11 27 3261.0000  
## 12 28 3163.0000  
## 13 29 2932.0000  
## 14 30 3081.0000  
## 15 31 3495.0000  
## 16 32 3887.0000  
## 17 33 3215.0000  
## 18 34 3463.0000  
## 19 35 4113.0000  
## 20 36 3098.0000  
## 21 37 3307.0000  
## 22 38 3487.0000  
## 23 39 3427.0000  
## 24 40 3326.0000  
## 25 41 3245.0000  
## 26 42 3268.0000  
## 27 43 2927.0000  
## 28 44 3161.0000  
## 29 45 3455.0000  
## 30 46 3339.0000  
## 31 47 3098.0000  
## 32 48 3099.0000  
## 33 49 3237.0000  
## 34 50 5082.0000  
## 35 51 3262.0000  
## 36 52 3035.0000  
## 37 53 3264.0000  
## 38 54 2641.0000  
## 39 55 2703.0000  
## 40 56 4008.0000  
## 41 57 2259.0000  
## 42 58 2408.0000  
## 43 59 2271.0000  
## 44 60 2575.0000  
## 45 61 2612.0000  
## 46 62 3787.0000  
## 47 63 2552.0000  
## 48 64 2402.0000  
## 49 65 2336.0000  
## 50 66 2461.0000  
## 51 67 732.6667  
## 52 68 601.5000  
## 53 69 1256.0000  
## 54 70 874.0000  
## 55 71 774.0000  
## 56 72 2858.0000  
## 57 73 1611.0000  
## 58 74 597.3333  
## 59 75 922.0000  
## 60 76 187.0000  
## 61 77 596.6667  
## 62 78 754.0000  
## 63 79 180.5000  
## 64 80 739.0000  
## 65 81 479.5000  
## 66 82 230.0000  
## 67 84 343.0000  
## 68 85 546.0000  
## 69 86 3103.0000  
## 70 87 127.6667

aggregate(sales$Unit.Price ~ sales$Customer.Age, data = sales, FUN = min)

## sales$Customer.Age sales$Unit.Price  
## 1 17 1.500000  
## 2 18 0.666667  
## 3 19 0.666667  
## 4 20 2.000000  
## 5 21 2.000000  
## 6 22 1.666667  
## 7 23 1.000000  
## 8 24 1.000000  
## 9 25 1.000000  
## 10 26 1.000000  
## 11 27 1.000000  
## 12 28 2.000000  
## 13 29 1.666667  
## 14 30 1.333333  
## 15 31 1.000000  
## 16 32 0.666667  
## 17 33 1.333333  
## 18 34 1.500000  
## 19 35 1.000000  
## 20 36 1.000000  
## 21 37 1.500000  
## 22 38 0.666667  
## 23 39 1.666667  
## 24 40 1.500000  
## 25 41 1.000000  
## 26 42 1.500000  
## 27 43 0.666667  
## 28 44 1.333333  
## 29 45 1.000000  
## 30 46 2.000000  
## 31 47 1.333333  
## 32 48 1.000000  
## 33 49 1.500000  
## 34 50 1.333333  
## 35 51 2.000000  
## 36 52 1.000000  
## 37 53 1.666667  
## 38 54 1.000000  
## 39 55 1.000000  
## 40 56 2.000000  
## 41 57 3.000000  
## 42 58 1.000000  
## 43 59 3.000000  
## 44 60 1.000000  
## 45 61 0.666667  
## 46 62 2.000000  
## 47 63 3.000000  
## 48 64 2.000000  
## 49 65 13.000000  
## 50 66 2.500000  
## 51 67 12.000000  
## 52 68 11.666667  
## 53 69 3.000000  
## 54 70 5.000000  
## 55 71 10.000000  
## 56 72 23.000000  
## 57 73 128.333333  
## 58 74 150.000000  
## 59 75 8.333333  
## 60 76 43.666667  
## 61 77 42.666667  
## 62 78 6.000000  
## 63 79 23.333333  
## 64 80 32.000000  
## 65 81 59.000000  
## 66 82 142.000000  
## 67 84 1.000000  
## 68 85 3.000000  
## 69 86 81.333333  
## 70 87 11.000000

aggregate(sales$Cost ~ sales$Customer.Age, data = sales, FUN = mean)

## sales$Customer.Age sales$Cost  
## 1 17 363.8207  
## 2 18 437.8337  
## 3 19 431.9785  
## 4 20 361.0134  
## 5 21 373.1159  
## 6 22 456.0042  
## 7 23 421.7617  
## 8 24 556.2538  
## 9 25 577.8988  
## 10 26 527.8701  
## 11 27 591.9139  
## 12 28 691.2827  
## 13 29 651.4408  
## 14 30 592.1852  
## 15 31 652.2188  
## 16 32 578.0509  
## 17 33 583.7794  
## 18 34 637.3730  
## 19 35 619.4302  
## 20 36 584.6063  
## 21 37 602.8505  
## 22 38 623.4382  
## 23 39 658.5117  
## 24 40 617.3129  
## 25 41 614.1902  
## 26 42 660.2393  
## 27 43 639.2810  
## 28 44 599.8595  
## 29 45 648.2793  
## 30 46 574.3223  
## 31 47 530.8604  
## 32 48 594.2537  
## 33 49 508.9887  
## 34 50 512.9847  
## 35 51 611.9908  
## 36 52 572.9026  
## 37 53 674.9355  
## 38 54 542.7083  
## 39 55 505.2340  
## 40 56 560.7669  
## 41 57 529.8860  
## 42 58 603.3189  
## 43 59 456.3656  
## 44 60 429.0950  
## 45 61 592.4767  
## 46 62 547.4789  
## 47 63 454.7350  
## 48 64 464.2762  
## 49 65 520.9800  
## 50 66 417.4348  
## 51 67 413.8636  
## 52 68 235.3103  
## 53 69 444.4211  
## 54 70 262.5833  
## 55 71 259.5000  
## 56 72 877.7000  
## 57 73 595.0000  
## 58 74 835.0000  
## 59 75 567.0000  
## 60 76 117.5000  
## 61 77 316.1250  
## 62 78 205.2727  
## 63 79 161.4000  
## 64 80 189.3333  
## 65 81 282.8333  
## 66 82 435.5000  
## 67 84 170.0000  
## 68 85 216.1250  
## 69 86 703.0000  
## 70 87 109.6667

aggregate(sales$Cost ~ sales$Customer.Age, data = sales, FUN = median)

## sales$Customer.Age sales$Cost  
## 1 17 145.0  
## 2 18 171.0  
## 3 19 183.0  
## 4 20 135.0  
## 5 21 150.0  
## 6 22 162.0  
## 7 23 162.0  
## 8 24 245.0  
## 9 25 252.0  
## 10 26 225.0  
## 11 27 315.0  
## 12 28 365.0  
## 13 29 385.0  
## 14 30 300.0  
## 15 31 418.0  
## 16 32 280.0  
## 17 33 300.0  
## 18 34 360.0  
## 19 35 300.0  
## 20 36 280.0  
## 21 37 385.0  
## 22 38 318.0  
## 23 39 350.0  
## 24 40 250.0  
## 25 41 250.0  
## 26 42 370.5  
## 27 43 300.0  
## 28 44 261.0  
## 29 45 344.0  
## 30 46 216.0  
## 31 47 200.0  
## 32 48 245.0  
## 33 49 207.0  
## 34 50 220.0  
## 35 51 260.0  
## 36 52 270.0  
## 37 53 280.0  
## 38 54 277.5  
## 39 55 200.0  
## 40 56 280.0  
## 41 57 290.0  
## 42 58 275.0  
## 43 59 244.0  
## 44 60 150.0  
## 45 61 285.0  
## 46 62 180.0  
## 47 63 135.0  
## 48 64 176.0  
## 49 65 206.5  
## 50 66 90.0  
## 51 67 225.0  
## 52 68 105.0  
## 53 69 135.0  
## 54 70 95.0  
## 55 71 118.5  
## 56 72 640.0  
## 57 73 456.0  
## 58 74 835.0  
## 59 75 347.5  
## 60 76 117.5  
## 61 77 139.5  
## 62 78 135.0  
## 63 79 120.0  
## 64 80 60.0  
## 65 81 167.5  
## 66 82 435.5  
## 67 84 60.0  
## 68 85 70.0  
## 69 86 431.0  
## 70 87 75.0

aggregate(sales$Cost ~ sales$Customer.Age, data = sales, FUN = max)

## sales$Customer.Age sales$Cost  
## 1 17 2443  
## 2 18 2443  
## 3 19 2443  
## 4 20 2443  
## 5 21 2443  
## 6 22 2443  
## 7 23 3120  
## 8 24 2443  
## 9 25 2443  
## 10 26 2760  
## 11 27 2443  
## 12 28 2443  
## 13 29 2443  
## 14 30 2443  
## 15 31 2443  
## 16 32 3120  
## 17 33 2443  
## 18 34 2443  
## 19 35 3120  
## 20 36 2443  
## 21 37 2443  
## 22 38 2443  
## 23 39 2443  
## 24 40 3240  
## 25 41 2443  
## 26 42 2760  
## 27 43 2443  
## 28 44 2443  
## 29 45 2443  
## 30 46 3600  
## 31 47 2443  
## 32 48 2443  
## 33 49 2443  
## 34 50 3240  
## 35 51 2443  
## 36 52 2443  
## 37 53 3600  
## 38 54 2443  
## 39 55 2443  
## 40 56 2640  
## 41 57 2443  
## 42 58 2443  
## 43 59 2443  
## 44 60 2320  
## 45 61 2384  
## 46 62 3000  
## 47 63 2443  
## 48 64 2320  
## 49 65 2320  
## 50 66 2384  
## 51 67 1778  
## 52 68 1242  
## 53 69 2295  
## 54 70 1100  
## 55 71 780  
## 56 72 2384  
## 57 73 1150  
## 58 74 1540  
## 59 75 2320  
## 60 76 140  
## 61 77 1270  
## 62 78 675  
## 63 79 318  
## 64 80 473  
## 65 81 700  
## 66 82 700  
## 67 84 795  
## 68 85 910  
## 69 86 1842  
## 70 87 240

aggregate(sales$Cost ~ sales$Customer.Age, data = sales, FUN = min)

## sales$Customer.Age sales$Cost  
## 1 17 2  
## 2 18 2  
## 3 19 2  
## 4 20 2  
## 5 21 4  
## 6 22 4  
## 7 23 2  
## 8 24 2  
## 9 25 2  
## 10 26 2  
## 11 27 2  
## 12 28 5  
## 13 29 2  
## 14 30 2  
## 15 31 2  
## 16 32 2  
## 17 33 2  
## 18 34 2  
## 19 35 2  
## 20 36 2  
## 21 37 2  
## 22 38 2  
## 23 39 4  
## 24 40 2  
## 25 41 2  
## 26 42 2  
## 27 43 2  
## 28 44 2  
## 29 45 2  
## 30 46 5  
## 31 47 2  
## 32 48 2  
## 33 49 2  
## 34 50 2  
## 35 51 2  
## 36 52 2  
## 37 53 2  
## 38 54 2  
## 39 55 2  
## 40 56 5  
## 41 57 5  
## 42 58 2  
## 43 59 5  
## 44 60 2  
## 45 61 2  
## 46 62 5  
## 47 63 5  
## 48 64 4  
## 49 65 11  
## 50 66 5  
## 51 67 20  
## 52 68 21  
## 53 69 5  
## 54 70 5  
## 55 71 7  
## 56 72 35  
## 57 73 318  
## 58 74 130  
## 59 75 20  
## 60 76 95  
## 61 77 40  
## 62 78 9  
## 63 79 28  
## 64 80 35  
## 65 81 68  
## 66 82 171  
## 67 84 2  
## 68 85 5  
## 69 86 108  
## 70 87 14

aggregate(sales$Revenue ~ sales$Customer.Age, data = sales, FUN = mean)

## sales$Customer.Age sales$Revenue  
## 1 17 410.1597  
## 2 18 503.9690  
## 3 19 482.2463  
## 4 20 415.0067  
## 5 21 431.6072  
## 6 22 515.3753  
## 7 23 468.6938  
## 8 24 628.2937  
## 9 25 649.5103  
## 10 26 588.4232  
## 11 27 653.8327  
## 12 28 753.3626  
## 13 29 712.0818  
## 14 30 658.4676  
## 15 31 715.3940  
## 16 32 642.3761  
## 17 33 645.1479  
## 18 34 705.1429  
## 19 35 685.9464  
## 20 36 644.1129  
## 21 37 672.0766  
## 22 38 689.7610  
## 23 39 728.8891  
## 24 40 692.6401  
## 25 41 693.1251  
## 26 42 738.0644  
## 27 43 706.0918  
## 28 44 675.6795  
## 29 45 714.1543  
## 30 46 642.2490  
## 31 47 592.8079  
## 32 48 663.0429  
## 33 49 562.2707  
## 34 50 580.1033  
## 35 51 689.8480  
## 36 52 646.1801  
## 37 53 744.5444  
## 38 54 607.6276  
## 39 55 560.5824  
## 40 56 620.0405  
## 41 57 590.0368  
## 42 58 651.4703  
## 43 59 504.3763  
## 44 60 473.0559  
## 45 61 650.1163  
## 46 62 608.7817  
## 47 63 502.2393  
## 48 64 525.7905  
## 49 65 547.2200  
## 50 66 431.9565  
## 51 67 500.6818  
## 52 68 284.7931  
## 53 69 540.8947  
## 54 70 349.4167  
## 55 71 315.2143  
## 56 72 996.1000  
## 57 73 799.0000  
## 58 74 971.0000  
## 59 75 714.5000  
## 60 76 159.0000  
## 61 77 420.2500  
## 62 78 247.0909  
## 63 79 182.6000  
## 64 80 288.0000  
## 65 81 397.0000  
## 66 82 487.0000  
## 67 84 205.3333  
## 68 85 262.8750  
## 69 86 1095.7500  
## 70 87 168.6667

aggregate(sales$Revenue ~ sales$Customer.Age, data = sales, FUN = median)

## sales$Customer.Age sales$Revenue  
## 1 17 165.0  
## 2 18 213.0  
## 3 19 235.0  
## 4 20 159.5  
## 5 21 190.5  
## 6 22 202.5  
## 7 23 191.0  
## 8 24 285.5  
## 9 25 317.0  
## 10 26 266.0  
## 11 27 398.5  
## 12 28 438.0  
## 13 29 457.5  
## 14 30 377.5  
## 15 31 469.0  
## 16 32 361.0  
## 17 33 361.0  
## 18 34 416.0  
## 19 35 352.0  
## 20 36 356.5  
## 21 37 455.0  
## 22 38 394.5  
## 23 39 417.0  
## 24 40 314.5  
## 25 41 293.0  
## 26 42 447.5  
## 27 43 387.5  
## 28 44 332.0  
## 29 45 394.0  
## 30 46 266.0  
## 31 47 239.0  
## 32 48 302.0  
## 33 49 250.0  
## 34 50 253.0  
## 35 51 301.5  
## 36 52 311.5  
## 37 53 360.5  
## 38 54 339.0  
## 39 55 245.0  
## 40 56 357.5  
## 41 57 352.0  
## 42 58 319.0  
## 43 59 272.0  
## 44 60 175.0  
## 45 61 325.0  
## 46 62 216.0  
## 47 63 161.0  
## 48 64 215.0  
## 49 65 239.0  
## 50 66 106.0  
## 51 67 281.0  
## 52 68 134.0  
## 53 69 168.0  
## 54 70 125.5  
## 55 71 125.0  
## 56 72 748.0  
## 57 73 600.0  
## 58 74 971.0  
## 59 75 490.0  
## 60 76 159.0  
## 61 77 183.0  
## 62 78 158.0  
## 63 79 144.0  
## 64 80 93.0  
## 65 81 232.0  
## 66 82 487.0  
## 67 84 70.0  
## 68 85 92.0  
## 69 86 596.5  
## 70 87 101.0

aggregate(sales$Revenue ~ sales$Customer.Age, data = sales, FUN = max)

## sales$Customer.Age sales$Revenue  
## 1 17 3180  
## 2 18 3193  
## 3 19 2821  
## 4 20 3397  
## 5 21 2839  
## 6 22 3312  
## 7 23 4550  
## 8 24 3561  
## 9 25 3378  
## 10 26 3428  
## 11 27 3261  
## 12 28 3406  
## 13 29 3219  
## 14 30 3415  
## 15 31 3537  
## 16 32 3887  
## 17 33 3260  
## 18 34 3520  
## 19 35 4113  
## 20 36 3284  
## 21 37 3307  
## 22 38 3487  
## 23 39 3616  
## 24 40 4219  
## 25 41 3419  
## 26 42 3703  
## 27 43 3282  
## 28 44 3284  
## 29 45 3681  
## 30 46 4923  
## 31 47 3260  
## 32 48 3158  
## 33 49 3345  
## 34 50 5082  
## 35 51 3262  
## 36 52 3344  
## 37 53 4266  
## 38 54 3173  
## 39 55 2968  
## 40 56 4008  
## 41 57 3113  
## 42 58 2620  
## 43 59 2702  
## 44 60 2617  
## 45 61 2697  
## 46 62 3787  
## 47 63 2552  
## 48 64 2723  
## 49 65 2462  
## 50 66 2461  
## 51 67 2198  
## 52 68 1203  
## 53 69 2512  
## 54 70 1748  
## 55 71 1042  
## 56 72 2858  
## 57 73 1611  
## 58 74 1792  
## 59 75 2641  
## 60 76 187  
## 61 77 1790  
## 62 78 778  
## 63 79 361  
## 64 80 739  
## 65 81 959  
## 66 82 690  
## 67 84 1029  
## 68 85 1092  
## 69 86 3103  
## 70 87 383

aggregate(sales$Revenue ~ sales$Customer.Age, data = sales, FUN = min)

## sales$Customer.Age sales$Revenue  
## 1 17 3  
## 2 18 2  
## 3 19 2  
## 4 20 2  
## 5 21 4  
## 6 22 3  
## 7 23 3  
## 8 24 3  
## 9 25 3  
## 10 26 3  
## 11 27 3  
## 12 28 5  
## 13 29 3  
## 14 30 3  
## 15 31 3  
## 16 32 2  
## 17 33 3  
## 18 34 2  
## 19 35 3  
## 20 36 2  
## 21 37 3  
## 22 38 2  
## 23 39 5  
## 24 40 3  
## 25 41 3  
## 26 42 3  
## 27 43 2  
## 28 44 4  
## 29 45 2  
## 30 46 6  
## 31 47 3  
## 32 48 3  
## 33 49 3  
## 34 50 3  
## 35 51 3  
## 36 52 3  
## 37 53 3  
## 38 54 2  
## 39 55 2  
## 40 56 5  
## 41 57 5  
## 42 58 3  
## 43 59 5  
## 44 60 2  
## 45 61 2  
## 46 62 6  
## 47 63 6  
## 48 64 6  
## 49 65 13  
## 50 66 5  
## 51 67 24  
## 52 68 24  
## 53 69 6  
## 54 70 5  
## 55 71 10  
## 56 72 54  
## 57 73 385  
## 58 74 150  
## 59 75 25  
## 60 76 131  
## 61 77 51  
## 62 78 12  
## 63 79 32  
## 64 80 32  
## 65 81 59  
## 66 82 284  
## 67 84 3  
## 68 85 6  
## 69 86 87  
## 70 87 22

aggregate(sales$Unit\_Margin ~ sales$Customer.Age, data = sales, FUN = mean)

## sales$Customer.Age sales$Unit\_Margin  
## 1 17 27.73846  
## 2 18 39.23951  
## 3 19 31.03289  
## 4 20 33.25282  
## 5 21 36.73479  
## 6 22 36.30958  
## 7 23 29.17001  
## 8 24 42.92482  
## 9 25 42.72785  
## 10 26 34.93351  
## 11 27 39.54832  
## 12 28 35.89831  
## 13 29 35.10184  
## 14 30 37.98823  
## 15 31 38.41863  
## 16 32 41.03138  
## 17 33 37.94185  
## 18 34 40.66389  
## 19 35 42.21082  
## 20 36 35.20568  
## 21 37 42.55742  
## 22 38 40.36098  
## 23 39 44.78015  
## 24 40 46.63001  
## 25 41 47.52067  
## 26 42 45.65404  
## 27 43 37.71071  
## 28 44 45.79162  
## 29 45 40.34616  
## 30 46 39.68077  
## 31 47 37.14185  
## 32 48 42.53989  
## 33 49 31.56869  
## 34 50 46.28110  
## 35 51 49.25161  
## 36 52 44.73548  
## 37 53 42.66843  
## 38 54 40.75662  
## 39 55 36.97922  
## 40 56 34.00241  
## 41 57 33.56066  
## 42 58 28.68741  
## 43 59 31.75014  
## 44 60 29.84724  
## 45 61 31.61601  
## 46 62 36.01315  
## 47 63 29.05000  
## 48 64 31.39838  
## 49 65 17.22373  
## 50 66 15.16696  
## 51 67 40.56030  
## 52 68 23.87920  
## 53 69 46.34263  
## 54 70 38.38208  
## 55 71 33.53619  
## 56 72 111.33267  
## 57 73 149.79083  
## 58 74 52.00167  
## 59 75 82.39967  
## 60 76 29.49833  
## 61 77 48.12542  
## 62 78 30.12121  
## 63 79 12.23267  
## 64 80 93.16667  
## 65 81 69.41667  
## 66 82 26.58500  
## 67 84 17.96296  
## 68 85 30.12500  
## 69 86 337.58417  
## 70 87 20.11111

aggregate(sales$Unit\_Margin ~ sales$Customer.Age, data = sales, FUN = median)

## sales$Customer.Age sales$Unit\_Margin  
## 1 17 10.000000  
## 2 18 12.500000  
## 3 19 12.000000  
## 4 20 12.000000  
## 5 21 13.003333  
## 6 22 12.668333  
## 7 23 10.583334  
## 8 24 14.581667  
## 9 25 16.333333  
## 10 26 13.000000  
## 11 27 14.585000  
## 12 28 15.333333  
## 13 29 15.000000  
## 14 30 13.998333  
## 15 31 13.500000  
## 16 32 16.500000  
## 17 33 14.831666  
## 18 34 15.000000  
## 19 35 16.000000  
## 20 36 14.668333  
## 21 37 16.583333  
## 22 38 17.001666  
## 23 39 15.500000  
## 24 40 17.500000  
## 25 41 16.000000  
## 26 42 16.500000  
## 27 43 15.335000  
## 28 44 17.500000  
## 29 45 13.000000  
## 30 46 13.500000  
## 31 47 12.666667  
## 32 48 15.000000  
## 33 49 11.336667  
## 34 50 13.996667  
## 35 51 15.418334  
## 36 52 15.000000  
## 37 53 15.500000  
## 38 54 17.416666  
## 39 55 14.336667  
## 40 56 15.750000  
## 41 57 12.500000  
## 42 58 14.330000  
## 43 59 12.416666  
## 44 60 10.000000  
## 45 61 14.665000  
## 46 62 10.335000  
## 47 63 15.000000  
## 48 64 13.000000  
## 49 65 11.835000  
## 50 66 6.000000  
## 51 67 25.583334  
## 52 68 11.996667  
## 53 69 25.000000  
## 54 70 10.500000  
## 55 71 15.668333  
## 56 72 18.000000  
## 57 73 57.915000  
## 58 74 52.001666  
## 59 75 58.500000  
## 60 76 29.498333  
## 61 77 24.500000  
## 62 78 20.000000  
## 63 79 9.000000  
## 64 80 16.500000  
## 65 81 64.500000  
## 66 82 26.585000  
## 67 84 6.000000  
## 68 85 22.000000  
## 69 86 55.168333  
## 70 87 8.666667

aggregate(sales$Unit\_Margin ~ sales$Customer.Age, data = sales, FUN = max)

## sales$Customer.Age sales$Unit\_Margin  
## 1 17 483.00000  
## 2 18 748.00000  
## 3 19 527.00000  
## 4 20 954.00000  
## 5 21 1026.00000  
## 6 22 828.00000  
## 7 23 922.00000  
## 8 24 1128.00000  
## 9 25 1058.00000  
## 10 26 951.00000  
## 11 27 941.00000  
## 12 28 779.00000  
## 13 29 750.00000  
## 14 30 892.00000  
## 15 31 1111.00000  
## 16 32 767.00000  
## 17 33 831.00000  
## 18 34 1079.00000  
## 19 35 993.00000  
## 20 36 803.00000  
## 21 37 864.00000  
## 22 38 1167.00000  
## 23 39 1043.00000  
## 24 40 1031.00000  
## 25 41 861.00000  
## 26 42 973.00000  
## 27 43 630.00000  
## 28 44 777.00000  
## 29 45 1071.00000  
## 30 46 1309.00000  
## 31 47 803.00000  
## 32 48 754.00000  
## 33 49 917.00000  
## 34 50 1842.00000  
## 35 51 942.00000  
## 36 52 918.00000  
## 37 53 880.00000  
## 38 54 430.00000  
## 39 55 525.00000  
## 40 56 1368.00000  
## 41 57 637.00000  
## 42 58 297.00000  
## 43 59 449.00000  
## 44 60 517.00000  
## 45 61 454.00000  
## 46 62 787.00000  
## 47 63 430.00000  
## 48 64 718.00000  
## 49 65 209.00000  
## 50 66 77.00000  
## 51 67 155.00000  
## 52 68 105.67000  
## 53 69 132.00000  
## 54 70 324.00000  
## 55 71 89.00000  
## 56 72 538.00000  
## 57 73 461.00000  
## 58 74 84.00333  
## 59 75 292.00000  
## 60 76 47.00000  
## 61 77 173.33667  
## 62 78 159.00000  
## 63 79 24.00000  
## 64 80 266.00000  
## 65 81 139.00000  
## 66 82 56.50000  
## 67 84 78.00000  
## 68 85 91.00000  
## 69 86 1261.00000  
## 70 87 47.66667

aggregate(sales$Unit\_Margin ~ sales$Customer.Age, data = sales, FUN = min)

## sales$Customer.Age sales$Unit\_Margin  
## 1 17 -208.000000  
## 2 18 -190.000000  
## 3 19 -248.000000  
## 4 20 -411.000000  
## 5 21 -375.000000  
## 6 22 -319.000000  
## 7 23 -214.000000  
## 8 24 -293.000000  
## 9 25 -385.000000  
## 10 26 -424.000000  
## 11 27 -620.000000  
## 12 28 -777.000000  
## 13 29 -920.000000  
## 14 30 -932.000000  
## 15 31 -937.000000  
## 16 32 -822.000000  
## 17 33 -700.000000  
## 18 34 -575.000000  
## 19 35 -555.000000  
## 20 36 -394.000000  
## 21 37 -741.000000  
## 22 38 -752.000000  
## 23 39 -303.500000  
## 24 40 -550.000000  
## 25 41 -805.000000  
## 26 42 -642.000000  
## 27 43 -666.000000  
## 28 44 -654.000000  
## 29 45 -472.000000  
## 30 46 -700.000000  
## 31 47 -513.000000  
## 32 48 -527.000000  
## 33 49 -688.000000  
## 34 50 -410.000000  
## 35 51 -618.000000  
## 36 52 -609.000000  
## 37 53 -733.000000  
## 38 54 -444.000000  
## 39 55 -742.000000  
## 40 56 -363.000000  
## 41 57 -163.000000  
## 42 58 -141.000000  
## 43 59 -293.000000  
## 44 60 -436.000000  
## 45 61 -665.000000  
## 46 62 -174.500000  
## 47 63 -288.000000  
## 48 64 -395.000000  
## 49 65 -210.000000  
## 50 66 -41.330000  
## 51 67 0.000000  
## 52 68 -19.500000  
## 53 69 0.500000  
## 54 70 -26.500000  
## 55 71 2.336667  
## 56 72 -48.670000  
## 57 73 22.333333  
## 58 74 20.000000  
## 59 75 1.663333  
## 60 76 11.996667  
## 61 77 7.666667  
## 62 78 1.500000  
## 63 79 2.663333  
## 64 80 -3.000000  
## 65 81 -10.000000  
## 66 82 -3.330000  
## 67 84 -10.666667  
## 68 85 0.500000  
## 69 86 -21.000000  
## 70 87 4.000000

aggregate(sales$Unit\_Margin\_percent ~ sales$Customer.Age, data = sales, FUN = mean)

## sales$Customer.Age sales$Unit\_Margin\_percent  
## 1 17 14.44263  
## 2 18 14.86441  
## 3 19 13.82936  
## 4 20 15.37810  
## 5 21 14.96045  
## 6 22 14.18071  
## 7 23 13.27859  
## 8 24 13.70026  
## 9 25 14.42946  
## 10 26 13.82957  
## 11 27 13.05947  
## 12 28 12.34609  
## 13 29 11.97791  
## 14 30 12.76331  
## 15 31 11.77716  
## 16 32 13.84977  
## 17 33 12.89439  
## 18 34 12.86276  
## 19 35 12.55605  
## 20 36 12.96797  
## 21 37 13.25334  
## 22 38 13.17738  
## 23 39 12.66963  
## 24 40 14.15326  
## 25 41 13.91629  
## 26 42 13.73345  
## 27 43 13.30771  
## 28 44 14.41719  
## 29 45 12.41052  
## 30 46 14.14480  
## 31 47 13.80401  
## 32 48 14.36560  
## 33 49 12.28426  
## 34 50 14.16218  
## 35 51 14.51240  
## 36 52 13.62333  
## 37 53 13.22063  
## 38 54 13.92299  
## 39 55 14.18963  
## 40 56 13.34270  
## 41 57 11.95033  
## 42 58 11.79043  
## 43 59 12.08688  
## 44 60 14.00073  
## 45 61 11.79593  
## 46 62 13.18141  
## 47 63 15.10085  
## 48 64 15.25076  
## 49 65 12.00780  
## 50 66 10.19522  
## 51 67 17.40773  
## 52 68 17.90655  
## 53 69 19.13737  
## 54 70 19.10417  
## 55 71 19.28857  
## 56 72 15.03600  
## 57 73 23.61750  
## 58 74 13.69500  
## 59 75 25.10500  
## 60 76 26.30000  
## 61 77 21.43250  
## 62 78 17.80000  
## 63 79 12.26200  
## 64 80 20.69667  
## 65 81 22.24500  
## 66 82 19.17000  
## 67 84 18.72778  
## 68 85 19.63500  
## 69 86 14.29250  
## 70 87 33.14667

aggregate(sales$Unit\_Margin\_percent ~ sales$Customer.Age, data = sales, FUN = median)

## sales$Customer.Age sales$Unit\_Margin\_percent  
## 1 17 15.530  
## 2 18 15.560  
## 3 19 14.290  
## 4 20 15.660  
## 5 21 14.575  
## 6 22 15.120  
## 7 23 13.810  
## 8 24 14.875  
## 9 25 15.660  
## 10 26 15.260  
## 11 27 15.460  
## 12 28 14.290  
## 13 29 13.660  
## 14 30 14.630  
## 15 31 13.910  
## 16 32 15.090  
## 17 33 14.425  
## 18 34 14.120  
## 19 35 14.480  
## 20 36 14.790  
## 21 37 14.290  
## 22 38 14.290  
## 23 39 13.910  
## 24 40 15.380  
## 25 41 15.440  
## 26 42 15.215  
## 27 43 15.010  
## 28 44 15.620  
## 29 45 14.290  
## 30 46 16.180  
## 31 47 14.290  
## 32 48 15.775  
## 33 49 13.980  
## 34 50 15.790  
## 35 51 16.280  
## 36 52 15.810  
## 37 53 15.055  
## 38 54 15.625  
## 39 55 15.495  
## 40 56 14.640  
## 41 57 12.405  
## 42 58 12.540  
## 43 59 12.895  
## 44 60 15.790  
## 45 61 13.380  
## 46 62 13.040  
## 47 63 16.750  
## 48 64 16.280  
## 49 65 13.365  
## 50 66 6.830  
## 51 67 18.515  
## 52 68 17.150  
## 53 69 18.520  
## 54 70 16.985  
## 55 71 23.115  
## 56 72 11.580  
## 57 73 24.225  
## 58 74 13.695  
## 59 75 23.905  
## 60 76 26.300  
## 61 77 20.460  
## 62 78 15.790  
## 63 79 11.910  
## 64 80 35.480  
## 65 81 28.455  
## 66 82 19.170  
## 67 84 22.740  
## 68 85 20.095  
## 69 86 20.335  
## 70 87 36.360

aggregate(sales$Unit\_Margin\_percent ~ sales$Customer.Age, data = sales, FUN = max)

## sales$Customer.Age sales$Unit\_Margin\_percent  
## 1 17 44.00  
## 2 18 41.18  
## 3 19 41.67  
## 4 20 40.85  
## 5 21 43.82  
## 6 22 43.32  
## 7 23 42.86  
## 8 24 42.86  
## 9 25 43.73  
## 10 26 44.77  
## 11 27 42.08  
## 12 28 43.22  
## 13 29 44.79  
## 14 30 45.72  
## 15 31 46.11  
## 16 32 42.31  
## 17 33 43.51  
## 18 34 42.86  
## 19 35 43.26  
## 20 36 44.71  
## 21 37 41.67  
## 22 38 41.24  
## 23 39 42.99  
## 24 40 46.00  
## 25 41 43.19  
## 26 42 43.07  
## 27 43 42.15  
## 28 44 49.75  
## 29 45 43.81  
## 30 46 44.74  
## 31 47 43.26  
## 32 48 41.75  
## 33 49 45.03  
## 34 50 49.75  
## 35 51 42.35  
## 36 52 41.94  
## 37 53 45.10  
## 38 54 41.85  
## 39 55 40.59  
## 40 56 41.02  
## 41 57 38.46  
## 42 58 41.03  
## 43 59 41.67  
## 44 60 42.86  
## 45 61 38.60  
## 46 62 39.42  
## 47 63 40.68  
## 48 64 39.72  
## 49 65 28.57  
## 50 66 27.43  
## 51 67 30.69  
## 52 68 40.00  
## 53 69 29.36  
## 54 70 39.94  
## 55 71 36.36  
## 56 72 37.66  
## 57 73 28.62  
## 58 74 14.06  
## 59 75 41.59  
## 60 76 27.47  
## 61 77 29.05  
## 62 78 26.67  
## 63 79 16.67  
## 64 80 35.99  
## 65 81 34.62  
## 66 82 39.79  
## 67 84 33.00  
## 68 85 23.91  
## 69 86 40.64  
## 70 87 37.34

aggregate(sales$Unit\_Margin\_percent ~ sales$Customer.Age, data = sales, FUN = min)

## sales$Customer.Age sales$Unit\_Margin\_percent  
## 1 17 -41.89  
## 2 18 -34.94  
## 3 19 -34.42  
## 4 20 -30.14  
## 5 21 -45.90  
## 6 22 -38.86  
## 7 23 -36.36  
## 8 24 -56.86  
## 9 25 -54.73  
## 10 26 -55.63  
## 11 27 -68.67  
## 12 28 -60.71  
## 13 29 -65.24  
## 14 30 -61.80  
## 15 31 -64.75  
## 16 32 -66.99  
## 17 33 -58.02  
## 18 34 -50.33  
## 19 35 -56.63  
## 20 36 -44.77  
## 21 37 -51.42  
## 22 38 -47.96  
## 23 39 -50.00  
## 24 40 -49.09  
## 25 41 -60.09  
## 26 42 -39.04  
## 27 43 -49.33  
## 28 44 -39.85  
## 29 45 -57.30  
## 30 46 -47.54  
## 31 47 -38.42  
## 32 48 -34.26  
## 33 49 -45.62  
## 34 50 -33.76  
## 35 51 -35.48  
## 36 52 -55.50  
## 37 53 -46.93  
## 38 54 -37.24  
## 39 55 -56.77  
## 40 56 -36.02  
## 41 57 -23.12  
## 42 58 -15.45  
## 43 59 -31.39  
## 44 60 -27.19  
## 45 61 -40.80  
## 46 62 -17.93  
## 47 63 -36.36  
## 48 64 -20.52  
## 49 65 -22.11  
## 50 66 -6.37  
## 51 67 0.00  
## 52 68 -3.38  
## 53 69 8.64  
## 54 70 -6.98  
## 55 71 4.61  
## 56 72 -6.52  
## 57 73 17.40  
## 58 74 13.33  
## 59 75 12.15  
## 60 76 25.13  
## 61 77 12.50  
## 62 78 8.78  
## 63 79 8.82  
## 64 80 -9.38  
## 65 81 -16.95  
## 66 82 -1.45  
## 67 84 -11.94  
## 68 85 12.65  
## 69 86 -24.14  
## 70 87 25.74

aggregate(sales$Margin ~ sales$Customer.Age, data = sales, FUN = mean)

## sales$Customer.Age sales$Margin  
## 1 17 46.33894  
## 2 18 66.13525  
## 3 19 50.26777  
## 4 20 53.99329  
## 5 21 58.49130  
## 6 22 59.37119  
## 7 23 46.93210  
## 8 24 72.03996  
## 9 25 71.61153  
## 10 26 60.55315  
## 11 27 61.91876  
## 12 28 62.07987  
## 13 29 60.64100  
## 14 30 66.28239  
## 15 31 63.17521  
## 16 32 64.32527  
## 17 33 61.36851  
## 18 34 67.76984  
## 19 35 66.51624  
## 20 36 59.50664  
## 21 37 69.22617  
## 22 38 66.32285  
## 23 39 70.37742  
## 24 40 75.32713  
## 25 41 78.93493  
## 26 42 77.82511  
## 27 43 66.81084  
## 28 44 75.81998  
## 29 45 65.87500  
## 30 46 67.92669  
## 31 47 61.94745  
## 32 48 68.78918  
## 33 49 53.28201  
## 34 50 67.11855  
## 35 51 77.85714  
## 36 52 73.27757  
## 37 53 69.60887  
## 38 54 64.91927  
## 39 55 55.34840  
## 40 56 59.27365  
## 41 57 60.15074  
## 42 58 48.15135  
## 43 59 48.01075  
## 44 60 43.96089  
## 45 61 57.63953  
## 46 62 61.30282  
## 47 63 47.50427  
## 48 64 61.51429  
## 49 65 26.24000  
## 50 66 14.52174  
## 51 67 86.81818  
## 52 68 49.48276  
## 53 69 96.47368  
## 54 70 86.83333  
## 55 71 55.71429  
## 56 72 118.40000  
## 57 73 204.00000  
## 58 74 136.00000  
## 59 75 147.50000  
## 60 76 41.50000  
## 61 77 104.12500  
## 62 78 41.81818  
## 63 79 21.20000  
## 64 80 98.66667  
## 65 81 114.16667  
## 66 82 51.50000  
## 67 84 35.33333  
## 68 85 46.75000  
## 69 86 392.75000  
## 70 87 59.00000

aggregate(sales$Margin ~ sales$Customer.Age, data = sales, FUN = median)

## sales$Customer.Age sales$Margin  
## 1 17 21.0  
## 2 18 23.0  
## 3 19 22.0  
## 4 20 21.0  
## 5 21 25.0  
## 6 22 23.0  
## 7 23 20.0  
## 8 24 26.0  
## 9 25 27.0  
## 10 26 26.0  
## 11 27 27.5  
## 12 28 28.0  
## 13 29 28.0  
## 14 30 25.0  
## 15 31 28.0  
## 16 32 31.0  
## 17 33 27.0  
## 18 34 26.0  
## 19 35 30.0  
## 20 36 25.0  
## 21 37 30.0  
## 22 38 30.5  
## 23 39 30.0  
## 24 40 32.0  
## 25 41 30.0  
## 26 42 30.0  
## 27 43 31.0  
## 28 44 31.0  
## 29 45 26.0  
## 30 46 27.0  
## 31 47 23.0  
## 32 48 29.0  
## 33 49 23.0  
## 34 50 28.0  
## 35 51 28.0  
## 36 52 28.0  
## 37 53 27.0  
## 38 54 32.0  
## 39 55 27.5  
## 40 56 28.5  
## 41 57 26.0  
## 42 58 27.0  
## 43 59 23.0  
## 44 60 19.0  
## 45 61 26.5  
## 46 62 20.0  
## 47 63 24.0  
## 48 64 21.0  
## 49 65 20.5  
## 50 66 10.0  
## 51 67 45.5  
## 52 68 21.0  
## 53 69 36.0  
## 54 70 21.5  
## 55 71 20.5  
## 56 72 21.5  
## 57 73 144.0  
## 58 74 136.0  
## 59 75 117.0  
## 60 76 41.5  
## 61 77 33.0  
## 62 78 23.0  
## 63 79 24.0  
## 64 80 33.0  
## 65 81 64.5  
## 66 82 51.5  
## 67 84 14.0  
## 68 85 22.0  
## 69 86 165.5  
## 70 87 26.0

aggregate(sales$Margin ~ sales$Customer.Age, data = sales, FUN = max)

