**Project Report**

**Rushikesh Khankar**

**High value customers identification for an E-Commerce company**

Project 6

DESCRIPTION

**Background of Problem Statement:**  
 A UK-based online retail store has captured the sales data for different products for the period of one year (Nov 2016 to Dec 2017). The organization sells gifts primarily on the online platform. The customers who make a purchase consume directly for themselves. There are small businesses that buy in bulk and sell to other customers through the retail outlet channel.

**Project Objective:**  
Find significant customers for the business who make high purchases of their favourite products. The organization wants to roll out a loyalty program to the high-value customers after identification of segments. Use the clustering methodology to segment customers into groups:

**Domain:** E-commerce

**Dataset Description:**  
This is a transnational dataset that contains all the transactions occurring between Nov-2016 to Dec-2017 for a UK-based online retail store.

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| InvoiceNo | Invoice number (A 6-digit integral number uniquely assigned to each transaction) |
| StockCode | Product (item) code |
| Description | Product (item) name |
| Quantity | The quantities of each product (item) per transaction |
| InvoiceDate | The day when each transaction was generated |
| UnitPrice | Unit price (Product price per unit) |
| CustomerID | Customer number (Unique ID assigned to each customer) |
| Country | Country name (The name of the country where each customer resides) |

**Analysis tasks to be performed:**  
 Use the clustering methodology to segment customers into groups:  
 Use the following clustering algorithms:

1. K means
2. Hierarchical

• Identify the right number of customer segments.  
• Provide the number of customers who are highly valued.  
• Identify the clustering algorithm that gives maximum accuracy and explains robust clusters.  
• If the number of observations is loaded in one of the clusters, break down that cluster further using the clustering algorithm. [ hint: Here loaded means if any cluster has more number of data points as compared to other clusters then split that clusters by increasing the number of clusters and observe, compare the results with previous results.]

**My Code : {Solution}**

library(dplyr)

> library(DataExplorer)

> library(ggplot2)

> library(cluster)

> library(factoextra)

> library(gridExtra)

> library(purrr)

> library(tidyverse)

> data = read.csv("~/Desktop/R/Project-2/Projects for Submission/Ecommerce/Project 2\_Dataset.csv")

> head(data)

InvoiceNo StockCode Description Quantity

1 536365 85123A WHITE HANGING HEART T-LIGHT HOLDER 6

2 536365 71053 WHITE METAL LANTERN 6

3 536365 84406B CREAM CUPID HEARTS COAT HANGER 8

4 536365 84029G KNITTED UNION FLAG HOT WATER BOTTLE 6

5 536365 84029E RED WOOLLY HOTTIE WHITE HEART. 6

6 536365 22752 SET 7 BABUSHKA NESTING BOXES 2

InvoiceDate UnitPrice CustomerID Country X

1 29-Nov-16 2.55 17850 United Kingdom NA

2 29-Nov-16 3.39 17850 United Kingdom NA

3 29-Nov-16 2.75 17850 United Kingdom NA

4 29-Nov-16 3.39 17850 United Kingdom NA

5 29-Nov-16 3.39 17850 United Kingdom NA

6 29-Nov-16 7.65 17850 United Kingdom NA

> str(data)

'data.frame': 541909 obs. of 9 variables:

$ InvoiceNo : chr "536365" "536365" "536365" "536365" ...

$ StockCode : chr "85123A" "71053" "84406B" "84029G" ...

$ Description: chr "WHITE HANGING HEART T-LIGHT HOLDER" "WHITE METAL LANTERN" "CREAM CUPID HEARTS COAT HANGER" "KNITTED UNION FLAG HOT WATER BOTTLE" ...

$ Quantity : int 6 6 8 6 6 2 6 6 6 32 ...

$ InvoiceDate: chr "29-Nov-16" "29-Nov-16" "29-Nov-16" "29-Nov-16" ...

$ UnitPrice : num 2.55 3.39 2.75 3.39 3.39 7.65 4.25 1.85 1.85 1.69 ...

$ CustomerID : int 17850 17850 17850 17850 17850 17850 17850 17850 17850 13047 ...

$ Country : chr "United Kingdom" "United Kingdom" "United Kingdom" "United Kingdom" ...

$ X : logi NA NA NA NA NA NA ...

