d212_task1_revision2.R

sigsp

2022-09-30

```
############################## TITLE ########################################

## Author: Stephen E. Porter
## Title: D212 Task 1 Clustering Analysis
## Course: WGU D212: Data Mining II
## Instructor: Dr.Keiona Middleton


############################## LIBRARIES ####################################
options(warn = -1)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(caret)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(dplyr)
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
library(ggplot2)
library(cluster)


############################## DATA PREPARATION ##############################

# Import CSV as data frame
df <- read.csv(file = 'C:/WGU/D212 Data Mining II/churn_clean.csv')
```

```r
# Checking for nulls
sapply(df, function(x) sum(is.na(x)))
```

```
##            CaseOrder          Customer_id          Interaction
##                    0                    0                    0
##                  UID                 City                State
##                    0                    0                    0
##               County                  Zip                  Lat
##                    0                    0                    0
##                  Lng           Population                 Area
##                    0                    0                    0
##             TimeZone                  Job             Children
##                    0                    0                    0
##                  Age               Income              Marital
##                    0                    0                    0
##               Gender                Churn   Outage_sec_perweek
##                    0                    0                    0
##                Email             Contacts Yearly_equip_failure
##                    0                    0                    0
##               Techie             Contract           Port_modem
##                    0                    0                    0
##               Tablet      InternetService                Phone
##                    0                    0                    0
##             Multiple       OnlineSecurity         OnlineBackup
##                    0                    0                    0
##     DeviceProtection          TechSupport          StreamingTV
##                    0                    0                    0
##      StreamingMovies     PaperlessBilling        PaymentMethod
##                    0                    0                    0
##               Tenure        MonthlyCharge    Bandwidth_GB_Year
##                    0                    0                    0
##                Item1                Item2                Item3
##                    0                    0                    0
##                Item4                Item5                Item6
##                    0                    0                    0
##                Item7                Item8
##                    0                    0
```

```r
str(df)
```

```
## 'data.frame':    10000 obs. of  50 variables:
##  $ CaseOrder          : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Customer_id        : chr  "K409198" "S120509" "K191035" "D90850" ...
##  $ Interaction        : chr  "aa90260b-4141-4a24-8e36-b04ce1f4f77b" "fb76459f-c047-4a9d-8af9-e0f7d4
##  $ UID                : chr  "e885b299883d4f9fb18e39c75155d990" "f2de8bef964785f41a2959829830fb8a"
##  $ City               : chr  "Point Baker" "West Branch" "Yamhill" "Del Mar" ...
##  $ State              : chr  "AK" "MI" "OR" "CA" ...
##  $ County             : chr  "Prince of Wales-Hyder" "Ogemaw" "Yamhill" "San Diego" ...
##  $ Zip                : int  99927 48661 97148 92014 77461 31030 37847 73109 34771 45237 ...
##  $ Lat                : num  56.3 44.3 45.4 33 29.4 ...
##  $ Lng                : num  -133.4 -84.2 -123.2 -117.2 -95.8 ...
##  $ Population         : int  38 10446 3735 13863 11352 17701 2535 23144 17351 20193 ...
##  $ Area               : chr  "Urban" "Urban" "Urban" "Suburban" ...
##  $ TimeZone           : chr  "America/Sitka" "America/Detroit" "America/Los_Angeles" "America/Los_A
```