## sales$Customer.Age sales$Margin  
## 1 17 796  
## 2 18 898  
## 3 19 527  
## 4 20 954  
## 5 21 1026  
## 6 22 1017  
## 7 23 1430  
## 8 24 1128  
## 9 25 1058  
## 10 26 1025  
## 11 27 1169  
## 12 28 963  
## 13 29 899  
## 14 30 1006  
## 15 31 1153  
## 16 32 812  
## 17 33 940  
## 18 34 1240  
## 19 35 1016  
## 20 36 930  
## 21 37 864  
## 22 38 1274  
## 23 39 1173  
## 24 40 1459  
## 25 41 976  
## 26 42 1062  
## 27 43 962  
## 28 44 964  
## 29 45 1297  
## 30 46 1323  
## 31 47 960  
## 32 48 863  
## 33 49 961  
## 34 50 1842  
## 35 51 955  
## 36 52 918  
## 37 53 1099  
## 38 54 730  
## 39 55 648  
## 40 56 1368  
## 41 57 670  
## 42 58 533  
## 43 59 470  
## 44 60 517  
## 45 61 521  
## 46 62 787  
## 47 63 678  
## 48 64 734  
## 49 65 209  
## 50 66 77  
## 51 67 420  
## 52 68 317  
## 53 69 396  
## 54 70 648  
## 55 71 262  
## 56 72 538  
## 57 73 461  
## 58 74 252  
## 59 75 336  
## 60 76 47  
## 61 77 520  
## 62 78 159  
## 63 79 43  
## 64 80 266  
## 65 81 278  
## 66 82 113  
## 67 84 234  
## 68 85 182  
## 69 86 1261  
## 70 87 143

aggregate(sales$Margin ~ sales$Customer.Age, data = sales, FUN = min)

## sales$Customer.Age sales$Margin  
## 1 17 -620  
## 2 18 -375  
## 3 19 -441  
## 4 20 -411  
## 5 21 -750  
## 6 22 -574  
## 7 23 -484  
## 8 24 -527  
## 9 25 -668  
## 10 26 -780  
## 11 27 -620  
## 12 28 -829  
## 13 29 -920  
## 14 30 -932  
## 15 31 -937  
## 16 32 -822  
## 17 33 -897  
## 18 34 -627  
## 19 35 -651  
## 20 36 -568  
## 21 37 -741  
## 22 38 -752  
## 23 39 -607  
## 24 40 -785  
## 25 41 -819  
## 26 42 -686  
## 27 43 -807  
## 28 44 -654  
## 29 45 -836  
## 30 46 -700  
## 31 47 -637  
## 32 48 -527  
## 33 49 -719  
## 34 50 -410  
## 35 51 -618  
## 36 52 -828  
## 37 53 -733  
## 38 54 -444  
## 39 55 -742  
## 40 56 -363  
## 41 57 -276  
## 42 58 -327  
## 43 59 -293  
## 44 60 -496  
## 45 61 -665  
## 46 62 -349  
## 47 63 -288  
## 48 64 -395  
## 49 65 -420  
## 50 66 -124  
## 51 67 0  
## 52 68 -39  
## 53 69 1  
## 54 70 -53  
## 55 71 3  
## 56 72 -146  
## 57 73 67  
## 58 74 20  
## 59 75 5  
## 60 76 36  
## 61 77 11  
## 62 78 3  
## 63 79 4  
## 64 80 -3  
## 65 81 -10  
## 66 82 -10  
## 67 84 -32  
## 68 85 1  
## 69 86 -21  
## 70 87 8

aggregate(sales$Margin\_percent ~ sales$Customer.Age, data = sales, FUN = mean)

## sales$Customer.Age sales$Margin\_percent  
## 1 17 14.44406  
## 2 18 14.86576  
## 3 19 13.83058  
## 4 20 15.37809  
## 5 21 14.96025  
## 6 22 14.18147  
## 7 23 13.27925  
## 8 24 13.70090  
## 9 25 14.42983  
## 10 26 13.82973  
## 11 27 13.05992  
## 12 28 12.34667  
## 13 29 11.97816  
## 14 30 12.76309  
## 15 31 11.77813  
## 16 32 13.85034  
## 17 33 12.89446  
## 18 34 12.86279  
## 19 35 12.55664  
## 20 36 12.96859  
## 21 37 13.25301  
## 22 38 13.17848  
## 23 39 12.66982  
## 24 40 14.15368  
## 25 41 13.91740  
## 26 42 13.73363  
## 27 43 13.30788  
## 28 44 14.41791  
## 29 45 12.40999  
## 30 46 14.14546  
## 31 47 13.80379  
## 32 48 14.36569  
## 33 49 12.28473  
## 34 50 14.16273  
## 35 51 14.51295  
## 36 52 13.62487  
## 37 53 13.22101  
## 38 54 13.92281  
## 39 55 14.19059  
## 40 56 13.34341  
## 41 57 11.95007  
## 42 58 11.79168  
## 43 59 12.08704  
## 44 60 14.00291  
## 45 61 11.80262  
## 46 62 13.18176  
## 47 63 15.09889  
## 48 64 15.24962  
## 49 65 12.00740  
## 50 66 10.19391  
## 51 67 17.40773  
## 52 68 17.90793  
## 53 69 19.13632  
## 54 70 19.10333  
## 55 71 19.28786  
## 56 72 15.03800  
## 57 73 23.61750  
## 58 74 13.69500  
## 59 75 25.11000  
## 60 76 26.30500  
## 61 77 21.43250  
## 62 78 17.80000  
## 63 79 12.26600  
## 64 80 20.69667  
## 65 81 22.24500  
## 66 82 19.17000  
## 67 84 18.76333  
## 68 85 19.63500  
## 69 86 14.29250  
## 70 87 33.14667

aggregate(sales$Margin\_percent ~ sales$Customer.Age, data = sales, FUN = median)

## sales$Customer.Age sales$Margin\_percent  
## 1 17 15.540  
## 2 18 15.560  
## 3 19 14.290  
## 4 20 15.660  
## 5 21 14.570  
## 6 22 15.120  
## 7 23 13.790  
## 8 24 14.875  
## 9 25 15.660  
## 10 26 15.260  
## 11 27 15.460  
## 12 28 14.290  
## 13 29 13.660  
## 14 30 14.630  
## 15 31 13.910  
## 16 32 15.090  
## 17 33 14.425  
## 18 34 14.120  
## 19 35 14.480  
## 20 36 14.790  
## 21 37 14.290  
## 22 38 14.290  
## 23 39 13.910  
## 24 40 15.380  
## 25 41 15.440  
## 26 42 15.215  
## 27 43 15.005  
## 28 44 15.620  
## 29 45 14.290  
## 30 46 16.180  
## 31 47 14.290  
## 32 48 15.790  
## 33 49 13.970  
## 34 50 15.790  
## 35 51 16.280  
## 36 52 15.810  
## 37 53 15.045  
## 38 54 15.620  
## 39 55 15.495  
## 40 56 14.640  
## 41 57 12.405  
## 42 58 12.540  
## 43 59 12.895  
## 44 60 15.790  
## 45 61 13.380  
## 46 62 13.040  
## 47 63 16.670  
## 48 64 16.280  
## 49 65 13.365  
## 50 66 6.830  
## 51 67 18.505  
## 52 68 17.150  
## 53 69 18.520  
## 54 70 16.985  
## 55 71 23.115  
## 56 72 11.580  
## 57 73 24.225  
## 58 74 13.695  
## 59 75 23.905  
## 60 76 26.305  
## 61 77 20.460  
## 62 78 15.790  
## 63 79 11.910  
## 64 80 35.480  
## 65 81 28.455  
## 66 82 19.170  
## 67 84 22.740  
## 68 85 20.095  
## 69 86 20.335  
## 70 87 36.360

aggregate(sales$Margin\_percent ~ sales$Customer.Age, data = sales, FUN = max)

## sales$Customer.Age sales$Margin\_percent  
## 1 17 44.00  
## 2 18 41.18  
## 3 19 41.67  
## 4 20 40.85  
## 5 21 43.82  
## 6 22 43.33  
## 7 23 42.86  
## 8 24 42.86  
## 9 25 43.73  
## 10 26 44.77  
## 11 27 42.08  
## 12 28 43.22  
## 13 29 44.79  
## 14 30 45.72  
## 15 31 46.11  
## 16 32 42.31  
## 17 33 43.51  
## 18 34 42.86  
## 19 35 43.26  
## 20 36 44.71  
## 21 37 41.67  
## 22 38 41.24  
## 23 39 42.99  
## 24 40 46.00  
## 25 41 43.18  
## 26 42 43.07  
## 27 43 42.15  
## 28 44 50.00  
## 29 45 43.82  
## 30 46 44.74  
## 31 47 43.26  
## 32 48 41.67  
## 33 49 45.03  
## 34 50 50.00  
## 35 51 42.35  
## 36 52 41.94  
## 37 53 45.10  
## 38 54 41.85  
## 39 55 40.59  
## 40 56 41.02  
## 41 57 38.46  
## 42 58 41.04  
## 43 59 41.67  
## 44 60 42.86  
## 45 61 38.60  
## 46 62 39.42  
## 47 63 40.68  
## 48 64 39.72  
## 49 65 28.57  
## 50 66 27.43  
## 51 67 30.69  
## 52 68 40.00  
## 53 69 29.36  
## 54 70 39.94  
## 55 71 36.36  
## 56 72 37.66  
## 57 73 28.62  
## 58 74 14.06  
## 59 75 41.59  
## 60 76 27.48  
## 61 77 29.05  
## 62 78 26.67  
## 63 79 16.67  
## 64 80 35.99  
## 65 81 34.62  
## 66 82 39.79  
## 67 84 33.33  
## 68 85 23.91  
## 69 86 40.64  
## 70 87 37.34

aggregate(sales$Margin\_percent ~ sales$Customer.Age, data = sales, FUN = min)

## sales$Customer.Age sales$Margin\_percent  
## 1 17 -41.89  
## 2 18 -34.94  
## 3 19 -34.42  
## 4 20 -30.14  
## 5 21 -45.90  
## 6 22 -38.86  
## 7 23 -36.36  
## 8 24 -56.86  
## 9 25 -54.73  
## 10 26 -55.63  
## 11 27 -68.67  
## 12 28 -60.71  
## 13 29 -65.24  
## 14 30 -61.80  
## 15 31 -64.75  
## 16 32 -66.99  
## 17 33 -58.02  
## 18 34 -50.34  
## 19 35 -56.63  
## 20 36 -44.77  
## 21 37 -51.42  
## 22 38 -47.96  
## 23 39 -50.00  
## 24 40 -49.09  
## 25 41 -60.09  
## 26 42 -39.04  
## 27 43 -49.33  
## 28 44 -39.85  
## 29 45 -57.30  
## 30 46 -47.54  
## 31 47 -38.42  
## 32 48 -34.26  
## 33 49 -45.62  
## 34 50 -33.75  
## 35 51 -35.48  
## 36 52 -55.50  
## 37 53 -46.93  
## 38 54 -37.25  
## 39 55 -56.77  
## 40 56 -36.02  
## 41 57 -23.12  
## 42 58 -15.45  
## 43 59 -31.39  
## 44 60 -27.19  
## 45 61 -40.80  
## 46 62 -17.93  
## 47 63 -36.36  
## 48 64 -20.52  
## 49 65 -22.11  
## 50 66 -6.37  
## 51 67 0.00  
## 52 68 -3.38  
## 53 69 8.64  
## 54 70 -6.98  
## 55 71 4.61  
## 56 72 -6.52  
## 57 73 17.40  
## 58 74 13.33  
## 59 75 12.15  
## 60 76 25.13  
## 61 77 12.50  
## 62 78 8.78  
## 63 79 8.82  
## 64 80 -9.38  
## 65 81 -16.95  
## 66 82 -1.45  
## 67 84 -11.94  
## 68 85 12.65  
## 69 86 -24.14  
## 70 87 25.74

aggregate(sales$Quantity ~ sales$Customer.Gender, data = sales, FUN = mean)

## sales$Customer.Gender sales$Quantity  
## 1 F 2.012544  
## 2 M 1.992923

aggregate(sales$Quantity ~ sales$Customer.Gender, data = sales, FUN = median)

## sales$Customer.Gender sales$Quantity  
## 1 F 2  
## 2 M 2

aggregate(sales$Quantity ~ sales$Customer.Gender, data = sales, FUN = max)

## sales$Customer.Gender sales$Quantity  
## 1 F 3  
## 2 M 3

aggregate(sales$Quantity ~ sales$Customer.Gender, data = sales, FUN = min)

## sales$Customer.Gender sales$Quantity  
## 1 F 1  
## 2 M 1

aggregate(sales$Unit.Cost ~ sales$Customer.Gender, data = sales, FUN = mean)

## sales$Customer.Gender sales$Unit.Cost  
## 1 F 349.3329  
## 2 M 350.4244

aggregate(sales$Unit.Cost ~ sales$Customer.Gender, data = sales, FUN = median)

## sales$Customer.Gender sales$Unit.Cost  
## 1 F 145  
## 2 M 155

aggregate(sales$Unit.Cost ~ sales$Customer.Gender, data = sales, FUN = max)

## sales$Customer.Gender sales$Unit.Cost  
## 1 F 3120  
## 2 M 3240

aggregate(sales$Unit.Cost ~ sales$Customer.Gender, data = sales, FUN = min)

## sales$Customer.Gender sales$Unit.Cost  
## 1 F 0.67  
## 2 M 0.67

aggregate(sales$Unit.Price ~ sales$Customer.Gender, data = sales, FUN = mean)

## sales$Customer.Gender sales$Unit.Price  
## 1 F 388.1039  
## 2 M 390.3349

aggregate(sales$Unit.Price ~ sales$Customer.Gender, data = sales, FUN = median)

## sales$Customer.Gender sales$Unit.Price  
## 1 F 174  
## 2 M 184

aggregate(sales$Unit.Price ~ sales$Customer.Gender, data = sales, FUN = max)

## sales$Customer.Gender sales$Unit.Price  
## 1 F 4113  
## 2 M 5082

aggregate(sales$Unit.Price ~ sales$Customer.Gender, data = sales, FUN = min)

## sales$Customer.Gender sales$Unit.Price  
## 1 F 0.666667  
## 2 M 0.666667

aggregate(sales$Cost ~ sales$Customer.Gender, data = sales, FUN = mean)

## sales$Customer.Gender sales$Cost  
## 1 F 576.4516  
## 2 M 575.6074

aggregate(sales$Cost ~ sales$Customer.Gender, data = sales, FUN = median)

## sales$Customer.Gender sales$Cost  
## 1 F 252  
## 2 M 270

aggregate(sales$Cost ~ sales$Customer.Gender, data = sales, FUN = max)

## sales$Customer.Gender sales$Cost  
## 1 F 3600  
## 2 M 3600

aggregate(sales$Cost ~ sales$Customer.Gender, data = sales, FUN = min)

## sales$Customer.Gender sales$Cost  
## 1 F 2  
## 2 M 2

aggregate(sales$Revenue ~ sales$Customer.Gender, data = sales, FUN = mean)

## sales$Customer.Gender sales$Revenue  
## 1 F 640.8327  
## 2 M 640.9403

aggregate(sales$Revenue ~ sales$Customer.Gender, data = sales, FUN = median)

## sales$Customer.Gender sales$Revenue  
## 1 F 304  
## 2 M 329

aggregate(sales$Revenue ~ sales$Customer.Gender, data = sales, FUN = max)

## sales$Customer.Gender sales$Revenue  
## 1 F 4266  
## 2 M 5082

aggregate(sales$Revenue ~ sales$Customer.Gender, data = sales, FUN = min)

## sales$Customer.Gender sales$Revenue  
## 1 F 2  
## 2 M 2

aggregate(sales$Unit\_Margin ~ sales$Customer.Gender, data = sales, FUN = mean)

## sales$Customer.Gender sales$Unit\_Margin  
## 1 F 38.77097  
## 2 M 39.91054

aggregate(sales$Unit\_Margin ~ sales$Customer.Gender, data = sales, FUN = median)

## sales$Customer.Gender sales$Unit\_Margin  
## 1 F 14  
## 2 M 15

aggregate(sales$Unit\_Margin ~ sales$Customer.Gender, data = sales, FUN = max)

## sales$Customer.Gender sales$Unit\_Margin  
## 1 F 1167  
## 2 M 1842

aggregate(sales$Unit\_Margin ~ sales$Customer.Gender, data = sales, FUN = min)

## sales$Customer.Gender sales$Unit\_Margin  
## 1 F -749  
## 2 M -937

aggregate(sales$Unit\_Margin\_percent ~ sales$Customer.Gender, data = sales, FUN = mean)

## sales$Customer.Gender sales$Unit\_Margin\_percent  
## 1 F 13.38132  
## 2 M 13.43094

aggregate(sales$Unit\_Margin\_percent ~ sales$Customer.Gender, data = sales, FUN = median)

## sales$Customer.Gender sales$Unit\_Margin\_percent  
## 1 F 14.63  
## 2 M 14.95

aggregate(sales$Unit\_Margin\_percent ~ sales$Customer.Gender, data = sales, FUN = max)

## sales$Customer.Gender sales$Unit\_Margin\_percent  
## 1 F 49.75  
## 2 M 49.75

aggregate(sales$Unit\_Margin\_percent ~ sales$Customer.Gender, data = sales, FUN = min)

## sales$Customer.Gender sales$Unit\_Margin\_percent  
## 1 F -68.67  
## 2 M -67.49

aggregate(sales$Margin ~ sales$Customer.Gender, data = sales, FUN = mean)

## sales$Customer.Gender sales$Margin  
## 1 F 64.38113  
## 2 M 65.33288

aggregate(sales$Margin ~ sales$Customer.Gender, data = sales, FUN = median)

## sales$Customer.Gender sales$Margin  
## 1 F 26  
## 2 M 28

aggregate(sales$Margin ~ sales$Customer.Gender, data = sales, FUN = max)

## sales$Customer.Gender sales$Margin  
## 1 F 1459  
## 2 M 1842

aggregate(sales$Margin ~ sales$Customer.Gender, data = sales, FUN = min)

## sales$Customer.Gender sales$Margin  
## 1 F -829  
## 2 M -937

aggregate(sales$Margin\_percent ~ sales$Customer.Gender, data = sales, FUN = mean)

## sales$Customer.Gender sales$Margin\_percent  
## 1 F 13.38174  
## 2 M 13.43144

aggregate(sales$Margin\_percent ~ sales$Customer.Gender, data = sales, FUN = median)

## sales$Customer.Gender sales$Margin\_percent  
## 1 F 14.63  
## 2 M 14.95

aggregate(sales$Margin\_percent ~ sales$Customer.Gender, data = sales, FUN = max)

## sales$Customer.Gender sales$Margin\_percent  
## 1 F 50  
## 2 M 50

aggregate(sales$Margin\_percent ~ sales$Customer.Gender, data = sales, FUN = min)

## sales$Customer.Gender sales$Margin\_percent  
## 1 F -68.67  
## 2 M -67.49

aggregate(sales$Customer.Age ~ sales$Country, data = sales, FUN = mean)

## sales$Country sales$Customer.Age  
## 1 France 35.22426  
## 2 Germany 34.86288  
## 3 United Kingdom 35.58200  
## 4 United States 37.43555

aggregate(sales$Customer.Age ~ sales$Country, data = sales, FUN = median)

## sales$Country sales$Customer.Age  
## 1 France 33  
## 2 Germany 33  
## 3 United Kingdom 34  
## 4 United States 37

aggregate(sales$Customer.Age ~ sales$Country, data = sales, FUN = max)

## sales$Country sales$Customer.Age  
## 1 France 86  
## 2 Germany 87  
## 3 United Kingdom 85  
## 4 United States 79

aggregate(sales$Customer.Age ~ sales$Country, data = sales, FUN = min)

## sales$Country sales$Customer.Age  
## 1 France 17  
## 2 Germany 17  
## 3 United Kingdom 17  
## 4 United States 17

aggregate(sales$Quantity ~ sales$Country, data = sales, FUN = mean)

## sales$Country sales$Quantity  
## 1 France 2.007159  
## 2 Germany 1.996923  
## 3 United Kingdom 1.996885  
## 4 United States 2.004813

aggregate(sales$Quantity ~ sales$Country, data = sales, FUN = median)

## sales$Country sales$Quantity  
## 1 France 2  
## 2 Germany 2  
## 3 United Kingdom 2  
## 4 United States 2

aggregate(sales$Quantity ~ sales$Country, data = sales, FUN = max)

## sales$Country sales$Quantity  
## 1 France 3  
## 2 Germany 3  
## 3 United Kingdom 3  
## 4 United States 3

aggregate(sales$Quantity ~ sales$Country, data = sales, FUN = min)

## sales$Country sales$Quantity  
## 1 France 1  
## 2 Germany 1  
## 3 United Kingdom 1  
## 4 United States 1

aggregate(sales$Unit.Cost ~ sales$Country, data = sales, FUN = mean)

## sales$Country sales$Unit.Cost  
## 1 France 374.4496  
## 2 Germany 386.9667  
## 3 United Kingdom 371.7177  
## 4 United States 324.4492

aggregate(sales$Unit.Cost ~ sales$Country, data = sales, FUN = median)

## sales$Country sales$Unit.Cost  
## 1 France 171.0  
## 2 Germany 180.0  
## 3 United Kingdom 170.0  
## 4 United States 137.5

aggregate(sales$Unit.Cost ~ sales$Country, data = sales, FUN = max)

## sales$Country sales$Unit.Cost  
## 1 France 2443  
## 2 Germany 3240  
## 3 United Kingdom 3120  
## 4 United States 3120

aggregate(sales$Unit.Cost ~ sales$Country, data = sales, FUN = min)

## sales$Country sales$Unit.Cost  
## 1 France 0.67  
## 2 Germany 0.67  
## 3 United Kingdom 0.67  
## 4 United States 0.67

aggregate(sales$Unit.Price ~ sales$Country, data = sales, FUN = mean)

## sales$Country sales$Unit.Price  
## 1 France 404.8099  
## 2 Germany 500.1085  
## 3 United Kingdom 402.6917  
## 4 United States 348.1224

aggregate(sales$Unit.Price ~ sales$Country, data = sales, FUN = median)

## sales$Country sales$Unit.Price  
## 1 France 192.3333  
## 2 Germany 245.0000  
## 3 United Kingdom 190.3333  
## 4 United States 158.4167

aggregate(sales$Unit.Price ~ sales$Country, data = sales, FUN = max)

## sales$Country sales$Unit.Price  
## 1 France 3495  
## 2 Germany 5082  
## 3 United Kingdom 4113  
## 4 United States 3887

aggregate(sales$Unit.Price ~ sales$Country, data = sales, FUN = min)

## sales$Country sales$Unit.Price  
## 1 France 0.666667  
## 2 Germany 1.000000  
## 3 United Kingdom 0.666667  
## 4 United States 0.666667

aggregate(sales$Cost ~ sales$Country, data = sales, FUN = mean)

## sales$Country sales$Cost  
## 1 France 614.0571  
## 2 Germany 631.8587  
## 3 United Kingdom 614.7430  
## 4 United States 535.3273

aggregate(sales$Cost ~ sales$Country, data = sales, FUN = median)

## sales$Country sales$Cost  
## 1 France 301  
## 2 Germany 318  
## 3 United Kingdom 293  
## 4 United States 234

aggregate(sales$Cost ~ sales$Country, data = sales, FUN = max)

## sales$Country sales$Cost  
## 1 France 2443  
## 2 Germany 3240  
## 3 United Kingdom 3120  
## 4 United States 3600

aggregate(sales$Cost ~ sales$Country, data = sales, FUN = min)

## sales$Country sales$Cost  
## 1 France 2  
## 2 Germany 2  
## 3 United Kingdom 2  
## 4 United States 2

aggregate(sales$Revenue ~ sales$Country, data = sales, FUN = mean)

## sales$Country sales$Revenue  
## 1 France 666.8158  
## 2 Germany 816.2465  
## 3 United Kingdom 665.9741  
## 4 United States 574.1172

aggregate(sales$Revenue ~ sales$Country, data = sales, FUN = median)

## sales$Country sales$Revenue  
## 1 France 361.0  
## 2 Germany 462.0  
## 3 United Kingdom 355.0  
## 4 United States 271.5

aggregate(sales$Revenue ~ sales$Country, data = sales, FUN = max)

## sales$Country sales$Revenue  
## 1 France 3681  
## 2 Germany 5082  
## 3 United Kingdom 4113  
## 4 United States 4923

aggregate(sales$Revenue ~ sales$Country, data = sales, FUN = min)

## sales$Country sales$Revenue  
## 1 France 2  
## 2 Germany 3  
## 3 United Kingdom 2  
## 4 United States 2

aggregate(sales$Unit\_Margin ~ sales$Country, data = sales, FUN = mean)

## sales$Country sales$Unit\_Margin  
## 1 France 30.36028  
## 2 Germany 113.14183  
## 3 United Kingdom 30.97398  
## 4 United States 23.67319

aggregate(sales$Unit\_Margin ~ sales$Country, data = sales, FUN = median)

## sales$Country sales$Unit\_Margin  
## 1 France 9.50000  
## 2 Germany 58.00333  
## 3 United Kingdom 14.00000  
## 4 United States 10.50000

aggregate(sales$Unit\_Margin ~ sales$Country, data = sales, FUN = max)

## sales$Country sales$Unit\_Margin  
## 1 France 1167  
## 2 Germany 1842  
## 3 United Kingdom 993  
## 4 United States 845

aggregate(sales$Unit\_Margin ~ sales$Country, data = sales, FUN = min)

## sales$Country sales$Unit\_Margin  
## 1 France -937  
## 2 Germany -31  
## 3 United Kingdom -376  
## 4 United States -472

aggregate(sales$Unit\_Margin\_percent ~ sales$Country, data = sales, FUN = mean)

## sales$Country sales$Unit\_Margin\_percent  
## 1 France 9.288651  
## 2 Germany 26.618885  
## 3 United Kingdom 12.415432  
## 4 United States 11.135300

aggregate(sales$Unit\_Margin\_percent ~ sales$Country, data = sales, FUN = median)

## sales$Country sales$Unit\_Margin\_percent  
## 1 France 13.375  
## 2 Germany 27.840  
## 3 United Kingdom 13.750  
## 4 United States 12.500

aggregate(sales$Unit\_Margin\_percent ~ sales$Country, data = sales, FUN = max)

## sales$Country sales$Unit\_Margin\_percent  
## 1 France 49.75  
## 2 Germany 49.75  
## 3 United Kingdom 34.15  
## 4 United States 33.52

aggregate(sales$Unit\_Margin\_percent ~ sales$Country, data = sales, FUN = min)

## sales$Country sales$Unit\_Margin\_percent  
## 1 France -68.67  
## 2 Germany -2.85  
## 3 United Kingdom -23.45  
## 4 United States -29.93

aggregate(sales$Margin ~ sales$Country, data = sales, FUN = mean)

## sales$Country sales$Margin  
## 1 France 52.75871  
## 2 Germany 184.38788  
## 3 United Kingdom 51.23112  
## 4 United States 38.78983

aggregate(sales$Margin ~ sales$Country, data = sales, FUN = median)

## sales$Country sales$Margin  
## 1 France 18  
## 2 Germany 107  
## 3 United Kingdom 26  
## 4 United States 20

aggregate(sales$Margin ~ sales$Country, data = sales, FUN = max)

## sales$Country sales$Margin  
## 1 France 1297  
## 2 Germany 1842  
## 3 United Kingdom 993  
## 4 United States 1323

aggregate(sales$Margin ~ sales$Country, data = sales, FUN = min)

## sales$Country sales$Margin  
## 1 France -937  
## 2 Germany -31  
## 3 United Kingdom -464  
## 4 United States -496

aggregate(sales$Margin\_percent ~ sales$Country, data = sales, FUN = mean)

## sales$Country sales$Margin\_percent  
## 1 France 9.289137  
## 2 Germany 26.619106  
## 3 United Kingdom 12.416180  
## 4 United States 11.135726

aggregate(sales$Margin\_percent ~ sales$Country, data = sales, FUN = median)

## sales$Country sales$Margin\_percent  
## 1 France 13.375  
## 2 Germany 27.840  
## 3 United Kingdom 13.750  
## 4 United States 12.500

aggregate(sales$Margin\_percent ~ sales$Country, data = sales, FUN = max)

## sales$Country sales$Margin\_percent  
## 1 France 50.00  
## 2 Germany 50.00  
## 3 United Kingdom 34.15  
## 4 United States 33.52

aggregate(sales$Margin\_percent ~ sales$Country, data = sales, FUN = min)

## sales$Country sales$Margin\_percent  
## 1 France -68.67  
## 2 Germany -2.85  
## 3 United Kingdom -23.45  
## 4 United States -29.93

aggregate(sales$Customer.Age ~ sales$State, data = sales, FUN = mean)

## sales$State sales$Customer.Age  
## 1 Alabama 40.00000  
## 2 Arizona 28.00000  
## 3 Bayern 33.84258  
## 4 Brandenburg 36.07447  
## 5 California 37.45828  
## 6 Charente-Maritime 31.78571  
## 7 England 35.58200  
## 8 Essonne 35.64086  
## 9 Florida 34.71429  
## 10 Garonne (Haute) 34.08000  
## 11 Georgia 30.00000  
## 12 Hamburg 34.50403  
## 13 Hauts de Seine 37.29862  
## 14 Hessen 34.88849  
## 15 Illinois 38.50000  
## 16 Kentucky 40.00000  
## 17 Loir et Cher 32.01754  
## 18 Loiret 30.98876  
## 19 Massachusetts 48.00000  
## 20 Minnesota 31.00000  
## 21 Mississippi 46.00000  
## 22 Missouri 35.00000  
## 23 Montana 35.00000  
## 24 Moselle 37.29282  
## 25 New York 37.00000  
## 26 Nord 35.95044  
## 27 Nordrhein-Westfalen 35.18019  
## 28 North Carolina 37.00000  
## 29 Ohio 34.61538  
## 30 Oregon 37.58095  
## 31 Pas de Calais 27.21429  
## 32 Saarland 35.23465  
## 33 Seine (Paris) 35.04487  
## 34 Seine et Marne 35.51913  
## 35 Seine Saint Denis 35.60553  
## 36 Somme 35.59016  
## 37 South Carolina 29.00000  
## 38 Texas 41.92857  
## 39 Utah 26.80000  
## 40 Val d'Oise 33.08462  
## 41 Val de Marne 33.53333  
## 42 Virginia 36.00000  
## 43 Washington 37.34704  
## 44 Wyoming 27.00000  
## 45 Yveline 34.44118

aggregate(sales$Customer.Age ~ sales$State, data = sales, FUN = median)

## sales$State sales$Customer.Age  
## 1 Alabama 40.0  
## 2 Arizona 28.0  
## 3 Bayern 32.0  
## 4 Brandenburg 35.5  
## 5 California 36.0  
## 6 Charente-Maritime 28.0  
## 7 England 34.0  
## 8 Essonne 35.0  
## 9 Florida 29.0  
## 10 Garonne (Haute) 31.0  
## 11 Georgia 30.0  
## 12 Hamburg 33.0  
## 13 Hauts de Seine 35.0  
## 14 Hessen 33.0  
## 15 Illinois 25.0  
## 16 Kentucky 40.0  
## 17 Loir et Cher 28.0  
## 18 Loiret 28.0  
## 19 Massachusetts 48.0  
## 20 Minnesota 31.0  
## 21 Mississippi 46.0  
## 22 Missouri 35.0  
## 23 Montana 35.0  
## 24 Moselle 38.0  
## 25 New York 40.0  
## 26 Nord 35.0  
## 27 Nordrhein-Westfalen 32.0  
## 28 North Carolina 37.0  
## 29 Ohio 38.0  
## 30 Oregon 37.0  
## 31 Pas de Calais 24.0  
## 32 Saarland 34.0  
## 33 Seine (Paris) 32.0  
## 34 Seine et Marne 32.0  
## 35 Seine Saint Denis 34.0  
## 36 Somme 31.0  
## 37 South Carolina 29.0  
## 38 Texas 47.0  
## 39 Utah 24.0  
## 40 Val d'Oise 30.0  
## 41 Val de Marne 32.0  
## 42 Virginia 36.0  
## 43 Washington 37.0  
## 44 Wyoming 28.0  
## 45 Yveline 32.0

aggregate(sales$Customer.Age ~ sales$State, data = sales, FUN = max)

## sales$State sales$Customer.Age  
## 1 Alabama 40  
## 2 Arizona 32  
## 3 Bayern 64  
## 4 Brandenburg 56  
## 5 California 79  
## 6 Charente-Maritime 54  
## 7 England 85  
## 8 Essonne 72  
## 9 Florida 49  
## 10 Garonne (Haute) 60  
## 11 Georgia 35  
## 12 Hamburg 67  
## 13 Hauts de Seine 64  
## 14 Hessen 81  
## 15 Illinois 61  
## 16 Kentucky 40  
## 17 Loir et Cher 59  
## 18 Loiret 62  
## 19 Massachusetts 48  
## 20 Minnesota 31  
## 21 Mississippi 46  
## 22 Missouri 35  
## 23 Montana 35  
## 24 Moselle 68  
## 25 New York 49  
## 26 Nord 80  
## 27 Nordrhein-Westfalen 85  
## 28 North Carolina 37  
## 29 Ohio 40  
## 30 Oregon 67  
## 31 Pas de Calais 48  
## 32 Saarland 87  
## 33 Seine (Paris) 82  
## 34 Seine et Marne 63  
## 35 Seine Saint Denis 84  
## 36 Somme 61  
## 37 South Carolina 29  
## 38 Texas 51  
## 39 Utah 31  
## 40 Val d'Oise 54  
## 41 Val de Marne 62  
## 42 Virginia 36  
## 43 Washington 70  
## 44 Wyoming 28  
## 45 Yveline 86

aggregate(sales$Customer.Age ~ sales$State, data = sales, FUN = min)

## sales$State sales$Customer.Age  
## 1 Alabama 40  
## 2 Arizona 24  
## 3 Bayern 17  
## 4 Brandenburg 17  
## 5 California 17  
## 6 Charente-Maritime 23  
## 7 England 17  
## 8 Essonne 17  
## 9 Florida 29  
## 10 Garonne (Haute) 17  
## 11 Georgia 25  
## 12 Hamburg 17  
## 13 Hauts de Seine 17  
## 14 Hessen 17  
## 15 Illinois 17  
## 16 Kentucky 40  
## 17 Loir et Cher 20  
## 18 Loiret 17  
## 19 Massachusetts 48  
## 20 Minnesota 31  
## 21 Mississippi 46  
## 22 Missouri 35  
## 23 Montana 35  
## 24 Moselle 21  
## 25 New York 21  
## 26 Nord 17  
## 27 Nordrhein-Westfalen 17  
## 28 North Carolina 37  
## 29 Ohio 20  
## 30 Oregon 17  
## 31 Pas de Calais 18  
## 32 Saarland 17  
## 33 Seine (Paris) 17  
## 34 Seine et Marne 19  
## 35 Seine Saint Denis 17  
## 36 Somme 21  
## 37 South Carolina 29  
## 38 Texas 20  
## 39 Utah 24  
## 40 Val d'Oise 17  
## 41 Val de Marne 19  
## 42 Virginia 36  
## 43 Washington 17  
## 44 Wyoming 25  
## 45 Yveline 17

aggregate(sales$Quantity ~ sales$State, data = sales, FUN = mean)

## sales$State sales$Quantity  
## 1 Alabama 3.000000  
## 2 Arizona 2.500000  
## 3 Bayern 2.007496  
## 4 Brandenburg 2.063830  
## 5 California 2.005614  
## 6 Charente-Maritime 2.128571  
## 7 England 1.996885  
## 8 Essonne 2.006452  
## 9 Florida 2.000000  
## 10 Garonne (Haute) 1.980000  
## 11 Georgia 2.000000  
## 12 Hamburg 1.996548  
## 13 Hauts de Seine 1.990177  
## 14 Hessen 1.987410  
## 15 Illinois 2.214286  
## 16 Kentucky 1.500000  
## 17 Loir et Cher 2.052632  
## 18 Loiret 1.966292  
## 19 Massachusetts 2.000000  
## 20 Minnesota 2.000000  
## 21 Mississippi 1.500000  
## 22 Missouri 2.000000  
## 23 Montana 1.500000  
## 24 Moselle 1.878453  
## 25 New York 1.700000  
## 26 Nord 2.012706  
## 27 Nordrhein-Westfalen 1.973527  
## 28 North Carolina 2.000000  
## 29 Ohio 1.692308  
## 30 Oregon 2.006541  
## 31 Pas de Calais 1.904762  
## 32 Saarland 2.016317  
## 33 Seine (Paris) 2.004579  
## 34 Seine et Marne 1.907104  
## 35 Seine Saint Denis 2.066583  
## 36 Somme 1.836066  
## 37 South Carolina 2.200000  
## 38 Texas 1.785714  
## 39 Utah 1.600000  
## 40 Val d'Oise 2.084615  
## 41 Val de Marne 1.973333  
## 42 Virginia 2.000000  
## 43 Washington 2.004420  
## 44 Wyoming 1.666667  
## 45 Yveline 2.024887

aggregate(sales$Quantity ~ sales$State, data = sales, FUN = median)

## sales$State sales$Quantity  
## 1 Alabama 3.0  
## 2 Arizona 2.5  
## 3 Bayern 2.0  
## 4 Brandenburg 2.0  
## 5 California 2.0  
## 6 Charente-Maritime 2.0  
## 7 England 2.0  
## 8 Essonne 2.0  
## 9 Florida 2.0  
## 10 Garonne (Haute) 2.0  
## 11 Georgia 2.0  
## 12 Hamburg 2.0  
## 13 Hauts de Seine 2.0  
## 14 Hessen 2.0  
## 15 Illinois 2.5  
## 16 Kentucky 1.0  
## 17 Loir et Cher 2.0  
## 18 Loiret 2.0  
## 19 Massachusetts 2.0  
## 20 Minnesota 2.0  
## 21 Mississippi 1.5  
## 22 Missouri 2.0  
## 23 Montana 1.5  
## 24 Moselle 2.0  
## 25 New York 1.5  
## 26 Nord 2.0  
## 27 Nordrhein-Westfalen 2.0  
## 28 North Carolina 2.0  
## 29 Ohio 1.0  
## 30 Oregon 2.0  
## 31 Pas de Calais 2.0  
## 32 Saarland 2.0  
## 33 Seine (Paris) 2.0  
## 34 Seine et Marne 2.0  
## 35 Seine Saint Denis 2.0  
## 36 Somme 2.0  
## 37 South Carolina 2.0  
## 38 Texas 2.0  
## 39 Utah 2.0  
## 40 Val d'Oise 2.0  
## 41 Val de Marne 2.0  
## 42 Virginia 2.0  
## 43 Washington 2.0  
## 44 Wyoming 1.0  
## 45 Yveline 2.0

aggregate(sales$Quantity ~ sales$State, data = sales, FUN = max)

## sales$State sales$Quantity  
## 1 Alabama 3  
## 2 Arizona 3  
## 3 Bayern 3  
## 4 Brandenburg 3  
## 5 California 3  
## 6 Charente-Maritime 3  
## 7 England 3  
## 8 Essonne 3  
## 9 Florida 3  
## 10 Garonne (Haute) 3  
## 11 Georgia 2  
## 12 Hamburg 3  
## 13 Hauts de Seine 3  
## 14 Hessen 3  
## 15 Illinois 3  
## 16 Kentucky 3  
## 17 Loir et Cher 3  
## 18 Loiret 3  
## 19 Massachusetts 2  
## 20 Minnesota 3  
## 21 Mississippi 2  
## 22 Missouri 2  
## 23 Montana 2  
## 24 Moselle 3  
## 25 New York 3  
## 26 Nord 3  
## 27 Nordrhein-Westfalen 3  
## 28 North Carolina 3  
## 29 Ohio 3  
## 30 Oregon 3  
## 31 Pas de Calais 3  
## 32 Saarland 3  
## 33 Seine (Paris) 3  
## 34 Seine et Marne 3  
## 35 Seine Saint Denis 3  
## 36 Somme 3  
## 37 South Carolina 3  
## 38 Texas 3  
## 39 Utah 2  
## 40 Val d'Oise 3  
## 41 Val de Marne 3  
## 42 Virginia 3  
## 43 Washington 3  
## 44 Wyoming 3  
## 45 Yveline 3

aggregate(sales$Quantity ~ sales$State, data = sales, FUN = min)

## sales$State sales$Quantity  
## 1 Alabama 3  
## 2 Arizona 2  
## 3 Bayern 1  
## 4 Brandenburg 1  
## 5 California 1  
## 6 Charente-Maritime 1  
## 7 England 1  
## 8 Essonne 1  
## 9 Florida 1  
## 10 Garonne (Haute) 1  
## 11 Georgia 2  
## 12 Hamburg 1  
## 13 Hauts de Seine 1  
## 14 Hessen 1  
## 15 Illinois 1  
## 16 Kentucky 1  
## 17 Loir et Cher 1  
## 18 Loiret 1  
## 19 Massachusetts 2  
## 20 Minnesota 1  
## 21 Mississippi 1  
## 22 Missouri 2  
## 23 Montana 1  
## 24 Moselle 1  
## 25 New York 1  
## 26 Nord 1  
## 27 Nordrhein-Westfalen 1  
## 28 North Carolina 1  
## 29 Ohio 1  
## 30 Oregon 1  
## 31 Pas de Calais 1  
## 32 Saarland 1  
## 33 Seine (Paris) 1  
## 34 Seine et Marne 1  
## 35 Seine Saint Denis 1  
## 36 Somme 1  
## 37 South Carolina 1  
## 38 Texas 1  
## 39 Utah 1  
## 40 Val d'Oise 1  
## 41 Val de Marne 1  
## 42 Virginia 1  
## 43 Washington 1  
## 44 Wyoming 1  
## 45 Yveline 1

aggregate(sales$Unit.Cost ~ sales$State, data = sales, FUN = mean)