> colSums(is.na(data))

InvoiceNo StockCode Description Quantity InvoiceDate UnitPrice

0 0 0 0 0 0

CustomerID Country X

135080 0 541909

> #Remove column 'X' which has all the rows wit NA

> data <-select(data, -c(X))

> #Remove rows of CustomerIDvwhich has NA data

> data<-na.omit(data)

> head(data)

InvoiceNo StockCode Description Quantity

1 536365 85123A WHITE HANGING HEART T-LIGHT HOLDER 6

2 536365 71053 WHITE METAL LANTERN 6

3 536365 84406B CREAM CUPID HEARTS COAT HANGER 8

4 536365 84029G KNITTED UNION FLAG HOT WATER BOTTLE 6

5 536365 84029E RED WOOLLY HOTTIE WHITE HEART. 6

6 536365 22752 SET 7 BABUSHKA NESTING BOXES 2

InvoiceDate UnitPrice CustomerID Country

1 29-Nov-16 2.55 17850 United Kingdom

2 29-Nov-16 3.39 17850 United Kingdom

3 29-Nov-16 2.75 17850 United Kingdom

4 29-Nov-16 3.39 17850 United Kingdom

5 29-Nov-16 3.39 17850 United Kingdom

6 29-Nov-16 7.65 17850 United Kingdom

> data$InvoiceDate <- as.Date(data$InvoiceDate, "%d-%b-%y")

> View(data)

> #Computing the line total

> data <- data %>% mutate(LineTotal = Quantity \* UnitPrice)

> head(data)

InvoiceNo StockCode Description Quantity

1 536365 85123A WHITE HANGING HEART T-LIGHT HOLDER 6

2 536365 71053 WHITE METAL LANTERN 6

3 536365 84406B CREAM CUPID HEARTS COAT HANGER 8

4 536365 84029G KNITTED UNION FLAG HOT WATER BOTTLE 6

5 536365 84029E RED WOOLLY HOTTIE WHITE HEART. 6

6 536365 22752 SET 7 BABUSHKA NESTING BOXES 2

InvoiceDate UnitPrice CustomerID Country LineTotal

1 2016-11-29 2.55 17850 United Kingdom 15.30

2 2016-11-29 3.39 17850 United Kingdom 20.34

3 2016-11-29 2.75 17850 United Kingdom 22.00

4 2016-11-29 3.39 17850 United Kingdom 20.34

5 2016-11-29 3.39 17850 United Kingdom 20.34

6 2016-11-29 7.65 17850 United Kingdom 15.30

> #Country summary

> countrySummary <- data %>% group\_by(Country) %>% summarise(revenue = sum(LineTotal), transactions = n\_distinct(InvoiceNo)) %>% mutate(aveOrdVal = (round((revenue/transactions), 2))) %>% ungroup() %>% arrange(desc(revenue))

> head(countrySummary)

# A tibble: 6 × 4

Country revenue transactions aveOrdVal

<chr> <dbl> <int> <dbl>

1 United Kingdom 6767873. 19857 341.

2 Netherlands 284662. 101 2818.

3 EIRE 250285. 319 785.

4 Germany 221698. 603 368.

5 France 196713. 458 430.

6 Australia 137077. 69 1987.

> #Total revenue generated and total item purchased by each customer

> customerData <- data %>% group\_by(CustomerID, Country) %>% summarise(TotalRevenue = sum(LineTotal), TotalItemsSold = sum(Quantity))

`summarise()` has grouped output by 'CustomerID'. You can override using the

`.groups` argument.

> head(customerData)

# A tibble: 6 × 4

# Groups: CustomerID [6]

CustomerID Country TotalRevenue TotalItemsSold

<int> <chr> <dbl> <int>

1 12346 United Kingdom 0 0

2 12347 Iceland 4310 2458

3 12348 Finland 1797. 2341

4 12349 Italy 1758. 631

5 12350 Norway 334. 197

6 12352 Norway 1545. 470

> n\_occur <- data.frame(table(customerData$CustomerID))

> single\_ResidentsIds = (n\_occur[n\_occur$Freq == 1,])$Var

> customerData <- subset(customerData, (customerData$CustomerID %in% single\_ResidentsIds))

> remove(n\_occur)

> remove(single\_ResidentsIds)

> set.seed(123) #To ensure the same result every time

> #Categorical variable - Country

> length(unique(data$Country))

[1] 37

> #Removing CustomerId and COuntry Columns

> customerData <- customerData[-c(1:2)]