```
##  $ Job                : chr  "Environmental health practitioner" "Programmer, multimedia" "Chief Fi
##  $ Children           : int  0 1 4 1 0 3 0 2 2 1 ...
##  $ Age                : int  68 27 50 48 83 83 79 30 49 86 ...
##  $ Income             : num  28562 21705 9610 18925 40074 ...
##  $ Marital            : chr  "Widowed" "Married" "Widowed" "Married" ...
##  $ Gender             : chr  "Male" "Female" "Female" "Male" ...
##  $ Churn              : chr  "No" "Yes" "No" "No" ...
##  $ Outage_sec_perweek : num  7.98 11.7 10.75 14.91 8.15 ...
##  $ Email              : int  10 12 9 15 16 15 10 16 20 18 ...
##  $ Contacts           : int  0 0 0 2 2 3 0 0 2 1 ...
##  $ Yearly_equip_failure: int  1 1 1 0 1 1 1 0 3 0 ...
##  $ Techie             : chr  "No" "Yes" "Yes" "Yes" ...
##  $ Contract           : chr  "One year" "Month-to-month" "Two Year" "Two Year" ...
##  $ Port_modem         : chr  "Yes" "No" "Yes" "No" ...
##  $ Tablet             : chr  "Yes" "Yes" "No" "No" ...
##  $ InternetService    : chr  "Fiber Optic" "Fiber Optic" "DSL" "DSL" ...
##  $ Phone              : chr  "Yes" "Yes" "Yes" "Yes" ...
##  $ Multiple           : chr  "No" "Yes" "Yes" "No" ...
##  $ OnlineSecurity     : chr  "Yes" "Yes" "No" "Yes" ...
##  $ OnlineBackup       : chr  "Yes" "No" "No" "No" ...
##  $ DeviceProtection   : chr  "No" "No" "No" "No" ...
##  $ TechSupport        : chr  "No" "No" "No" "No" ...
##  $ StreamingTV        : chr  "No" "Yes" "No" "Yes" ...
##  $ StreamingMovies    : chr  "Yes" "Yes" "Yes" "No" ...
##  $ PaperlessBilling   : chr  "Yes" "Yes" "Yes" "Yes" ...
##  $ PaymentMethod      : chr  "Credit Card (automatic)" "Bank Transfer(automatic)" "Credit Card (aut
##  $ Tenure             : num  6.8 1.16 15.75 17.09 1.67 ...
##  $ MonthlyCharge      : num  172 243 160 120 150 ...
##  $ Bandwidth_GB_Year  : num  905 801 2055 2165 271 ...
##  $ Item1              : int  5 3 4 4 4 3 6 2 5 2 ...
##  $ Item2              : int  5 4 4 4 4 3 5 2 4 2 ...
##  $ Item3              : int  5 3 2 4 4 3 6 2 4 2 ...
##  $ Item4              : int  3 3 4 2 3 2 4 5 3 2 ...
##  $ Item5              : int  4 4 4 5 4 4 1 2 4 5 ...
##  $ Item6              : int  4 3 3 4 4 3 5 3 3 2 ...
##  $ Item7              : int  3 4 3 3 4 3 5 4 4 3 ...
##  $ Item8              : int  4 4 3 3 5 3 5 5 4 3 ...
```

```
summary(df)
```

```
##    CaseOrder     Customer_id        Interaction            UID
##  Min.   :    1   Length:10000       Length:10000       Length:10000
##  1st Qu.: 2501   Class :character   Class :character   Class :character
##  Median : 5000   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 5000
##  3rd Qu.: 7500
##  Max.   :10000
##      City              State              County               Zip
##  Length:10000       Length:10000       Length:10000       Min.   :  601
##  Class :character   Class :character   Class :character   1st Qu.:26293
##  Mode  :character   Mode  :character   Mode  :character   Median :48870
##                                                           Mean   :49153
##                                                           3rd Qu.:71867
##                                                           Max.   :99929
##       Lat              Lng            Population         Area
```

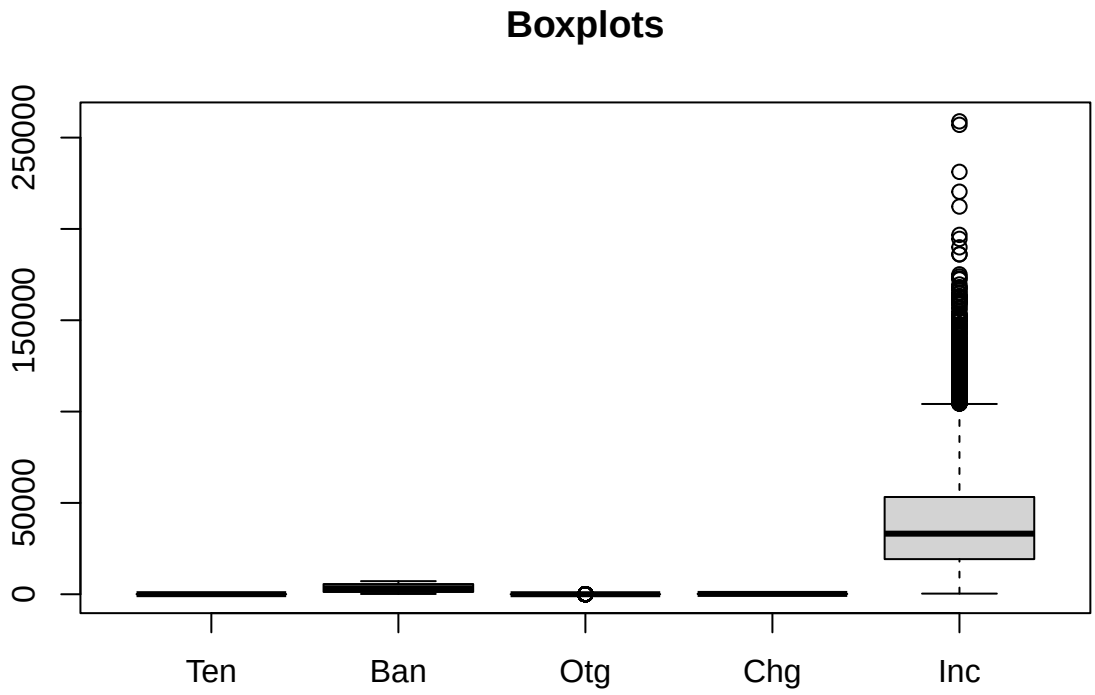
```