## sales$State sales$Unit.Cost  
## 1 Alabama 167.0000  
## 2 Arizona 361.4150  
## 3 Bayern 373.5070  
## 4 Brandenburg 418.3190  
## 5 California 333.6559  
## 6 Charente-Maritime 345.5521  
## 7 England 371.7177  
## 8 Essonne 412.1802  
## 9 Florida 218.7614  
## 10 Garonne (Haute) 417.6051  
## 11 Georgia 43.7500  
## 12 Hamburg 392.3321  
## 13 Hauts de Seine 391.3077  
## 14 Hessen 385.0130  
## 15 Illinois 161.0000  
## 16 Kentucky 319.5000  
## 17 Loir et Cher 284.0381  
## 18 Loiret 369.7330  
## 19 Massachusetts 1024.5000  
## 20 Minnesota 481.8900  
## 21 Mississippi 661.2500  
## 22 Missouri 332.5000  
## 23 Montana 538.7500  
## 24 Moselle 394.9096  
## 25 New York 172.8330  
## 26 Nord 363.6804  
## 27 Nordrhein-Westfalen 384.7832  
## 28 North Carolina 46.5000  
## 29 Ohio 437.0262  
## 30 Oregon 322.2335  
## 31 Pas de Calais 272.0952  
## 32 Saarland 391.7042  
## 33 Seine (Paris) 371.0289  
## 34 Seine et Marne 421.7030  
## 35 Seine Saint Denis 357.4438  
## 36 Somme 366.5902  
## 37 South Carolina 59.2340  
## 38 Texas 157.8686  
## 39 Utah 342.5000  
## 40 Val d'Oise 293.2845  
## 41 Val de Marne 334.5247  
## 42 Virginia 149.1650  
## 43 Washington 308.3866  
## 44 Wyoming 318.1100  
## 45 Yveline 395.4806

aggregate(sales$Unit.Cost ~ sales$State, data = sales, FUN = median)

## sales$State sales$Unit.Cost  
## 1 Alabama 167.000  
## 2 Arizona 361.415  
## 3 Bayern 180.000  
## 4 Brandenburg 198.165  
## 5 California 140.665  
## 6 Charente-Maritime 157.500  
## 7 England 170.000  
## 8 Essonne 191.670  
## 9 Florida 140.000  
## 10 Garonne (Haute) 167.000  
## 11 Georgia 43.750  
## 12 Hamburg 180.000  
## 13 Hauts de Seine 176.000  
## 14 Hessen 178.000  
## 15 Illinois 91.250  
## 16 Kentucky 96.500  
## 17 Loir et Cher 126.000  
## 18 Loiret 182.335  
## 19 Massachusetts 1024.500  
## 20 Minnesota 378.000  
## 21 Mississippi 661.250  
## 22 Missouri 332.500  
## 23 Montana 538.750  
## 24 Moselle 175.000  
## 25 New York 90.000  
## 26 Nord 175.000  
## 27 Nordrhein-Westfalen 175.000  
## 28 North Carolina 46.500  
## 29 Ohio 95.000  
## 30 Oregon 125.500  
## 31 Pas de Calais 183.750  
## 32 Saarland 175.000  
## 33 Seine (Paris) 163.165  
## 34 Seine et Marne 220.000  
## 35 Seine Saint Denis 145.000  
## 36 Somme 140.000  
## 37 South Carolina 44.000  
## 38 Texas 47.500  
## 39 Utah 262.500  
## 40 Val d'Oise 116.500  
## 41 Val de Marne 162.000  
## 42 Virginia 149.165  
## 43 Washington 127.000  
## 44 Wyoming 117.000  
## 45 Yveline 188.750

aggregate(sales$Unit.Cost ~ sales$State, data = sales, FUN = max)

## sales$State sales$Unit.Cost  
## 1 Alabama 315.00  
## 2 Arizona 690.33  
## 3 Bayern 2640.00  
## 4 Brandenburg 2443.00  
## 5 California 3120.00  
## 6 Charente-Maritime 2384.00  
## 7 England 3120.00  
## 8 Essonne 2443.00  
## 9 Florida 690.33  
## 10 Garonne (Haute) 2443.00  
## 11 Georgia 45.00  
## 12 Hamburg 2443.00  
## 13 Hauts de Seine 2443.00  
## 14 Hessen 3240.00  
## 15 Illinois 690.33  
## 16 Kentucky 1015.00  
## 17 Loir et Cher 2320.00  
## 18 Loiret 2384.00  
## 19 Massachusetts 1024.50  
## 20 Minnesota 1050.00  
## 21 Mississippi 850.00  
## 22 Missouri 332.50  
## 23 Montana 815.00  
## 24 Moselle 2443.00  
## 25 New York 567.00  
## 26 Nord 2443.00  
## 27 Nordrhein-Westfalen 2443.00  
## 28 North Carolina 80.00  
## 29 Ohio 1842.00  
## 30 Oregon 2443.00  
## 31 Pas de Calais 875.00  
## 32 Saarland 2443.00  
## 33 Seine (Paris) 2443.00  
## 34 Seine et Marne 2443.00  
## 35 Seine Saint Denis 2443.00  
## 36 Somme 2295.00  
## 37 South Carolina 145.00  
## 38 Texas 769.00  
## 39 Utah 742.00  
## 40 Val d'Oise 2443.00  
## 41 Val de Marne 2384.00  
## 42 Virginia 233.33  
## 43 Washington 2443.00  
## 44 Wyoming 773.33  
## 45 Yveline 2443.00

aggregate(sales$Unit.Cost ~ sales$State, data = sales, FUN = min)

## sales$State sales$Unit.Cost  
## 1 Alabama 19.00  
## 2 Arizona 32.50  
## 3 Bayern 1.67  
## 4 Brandenburg 5.00  
## 5 California 0.67  
## 6 Charente-Maritime 1.67  
## 7 England 0.67  
## 8 Essonne 4.00  
## 9 Florida 30.00  
## 10 Garonne (Haute) 3.33  
## 11 Georgia 42.50  
## 12 Hamburg 0.67  
## 13 Hauts de Seine 0.67  
## 14 Hessen 1.00  
## 15 Illinois 7.00  
## 16 Kentucky 70.00  
## 17 Loir et Cher 6.67  
## 18 Loiret 1.00  
## 19 Massachusetts 1024.50  
## 20 Minnesota 17.67  
## 21 Mississippi 472.50  
## 22 Missouri 332.50  
## 23 Montana 262.50  
## 24 Moselle 1.33  
## 25 New York 18.33  
## 26 Nord 0.67  
## 27 Nordrhein-Westfalen 1.33  
## 28 North Carolina 13.00  
## 29 Ohio 16.00  
## 30 Oregon 0.67  
## 31 Pas de Calais 2.50  
## 32 Saarland 1.00  
## 33 Seine (Paris) 1.33  
## 34 Seine et Marne 0.67  
## 35 Seine Saint Denis 0.67  
## 36 Somme 3.00  
## 37 South Carolina 10.00  
## 38 Texas 5.00  
## 39 Utah 28.00  
## 40 Val d'Oise 1.33  
## 41 Val de Marne 1.67  
## 42 Virginia 65.00  
## 43 Washington 0.67  
## 44 Wyoming 64.00  
## 45 Yveline 1.67

aggregate(sales$Unit.Price ~ sales$State, data = sales, FUN = mean)

## sales$State sales$Unit.Price  
## 1 Alabama 202.33333  
## 2 Arizona 342.58333  
## 3 Bayern 487.63768  
## 4 Brandenburg 541.75709  
## 5 California 357.18217  
## 6 Charente-Maritime 322.10476  
## 7 England 402.69166  
## 8 Essonne 450.15986  
## 9 Florida 224.30952  
## 10 Garonne (Haute) 459.17167  
## 11 Georgia 51.00000  
## 12 Hamburg 499.57192  
## 13 Hauts de Seine 424.36509  
## 14 Hessen 498.44619  
## 15 Illinois 167.52381  
## 16 Kentucky 382.25000  
## 17 Loir et Cher 315.71053  
## 18 Loiret 402.20037  
## 19 Massachusetts 987.50000  
## 20 Minnesota 561.88889  
## 21 Mississippi 739.25000  
## 22 Missouri 383.50000  
## 23 Montana 589.50000  
## 24 Moselle 434.17772  
## 25 New York 196.53333  
## 26 Nord 386.72067  
## 27 Nordrhein-Westfalen 499.60803  
## 28 North Carolina 51.66667  
## 29 Ohio 485.88462  
## 30 Oregon 345.98760  
## 31 Pas de Calais 315.61111  
## 32 Saarland 505.78361  
## 33 Seine (Paris) 400.45910  
## 34 Seine et Marne 456.62842  
## 35 Seine Saint Denis 386.08501  
## 36 Somme 387.89071  
## 37 South Carolina 75.13333  
## 38 Texas 178.77381  
## 39 Utah 358.00000  
## 40 Val d'Oise 324.02949  
## 41 Val de Marne 364.98000  
## 42 Virginia 180.66667  
## 43 Washington 332.29932  
## 44 Wyoming 306.00000  
## 45 Yveline 432.44796

aggregate(sales$Unit.Price ~ sales$State, data = sales, FUN = median)

## sales$State sales$Unit.Price  
## 1 Alabama 202.33333  
## 2 Arizona 342.58333  
## 3 Bayern 259.00000  
## 4 Brandenburg 283.58333  
## 5 California 166.00000  
## 6 Charente-Maritime 150.83333  
## 7 England 190.33333  
## 8 Essonne 217.00000  
## 9 Florida 185.50000  
## 10 Garonne (Haute) 181.33333  
## 11 Georgia 51.00000  
## 12 Hamburg 242.50000  
## 13 Hauts de Seine 199.33333  
## 14 Hessen 242.25000  
## 15 Illinois 99.75000  
## 16 Kentucky 116.50000  
## 17 Loir et Cher 135.00000  
## 18 Loiret 198.66667  
## 19 Massachusetts 987.50000  
## 20 Minnesota 428.00000  
## 21 Mississippi 739.25000  
## 22 Missouri 383.50000  
## 23 Montana 589.50000  
## 24 Moselle 192.66667  
## 25 New York 102.75000  
## 26 Nord 200.00000  
## 27 Nordrhein-Westfalen 245.00000  
## 28 North Carolina 51.66667  
## 29 Ohio 112.00000  
## 30 Oregon 148.00000  
## 31 Pas de Calais 213.16667  
## 32 Saarland 234.00000  
## 33 Seine (Paris) 188.75000  
## 34 Seine et Marne 253.00000  
## 35 Seine Saint Denis 179.00000  
## 36 Somme 219.66667  
## 37 South Carolina 56.66667  
## 38 Texas 55.33333  
## 39 Utah 287.50000  
## 40 Val d'Oise 138.50000  
## 41 Val de Marne 195.66667  
## 42 Virginia 180.66667  
## 43 Washington 148.41667  
## 44 Wyoming 140.00000  
## 45 Yveline 219.50000

aggregate(sales$Unit.Price ~ sales$State, data = sales, FUN = max)

## sales$State sales$Unit.Price  
## 1 Alabama 385.0000  
## 2 Arizona 649.6667  
## 3 Bayern 4008.0000  
## 4 Brandenburg 3264.0000  
## 5 California 3887.0000  
## 6 Charente-Maritime 2083.0000  
## 7 England 4113.0000  
## 8 Essonne 3423.0000  
## 9 Florida 636.0000  
## 10 Garonne (Haute) 3307.0000  
## 11 Georgia 52.0000  
## 12 Hamburg 3397.0000  
## 13 Hauts de Seine 3095.0000  
## 14 Hessen 5082.0000  
## 15 Illinois 629.0000  
## 16 Kentucky 1205.0000  
## 17 Loir et Cher 2211.0000  
## 18 Loiret 2697.0000  
## 19 Massachusetts 987.5000  
## 20 Minnesota 1234.0000  
## 21 Mississippi 919.0000  
## 22 Missouri 383.5000  
## 23 Montana 879.0000  
## 24 Moselle 2807.0000  
## 25 New York 554.3333  
## 26 Nord 3224.0000  
## 27 Nordrhein-Westfalen 3378.0000  
## 28 North Carolina 90.0000  
## 29 Ohio 1950.0000  
## 30 Oregon 2909.0000  
## 31 Pas de Calais 1006.0000  
## 32 Saarland 3427.0000  
## 33 Seine (Paris) 3261.0000  
## 34 Seine et Marne 2616.0000  
## 35 Seine Saint Denis 3326.0000  
## 36 Somme 3249.0000  
## 37 South Carolina 183.0000  
## 38 Texas 762.0000  
## 39 Utah 785.0000  
## 40 Val d'Oise 2253.0000  
## 41 Val de Marne 2499.0000  
## 42 Virginia 285.3333  
## 43 Washington 2954.0000  
## 44 Wyoming 698.0000  
## 45 Yveline 3495.0000

aggregate(sales$Unit.Price ~ sales$State, data = sales, FUN = min)

## sales$State sales$Unit.Price  
## 1 Alabama 19.666667  
## 2 Arizona 35.500000  
## 3 Bayern 2.000000  
## 4 Brandenburg 8.000000  
## 5 California 0.666667  
## 6 Charente-Maritime 2.000000  
## 7 England 0.666667  
## 8 Essonne 3.333333  
## 9 Florida 36.000000  
## 10 Garonne (Haute) 4.666667  
## 11 Georgia 50.000000  
## 12 Hamburg 1.000000  
## 13 Hauts de Seine 1.000000  
## 14 Hessen 1.500000  
## 15 Illinois 8.000000  
## 16 Kentucky 91.000000  
## 17 Loir et Cher 6.333333  
## 18 Loiret 1.000000  
## 19 Massachusetts 987.500000  
## 20 Minnesota 23.666667  
## 21 Mississippi 559.500000  
## 22 Missouri 383.500000  
## 23 Montana 300.000000  
## 24 Moselle 1.666667  
## 25 New York 21.000000  
## 26 Nord 1.000000  
## 27 Nordrhein-Westfalen 2.000000  
## 28 North Carolina 13.333333  
## 29 Ohio 18.000000  
## 30 Oregon 0.666667  
## 31 Pas de Calais 2.500000  
## 32 Saarland 1.500000  
## 33 Seine (Paris) 1.666667  
## 34 Seine et Marne 0.666667  
## 35 Seine Saint Denis 1.000000  
## 36 Somme 3.666667  
## 37 South Carolina 13.500000  
## 38 Texas 6.000000  
## 39 Utah 31.000000  
## 40 Val d'Oise 1.333333  
## 41 Val de Marne 2.666667  
## 42 Virginia 76.000000  
## 43 Washington 1.000000  
## 44 Wyoming 80.000000  
## 45 Yveline 1.666667

aggregate(sales$Cost ~ sales$State, data = sales, FUN = mean)

## sales$State sales$Cost  
## 1 Alabama 501.0000  
## 2 Arizona 1068.0000  
## 3 Bayern 619.2504  
## 4 Brandenburg 730.5213  
## 5 California 550.1227  
## 6 Charente-Maritime 559.8429  
## 7 England 614.7430  
## 8 Essonne 676.5032  
## 9 Florida 519.1429  
## 10 Garonne (Haute) 631.9400  
## 11 Georgia 87.5000  
## 12 Hamburg 643.9333  
## 13 Hauts de Seine 624.9862  
## 14 Hessen 639.1781  
## 15 Illinois 359.2143  
## 16 Kentucky 363.5000  
## 17 Loir et Cher 487.1228  
## 18 Loiret 605.9888  
## 19 Massachusetts 2049.0000  
## 20 Minnesota 619.6667  
## 21 Mississippi 897.5000  
## 22 Missouri 665.0000  
## 23 Montana 670.0000  
## 24 Moselle 612.2320  
## 25 New York 328.1000  
## 26 Nord 611.7624  
## 27 Nordrhein-Westfalen 609.9769  
## 28 North Carolina 59.5000  
## 29 Ohio 530.0000  
## 30 Oregon 525.3164  
## 31 Pas de Calais 470.8571  
## 32 Saarland 636.6193  
## 33 Seine (Paris) 606.5485  
## 34 Seine et Marne 662.7377  
## 35 Seine Saint Denis 599.4774  
## 36 Somme 558.7869  
## 37 South Carolina 100.6000  
## 38 Texas 248.5000  
## 39 Utah 531.0000  
## 40 Val d'Oise 519.6615  
## 41 Val de Marne 519.7200  
## 42 Virginia 382.5000  
## 43 Washington 512.3855  
## 44 Wyoming 833.6667  
## 45 Yveline 654.4253

aggregate(sales$Cost ~ sales$State, data = sales, FUN = median)

## sales$State sales$Cost  
## 1 Alabama 501.0  
## 2 Arizona 1068.0  
## 3 Bayern 315.0  
## 4 Brandenburg 474.0  
## 5 California 245.0  
## 6 Charente-Maritime 233.5  
## 7 England 293.0  
## 8 Essonne 330.0  
## 9 Florida 210.0  
## 10 Garonne (Haute) 315.0  
## 11 Georgia 87.5  
## 12 Hamburg 350.0  
## 13 Hauts de Seine 294.0  
## 14 Hessen 300.0  
## 15 Illinois 151.0  
## 16 Kentucky 184.5  
## 17 Loir et Cher 216.0  
## 18 Loiret 307.5  
## 19 Massachusetts 2049.0  
## 20 Minnesota 756.0  
## 21 Mississippi 897.5  
## 22 Missouri 665.0  
## 23 Montana 670.0  
## 24 Moselle 275.0  
## 25 New York 125.0  
## 26 Nord 350.0  
## 27 Nordrhein-Westfalen 319.0  
## 28 North Carolina 59.5  
## 29 Ohio 150.0  
## 30 Oregon 210.0  
## 31 Pas de Calais 382.5  
## 32 Saarland 301.0  
## 33 Seine (Paris) 290.0  
## 34 Seine et Marne 464.0  
## 35 Seine Saint Denis 265.5  
## 36 Somme 230.0  
## 37 South Carolina 132.0  
## 38 Texas 115.0  
## 39 Utah 525.0  
## 40 Val d'Oise 151.5  
## 41 Val de Marne 300.0  
## 42 Virginia 382.5  
## 43 Washington 220.0  
## 44 Wyoming 117.0  
## 45 Yveline 376.0

aggregate(sales$Cost ~ sales$State, data = sales, FUN = max)

## sales$State sales$Cost  
## 1 Alabama 945  
## 2 Arizona 2071  
## 3 Bayern 2640  
## 4 Brandenburg 2443  
## 5 California 3600  
## 6 Charente-Maritime 2384  
## 7 England 3120  
## 8 Essonne 2443  
## 9 Florida 2071  
## 10 Garonne (Haute) 2443  
## 11 Georgia 90  
## 12 Hamburg 2443  
## 13 Hauts de Seine 2443  
## 14 Hessen 3240  
## 15 Illinois 2071  
## 16 Kentucky 1015  
## 17 Loir et Cher 2320  
## 18 Loiret 2443  
## 19 Massachusetts 2049  
## 20 Minnesota 1050  
## 21 Mississippi 945  
## 22 Missouri 665  
## 23 Montana 815  
## 24 Moselle 2443  
## 25 New York 1701  
## 26 Nord 2443  
## 27 Nordrhein-Westfalen 2760  
## 28 North Carolina 80  
## 29 Ohio 1842  
## 30 Oregon 2443  
## 31 Pas de Calais 2295  
## 32 Saarland 3120  
## 33 Seine (Paris) 2443  
## 34 Seine et Marne 2443  
## 35 Seine Saint Denis 2443  
## 36 Somme 2295  
## 37 South Carolina 171  
## 38 Texas 805  
## 39 Utah 910  
## 40 Val d'Oise 2443  
## 41 Val de Marne 2384  
## 42 Virginia 700  
## 43 Washington 3240  
## 44 Wyoming 2320  
## 45 Yveline 2443

aggregate(sales$Cost ~ sales$State, data = sales, FUN = min)

## sales$State sales$Cost  
## 1 Alabama 57  
## 2 Arizona 65  
## 3 Bayern 4  
## 4 Brandenburg 5  
## 5 California 2  
## 6 Charente-Maritime 5  
## 7 England 2  
## 8 Essonne 8  
## 9 Florida 60  
## 10 Garonne (Haute) 10  
## 11 Georgia 85  
## 12 Hamburg 2  
## 13 Hauts de Seine 2  
## 14 Hessen 2  
## 15 Illinois 7  
## 16 Kentucky 70  
## 17 Loir et Cher 15  
## 18 Loiret 2  
## 19 Massachusetts 2049  
## 20 Minnesota 53  
## 21 Mississippi 850  
## 22 Missouri 665  
## 23 Montana 525  
## 24 Moselle 4  
## 25 New York 50  
## 26 Nord 2  
## 27 Nordrhein-Westfalen 2  
## 28 North Carolina 39  
## 29 Ohio 20  
## 30 Oregon 2  
## 31 Pas de Calais 5  
## 32 Saarland 2  
## 33 Seine (Paris) 4  
## 34 Seine et Marne 2  
## 35 Seine Saint Denis 2  
## 36 Somme 9  
## 37 South Carolina 20  
## 38 Texas 5  
## 39 Utah 28  
## 40 Val d'Oise 4  
## 41 Val de Marne 5  
## 42 Virginia 65  
## 43 Washington 2  
## 44 Wyoming 64  
## 45 Yveline 4

aggregate(sales$Revenue ~ sales$State, data = sales, FUN = mean)

## sales$State sales$Revenue  
## 1 Alabama 607.0000  
## 2 Arizona 1010.0000  
## 3 Bayern 805.6672  
## 4 Brandenburg 940.8298  
## 5 California 588.1645  
## 6 Charente-Maritime 540.9571  
## 7 England 665.9741  
## 8 Essonne 725.9505  
## 9 Florida 508.7143  
## 10 Garonne (Haute) 696.8700  
## 11 Georgia 102.0000  
## 12 Hamburg 821.6755  
## 13 Hauts de Seine 687.7505  
## 14 Hessen 824.7365  
## 15 Illinois 359.2143  
## 16 Kentucky 436.7500  
## 17 Loir et Cher 554.1579  
## 18 Loiret 666.0393  
## 19 Massachusetts 1975.0000  
## 20 Minnesota 720.3333  
## 21 Mississippi 1019.0000  
## 22 Missouri 767.0000  
## 23 Montana 739.5000  
## 24 Moselle 667.2376  
## 25 New York 355.4000  
## 26 Nord 651.6798  
## 27 Nordrhein-Westfalen 795.6251  
## 28 North Carolina 65.0000  
## 29 Ohio 584.3077  
## 30 Oregon 565.4890  
## 31 Pas de Calais 548.5238  
## 32 Saarland 820.3916  
## 33 Seine (Paris) 658.5604  
## 34 Seine et Marne 726.3770  
## 35 Seine Saint Denis 653.7412  
## 36 Somme 590.7541  
## 37 South Carolina 128.0000  
## 38 Texas 286.0000  
## 39 Utah 552.8000  
## 40 Val d'Oise 586.4769  
## 41 Val de Marne 570.8667  
## 42 Virginia 466.0000  
## 43 Washington 552.1735  
## 44 Wyoming 771.3333  
## 45 Yveline 716.1765

aggregate(sales$Revenue ~ sales$State, data = sales, FUN = median)

## sales$State sales$Revenue  
## 1 Alabama 607.0  
## 2 Arizona 1010.0  
## 3 Bayern 454.0  
## 4 Brandenburg 615.5  
## 5 California 285.5  
## 6 Charente-Maritime 258.5  
## 7 England 355.0  
## 8 Essonne 385.0  
## 9 Florida 250.0  
## 10 Garonne (Haute) 370.5  
## 11 Georgia 102.0  
## 12 Hamburg 493.0  
## 13 Hauts de Seine 381.0  
## 14 Hessen 443.0  
## 15 Illinois 191.0  
## 16 Kentucky 225.5  
## 17 Loir et Cher 228.0  
## 18 Loiret 366.0  
## 19 Massachusetts 1975.0  
## 20 Minnesota 856.0  
## 21 Mississippi 1019.0  
## 22 Missouri 767.0  
## 23 Montana 739.5  
## 24 Moselle 321.0  
## 25 New York 139.0  
## 26 Nord 382.0  
## 27 Nordrhein-Westfalen 465.0  
## 28 North Carolina 65.0  
## 29 Ohio 161.0  
## 30 Oregon 248.0  
## 31 Pas de Calais 388.5  
## 32 Saarland 450.0  
## 33 Seine (Paris) 355.0  
## 34 Seine et Marne 481.0  
## 35 Seine Saint Denis 342.5  
## 36 Somme 276.0  
## 37 South Carolina 170.0  
## 38 Texas 131.5  
## 39 Utah 575.0  
## 40 Val d'Oise 207.5  
## 41 Val de Marne 363.0  
## 42 Virginia 466.0  
## 43 Washington 259.0  
## 44 Wyoming 140.0  
## 45 Yveline 393.5

aggregate(sales$Revenue ~ sales$State, data = sales, FUN = max)

## sales$State sales$Revenue  
## 1 Alabama 1155  
## 2 Arizona 1949  
## 3 Bayern 4008  
## 4 Brandenburg 3264  
## 5 California 4923  
## 6 Charente-Maritime 2625  
## 7 England 4113  
## 8 Essonne 3423  
## 9 Florida 1908  
## 10 Garonne (Haute) 3307  
## 11 Georgia 104  
## 12 Hamburg 3561  
## 13 Hauts de Seine 3681  
## 14 Hessen 5082  
## 15 Illinois 1720  
## 16 Kentucky 1205  
## 17 Loir et Cher 2652  
## 18 Loiret 3137  
## 19 Massachusetts 1975  
## 20 Minnesota 1234  
## 21 Mississippi 1119  
## 22 Missouri 767  
## 23 Montana 879  
## 24 Moselle 3282  
## 25 New York 1663  
## 26 Nord 3310  
## 27 Nordrhein-Westfalen 4219  
## 28 North Carolina 90  
## 29 Ohio 1950  
## 30 Oregon 2909  
## 31 Pas de Calais 2853  
## 32 Saarland 4550  
## 33 Seine (Paris) 3409  
## 34 Seine et Marne 2990  
## 35 Seine Saint Denis 3326  
## 36 Somme 3249  
## 37 South Carolina 215  
## 38 Texas 934  
## 39 Utah 892  
## 40 Val d'Oise 3406  
## 41 Val de Marne 2779  
## 42 Virginia 856  
## 43 Washington 4216  
## 44 Wyoming 2094  
## 45 Yveline 3495

aggregate(sales$Revenue ~ sales$State, data = sales, FUN = min)

## sales$State sales$Revenue  
## 1 Alabama 59  
## 2 Arizona 71  
## 3 Bayern 6  
## 4 Brandenburg 8  
## 5 California 2  
## 6 Charente-Maritime 4  
## 7 England 2  
## 8 Essonne 9  
## 9 Florida 64  
## 10 Garonne (Haute) 14  
## 11 Georgia 100  
## 12 Hamburg 3  
## 13 Hauts de Seine 2  
## 14 Hessen 3  
## 15 Illinois 8  
## 16 Kentucky 91  
## 17 Loir et Cher 19  
## 18 Loiret 2  
## 19 Massachusetts 1975  
## 20 Minnesota 71  
## 21 Mississippi 919  
## 22 Missouri 767  
## 23 Montana 600  
## 24 Moselle 4  
## 25 New York 59  
## 26 Nord 2  
## 27 Nordrhein-Westfalen 3  
## 28 North Carolina 40  
## 29 Ohio 23  
## 30 Oregon 2  
## 31 Pas de Calais 5  
## 32 Saarland 3  
## 33 Seine (Paris) 4  
## 34 Seine et Marne 2  
## 35 Seine Saint Denis 3  
## 36 Somme 10  
## 37 South Carolina 27  
## 38 Texas 6  
## 39 Utah 31  
## 40 Val d'Oise 4  
## 41 Val de Marne 8  
## 42 Virginia 76  
## 43 Washington 2  
## 44 Wyoming 80  
## 45 Yveline 4

aggregate(sales$Unit\_Margin ~ sales$State, data = sales, FUN = mean)

## sales$State sales$Unit\_Margin  
## 1 Alabama 35.333334  
## 2 Arizona -18.831667  
## 3 Bayern 114.130635  
## 4 Brandenburg 123.438050  
## 5 California 23.526271  
## 6 Charente-Maritime -23.447381  
## 7 England 30.973983  
## 8 Essonne 37.979620  
## 9 Florida 5.548095  
## 10 Garonne (Haute) 41.566567  
## 11 Georgia 7.250000  
## 12 Hamburg 107.239804  
## 13 Hauts de Seine 33.057413  
## 14 Hessen 113.433234  
## 15 Illinois 6.523809  
## 16 Kentucky 62.750000  
## 17 Loir et Cher 31.672456  
## 18 Loiret 32.467341  
## 19 Massachusetts -37.000000  
## 20 Minnesota 79.998889  
## 21 Mississippi 78.000000  
## 22 Missouri 51.000000  
## 23 Montana 50.750000  
## 24 Moselle 39.268158  
## 25 New York 23.700333  
## 26 Nord 23.040313  
## 27 Nordrhein-Westfalen 114.824799  
## 28 North Carolina 5.166666  
## 29 Ohio 48.858462  
## 30 Oregon 23.754079  
## 31 Pas de Calais 43.515873  
## 32 Saarland 114.079417  
## 33 Seine (Paris) 29.430177  
## 34 Seine et Marne 34.925410  
## 35 Seine Saint Denis 28.641214  
## 36 Somme 21.300546  
## 37 South Carolina 15.899333  
## 38 Texas 20.905238  
## 39 Utah 15.500000  
## 40 Val d'Oise 30.745026  
## 41 Val de Marne 30.455333  
## 42 Virginia 31.501666  
## 43 Washington 23.912695  
## 44 Wyoming -12.110000  
## 45 Yveline 36.967353

aggregate(sales$Unit\_Margin ~ sales$State, data = sales, FUN = median)

## sales$State sales$Unit\_Margin  
## 1 Alabama 35.3333335  
## 2 Arizona -18.8316665  
## 3 Bayern 60.6633330  
## 4 Brandenburg 56.9166665  
## 5 California 10.5000000  
## 6 Charente-Maritime 0.6683335  
## 7 England 14.0000000  
## 8 Essonne 11.0000000  
## 9 Florida 6.0000000  
## 10 Garonne (Haute) 8.2500000  
## 11 Georgia 7.2500000  
## 12 Hamburg 56.0000000  
## 13 Hauts de Seine 9.6666670  
## 14 Hessen 59.5000000  
## 15 Illinois 4.9983335  
## 16 Kentucky 21.0000000  
## 17 Loir et Cher 18.0000000  
## 18 Loiret 10.0000000  
## 19 Massachusetts -37.0000000  
## 20 Minnesota 50.0000000  
## 21 Mississippi 78.0000000  
## 22 Missouri 51.0000000  
## 23 Montana 50.7500000  
## 24 Moselle 13.0000000  
## 25 New York 11.5000000  
## 26 Nord 6.5000000  
## 27 Nordrhein-Westfalen 59.6633330  
## 28 North Carolina 5.1666665  
## 29 Ohio 17.0000000  
## 30 Oregon 11.0000000  
## 31 Pas de Calais 14.8333335  
## 32 Saarland 57.0000000  
## 33 Seine (Paris) 10.0000000  
## 34 Seine et Marne 13.0000000  
## 35 Seine Saint Denis 11.5000000  
## 36 Somme 7.0000000  
## 37 South Carolina 12.6666670  
## 38 Texas 8.5000000  
## 39 Utah 15.5000000  
## 40 Val d'Oise 9.0016665  
## 41 Val de Marne 5.5000000  
## 42 Virginia 31.5016665  
## 43 Washington 10.3300000  
## 44 Wyoming 16.0000000  
## 45 Yveline 10.0000000

aggregate(sales$Unit\_Margin ~ sales$State, data = sales, FUN = max)

## sales$State sales$Unit\_Margin  
## 1 Alabama 70.00000  
## 2 Arizona 3.00000  
## 3 Bayern 1368.00000  
## 4 Brandenburg 880.00000  
## 5 California 845.00000  
## 6 Charente-Maritime 191.66667  
## 7 England 993.00000  
## 8 Essonne 1128.00000  
## 9 Florida 45.50000  
## 10 Garonne (Haute) 864.00000  
## 11 Georgia 9.50000  
## 12 Hamburg 954.00000  
## 13 Hauts de Seine 692.00000  
## 14 Hessen 1842.00000  
## 15 Illinois 90.00000  
## 16 Kentucky 190.00000  
## 17 Loir et Cher 281.00000  
## 18 Loiret 628.00000  
## 19 Massachusetts -37.00000  
## 20 Minnesota 184.00000  
## 21 Mississippi 87.00000  
## 22 Missouri 51.00000  
## 23 Montana 64.00000  
## 24 Moselle 1037.00000  
## 25 New York 90.00000  
## 26 Nord 781.00000  
## 27 Nordrhein-Westfalen 1058.00000  
## 28 North Carolina 10.00000  
## 29 Ohio 301.00000  
## 30 Oregon 589.00000  
## 31 Pas de Calais 253.00000  
## 32 Saarland 1261.00000  
## 33 Seine (Paris) 941.00000  
## 34 Seine et Marne 532.00000  
## 35 Seine Saint Denis 1031.00000  
## 36 Somme 954.00000  
## 37 South Carolina 38.00000  
## 38 Texas 71.00000  
## 39 Utah 43.00000  
## 40 Val d'Oise 321.00333  
## 41 Val de Marne 987.00000  
## 42 Virginia 52.00333  
## 43 Washington 565.00000  
## 44 Wyoming 23.00000  
## 45 Yveline 1167.00000

aggregate(sales$Unit\_Margin ~ sales$State, data = sales, FUN = min)

## sales$State sales$Unit\_Margin  
## 1 Alabama 0.666667  
## 2 Arizona -40.663333  
## 3 Bayern 0.000000  
## 4 Brandenburg 3.000000  
## 5 California -472.000000  
## 6 Charente-Maritime -562.000000  
## 7 England -376.000000  
## 8 Essonne -550.000000  
## 9 Florida -54.330000  
## 10 Garonne (Haute) -742.000000  
## 11 Georgia 5.000000  
## 12 Hamburg 0.330000  
## 13 Hauts de Seine -752.000000  
## 14 Hessen -31.000000  
## 15 Illinois -116.996667  
## 16 Kentucky 19.000000  
## 17 Loir et Cher -234.000000  
## 18 Loiret -777.000000  
## 19 Massachusetts -37.000000  
## 20 Minnesota 5.996667  
## 21 Mississippi 69.000000  
## 22 Missouri 51.000000  
## 23 Montana 37.500000  
## 24 Moselle -666.000000  
## 25 New York -12.666667  
## 26 Nord -741.000000  
## 27 Nordrhein-Westfalen -9.336667  
## 28 North Carolina 0.333333  
## 29 Ohio 0.663333  
## 30 Oregon -460.000000  
## 31 Pas de Calais -107.000000  
## 32 Saarland -1.500000  
## 33 Seine (Paris) -937.000000  
## 34 Seine et Marne -700.000000  
## 35 Seine Saint Denis -932.000000  
## 36 Somme -599.000000  
## 37 South Carolina 3.330000  
## 38 Texas -7.000000  
## 39 Utah -9.000000  
## 40 Val d'Oise -418.000000  
## 41 Val de Marne -518.000000  
## 42 Virginia 11.000000  
## 43 Washington -424.000000  
## 44 Wyoming -75.330000  
## 45 Yveline -749.000000

aggregate(sales$Unit\_Margin\_percent ~ sales$State, data = sales, FUN = mean)

## sales$State sales$Unit\_Margin\_percent  
## 1 Alabama 10.785000  
## 2 Arizona 1.095000  
## 3 Bayern 26.833733  
## 4 Brandenburg 25.895426  
## 5 California 10.948117  
## 6 Charente-Maritime 2.076714  
## 7 England 12.415432  
## 8 Essonne 9.641699  
## 9 Florida 8.347143  
## 10 Garonne (Haute) 7.720000  
## 11 Georgia 14.135000  
## 12 Hamburg 26.169586  
## 13 Hauts de Seine 9.974951  
## 14 Hessen 26.766187  
## 15 Illinois 8.299286  
## 16 Kentucky 18.360000  
## 17 Loir et Cher 11.095789  
## 18 Loiret 9.529045  
## 19 Massachusetts -3.750000  
## 20 Minnesota 17.310000  
## 21 Mississippi 11.530000  
## 22 Missouri 13.300000  
## 23 Montana 9.890000  
## 24 Moselle 9.576243  
## 25 New York 14.155000  
## 26 Nord 7.732910  
## 27 Nordrhein-Westfalen 26.881153  
## 28 North Carolina 6.805000  
## 29 Ohio 10.561538  
## 30 Oregon 11.387428  
## 31 Pas de Calais 12.183571  
## 32 Saarland 26.497848  
## 33 Seine (Paris) 8.591465  
## 34 Seine et Marne 9.929672  
## 35 Seine Saint Denis 10.807751  
## 36 Somme 7.528852  
## 37 South Carolina 22.344000  
## 38 Texas 14.263571  
## 39 Utah 5.656000  
## 40 Val d'Oise 12.440846  
## 41 Val de Marne 9.563467  
## 42 Virginia 16.350000  
## 43 Washington 11.379168  
## 44 Wyoming 8.546667  
## 45 Yveline 9.661742

aggregate(sales$Unit\_Margin\_percent ~ sales$State, data = sales, FUN = median)

## sales$State sales$Unit\_Margin\_percent  
## 1 Alabama 10.785  
## 2 Arizona 1.095  
## 3 Bayern 28.170  
## 4 Brandenburg 27.110  
## 5 California 12.340  
## 6 Charente-Maritime 3.910  
## 7 England 13.750  
## 8 Essonne 14.290  
## 9 Florida 16.000  
## 10 Garonne (Haute) 14.195  
## 11 Georgia 14.135  
## 12 Hamburg 27.260  
## 13 Hauts de Seine 14.610  
## 14 Hessen 27.860  
## 15 Illinois 13.075  
## 16 Kentucky 17.520  
## 17 Loir et Cher 17.010  
## 18 Loiret 12.315  
## 19 Massachusetts -3.750  
## 20 Minnesota 14.910  
## 21 Mississippi 11.530  
## 22 Missouri 13.300  
## 23 Montana 9.890  
## 24 Moselle 15.250  
## 25 New York 16.270  
## 26 Nord 11.110  
## 27 Nordrhein-Westfalen 28.190  
## 28 North Carolina 6.805  
## 29 Ohio 9.510  
## 30 Oregon 12.500  
## 31 Pas de Calais 13.745  
## 32 Saarland 27.840  
## 33 Seine (Paris) 12.665  
## 34 Seine et Marne 12.530  
## 35 Seine Saint Denis 14.410  
## 36 Somme 11.100  
## 37 South Carolina 22.200  
## 38 Texas 16.670  
## 39 Utah 6.440  
## 40 Val d'Oise 16.000  
## 41 Val de Marne 13.790  
## 42 Virginia 16.350  
## 43 Washington 12.730  
## 44 Wyoming 16.430  
## 45 Yveline 13.470

aggregate(sales$Unit\_Margin\_percent ~ sales$State, data = sales, FUN = max)

## sales$State sales$Unit\_Margin\_percent  
## 1 Alabama 18.18  
## 2 Arizona 8.45  
## 3 Bayern 41.52  
## 4 Brandenburg 38.90  
## 5 California 33.33  
## 6 Charente-Maritime 41.39  
## 7 England 34.15  
## 8 Essonne 43.51  
## 9 Florida 24.53  
## 10 Garonne (Haute) 38.78  
## 11 Georgia 18.27  
## 12 Hamburg 49.75  
## 13 Hauts de Seine 46.11  
## 14 Hessen 41.72  
## 15 Illinois 23.91  
## 16 Kentucky 23.08  
## 17 Loir et Cher 42.19  
## 18 Loiret 43.18  
## 19 Massachusetts -3.75  
## 20 Minnesota 25.34  
## 21 Mississippi 15.55  
## 22 Missouri 13.30  
## 23 Montana 12.50  
## 24 Moselle 45.72  
## 25 New York 22.61  
## 26 Nord 46.00  
## 27 Nordrhein-Westfalen 41.67  
## 28 North Carolina 11.11  
## 29 Ohio 22.28  
## 30 Oregon 33.52  
## 31 Pas de Calais 42.31  
## 32 Saarland 42.86  
## 33 Seine (Paris) 44.77  
## 34 Seine et Marne 44.74  
## 35 Seine Saint Denis 49.75  
## 36 Somme 37.50  
## 37 South Carolina 25.93  
## 38 Texas 24.73  
## 39 Utah 9.68  
## 40 Val d'Oise 43.26  
## 41 Val de Marne 42.55  
## 42 Virginia 18.23  
## 43 Washington 33.33  
## 44 Wyoming 20.00  
## 45 Yveline 44.71

aggregate(sales$Unit\_Margin\_percent ~ sales$State, data = sales, FUN = min)

## sales$State sales$Unit\_Margin\_percent  
## 1 Alabama 3.39  
## 2 Arizona -6.26  
## 3 Bayern 0.00  
## 4 Brandenburg 1.02  
## 5 California -29.89  
## 6 Charente-Maritime -55.63  
## 7 England -23.45  
## 8 Essonne -60.71  
## 9 Florida -13.15  
## 10 Garonne (Haute) -56.77  
## 11 Georgia 10.00  
## 12 Hamburg 0.41  
## 13 Hauts de Seine -53.52  
## 14 Hessen -2.85  
## 15 Illinois -20.41  
## 16 Kentucky 15.32  
## 17 Loir et Cher -28.23  
## 18 Loiret -60.09  
## 19 Massachusetts -3.75  
## 20 Minnesota 11.68  
## 21 Mississippi 7.51  
## 22 Missouri 13.30  
## 23 Montana 7.28  
## 24 Moselle -68.67  
## 25 New York -2.29  
## 26 Nord -61.19  
## 27 Nordrhein-Westfalen -1.19  
## 28 North Carolina 2.50  
## 29 Ohio 2.12  
## 30 Oregon -28.55  
## 31 Pas de Calais -24.71  
## 32 Saarland -0.56  
## 33 Seine (Paris) -66.99  
## 34 Seine et Marne -65.24  
## 35 Seine Saint Denis -61.80  
## 36 Somme -56.86  
## 37 South Carolina 20.47  
## 38 Texas -5.88  
## 39 Utah -2.02  
## 40 Val d'Oise -32.97  
## 41 Val de Marne -34.02  
## 42 Virginia 14.47  
## 43 Washington -29.93  
## 44 Wyoming -10.79  
## 45 Yveline -67.49

aggregate(sales$Margin ~ sales$State, data = sales, FUN = mean)