> #Visualizing outliers

> boxplot(customerData)$out

[1] 4310.00 6207.67 6245.53 5154.58 4008.62 11056.93 123725.45

[8] 7877.20 5059.32 13375.87 7829.89 4896.66 5005.46 4067.29

[15] 8933.52 18740.92 6229.48 7116.47 6546.58 13117.01 5590.86

[22] 4495.27 5417.93 4206.27 3723.87 4485.72 4106.82 3852.50

[29] 13309.04 3652.25 11990.96 3781.74 9114.94 4064.99 7187.34

[36] 9861.38 3683.13 13612.07 6388.35 4478.53 5953.25 3669.92

[43] 3982.72 3817.08 4863.88 3743.18 17588.26 13677.59 12288.22

[50] 8221.09 4873.81 4867.99 6814.24 9294.10 4241.63 4708.86

[57] 5713.28 3947.70 18793.41 9120.39 4196.01 29072.10 3868.20

[64] 4366.78 21024.01 4227.70 6989.63 3868.70 6977.04 6748.40

[71] 5583.62 3876.40 3986.22 16293.10 16389.74 33462.81 11581.80

[78] 4143.02 4734.26 3979.04 10930.26 7092.06 6854.51 9797.92

[85] 5613.43 4786.99 7272.75 6912.00 5684.61 3717.82 12432.28

[92] 27964.48 3595.48 57385.88 8689.39 7741.47 5893.32 28658.88

[99] 6336.29 10510.00 3866.33 3605.08 5055.79 5725.47 3957.78

[106] 9803.96 5820.16 6083.04 7454.07 3717.67 4404.44 5315.92

[113] 5815.49 10664.34 7072.80 3617.57 10736.48 3626.81 4122.28

[120] 27487.41 5845.44 3794.49 9537.92 5656.75 8910.61 4282.88

[127] 3755.38 5613.08 4149.28 6703.30 5030.60 7792.36 6216.07

[134] 62653.10 6249.11 5145.12 16945.71 25748.35 36351.42 3906.27

[141] 4014.91 8025.02 6389.80 16989.52 4912.41 8986.69 7024.53

[148] 4098.80 3865.26 4582.64 4328.46 14634.64 5639.15 15477.34

[155] 8124.40 6147.40 6135.38 10464.85 4435.79 50415.49 4063.10

[162] 57120.91 5976.79 5669.65 3580.13 4690.31 113384.14 4678.30

[169] 4640.77 4604.31 4660.01 11216.75 12650.08 3957.00 3729.34

[176] 4595.81 3873.30 4819.33 50862.44 4928.74 3717.35 9167.82

[183] 4592.47 4039.98 5811.56 4263.64 3918.90 5818.70 4221.37

[190] 7711.38 4075.35 3986.74 4709.22 11713.85 11341.10 279489.02

[197] 9231.20 26932.34 5107.38 4911.14 9451.54 5807.69 3899.10

[204] 4965.38 10641.60 7839.51 4236.34 7904.28 14197.45 11020.26

[211] 132572.62 8117.42 5842.95 7912.79 7006.58 6180.49 6763.16

[218] 4464.10 4277.66 19786.44 7629.38 5064.01 54228.74 9623.35

[225] 4410.14 11117.05 6530.04 5303.97 4684.24 18573.68 4670.08

[232] 15802.14 7521.17 3861.00 5756.89 6657.36 10327.11 7803.77

[239] 4596.51 3610.31 3774.10 3787.09 5028.12 59419.34 8234.80

[246] 3631.89 5927.86 3974.37 6945.02 6568.72 12396.46 5541.46

[253] 14520.08 3967.78 4954.84 4758.20 4109.97 6745.36 10360.04

[260] 12433.34 6423.60 5043.96 5178.96 4788.77 21535.90 51823.72

[267] 4834.20 3848.55 5452.17 3724.77 33350.76 3650.87 16173.67

[274] 3797.79 4330.67 5126.34 6102.26 4389.81 4199.85 12393.70

[281] 33366.25 50992.61 8727.61 6287.77 14305.66 3741.98 5352.97

[288] 3770.60 10217.48 6617.65 3991.94 6841.24 4913.32 21086.30

[295] 3693.81 3598.26 3665.62 5203.51 5423.15 5305.83 3844.22

[302] 26626.80 6675.71 4061.24 33805.69 3640.67 9295.67 11895.57

[309] 6748.80 4154.64 5664.57 8257.04 3622.95 4073.28 5450.96

[316] 4379.65 6654.26 3794.52 8568.24 4022.46 8093.59 65892.08

[323] 5040.00 5150.27 13946.13 3645.43 6840.23 4711.21 7076.28