##  Min.   :17.97   Min.   :-171.69   Min.   :     0   Length:10000
##  1st Qu.:35.34   1st Qu.: -97.08   1st Qu.:   738   Class :character
##  Median :39.40   Median : -87.92   Median :  2910   Mode  :character
##  Mean   :38.76   Mean   : -90.78   Mean   :  9757
##  3rd Qu.:42.11   3rd Qu.: -80.09   3rd Qu.: 13168
##  Max.   :70.64   Max.   : -65.67   Max.   :111850
##    TimeZone             Job               Children           Age
##  Length:10000       Length:10000       Min.   : 0.000   Min.   :18.00
##  Class :character   Class :character   1st Qu.: 0.000   1st Qu.:35.00
##  Mode  :character   Mode  :character   Median : 1.000   Median :53.00
##                                        Mean   : 2.088   Mean   :53.08
##                                        3rd Qu.: 3.000   3rd Qu.:71.00
##                                        Max.   :10.000   Max.   :89.00
##      Income            Marital             Gender             Churn
##  Min.   :   348.7   Length:10000       Length:10000       Length:10000
##  1st Qu.: 19224.7   Class :character   Class :character   Class :character
##  Median : 33170.6   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 39806.9
##  3rd Qu.: 53246.2
##  Max.   :258900.7
##  Outage_sec_perweek     Email           Contacts      Yearly_equip_failure
##  Min.   : 0.09975   Min.   : 1.00   Min.   :0.0000   Min.   :0.000
##  1st Qu.: 8.01821   1st Qu.:10.00   1st Qu.:0.0000   1st Qu.:0.000
##  Median :10.01856   Median :12.00   Median :1.0000   Median :0.000
##  Mean   :10.00185   Mean   :12.02   Mean   :0.9942   Mean   :0.398
##  3rd Qu.:11.96949   3rd Qu.:14.00   3rd Qu.:2.0000   3rd Qu.:1.000
##  Max.   :21.20723   Max.   :23.00   Max.   :7.0000   Max.   :6.000
##     Techie            Contract          Port_modem           Tablet
##  Length:10000       Length:10000       Length:10000       Length:10000
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##  InternetService       Phone             Multiple         OnlineSecurity
##  Length:10000       Length:10000       Length:10000       Length:10000
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##  OnlineBackup       DeviceProtection   TechSupport        StreamingTV
##  Length:10000       Length:10000       Length:10000       Length:10000
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##  StreamingMovies    PaperlessBilling   PaymentMethod          Tenure
##  Length:10000       Length:10000       Length:10000       Min.   : 1.000
##  Class :character   Class :character   Class :character   1st Qu.: 7.918
##  Mode  :character   Mode  :character   Mode  :character   Median :35.431
##                                                           Mean   :34.526
##                                                           3rd Qu.:61.480
```

```
##                                                       Max.   :71.999
##  MonthlyCharge    Bandwidth_GB_Year     Item1           Item2
##  Min.   : 79.98   Min.   : 155.5    Min.   :1.000   Min.   :1.000
##  1st Qu.:139.98   1st Qu.:1236.5    1st Qu.:3.000   1st Qu.:3.000
##  Median :167.48   Median :3279.5    Median :3.000   Median :4.000
##  Mean   :172.62   Mean   :3392.3    Mean   :3.491   Mean   :3.505
##  3rd Qu.:200.73   3rd Qu.:5586.1    3rd Qu.:4.000   3rd Qu.:4.000
##  Max.   :290.16   Max.   :7159.0    Max.   :7.000   Max.   :7.000
##      Item3           Item4           Item5           Item6           Item7
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.00
##  1st Qu.:3.000   1st Qu.:3.000   1st Qu.:3.000   1st Qu.:3.000   1st Qu.:3.00
##  Median :3.000   Median :3.000   Median :3.000   Median :3.000   Median :4.00
##  Mean   :3.487   Mean   :3.498   Mean   :3.493   Mean   :3.497   Mean   :3.51
##  3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.00
##  Max.   :8.000   Max.   :7.000   Max.   :7.000   Max.   :8.000   Max.   :7.00
##      Item8
##  Min.   :1.000
##  1st Qu.:3.000
##  Median :3.000
##  Mean   :3.496
##  3rd Qu.:4.000
##  Max.   :8.000
```

```
# Keeping desired columns
to_keep <- c('Tenure', 'Bandwidth_GB_Year', 'Outage_sec_perweek',
             'MonthlyCharge', 'Income')

dfDropped = df[to_keep]
str(dfDropped)
```