## sales$State sales$Margin  
## 1 Alabama 106.00000  
## 2 Arizona -58.00000  
## 3 Bayern 186.41679  
## 4 Brandenburg 210.30851  
## 5 California 38.04181  
## 6 Charente-Maritime -18.88571  
## 7 England 51.23112  
## 8 Essonne 49.44731  
## 9 Florida -10.42857  
## 10 Garonne (Haute) 64.93000  
## 11 Georgia 14.50000  
## 12 Hamburg 177.74223  
## 13 Hauts de Seine 62.76424  
## 14 Hessen 185.55845  
## 15 Illinois 0.00000  
## 16 Kentucky 73.25000  
## 17 Loir et Cher 67.03509  
## 18 Loiret 60.05056  
## 19 Massachusetts -74.00000  
## 20 Minnesota 100.66667  
## 21 Mississippi 121.50000  
## 22 Missouri 102.00000  
## 23 Montana 69.50000  
## 24 Moselle 55.00552  
## 25 New York 27.30000  
## 26 Nord 39.91741  
## 27 Nordrhein-Westfalen 185.64816  
## 28 North Carolina 5.50000  
## 29 Ohio 54.30769  
## 30 Oregon 40.17253  
## 31 Pas de Calais 77.66667  
## 32 Saarland 183.77234  
## 33 Seine (Paris) 52.01190  
## 34 Seine et Marne 63.63934  
## 35 Seine Saint Denis 54.26382  
## 36 Somme 31.96721  
## 37 South Carolina 27.40000  
## 38 Texas 37.50000  
## 39 Utah 21.80000  
## 40 Val d'Oise 66.81538  
## 41 Val de Marne 51.14667  
## 42 Virginia 83.50000  
## 43 Washington 39.78805  
## 44 Wyoming -62.33333  
## 45 Yveline 61.75113

aggregate(sales$Margin ~ sales$State, data = sales, FUN = median)

## sales$State sales$Margin  
## 1 Alabama 106.0  
## 2 Arizona -58.0  
## 3 Bayern 116.0  
## 4 Brandenburg 125.0  
## 5 California 20.0  
## 6 Charente-Maritime 2.0  
## 7 England 26.0  
## 8 Essonne 20.0  
## 9 Florida 18.0  
## 10 Garonne (Haute) 15.0  
## 11 Georgia 14.5  
## 12 Hamburg 103.0  
## 13 Hauts de Seine 22.0  
## 14 Hessen 106.0  
## 15 Illinois 12.0  
## 16 Kentucky 42.0  
## 17 Loir et Cher 28.0  
## 18 Loiret 17.0  
## 19 Massachusetts -74.0  
## 20 Minnesota 100.0  
## 21 Mississippi 121.5  
## 22 Missouri 102.0  
## 23 Montana 69.5  
## 24 Moselle 22.0  
## 25 New York 14.0  
## 26 Nord 12.0  
## 27 Nordrhein-Westfalen 106.0  
## 28 North Carolina 5.5  
## 29 Ohio 19.0  
## 30 Oregon 19.0  
## 31 Pas de Calais 30.5  
## 32 Saarland 104.0  
## 33 Seine (Paris) 18.0  
## 34 Seine et Marne 24.0  
## 35 Seine Saint Denis 23.5  
## 36 Somme 12.0  
## 37 South Carolina 38.0  
## 38 Texas 10.5  
## 39 Utah 31.0  
## 40 Val d'Oise 21.0  
## 41 Val de Marne 11.0  
## 42 Virginia 83.5  
## 43 Washington 19.0  
## 44 Wyoming 16.0  
## 45 Yveline 19.0

aggregate(sales$Margin ~ sales$State, data = sales, FUN = max)

## sales$State sales$Margin  
## 1 Alabama 210  
## 2 Arizona 6  
## 3 Bayern 1368  
## 4 Brandenburg 899  
## 5 California 1323  
## 6 Charente-Maritime 575  
## 7 England 993  
## 8 Essonne 1128  
## 9 Florida 91  
## 10 Garonne (Haute) 899  
## 11 Georgia 19  
## 12 Hamburg 1169  
## 13 Hauts de Seine 1297  
## 14 Hessen 1842  
## 15 Illinois 102  
## 16 Kentucky 190  
## 17 Loir et Cher 580  
## 18 Loiret 948  
## 19 Massachusetts -74  
## 20 Minnesota 184  
## 21 Mississippi 174  
## 22 Missouri 102  
## 23 Montana 75  
## 24 Moselle 1037  
## 25 New York 93  
## 26 Nord 1006  
## 27 Nordrhein-Westfalen 1459  
## 28 North Carolina 10  
## 29 Ohio 301  
## 30 Oregon 720  
## 31 Pas de Calais 558  
## 32 Saarland 1430  
## 33 Seine (Paris) 1025  
## 34 Seine et Marne 695  
## 35 Seine Saint Denis 1274  
## 36 Somme 954  
## 37 South Carolina 44  
## 38 Texas 142  
## 39 Utah 50  
## 40 Val d'Oise 963  
## 41 Val de Marne 987  
## 42 Virginia 156  
## 43 Washington 976  
## 44 Wyoming 23  
## 45 Yveline 1167

aggregate(sales$Margin ~ sales$State, data = sales, FUN = min)

## sales$State sales$Margin  
## 1 Alabama 2  
## 2 Arizona -122  
## 3 Bayern 0  
## 4 Brandenburg 3  
## 5 California -496  
## 6 Charente-Maritime -780  
## 7 England -464  
## 8 Essonne -836  
## 9 Florida -163  
## 10 Garonne (Haute) -787  
## 11 Georgia 10  
## 12 Hamburg 1  
## 13 Hauts de Seine -752  
## 14 Hessen -31  
## 15 Illinois -351  
## 16 Kentucky 19  
## 17 Loir et Cher -287  
## 18 Loiret -819  
## 19 Massachusetts -74  
## 20 Minnesota 18  
## 21 Mississippi 69  
## 22 Missouri 102  
## 23 Montana 64  
## 24 Moselle -807  
## 25 New York -38  
## 26 Nord -785  
## 27 Nordrhein-Westfalen -28  
## 28 North Carolina 1  
## 29 Ohio 2  
## 30 Oregon -460  
## 31 Pas de Calais -107  
## 32 Saarland -3  
## 33 Seine (Paris) -937  
## 34 Seine et Marne -916  
## 35 Seine Saint Denis -932  
## 36 Somme -599  
## 37 South Carolina 7  
## 38 Texas -7  
## 39 Utah -18  
## 40 Val d'Oise -543  
## 41 Val de Marne -518  
## 42 Virginia 11  
## 43 Washington -464  
## 44 Wyoming -226  
## 45 Yveline -749

aggregate(sales$Margin\_percent ~ sales$State, data = sales, FUN = mean)

## sales$State sales$Margin\_percent  
## 1 Alabama 10.785000  
## 2 Arizona 1.095000  
## 3 Bayern 26.833913  
## 4 Brandenburg 25.895957  
## 5 California 10.948445  
## 6 Charente-Maritime 2.079143  
## 7 England 12.416180  
## 8 Essonne 9.641634  
## 9 Florida 8.347143  
## 10 Garonne (Haute) 7.720200  
## 11 Georgia 14.135000  
## 12 Hamburg 26.170127  
## 13 Hauts de Seine 9.976621  
## 14 Hessen 26.766466  
## 15 Illinois 8.299286  
## 16 Kentucky 18.360000  
## 17 Loir et Cher 11.097018  
## 18 Loiret 9.528820  
## 19 Massachusetts -3.750000  
## 20 Minnesota 17.313333  
## 21 Mississippi 11.530000  
## 22 Missouri 13.300000  
## 23 Montana 9.890000  
## 24 Moselle 9.572873  
## 25 New York 14.154000  
## 26 Nord 7.733278  
## 27 Nordrhein-Westfalen 26.880760  
## 28 North Carolina 6.805000  
## 29 Ohio 10.563846  
## 30 Oregon 11.388373  
## 31 Pas de Calais 12.184286  
## 32 Saarland 26.498361  
## 33 Seine (Paris) 8.591740  
## 34 Seine et Marne 9.933825  
## 35 Seine Saint Denis 10.808505  
## 36 Somme 7.528689  
## 37 South Carolina 22.348000  
## 38 Texas 14.262143  
## 39 Utah 5.656000  
## 40 Val d'Oise 12.436077  
## 41 Val de Marne 9.566533  
## 42 Virginia 16.345000  
## 43 Washington 11.379550  
## 44 Wyoming 8.546667  
## 45 Yveline 9.662873

aggregate(sales$Margin\_percent ~ sales$State, data = sales, FUN = median)

## sales$State sales$Margin\_percent  
## 1 Alabama 10.785  
## 2 Arizona 1.095  
## 3 Bayern 28.170  
## 4 Brandenburg 27.110  
## 5 California 12.340  
## 6 Charente-Maritime 3.920  
## 7 England 13.750  
## 8 Essonne 14.290  
## 9 Florida 16.000  
## 10 Garonne (Haute) 14.195  
## 11 Georgia 14.135  
## 12 Hamburg 27.260  
## 13 Hauts de Seine 14.610  
## 14 Hessen 27.865  
## 15 Illinois 13.075  
## 16 Kentucky 17.520  
## 17 Loir et Cher 17.010  
## 18 Loiret 12.315  
## 19 Massachusetts -3.750  
## 20 Minnesota 14.910  
## 21 Mississippi 11.530  
## 22 Missouri 13.300  
## 23 Montana 9.890  
## 24 Moselle 15.250  
## 25 New York 16.270  
## 26 Nord 11.110  
## 27 Nordrhein-Westfalen 28.190  
## 28 North Carolina 6.805  
## 29 Ohio 9.510  
## 30 Oregon 12.500  
## 31 Pas de Calais 13.740  
## 32 Saarland 27.840  
## 33 Seine (Paris) 12.665  
## 34 Seine et Marne 12.530  
## 35 Seine Saint Denis 14.410  
## 36 Somme 11.100  
## 37 South Carolina 22.220  
## 38 Texas 16.670  
## 39 Utah 6.440  
## 40 Val d'Oise 16.000  
## 41 Val de Marne 13.790  
## 42 Virginia 16.345  
## 43 Washington 12.730  
## 44 Wyoming 16.430  
## 45 Yveline 13.470

aggregate(sales$Margin\_percent ~ sales$State, data = sales, FUN = max)

## sales$State sales$Margin\_percent  
## 1 Alabama 18.18  
## 2 Arizona 8.45  
## 3 Bayern 41.52  
## 4 Brandenburg 38.90  
## 5 California 33.33  
## 6 Charente-Maritime 41.38  
## 7 England 34.15  
## 8 Essonne 43.51  
## 9 Florida 24.53  
## 10 Garonne (Haute) 38.78  
## 11 Georgia 18.27  
## 12 Hamburg 50.00  
## 13 Hauts de Seine 46.11  
## 14 Hessen 41.72  
## 15 Illinois 23.91  
## 16 Kentucky 23.08  
## 17 Loir et Cher 42.19  
## 18 Loiret 43.18  
## 19 Massachusetts -3.75  
## 20 Minnesota 25.35  
## 21 Mississippi 15.55  
## 22 Missouri 13.30  
## 23 Montana 12.50  
## 24 Moselle 45.72  
## 25 New York 22.61  
## 26 Nord 46.00  
## 27 Nordrhein-Westfalen 41.67  
## 28 North Carolina 11.11  
## 29 Ohio 22.28  
## 30 Oregon 33.52  
## 31 Pas de Calais 42.31  
## 32 Saarland 42.86  
## 33 Seine (Paris) 44.77  
## 34 Seine et Marne 44.74  
## 35 Seine Saint Denis 50.00  
## 36 Somme 37.50  
## 37 South Carolina 25.93  
## 38 Texas 24.73  
## 39 Utah 9.68  
## 40 Val d'Oise 43.26  
## 41 Val de Marne 42.55  
## 42 Virginia 18.22  
## 43 Washington 33.33  
## 44 Wyoming 20.00  
## 45 Yveline 44.71

aggregate(sales$Margin\_percent ~ sales$State, data = sales, FUN = min)

## sales$State sales$Margin\_percent  
## 1 Alabama 3.39  
## 2 Arizona -6.26  
## 3 Bayern 0.00  
## 4 Brandenburg 1.02  
## 5 California -29.89  
## 6 Charente-Maritime -55.63  
## 7 England -23.45  
## 8 Essonne -60.71  
## 9 Florida -13.15  
## 10 Garonne (Haute) -56.77  
## 11 Georgia 10.00  
## 12 Hamburg 0.41  
## 13 Hauts de Seine -53.52  
## 14 Hessen -2.85  
## 15 Illinois -20.41  
## 16 Kentucky 15.32  
## 17 Loir et Cher -28.21  
## 18 Loiret -60.09  
## 19 Massachusetts -3.75  
## 20 Minnesota 11.68  
## 21 Mississippi 7.51  
## 22 Missouri 13.30  
## 23 Montana 7.28  
## 24 Moselle -68.67  
## 25 New York -2.29  
## 26 Nord -61.19  
## 27 Nordrhein-Westfalen -1.19  
## 28 North Carolina 2.50  
## 29 Ohio 2.12  
## 30 Oregon -28.55  
## 31 Pas de Calais -24.71  
## 32 Saarland -0.56  
## 33 Seine (Paris) -66.99  
## 34 Seine et Marne -65.24  
## 35 Seine Saint Denis -61.80  
## 36 Somme -56.86  
## 37 South Carolina 20.47  
## 38 Texas -5.88  
## 39 Utah -2.02  
## 40 Val d'Oise -32.97  
## 41 Val de Marne -34.02  
## 42 Virginia 14.47  
## 43 Washington -29.93  
## 44 Wyoming -10.79  
## 45 Yveline -67.49

aggregate(sales$Customer.Age ~ sales$Product.Category, data = sales, FUN = mean)

## sales$Product.Category sales$Customer.Age  
## 1 Accessories 36.56859  
## 2 Bikes 35.82109  
## 3 Clothing 36.34358

aggregate(sales$Customer.Age ~ sales$Product.Category, data = sales, FUN = median)

## sales$Product.Category sales$Customer.Age  
## 1 Accessories 35  
## 2 Bikes 34  
## 3 Clothing 35

aggregate(sales$Customer.Age ~ sales$Product.Category, data = sales, FUN = max)

## sales$Product.Category sales$Customer.Age  
## 1 Accessories 87  
## 2 Bikes 75  
## 3 Clothing 86

aggregate(sales$Customer.Age ~ sales$Product.Category, data = sales, FUN = min)

## sales$Product.Category sales$Customer.Age  
## 1 Accessories 17  
## 2 Bikes 17  
## 3 Clothing 17

aggregate(sales$Quantity ~ sales$Product.Category, data = sales, FUN = mean)

## sales$Product.Category sales$Quantity  
## 1 Accessories 1.999112  
## 2 Bikes 2.014803  
## 3 Clothing 2.000573

aggregate(sales$Quantity ~ sales$Product.Category, data = sales, FUN = median)

## sales$Product.Category sales$Quantity  
## 1 Accessories 2  
## 2 Bikes 2  
## 3 Clothing 2

aggregate(sales$Quantity ~ sales$Product.Category, data = sales, FUN = max)

## sales$Product.Category sales$Quantity  
## 1 Accessories 3  
## 2 Bikes 3  
## 3 Clothing 3

aggregate(sales$Quantity ~ sales$Product.Category, data = sales, FUN = min)

## sales$Product.Category sales$Quantity  
## 1 Accessories 1  
## 2 Bikes 1  
## 3 Clothing 1

aggregate(sales$Unit.Cost ~ sales$Product.Category, data = sales, FUN = mean)

## sales$Product.Category sales$Unit.Cost  
## 1 Accessories 163.2153  
## 2 Bikes 952.9479  
## 3 Clothing 336.3117

aggregate(sales$Unit.Cost ~ sales$Product.Category, data = sales, FUN = median)

## sales$Product.Category sales$Unit.Cost  
## 1 Accessories 75.00  
## 2 Bikes 773.33  
## 3 Clothing 198.00

aggregate(sales$Unit.Cost ~ sales$Product.Category, data = sales, FUN = max)

## sales$Product.Category sales$Unit.Cost  
## 1 Accessories 3240  
## 2 Bikes 2443  
## 3 Clothing 2100

aggregate(sales$Unit.Cost ~ sales$Product.Category, data = sales, FUN = min)

## sales$Product.Category sales$Unit.Cost  
## 1 Accessories 0.67  
## 2 Bikes 180.00  
## 3 Clothing 3.00

aggregate(sales$Unit.Price ~ sales$Product.Category, data = sales, FUN = mean)

## sales$Product.Category sales$Unit.Price  
## 1 Accessories 200.4656  
## 2 Bikes 981.5082  
## 3 Clothing 399.3197

aggregate(sales$Unit.Price ~ sales$Product.Category, data = sales, FUN = median)

## sales$Product.Category sales$Unit.Price  
## 1 Accessories 90.0  
## 2 Bikes 780.0  
## 3 Clothing 231.5

aggregate(sales$Unit.Price ~ sales$Product.Category, data = sales, FUN = max)

## sales$Product.Category sales$Unit.Price  
## 1 Accessories 5082  
## 2 Bikes 3495  
## 3 Clothing 3339

aggregate(sales$Unit.Price ~ sales$Product.Category, data = sales, FUN = min)

## sales$Product.Category sales$Unit.Price  
## 1 Accessories 0.666667  
## 2 Bikes 125.000000  
## 3 Clothing 2.666667

aggregate(sales$Cost ~ sales$Product.Category, data = sales, FUN = mean)

## sales$Product.Category sales$Cost  
## 1 Accessories 268.1345  
## 2 Bikes 1571.7967  
## 3 Clothing 552.0758

aggregate(sales$Cost ~ sales$Product.Category, data = sales, FUN = median)

## sales$Product.Category sales$Cost  
## 1 Accessories 130  
## 2 Bikes 1701  
## 3 Clothing 300

aggregate(sales$Cost ~ sales$Product.Category, data = sales, FUN = max)

## sales$Product.Category sales$Cost  
## 1 Accessories 3600  
## 2 Bikes 2443  
## 3 Clothing 2100

aggregate(sales$Cost ~ sales$Product.Category, data = sales, FUN = min)

## sales$Product.Category sales$Cost  
## 1 Accessories 2  
## 2 Bikes 540  
## 3 Clothing 9

aggregate(sales$Revenue ~ sales$Product.Category, data = sales, FUN = mean)

## sales$Product.Category sales$Revenue  
## 1 Accessories 329.3230  
## 2 Bikes 1619.3931  
## 3 Clothing 656.1475

aggregate(sales$Revenue ~ sales$Product.Category, data = sales, FUN = median)

## sales$Product.Category sales$Revenue  
## 1 Accessories 159  
## 2 Bikes 1732  
## 3 Clothing 376

aggregate(sales$Revenue ~ sales$Product.Category, data = sales, FUN = max)

## sales$Product.Category sales$Revenue  
## 1 Accessories 5082  
## 2 Bikes 3681  
## 3 Clothing 3339

aggregate(sales$Revenue ~ sales$Product.Category, data = sales, FUN = min)

## sales$Product.Category sales$Revenue  
## 1 Accessories 2  
## 2 Bikes 335  
## 3 Clothing 8

aggregate(sales$Unit\_Margin ~ sales$Product.Category, data = sales, FUN = mean)

## sales$Product.Category sales$Unit\_Margin  
## 1 Accessories 37.25038  
## 2 Bikes 28.56034  
## 3 Clothing 63.00796

aggregate(sales$Unit\_Margin ~ sales$Product.Category, data = sales, FUN = median)

## sales$Product.Category sales$Unit\_Margin  
## 1 Accessories 13.5  
## 2 Bikes 7.5  
## 3 Clothing 26.0

aggregate(sales$Unit\_Margin ~ sales$Product.Category, data = sales, FUN = max)

## sales$Product.Category sales$Unit\_Margin  
## 1 Accessories 1842  
## 2 Bikes 1167  
## 3 Clothing 1309

aggregate(sales$Unit\_Margin ~ sales$Product.Category, data = sales, FUN = min)

## sales$Product.Category sales$Unit\_Margin  
## 1 Accessories -208  
## 2 Bikes -937  
## 3 Clothing -426

aggregate(sales$Unit\_Margin\_percent ~ sales$Product.Category, data = sales, FUN = mean)

## sales$Product.Category sales$Unit\_Margin\_percent  
## 1 Accessories 16.978851  
## 2 Bikes 1.030368  
## 3 Clothing 14.798675

aggregate(sales$Unit\_Margin\_percent ~ sales$Product.Category, data = sales, FUN = median)

## sales$Product.Category sales$Unit\_Margin\_percent  
## 1 Accessories 17.24  
## 2 Bikes 1.28  
## 3 Clothing 15.62

aggregate(sales$Unit\_Margin\_percent ~ sales$Product.Category, data = sales, FUN = max)

## sales$Product.Category sales$Unit\_Margin\_percent  
## 1 Accessories 49.75  
## 2 Bikes 37.02  
## 3 Clothing 46.11

aggregate(sales$Unit\_Margin\_percent ~ sales$Product.Category, data = sales, FUN = min)

## sales$Product.Category sales$Unit\_Margin\_percent  
## 1 Accessories -40.66  
## 2 Bikes -68.67  
## 3 Clothing -50.00

aggregate(sales$Margin ~ sales$Product.Category, data = sales, FUN = mean)

## sales$Product.Category sales$Margin  
## 1 Accessories 61.18852  
## 2 Bikes 47.59636  
## 3 Clothing 104.07177

aggregate(sales$Margin ~ sales$Product.Category, data = sales, FUN = median)

## sales$Product.Category sales$Margin  
## 1 Accessories 25  
## 2 Bikes 15  
## 3 Clothing 48

aggregate(sales$Margin ~ sales$Product.Category, data = sales, FUN = max)

## sales$Product.Category sales$Margin  
## 1 Accessories 1842  
## 2 Bikes 1297  
## 3 Clothing 1309

aggregate(sales$Margin ~ sales$Product.Category, data = sales, FUN = min)

## sales$Product.Category sales$Margin  
## 1 Accessories -255  
## 2 Bikes -937  
## 3 Clothing -620

aggregate(sales$Margin\_percent ~ sales$Product.Category, data = sales, FUN = mean)

## sales$Product.Category sales$Margin\_percent  
## 1 Accessories 16.979606  
## 2 Bikes 1.030275  
## 3 Clothing 14.798643

aggregate(sales$Margin\_percent ~ sales$Product.Category, data = sales, FUN = median)

## sales$Product.Category sales$Margin\_percent  
## 1 Accessories 17.24  
## 2 Bikes 1.28  
## 3 Clothing 15.62

aggregate(sales$Margin\_percent ~ sales$Product.Category, data = sales, FUN = max)

## sales$Product.Category sales$Margin\_percent  
## 1 Accessories 50.00  
## 2 Bikes 37.02  
## 3 Clothing 46.11

aggregate(sales$Margin\_percent ~ sales$Product.Category, data = sales, FUN = min)

## sales$Product.Category sales$Margin\_percent  
## 1 Accessories -40.66  
## 2 Bikes -68.67  
## 3 Clothing -50.00

aggregate(sales$Customer.Age ~ sales$Sub.Category, data = sales, FUN = mean)

## sales$Sub.Category sales$Customer.Age  
## 1 Bike Racks 38.67961  
## 2 Bike Stands 36.77241  
## 3 Bottles and Cages 36.50340  
## 4 Caps 35.75214  
## 5 Cleaners 37.23119  
## 6 Fenders 37.90289  
## 7 Gloves 37.16042  
## 8 Helmets 36.44780  
## 9 Hydration Packs 37.26010  
## 10 Jerseys 36.13500  
## 11 Mountain Bikes 36.66496  
## 12 Road Bikes 34.99570  
## 13 Shorts 37.23498  
## 14 Socks 36.67582  
## 15 Tires and Tubes 36.47417  
## 16 Touring Bikes 35.95952  
## 17 Vests 37.29487

aggregate(sales$Customer.Age ~ sales$Sub.Category, data = sales, FUN = median)

## sales$Sub.Category sales$Customer.Age  
## 1 Bike Racks 38.0  
## 2 Bike Stands 35.0  
## 3 Bottles and Cages 36.0  
## 4 Caps 35.0  
## 5 Cleaners 37.0  
## 6 Fenders 37.0  
## 7 Gloves 36.0  
## 8 Helmets 35.0  
## 9 Hydration Packs 37.0  
## 10 Jerseys 34.0  
## 11 Mountain Bikes 35.0  
## 12 Road Bikes 33.0  
## 13 Shorts 36.5  
## 14 Socks 35.0  
## 15 Tires and Tubes 35.0  
## 16 Touring Bikes 35.0  
## 17 Vests 36.0

aggregate(sales$Customer.Age ~ sales$Sub.Category, data = sales, FUN = max)

## sales$Sub.Category sales$Customer.Age  
## 1 Bike Racks 87  
## 2 Bike Stands 84  
## 3 Bottles and Cages 78  
## 4 Caps 82  
## 5 Cleaners 78  
## 6 Fenders 86  
## 7 Gloves 84  
## 8 Helmets 85  
## 9 Hydration Packs 75  
## 10 Jerseys 75  
## 11 Mountain Bikes 75  
## 12 Road Bikes 65  
## 13 Shorts 82  
## 14 Socks 78  
## 15 Tires and Tubes 87  
## 16 Touring Bikes 72  
## 17 Vests 86

aggregate(sales$Customer.Age ~ sales$Sub.Category, data = sales, FUN = min)

## sales$Sub.Category sales$Customer.Age  
## 1 Bike Racks 17  
## 2 Bike Stands 18  
## 3 Bottles and Cages 17  
## 4 Caps 17  
## 5 Cleaners 17  
## 6 Fenders 17  
## 7 Gloves 17  
## 8 Helmets 17  
## 9 Hydration Packs 17  
## 10 Jerseys 17  
## 11 Mountain Bikes 17  
## 12 Road Bikes 17  
## 13 Shorts 17  
## 14 Socks 17  
## 15 Tires and Tubes 17  
## 16 Touring Bikes 17  
## 17 Vests 17

aggregate(sales$Quantity ~ sales$Sub.Category, data = sales, FUN = mean)

## sales$Sub.Category sales$Quantity  
## 1 Bike Racks 1.980583  
## 2 Bike Stands 2.096552  
## 3 Bottles and Cages 1.993955  
## 4 Caps 1.990771  
## 5 Cleaners 2.022018  
## 6 Fenders 1.960630  
## 7 Gloves 1.902083  
## 8 Helmets 2.008381  
## 9 Hydration Packs 1.984848  
## 10 Jerseys 2.016500  
## 11 Mountain Bikes 2.009134  
## 12 Road Bikes 2.024818  
## 13 Shorts 1.994700  
## 14 Socks 2.060440  
## 15 Tires and Tubes 1.999010  
## 16 Touring Bikes 2.003748  
## 17 Vests 2.038462

aggregate(sales$Quantity ~ sales$Sub.Category, data = sales, FUN = median)

## sales$Sub.Category sales$Quantity  
## 1 Bike Racks 2  
## 2 Bike Stands 2  
## 3 Bottles and Cages 2  
## 4 Caps 2  
## 5 Cleaners 2  
## 6 Fenders 2  
## 7 Gloves 2  
## 8 Helmets 2  
## 9 Hydration Packs 2  
## 10 Jerseys 2  
## 11 Mountain Bikes 2  
## 12 Road Bikes 2  
## 13 Shorts 2  
## 14 Socks 2  
## 15 Tires and Tubes 2  
## 16 Touring Bikes 2  
## 17 Vests 2

aggregate(sales$Quantity ~ sales$Sub.Category, data = sales, FUN = max)

## sales$Sub.Category sales$Quantity  
## 1 Bike Racks 3  
## 2 Bike Stands 3  
## 3 Bottles and Cages 3  
## 4 Caps 3  
## 5 Cleaners 3  
## 6 Fenders 3  
## 7 Gloves 3  
## 8 Helmets 3  
## 9 Hydration Packs 3  
## 10 Jerseys 3  
## 11 Mountain Bikes 3  
## 12 Road Bikes 3  
## 13 Shorts 3  
## 14 Socks 3  
## 15 Tires and Tubes 3  
## 16 Touring Bikes 3  
## 17 Vests 3

aggregate(sales$Quantity ~ sales$Sub.Category, data = sales, FUN = min)

## sales$Sub.Category sales$Quantity  
## 1 Bike Racks 1  
## 2 Bike Stands 1  
## 3 Bottles and Cages 1  
## 4 Caps 1  
## 5 Cleaners 1  
## 6 Fenders 1  
## 7 Gloves 1  
## 8 Helmets 1  
## 9 Hydration Packs 1  
## 10 Jerseys 1  
## 11 Mountain Bikes 1  
## 12 Road Bikes 1  
## 13 Shorts 1  
## 14 Socks 1  
## 15 Tires and Tubes 1  
## 16 Touring Bikes 1  
## 17 Vests 1

aggregate(sales$Unit.Cost ~ sales$Sub.Category, data = sales, FUN = mean)

## sales$Sub.Category sales$Unit.Cost  
## 1 Bike Racks 666.79612  
## 2 Bike Stands 511.35862  
## 3 Bottles and Cages 67.02736  
## 4 Caps 86.65689  
## 5 Cleaners 76.18905  
## 6 Fenders 216.06008  
## 7 Gloves 239.73892  
## 8 Helmets 320.99821  
## 9 Hydration Packs 513.40283  
## 10 Jerseys 462.68964  
## 11 Mountain Bikes 1116.13306  
## 12 Road Bikes 762.72682  
## 13 Shorts 658.48645  
## 14 Socks 82.45467  
## 15 Tires and Tubes 128.69896  
## 16 Touring Bikes 1049.05750  
## 17 Vests 600.34513

aggregate(sales$Unit.Cost ~ sales$Sub.Category, data = sales, FUN = median)

## sales$Sub.Category sales$Unit.Cost  
## 1 Bike Racks 360.00  
## 2 Bike Stands 397.50  
## 3 Bottles and Cages 50.00  
## 4 Caps 72.00  
## 5 Cleaners 63.67  
## 6 Fenders 176.00  
## 7 Gloves 208.00  
## 8 Helmets 262.50  
## 9 Hydration Packs 403.33  
## 10 Jerseys 378.00  
## 11 Mountain Bikes 773.33  
## 12 Road Bikes 560.00  
## 13 Shorts 525.00  
## 14 Socks 70.50  
## 15 Tires and Tubes 52.00  
## 16 Touring Bikes 794.67  
## 17 Vests 487.00

aggregate(sales$Unit.Cost ~ sales$Sub.Category, data = sales, FUN = max)

## sales$Sub.Category sales$Unit.Cost  
## 1 Bike Racks 3240  
## 2 Bike Stands 1590  
## 3 Bottles and Cages 300  
## 4 Caps 270  
## 5 Cleaners 239  
## 6 Fenders 659  
## 7 Gloves 735  
## 8 Helmets 1050  
## 9 Hydration Packs 1650  
## 10 Jerseys 1620  
## 11 Mountain Bikes 2320  
## 12 Road Bikes 2443  
## 13 Shorts 2100  
## 14 Socks 270  
## 15 Tires and Tubes 1050  
## 16 Touring Bikes 2384  
## 17 Vests 1905

aggregate(sales$Unit.Cost ~ sales$Sub.Category, data = sales, FUN = min)

## sales$Sub.Category sales$Unit.Cost  
## 1 Bike Racks 40.00  
## 2 Bike Stands 53.00  
## 3 Bottles and Cages 1.67  
## 4 Caps 3.00  
## 5 Cleaners 2.67  
## 6 Fenders 7.33  
## 7 Gloves 8.00  
## 8 Helmets 11.67  
## 9 Hydration Packs 18.33  
## 10 Jerseys 16.67  
## 11 Mountain Bikes 180.00  
## 12 Road Bikes 180.00  
## 13 Shorts 23.33  
## 14 Socks 3.00  
## 15 Tires and Tubes 0.67  
## 16 Touring Bikes 247.33  
## 17 Vests 21.33

aggregate(sales$Unit.Price ~ sales$Sub.Category, data = sales, FUN = mean)

## sales$Sub.Category sales$Unit.Price  
## 1 Bike Racks 885.83981  
## 2 Bike Stands 612.24368  
## 3 Bottles and Cages 81.95715  
## 4 Caps 104.15140  
## 5 Cleaners 92.45229  
## 6 Fenders 276.91689  
## 7 Gloves 299.73299  
## 8 Helmets 395.60065  
## 9 Hydration Packs 630.19024  
## 10 Jerseys 553.08033  
## 11 Mountain Bikes 1149.45707  
## 12 Road Bikes 781.17158  
## 13 Shorts 753.05860  
## 14 Socks 97.61493  
## 15 Tires and Tubes 156.60669  
## 16 Touring Bikes 1090.75962  
## 17 Vests 712.31571

aggregate(sales$Unit.Price ~ sales$Sub.Category, data = sales, FUN = median)

## sales$Sub.Category sales$Unit.Price  
## 1 Bike Racks 469.50000  
## 2 Bike Stands 467.00000  
## 3 Bottles and Cages 61.00000  
## 4 Caps 83.00000  
## 5 Cleaners 74.33333  
## 6 Fenders 225.33333  
## 7 Gloves 260.75000  
## 8 Helmets 322.00000  
## 9 Hydration Packs 486.08333  
## 10 Jerseys 440.83333  
## 11 Mountain Bikes 921.00000  
## 12 Road Bikes 592.50000  
## 13 Shorts 599.66667  
## 14 Socks 81.50000  
## 15 Tires and Tubes 64.00000  
## 16 Touring Bikes 859.33333  
## 17 Vests 581.25000

aggregate(sales$Unit.Price ~ sales$Sub.Category, data = sales, FUN = max)

## sales$Sub.Category sales$Unit.Price  
## 1 Bike Racks 5082  
## 2 Bike Stands 2144  
## 3 Bottles and Cages 465  
## 4 Caps 478  
## 5 Cleaners 351  
## 6 Fenders 1007  
## 7 Gloves 1084  
## 8 Helmets 1637  
## 9 Hydration Packs 2412  
## 10 Jerseys 2592  
## 11 Mountain Bikes 3487  
## 12 Road Bikes 3397  
## 13 Shorts 3339  
## 14 Socks 415  
## 15 Tires and Tubes 1640  
## 16 Touring Bikes 3495  
## 17 Vests 3103

aggregate(sales$Unit.Price ~ sales$Sub.Category, data = sales, FUN = min)

## sales$Sub.Category sales$Unit.Price  
## 1 Bike Racks 46.000000  
## 2 Bike Stands 56.000000  
## 3 Bottles and Cages 1.666667  
## 4 Caps 2.666667  
## 5 Cleaners 3.000000  
## 6 Fenders 6.333333  
## 7 Gloves 8.000000  
## 8 Helmets 9.333333  
## 9 Hydration Packs 16.000000  
## 10 Jerseys 16.333333  
## 11 Mountain Bikes 133.666667  
## 12 Road Bikes 125.000000  
## 13 Shorts 26.000000  
## 14 Socks 3.000000  
## 15 Tires and Tubes 0.666667  
## 16 Touring Bikes 186.000000  
## 17 Vests 22.333333

aggregate(sales$Cost ~ sales$Sub.Category, data = sales, FUN = mean)

## sales$Sub.Category sales$Cost  
## 1 Bike Racks 1028.7379  
## 2 Bike Stands 866.2759  
## 3 Bottles and Cages 109.5240  
## 4 Caps 140.0013  
## 5 Cleaners 125.1798  
## 6 Fenders 338.3215  
## 7 Gloves 380.2833  
## 8 Helmets 531.5457  
## 9 Hydration Packs 835.6944  
## 10 Jerseys 766.6170  
## 11 Mountain Bikes 1838.4468  
## 12 Road Bikes 1265.3286  
## 13 Shorts 1063.8516  
## 14 Socks 142.0220  
## 15 Tires and Tubes 211.8242  
## 16 Touring Bikes 1718.9670  
## 17 Vests 994.6699

aggregate(sales$Cost ~ sales$Sub.Category, data = sales, FUN = median)

## sales$Sub.Category sales$Cost  
## 1 Bike Racks 720  
## 2 Bike Stands 954  
## 3 Bottles and Cages 100  
## 4 Caps 144  
## 5 Cleaners 127  
## 6 Fenders 330  
## 7 Gloves 392  
## 8 Helmets 525  
## 9 Hydration Packs 825  
## 10 Jerseys 750  
## 11 Mountain Bikes 2295  
## 12 Road Bikes 1120  
## 13 Shorts 1050  
## 14 Socks 144  
## 15 Tires and Tubes 95  
## 16 Touring Bikes 2384  
## 17 Vests 1016

aggregate(sales$Cost ~ sales$Sub.Category, data = sales, FUN = max)

## sales$Sub.Category sales$Cost  
## 1 Bike Racks 3600  
## 2 Bike Stands 1590  
## 3 Bottles and Cages 300  
## 4 Caps 270  
## 5 Cleaners 239  
## 6 Fenders 659  
## 7 Gloves 735  
## 8 Helmets 1050  
## 9 Hydration Packs 1650  
## 10 Jerseys 1620  
## 11 Mountain Bikes 2320  
## 12 Road Bikes 2443  
## 13 Shorts 2100  
## 14 Socks 270  
## 15 Tires and Tubes 1050  
## 16 Touring Bikes 2384  
## 17 Vests 1905

aggregate(sales$Cost ~ sales$Sub.Category, data = sales, FUN = min)

## sales$Sub.Category sales$Cost  
## 1 Bike Racks 120  
## 2 Bike Stands 159  
## 3 Bottles and Cages 5  
## 4 Caps 9  
## 5 Cleaners 8  
## 6 Fenders 22  
## 7 Gloves 24  
## 8 Helmets 35  
## 9 Hydration Packs 55  
## 10 Jerseys 50  
## 11 Mountain Bikes 540  
## 12 Road Bikes 540  
## 13 Shorts 70  
## 14 Socks 9  
## 15 Tires and Tubes 2  
## 16 Touring Bikes 742  
## 17 Vests 64

aggregate(sales$Revenue ~ sales$Sub.Category, data = sales, FUN = mean)

## sales$Sub.Category sales$Revenue  
## 1 Bike Racks 1367.5146  
## 2 Bike Stands 1040.7655  
## 3 Bottles and Cages 133.9968  
## 4 Caps 168.7488  
## 5 Cleaners 152.0862  
## 6 Fenders 432.0262  
## 7 Gloves 475.7354  
## 8 Helmets 655.7016  
## 9 Hydration Packs 1018.3737  
## 10 Jerseys 917.0550  
## 11 Mountain Bikes 1891.2883  
## 12 Road Bikes 1297.8124  
## 13 Shorts 1217.6396  
## 14 Socks 168.2335  
## 15 Tires and Tubes 257.9117  
## 16 Touring Bikes 1790.0375  
## 17 Vests 1181.6699

aggregate(sales$Revenue ~ sales$Sub.Category, data = sales, FUN = median)

## sales$Sub.Category sales$Revenue  
## 1 Bike Racks 906.0  
## 2 Bike Stands 1041.0  
## 3 Bottles and Cages 119.0  
## 4 Caps 169.0  
## 5 Cleaners 155.0  
## 6 Fenders 429.0  
## 7 Gloves 467.5  
## 8 Helmets 639.0  
## 9 Hydration Packs 1011.0  
## 10 Jerseys 878.0  
## 11 Mountain Bikes 2137.0  
## 12 Road Bikes 1099.5  
## 13 Shorts 1178.0  
## 14 Socks 163.5  
## 15 Tires and Tubes 114.0  
## 16 Touring Bikes 2050.0  
## 17 Vests 1196.0

aggregate(sales$Revenue ~ sales$Sub.Category, data = sales, FUN = max)

## sales$Sub.Category sales$Revenue  
## 1 Bike Racks 5082  
## 2 Bike Stands 2153  
## 3 Bottles and Cages 465  
## 4 Caps 478  
## 5 Cleaners 367  
## 6 Fenders 1078  
## 7 Gloves 1256  
## 8 Helmets 1826  
## 9 Hydration Packs 2555  
## 10 Jerseys 2631  
## 11 Mountain Bikes 3487  
## 12 Road Bikes 3616  
## 13 Shorts 3339  
## 14 Socks 465  
## 15 Tires and Tubes 1641  
## 16 Touring Bikes 3681  
## 17 Vests 3179

aggregate(sales$Revenue ~ sales$Sub.Category, data = sales, FUN = min)

## sales$Sub.Category sales$Revenue  
## 1 Bike Racks 134  
## 2 Bike Stands 168  
## 3 Bottles and Cages 4  
## 4 Caps 8  
## 5 Cleaners 8  
## 6 Fenders 19  
## 7 Gloves 24  
## 8 Helmets 28  
## 9 Hydration Packs 48  
## 10 Jerseys 34  
## 11 Mountain Bikes 335  
## 12 Road Bikes 336  
## 13 Shorts 60  
## 14 Socks 9  
## 15 Tires and Tubes 2  
## 16 Touring Bikes 443  
## 17 Vests 57

aggregate(sales$Unit\_Margin ~ sales$Sub.Category, data = sales, FUN = mean)