[330] 7170.70 6694.31 5580.60 10454.05 6500.60 15357.78 10183.82

[337] 7066.31 3754.27 6666.83 4604.22 4345.50 4392.42 5461.62

[344] 3989.57 8574.11 5594.78 5116.13 4531.34 3802.81 10380.43

[351] 4892.24 10585.57 3755.65 3744.65 8628.55 8600.73 6204.46

[358] 6612.18 11891.18 4455.73 4801.56 3925.92 20272.43 31300.08

[365] 7330.80 30300.82 6838.34 4200.97 5022.26 17078.45 -4287.63

[372] 187482.17 6100.74 88125.38 4012.56 10736.11 3911.56 5050.77

[379] 3947.87 4509.37 3886.67 4737.13 20098.10 16219.22 5739.46

[386] 4748.45 10097.37 5505.79 5367.80 5288.83 13110.02 5585.49

[393] 3960.63 4627.62 7677.71 3661.07 40340.78 5288.63 26763.34

[400] 5155.66 10158.54 4204.10 4301.22 52750.84 4393.61 4161.14

[407] 6752.14 3864.44 3651.97 3770.12 9065.76 256438.49 7925.32

[414] 5595.77 8393.22 7490.17 5083.06 6315.23 5361.02 5192.10

[421] 7276.90 4314.72 2458.00 2341.00 2708.00 2212.00 2529.00

[428] 5543.00 77242.00 3503.00 11071.00 2918.00 6455.00 7965.00

[435] 4020.00 2363.00 4764.00 2337.00 3620.00 3290.00 2185.00

[442] 3064.00 2671.00 3699.00 2652.00 6677.00 4384.00 4089.00

[449] 5009.00 4279.00 2187.00 3354.00 7404.00 3350.00 2752.00

[456] 2990.00 10899.00 7518.00 5575.00 4903.00 3028.00 2963.00

[463] 3973.00 3832.00 3069.00 4654.00 3143.00 8366.00 5234.00

[470] 24210.00 2262.00 11250.00 6479.00 2385.00 3646.00 5256.00

[477] 9847.00 3804.00 2683.00 4274.00 20915.00 9454.00 23377.00

[484] 4876.00 2363.00 3642.00 2531.00 9203.00 2915.00 5497.00

[491] 4796.00 2217.00 5434.00 17280.00 3707.00 5466.00 5557.00

[498] 19021.00 4313.00 30787.00 2194.00 3943.00 2341.00 15911.00

[505] 4160.00 2594.00 4300.00 2225.00 2418.00 3570.00 2208.00

[512] 2645.00 6172.00 4232.00 3352.00 12540.00 4780.00 4299.00

[519] 3635.00 3509.00 6947.00 4686.00 5200.00 5770.00 2415.00

[526] 2215.00 16119.00 3839.00 2224.00 5460.00 2581.00 4512.00

[533] 2461.00 2876.00 2660.00 2640.00 2573.00 3169.00 4372.00

[540] 2182.00 3637.00 5056.00 61803.00 2304.00 3501.00 3164.00

[547] 7258.00 12804.00 2493.00 23514.00 2281.00 2557.00 3700.00

[554] 2278.00 2801.00 10062.00 2731.00 2307.00 2474.00 2724.00

[561] 4483.00 3439.00 3025.00 2231.00 2668.00 2478.00 5370.00

[568] 3846.00 3741.00 4232.00 4708.00 5374.00 5593.00 3183.00

[575] 12593.00 16335.00 2177.00 6942.00 2455.00 3366.00 3850.00

[582] 57025.00 4395.00 3919.00 3657.00 5783.00 2190.00 7098.00

[589] 2668.00 4306.00 2748.00 58021.00 2406.00 3075.00 4410.00

[596] 4505.00 2509.00 2973.00 2906.00 2339.00 2901.00 2743.00

[603] 2662.00 2777.00 2658.00 5978.00 7998.00 7026.00 196719.00

[610] 5382.00 2498.00 12474.00 3222.00 2482.00 3725.00 3817.00

[617] 2217.00 7406.00 2688.00 5942.00 3634.00 5656.00 6569.00

[624] 4823.00 77180.00 4083.00 2801.00 3295.00 3692.00 4204.00

[631] 2863.00 2668.00 2411.00 9191.00 2786.00 3307.00 28590.00

[638] 5069.00 3733.00 2362.00 3096.00 5991.00 3026.00 3633.00

[645] 2694.00 2242.00 11394.00 6035.00 5648.00 3329.00 2252.00

[652] 3557.00 6084.00 4431.00 2752.00 2940.00 2768.00 37720.00

[659] 6927.00 2924.00 2293.00 4003.00 2998.00 3136.00 6084.00