```
## 'data.frame':    10000 obs. of  5 variables:
##  $ Tenure            : num  6.8 1.16 15.75 17.09 1.67 ...
##  $ Bandwidth_GB_Year : num  905 801 2055 2165 271 ...
##  $ Outage_sec_perweek: num  7.98 11.7 10.75 14.91 8.15 ...
##  $ MonthlyCharge     : num  172 243 160 120 150 ...
##  $ Income            : num  28562 21705 9610 18925 40074 ...
```

```
################################ OUTLIER DETECTION ################################

#Check for outliers in boxplot
boxplot(dfDropped$Tenure, dfDropped$Bandwidth_GB_Year,
        dfDropped$Outage_sec_perweek, dfDropped$MonthlyCharge, dfDropped$Income,
        main = "Boxplots",
        names = c("Ten", "Ban", "Otg", "Chg", "Inc"),
        horizontal = FALSE)
```




```
# Check each column for outliers
tenOut <- boxplot(dfDropped$Tenure, plot=FALSE)$out
tenOut
```

```
## numeric(0)
```

```
banOut <- boxplot(dfDropped$Bandwidth_GB_Year, plot=FALSE)$out
banOut
```

```
## numeric(0)
```

```
otgOut <- boxplot(dfDropped$Outage_sec_perweek, plot=FALSE)$out
otgOut
```

```
##  [1] 18.19542503 18.39537758 19.07180624 18.30717385 18.30369591  1.18025898
##  [7] 19.08168517  0.76027743 19.26778150 17.96334654 18.94289163  0.12005772
## [13]  1.72652484 18.28180588 17.94420077 18.07990420  0.63660795  0.50737490
## [19]  2.01774600 19.50058000 18.31879000 18.44059000 18.77915000 20.30462000
## [25]  0.99528960 18.40676000 18.19254000  2.02083400  1.55678400 18.34115000
## [31] 18.78705000 18.21093000 17.90595000  1.51649700  0.23227950 17.99204000
## [37] 18.85173000  1.86467600  1.27643800  0.90033260 21.20723000 18.25245000
## [43]  0.35504830 19.26111000  0.82699800 19.71756000 20.62504000 19.01962000
## [49] 18.11802000  0.94033040  0.39186590  2.03977100  1.33256000 18.30895000
## [55] 18.15330000  1.14479600  2.08173300 17.97393000  1.55649900  1.63663400
## [61]  1.92368900  1.88242600  1.28345800 17.91239000 19.10781000 18.19674000
## [67]  1.89642200  0.09974694 18.45023000 18.17620000  1.45088000 19.65711000
## [73] 19.01629000  2.01514300 19.20969000  0.82754400
```

```
# Outage seconds per week has outliers. Create temp data frame & remove outliers
temp <- dfDropped
temp <- temp[-which(temp$Outage_sec_perweek %in% otgOut),]
str(temp)
```

```
## 'data.frame':    9924 obs. of  5 variables:
##  $ Tenure            : num  6.8 1.16 15.75 17.09 1.67 ...
##  $ Bandwidth_GB_Year : num  905 801 2055 2165 271 ...
##  $ Outage_sec_perweek: num  7.98 11.7 10.75 14.91 8.15 ...
##  $ MonthlyCharge     : num  172 243 160 120 150 ...
##  $ Income            : num  28562 21705 9610 18925 40074 ...
```

```
chgOut <- boxplot(temp$MonthlyCharge, plot=FALSE)$out
chgOut
```

```
## numeric(0)
```

```
incOut <- boxplot(temp$Income, plot=FALSE)$out
incOut
```

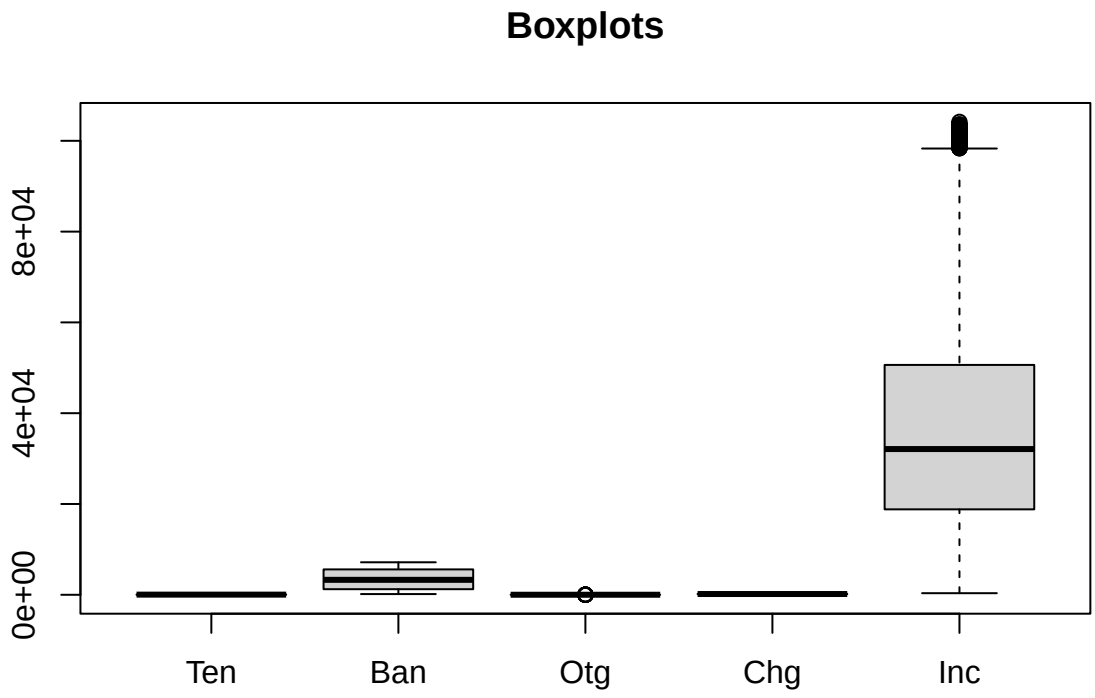
```
##   [1] 115114.6 132116.3 115510.5 125814.9 122957.2 107111.8 135727.7 118022.1
##   [9] 123763.1 119968.6 114398.4 105646.7 122263.8 112429.2 119964.8 156740.7
##  [17] 146494.7 159315.5 163086.2 172884.1 111380.5 114609.0 169580.7 168097.1
##  [25] 120435.6 112914.4 106964.8 112245.1 125002.3 113086.6 132140.0 131500.5
##  [33] 186035.0 109366.1 154718.9 117194.6 126236.2 162360.1 132334.8 122741.6
##  [41] 109373.7 112031.3 135171.3 121993.2 115520.5 122844.9 106608.4 108847.9
##  [49] 149502.9 121849.0 108698.7 118120.2 116572.8 117333.4 106581.9 104519.8
##  [57] 165151.0 108059.1 147436.9 152172.9 116888.8 115594.6 137589.2 127881.0
##  [65] 116572.3 115414.9 146782.3 104548.7 113438.3 111971.9 137119.7 143972.7
##  [73] 143217.5 116086.2 134691.3 108839.1 108135.1 111592.0 142650.5 128114.3
##  [81] 138555.9 108960.7 105396.3 115440.4 175137.3 131511.8 108914.2 152131.7
##  [89] 146951.6 145163.1 105033.0 116453.0 107174.6 122915.6 104362.5 104867.5
##  [97] 113641.2 119667.4 123891.9 117623.7 145569.5 104539.2 129628.7 160589.2
## [105] 120286.2 167566.6 117185.4 111779.6 110886.7 159113.5 121055.6 149959.8
## [113] 142086.1 125660.1 138122.3 108287.6 125769.6 152972.9 147682.0 114160.2
## [121] 172372.2 105302.4 131265.4 128906.6 138723.0 128728.3 121668.3 166553.1
## [129] 139474.8 189938.4 116396.5 129473.6 104452.6 149968.0 126678.4 119318.8
## [137] 105157.2 124493.2 111143.1 125288.6 107570.9 258900.7 105969.1 115681.5
## [145] 122756.3 114390.3 114051.7 130732.2 134838.8 156571.0 110960.3 143794.0
## [153] 114401.9 118179.2 116628.8 133161.6 120301.3 114551.7 107772.3 162842.8
## [161] 113511.5 108409.6 129137.2 120650.1 151312.2 135891.8 146518.2 120324.8
## [169] 112181.6 152813.5 141362.6 131834.1 111892.6 113028.1 115405.2 124735.8
## [177] 104931.3 116303.0 131647.5 160862.9 106700.6 220383.0 116562.7 130048.4
## [185] 106862.5 114405.4 135516.9 110579.2 151181.8 161251.0 212255.3 120330.5
## [193] 125041.1 137978.3 108982.2 114851.2 105022.5 131680.1 106634.6 115239.5
## [201] 121118.5 167846.0 132149.5 113595.2 129787.2 105193.1 123206.3 109866.0
## [209] 120864.5 119415.5 104558.0 105425.1 121444.9 110391.6 112097.3 146317.6
## [217] 196746.0 153026.4 231252.0 123991.0 108072.0 146958.0 115022.1 127578.3
## [225] 194550.7 113002.5 133882.2 124025.1 115783.2 117468.7 139625.0 118070.7
## [233] 105739.9 106704.4 121964.1 146544.6 106307.8 112803.7 142974.0 112687.7
## [241] 118983.0 110491.4 108805.1 113865.7 137977.7 109653.3 107720.5 135166.6
## [249] 140367.2 113095.0 131421.6 149891.1 126992.3 112837.8 106084.8 138155.7
## [257] 123006.6 159532.5 107804.6 150264.3 126572.7 105745.6 113254.1 117867.2
## [265] 126024.3 126805.9 146536.7 111442.1 146590.9 105005.6 113593.3 113912.9
## [273] 119599.4 146600.8 164529.0 114044.9 107341.5 108956.8 115758.0 112961.8
```

```
## [281] 112839.8 105294.0 116972.1 124590.0 128842.9 121986.8 143641.2 111778.1
## [289] 117715.3 110343.8 186156.6 140110.0 117890.3 117418.8 140030.5 107207.5
## [297] 160216.1 119017.8 142119.3 158549.3 125568.9 128998.6 256998.4 147889.4
## [305] 125034.1 163082.1 173978.0 146719.5 123006.3 117683.5 163156.7 122765.4
## [313] 123272.3 134967.2 120895.0 115029.8 105644.8 130319.3 149952.7 136818.5
## [321] 111497.4 118340.8 112773.2 109574.2 105986.5 113392.5 134443.3 108806.6
## [329] 128468.0 121219.6 109058.0 117089.4
```

```
# Income has many outliers - remove them
temp <- temp[-which(temp$Income %in% incOut),]
str(temp)
```