## sales$Sub.Category sales$Unit\_Margin  
## 1 Bike Racks 219.04369  
## 2 Bike Stands 100.88506  
## 3 Bottles and Cages 14.92979  
## 4 Caps 17.49451  
## 5 Cleaners 16.26325  
## 6 Fenders 60.85681  
## 7 Gloves 59.99407  
## 8 Helmets 74.60244  
## 9 Hydration Packs 116.78741  
## 10 Jerseys 90.39069  
## 11 Mountain Bikes 33.32401  
## 12 Road Bikes 18.44476  
## 13 Shorts 94.57215  
## 14 Socks 15.16026  
## 15 Tires and Tubes 27.90773  
## 16 Touring Bikes 41.70212  
## 17 Vests 111.97058

aggregate(sales$Unit\_Margin ~ sales$Sub.Category, data = sales, FUN = median)

## sales$Sub.Category sales$Unit\_Margin  
## 1 Bike Racks 90.000000  
## 2 Bike Stands 62.666667  
## 3 Bottles and Cages 9.000000  
## 4 Caps 11.000000  
## 5 Cleaners 10.666667  
## 6 Fenders 41.500000  
## 7 Gloves 43.500000  
## 8 Helmets 47.500000  
## 9 Hydration Packs 71.001667  
## 10 Jerseys 50.000000  
## 11 Mountain Bikes 7.500000  
## 12 Road Bikes 2.583333  
## 13 Shorts 56.001666  
## 14 Socks 8.500000  
## 15 Tires and Tubes 9.330000  
## 16 Touring Bikes 20.333334  
## 17 Vests 68.585000

aggregate(sales$Unit\_Margin ~ sales$Sub.Category, data = sales, FUN = max)

## sales$Sub.Category sales$Unit\_Margin  
## 1 Bike Racks 1842  
## 2 Bike Stands 554  
## 3 Bottles and Cages 180  
## 4 Caps 208  
## 5 Cleaners 128  
## 6 Fenders 396  
## 7 Gloves 433  
## 8 Helmets 692  
## 9 Hydration Packs 1037  
## 10 Jerseys 1026  
## 11 Mountain Bikes 1167  
## 12 Road Bikes 954  
## 13 Shorts 1309  
## 14 Socks 145  
## 15 Tires and Tubes 590  
## 16 Touring Bikes 1111  
## 17 Vests 1261

aggregate(sales$Unit\_Margin ~ sales$Sub.Category, data = sales, FUN = min)

## sales$Sub.Category sales$Unit\_Margin  
## 1 Bike Racks -47.00000  
## 2 Bike Stands -70.00000  
## 3 Bottles and Cages -72.00000  
## 4 Caps -87.00000  
## 5 Cleaners -47.00000  
## 6 Fenders -32.66333  
## 7 Gloves -102.00000  
## 8 Helmets -208.00000  
## 9 Hydration Packs -68.99667  
## 10 Jerseys -337.00000  
## 11 Mountain Bikes -822.00000  
## 12 Road Bikes -932.00000  
## 13 Shorts -373.00000  
## 14 Socks -20.00000  
## 15 Tires and Tubes -190.00000  
## 16 Touring Bikes -937.00000  
## 17 Vests -426.00000

aggregate(sales$Unit\_Margin\_percent ~ sales$Sub.Category, data = sales, FUN = mean)

## sales$Sub.Category sales$Unit\_Margin\_percent  
## 1 Bike Racks 22.7316505  
## 2 Bike Stands 16.2435172  
## 3 Bottles and Cages 16.9778258  
## 4 Caps 15.4679038  
## 5 Cleaners 16.5853211  
## 6 Fenders 20.7420866  
## 7 Gloves 18.9316667  
## 8 Helmets 17.2872079  
## 9 Hydration Packs 16.3340404  
## 10 Jerseys 14.5194600  
## 11 Mountain Bikes 1.0430581  
## 12 Road Bikes 0.5469788  
## 13 Shorts 11.4514488  
## 14 Socks 13.6212363  
## 15 Tires and Tubes 16.6039462  
## 16 Touring Bikes 2.0993853  
## 17 Vests 14.4220513

aggregate(sales$Unit\_Margin\_percent ~ sales$Sub.Category, data = sales, FUN = median)

## sales$Sub.Category sales$Unit\_Margin\_percent  
## 1 Bike Racks 22.550  
## 2 Bike Stands 15.430  
## 3 Bottles and Cages 17.140  
## 4 Caps 16.810  
## 5 Cleaners 17.030  
## 6 Fenders 20.525  
## 7 Gloves 18.800  
## 8 Helmets 17.750  
## 9 Hydration Packs 16.395  
## 10 Jerseys 15.445  
## 11 Mountain Bikes 1.030  
## 12 Road Bikes 0.550  
## 13 Shorts 11.480  
## 14 Socks 14.290  
## 15 Tires and Tubes 16.670  
## 16 Touring Bikes 2.770  
## 17 Vests 14.585

aggregate(sales$Unit\_Margin\_percent ~ sales$Sub.Category, data = sales, FUN = max)

## sales$Sub.Category sales$Unit\_Margin\_percent  
## 1 Bike Racks 43.07  
## 2 Bike Stands 40.75  
## 3 Bottles and Cages 44.77  
## 4 Caps 46.00  
## 5 Cleaners 41.72  
## 6 Fenders 43.73  
## 7 Gloves 45.72  
## 8 Helmets 45.10  
## 9 Hydration Packs 42.99  
## 10 Jerseys 44.71  
## 11 Mountain Bikes 37.02  
## 12 Road Bikes 34.40  
## 13 Shorts 39.20  
## 14 Socks 46.11  
## 15 Tires and Tubes 49.75  
## 16 Touring Bikes 35.23  
## 17 Vests 40.64

aggregate(sales$Unit\_Margin\_percent ~ sales$Sub.Category, data = sales, FUN = min)

## sales$Sub.Category sales$Unit\_Margin\_percent  
## 1 Bike Racks -10.85  
## 2 Bike Stands -7.72  
## 3 Bottles and Cages -40.23  
## 4 Caps -50.00  
## 5 Cleaners -39.55  
## 6 Fenders -27.68  
## 7 Gloves -23.74  
## 8 Helmets -39.11  
## 9 Hydration Packs -30.76  
## 10 Jerseys -47.06  
## 11 Mountain Bikes -66.99  
## 12 Road Bikes -68.67  
## 13 Shorts -41.89  
## 14 Socks -35.48  
## 15 Tires and Tubes -40.66  
## 16 Touring Bikes -67.49  
## 17 Vests -34.78

aggregate(sales$Margin ~ sales$Sub.Category, data = sales, FUN = mean)

## sales$Sub.Category sales$Margin  
## 1 Bike Racks 338.77670  
## 2 Bike Stands 174.48966  
## 3 Bottles and Cages 24.47280  
## 4 Caps 28.74753  
## 5 Cleaners 26.90642  
## 6 Fenders 93.70472  
## 7 Gloves 95.45208  
## 8 Helmets 124.15589  
## 9 Hydration Packs 182.67929  
## 10 Jerseys 150.43800  
## 11 Mountain Bikes 52.84143  
## 12 Road Bikes 32.48379  
## 13 Shorts 153.78799  
## 14 Socks 26.21154  
## 15 Tires and Tubes 46.08747  
## 16 Touring Bikes 71.07046  
## 17 Vests 187.00000

aggregate(sales$Margin ~ sales$Sub.Category, data = sales, FUN = median)

## sales$Sub.Category sales$Margin  
## 1 Bike Racks 186.0  
## 2 Bike Stands 123.0  
## 3 Bottles and Cages 17.0  
## 4 Caps 21.0  
## 5 Cleaners 21.0  
## 6 Fenders 77.0  
## 7 Gloves 76.0  
## 8 Helmets 90.0  
## 9 Hydration Packs 138.5  
## 10 Jerseys 99.0  
## 11 Mountain Bikes 15.0  
## 12 Road Bikes 4.0  
## 13 Shorts 116.0  
## 14 Socks 17.0  
## 15 Tires and Tubes 17.0  
## 16 Touring Bikes 40.0  
## 17 Vests 134.0

aggregate(sales$Margin ~ sales$Sub.Category, data = sales, FUN = max)

## sales$Sub.Category sales$Margin  
## 1 Bike Racks 1842  
## 2 Bike Stands 722  
## 3 Bottles and Cages 192  
## 4 Caps 208  
## 5 Cleaners 128  
## 6 Fenders 441  
## 7 Gloves 521  
## 8 Helmets 776  
## 9 Hydration Packs 1037  
## 10 Jerseys 1026  
## 11 Mountain Bikes 1167  
## 12 Road Bikes 1173  
## 13 Shorts 1309  
## 14 Socks 195  
## 15 Tires and Tubes 657  
## 16 Touring Bikes 1297  
## 17 Vests 1274

aggregate(sales$Margin ~ sales$Sub.Category, data = sales, FUN = min)

## sales$Sub.Category sales$Margin  
## 1 Bike Racks -47  
## 2 Bike Stands -70  
## 3 Bottles and Cages -72  
## 4 Caps -87  
## 5 Cleaners -47  
## 6 Fenders -98  
## 7 Gloves -102  
## 8 Helmets -255  
## 9 Hydration Packs -207  
## 10 Jerseys -337  
## 11 Mountain Bikes -916  
## 12 Road Bikes -932  
## 13 Shorts -620  
## 14 Socks -33  
## 15 Tires and Tubes -214  
## 16 Touring Bikes -937  
## 17 Vests -426

aggregate(sales$Margin\_percent ~ sales$Sub.Category, data = sales, FUN = mean)

## sales$Sub.Category sales$Margin\_percent  
## 1 Bike Racks 22.7316505  
## 2 Bike Stands 16.2435172  
## 3 Bottles and Cages 16.9785342  
## 4 Caps 15.4678972  
## 5 Cleaners 16.5852844  
## 6 Fenders 20.7418504  
## 7 Gloves 18.9316667  
## 8 Helmets 17.2874066  
## 9 Hydration Packs 16.3341414  
## 10 Jerseys 14.5194700  
## 11 Mountain Bikes 1.0429448  
## 12 Road Bikes 0.5468961  
## 13 Shorts 11.4512191  
## 14 Socks 13.6212363  
## 15 Tires and Tubes 16.6050783  
## 16 Touring Bikes 2.0993103  
## 17 Vests 14.4218910

aggregate(sales$Margin\_percent ~ sales$Sub.Category, data = sales, FUN = median)

## sales$Sub.Category sales$Margin\_percent  
## 1 Bike Racks 22.550  
## 2 Bike Stands 15.430  
## 3 Bottles and Cages 17.140  
## 4 Caps 16.810  
## 5 Cleaners 17.030  
## 6 Fenders 20.525  
## 7 Gloves 18.800  
## 8 Helmets 17.750  
## 9 Hydration Packs 16.400  
## 10 Jerseys 15.445  
## 11 Mountain Bikes 1.030  
## 12 Road Bikes 0.550  
## 13 Shorts 11.480  
## 14 Socks 14.290  
## 15 Tires and Tubes 16.670  
## 16 Touring Bikes 2.770  
## 17 Vests 14.585

aggregate(sales$Margin\_percent ~ sales$Sub.Category, data = sales, FUN = max)

## sales$Sub.Category sales$Margin\_percent  
## 1 Bike Racks 43.07  
## 2 Bike Stands 40.75  
## 3 Bottles and Cages 44.77  
## 4 Caps 46.00  
## 5 Cleaners 41.72  
## 6 Fenders 43.73  
## 7 Gloves 45.72  
## 8 Helmets 45.10  
## 9 Hydration Packs 42.99  
## 10 Jerseys 44.71  
## 11 Mountain Bikes 37.02  
## 12 Road Bikes 34.40  
## 13 Shorts 39.20  
## 14 Socks 46.11  
## 15 Tires and Tubes 50.00  
## 16 Touring Bikes 35.23  
## 17 Vests 40.64

aggregate(sales$Margin\_percent ~ sales$Sub.Category, data = sales, FUN = min)

## sales$Sub.Category sales$Margin\_percent  
## 1 Bike Racks -10.85  
## 2 Bike Stands -7.72  
## 3 Bottles and Cages -40.24  
## 4 Caps -50.00  
## 5 Cleaners -39.56  
## 6 Fenders -27.68  
## 7 Gloves -23.74  
## 8 Helmets -39.11  
## 9 Hydration Packs -30.76  
## 10 Jerseys -47.06  
## 11 Mountain Bikes -66.99  
## 12 Road Bikes -68.67  
## 13 Shorts -41.89  
## 14 Socks -35.48  
## 15 Tires and Tubes -40.66  
## 16 Touring Bikes -67.49  
## 17 Vests -34.78

aggregate(sales$Quantity ~ sales$Year\_Month, data = sales, FUN = mean)

## sales$Year\_Month sales$Quantity  
## 1 2015-01-01 1.942029  
## 2 2015-02-01 1.922078  
## 3 2015-03-01 2.006711  
## 4 2015-04-01 1.949153  
## 5 2015-05-01 2.063725  
## 6 2015-06-01 1.995146  
## 7 2015-07-01 1.980603  
## 8 2015-08-01 2.020408  
## 9 2015-09-01 2.000000  
## 10 2015-10-01 2.017619  
## 11 2015-11-01 1.999598  
## 12 2015-12-01 1.990627  
## 13 2016-01-01 1.985554  
## 14 2016-02-01 2.033663  
## 15 2016-03-01 2.007984  
## 16 2016-04-01 2.009015  
## 17 2016-05-01 1.984447  
## 18 2016-06-01 1.995107  
## 19 2016-07-01 2.023310

aggregate(sales$Quantity ~ sales$Year\_Month, data = sales, FUN = median)

## sales$Year\_Month sales$Quantity  
## 1 2015-01-01 2  
## 2 2015-02-01 2  
## 3 2015-03-01 2  
## 4 2015-04-01 2  
## 5 2015-05-01 2  
## 6 2015-06-01 2  
## 7 2015-07-01 2  
## 8 2015-08-01 2  
## 9 2015-09-01 2  
## 10 2015-10-01 2  
## 11 2015-11-01 2  
## 12 2015-12-01 2  
## 13 2016-01-01 2  
## 14 2016-02-01 2  
## 15 2016-03-01 2  
## 16 2016-04-01 2  
## 17 2016-05-01 2  
## 18 2016-06-01 2  
## 19 2016-07-01 2

aggregate(sales$Quantity ~ sales$Year\_Month, data = sales, FUN = max)

## sales$Year\_Month sales$Quantity  
## 1 2015-01-01 3  
## 2 2015-02-01 3  
## 3 2015-03-01 3  
## 4 2015-04-01 3  
## 5 2015-05-01 3  
## 6 2015-06-01 3  
## 7 2015-07-01 3  
## 8 2015-08-01 3  
## 9 2015-09-01 3  
## 10 2015-10-01 3  
## 11 2015-11-01 3  
## 12 2015-12-01 3  
## 13 2016-01-01 3  
## 14 2016-02-01 3  
## 15 2016-03-01 3  
## 16 2016-04-01 3  
## 17 2016-05-01 3  
## 18 2016-06-01 3  
## 19 2016-07-01 3

aggregate(sales$Quantity ~ sales$Year\_Month, data = sales, FUN = min)

## sales$Year\_Month sales$Quantity  
## 1 2015-01-01 1  
## 2 2015-02-01 1  
## 3 2015-03-01 1  
## 4 2015-04-01 1  
## 5 2015-05-01 1  
## 6 2015-06-01 1  
## 7 2015-07-01 1  
## 8 2015-08-01 1  
## 9 2015-09-01 1  
## 10 2015-10-01 1  
## 11 2015-11-01 1  
## 12 2015-12-01 1  
## 13 2016-01-01 1  
## 14 2016-02-01 1  
## 15 2016-03-01 1  
## 16 2016-04-01 1  
## 17 2016-05-01 1  
## 18 2016-06-01 1  
## 19 2016-07-01 1

aggregate(sales$Unit.Cost ~ sales$Year\_Month, data = sales, FUN = mean)

## sales$Year\_Month sales$Unit.Cost  
## 1 2015-01-01 1051.3594  
## 2 2015-02-01 1077.3684  
## 3 2015-03-01 1058.6615  
## 4 2015-04-01 1040.9053  
## 5 2015-05-01 963.6790  
## 6 2015-06-01 1033.2695  
## 7 2015-07-01 521.9190  
## 8 2015-08-01 312.2016  
## 9 2015-09-01 324.8164  
## 10 2015-10-01 303.5263  
## 11 2015-11-01 339.0641  
## 12 2015-12-01 363.7626  
## 13 2016-01-01 318.9688  
## 14 2016-02-01 322.9112  
## 15 2016-03-01 319.3851  
## 16 2016-04-01 310.1116  
## 17 2016-05-01 345.8606  
## 18 2016-06-01 345.7589  
## 19 2016-07-01 181.3588

aggregate(sales$Unit.Cost ~ sales$Year\_Month, data = sales, FUN = median)

## sales$Year\_Month sales$Unit.Cost  
## 1 2015-01-01 1000.000  
## 2 2015-02-01 1000.000  
## 3 2015-03-01 814.330  
## 4 2015-04-01 783.000  
## 5 2015-05-01 727.330  
## 6 2015-06-01 783.000  
## 7 2015-07-01 262.500  
## 8 2015-08-01 130.000  
## 9 2015-09-01 140.000  
## 10 2015-10-01 126.000  
## 11 2015-11-01 140.000  
## 12 2015-12-01 163.000  
## 13 2016-01-01 133.330  
## 14 2016-02-01 140.000  
## 15 2016-03-01 135.665  
## 16 2016-04-01 139.665  
## 17 2016-05-01 150.000  
## 18 2016-06-01 150.000  
## 19 2016-07-01 81.670

aggregate(sales$Unit.Cost ~ sales$Year\_Month, data = sales, FUN = max)

## sales$Year\_Month sales$Unit.Cost  
## 1 2015-01-01 2443  
## 2 2015-02-01 2443  
## 3 2015-03-01 2443  
## 4 2015-04-01 2443  
## 5 2015-05-01 2443  
## 6 2015-06-01 2443  
## 7 2015-07-01 2443  
## 8 2015-08-01 2443  
## 9 2015-09-01 2443  
## 10 2015-10-01 2443  
## 11 2015-11-01 2443  
## 12 2015-12-01 2443  
## 13 2016-01-01 3120  
## 14 2016-02-01 2443  
## 15 2016-03-01 2443  
## 16 2016-04-01 2760  
## 17 2016-05-01 3120  
## 18 2016-06-01 3000  
## 19 2016-07-01 3240

aggregate(sales$Unit.Cost ~ sales$Year\_Month, data = sales, FUN = min)

## sales$Year\_Month sales$Unit.Cost  
## 1 2015-01-01 261.00  
## 2 2015-02-01 261.00  
## 3 2015-03-01 261.00  
## 4 2015-04-01 261.00  
## 5 2015-05-01 261.00  
## 6 2015-06-01 261.00  
## 7 2015-07-01 1.67  
## 8 2015-08-01 0.67  
## 9 2015-09-01 1.00  
## 10 2015-10-01 0.67  
## 11 2015-11-01 0.67  
## 12 2015-12-01 1.00  
## 13 2016-01-01 0.67  
## 14 2016-02-01 0.67  
## 15 2016-03-01 0.67  
## 16 2016-04-01 0.67  
## 17 2016-05-01 0.67  
## 18 2016-06-01 1.00  
## 19 2016-07-01 0.67

aggregate(sales$Unit.Price ~ sales$Year\_Month, data = sales, FUN = mean)

## sales$Year\_Month sales$Unit.Price  
## 1 2015-01-01 1031.8925  
## 2 2015-02-01 1060.8485  
## 3 2015-03-01 1031.9642  
## 4 2015-04-01 1022.9972  
## 5 2015-05-01 944.2492  
## 6 2015-06-01 1019.4256  
## 7 2015-07-01 531.0392  
## 8 2015-08-01 328.8168  
## 9 2015-09-01 339.7708  
## 10 2015-10-01 319.4170  
## 11 2015-11-01 351.9415  
## 12 2015-12-01 376.0639  
## 13 2016-01-01 380.4615  
## 14 2016-02-01 380.1641  
## 15 2016-03-01 380.6658  
## 16 2016-04-01 369.3159  
## 17 2016-05-01 408.9048  
## 18 2016-06-01 407.5579  
## 19 2016-07-01 233.4349

aggregate(sales$Unit.Price ~ sales$Year\_Month, data = sales, FUN = median)

## sales$Year\_Month sales$Unit.Price  
## 1 2015-01-01 896.0000  
## 2 2015-02-01 908.0000  
## 3 2015-03-01 848.0000  
## 4 2015-04-01 826.3333  
## 5 2015-05-01 745.6667  
## 6 2015-06-01 779.8333  
## 7 2015-07-01 284.6667  
## 8 2015-08-01 142.3333  
## 9 2015-09-01 158.0000  
## 10 2015-10-01 145.0833  
## 11 2015-11-01 157.0000  
## 12 2015-12-01 175.0000  
## 13 2016-01-01 168.0000  
## 14 2016-02-01 178.0000  
## 15 2016-03-01 171.0000  
## 16 2016-04-01 173.6667  
## 17 2016-05-01 189.0000  
## 18 2016-06-01 189.0000  
## 19 2016-07-01 100.0000

aggregate(sales$Unit.Price ~ sales$Year\_Month, data = sales, FUN = max)

## sales$Year\_Month sales$Unit.Price  
## 1 2015-01-01 2683  
## 2 2015-02-01 2805  
## 3 2015-03-01 2808  
## 4 2015-04-01 2932  
## 5 2015-05-01 3009  
## 6 2015-06-01 3224  
## 7 2015-07-01 3095  
## 8 2015-08-01 3215  
## 9 2015-09-01 3014  
## 10 2015-10-01 3068  
## 11 2015-11-01 2984  
## 12 2015-12-01 3182  
## 13 2016-01-01 4113  
## 14 2016-02-01 3397  
## 15 2016-03-01 3256  
## 16 2016-04-01 3463  
## 17 2016-05-01 3887  
## 18 2016-06-01 4008  
## 19 2016-07-01 5082

aggregate(sales$Unit.Price ~ sales$Year\_Month, data = sales, FUN = min)

## sales$Year\_Month sales$Unit.Price  
## 1 2015-01-01 223.000000  
## 2 2015-02-01 174.666667  
## 3 2015-03-01 216.000000  
## 4 2015-04-01 219.000000  
## 5 2015-05-01 168.333333  
## 6 2015-06-01 169.000000  
## 7 2015-07-01 1.666667  
## 8 2015-08-01 0.666667  
## 9 2015-09-01 1.500000  
## 10 2015-10-01 0.666667  
## 11 2015-11-01 1.000000  
## 12 2015-12-01 1.000000  
## 13 2016-01-01 1.000000  
## 14 2016-02-01 0.666667  
## 15 2016-03-01 1.000000  
## 16 2016-04-01 1.000000  
## 17 2016-05-01 1.000000  
## 18 2016-06-01 1.500000  
## 19 2016-07-01 1.000000

aggregate(sales$Cost ~ sales$Year\_Month, data = sales, FUN = mean)

## sales$Year\_Month sales$Cost  
## 1 2015-01-01 1712.5217  
## 2 2015-02-01 1713.8766  
## 3 2015-03-01 1735.0470  
## 4 2015-04-01 1633.2712  
## 5 2015-05-01 1609.9559  
## 6 2015-06-01 1681.7816  
## 7 2015-07-01 833.9978  
## 8 2015-08-01 513.3474  
## 9 2015-09-01 535.4208  
## 10 2015-10-01 514.4941  
## 11 2015-11-01 556.4498  
## 12 2015-12-01 600.0729  
## 13 2016-01-01 519.9957  
## 14 2016-02-01 538.1398  
## 15 2016-03-01 527.3456  
## 16 2016-04-01 517.9012  
## 17 2016-05-01 561.1397  
## 18 2016-06-01 571.8710  
## 19 2016-07-01 298.7257

aggregate(sales$Cost ~ sales$Year\_Month, data = sales, FUN = median)

## sales$Year\_Month sales$Cost  
## 1 2015-01-01 2049  
## 2 2015-02-01 2049  
## 3 2015-03-01 2049  
## 4 2015-04-01 2049  
## 5 2015-05-01 2049  
## 6 2015-06-01 2049  
## 7 2015-07-01 540  
## 8 2015-08-01 220  
## 9 2015-09-01 250  
## 10 2015-10-01 207  
## 11 2015-11-01 250  
## 12 2015-12-01 261  
## 13 2016-01-01 234  
## 14 2016-02-01 245  
## 15 2016-03-01 230  
## 16 2016-04-01 245  
## 17 2016-05-01 252  
## 18 2016-06-01 261  
## 19 2016-07-01 140

aggregate(sales$Cost ~ sales$Year\_Month, data = sales, FUN = max)

## sales$Year\_Month sales$Cost  
## 1 2015-01-01 2443  
## 2 2015-02-01 2443  
## 3 2015-03-01 2443  
## 4 2015-04-01 2443  
## 5 2015-05-01 2443  
## 6 2015-06-01 2443  
## 7 2015-07-01 2443  
## 8 2015-08-01 2443  
## 9 2015-09-01 2443  
## 10 2015-10-01 2443  
## 11 2015-11-01 2443  
## 12 2015-12-01 2443  
## 13 2016-01-01 3600  
## 14 2016-02-01 3600  
## 15 2016-03-01 2443  
## 16 2016-04-01 2760  
## 17 2016-05-01 3120  
## 18 2016-06-01 3240  
## 19 2016-07-01 3240

aggregate(sales$Cost ~ sales$Year\_Month, data = sales, FUN = min)

## sales$Year\_Month sales$Cost  
## 1 2015-01-01 783  
## 2 2015-02-01 783  
## 3 2015-03-01 783  
## 4 2015-04-01 783  
## 5 2015-05-01 783  
## 6 2015-06-01 783  
## 7 2015-07-01 2  
## 8 2015-08-01 2  
## 9 2015-09-01 2  
## 10 2015-10-01 2  
## 11 2015-11-01 2  
## 12 2015-12-01 2  
## 13 2016-01-01 2  
## 14 2016-02-01 2  
## 15 2016-03-01 2  
## 16 2016-04-01 2  
## 17 2016-05-01 2  
## 18 2016-06-01 2  
## 19 2016-07-01 2

aggregate(sales$Revenue ~ sales$Year\_Month, data = sales, FUN = mean)

## sales$Year\_Month sales$Revenue  
## 1 2015-01-01 1670.6449  
## 2 2015-02-01 1687.3831  
## 3 2015-03-01 1680.2550  
## 4 2015-04-01 1605.3277  
## 5 2015-05-01 1571.7108  
## 6 2015-06-01 1639.5922  
## 7 2015-07-01 850.2737  
## 8 2015-08-01 541.9700  
## 9 2015-09-01 562.4168  
## 10 2015-10-01 539.1421  
## 11 2015-11-01 578.3473  
## 12 2015-12-01 619.8292  
## 13 2016-01-01 621.1889  
## 14 2016-02-01 634.6052  
## 15 2016-03-01 627.0719  
## 16 2016-04-01 616.9823  
## 17 2016-05-01 663.9375  
## 18 2016-06-01 674.7925  
## 19 2016-07-01 381.9829

aggregate(sales$Revenue ~ sales$Year\_Month, data = sales, FUN = median)

## sales$Year\_Month sales$Revenue  
## 1 2015-01-01 1886.0  
## 2 2015-02-01 1906.0  
## 3 2015-03-01 1883.0  
## 4 2015-04-01 1790.0  
## 5 2015-05-01 1799.5  
## 6 2015-06-01 1869.0  
## 7 2015-07-01 545.0  
## 8 2015-08-01 245.0  
## 9 2015-09-01 273.0  
## 10 2015-10-01 231.5  
## 11 2015-11-01 272.0  
## 12 2015-12-01 304.0  
## 13 2016-01-01 298.0  
## 14 2016-02-01 309.0  
## 15 2016-03-01 296.0  
## 16 2016-04-01 313.0  
## 17 2016-05-01 331.0  
## 18 2016-06-01 330.0  
## 19 2016-07-01 181.0

aggregate(sales$Revenue ~ sales$Year\_Month, data = sales, FUN = max)

## sales$Year\_Month sales$Revenue  
## 1 2015-01-01 3173  
## 2 2015-02-01 2879  
## 3 2015-03-01 2820  
## 4 2015-04-01 2932  
## 5 2015-05-01 3009  
## 6 2015-06-01 3224  
## 7 2015-07-01 3095  
## 8 2015-08-01 3215  
## 9 2015-09-01 3014  
## 10 2015-10-01 3068  
## 11 2015-11-01 3089  
## 12 2015-12-01 3182  
## 13 2016-01-01 4266  
## 14 2016-02-01 4923  
## 15 2016-03-01 3681  
## 16 2016-04-01 3537  
## 17 2016-05-01 3887  
## 18 2016-06-01 4219  
## 19 2016-07-01 5082

aggregate(sales$Revenue ~ sales$Year\_Month, data = sales, FUN = min)

## sales$Year\_Month sales$Revenue  
## 1 2015-01-01 669  
## 2 2015-02-01 524  
## 3 2015-03-01 581  
## 4 2015-04-01 636  
## 5 2015-05-01 505  
## 6 2015-06-01 507  
## 7 2015-07-01 3  
## 8 2015-08-01 2  
## 9 2015-09-01 2  
## 10 2015-10-01 2  
## 11 2015-11-01 2  
## 12 2015-12-01 2  
## 13 2016-01-01 3  
## 14 2016-02-01 2  
## 15 2016-03-01 3  
## 16 2016-04-01 2  
## 17 2016-05-01 2  
## 18 2016-06-01 3  
## 19 2016-07-01 3

aggregate(sales$Unit\_Margin ~ sales$Year\_Month, data = sales, FUN = mean)

## sales$Year\_Month sales$Unit\_Margin  
## 1 2015-01-01 -19.466908  
## 2 2015-02-01 -16.519957  
## 3 2015-03-01 -26.697271  
## 4 2015-04-01 -17.908079  
## 5 2015-05-01 -19.429788  
## 6 2015-06-01 -13.843900  
## 7 2015-07-01 9.120165  
## 8 2015-08-01 16.615119  
## 9 2015-09-01 14.954445  
## 10 2015-10-01 15.890685  
## 11 2015-11-01 12.877381  
## 12 2015-12-01 12.301297  
## 13 2016-01-01 61.492717  
## 14 2016-02-01 57.252872  
## 15 2016-03-01 61.280729  
## 16 2016-04-01 59.204367  
## 17 2016-05-01 63.044183  
## 18 2016-06-01 61.798980  
## 19 2016-07-01 52.076097

aggregate(sales$Unit\_Margin ~ sales$Year\_Month, data = sales, FUN = median)

## sales$Year\_Month sales$Unit\_Margin  
## 1 2015-01-01 -44.416667  
## 2 2015-02-01 -22.000000  
## 3 2015-03-01 -29.000000  
## 4 2015-04-01 -31.000000  
## 5 2015-05-01 -34.500000  
## 6 2015-06-01 -23.250000  
## 7 2015-07-01 6.166666  
## 8 2015-08-01 6.000000  
## 9 2015-09-01 6.000000  
## 10 2015-10-01 6.000000  
## 11 2015-11-01 6.000000  
## 12 2015-12-01 6.003333  
## 13 2016-01-01 24.000000  
## 14 2016-02-01 23.000000  
## 15 2016-03-01 24.416666  
## 16 2016-04-01 23.583333  
## 17 2016-05-01 26.500000  
## 18 2016-06-01 24.833334  
## 19 2016-07-01 19.500000

aggregate(sales$Unit\_Margin ~ sales$Year\_Month, data = sales, FUN = max)

## sales$Year\_Month sales$Unit\_Margin  
## 1 2015-01-01 522  
## 2 2015-02-01 576  
## 3 2015-03-01 452  
## 4 2015-04-01 750  
## 5 2015-05-01 568  
## 6 2015-06-01 781  
## 7 2015-07-01 652  
## 8 2015-08-01 831  
## 9 2015-09-01 630  
## 10 2015-10-01 684  
## 11 2015-11-01 746  
## 12 2015-12-01 739  
## 13 2016-01-01 1058  
## 14 2016-02-01 1037  
## 15 2016-03-01 1261  
## 16 2016-04-01 1079  
## 17 2016-05-01 1128  
## 18 2016-06-01 1368  
## 19 2016-07-01 1842

aggregate(sales$Unit\_Margin ~ sales$Year\_Month, data = sales, FUN = min)

## sales$Year\_Month sales$Unit\_Margin  
## 1 2015-01-01 -395.5  
## 2 2015-02-01 -822.0  
## 3 2015-03-01 -805.0  
## 4 2015-04-01 -769.0  
## 5 2015-05-01 -599.0  
## 6 2015-06-01 -741.0  
## 7 2015-07-01 -932.0  
## 8 2015-08-01 -937.0  
## 9 2015-09-01 -746.0  
## 10 2015-10-01 -472.0  
## 11 2015-11-01 -920.0  
## 12 2015-12-01 -726.0  
## 13 2016-01-01 -749.0  
## 14 2016-02-01 -688.0  
## 15 2016-03-01 -733.0  
## 16 2016-04-01 -620.0  
## 17 2016-05-01 -654.0  
## 18 2016-06-01 -777.0  
## 19 2016-07-01 -82.0

aggregate(sales$Unit\_Margin\_percent ~ sales$Year\_Month, data = sales, FUN = mean)

## sales$Year\_Month sales$Unit\_Margin\_percent  
## 1 2015-01-01 -4.213841  
## 2 2015-02-01 -3.822727  
## 3 2015-03-01 -5.395369  
## 4 2015-04-01 -3.950000  
## 5 2015-05-01 -4.056471  
## 6 2015-06-01 -5.129369  
## 7 2015-07-01 5.609256  
## 8 2015-08-01 7.969705  
## 9 2015-09-01 7.414972  
## 10 2015-10-01 8.017134  
## 11 2015-11-01 7.236029  
## 12 2015-12-01 7.077616  
## 13 2016-01-01 18.892712  
## 14 2016-02-01 18.208888  
## 15 2016-03-01 18.376427  
## 16 2016-04-01 18.586713  
## 17 2016-05-01 18.482676  
## 18 2016-06-01 18.163267  
## 19 2016-07-01 20.499270

aggregate(sales$Unit\_Margin\_percent ~ sales$Year\_Month, data = sales, FUN = median)

## sales$Year\_Month sales$Unit\_Margin\_percent  
## 1 2015-01-01 -5.735  
## 2 2015-02-01 -3.855  
## 3 2015-03-01 -4.680  
## 4 2015-04-01 -5.000  
## 5 2015-05-01 -4.690  
## 6 2015-06-01 -4.330  
## 7 2015-07-01 6.970  
## 8 2015-08-01 9.000  
## 9 2015-09-01 8.890  
## 10 2015-10-01 8.885  
## 11 2015-11-01 8.545  
## 12 2015-12-01 8.570  
## 13 2016-01-01 19.640  
## 14 2016-02-01 19.070  
## 15 2016-03-01 19.420  
## 16 2016-04-01 19.540  
## 17 2016-05-01 19.230  
## 18 2016-06-01 19.240  
## 19 2016-07-01 20.000

aggregate(sales$Unit\_Margin\_percent ~ sales$Year\_Month, data = sales, FUN = max)

## sales$Year\_Month sales$Unit\_Margin\_percent  
## 1 2015-01-01 23.01  
## 2 2015-02-01 26.09  
## 3 2015-03-01 22.62  
## 4 2015-04-01 25.58  
## 5 2015-05-01 22.46  
## 6 2015-06-01 24.33  
## 7 2015-07-01 35.32  
## 8 2015-08-01 39.96  
## 9 2015-09-01 35.48  
## 10 2015-10-01 35.78  
## 11 2015-11-01 35.88  
## 12 2015-12-01 37.50  
## 13 2016-01-01 43.73  
## 14 2016-02-01 43.81  
## 15 2016-03-01 44.00  
## 16 2016-04-01 46.00  
## 17 2016-05-01 49.75  
## 18 2016-06-01 46.11  
## 19 2016-07-01 45.10

aggregate(sales$Unit\_Margin\_percent ~ sales$Year\_Month, data = sales, FUN = min)

## sales$Year\_Month sales$Unit\_Margin\_percent  
## 1 2015-01-01 -61.80  
## 2 2015-02-01 -66.99  
## 3 2015-03-01 -60.09  
## 4 2015-04-01 -56.42  
## 5 2015-05-01 -55.05  
## 6 2015-06-01 -54.44  
## 7 2015-07-01 -61.68  
## 8 2015-08-01 -64.75  
## 9 2015-09-01 -68.67  
## 10 2015-10-01 -56.63  
## 11 2015-11-01 -65.24  
## 12 2015-12-01 -60.71  
## 13 2016-01-01 -45.81  
## 14 2016-02-01 -46.74  
## 15 2016-03-01 -52.26  
## 16 2016-04-01 -42.82  
## 17 2016-05-01 -44.77  
## 18 2016-06-01 -50.36  
## 19 2016-07-01 -22.90

aggregate(sales$Margin ~ sales$Year\_Month, data = sales, FUN = mean)

## sales$Year\_Month sales$Margin  
## 1 2015-01-01 -41.87681  
## 2 2015-02-01 -26.49351  
## 3 2015-03-01 -54.79195  
## 4 2015-04-01 -27.94350  
## 5 2015-05-01 -38.24510  
## 6 2015-06-01 -42.18932  
## 7 2015-07-01 16.27586  
## 8 2015-08-01 28.62267  
## 9 2015-09-01 26.99609  
## 10 2015-10-01 24.64800  
## 11 2015-11-01 21.89751  
## 12 2015-12-01 19.75630  
## 13 2016-01-01 101.19321  
## 14 2016-02-01 96.46542  
## 15 2016-03-01 99.72621  
## 16 2016-04-01 99.08113  
## 17 2016-05-01 102.79781  
## 18 2016-06-01 102.92142  
## 19 2016-07-01 83.25719

aggregate(sales$Margin ~ sales$Year\_Month, data = sales, FUN = median)

## sales$Year\_Month sales$Margin  
## 1 2015-01-01 -70.5  
## 2 2015-02-01 -53.0  
## 3 2015-03-01 -64.0  
## 4 2015-04-01 -59.0  
## 5 2015-05-01 -69.0  
## 6 2015-06-01 -64.0  
## 7 2015-07-01 12.0  
## 8 2015-08-01 12.0  
## 9 2015-09-01 11.0  
## 10 2015-10-01 12.0  
## 11 2015-11-01 11.0  
## 12 2015-12-01 12.0  
## 13 2016-01-01 44.0  
## 14 2016-02-01 43.0  
## 15 2016-03-01 43.0  
## 16 2016-04-01 43.5  
## 17 2016-05-01 47.0  
## 18 2016-06-01 46.0  
## 19 2016-07-01 34.0

aggregate(sales$Margin ~ sales$Year\_Month, data = sales, FUN = max)

## sales$Year\_Month sales$Margin  
## 1 2015-01-01 730  
## 2 2015-02-01 576  
## 3 2015-03-01 638  
## 4 2015-04-01 750  
## 5 2015-05-01 632  
## 6 2015-06-01 781  
## 7 2015-07-01 652  
## 8 2015-08-01 831  
## 9 2015-09-01 666  
## 10 2015-10-01 684  
## 11 2015-11-01 746  
## 12 2015-12-01 739  
## 13 2016-01-01 1058  
## 14 2016-02-01 1323  
## 15 2016-03-01 1297  
## 16 2016-04-01 1153  
## 17 2016-05-01 1240  
## 18 2016-06-01 1459  
## 19 2016-07-01 1842

aggregate(sales$Margin ~ sales$Year\_Month, data = sales, FUN = min)

## sales$Year\_Month sales$Margin  
## 1 2015-01-01 -829  
## 2 2015-02-01 -822  
## 3 2015-03-01 -819  
## 4 2015-04-01 -787  
## 5 2015-05-01 -772  
## 6 2015-06-01 -807  
## 7 2015-07-01 -932  
## 8 2015-08-01 -937  
## 9 2015-09-01 -746  
## 10 2015-10-01 -750  
## 11 2015-11-01 -920  
## 12 2015-12-01 -897  
## 13 2016-01-01 -749  
## 14 2016-02-01 -688  
## 15 2016-03-01 -733  
## 16 2016-04-01 -620  
## 17 2016-05-01 -654  
## 18 2016-06-01 -777  
## 19 2016-07-01 -82

aggregate(sales$Margin\_percent ~ sales$Year\_Month, data = sales, FUN = mean)

## sales$Year\_Month sales$Margin\_percent  
## 1 2015-01-01 -4.213986  
## 2 2015-02-01 -3.822727  
## 3 2015-03-01 -5.395369  
## 4 2015-04-01 -3.950000  
## 5 2015-05-01 -4.056569  
## 6 2015-06-01 -5.129515  
## 7 2015-07-01 5.609687  
## 8 2015-08-01 7.970682  
## 9 2015-09-01 7.415436  
## 10 2015-10-01 8.018645  
## 11 2015-11-01 7.236162  
## 12 2015-12-01 7.077791  
## 13 2016-01-01 18.892871  
## 14 2016-02-01 18.209528  
## 15 2016-03-01 18.376896  
## 16 2016-04-01 18.587193  
## 17 2016-05-01 18.482995  
## 18 2016-06-01 18.163356  
## 19 2016-07-01 20.500218

aggregate(sales$Margin\_percent ~ sales$Year\_Month, data = sales, FUN = median)