[666] 3823.00 2348.00 5545.00 3489.00 3105.00 4032.00 4410.00

[673] 4978.00 5493.00 6572.00 2231.00 2724.00 3065.00 2967.00

[680] 9014.00 27660.00 3331.00 2515.00 3337.00 9007.00 10941.00

[687] 2543.00 2506.00 2409.00 2311.00 4725.00 2256.00 5110.00

[694] 13845.00 32148.00 3516.00 8738.00 3072.00 2506.00 5853.00

[701] 3791.00 3008.00 5004.00 3203.00 7559.00 3137.00 2994.00

[708] 2627.00 2621.00 8000.00 2571.00 32184.00 2317.00 3858.00

[715] 32592.00 4233.00 6493.00 2365.00 4595.00 5346.00 2914.00

[722] 2822.00 2410.00 2670.00 3267.00 2367.00 8873.00 2524.00

[729] 2812.00 2368.00 49390.00 4019.00 6211.00 5458.00 4264.00

[736] 4989.00 5769.00 6727.00 4662.00 4280.00 4435.00 5236.00

[743] 2768.00 8470.00 2234.00 10154.00 3825.00 3183.00 3158.00

[750] 2725.00 2408.00 6613.00 2669.00 4537.00 2730.00 5163.00

[757] 2856.00 2850.00 3362.00 11470.00 3856.00 3123.00 5902.00

[764] 2680.00 4777.00 25646.00 7430.00 32324.00 4266.00 2195.00

[771] 3082.00 9435.00 69029.00 3530.00 63012.00 2752.00 2428.00

[778] 5861.00 3095.00 4400.00 4135.00 2552.00 2470.00 3100.00

[785] 2479.00 8278.00 9725.00 2478.00 3833.00 3004.00 2455.00

[792] 3567.00 8439.00 3376.00 3957.00 3316.00 4128.00 22865.00

[799] 19086.00 2706.00 2638.00 5698.00 2516.00 9751.00 27571.00

[806] 2229.00 2571.00 3810.00 2450.00 3098.00 7432.00 3571.00

[813] 4250.00 2518.00 64122.00 4219.00 2877.00 5461.00 2625.00

[820] 3170.00 3839.00 2890.00 3206.00 3459.00 2488.00 7824.00

> boxplot(customerData, horizontal = TRUE)$out

[1] 4310.00 6207.67 6245.53 5154.58 4008.62 11056.93 123725.45

[8] 7877.20 5059.32 13375.87 7829.89 4896.66 5005.46 4067.29

[15] 8933.52 18740.92 6229.48 7116.47 6546.58 13117.01 5590.86

[22] 4495.27 5417.93 4206.27 3723.87 4485.72 4106.82 3852.50

[29] 13309.04 3652.25 11990.96 3781.74 9114.94 4064.99 7187.34

[36] 9861.38 3683.13 13612.07 6388.35 4478.53 5953.25 3669.92

[43] 3982.72 3817.08 4863.88 3743.18 17588.26 13677.59 12288.22

[50] 8221.09 4873.81 4867.99 6814.24 9294.10 4241.63 4708.86

[57] 5713.28 3947.70 18793.41 9120.39 4196.01 29072.10 3868.20

[64] 4366.78 21024.01 4227.70 6989.63 3868.70 6977.04 6748.40

[71] 5583.62 3876.40 3986.22 16293.10 16389.74 33462.81 11581.80

[78] 4143.02 4734.26 3979.04 10930.26 7092.06 6854.51 9797.92

[85] 5613.43 4786.99 7272.75 6912.00 5684.61 3717.82 12432.28

[92] 27964.48 3595.48 57385.88 8689.39 7741.47 5893.32 28658.88

[99] 6336.29 10510.00 3866.33 3605.08 5055.79 5725.47 3957.78

[106] 9803.96 5820.16 6083.04 7454.07 3717.67 4404.44 5315.92

[113] 5815.49 10664.34 7072.80 3617.57 10736.48 3626.81 4122.28

[120] 27487.41 5845.44 3794.49 9537.92 5656.75 8910.61 4282.88

[127] 3755.38 5613.08 4149.28 6703.30 5030.60 7792.36 6216.07

[134] 62653.10 6249.11 5145.12 16945.71 25748.35 36351.42 3906.27

[141] 4014.91 8025.02 6389.80 16989.52 4912.41 8986.69 7024.53

[148] 4098.80 3865.26 4582.64 4328.46 14634.64 5639.15 15477.34

[155] 8124.40 6147.40 6135.38 10464.85 4435.79 50415.49 4063.10

[162] 57120.91 5976.79 5669.65 3580.13 4690.31 113384.14 4678.30