```
## 'data.frame':    9592 obs. of  5 variables:
##  $ Tenure            : num  6.8 1.16 15.75 17.09 1.67 ...
##  $ Bandwidth_GB_Year : num  905 801 2055 2165 271 ...
##  $ Outage_sec_perweek: num  7.98 11.7 10.75 14.91 8.15 ...
##  $ MonthlyCharge     : num  172 243 160 120 150 ...
##  $ Income            : num  28562 21705 9610 18925 40074 ...
```

```
# View boxplot for outliers
boxplot(temp$Tenure, temp$Bandwidth_GB_Year,
        temp$Outage_sec_perweek, temp$MonthlyCharge, temp$Income,
        main = "Boxplots",
        names = c("Ten", "Ban", "Otg", "Chg", "Inc"),
        horizontal = FALSE)
```




```
# Income still has outliers. Repeat process until none remain
incOut <- boxplot(temp$Income, plot=FALSE)$out
```

```
incOut
```

```
##  [1] 100076.65 103311.26  99195.08 100232.53  99519.26  99482.26 101000.30
##  [8]  99007.42 100626.29  99100.10 101771.45  98906.55  99800.11  99754.87
## [15] 100437.39 102905.68 102080.72 100033.86  99787.78  99199.26  98366.83
## [22] 101907.80  99291.94 100861.70 101807.80 103435.70 102090.50 101534.00
## [29]  98436.93 103306.60  99411.44  98555.98 100685.60 103112.30  98660.88
## [36]  99120.55 104166.70 102089.70 100860.90 101628.90 102072.00 100585.10
## [43] 100785.50  99168.20 100171.60 101766.00  99537.72 102059.00  98665.78
## [50] 100029.10 103625.10 103476.10 100352.40 102806.50 102609.30 103510.70
## [57]  98376.58  99873.57 101771.00  99342.82 102544.20  99108.60 100224.40
## [64]  98862.21 103499.70 101607.90 102823.40 102504.90 101681.00 103076.70
## [71] 101429.40 100711.60 100608.20 102928.60  99932.29  98425.53 101307.00
## [78] 103098.00 100257.60  99132.61  99699.68 102431.30 102702.50 100050.00
## [85]  98836.20 102633.90  99071.31 102173.50 102629.60
```

```
temp <- temp[-which(temp$Income %in% incOut),]
str(temp)
```

```
## 'data.frame':    9503 obs. of  5 variables:
##  $ Tenure            : num  6.8 1.16 15.75 17.09 1.67 ...
##  $ Bandwidth_GB_Year : num  905 801 2055 2165 271 ...
##  $ Outage_sec_perweek: num  7.98 11.7 10.75 14.91 8.15 ...
##  $ MonthlyCharge     : num  172 243 160 120 150 ...
##  $ Income            : num  28562 21705 9610 18925 40074 ...
```

```
incOut <- boxplot(temp$Income, plot=FALSE)$out
incOut
```

```
##  [1] 97761.18 98173.49 97592.52 97479.21 97462.46 97463.90 98189.95 98298.22
##  [9] 97763.56 97694.83 97691.33 98147.26 98176.66 98072.18 97769.66 97916.45
## [17] 97310.88 97539.36 97871.03 97230.00 97729.46 97997.05 97499.39 98120.00
```

```
temp <- temp[-which(temp$Income %in% incOut),]
str(temp)
```

```
## 'data.frame':    9479 obs. of  5 variables:
##  $ Tenure            : num  6.8 1.16 15.75 17.09 1.67 ...
##  $ Bandwidth_GB_Year : num  905 801 2055 2165 271 ...
##  $ Outage_sec_perweek: num  7.98 11.7 10.75 14.91 8.15 ...
##  $ MonthlyCharge     : num  172 243 160 120 150 ...
##  $ Income            : num  28562 21705 9610 18925 40074 ...
```

```
incOut <- boxplot(temp$Income, plot=FALSE)$out
incOut
```

```
## [1] 96857.54 97057.93 96753.80 96788.12 97020.52 96898.83 97088.50 96925.17
```

```
temp <- temp[-which(temp$Income %in% incOut),]
str(temp)
```

```
## 'data.frame':    9471 obs. of  5 variables:
##  $ Tenure            : num  6.8 1.16 15.75 17.09 1.67 ...
##  $ Bandwidth_GB_Year : num  905 801 2055 2165 271 ...
##  $ Outage_sec_perweek: num  7.98 11.7 10.75 14.91 8.15 ...
##  $ MonthlyCharge     : num  172 243 160 120 150 ...
##  $ Income            : num  28562 21705 9610 18925 40074 ...
```

```
incOut <- boxplot(temp$Income, plot=FALSE)$out
incOut
```

```
## [1] 96624.28 96579.40 96575.06
```

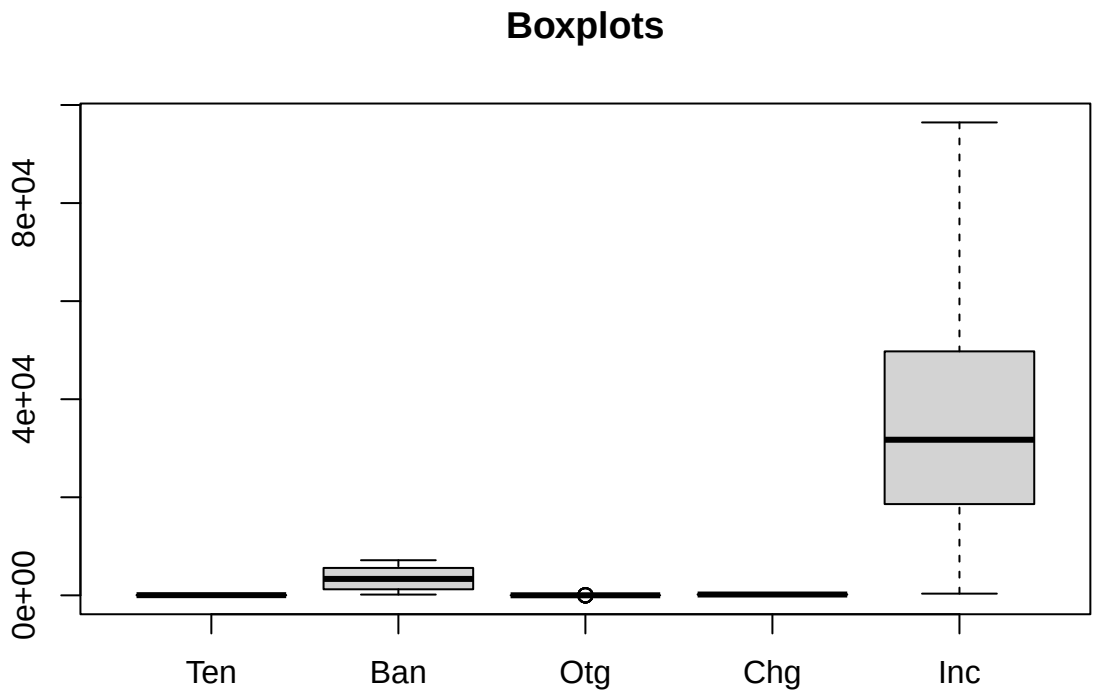
```
temp <- temp[-which(temp$Income %in% incOut),]
str(temp)
```