## sales$Year\_Month sales$Margin\_percent  
## 1 2015-01-01 -5.735  
## 2 2015-02-01 -3.855  
## 3 2015-03-01 -4.680  
## 4 2015-04-01 -5.000  
## 5 2015-05-01 -4.690  
## 6 2015-06-01 -4.330  
## 7 2015-07-01 6.970  
## 8 2015-08-01 9.000  
## 9 2015-09-01 8.890  
## 10 2015-10-01 8.885  
## 11 2015-11-01 8.560  
## 12 2015-12-01 8.570  
## 13 2016-01-01 19.640  
## 14 2016-02-01 19.070  
## 15 2016-03-01 19.420  
## 16 2016-04-01 19.540  
## 17 2016-05-01 19.230  
## 18 2016-06-01 19.240  
## 19 2016-07-01 20.000

aggregate(sales$Margin\_percent ~ sales$Year\_Month, data = sales, FUN = max)

## sales$Year\_Month sales$Margin\_percent  
## 1 2015-01-01 23.01  
## 2 2015-02-01 26.09  
## 3 2015-03-01 22.62  
## 4 2015-04-01 25.58  
## 5 2015-05-01 22.46  
## 6 2015-06-01 24.33  
## 7 2015-07-01 35.32  
## 8 2015-08-01 39.96  
## 9 2015-09-01 35.48  
## 10 2015-10-01 35.78  
## 11 2015-11-01 35.90  
## 12 2015-12-01 37.50  
## 13 2016-01-01 43.73  
## 14 2016-02-01 43.82  
## 15 2016-03-01 44.00  
## 16 2016-04-01 46.00  
## 17 2016-05-01 50.00  
## 18 2016-06-01 46.11  
## 19 2016-07-01 45.10

aggregate(sales$Margin\_percent ~ sales$Year\_Month, data = sales, FUN = min)

## sales$Year\_Month sales$Margin\_percent  
## 1 2015-01-01 -61.80  
## 2 2015-02-01 -66.99  
## 3 2015-03-01 -60.09  
## 4 2015-04-01 -56.42  
## 5 2015-05-01 -55.05  
## 6 2015-06-01 -54.44  
## 7 2015-07-01 -61.68  
## 8 2015-08-01 -64.75  
## 9 2015-09-01 -68.67  
## 10 2015-10-01 -56.63  
## 11 2015-11-01 -65.24  
## 12 2015-12-01 -60.71  
## 13 2016-01-01 -45.81  
## 14 2016-02-01 -46.74  
## 15 2016-03-01 -52.26  
## 16 2016-04-01 -42.82  
## 17 2016-05-01 -44.77  
## 18 2016-06-01 -50.36  
## 19 2016-07-01 -22.90

## SUMMARY STATISTICS.

skimr::skim\_without\_charts(sales)

Data summary

|  |  |
| --- | --- |
| Name | sales |
| Number of rows | 34865 |
| Number of columns | 17 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| character | 5 |
| Date | 2 |
| numeric | 10 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

**Variable type: character**

| skim\_variable | n\_missing | complete\_rate | min | max | empty | n\_unique | whitespace |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Customer.Gender | 0 | 1 | 1 | 1 | 0 | 2 | 0 |
| Country | 0 | 1 | 6 | 14 | 0 | 4 | 0 |
| State | 0 | 1 | 4 | 19 | 0 | 45 | 0 |
| Product.Category | 0 | 1 | 5 | 11 | 0 | 3 | 0 |
| Sub.Category | 0 | 1 | 4 | 17 | 0 | 17 | 0 |

**Variable type: Date**

| skim\_variable | n\_missing | complete\_rate | min | max | median | n\_unique |
| --- | --- | --- | --- | --- | --- | --- |
| Date | 0 | 1 | 2015-01-01 | 2016-07-31 | 2016-01-28 | 576 |
| Year\_Month | 0 | 1 | 2015-01-01 | 2016-07-01 | 2016-01-01 | 19 |

**Variable type: numeric**

| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Customer.Age | 0 | 1 | 36.38 | 11.11 | 17.00 | 28.00 | 35.00 | 44.00 | 87.00 |
| Quantity | 0 | 1 | 2.00 | 0.81 | 1.00 | 1.00 | 2.00 | 3.00 | 3.00 |
| Unit.Cost | 0 | 1 | 349.89 | 490.02 | 0.67 | 45.00 | 150.00 | 455.00 | 3240.00 |
| Unit.Price | 0 | 1 | 389.24 | 525.32 | 0.67 | 53.67 | 179.00 | 521.00 | 5082.00 |
| Cost | 0 | 1 | 576.02 | 690.50 | 2.00 | 85.00 | 261.00 | 769.00 | 3600.00 |
| Revenue | 0 | 1 | 640.89 | 736.65 | 2.00 | 102.00 | 319.00 | 902.00 | 5082.00 |
| Unit\_Margin | 0 | 1 | 39.35 | 104.94 | -937.00 | 3.00 | 14.50 | 53.00 | 1842.00 |
| Unit\_Margin\_percent | 0 | 1 | 13.41 | 13.54 | -68.67 | 6.17 | 14.79 | 22.57 | 49.75 |
| Margin | 0 | 1 | 64.87 | 152.88 | -937.00 | 5.00 | 27.00 | 96.00 | 1842.00 |
| Margin\_percent | 0 | 1 | 13.41 | 13.54 | -68.67 | 6.17 | 14.80 | 22.57 | 50.00 |

## Comparing different columns of our dataframe with Year\_Month.

aggregate(sales$Unit.Cost ~ sales$Customer.Gender + sales$Year\_Month, FUN = mean)

## sales$Customer.Gender sales$Year\_Month sales$Unit.Cost  
## 1 F 2015-01-01 1024.2782  
## 2 M 2015-01-01 1072.8134  
## 3 F 2015-02-01 1090.6681  
## 4 M 2015-02-01 1064.0688  
## 5 F 2015-03-01 953.7260  
## 6 M 2015-03-01 1156.7830  
## 7 F 2015-04-01 997.3656  
## 8 M 2015-04-01 1082.0526  
## 9 F 2015-05-01 899.2399  
## 10 M 2015-05-01 1024.4358  
## 11 F 2015-06-01 1018.0922  
## 12 M 2015-06-01 1051.0029  
## 13 F 2015-07-01 540.4408  
## 14 M 2015-07-01 499.9978  
## 15 F 2015-08-01 303.5210  
## 16 M 2015-08-01 321.8956  
## 17 F 2015-09-01 340.5652  
## 18 M 2015-09-01 309.7761  
## 19 F 2015-10-01 317.9064  
## 20 M 2015-10-01 288.6653  
## 21 F 2015-11-01 345.1110  
## 22 M 2015-11-01 333.0269  
## 23 F 2015-12-01 363.0341  
## 24 M 2015-12-01 364.4843  
## 25 F 2016-01-01 324.6154  
## 26 M 2016-01-01 313.6044  
## 27 F 2016-02-01 317.1253  
## 28 M 2016-02-01 327.7523  
## 29 F 2016-03-01 320.9585  
## 30 M 2016-03-01 318.0354  
## 31 F 2016-04-01 307.1809  
## 32 M 2016-04-01 312.8382  
## 33 F 2016-05-01 340.5115  
## 34 M 2016-05-01 350.8753  
## 35 F 2016-06-01 337.6228  
## 36 M 2016-06-01 353.0597  
## 37 F 2016-07-01 163.9390  
## 38 M 2016-07-01 200.2161

aggregate(sales$Unit.Cost ~ sales$Country + sales$Year\_Month, FUN = mean)

## sales$Country sales$Year\_Month sales$Unit.Cost  
## 1 France 2015-01-01 1063.6582  
## 2 Germany 2015-01-01 998.9676  
## 3 United Kingdom 2015-01-01 1121.3624  
## 4 United States 2015-01-01 1025.9423  
## 5 France 2015-02-01 965.5635  
## 6 Germany 2015-02-01 1307.8378  
## 7 United Kingdom 2015-02-01 1091.3868  
## 8 United States 2015-02-01 997.6150  
## 9 France 2015-03-01 1392.6558  
## 10 Germany 2015-03-01 1173.0738  
## 11 United Kingdom 2015-03-01 1025.9947  
## 12 United States 2015-03-01 800.5484  
## 13 France 2015-04-01 1056.4231  
## 14 Germany 2015-04-01 1202.4452  
## 15 United Kingdom 2015-04-01 1093.3440  
## 16 United States 2015-04-01 921.7679  
## 17 France 2015-05-01 906.6355  
## 18 Germany 2015-05-01 937.7876  
## 19 United Kingdom 2015-05-01 1098.5966  
## 20 United States 2015-05-01 934.7413  
## 21 France 2015-06-01 1171.0151  
## 22 Germany 2015-06-01 1250.9939  
## 23 United Kingdom 2015-06-01 1115.6121  
## 24 United States 2015-06-01 832.2802  
## 25 France 2015-07-01 594.4692  
## 26 Germany 2015-07-01 488.4916  
## 27 United Kingdom 2015-07-01 554.2560  
## 28 United States 2015-07-01 496.5382  
## 29 France 2015-08-01 321.0403  
## 30 Germany 2015-08-01 344.5818  
## 31 United Kingdom 2015-08-01 309.1548  
## 32 United States 2015-08-01 302.9127  
## 33 France 2015-09-01 325.1658  
## 34 Germany 2015-09-01 351.3401  
## 35 United Kingdom 2015-09-01 323.7445  
## 36 United States 2015-09-01 318.0430  
## 37 France 2015-10-01 304.8455  
## 38 Germany 2015-10-01 306.6312  
## 39 United Kingdom 2015-10-01 315.4241  
## 40 United States 2015-10-01 298.0517  
## 41 France 2015-11-01 341.1749  
## 42 Germany 2015-11-01 365.0384  
## 43 United Kingdom 2015-11-01 362.2252  
## 44 United States 2015-11-01 325.8737  
## 45 France 2015-12-01 369.1429  
## 46 Germany 2015-12-01 372.7972  
## 47 United Kingdom 2015-12-01 335.1831  
## 48 United States 2015-12-01 370.0076  
## 49 France 2016-01-01 331.6989  
## 50 Germany 2016-01-01 355.6025  
## 51 United Kingdom 2016-01-01 344.9067  
## 52 United States 2016-01-01 295.1016  
## 53 France 2016-02-01 369.1255  
## 54 Germany 2016-02-01 341.0294  
## 55 United Kingdom 2016-02-01 335.5830  
## 56 United States 2016-02-01 298.3892  
## 57 France 2016-03-01 319.8872  
## 58 Germany 2016-03-01 382.3906  
## 59 United Kingdom 2016-03-01 366.3124  
## 60 United States 2016-03-01 282.2046  
## 61 France 2016-04-01 336.5350  
## 62 Germany 2016-04-01 349.7710  
## 63 United Kingdom 2016-04-01 360.4018  
## 64 United States 2016-04-01 271.5368  
## 65 France 2016-05-01 378.6312  
## 66 Germany 2016-05-01 360.9699  
## 67 United Kingdom 2016-05-01 352.4468  
## 68 United States 2016-05-01 330.7148  
## 69 France 2016-06-01 356.3635  
## 70 Germany 2016-06-01 375.4935  
## 71 United Kingdom 2016-06-01 376.4320  
## 72 United States 2016-06-01 321.1690  
## 73 France 2016-07-01 162.9963  
## 74 Germany 2016-07-01 237.9910  
## 75 United Kingdom 2016-07-01 185.6347  
## 76 United States 2016-07-01 172.0732

aggregate(sales$Unit.Cost ~ sales$Product.Category + sales$Year\_Month, FUN = mean)

## sales$Product.Category sales$Year\_Month sales$Unit.Cost  
## 1 Bikes 2015-01-01 1051.3594  
## 2 Bikes 2015-02-01 1077.3684  
## 3 Bikes 2015-03-01 1058.6615  
## 4 Bikes 2015-04-01 1040.9053  
## 5 Bikes 2015-05-01 963.6790  
## 6 Bikes 2015-06-01 1033.2695  
## 7 Accessories 2015-07-01 181.2393  
## 8 Bikes 2015-07-01 1024.8962  
## 9 Clothing 2015-07-01 324.7959  
## 10 Accessories 2015-08-01 158.8707  
## 11 Bikes 2015-08-01 958.4090  
## 12 Clothing 2015-08-01 354.4951  
## 13 Accessories 2015-09-01 170.1438  
## 14 Bikes 2015-09-01 932.7025  
## 15 Clothing 2015-09-01 365.6668  
## 16 Accessories 2015-10-01 155.5605  
## 17 Bikes 2015-10-01 880.3178  
## 18 Clothing 2015-10-01 363.6709  
## 19 Accessories 2015-11-01 160.4678  
## 20 Bikes 2015-11-01 976.6356  
## 21 Clothing 2015-11-01 360.4438  
## 22 Accessories 2015-12-01 165.6234  
## 23 Bikes 2015-12-01 938.4188  
## 24 Clothing 2015-12-01 370.4224  
## 25 Accessories 2016-01-01 160.6589  
## 26 Bikes 2016-01-01 967.9178  
## 27 Clothing 2016-01-01 326.5256  
## 28 Accessories 2016-02-01 160.8688  
## 29 Bikes 2016-02-01 927.0671  
## 30 Clothing 2016-02-01 324.7894  
## 31 Accessories 2016-03-01 163.1881  
## 32 Bikes 2016-03-01 928.8091  
## 33 Clothing 2016-03-01 312.1734  
## 34 Accessories 2016-04-01 163.6281  
## 35 Bikes 2016-04-01 858.1909  
## 36 Clothing 2016-04-01 325.5440  
## 37 Accessories 2016-05-01 168.4882  
## 38 Bikes 2016-05-01 951.4282  
## 39 Clothing 2016-05-01 332.9817  
## 40 Accessories 2016-06-01 166.4835  
## 41 Bikes 2016-06-01 948.8526  
## 42 Clothing 2016-06-01 305.4224  
## 43 Accessories 2016-07-01 153.4127  
## 44 Clothing 2016-07-01 307.1161

aggregate(sales$Unit.Cost ~ sales$Sub.Category + sales$Year\_Month, FUN = mean)

## sales$Sub.Category sales$Year\_Month sales$Unit.Cost  
## 1 Mountain Bikes 2015-01-01 1266.27356  
## 2 Road Bikes 2015-01-01 947.36871  
## 3 Mountain Bikes 2015-02-01 1261.14898  
## 4 Road Bikes 2015-02-01 991.60419  
## 5 Mountain Bikes 2015-03-01 1312.74587  
## 6 Road Bikes 2015-03-01 945.18689  
## 7 Mountain Bikes 2015-04-01 1327.07981  
## 8 Road Bikes 2015-04-01 921.85664  
## 9 Mountain Bikes 2015-05-01 1200.74919  
## 10 Road Bikes 2015-05-01 828.73131  
## 11 Mountain Bikes 2015-06-01 1264.69427  
## 12 Road Bikes 2015-06-01 880.23048  
## 13 Bike Stands 2015-07-01 459.33333  
## 14 Bottles and Cages 2015-07-01 64.97831  
## 15 Caps 2015-07-01 85.12500  
## 16 Cleaners 2015-07-01 77.82077  
## 17 Helmets 2015-07-01 346.64311  
## 18 Hydration Packs 2015-07-01 541.53846  
## 19 Jerseys 2015-07-01 462.79450  
## 20 Mountain Bikes 2015-07-01 1276.67566  
## 21 Road Bikes 2015-07-01 924.63796  
## 22 Shorts 2015-07-01 505.55667  
## 23 Socks 2015-07-01 74.25000  
## 24 Tires and Tubes 2015-07-01 162.07395  
## 25 Touring Bikes 2015-07-01 813.16924  
## 26 Vests 2015-07-01 521.11000  
## 27 Bike Stands 2015-08-01 517.76923  
## 28 Bottles and Cages 2015-08-01 67.63262  
## 29 Caps 2015-08-01 86.03571  
## 30 Cleaners 2015-08-01 87.17488  
## 31 Helmets 2015-08-01 318.58083  
## 32 Hydration Packs 2015-08-01 426.75926  
## 33 Jerseys 2015-08-01 467.01445  
## 34 Mountain Bikes 2015-08-01 1172.80199  
## 35 Road Bikes 2015-08-01 758.14542  
## 36 Shorts 2015-08-01 730.07400  
## 37 Socks 2015-08-01 88.59375  
## 38 Tires and Tubes 2015-08-01 127.98639  
## 39 Touring Bikes 2015-08-01 964.93127  
## 40 Vests 2015-08-01 667.50000  
## 41 Bike Stands 2015-09-01 471.70000  
## 42 Bottles and Cages 2015-09-01 68.63554  
## 43 Caps 2015-09-01 87.97059  
## 44 Cleaners 2015-09-01 75.30359  
## 45 Helmets 2015-09-01 345.84165  
## 46 Hydration Packs 2015-09-01 466.64031  
## 47 Jerseys 2015-09-01 440.08675  
## 48 Mountain Bikes 2015-09-01 1118.12584  
## 49 Road Bikes 2015-09-01 737.99210  
## 50 Shorts 2015-09-01 638.16671  
## 51 Socks 2015-09-01 105.30000  
## 52 Tires and Tubes 2015-09-01 137.46222  
## 53 Touring Bikes 2015-09-01 998.94026  
## 54 Vests 2015-09-01 580.59200  
## 55 Bike Stands 2015-10-01 511.07143  
## 56 Bottles and Cages 2015-10-01 65.98896  
## 57 Caps 2015-10-01 88.50000  
## 58 Cleaners 2015-10-01 84.90476  
## 59 Helmets 2015-10-01 319.65115  
## 60 Hydration Packs 2015-10-01 510.46906  
## 61 Jerseys 2015-10-01 497.68054  
## 62 Mountain Bikes 2015-10-01 1011.18593  
## 63 Road Bikes 2015-10-01 687.31728  
## 64 Shorts 2015-10-01 570.98039  
## 65 Socks 2015-10-01 72.15000  
## 66 Tires and Tubes 2015-10-01 125.95893  
## 67 Touring Bikes 2015-10-01 1001.52490  
## 68 Vests 2015-10-01 599.28000  
## 69 Bike Stands 2015-11-01 505.53846  
## 70 Bottles and Cages 2015-11-01 66.34288  
## 71 Caps 2015-11-01 100.00472  
## 72 Cleaners 2015-11-01 79.63805  
## 73 Helmets 2015-11-01 297.58798  
## 74 Hydration Packs 2015-11-01 396.28231  
## 75 Jerseys 2015-11-01 445.79994  
## 76 Mountain Bikes 2015-11-01 1124.59605  
## 77 Road Bikes 2015-11-01 713.13753  
## 78 Shorts 2015-11-01 625.22750  
## 79 Socks 2015-11-01 67.95000  
## 80 Tires and Tubes 2015-11-01 139.98462  
## 81 Touring Bikes 2015-11-01 1210.66402  
## 82 Vests 2015-11-01 743.79967  
## 83 Bike Stands 2015-12-01 627.71875  
## 84 Bottles and Cages 2015-12-01 66.99461  
## 85 Caps 2015-12-01 91.27742  
## 86 Cleaners 2015-12-01 92.35404  
## 87 Helmets 2015-12-01 322.76458  
## 88 Hydration Packs 2015-12-01 538.48860  
## 89 Jerseys 2015-12-01 460.62046  
## 90 Mountain Bikes 2015-12-01 1042.98016  
## 91 Road Bikes 2015-12-01 740.02792  
## 92 Shorts 2015-12-01 843.18182  
## 93 Socks 2015-12-01 76.80882  
## 94 Tires and Tubes 2015-12-01 135.11354  
## 95 Touring Bikes 2015-12-01 1054.60712  
## 96 Vests 2015-12-01 654.70407  
## 97 Bike Racks 2016-01-01 880.00000  
## 98 Bike Stands 2016-01-01 408.10000  
## 99 Bottles and Cages 2016-01-01 66.93960  
## 100 Caps 2016-01-01 90.21239  
## 101 Cleaners 2016-01-01 55.32745  
## 102 Fenders 2016-01-01 212.73733  
## 103 Gloves 2016-01-01 216.58215  
## 104 Helmets 2016-01-01 312.13045  
## 105 Hydration Packs 2016-01-01 565.45229  
## 106 Jerseys 2016-01-01 450.78872  
## 107 Mountain Bikes 2016-01-01 1121.45678  
## 108 Road Bikes 2016-01-01 688.51776  
## 109 Shorts 2016-01-01 664.38579  
## 110 Socks 2016-01-01 84.05000  
## 111 Tires and Tubes 2016-01-01 129.53094  
## 112 Touring Bikes 2016-01-01 1105.41259  
## 113 Vests 2016-01-01 603.00640  
## 114 Bike Racks 2016-02-01 433.33333  
## 115 Bike Stands 2016-02-01 612.81250  
## 116 Bottles and Cages 2016-02-01 65.92770  
## 117 Caps 2016-02-01 84.82377  
## 118 Cleaners 2016-02-01 79.08320  
## 119 Fenders 2016-02-01 219.09400  
## 120 Gloves 2016-02-01 229.03279  
## 121 Helmets 2016-02-01 319.48882  
## 122 Hydration Packs 2016-02-01 596.48857  
## 123 Jerseys 2016-02-01 481.92043  
## 124 Mountain Bikes 2016-02-01 1048.93045  
## 125 Road Bikes 2016-02-01 678.65144  
## 126 Shorts 2016-02-01 603.48500  
## 127 Socks 2016-02-01 77.88462  
## 128 Tires and Tubes 2016-02-01 128.46216  
## 129 Touring Bikes 2016-02-01 1114.29185  
## 130 Vests 2016-02-01 628.38040  
## 131 Bike Racks 2016-03-01 369.41176  
## 132 Bike Stands 2016-03-01 745.31250  
## 133 Bottles and Cages 2016-03-01 65.87172  
## 134 Caps 2016-03-01 77.00000  
## 135 Cleaners 2016-03-01 69.25714  
## 136 Fenders 2016-03-01 221.11544  
## 137 Gloves 2016-03-01 216.14516  
## 138 Helmets 2016-03-01 327.86114  
## 139 Hydration Packs 2016-03-01 545.90426  
## 140 Jerseys 2016-03-01 463.74545  
## 141 Mountain Bikes 2016-03-01 1086.80957  
## 142 Road Bikes 2016-03-01 686.30776  
## 143 Shorts 2016-03-01 626.30049  
## 144 Socks 2016-03-01 89.22727  
## 145 Tires and Tubes 2016-03-01 117.60412  
## 146 Touring Bikes 2016-03-01 1041.07279  
## 147 Vests 2016-03-01 699.14625  
## 148 Bike Racks 2016-04-01 461.66667  
## 149 Bike Stands 2016-04-01 474.05556  
## 150 Bottles and Cages 2016-04-01 66.40566  
## 151 Caps 2016-04-01 85.20629  
## 152 Cleaners 2016-04-01 83.46650  
## 153 Fenders 2016-04-01 204.23918  
## 154 Gloves 2016-04-01 275.04930  
## 155 Helmets 2016-04-01 329.02138  
## 156 Hydration Packs 2016-04-01 570.00000  
## 157 Jerseys 2016-04-01 463.66295  
## 158 Mountain Bikes 2016-04-01 984.86238  
## 159 Road Bikes 2016-04-01 668.66399  
## 160 Shorts 2016-04-01 601.66653  
## 161 Socks 2016-04-01 74.48571  
## 162 Tires and Tubes 2016-04-01 127.17569  
## 163 Touring Bikes 2016-04-01 994.96678  
## 164 Vests 2016-04-01 558.91676  
## 165 Bike Racks 2016-05-01 713.57143  
## 166 Bike Stands 2016-05-01 442.92857  
## 167 Bottles and Cages 2016-05-01 68.70092  
## 168 Caps 2016-05-01 90.32847  
## 169 Cleaners 2016-05-01 73.60273  
## 170 Fenders 2016-05-01 204.57609  
## 171 Gloves 2016-05-01 235.50863  
## 172 Helmets 2016-05-01 328.41666  
## 173 Hydration Packs 2016-05-01 550.27758  
## 174 Jerseys 2016-05-01 499.13372  
## 175 Mountain Bikes 2016-05-01 1085.77396  
## 176 Road Bikes 2016-05-01 727.01520  
## 177 Shorts 2016-05-01 714.45630  
## 178 Socks 2016-05-01 107.77273  
## 179 Tires and Tubes 2016-05-01 121.71231  
## 180 Touring Bikes 2016-05-01 1089.83595  
## 181 Vests 2016-05-01 464.33897  
## 182 Bike Racks 2016-06-01 924.00000  
## 183 Bike Stands 2016-06-01 469.42857  
## 184 Bottles and Cages 2016-06-01 70.59320  
## 185 Caps 2016-06-01 83.43413  
## 186 Cleaners 2016-06-01 68.01870  
## 187 Fenders 2016-06-01 233.30442  
## 188 Gloves 2016-06-01 249.17629  
## 189 Helmets 2016-06-01 310.05751  
## 190 Hydration Packs 2016-06-01 446.66659  
## 191 Jerseys 2016-06-01 450.85872  
## 192 Mountain Bikes 2016-06-01 1150.30572  
## 193 Road Bikes 2016-06-01 655.03219  
## 194 Shorts 2016-06-01 639.10020  
## 195 Socks 2016-06-01 72.00000  
## 196 Tires and Tubes 2016-06-01 127.82509  
## 197 Touring Bikes 2016-06-01 1038.18950  
## 198 Vests 2016-06-01 565.79150  
## 199 Bike Racks 2016-07-01 990.00000  
## 200 Bike Stands 2016-07-01 251.75000  
## 201 Bottles and Cages 2016-07-01 59.13954  
## 202 Caps 2016-07-01 72.04918  
## 203 Cleaners 2016-07-01 61.91391  
## 204 Fenders 2016-07-01 217.28649  
## 205 Gloves 2016-07-01 258.00892  
## 206 Helmets 2016-07-01 299.73786  
## 207 Hydration Packs 2016-07-01 536.90500  
## 208 Jerseys 2016-07-01 400.11842  
## 209 Shorts 2016-07-01 629.22233  
## 210 Socks 2016-07-01 75.00000  
## 211 Tires and Tubes 2016-07-01 117.45493  
## 212 Vests 2016-07-01 429.89778

aggregate(sales$Unit.Price ~ sales$Customer.Gender + sales$Year\_Month, FUN = mean)

## sales$Customer.Gender sales$Year\_Month sales$Unit.Price  
## 1 F 2015-01-01 1007.6831  
## 2 M 2015-01-01 1051.0714  
## 3 F 2015-02-01 1076.0195  
## 4 M 2015-02-01 1045.6775  
## 5 F 2015-03-01 946.8264  
## 6 M 2015-03-01 1111.5736  
## 7 F 2015-04-01 982.2888  
## 8 M 2015-04-01 1061.4689  
## 9 F 2015-05-01 876.4899  
## 10 M 2015-05-01 1008.1365  
## 11 F 2015-06-01 1016.1877  
## 12 M 2015-06-01 1023.2088  
## 13 F 2015-07-01 553.5162  
## 14 M 2015-07-01 504.4369  
## 15 F 2015-08-01 320.8842  
## 16 M 2015-08-01 337.6752  
## 17 F 2015-09-01 357.7796  
## 18 M 2015-09-01 322.5723  
## 19 F 2015-10-01 333.2364  
## 20 M 2015-10-01 305.1355  
## 21 F 2015-11-01 358.8886  
## 22 M 2015-11-01 345.0055  
## 23 F 2015-12-01 374.4239  
## 24 M 2015-12-01 377.6886  
## 25 F 2016-01-01 388.4534  
## 26 M 2016-01-01 372.8691  
## 27 F 2016-02-01 372.3343  
## 28 M 2016-02-01 386.7153  
## 29 F 2016-03-01 382.3212  
## 30 M 2016-03-01 379.2458  
## 31 F 2016-04-01 368.5901  
## 32 M 2016-04-01 369.9913  
## 33 F 2016-05-01 401.4761  
## 34 M 2016-05-01 415.8691  
## 35 F 2016-06-01 398.6182  
## 36 M 2016-06-01 415.5797  
## 37 F 2016-07-01 208.5224  
## 38 M 2016-07-01 260.4032

aggregate(sales$Unit.Price ~ sales$Country + sales$Year\_Month, FUN = mean)

## sales$Country sales$Year\_Month sales$Unit.Price  
## 1 France 2015-01-01 1098.0530  
## 2 Germany 2015-01-01 1134.4683  
## 3 United Kingdom 2015-01-01 1062.6569  
## 4 United States 2015-01-01 955.5710  
## 5 France 2015-02-01 913.3782  
## 6 Germany 2015-02-01 1478.9688  
## 7 United Kingdom 2015-02-01 1029.6422  
## 8 United States 2015-02-01 924.0000  
## 9 France 2015-03-01 1292.3333  
## 10 Germany 2015-03-01 1335.9080  
## 11 United Kingdom 2015-03-01 973.4213  
## 12 United States 2015-03-01 731.9837  
## 13 France 2015-04-01 1037.9524  
## 14 Germany 2015-04-01 1356.0667  
## 15 United Kingdom 2015-04-01 1027.1167  
## 16 United States 2015-04-01 828.9722  
## 17 France 2015-05-01 889.4697  
## 18 Germany 2015-05-01 1079.1441  
## 19 United Kingdom 2015-05-01 1038.2642  
## 20 United States 2015-05-01 868.5717  
## 21 France 2015-06-01 1166.0220  
## 22 Germany 2015-06-01 1408.7222  
## 23 United Kingdom 2015-06-01 1054.1667  
## 24 United States 2015-06-01 765.9651  
## 25 France 2015-07-01 586.8856  
## 26 Germany 2015-07-01 575.1996  
## 27 United Kingdom 2015-07-01 551.4121  
## 28 United States 2015-07-01 482.5511  
## 29 France 2015-08-01 336.5180  
## 30 Germany 2015-08-01 422.0995  
## 31 United Kingdom 2015-08-01 314.9219  
## 32 United States 2015-08-01 308.4959  
## 33 France 2015-09-01 329.1444  
## 34 Germany 2015-09-01 427.9246  
## 35 United Kingdom 2015-09-01 332.7986  
## 36 United States 2015-09-01 322.0979  
## 37 France 2015-10-01 312.3935  
## 38 Germany 2015-10-01 372.4438  
## 39 United Kingdom 2015-10-01 323.7659  
## 40 United States 2015-10-01 305.0853  
## 41 France 2015-11-01 349.2430  
## 42 Germany 2015-11-01 442.6695  
## 43 United Kingdom 2015-11-01 367.9020  
## 44 United States 2015-11-01 328.6765  
## 45 France 2015-12-01 371.7726  
## 46 Germany 2015-12-01 451.6632  
## 47 United Kingdom 2015-12-01 341.6866  
## 48 United States 2015-12-01 369.3323  
## 49 France 2016-01-01 389.9802  
## 50 Germany 2016-01-01 487.9775  
## 51 United Kingdom 2016-01-01 400.1743  
## 52 United States 2016-01-01 337.6571  
## 53 France 2016-02-01 418.5269  
## 54 Germany 2016-02-01 466.6042  
## 55 United Kingdom 2016-02-01 390.5973  
## 56 United States 2016-02-01 339.7211  
## 57 France 2016-03-01 365.9786  
## 58 Germany 2016-03-01 523.6957  
## 59 United Kingdom 2016-03-01 422.7770  
## 60 United States 2016-03-01 325.9792  
## 61 France 2016-04-01 384.1493  
## 62 Germany 2016-04-01 479.1795  
## 63 United Kingdom 2016-04-01 416.9673  
## 64 United States 2016-04-01 313.9410  
## 65 France 2016-05-01 433.8619  
## 66 Germany 2016-05-01 496.7594  
## 67 United Kingdom 2016-05-01 405.1524  
## 68 United States 2016-05-01 376.1516  
## 69 France 2016-06-01 407.2777  
## 70 Germany 2016-06-01 515.0317  
## 71 United Kingdom 2016-06-01 429.5893  
## 72 United States 2016-06-01 365.0970  
## 73 France 2016-07-01 206.3788  
## 74 Germany 2016-07-01 364.2667  
## 75 United Kingdom 2016-07-01 233.8816  
## 76 United States 2016-07-01 210.8536

aggregate(sales$Unit.Price ~ sales$Product.Category + sales$Year\_Month, FUN = mean)

## sales$Product.Category sales$Year\_Month sales$Unit.Price  
## 1 Bikes 2015-01-01 1031.8925  
## 2 Bikes 2015-02-01 1060.8485  
## 3 Bikes 2015-03-01 1031.9642  
## 4 Bikes 2015-04-01 1022.9972  
## 5 Bikes 2015-05-01 944.2492  
## 6 Bikes 2015-06-01 1019.4256  
## 7 Accessories 2015-07-01 211.1904  
## 8 Bikes 2015-07-01 997.2239  
## 9 Clothing 2015-07-01 364.4468  
## 10 Accessories 2015-08-01 180.2468  
## 11 Bikes 2015-08-01 932.6494  
## 12 Clothing 2015-08-01 392.8160  
## 13 Accessories 2015-09-01 192.5845  
## 14 Bikes 2015-09-01 900.8925  
## 15 Clothing 2015-09-01 396.0332  
## 16 Accessories 2015-10-01 176.4247  
## 17 Bikes 2015-10-01 855.6565  
## 18 Clothing 2015-10-01 399.4013  
## 19 Accessories 2015-11-01 182.4555  
## 20 Bikes 2015-11-01 939.4872  
## 21 Clothing 2015-11-01 394.0489  
## 22 Accessories 2015-12-01 189.1046  
## 23 Bikes 2015-12-01 902.7163  
## 24 Clothing 2015-12-01 406.1636  
## 25 Accessories 2016-01-01 208.0102  
## 26 Bikes 2016-01-01 1064.3005  
## 27 Clothing 2016-01-01 412.2412  
## 28 Accessories 2016-02-01 207.0103  
## 29 Bikes 2016-02-01 1009.3640  
## 30 Clothing 2016-02-01 401.4295  
## 31 Accessories 2016-03-01 210.2354  
## 32 Bikes 2016-03-01 1025.9017  
## 33 Clothing 2016-03-01 394.7473  
## 34 Accessories 2016-04-01 209.6121  
## 35 Bikes 2016-04-01 948.7100  
## 36 Clothing 2016-04-01 404.9961  
## 37 Accessories 2016-05-01 217.9263  
## 38 Bikes 2016-05-01 1042.4823  
## 39 Clothing 2016-05-01 419.4961  
## 40 Accessories 2016-06-01 215.6921  
## 41 Bikes 2016-06-01 1037.3127  
## 42 Clothing 2016-06-01 383.5577  
## 43 Accessories 2016-07-01 199.1130  
## 44 Clothing 2016-07-01 387.8832

aggregate(sales$Unit.Price ~ sales$Sub.Category + sales$Year\_Month, FUN = mean)

## sales$Sub.Category sales$Year\_Month sales$Unit.Price  
## 1 Mountain Bikes 2015-01-01 1220.46296  
## 2 Road Bikes 2015-01-01 940.64875  
## 3 Mountain Bikes 2015-02-01 1227.37075  
## 4 Road Bikes 2015-02-01 983.13810  
## 5 Mountain Bikes 2015-03-01 1260.74275  
## 6 Road Bikes 2015-03-01 929.79126  
## 7 Mountain Bikes 2015-04-01 1288.49679  
## 8 Road Bikes 2015-04-01 912.54933  
## 9 Mountain Bikes 2015-05-01 1148.90090  
## 10 Road Bikes 2015-05-01 827.75513  
## 11 Mountain Bikes 2015-06-01 1252.86382  
## 12 Road Bikes 2015-06-01 865.05511  
## 13 Bike Stands 2015-07-01 487.00000  
## 14 Bottles and Cages 2015-07-01 76.48646  
## 15 Caps 2015-07-01 93.98750  
## 16 Cleaners 2015-07-01 90.17949  
## 17 Helmets 2015-07-01 403.64308  
## 18 Hydration Packs 2015-07-01 624.48718  
## 19 Jerseys 2015-07-01 519.55000  
## 20 Mountain Bikes 2015-07-01 1240.06557  
## 21 Road Bikes 2015-07-01 894.82335  
## 22 Shorts 2015-07-01 479.44444  
## 23 Socks 2015-07-01 89.29167  
## 24 Tires and Tubes 2015-07-01 189.74280  
## 25 Touring Bikes 2015-07-01 807.43939  
## 26 Vests 2015-07-01 616.42593  
## 27 Bike Stands 2015-08-01 607.30769  
## 28 Bottles and Cages 2015-08-01 75.65904  
## 29 Caps 2015-08-01 96.66964  
## 30 Cleaners 2015-08-01 96.30488  
## 31 Helmets 2015-08-01 363.99945  
## 32 Hydration Packs 2015-08-01 467.82716  
## 33 Jerseys 2015-08-01 523.12044  
## 34 Mountain Bikes 2015-08-01 1120.51773  
## 35 Road Bikes 2015-08-01 745.23203  
## 36 Shorts 2015-08-01 777.05185  
## 37 Socks 2015-08-01 92.00521  
## 38 Tires and Tubes 2015-08-01 145.05856  
## 39 Touring Bikes 2015-08-01 967.33862  
## 40 Vests 2015-08-01 775.41667  
## 41 Bike Stands 2015-09-01 524.23333  
## 42 Bottles and Cages 2015-09-01 76.42031  
## 43 Caps 2015-09-01 99.71569  
## 44 Cleaners 2015-09-01 83.82479  
## 45 Helmets 2015-09-01 392.67162  
## 46 Hydration Packs 2015-09-01 528.50000  
## 47 Jerseys 2015-09-01 481.34091  
## 48 Mountain Bikes 2015-09-01 1068.71168  
## 49 Road Bikes 2015-09-01 717.03716  
## 50 Shorts 2015-09-01 676.42381  
## 51 Socks 2015-09-01 109.50667  
## 52 Tires and Tubes 2015-09-01 155.84234  
## 53 Touring Bikes 2015-09-01 976.20085  
## 54 Vests 2015-09-01 627.17500  
## 55 Bike Stands 2015-10-01 596.86508  
## 56 Bottles and Cages 2015-10-01 75.42556  
## 57 Caps 2015-10-01 97.33478  
## 58 Cleaners 2015-10-01 95.07937  
## 59 Helmets 2015-10-01 361.45946  
## 60 Hydration Packs 2015-10-01 588.68750  
## 61 Jerseys 2015-10-01 547.93552  
## 62 Mountain Bikes 2015-10-01 983.69444  
## 63 Road Bikes 2015-10-01 663.14815  
## 64 Shorts 2015-10-01 617.20261  
## 65 Socks 2015-10-01 81.50000  
## 66 Tires and Tubes 2015-10-01 142.28387  
## 67 Touring Bikes 2015-10-01 980.45486  
## 68 Vests 2015-10-01 664.33871  
## 69 Bike Stands 2015-11-01 571.73077  
## 70 Bottles and Cages 2015-11-01 75.18705  
## 71 Caps 2015-11-01 106.57233  
## 72 Cleaners 2015-11-01 89.96748  
## 73 Helmets 2015-11-01 339.95382  
## 74 Hydration Packs 2015-11-01 440.98718  
## 75 Jerseys 2015-11-01 488.40430  
## 76 Mountain Bikes 2015-11-01 1071.97428  
## 77 Road Bikes 2015-11-01 690.43369  
## 78 Shorts 2015-11-01 678.99242  
## 79 Socks 2015-11-01 75.15000  
## 80 Tires and Tubes 2015-11-01 159.02646  
## 81 Touring Bikes 2015-11-01 1171.83333  
## 82 Vests 2015-11-01 823.27778  
## 83 Bike Stands 2015-12-01 699.19792  
## 84 Bottles and Cages 2015-12-01 76.16579  
## 85 Caps 2015-12-01 101.47097  
## 86 Cleaners 2015-12-01 105.73684  
## 87 Helmets 2015-12-01 372.03137  
## 88 Hydration Packs 2015-12-01 614.85659  
## 89 Jerseys 2015-12-01 512.80799  
## 90 Mountain Bikes 2015-12-01 1011.19872  
## 91 Road Bikes 2015-12-01 711.64962  
## 92 Shorts 2015-12-01 909.20303  
## 93 Socks 2015-12-01 82.45098  
## 94 Tires and Tubes 2015-12-01 153.22492  
## 95 Touring Bikes 2015-12-01 1000.33431  
## 96 Vests 2015-12-01 681.14815  
## 97 Bike Racks 2016-01-01 1157.50000  
## 98 Bike Stands 2016-01-01 538.33333  
## 99 Bottles and Cages 2016-01-01 86.56289  
## 100 Caps 2016-01-01 113.02212  
## 101 Cleaners 2016-01-01 71.57576  
## 102 Fenders 2016-01-01 272.59722  
## 103 Gloves 2016-01-01 271.08718  
## 104 Helmets 2016-01-01 409.23355  
## 105 Hydration Packs 2016-01-01 743.90000  
## 106 Jerseys 2016-01-01 579.79845  
## 107 Mountain Bikes 2016-01-01 1229.88704  
## 108 Road Bikes 2016-01-01 771.21902  
## 109 Shorts 2016-01-01 817.17982  
## 110 Socks 2016-01-01 103.02778  
## 111 Tires and Tubes 2016-01-01 165.72914  
## 112 Touring Bikes 2016-01-01 1201.50000  
## 113 Vests 2016-01-01 734.46667  
## 114 Bike Racks 2016-02-01 580.27778  
## 115 Bike Stands 2016-02-01 754.56250  
## 116 Bottles and Cages 2016-02-01 84.04549  
## 117 Caps 2016-02-01 106.15437  
## 118 Cleaners 2016-02-01 100.58000  
## 119 Fenders 2016-02-01 280.26471  
## 120 Gloves 2016-02-01 284.38798  
## 121 Helmets 2016-02-01 413.40498  
## 122 Hydration Packs 2016-02-01 764.55952  
## 123 Jerseys 2016-02-01 606.61836  
## 124 Mountain Bikes 2016-02-01 1159.22637  
## 125 Road Bikes 2016-02-01 734.21481  
## 126 Shorts 2016-02-01 725.40152  
## 127 Socks 2016-02-01 100.02564  
## 128 Tires and Tubes 2016-02-01 165.20468  
## 129 Touring Bikes 2016-02-01 1189.03549  
## 130 Vests 2016-02-01 738.58000  
## 131 Bike Racks 2016-03-01 461.41176  
## 132 Bike Stands 2016-03-01 888.56250  
## 133 Bottles and Cages 2016-03-01 85.45197  
## 134 Caps 2016-03-01 98.64583  
## 135 Cleaners 2016-03-01 91.31905  
## 136 Fenders 2016-03-01 285.36988  
## 137 Gloves 2016-03-01 267.33602  
## 138 Helmets 2016-03-01 426.74468  
## 139 Hydration Packs 2016-03-01 715.02128  
## 140 Jerseys 2016-03-01 588.63030  
## 141 Mountain Bikes 2016-03-01 1201.42468  
## 142 Road Bikes 2016-03-01 753.46181  
## 143 Shorts 2016-03-01 757.00813  
## 144 Socks 2016-03-01 109.38889  
## 145 Tires and Tubes 2016-03-01 149.50370  
## 146 Touring Bikes 2016-03-01 1155.40710  
## 147 Vests 2016-03-01 941.06250  
## 148 Bike Racks 2016-04-01 587.94444  
## 149 Bike Stands 2016-04-01 614.25000  
## 150 Bottles and Cages 2016-04-01 84.97649  
## 151 Caps 2016-04-01 110.98252  
## 152 Cleaners 2016-04-01 108.93750  
## 153 Fenders 2016-04-01 255.34545  
## 154 Gloves 2016-04-01 336.90845  
## 155 Helmets 2016-04-01 420.65148  
## 156 Hydration Packs 2016-04-01 731.46970  
## 157 Jerseys 2016-04-01 578.85337  
## 158 Mountain Bikes 2016-04-01 1086.81436  
## 159 Road Bikes 2016-04-01 737.31040  
## 160 Shorts 2016-04-01 727.28231  
## 161 Socks 2016-04-01 95.71905  
## 162 Tires and Tubes 2016-04-01 163.78947  
## 163 Touring Bikes 2016-04-01 1106.86667  
## 164 Vests 2016-04-01 701.90686  
## 165 Bike Racks 2016-05-01 937.70238  
## 166 Bike Stands 2016-05-01 564.53571  
## 167 Bottles and Cages 2016-05-01 88.66448  
## 168 Caps 2016-05-01 115.95864  
## 169 Cleaners 2016-05-01 96.07879  
## 170 Fenders 2016-05-01 263.71618  
## 171 Gloves 2016-05-01 293.39649  
## 172 Helmets 2016-05-01 427.10871  
## 173 Hydration Packs 2016-05-01 707.29798  
## 174 Jerseys 2016-05-01 633.39826  
## 175 Mountain Bikes 2016-05-01 1186.75617  
## 176 Road Bikes 2016-05-01 807.98568  
## 177 Shorts 2016-05-01 858.28623  
## 178 Socks 2016-05-01 135.80303  
## 179 Tires and Tubes 2016-05-01 156.46795  
## 180 Touring Bikes 2016-05-01 1180.24183  
## 181 Vests 2016-05-01 624.68391  
## 182 Bike Racks 2016-06-01 1237.19167  
## 183 Bike Stands 2016-06-01 607.33333  
## 184 Bottles and Cages 2016-06-01 90.08082  
## 185 Caps 2016-06-01 106.68064  
## 186 Cleaners 2016-06-01 86.14815  
## 187 Fenders 2016-06-01 300.34662  
## 188 Gloves 2016-06-01 315.62547  
## 189 Helmets 2016-06-01 408.10673  
## 190 Hydration Packs 2016-06-01 563.44318  
## 191 Jerseys 2016-06-01 570.30624  
## 192 Mountain Bikes 2016-06-01 1255.29221  
## 193 Road Bikes 2016-06-01 720.35857  
## 194 Shorts 2016-06-01 770.85667  
## 195 Socks 2016-06-01 91.70000  
## 196 Tires and Tubes 2016-06-01 163.99404  
## 197 Touring Bikes 2016-06-01 1132.23929  
## 198 Vests 2016-06-01 717.75000  
## 199 Bike Racks 2016-07-01 1429.29167  
## 200 Bike Stands 2016-07-01 320.87500  
## 201 Bottles and Cages 2016-07-01 75.32462  
## 202 Caps 2016-07-01 90.52732  
## 203 Cleaners 2016-07-01 79.64493  
## 204 Fenders 2016-07-01 280.97661  
## 205 Gloves 2016-07-01 336.34685  
## 206 Helmets 2016-07-01 385.45178  
## 207 Hydration Packs 2016-07-01 705.45238  
## 208 Jerseys 2016-07-01 512.48684  
## 209 Shorts 2016-07-01 781.60000  
## 210 Socks 2016-07-01 99.08333  
## 211 Tires and Tubes 2016-07-01 151.83123  
## 212 Vests 2016-07-01 511.75926

aggregate(sales$Cost ~ sales$Customer.Gender + sales$Year\_Month, FUN = mean)