[169] 4640.77 4604.31 4660.01 11216.75 12650.08 3957.00 3729.34

[176] 4595.81 3873.30 4819.33 50862.44 4928.74 3717.35 9167.82

[183] 4592.47 4039.98 5811.56 4263.64 3918.90 5818.70 4221.37

[190] 7711.38 4075.35 3986.74 4709.22 11713.85 11341.10 279489.02

[197] 9231.20 26932.34 5107.38 4911.14 9451.54 5807.69 3899.10

[204] 4965.38 10641.60 7839.51 4236.34 7904.28 14197.45 11020.26

[211] 132572.62 8117.42 5842.95 7912.79 7006.58 6180.49 6763.16

[218] 4464.10 4277.66 19786.44 7629.38 5064.01 54228.74 9623.35

[225] 4410.14 11117.05 6530.04 5303.97 4684.24 18573.68 4670.08

[232] 15802.14 7521.17 3861.00 5756.89 6657.36 10327.11 7803.77

[239] 4596.51 3610.31 3774.10 3787.09 5028.12 59419.34 8234.80

[246] 3631.89 5927.86 3974.37 6945.02 6568.72 12396.46 5541.46

[253] 14520.08 3967.78 4954.84 4758.20 4109.97 6745.36 10360.04

[260] 12433.34 6423.60 5043.96 5178.96 4788.77 21535.90 51823.72

[267] 4834.20 3848.55 5452.17 3724.77 33350.76 3650.87 16173.67

[274] 3797.79 4330.67 5126.34 6102.26 4389.81 4199.85 12393.70

[281] 33366.25 50992.61 8727.61 6287.77 14305.66 3741.98 5352.97

[288] 3770.60 10217.48 6617.65 3991.94 6841.24 4913.32 21086.30

[295] 3693.81 3598.26 3665.62 5203.51 5423.15 5305.83 3844.22

[302] 26626.80 6675.71 4061.24 33805.69 3640.67 9295.67 11895.57

[309] 6748.80 4154.64 5664.57 8257.04 3622.95 4073.28 5450.96

[316] 4379.65 6654.26 3794.52 8568.24 4022.46 8093.59 65892.08

[323] 5040.00 5150.27 13946.13 3645.43 6840.23 4711.21 7076.28

[330] 7170.70 6694.31 5580.60 10454.05 6500.60 15357.78 10183.82

[337] 7066.31 3754.27 6666.83 4604.22 4345.50 4392.42 5461.62

[344] 3989.57 8574.11 5594.78 5116.13 4531.34 3802.81 10380.43

[351] 4892.24 10585.57 3755.65 3744.65 8628.55 8600.73 6204.46

[358] 6612.18 11891.18 4455.73 4801.56 3925.92 20272.43 31300.08

[365] 7330.80 30300.82 6838.34 4200.97 5022.26 17078.45 -4287.63

[372] 187482.17 6100.74 88125.38 4012.56 10736.11 3911.56 5050.77

[379] 3947.87 4509.37 3886.67 4737.13 20098.10 16219.22 5739.46

[386] 4748.45 10097.37 5505.79 5367.80 5288.83 13110.02 5585.49

[393] 3960.63 4627.62 7677.71 3661.07 40340.78 5288.63 26763.34

[400] 5155.66 10158.54 4204.10 4301.22 52750.84 4393.61 4161.14

[407] 6752.14 3864.44 3651.97 3770.12 9065.76 256438.49 7925.32

[414] 5595.77 8393.22 7490.17 5083.06 6315.23 5361.02 5192.10

[421] 7276.90 4314.72 2458.00 2341.00 2708.00 2212.00 2529.00

[428] 5543.00 77242.00 3503.00 11071.00 2918.00 6455.00 7965.00

[435] 4020.00 2363.00 4764.00 2337.00 3620.00 3290.00 2185.00

[442] 3064.00 2671.00 3699.00 2652.00 6677.00 4384.00 4089.00

[449] 5009.00 4279.00 2187.00 3354.00 7404.00 3350.00 2752.00

[456] 2990.00 10899.00 7518.00 5575.00 4903.00 3028.00 2963.00

[463] 3973.00 3832.00 3069.00 4654.00 3143.00 8366.00 5234.00

[470] 24210.00 2262.00 11250.00 6479.00 2385.00 3646.00 5256.00

[477] 9847.00 3804.00 2683.00 4274.00 20915.00 9454.00 23377.00

[484] 4876.00 2363.00 3642.00 2531.00 9203.00 2915.00 5497.00

[491] 4796.00 2217.00 5434.00 17280.00 3707.00 5466.00 5557.00

[498] 19021.00 4313.00 30787.00 2194.00 3943.00 2341.00 15911.00