```
## 'data.frame':    9468 obs. of  5 variables:
##  $ Tenure            : num  6.8 1.16 15.75 17.09 1.67 ...
##  $ Bandwidth_GB_Year : num  905 801 2055 2165 271 ...
##  $ Outage_sec_perweek: num  7.98 11.7 10.75 14.91 8.15 ...
##  $ MonthlyCharge     : num  172 243 160 120 150 ...
##  $ Income            : num  28562 21705 9610 18925 40074 ...
```

```
incOut <- boxplot(temp$Income, plot=FALSE)$out
incOut
```

```
## numeric(0)
```

```
# Income has no outliers. Check boxplots
boxplot(temp$Tenure, temp$Bandwidth_GB_Year,
        temp$Outage_sec_perweek, temp$MonthlyCharge, temp$Income,
        main = "Boxplots",
        names = c("Ten", "Ban", "Otg", "Chg", "Inc"),
        horizontal = FALSE)
```




```
# Removing rows has created outliers in Outage - repeat process
otgOut <- boxplot(temp$Outage_sec_perweek, plot=FALSE)$out
```

```
otgOut
```

```
## [1]  2.110607  2.096375  2.094319  2.104824 17.833720 17.861530
```

```
temp <- temp[-which(temp$Outage_sec_perweek %in% otgOut),]
str(temp)
```

```
## 'data.frame':    9462 obs. of  5 variables:
##  $ Tenure            : num  6.8 1.16 15.75 17.09 1.67 ...
##  $ Bandwidth_GB_Year : num  905 801 2055 2165 271 ...
##  $ Outage_sec_perweek: num  7.98 11.7 10.75 14.91 8.15 ...
##  $ MonthlyCharge     : num  172 243 160 120 150 ...
##  $ Income            : num  28562 21705 9610 18925 40074 ...
```

```
otgOut <- boxplot(temp$Outage_sec_perweek, plot=FALSE)$out
otgOut
```

```
## [1] 17.82932
```

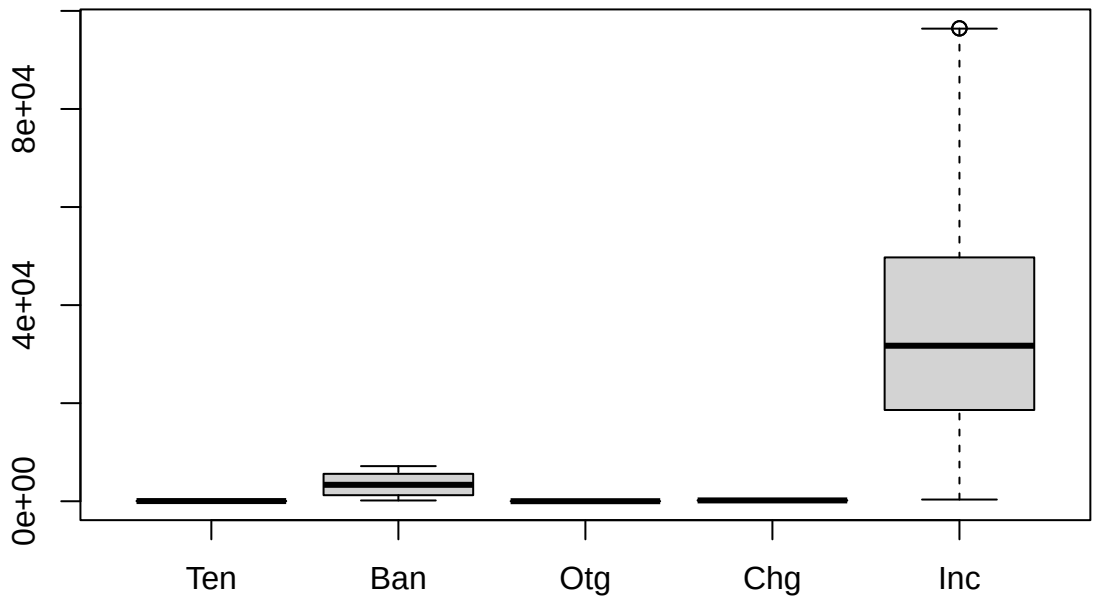
```
temp <- temp[-which(temp$Outage_sec_perweek %in% otgOut),]
str(temp)
```

```
## 'data.frame':    9461 obs. of  5 variables:
##  $ Tenure            : num  6.8 1.16 15.75 17.09 1.67 ...
##  $ Bandwidth_GB_Year : num  905 801 2055 2165 271 ...
##  $ Outage_sec_perweek: num  7.98 11.7 10.75 14.91 8.15 ...
##  $ MonthlyCharge     : num  172 243 160 120 150 ...
##  $ Income            : num  28562 21705 9610 18925 40074 ...
```

```
otgOut <- boxplot(temp$Outage_sec_perweek, plot=FALSE)$out
otgOut
```

```
## numeric(0)
```

```
boxplot(temp$Tenure, temp$Bandwidth_GB_Year,
        temp$Outage_sec_perweek, temp$MonthlyCharge, temp$Income,
        main = "Boxplots",
        names = c("Ten", "Ban", "Otg", "Chg", "Inc"),
        horizontal = FALSE)
```




```
# Removing rows has caused outliers in Income - repeat process
incOut <- boxplot(temp$Income, plot=FALSE)$out
incOut
```

```
## [1] 96431.37 96442.41
```

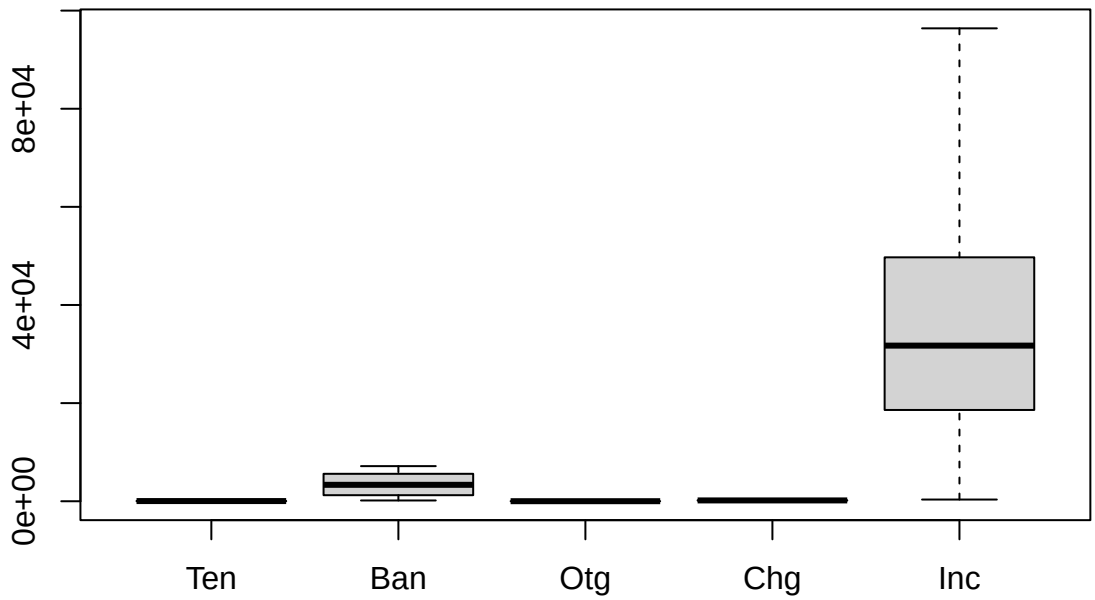
```
temp <- temp[-which(temp$Income %in% incOut),]
str(temp)
```