## sales$Customer.Gender sales$Year\_Month sales$Cost  
## 1 F 2015-01-01 1759.9180  
## 2 M 2015-01-01 1674.9740  
## 3 F 2015-02-01 1814.7013  
## 4 M 2015-02-01 1613.0519  
## 5 F 2015-03-01 1683.9583  
## 6 M 2015-03-01 1782.8182  
## 7 F 2015-04-01 1573.1395  
## 8 M 2015-04-01 1690.0989  
## 9 F 2015-05-01 1518.8990  
## 10 M 2015-05-01 1695.8095  
## 11 F 2015-06-01 1733.6486  
## 12 M 2015-06-01 1621.1789  
## 13 F 2015-07-01 838.8867  
## 14 M 2015-07-01 828.2118  
## 15 F 2015-08-01 498.5934  
## 16 M 2015-08-01 529.8235  
## 17 F 2015-09-01 550.3093  
## 18 M 2015-09-01 521.2020  
## 19 F 2015-10-01 530.7149  
## 20 M 2015-10-01 497.7309  
## 21 F 2015-11-01 566.8882  
## 22 M 2015-11-01 546.0281  
## 23 F 2015-12-01 593.2001  
## 24 M 2015-12-01 606.8816  
## 25 F 2016-01-01 528.0786  
## 26 M 2016-01-01 512.3169  
## 27 F 2016-02-01 534.6747  
## 28 M 2016-02-01 541.0390  
## 29 F 2016-03-01 521.8811  
## 30 M 2016-03-01 532.0334  
## 31 F 2016-04-01 518.7842  
## 32 M 2016-04-01 517.0796  
## 33 F 2016-05-01 560.7988  
## 34 M 2016-05-01 561.4593  
## 35 F 2016-06-01 570.2362  
## 36 M 2016-06-01 573.3381  
## 37 F 2016-07-01 271.3767  
## 38 M 2016-07-01 328.3317

aggregate(sales$Cost ~ sales$Country + sales$Year\_Month, FUN = mean)

## sales$Country sales$Year\_Month sales$Cost  
## 1 France 2015-01-01 1851.0455  
## 2 Germany 2015-01-01 1805.3810  
## 3 United Kingdom 2015-01-01 1816.0294  
## 4 United States 2015-01-01 1572.9016  
## 5 France 2015-02-01 1579.1538  
## 6 Germany 2015-02-01 2006.0000  
## 7 United Kingdom 2015-02-01 1866.6471  
## 8 United States 2015-02-01 1535.8226  
## 9 France 2015-03-01 1969.9394  
## 10 Germany 2015-03-01 1822.8276  
## 11 United Kingdom 2015-03-01 1714.2222  
## 12 United States 2015-03-01 1547.8431  
## 13 France 2015-04-01 1696.0857  
## 14 Germany 2015-04-01 1835.1500  
## 15 United Kingdom 2015-04-01 1639.0000  
## 16 United States 2015-04-01 1488.1944  
## 17 France 2015-05-01 1651.0606  
## 18 Germany 2015-05-01 1497.6486  
## 19 United Kingdom 2015-05-01 1722.7073  
## 20 United States 2015-05-01 1590.3441  
## 21 France 2015-06-01 1726.0566  
## 22 Germany 2015-06-01 1932.7576  
## 23 United Kingdom 2015-06-01 1682.0000  
## 24 United States 2015-06-01 1558.1047  
## 25 France 2015-07-01 945.4891  
## 26 Germany 2015-07-01 806.0110  
## 27 United Kingdom 2015-07-01 875.2537  
## 28 United States 2015-07-01 788.7206  
## 29 France 2015-08-01 538.4577  
## 30 Germany 2015-08-01 556.8066  
## 31 United Kingdom 2015-08-01 517.6304  
## 32 United States 2015-08-01 494.6492  
## 33 France 2015-09-01 534.2145  
## 34 Germany 2015-09-01 587.1867  
## 35 United Kingdom 2015-09-01 527.5882  
## 36 United States 2015-09-01 524.9123  
## 37 France 2015-10-01 519.2083  
## 38 Germany 2015-10-01 513.4447  
## 39 United Kingdom 2015-10-01 543.7151  
## 40 United States 2015-10-01 503.0642  
## 41 France 2015-11-01 574.3932  
## 42 Germany 2015-11-01 562.0780  
## 43 United Kingdom 2015-11-01 588.7261  
## 44 United States 2015-11-01 541.0042  
## 45 France 2015-12-01 616.7893  
## 46 Germany 2015-12-01 591.8994  
## 47 United Kingdom 2015-12-01 576.8399  
## 48 United States 2015-12-01 605.5617  
## 49 France 2016-01-01 535.7494  
## 50 Germany 2016-01-01 587.5034  
## 51 United Kingdom 2016-01-01 569.3127  
## 52 United States 2016-01-01 477.5603  
## 53 France 2016-02-01 606.2511  
## 54 Germany 2016-02-01 556.6228  
## 55 United Kingdom 2016-02-01 596.0335  
## 56 United States 2016-02-01 490.9437  
## 57 France 2016-03-01 516.5375  
## 58 Germany 2016-03-01 614.2141  
## 59 United Kingdom 2016-03-01 603.4448  
## 60 United States 2016-03-01 475.2596  
## 61 France 2016-04-01 553.6752  
## 62 Germany 2016-04-01 602.9478  
## 63 United Kingdom 2016-04-01 583.0234  
## 64 United States 2016-04-01 457.1253  
## 65 France 2016-05-01 586.9936  
## 66 Germany 2016-05-01 604.6908  
## 67 United Kingdom 2016-05-01 582.4658  
## 68 United States 2016-05-01 534.1341  
## 69 France 2016-06-01 613.5000  
## 70 Germany 2016-06-01 624.6996  
## 71 United Kingdom 2016-06-01 605.4553  
## 72 United States 2016-06-01 529.6919  
## 73 France 2016-07-01 288.9375  
## 74 Germany 2016-07-01 350.4970  
## 75 United Kingdom 2016-07-01 321.1777  
## 76 United States 2016-07-01 283.7156

aggregate(sales$Cost ~ sales$Product.Category + sales$Year\_Month, FUN = mean)

## sales$Product.Category sales$Year\_Month sales$Cost  
## 1 Bikes 2015-01-01 1712.5217  
## 2 Bikes 2015-02-01 1713.8766  
## 3 Bikes 2015-03-01 1735.0470  
## 4 Bikes 2015-04-01 1633.2712  
## 5 Bikes 2015-05-01 1609.9559  
## 6 Bikes 2015-06-01 1681.7816  
## 7 Accessories 2015-07-01 290.0416  
## 8 Bikes 2015-07-01 1631.1465  
## 9 Clothing 2015-07-01 537.4483  
## 10 Accessories 2015-08-01 272.0881  
## 11 Bikes 2015-08-01 1541.6303  
## 12 Clothing 2015-08-01 568.0231  
## 13 Accessories 2015-09-01 277.1891  
## 14 Bikes 2015-09-01 1537.5780  
## 15 Clothing 2015-09-01 616.3881  
## 16 Accessories 2015-10-01 254.9869  
## 17 Bikes 2015-10-01 1558.1127  
## 18 Clothing 2015-10-01 586.9013  
## 19 Accessories 2015-11-01 265.6157  
## 20 Bikes 2015-11-01 1577.9802  
## 21 Clothing 2015-11-01 612.1068  
## 22 Accessories 2015-12-01 271.7408  
## 23 Bikes 2015-12-01 1557.0898  
## 24 Clothing 2015-12-01 603.8217  
## 25 Accessories 2016-01-01 263.9594  
## 26 Bikes 2016-01-01 1558.1659  
## 27 Clothing 2016-01-01 543.8307  
## 28 Accessories 2016-02-01 270.1630  
## 29 Bikes 2016-02-01 1557.4622  
## 30 Clothing 2016-02-01 517.4976  
## 31 Accessories 2016-03-01 268.4541  
## 32 Bikes 2016-03-01 1538.7778  
## 33 Clothing 2016-03-01 513.9104  
## 34 Accessories 2016-04-01 265.5092  
## 35 Bikes 2016-04-01 1478.4150  
## 36 Clothing 2016-04-01 527.6932  
## 37 Accessories 2016-05-01 269.0618  
## 38 Bikes 2016-05-01 1552.6053  
## 39 Clothing 2016-05-01 547.5156  
## 40 Accessories 2016-06-01 272.8484  
## 41 Bikes 2016-06-01 1575.5669  
## 42 Clothing 2016-06-01 507.3216  
## 43 Accessories 2016-07-01 254.1130  
## 44 Clothing 2016-07-01 499.4829

aggregate(sales$Cost ~ sales$Sub.Category + sales$Year\_Month, FUN = mean)

## sales$Sub.Category sales$Year\_Month sales$Cost  
## 1 Mountain Bikes 2015-01-01 2059.75556  
## 2 Road Bikes 2015-01-01 1544.50538  
## 3 Mountain Bikes 2015-02-01 2059.32653  
## 4 Road Bikes 2015-02-01 1552.66667  
## 5 Mountain Bikes 2015-03-01 2059.52174  
## 6 Road Bikes 2015-03-01 1590.13592  
## 7 Mountain Bikes 2015-04-01 2059.15385  
## 8 Road Bikes 2015-04-01 1456.10400  
## 9 Mountain Bikes 2015-05-01 2059.40541  
## 10 Road Bikes 2015-05-01 1354.11538  
## 11 Mountain Bikes 2015-06-01 2061.87805  
## 12 Road Bikes 2015-06-01 1430.42742  
## 13 Bike Stands 2015-07-01 848.00000  
## 14 Bottles and Cages 2015-07-01 107.37500  
## 15 Caps 2015-07-01 123.97500  
## 16 Cleaners 2015-07-01 119.92308  
## 17 Helmets 2015-07-01 540.18868  
## 18 Hydration Packs 2015-07-01 926.53846  
## 19 Jerseys 2015-07-01 732.30000  
## 20 Mountain Bikes 2015-07-01 1861.38525  
## 21 Road Bikes 2015-07-01 1500.72455  
## 22 Shorts 2015-07-01 1260.00000  
## 23 Socks 2015-07-01 193.50000  
## 24 Tires and Tubes 2015-07-01 259.01852  
## 25 Touring Bikes 2015-07-01 1535.56061  
## 26 Vests 2015-07-01 988.11111  
## 27 Bike Stands 2015-08-01 868.38462  
## 28 Bottles and Cages 2015-08-01 110.74695  
## 29 Caps 2015-08-01 152.11607  
## 30 Cleaners 2015-08-01 134.90244  
## 31 Helmets 2015-08-01 553.64686  
## 32 Hydration Packs 2015-08-01 871.85185  
## 33 Jerseys 2015-08-01 748.26277  
## 34 Mountain Bikes 2015-08-01 1877.97163  
## 35 Road Bikes 2015-08-01 1217.15033  
## 36 Shorts 2015-08-01 1146.44444  
## 37 Socks 2015-08-01 142.59375  
## 38 Tires and Tubes 2015-08-01 214.97860  
## 39 Touring Bikes 2015-08-01 1576.88889  
## 40 Vests 2015-08-01 1041.70000  
## 41 Bike Stands 2015-09-01 731.40000  
## 42 Bottles and Cages 2015-09-01 110.46064  
## 43 Caps 2015-09-01 144.35294  
## 44 Cleaners 2015-09-01 116.74359  
## 45 Helmets 2015-09-01 552.95380  
## 46 Hydration Packs 2015-09-01 773.43750  
## 47 Jerseys 2015-09-01 731.49351  
## 48 Mountain Bikes 2015-09-01 1862.63504  
## 49 Road Bikes 2015-09-01 1211.00000  
## 50 Shorts 2015-09-01 1081.00000  
## 51 Socks 2015-09-01 156.60000  
## 52 Tires and Tubes 2015-09-01 228.52941  
## 53 Touring Bikes 2015-09-01 1623.98718  
## 54 Vests 2015-09-01 1086.05000  
## 55 Bike Stands 2015-10-01 825.28571  
## 56 Bottles and Cages 2015-10-01 106.90571  
## 57 Caps 2015-10-01 141.65217  
## 58 Cleaners 2015-10-01 131.64286  
## 59 Helmets 2015-10-01 540.13514  
## 60 Hydration Packs 2015-10-01 715.00000  
## 61 Jerseys 2015-10-01 774.48810  
## 62 Mountain Bikes 2015-10-01 1803.74667  
## 63 Road Bikes 2015-10-01 1224.54321  
## 64 Shorts 2015-10-01 952.54902  
## 65 Socks 2015-10-01 129.30000  
## 66 Tires and Tubes 2015-10-01 206.66562  
## 67 Touring Bikes 2015-10-01 1737.20833  
## 68 Vests 2015-10-01 1063.32258  
## 69 Bike Stands 2015-11-01 966.23077  
## 70 Bottles and Cages 2015-11-01 109.38315  
## 71 Caps 2015-11-01 149.43396  
## 72 Cleaners 2015-11-01 138.92683  
## 73 Helmets 2015-11-01 520.57229  
## 74 Hydration Packs 2015-11-01 755.19231  
## 75 Jerseys 2015-11-01 799.27742  
## 76 Mountain Bikes 2015-11-01 1809.04938  
## 77 Road Bikes 2015-11-01 1240.59140  
## 78 Shorts 2015-11-01 1054.77273  
## 79 Socks 2015-11-01 112.20000  
## 80 Tires and Tubes 2015-11-01 216.29730  
## 81 Touring Bikes 2015-11-01 1814.62617  
## 82 Vests 2015-11-01 1130.50000  
## 83 Bike Stands 2015-12-01 924.18750  
## 84 Bottles and Cages 2015-12-01 109.74346  
## 85 Caps 2015-12-01 142.49032  
## 86 Cleaners 2015-12-01 132.96491  
## 87 Helmets 2015-12-01 530.55556  
## 88 Hydration Packs 2015-12-01 891.51163  
## 89 Jerseys 2015-12-01 787.29954  
## 90 Mountain Bikes 2015-12-01 1801.39423  
## 91 Road Bikes 2015-12-01 1177.07576  
## 92 Shorts 2015-12-01 1277.81818  
## 93 Socks 2015-12-01 138.70588  
## 94 Tires and Tubes 2015-12-01 223.65333  
## 95 Touring Bikes 2015-12-01 1698.85882  
## 96 Vests 2015-12-01 990.33333  
## 97 Bike Racks 2016-01-01 1540.00000  
## 98 Bike Stands 2016-01-01 906.30000  
## 99 Bottles and Cages 2016-01-01 112.68632  
## 100 Caps 2016-01-01 140.65487  
## 101 Cleaners 2016-01-01 105.58182  
## 102 Fenders 2016-01-01 333.11667  
## 103 Gloves 2016-01-01 380.84615  
## 104 Helmets 2016-01-01 505.85761  
## 105 Hydration Packs 2016-01-01 887.85714  
## 106 Jerseys 2016-01-01 750.81395  
## 107 Mountain Bikes 2016-01-01 1774.54444  
## 108 Road Bikes 2016-01-01 1176.64744  
## 109 Shorts 2016-01-01 1085.00000  
## 110 Socks 2016-01-01 151.80000  
## 111 Tires and Tubes 2016-01-01 213.56503  
## 112 Touring Bikes 2016-01-01 1735.48276  
## 113 Vests 2016-01-01 1013.76000  
## 114 Bike Racks 2016-02-01 790.00000  
## 115 Bike Stands 2016-02-01 874.50000  
## 116 Bottles and Cages 2016-02-01 108.66443  
## 117 Caps 2016-02-01 138.68852  
## 118 Cleaners 2016-02-01 134.22000  
## 119 Fenders 2016-02-01 370.41176  
## 120 Gloves 2016-02-01 335.13115  
## 121 Helmets 2016-02-01 539.82866  
## 122 Hydration Packs 2016-02-01 919.28571  
## 123 Jerseys 2016-02-01 798.30435  
## 124 Mountain Bikes 2016-02-01 1760.44776  
## 125 Road Bikes 2016-02-01 1194.05556  
## 126 Shorts 2016-02-01 930.68182  
## 127 Socks 2016-02-01 148.84615  
## 128 Tires and Tubes 2016-02-01 218.45838  
## 129 Touring Bikes 2016-02-01 1785.36111  
## 130 Vests 2016-02-01 917.20000  
## 131 Bike Racks 2016-03-01 670.58824  
## 132 Bike Stands 2016-03-01 914.25000  
## 133 Bottles and Cages 2016-03-01 105.73308  
## 134 Caps 2016-03-01 130.43750  
## 135 Cleaners 2016-03-01 117.25714  
## 136 Fenders 2016-03-01 345.78947  
## 137 Gloves 2016-03-01 388.19355  
## 138 Helmets 2016-03-01 543.52394  
## 139 Hydration Packs 2016-03-01 847.23404  
## 140 Jerseys 2016-03-01 756.52121  
## 141 Mountain Bikes 2016-03-01 1747.50962  
## 142 Road Bikes 2016-03-01 1164.93229  
## 143 Shorts 2016-03-01 1031.21951  
## 144 Socks 2016-03-01 140.72727  
## 145 Tires and Tubes 2016-03-01 199.60932  
## 146 Touring Bikes 2016-03-01 1771.25410  
## 147 Vests 2016-03-01 1100.95833  
## 148 Bike Racks 2016-04-01 760.00000  
## 149 Bike Stands 2016-04-01 936.33333  
## 150 Bottles and Cages 2016-04-01 110.00000  
## 151 Caps 2016-04-01 137.83217  
## 152 Cleaners 2016-04-01 129.27500  
## 153 Fenders 2016-04-01 334.08182  
## 154 Gloves 2016-04-01 413.50704  
## 155 Helmets 2016-04-01 537.39583  
## 156 Hydration Packs 2016-04-01 797.50000  
## 157 Jerseys 2016-04-01 762.79781  
## 158 Mountain Bikes 2016-04-01 1745.61386  
## 159 Road Bikes 2016-04-01 1132.53670  
## 160 Shorts 2016-04-01 908.57143  
## 161 Socks 2016-04-01 133.20000  
## 162 Tires and Tubes 2016-04-01 203.51012  
## 163 Touring Bikes 2016-04-01 1664.73913  
## 164 Vests 2016-04-01 997.61765  
## 165 Bike Racks 2016-05-01 998.57143  
## 166 Bike Stands 2016-05-01 840.42857  
## 167 Bottles and Cages 2016-05-01 110.60600  
## 168 Caps 2016-05-01 139.53285  
## 169 Cleaners 2016-05-01 121.80000  
## 170 Fenders 2016-05-01 313.76812  
## 171 Gloves 2016-05-01 382.97895  
## 172 Helmets 2016-05-01 511.87500  
## 173 Hydration Packs 2016-05-01 868.33333  
## 174 Jerseys 2016-05-01 810.83721  
## 175 Mountain Bikes 2016-05-01 1807.41481  
## 176 Road Bikes 2016-05-01 1143.01953  
## 177 Shorts 2016-05-01 1171.73913  
## 178 Socks 2016-05-01 174.00000  
## 179 Tires and Tubes 2016-05-01 203.05000  
## 180 Touring Bikes 2016-05-01 1788.26144  
## 181 Vests 2016-05-01 887.00000  
## 182 Bike Racks 2016-06-01 1296.00000  
## 183 Bike Stands 2016-06-01 795.00000  
## 184 Bottles and Cages 2016-06-01 114.11545  
## 185 Caps 2016-06-01 137.04790  
## 186 Cleaners 2016-06-01 128.29630  
## 187 Fenders 2016-06-01 341.63768  
## 188 Gloves 2016-06-01 388.19101  
## 189 Helmets 2016-06-01 529.56897  
## 190 Hydration Packs 2016-06-01 811.25000  
## 191 Jerseys 2016-06-01 756.67980  
## 192 Mountain Bikes 2016-06-01 1858.07895  
## 193 Road Bikes 2016-06-01 1153.15936  
## 194 Shorts 2016-06-01 1076.60000  
## 195 Socks 2016-06-01 131.17500  
## 196 Tires and Tubes 2016-06-01 207.04397  
## 197 Touring Bikes 2016-06-01 1719.42857  
## 198 Vests 2016-06-01 927.35000  
## 199 Bike Racks 2016-07-01 1605.00000  
## 200 Bike Stands 2016-07-01 556.50000  
## 201 Bottles and Cages 2016-07-01 97.85621  
## 202 Caps 2016-07-01 133.37705  
## 203 Cleaners 2016-07-01 102.43478  
## 204 Fenders 2016-07-01 346.08772  
## 205 Gloves 2016-07-01 350.78378  
## 206 Helmets 2016-07-01 502.76730  
## 207 Hydration Packs 2016-07-01 801.42857  
## 208 Jerseys 2016-07-01 690.71053  
## 209 Shorts 2016-07-01 970.66667  
## 210 Socks 2016-07-01 147.00000  
## 211 Tires and Tubes 2016-07-01 195.74803  
## 212 Vests 2016-07-01 688.11111

aggregate(sales$Revenue ~ sales$Customer.Gender + sales$Year\_Month, FUN = mean)

## sales$Customer.Gender sales$Year\_Month sales$Revenue  
## 1 F 2015-01-01 1702.3115  
## 2 M 2015-01-01 1645.5584  
## 3 F 2015-02-01 1792.8701  
## 4 M 2015-02-01 1581.8961  
## 5 F 2015-03-01 1645.0417  
## 6 M 2015-03-01 1713.1818  
## 7 F 2015-04-01 1548.7326  
## 8 M 2015-04-01 1658.8132  
## 9 F 2015-05-01 1476.6364  
## 10 M 2015-05-01 1661.3524  
## 11 F 2015-06-01 1700.3243  
## 12 M 2015-06-01 1568.6316  
## 13 F 2015-07-01 858.0915  
## 14 M 2015-07-01 841.0212  
## 15 F 2015-08-01 529.1267  
## 16 M 2015-08-01 556.3125  
## 17 F 2015-09-01 581.7298  
## 18 M 2015-09-01 543.9728  
## 19 F 2015-10-01 554.4299  
## 20 M 2015-10-01 523.3432  
## 21 F 2015-11-01 589.5752  
## 22 M 2015-11-01 567.1373  
## 23 F 2015-12-01 613.0736  
## 24 M 2015-12-01 626.5219  
## 25 F 2016-01-01 632.3277  
## 26 M 2016-01-01 610.6070  
## 27 F 2016-02-01 631.4313  
## 28 M 2016-02-01 637.2608  
## 29 F 2016-03-01 621.4020  
## 30 M 2016-03-01 631.9357  
## 31 F 2016-04-01 622.8450  
## 32 M 2016-04-01 611.5277  
## 33 F 2016-05-01 662.6560  
## 34 M 2016-05-01 665.1390  
## 35 F 2016-06-01 671.6239  
## 36 M 2016-06-01 677.6357  
## 37 F 2016-07-01 344.2616  
## 38 M 2016-07-01 422.8172

aggregate(sales$Revenue ~ sales$Country + sales$Year\_Month, FUN = mean)

## sales$Country sales$Year\_Month sales$Revenue  
## 1 France 2015-01-01 1858.3182  
## 2 Germany 2015-01-01 2030.2857  
## 3 United Kingdom 2015-01-01 1718.2059  
## 4 United States 2015-01-01 1452.6393  
## 5 France 2015-02-01 1525.8846  
## 6 Germany 2015-02-01 2261.2188  
## 7 United Kingdom 2015-02-01 1761.9412  
## 8 United States 2015-02-01 1418.0484  
## 9 France 2015-03-01 1802.4545  
## 10 Germany 2015-03-01 2071.8276  
## 11 United Kingdom 2015-03-01 1627.3333  
## 12 United States 2015-03-01 1415.8824  
## 13 France 2015-04-01 1654.3143  
## 14 Germany 2015-04-01 2075.8250  
## 15 United Kingdom 2015-04-01 1539.3667  
## 16 United States 2015-04-01 1347.6111  
## 17 France 2015-05-01 1608.1818  
## 18 Germany 2015-05-01 1707.4865  
## 19 United Kingdom 2015-05-01 1632.7317  
## 20 United States 2015-05-01 1477.8495  
## 21 France 2015-06-01 1668.8113  
## 22 Germany 2015-06-01 2165.1212  
## 23 United Kingdom 2015-06-01 1592.1471  
## 24 United States 2015-06-01 1438.6860  
## 25 France 2015-07-01 939.7810  
## 26 Germany 2015-07-01 950.2143  
## 27 United Kingdom 2015-07-01 872.0498  
## 28 United States 2015-07-01 764.9093  
## 29 France 2015-08-01 572.4519  
## 30 Germany 2015-08-01 681.5049  
## 31 United Kingdom 2015-08-01 527.0886  
## 32 United States 2015-08-01 504.5611  
## 33 France 2015-09-01 547.8468  
## 34 Germany 2015-09-01 716.3259  
## 35 United Kingdom 2015-09-01 541.9910  
## 36 United States 2015-09-01 533.4317  
## 37 France 2015-10-01 532.3993  
## 38 Germany 2015-10-01 620.1294  
## 39 United Kingdom 2015-10-01 556.4558  
## 40 United States 2015-10-01 512.5065  
## 41 France 2015-11-01 596.5139  
## 42 Germany 2015-11-01 683.6678  
## 43 United Kingdom 2015-11-01 600.1693  
## 44 United States 2015-11-01 545.4581  
## 45 France 2015-12-01 623.5397  
## 46 Germany 2015-12-01 716.2495  
## 47 United Kingdom 2015-12-01 587.9874  
## 48 United States 2015-12-01 604.0722  
## 49 France 2016-01-01 635.3902  
## 50 Germany 2016-01-01 805.7506  
## 51 United Kingdom 2016-01-01 657.3227  
## 52 United States 2016-01-01 547.4850  
## 53 France 2016-02-01 688.5764  
## 54 Germany 2016-02-01 762.1038  
## 55 United Kingdom 2016-02-01 694.3201  
## 56 United States 2016-02-01 560.6933  
## 57 France 2016-03-01 593.0600  
## 58 Germany 2016-03-01 843.3620  
## 59 United Kingdom 2016-03-01 694.2328  
## 60 United States 2016-03-01 546.6939  
## 61 France 2016-04-01 638.9272  
## 62 Germany 2016-04-01 823.0459  
## 63 United Kingdom 2016-04-01 673.9495  
## 64 United States 2016-04-01 526.5288  
## 65 France 2016-05-01 677.6788  
## 66 Germany 2016-05-01 829.7375  
## 67 United Kingdom 2016-05-01 669.5719  
## 68 United States 2016-05-01 606.5510  
## 69 France 2016-06-01 700.5059  
## 70 Germany 2016-06-01 854.9029  
## 71 United Kingdom 2016-06-01 691.9824  
## 72 United States 2016-06-01 603.6751  
## 73 France 2016-07-01 361.3523  
## 74 Germany 2016-07-01 534.0606  
## 75 United Kingdom 2016-07-01 403.7614  
## 76 United States 2016-07-01 347.6008

aggregate(sales$Revenue ~ sales$Product.Category + sales$Year\_Month, FUN = mean)

## sales$Product.Category sales$Year\_Month sales$Revenue  
## 1 Bikes 2015-01-01 1670.6449  
## 2 Bikes 2015-02-01 1687.3831  
## 3 Bikes 2015-03-01 1680.2550  
## 4 Bikes 2015-04-01 1605.3277  
## 5 Bikes 2015-05-01 1571.7108  
## 6 Bikes 2015-06-01 1639.5922  
## 7 Accessories 2015-07-01 338.3523  
## 8 Bikes 2015-07-01 1591.1437  
## 9 Clothing 2015-07-01 599.7500  
## 10 Accessories 2015-08-01 308.1500  
## 11 Bikes 2015-08-01 1504.4874  
## 12 Clothing 2015-08-01 630.1012  
## 13 Accessories 2015-09-01 314.1583  
## 14 Bikes 2015-09-01 1496.2527  
## 15 Clothing 2015-09-01 669.9542  
## 16 Accessories 2015-10-01 289.9737  
## 17 Bikes 2015-10-01 1507.0784  
## 18 Clothing 2015-10-01 643.8911  
## 19 Accessories 2015-11-01 301.6529  
## 20 Bikes 2015-11-01 1520.3099  
## 21 Clothing 2015-11-01 668.5753  
## 22 Accessories 2015-12-01 310.5890  
## 23 Bikes 2015-12-01 1495.4598  
## 24 Clothing 2015-12-01 662.7049  
## 25 Accessories 2016-01-01 342.1478  
## 26 Bikes 2016-01-01 1713.3960  
## 27 Clothing 2016-01-01 687.2054  
## 28 Accessories 2016-02-01 347.7560  
## 29 Bikes 2016-02-01 1699.4969  
## 30 Clothing 2016-02-01 643.3269  
## 31 Accessories 2016-03-01 345.3310  
## 32 Bikes 2016-03-01 1697.6513  
## 33 Clothing 2016-03-01 645.9744  
## 34 Accessories 2016-04-01 341.7096  
## 35 Bikes 2016-04-01 1633.6112  
## 36 Clothing 2016-04-01 659.8252  
## 37 Accessories 2016-05-01 348.2569  
## 38 Bikes 2016-05-01 1703.6583  
## 39 Clothing 2016-05-01 691.4707  
## 40 Accessories 2016-06-01 352.9072  
## 41 Bikes 2016-06-01 1726.3424  
## 42 Clothing 2016-06-01 640.5905  
## 43 Accessories 2016-07-01 327.9373  
## 44 Clothing 2016-07-01 625.1880

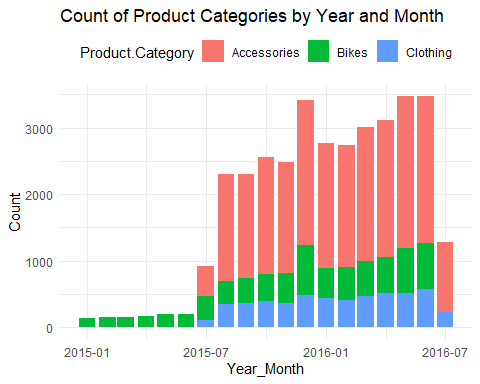
aggregate(sales$Revenue ~ sales$Sub.Category + sales$Year\_Month, FUN = mean)