[505] 4160.00 2594.00 4300.00 2225.00 2418.00 3570.00 2208.00

[512] 2645.00 6172.00 4232.00 3352.00 12540.00 4780.00 4299.00

[519] 3635.00 3509.00 6947.00 4686.00 5200.00 5770.00 2415.00

[526] 2215.00 16119.00 3839.00 2224.00 5460.00 2581.00 4512.00

[533] 2461.00 2876.00 2660.00 2640.00 2573.00 3169.00 4372.00

[540] 2182.00 3637.00 5056.00 61803.00 2304.00 3501.00 3164.00

[547] 7258.00 12804.00 2493.00 23514.00 2281.00 2557.00 3700.00

[554] 2278.00 2801.00 10062.00 2731.00 2307.00 2474.00 2724.00

[561] 4483.00 3439.00 3025.00 2231.00 2668.00 2478.00 5370.00

[568] 3846.00 3741.00 4232.00 4708.00 5374.00 5593.00 3183.00

[575] 12593.00 16335.00 2177.00 6942.00 2455.00 3366.00 3850.00

[582] 57025.00 4395.00 3919.00 3657.00 5783.00 2190.00 7098.00

[589] 2668.00 4306.00 2748.00 58021.00 2406.00 3075.00 4410.00

[596] 4505.00 2509.00 2973.00 2906.00 2339.00 2901.00 2743.00

[603] 2662.00 2777.00 2658.00 5978.00 7998.00 7026.00 196719.00

[610] 5382.00 2498.00 12474.00 3222.00 2482.00 3725.00 3817.00

[617] 2217.00 7406.00 2688.00 5942.00 3634.00 5656.00 6569.00

[624] 4823.00 77180.00 4083.00 2801.00 3295.00 3692.00 4204.00

[631] 2863.00 2668.00 2411.00 9191.00 2786.00 3307.00 28590.00

[638] 5069.00 3733.00 2362.00 3096.00 5991.00 3026.00 3633.00

[645] 2694.00 2242.00 11394.00 6035.00 5648.00 3329.00 2252.00

[652] 3557.00 6084.00 4431.00 2752.00 2940.00 2768.00 37720.00

[659] 6927.00 2924.00 2293.00 4003.00 2998.00 3136.00 6084.00

[666] 3823.00 2348.00 5545.00 3489.00 3105.00 4032.00 4410.00

[673] 4978.00 5493.00 6572.00 2231.00 2724.00 3065.00 2967.00

[680] 9014.00 27660.00 3331.00 2515.00 3337.00 9007.00 10941.00

[687] 2543.00 2506.00 2409.00 2311.00 4725.00 2256.00 5110.00

[694] 13845.00 32148.00 3516.00 8738.00 3072.00 2506.00 5853.00

[701] 3791.00 3008.00 5004.00 3203.00 7559.00 3137.00 2994.00

[708] 2627.00 2621.00 8000.00 2571.00 32184.00 2317.00 3858.00

[715] 32592.00 4233.00 6493.00 2365.00 4595.00 5346.00 2914.00

[722] 2822.00 2410.00 2670.00 3267.00 2367.00 8873.00 2524.00

[729] 2812.00 2368.00 49390.00 4019.00 6211.00 5458.00 4264.00

[736] 4989.00 5769.00 6727.00 4662.00 4280.00 4435.00 5236.00

[743] 2768.00 8470.00 2234.00 10154.00 3825.00 3183.00 3158.00

[750] 2725.00 2408.00 6613.00 2669.00 4537.00 2730.00 5163.00

[757] 2856.00 2850.00 3362.00 11470.00 3856.00 3123.00 5902.00

[764] 2680.00 4777.00 25646.00 7430.00 32324.00 4266.00 2195.00

[771] 3082.00 9435.00 69029.00 3530.00 63012.00 2752.00 2428.00

[778] 5861.00 3095.00 4400.00 4135.00 2552.00 2470.00 3100.00

[785] 2479.00 8278.00 9725.00 2478.00 3833.00 3004.00 2455.00

[792] 3567.00 8439.00 3376.00 3957.00 3316.00 4128.00 22865.00

[799] 19086.00 2706.00 2638.00 5698.00 2516.00 9751.00 27571.00

[806] 2229.00 2571.00 3810.00 2450.00 3098.00 7432.00 3571.00

[813] 4250.00 2518.00 64122.00 4219.00 2877.00 5461.00 2625.00

[820] 3170.00 3839.00 2890.00 3206.00 3459.00 2488.00 7824.00

> #Eliminating outliers-1-TotalRevenue

> iqr <- IQR(customerData$TotalRevenue)

> Q <- quantile(customerData$TotalRevenue, probs = c(.25, .75), na.rm = FALSE)