```
## 'data.frame':    9459 obs. of  5 variables:
##  $ Tenure            : num  6.8 1.16 15.75 17.09 1.67 ...
##  $ Bandwidth_GB_Year : num  905 801 2055 2165 271 ...
##  $ Outage_sec_perweek: num  7.98 11.7 10.75 14.91 8.15 ...
##  $ MonthlyCharge     : num  172 243 160 120 150 ...
##  $ Income            : num  28562 21705 9610 18925 40074 ...
```

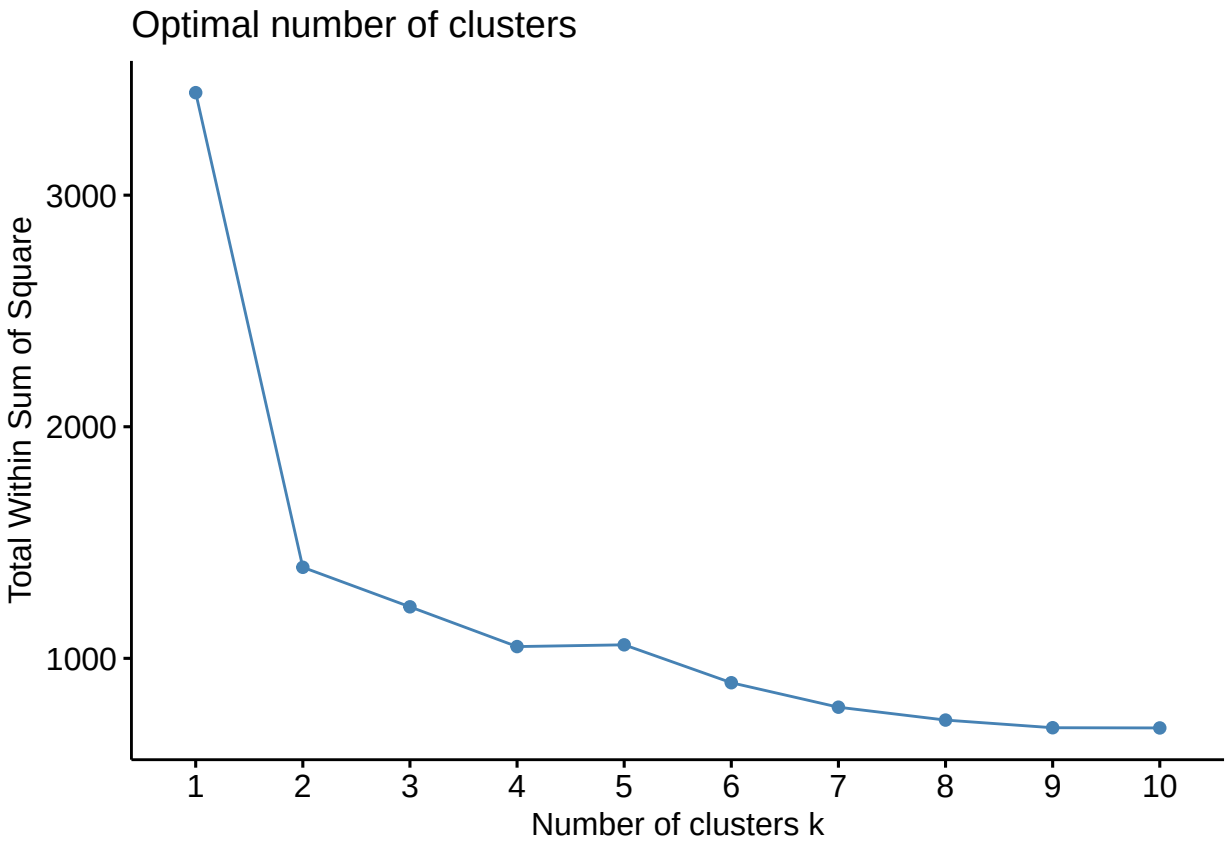
```
incOut <- boxplot(temp$Income, plot=FALSE)$out
incOut
```

```
## numeric(0)
```

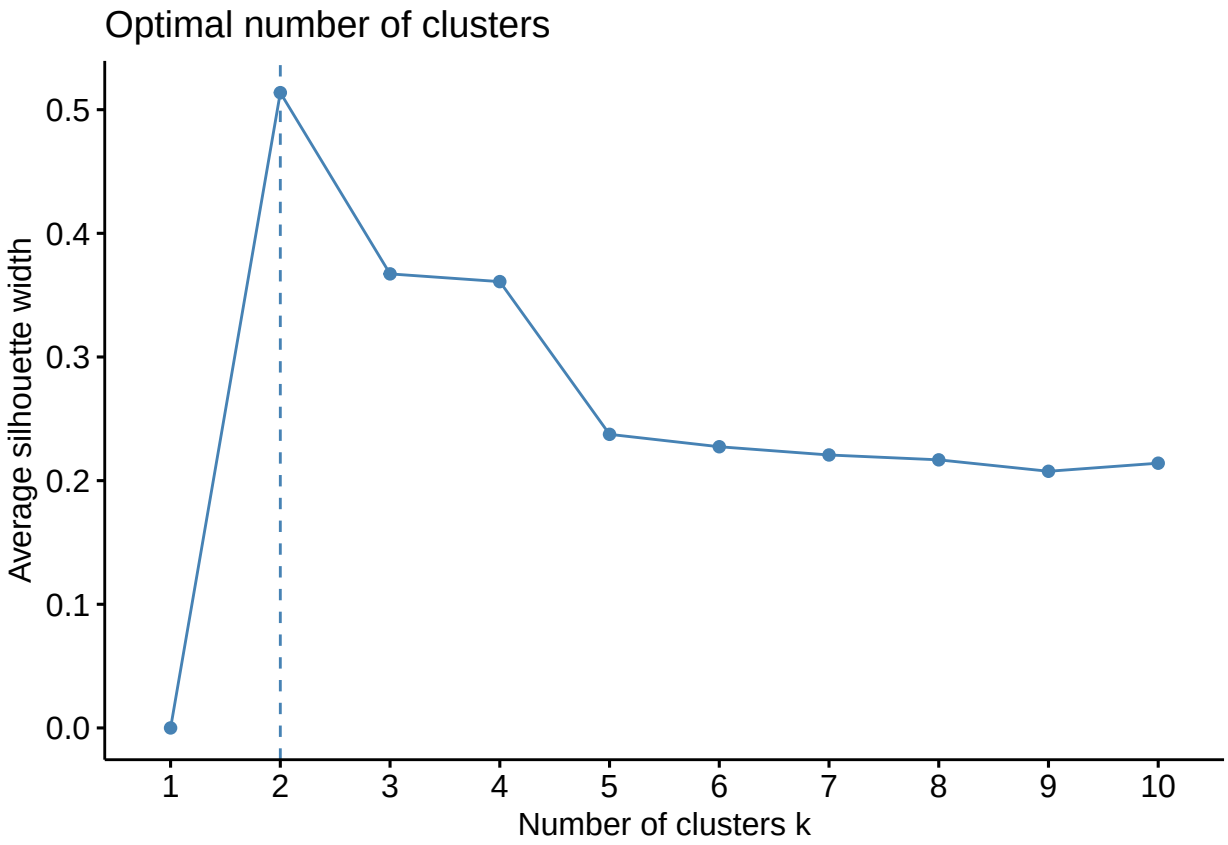
```
# Check boxplots for outliers
boxplot(temp$Tenure, temp$Bandwidth_GB_Year,
        temp$Outage_sec_perweek, temp$MonthlyCharge, temp$Income,
        main = "Boxplots",
        names = c("Ten", "Ban", "Otg", "Chg", "Inc"),
        horizontal = FALSE)
```




```
# No outliers remain. Ready for kmeans


################################ NORMALIZE & EXPORT ##############################

# Normalize Function
normalize = function(x) {
  result = (x - min(x)) / (max(x) - min(x))
  return(result)
}

# Normalize data set
dfNorm <- temp
for (i in colnames(dfNorm)) {
  dfNorm[i] <- normalize(dfNorm[i])
}