## sales$Sub.Category sales$Year\_Month sales$Revenue  
## 1 Mountain Bikes 2015-01-01 1948.6889  
## 2 Road Bikes 2015-01-01 1536.1075  
## 3 Mountain Bikes 2015-02-01 2026.5714  
## 4 Road Bikes 2015-02-01 1529.0952  
## 5 Mountain Bikes 2015-03-01 1955.4130  
## 6 Road Bikes 2015-03-01 1557.3689  
## 7 Mountain Bikes 2015-04-01 1977.3846  
## 8 Road Bikes 2015-04-01 1450.5520  
## 9 Mountain Bikes 2015-05-01 1963.4189  
## 10 Road Bikes 2015-05-01 1348.7385  
## 11 Mountain Bikes 2015-06-01 2028.8902  
## 12 Road Bikes 2015-06-01 1382.1532  
## 13 Bike Stands 2015-07-01 893.3333  
## 14 Bottles and Cages 2015-07-01 126.0812  
## 15 Caps 2015-07-01 136.7000  
## 16 Cleaners 2015-07-01 135.9231  
## 17 Helmets 2015-07-01 630.7358  
## 18 Hydration Packs 2015-07-01 1065.2308  
## 19 Jerseys 2015-07-01 818.4167  
## 20 Mountain Bikes 2015-07-01 1810.4672  
## 21 Road Bikes 2015-07-01 1459.2395  
## 22 Shorts 2015-07-01 1181.6667  
## 23 Socks 2015-07-01 231.2500  
## 24 Tires and Tubes 2015-07-01 304.3272  
## 25 Touring Bikes 2015-07-01 1519.4848  
## 26 Vests 2015-07-01 1169.7778  
## 27 Bike Stands 2015-08-01 1003.3846  
## 28 Bottles and Cages 2015-08-01 124.0701  
## 29 Caps 2015-08-01 169.5357  
## 30 Cleaners 2015-08-01 148.4878  
## 31 Helmets 2015-08-01 631.9241  
## 32 Hydration Packs 2015-08-01 960.2593  
## 33 Jerseys 2015-08-01 846.1168  
## 34 Mountain Bikes 2015-08-01 1804.0213  
## 35 Road Bikes 2015-08-01 1197.6536  
## 36 Shorts 2015-08-01 1215.7556  
## 37 Socks 2015-08-01 146.8750  
## 38 Tires and Tubes 2015-08-01 243.0327  
## 39 Touring Bikes 2015-08-01 1579.2698  
## 40 Vests 2015-08-01 1185.0000  
## 41 Bike Stands 2015-09-01 816.7000  
## 42 Bottles and Cages 2015-09-01 123.8076  
## 43 Caps 2015-09-01 163.7059  
## 44 Cleaners 2015-09-01 131.5641  
## 45 Helmets 2015-09-01 629.1386  
## 46 Hydration Packs 2015-09-01 873.4062  
## 47 Jerseys 2015-09-01 799.3182  
## 48 Mountain Bikes 2015-09-01 1802.3358  
## 49 Road Bikes 2015-09-01 1174.2293  
## 50 Shorts 2015-09-01 1154.6429  
## 51 Socks 2015-09-01 163.4800  
## 52 Tires and Tubes 2015-09-01 258.9976  
## 53 Touring Bikes 2015-09-01 1606.8205  
## 54 Vests 2015-09-01 1192.4000  
## 55 Bike Stands 2015-10-01 942.0000  
## 56 Bottles and Cages 2015-10-01 122.7792  
## 57 Caps 2015-10-01 155.9391  
## 58 Cleaners 2015-10-01 149.5714  
## 59 Helmets 2015-10-01 612.7635  
## 60 Hydration Packs 2015-10-01 815.0938  
## 61 Jerseys 2015-10-01 848.9167  
## 62 Mountain Bikes 2015-10-01 1749.2600  
## 63 Road Bikes 2015-10-01 1172.9321  
## 64 Shorts 2015-10-01 1033.4314  
## 65 Socks 2015-10-01 143.4333  
## 66 Tires and Tubes 2015-10-01 234.8370  
## 67 Touring Bikes 2015-10-01 1692.5417  
## 68 Vests 2015-10-01 1186.3871  
## 69 Bike Stands 2015-11-01 1091.2308  
## 70 Bottles and Cages 2015-11-01 123.7283  
## 71 Caps 2015-11-01 161.3679  
## 72 Cleaners 2015-11-01 156.5854  
## 73 Helmets 2015-11-01 594.9940  
## 74 Hydration Packs 2015-11-01 854.7308  
## 75 Jerseys 2015-11-01 875.5806  
## 76 Mountain Bikes 2015-11-01 1718.9938  
## 77 Road Bikes 2015-11-01 1207.5645  
## 78 Shorts 2015-11-01 1149.4545  
## 79 Socks 2015-11-01 123.0667  
## 80 Tires and Tubes 2015-11-01 244.6599  
## 81 Touring Bikes 2015-11-01 1763.1495  
## 82 Vests 2015-11-01 1231.4000  
## 83 Bike Stands 2015-12-01 1028.8750  
## 84 Bottles and Cages 2015-12-01 124.6946  
## 85 Caps 2015-12-01 159.2452  
## 86 Cleaners 2015-12-01 151.8421  
## 87 Helmets 2015-12-01 613.4626  
## 88 Hydration Packs 2015-12-01 1018.3721  
## 89 Jerseys 2015-12-01 876.2903  
## 90 Mountain Bikes 2015-12-01 1738.3814  
## 91 Road Bikes 2015-12-01 1137.3598  
## 92 Shorts 2015-12-01 1366.3636  
## 93 Socks 2015-12-01 148.9118  
## 94 Tires and Tubes 2015-12-01 253.5143  
## 95 Touring Bikes 2015-12-01 1605.7353  
## 96 Vests 2015-12-01 1049.9630  
## 97 Bike Racks 2016-01-01 2029.5000  
## 98 Bike Stands 2016-01-01 1192.4000  
## 99 Bottles and Cages 2016-01-01 145.6415  
## 100 Caps 2016-01-01 178.0000  
## 101 Cleaners 2016-01-01 136.1091  
## 102 Fenders 2016-01-01 428.4917  
## 103 Gloves 2016-01-01 480.1692  
## 104 Helmets 2016-01-01 663.2201  
## 105 Hydration Packs 2016-01-01 1156.2286  
## 106 Jerseys 2016-01-01 958.5581  
## 107 Mountain Bikes 2016-01-01 1929.6778  
## 108 Road Bikes 2016-01-01 1323.4167  
## 109 Shorts 2016-01-01 1348.6579  
## 110 Socks 2016-01-01 183.4667  
## 111 Tires and Tubes 2016-01-01 274.3432  
## 112 Touring Bikes 2016-01-01 1902.2414  
## 113 Vests 2016-01-01 1259.2800  
## 114 Bike Racks 2016-02-01 1063.7500  
## 115 Bike Stands 2016-02-01 1090.6250  
## 116 Bottles and Cages 2016-02-01 138.1432  
## 117 Caps 2016-02-01 175.4098  
## 118 Cleaners 2016-02-01 169.6200  
## 119 Fenders 2016-02-01 470.0118  
## 120 Gloves 2016-02-01 419.5410  
## 121 Helmets 2016-02-01 700.1620  
## 122 Hydration Packs 2016-02-01 1165.6786  
## 123 Jerseys 2016-02-01 1007.4130  
## 124 Mountain Bikes 2016-02-01 1943.6766  
## 125 Road Bikes 2016-02-01 1292.7500  
## 126 Shorts 2016-02-01 1126.6364  
## 127 Socks 2016-02-01 187.3846  
## 128 Tires and Tubes 2016-02-01 281.2258  
## 129 Touring Bikes 2016-02-01 1922.9630  
## 130 Vests 2016-02-01 1086.6000  
## 131 Bike Racks 2016-03-01 834.0000  
## 132 Bike Stands 2016-03-01 1095.6250  
## 133 Bottles and Cages 2016-03-01 136.9149  
## 134 Caps 2016-03-01 168.4514  
## 135 Cleaners 2016-03-01 152.8286  
## 136 Fenders 2016-03-01 444.4912  
## 137 Gloves 2016-03-01 474.2258  
## 138 Helmets 2016-03-01 707.3537  
## 139 Hydration Packs 2016-03-01 1106.5532  
## 140 Jerseys 2016-03-01 953.9818  
## 141 Mountain Bikes 2016-03-01 1932.7067  
## 142 Road Bikes 2016-03-01 1285.2396  
## 143 Shorts 2016-03-01 1242.5610  
## 144 Socks 2016-03-01 173.1212  
## 145 Tires and Tubes 2016-03-01 253.1853  
## 146 Touring Bikes 2016-03-01 1945.9426  
## 147 Vests 2016-03-01 1468.2500  
## 148 Bike Racks 2016-04-01 988.5833  
## 149 Bike Stands 2016-04-01 1232.8333  
## 150 Bottles and Cages 2016-04-01 141.2054  
## 151 Caps 2016-04-01 178.3846  
## 152 Cleaners 2016-04-01 168.6000  
## 153 Fenders 2016-04-01 418.7455  
## 154 Gloves 2016-04-01 506.7042  
## 155 Helmets 2016-04-01 692.3932  
## 156 Hydration Packs 2016-04-01 1027.0455  
## 157 Jerseys 2016-04-01 959.8415  
## 158 Mountain Bikes 2016-04-01 1918.8564  
## 159 Road Bikes 2016-04-01 1249.5780  
## 160 Shorts 2016-04-01 1102.9388  
## 161 Socks 2016-04-01 172.6286  
## 162 Tires and Tubes 2016-04-01 262.3077  
## 163 Touring Bikes 2016-04-01 1860.5652  
## 164 Vests 2016-04-01 1252.5882  
## 165 Bike Racks 2016-05-01 1324.6429  
## 166 Bike Stands 2016-05-01 1076.7143  
## 167 Bottles and Cages 2016-05-01 143.0844  
## 168 Caps 2016-05-01 178.6496  
## 169 Cleaners 2016-05-01 157.8182  
## 170 Fenders 2016-05-01 402.9565  
## 171 Gloves 2016-05-01 479.8737  
## 172 Helmets 2016-05-01 667.9455  
## 173 Hydration Packs 2016-05-01 1109.6364  
## 174 Jerseys 2016-05-01 1040.3721  
## 175 Mountain Bikes 2016-05-01 1977.5889  
## 176 Road Bikes 2016-05-01 1274.9141  
## 177 Shorts 2016-05-01 1403.6957  
## 178 Socks 2016-05-01 220.6667  
## 179 Tires and Tubes 2016-05-01 260.7154  
## 180 Touring Bikes 2016-05-01 1937.6275  
## 181 Vests 2016-05-01 1143.9310  
## 182 Bike Racks 2016-06-01 1728.9000  
## 183 Bike Stands 2016-06-01 1018.8571  
## 184 Bottles and Cages 2016-06-01 145.5719  
## 185 Caps 2016-06-01 174.9701  
## 186 Cleaners 2016-06-01 163.1296  
## 187 Fenders 2016-06-01 436.2029  
## 188 Gloves 2016-06-01 493.6517  
## 189 Helmets 2016-06-01 693.9212  
## 190 Hydration Packs 2016-06-01 1010.0227  
## 191 Jerseys 2016-06-01 961.2069  
## 192 Mountain Bikes 2016-06-01 2033.0296  
## 193 Road Bikes 2016-06-01 1271.9203  
## 194 Shorts 2016-06-01 1311.1400  
## 195 Socks 2016-06-01 170.2500  
## 196 Tires and Tubes 2016-06-01 266.9530  
## 197 Touring Bikes 2016-06-01 1875.1071  
## 198 Vests 2016-06-01 1192.4500  
## 199 Bike Racks 2016-07-01 2275.3750  
## 200 Bike Stands 2016-07-01 694.7500  
## 201 Bottles and Cages 2016-07-01 124.2745  
## 202 Caps 2016-07-01 166.2623  
## 203 Cleaners 2016-07-01 130.2174  
## 204 Fenders 2016-07-01 443.7895  
## 205 Gloves 2016-07-01 449.9730  
## 206 Helmets 2016-07-01 645.5220  
## 207 Hydration Packs 2016-07-01 1049.0714  
## 208 Jerseys 2016-07-01 874.7632  
## 209 Shorts 2016-07-01 1191.2333  
## 210 Socks 2016-07-01 187.0000  
## 211 Tires and Tubes 2016-07-01 251.5055  
## 212 Vests 2016-07-01 835.5556

## Data Visualization:

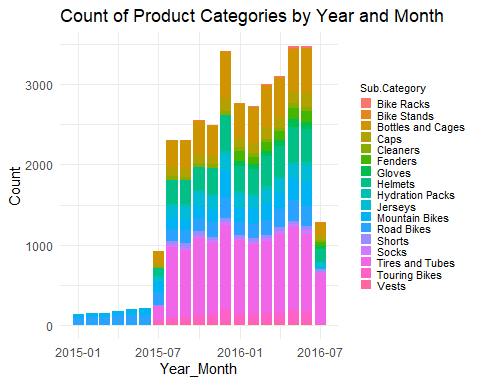
Let’s check for insights from Product category and Year\_Month.

ggplot(sales, aes(x = Year\_Month, fill = Product.Category)) +  
 geom\_bar() +  
 labs(title = "Count of Product Categories by Year and Month", x = "Year\_Month", y = "Count") +  
 theme\_minimal() +  
 theme(legend.position = "top")



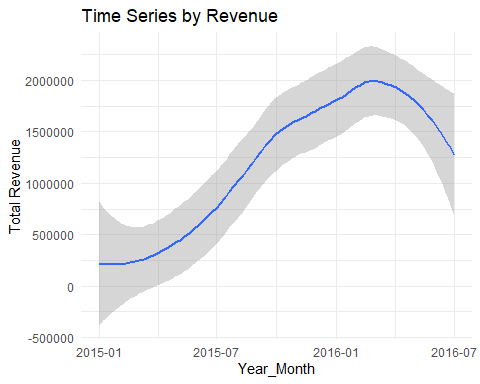
Let’s check for insights from Sub category and Year\_Month.

ggplot(sales, aes(x = Year\_Month, fill = Sub.Category)) +  
 geom\_bar() +  
 labs(title = "Count of Product Categories by Year and Month", x = "Year\_Month", y = "Count") +  
 theme\_minimal() +  
 theme(legend.text = element\_text(size = 8),   
 legend.title = element\_text(size = 8),  
 legend.key.size = unit(0.3, "cm"))



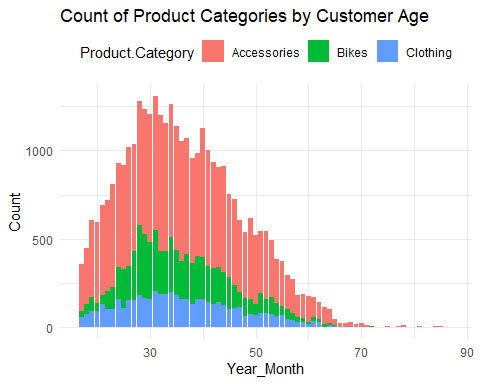
Let’s see in which year\_month most of the revenue is generated.

total\_revenue <- sales %>%  
 group\_by(Year\_Month) %>%  
 summarise(Total\_Revenue = sum(Revenue, na.rm = TRUE))  
ggplot(total\_revenue, aes(x = Year\_Month, y = Total\_Revenue)) +  
 geom\_smooth(method = 'loess', formula = 'y ~ x') +  
 labs(title = "Time Series by Revenue", x = "Year\_Month", y = "Total Revenue") +  
 theme\_minimal()



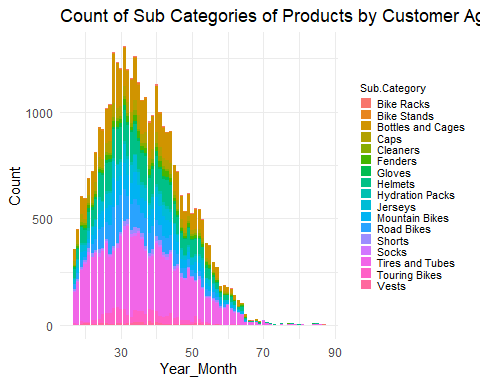
Let’s see which age group people choose which product category.

ggplot(sales, aes(x = Customer.Age, fill = Product.Category)) +  
 geom\_bar() +  
 labs(title = "Count of Product Categories by Customer Age", x = "Year\_Month", y = "Count") +  
 theme\_minimal() +  
 theme(legend.position = "top")



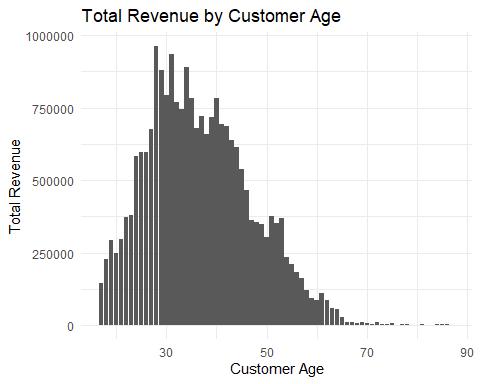
Let’s see which age group people choose which sub category.

ggplot(sales, aes(x = Customer.Age, fill = Sub.Category)) +  
 geom\_bar() +  
 labs(title = "Count of Sub Categories of Products by Customer Age", x = "Year\_Month", y = "Count") +  
 theme\_minimal() +  
 theme(legend.text = element\_text(size = 8),   
 legend.title = element\_text(size = 8),  
 legend.key.size = unit(0.3, "cm"))



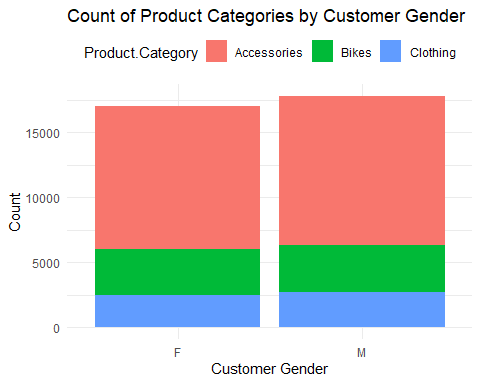
Let’s see from which group of people most of the revenue was generated.

total\_revenue <- aggregate(Revenue ~ Customer.Age, data = sales, sum)  
ggplot(total\_revenue, aes(x = Customer.Age, y = Revenue)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Total Revenue by Customer Age", x = "Customer Age", y = "Total Revenue") +  
 theme\_minimal()



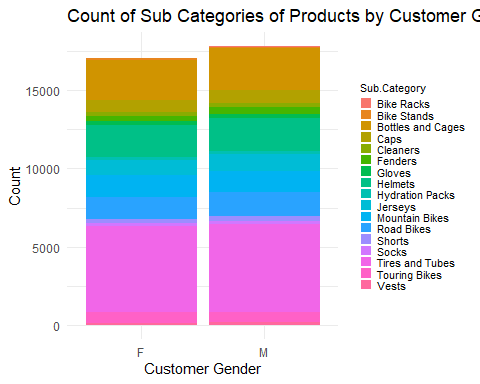
Let’s see which gender of the customer choose which product category.

ggplot(sales, aes(x = Customer.Gender, fill = Product.Category)) +  
 geom\_bar() +  
 labs(title = "Count of Product Categories by Customer Gender", x = "Customer Gender", y = "Count") +  
 theme\_minimal() +  
 theme(legend.position = "top")



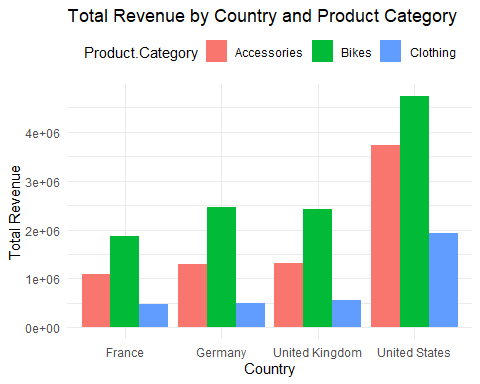
Let’s see which gender of the customer choose which sub category.

ggplot(sales, aes(x = Customer.Gender, fill = Sub.Category)) +  
 geom\_bar() +  
 labs(title = "Count of Sub Categories of Products by Customer Gender", x = "Customer Gender", y = "Count") +  
 theme\_minimal() +  
 theme(legend.text = element\_text(size = 8),   
 legend.title = element\_text(size = 8),  
 legend.key.size = unit(0.3, "cm"))



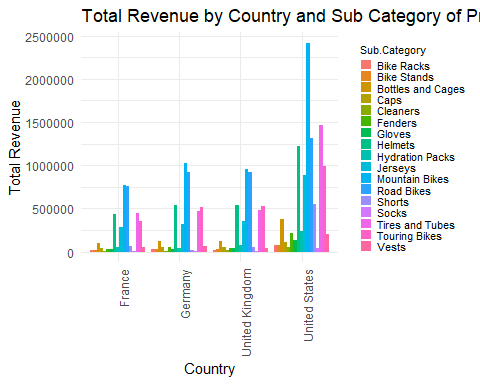
Let’s see which country choose which product category and have the most revenue generated from it.

total\_revenue <- sales %>%  
 group\_by(Country, Product.Category) %>%  
 summarise(Total\_Revenue = sum(Revenue, na.rm = TRUE), .groups = "drop")  
ggplot(total\_revenue, aes(x = Country, y = Total\_Revenue, fill = Product.Category)) +  
 geom\_col(position = "dodge") +  
 labs(title = "Total Revenue by Country and Product Category",  
 x = "Country",  
 y = "Total Revenue") +  
 theme\_minimal() +  
 theme(legend.position = "top")



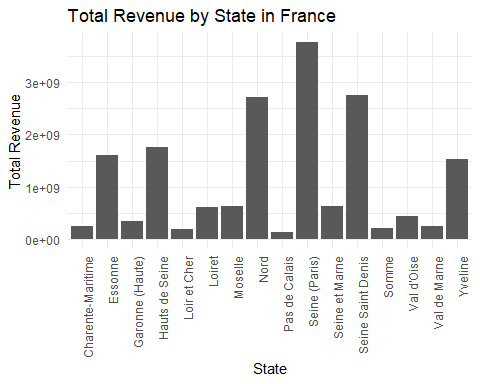
Now let’s check for the same insights but with sub category.

total\_revenue <- sales %>%  
 group\_by(Country, Sub.Category) %>%  
 summarise(Total\_Revenue = sum(Revenue, na.rm = TRUE), .groups = "drop")  
ggplot(total\_revenue, aes(x = Country, y = Total\_Revenue, fill = Sub.Category)) +  
 geom\_col(position = "dodge") +  
 labs(title = "Total Revenue by Country and Sub Category of Products",  
 x = "Country",  
 y = "Total Revenue") +  
 theme\_minimal() +  
 theme(legend.text = element\_text(size = 8),   
 legend.title = element\_text(size = 8),  
 legend.key.size = unit(0.3, "cm")) +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



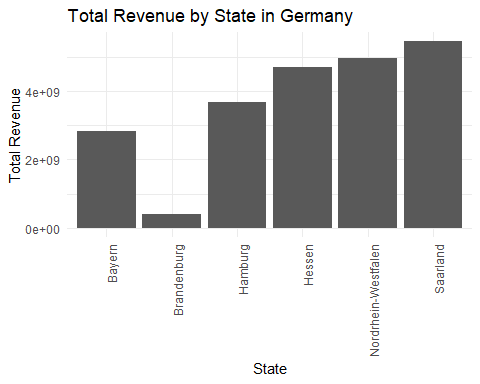
Let’s see which state of the country France got the revenue generated.

filtered\_sales <- subset(sales, Country == "France")  
ggplot(filtered\_sales, aes(x = State, y = sum(Revenue))) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Total Revenue by State in France", x = "State", y = "Total Revenue") +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



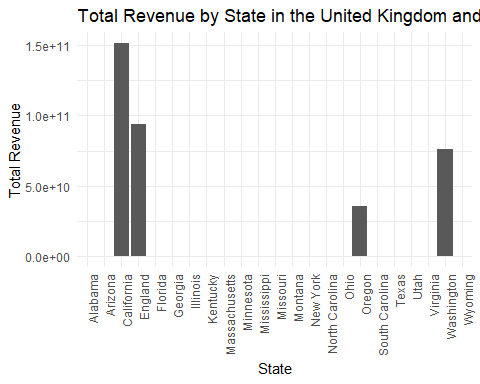
Let’s check for the same insights for the states of Germany.

filtered\_sales <- subset(sales, Country == "Germany")  
ggplot(filtered\_sales, aes(x = State, y = sum(Revenue))) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Total Revenue by State in Germany", x = "State", y = "Total Revenue") +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



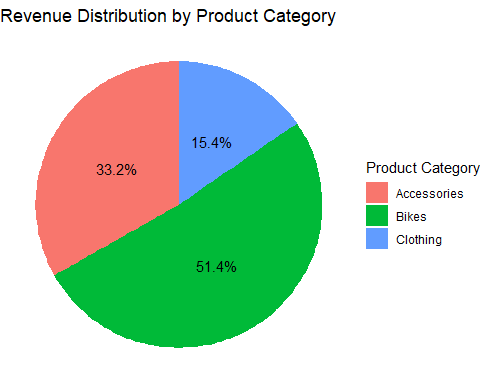
Let’s check for the same insights but for both UK and USA because there is only one state from UK.

filtered\_sales <- subset(sales, Country %in% c("United Kingdom", "United States"))  
ggplot(filtered\_sales, aes(x = State, y = sum(Revenue))) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Total Revenue by State in the United Kingdom and United States", x = "State", y = "Total Revenue") +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



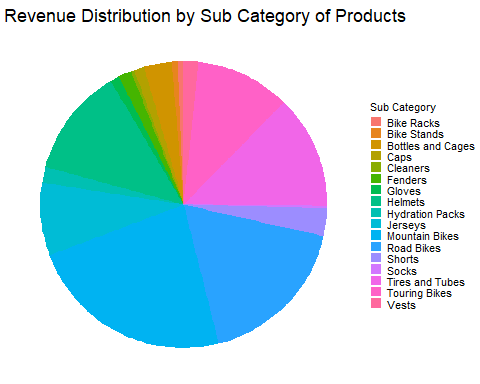
Let’s check for revenue distribution of prouct category in percentage values.

total\_revenue <- sales %>%  
 group\_by(Product.Category) %>%  
 summarise(Total\_Revenue = sum(Revenue, na.rm = TRUE))  
total\_revenue$Percentage <- total\_revenue$Total\_Revenue / sum(total\_revenue$Total\_Revenue) \* 100  
ggplot(total\_revenue, aes(x = "", y = Total\_Revenue, fill = Product.Category)) +  
 geom\_bar(stat = "identity") +  
 coord\_polar(theta = "y") +  
 geom\_text(aes(label = paste0(round(Percentage, 1), "%")), position = position\_stack(vjust = 0.5)) +  
 labs(title = "Revenue Distribution by Product Category", fill = "Product Category") +  
 theme\_void()



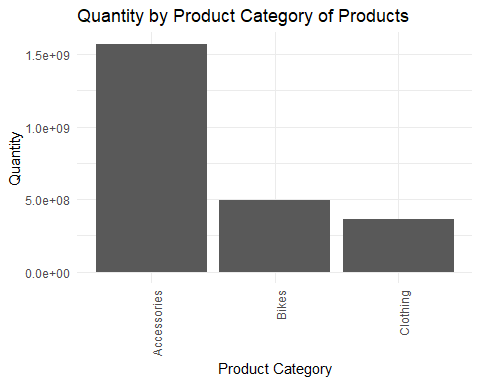
Let’s check for same insights but for sub category.

total\_revenue <- sales %>%  
 group\_by(Sub.Category) %>%  
 summarise(Total\_Revenue = sum(Revenue, na.rm = TRUE))  
total\_revenue$Percentage <- total\_revenue$Total\_Revenue / sum(total\_revenue$Total\_Revenue) \* 100  
ggplot(total\_revenue, aes(x = "", y = Total\_Revenue, fill = Sub.Category)) +  
 geom\_bar(stat = "identity") +  
 coord\_polar(theta = "y") +  
 labs(title = "Revenue Distribution by Sub Category of Products", fill = "Sub Category") +  
 theme\_void() +  
 theme(legend.text = element\_text(size = 8),   
 legend.title = element\_text(size = 8),  
 legend.key.size = unit(0.3, "cm"))



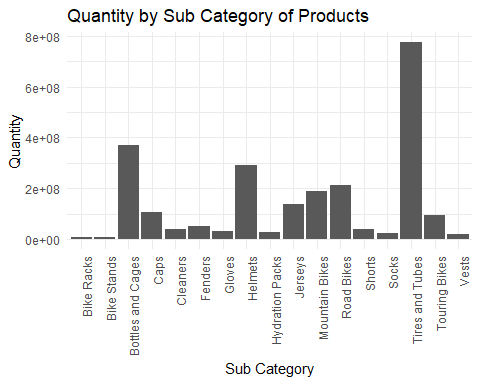
Let’s check which product category was should in most quantity.

ggplot(sales, aes(x = Product.Category, y = sum(Quantity))) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Quantity by Product Category of Products", x = "Product Category", y = "Quantity") +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



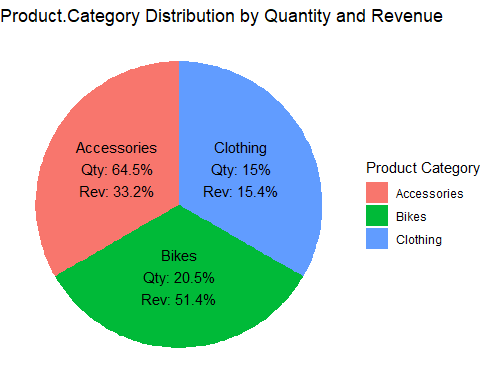
Let’s check for same insights but for sub category.

ggplot(sales, aes(x = Sub.Category, y = sum(Quantity))) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Quantity by Sub Category of Products", x = "Sub Category", y = "Quantity") +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



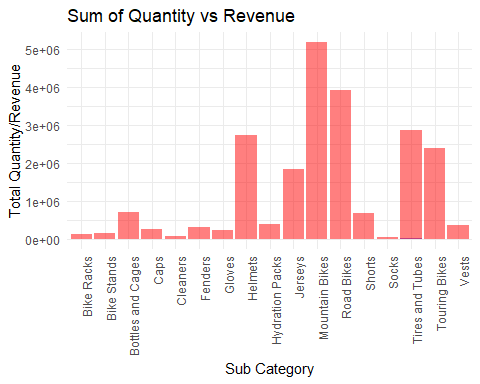
Let’s check for product category distribution by quantity and revenue.

total\_data <- sales %>%  
 group\_by(Product.Category) %>%  
 summarise(Total\_Quantity = sum(Quantity, na.rm = TRUE),  
 Total\_Revenue = sum(Revenue, na.rm = TRUE))  
total\_data$Percentage\_Quantity <- total\_data$Total\_Quantity / sum(total\_data$Total\_Quantity) \* 100  
total\_data$Percentage\_Revenue <- total\_data$Total\_Revenue / sum(total\_data$Total\_Revenue) \* 100  
ggplot(total\_data, aes(x = "", y = 1, fill = Product.Category)) +  
 geom\_bar(stat = "identity", width = 1) +  
 geom\_text(aes(label = paste0(Product.Category, "\n",  
 "Qty: ", round(Percentage\_Quantity, 1), "%\n",  
 "Rev: ", round(Percentage\_Revenue, 1), "%")),  
 position = position\_stack(vjust = 0.5),  
 size = 4) +  
 coord\_polar(theta = "y") +  
 labs(title = "Product.Category Distribution by Quantity and Revenue",  
 fill = "Product Category") +  
 theme\_void()



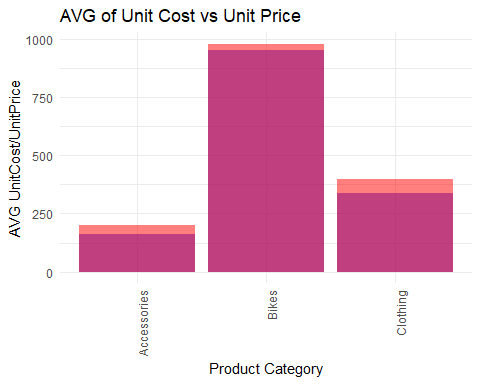
Let’s check for same insights but for sub category.

total\_data <- sales %>%  
 group\_by(Sub.Category) %>%  
 summarise(Total\_Quantity = sum(Quantity, na.rm = TRUE),  
 Total\_Revenue = sum(Revenue, na.rm = TRUE))  
ggplot(total\_data, aes(x = Sub.Category)) +  
 geom\_col(aes(y = Total\_Quantity), fill = "blue", alpha = 0.5) +  
 geom\_col(aes(y = Total\_Revenue), fill = "red", alpha = 0.5) +  
 labs(title = "Sum of Quantity vs Revenue",  
 x = "Sub Category",  
 y = "Total Quantity/Revenue") +  
 theme\_minimal() +  
 theme(legend.text = element\_text(size = 8),   
 legend.title = element\_text(size = 8),  
 axis.text.x = element\_text(angle = 90, hjust = 1))



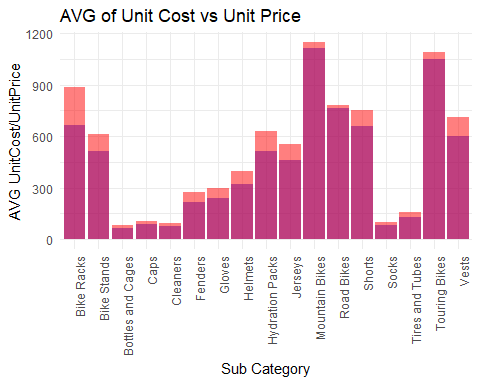
Let’s check which product category cost less and sold for more amount.

total\_data <- sales %>%  
 group\_by(Product.Category) %>%  
 summarise(Mean\_Cost = mean(Unit.Cost, na.rm = TRUE),  
 Mean\_Price = mean(Unit.Price, na.rm = TRUE))  
ggplot(total\_data, aes(x = Product.Category)) +  
 geom\_col(aes(y = Mean\_Cost), fill = "blue", alpha = 0.5) +  
 geom\_col(aes(y = Mean\_Price), fill = "red", alpha = 0.5) +  
 labs(title = "AVG of Unit Cost vs Unit Price",  
 x = "Product Category",  
 y = "AVG UnitCost/UnitPrice") +  
 theme\_minimal() +  
 theme(legend.text = element\_text(size = 8),   
 legend.title = element\_text(size = 8),  
 axis.text.x = element\_text(angle = 90, hjust = 1))



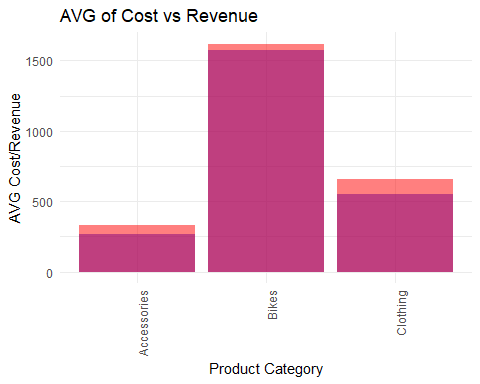
Let’s check for same insights but for sub category.

total\_data <- sales %>%  
 group\_by(Sub.Category) %>%  
 summarise(Mean\_Cost = mean(Unit.Cost, na.rm = TRUE),  
 Mean\_Price = mean(Unit.Price, na.rm = TRUE))  
ggplot(total\_data, aes(x = Sub.Category)) +  
 geom\_col(aes(y = Mean\_Cost), fill = "blue", alpha = 0.5) +  
 geom\_col(aes(y = Mean\_Price), fill = "red", alpha = 0.5) +  
 labs(title = "AVG of Unit Cost vs Unit Price",  
 x = "Sub Category",  
 y = "AVG UnitCost/UnitPrice") +  
 theme\_minimal() +  
 theme(legend.text = element\_text(size = 8),   
 legend.title = element\_text(size = 8),  
 axis.text.x = element\_text(angle = 90, hjust = 1))



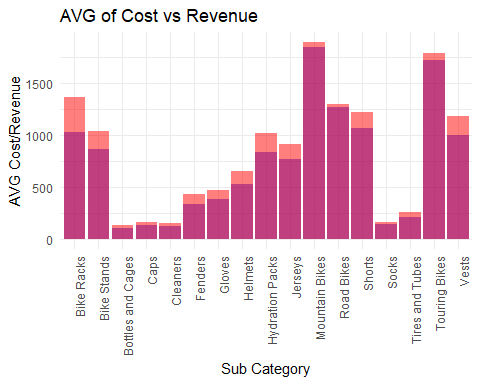
Let’s compare the average of cost and revenue by product category.

total\_data <- sales %>%  
 group\_by(Product.Category) %>%  
 summarise(Mean\_Cost = mean(Cost, na.rm = TRUE),  
 Mean\_Revenue = mean(Revenue, na.rm = TRUE))  
ggplot(total\_data, aes(x = Product.Category)) +  
 geom\_col(aes(y = Mean\_Cost), fill = "blue", alpha = 0.5) +  
 geom\_col(aes(y = Mean\_Revenue), fill = "red", alpha = 0.5) +  
 labs(title = "AVG of Cost vs Revenue",  
 x = "Product Category",  
 y = "AVG Cost/Revenue") +  
 theme\_minimal() +  
 theme(legend.text = element\_text(size = 8),   
 legend.title = element\_text(size = 8),  
 axis.text.x = element\_text(angle = 90, hjust = 1))



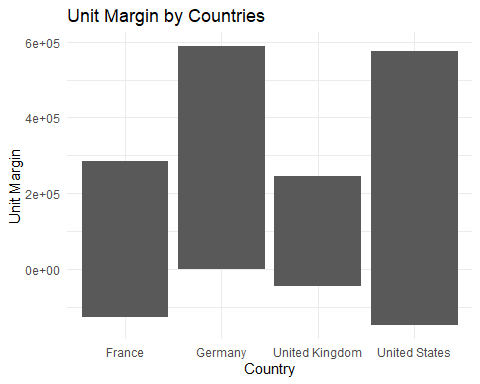
Let’s check for the same insights but for sub category.

total\_data <- sales %>%  
 group\_by(Sub.Category) %>%  
 summarise(Mean\_Cost = mean(Cost, na.rm = TRUE),  
 Mean\_Revenue = mean(Revenue, na.rm = TRUE))  
ggplot(total\_data, aes(x = Sub.Category)) +  
 geom\_col(aes(y = Mean\_Cost), fill = "blue", alpha = 0.5) +  
 geom\_col(aes(y = Mean\_Revenue), fill = "red", alpha = 0.5) +  
 labs(title = "AVG of Cost vs Revenue",  
 x = "Sub Category",  
 y = "AVG Cost/Revenue") +  
 theme\_minimal() +  
 theme(legend.text = element\_text(size = 8),   
 legend.title = element\_text(size = 8),  
 axis.text.x = element\_text(angle = 90, hjust = 1))



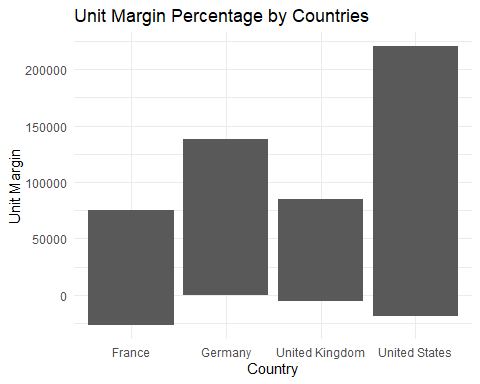
Let’s check for the country with most unit margin.

ggplot(sales, aes(x = Country, y = Unit\_Margin)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Unit Margin by Countries", x = "Country", y = "Unit Margin") +  
 theme\_minimal()



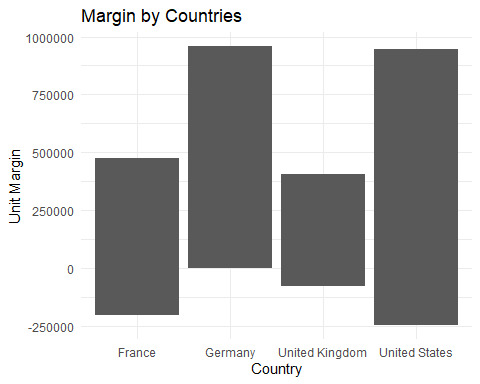
Let’s check for the country with most unit margin percentage.

ggplot(sales, aes(x = Country, y = Unit\_Margin\_percent)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Unit Margin Percentage by Countries", x = "Country", y = "Unit Margin") +  
 theme\_minimal()

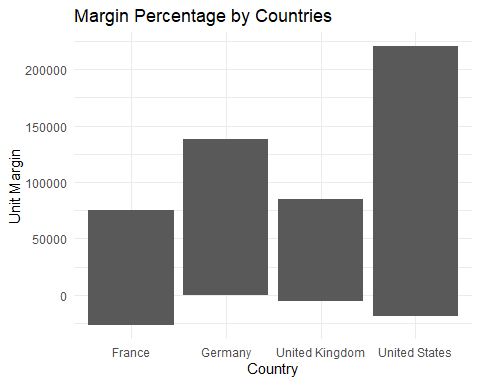


Let’s check for the country with most margin and margin percentage.

ggplot(sales, aes(x = Country, y = Margin)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Margin by Countries", x = "Country", y = "Unit Margin") +  
 theme\_minimal()

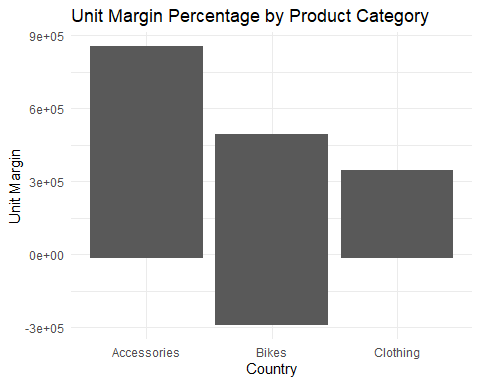


ggplot(sales, aes(x = Country, y = Margin\_percent)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Margin Percentage by Countries", x = "Country", y = "Unit Margin") +  
 theme\_minimal()

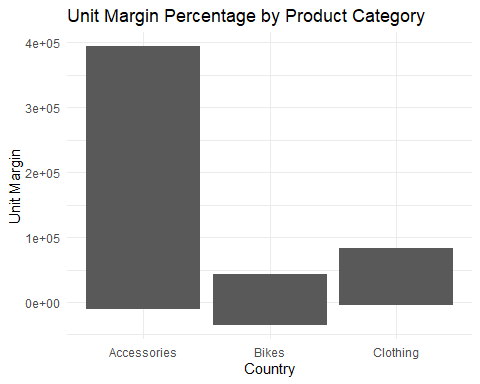


Let’s check which product category has the most unit margin, unit margin percentage, margine and margine percentage.

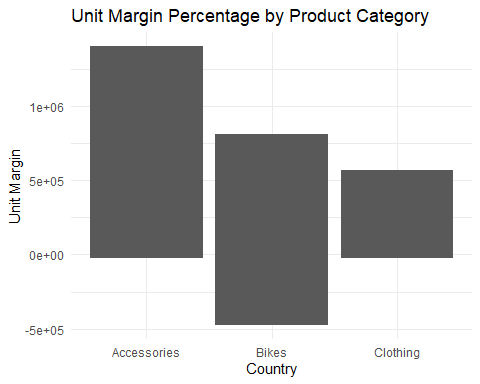
ggplot(sales, aes(x = Product.Category, y = Unit\_Margin)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Unit Margin Percentage by Product Category", x = "Country", y = "Unit Margin") +  
 theme\_minimal()



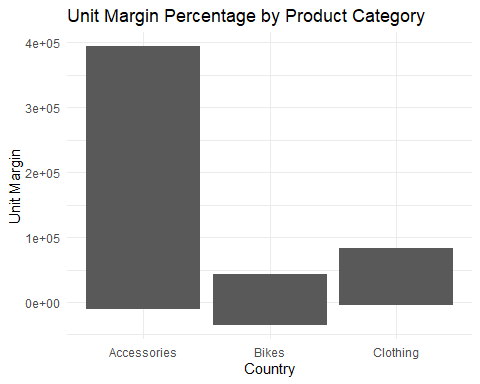
ggplot(sales, aes(x = Product.Category, y = Unit\_Margin\_percent)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Unit Margin Percentage by Product Category", x = "Country", y = "Unit Margin") +  
 theme\_minimal()



ggplot(sales, aes(x = Product.Category, y = Margin)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Unit Margin Percentage by Product Category", x = "Country", y = "Unit Margin") +  
 theme\_minimal()

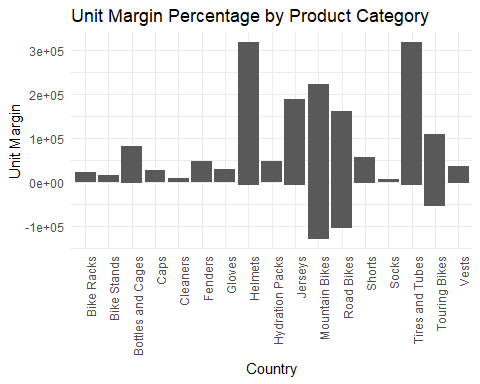


ggplot(sales, aes(x = Product.Category, y = Margin\_percent)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Unit Margin Percentage by Product Category", x = "Country", y = "Unit Margin") +  
 theme\_minimal()

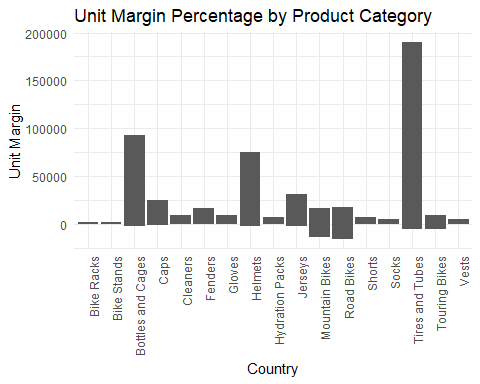


Let’s check for the same insights for all four column but of sub category.

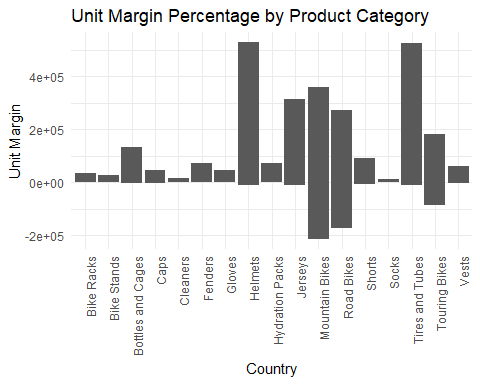
ggplot(sales, aes(x = Sub.Category, y = Unit\_Margin)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Unit Margin Percentage by Product Category", x = "Country", y = "Unit Margin") +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



ggplot(sales, aes(x = Sub.Category, y = Unit\_Margin\_percent)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Unit Margin Percentage by Product Category", x = "Country", y = "Unit Margin") +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



ggplot(sales, aes(x = Sub.Category, y = Margin)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Unit Margin Percentage by Product Category", x = "Country", y = "Unit Margin") +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



ggplot(sales, aes(x = Sub.Category, y = Margin\_percent)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Unit Margin Percentage by Product Category", x = "Country", y = "Unit Margin") +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))