> eliminated <- subset(customerData, customerData$TotalRevenue > (Q[1] - 1.5\*iqr) & customerData$TotalRevenue < (Q[2]+1.5\*iqr))

> #Emliminating outliers-2-TotalRevenue

> iqr <- IQR(customerData$TotalItemsSold)

> Q <- quantile(eliminated$TotalItemsSold, probs = c(.25,.75), na.rm = FALSE)

> customerData <- subset(eliminated, eliminated$TotalItemsSold > (Q[1]-1.5\*iqr) & eliminated$TotalItemsSold < (Q[2]+1.5\*iqr))

> remove(eliminated)

> #Scaling the data before applying the clustering algorithms

> customerData <- scale(customerData)

> head(customerData)

TotalRevenue TotalItemsSold

[1,] -1.0656289 -1.04819627

[2,] 1.2795158 0.38622104

[3,] -0.6194303 -0.60036709

[4,] 0.9964518 0.02022867

[5,] -0.9468739 -1.00273138

[6,] 0.3746426 0.15662334

> #Determine optimal number of clusters using Elbow Method

> set.seed(123)

> #function to compute total within-cluster sum of square

> wss <- function(k){

+ kmeans(customerData, k, nstart = 10)$tot.withinss

+ }

> #Compute and plot wss for k=1 to k=15

> k.values <- 1:15

> #extract wss for 2-15 clusters

> wss\_values <- map\_dbl(k.values, wss)

There were 16 warnings (use warnings() to see them)

> print(wss\_values)

[1] 7650.0000 2370.8968 1356.3542 987.4831 803.2673 684.6931 595.1532

[8] 536.8576 478.2400 438.4974 405.6129 377.9827 357.5096 334.3090

[15] 317.7990

> plot(k.values, wss\_values,

+ type = "b", pch = 19, frame =FALSE,

+ xlab = "Number of clusters K",

+ ylab = "Total within-clusters sum of squares")

> #Dissimilarity matrix

> d <- dist(customerData, method = "euclidean")

> #Hierarchical clustering using complete linkage

> hc1 <- hclust(d, method = "complete")

> #Plot the obtained dendrogram

> plot(hc1, cex = 0.6, hang = -1)

> #compute divisive heirarchical clustering

> hc2 <- diana(customerData)

> #Divise coefficient; amount of clustering structure found

> hc2$dc

[1] 0.9958585

> #plot dendrogram

> pltree(hc2, cex=0.6, hang=-1, main="Dendrogram of diana")

> #Methods to asses

> m <- c("average", "single", "complete","ward")

> names(m) <- c("average", "single", "complete", "ward")

> #function to compute coefficient

> ac <- function(x){

+ agnes(customerData, method = x)$ac

+ }

> map\_dbl(m, ac)

#Ward's method

#Ward's method

#cut tree into 3 groups

str(sub\_grp)

head(sub\_grp)

table(sub\_grp)

fviz\_cluster(list(data = customerData, cluster = sub\_grp))

#Dendogram with border around 3 clusters

rect.hclust(hc3, k=3, border = 5:10)

#Determining Optimal clusters in Hierachical Clustering using Elbow Method

wssForHierachical <- function(k){

#calculating WSS at hcut=3

Diagram

Description automatically generated with medium confidence

Chart, line chart

Description automatically generated

Chart, scatter chart

Description automatically generated

Diagram

Description automatically generated

Chart, box and whisker chart

Description automatically generated with medium confidence

Chart, scatter chart

Description automatically generated

Chart, line chart

Description automatically generated