# Export data set for analysis
write.csv(dfNorm, 'C:/WGU/D212 Data Mining II/churn_kmeans.csv',
          row.names = FALSE)


################################ KMEANS ##########################################

# Identify optimal number of clusters
fviz_nbclust(dfNorm, FUNcluster = kmeans, method = "wss")
```




```
fviz_nbclust(dfNorm, FUNcluster = kmeans, method = "silhouette")
```




```
# k-means: 2 centers, 50 starting assignments
clusters2 <-kmeans(dfNorm, centers=2, nstart=50)
clusters2$centers
```

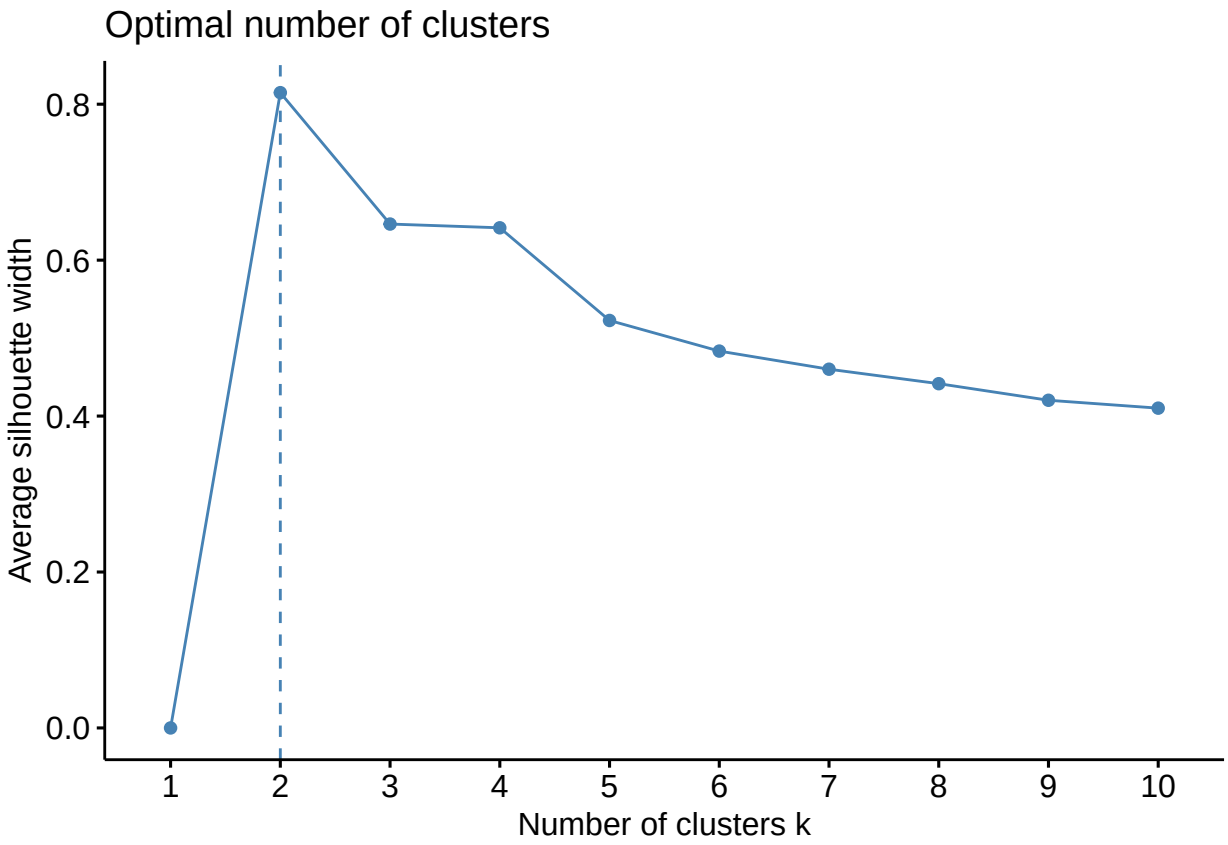
```
##      Tenure Bandwidth_GB_Year Outage_sec_perweek MonthlyCharge    Income
## 1 0.1143167         0.1648935          0.4987102     0.4412135 0.3699743
## 2 0.8304326         0.7597666          0.4997691     0.4411817 0.3700493
```

```
clusters2$betweenss / clusters2$totss
```

```
## [1] 0.5953391
```

```
# View clusters in plot
fviz_cluster(object=clusters2, data=dfNorm, geom = "point")
```


```
################################ TENURE & BANDWIDTH ##############################

dfFinal <-dfNorm[c('Tenure', 'Bandwidth_GB_Year')]
fviz_nbclust(dfFinal, FUNcluster = kmeans, method = "silhouette")
```



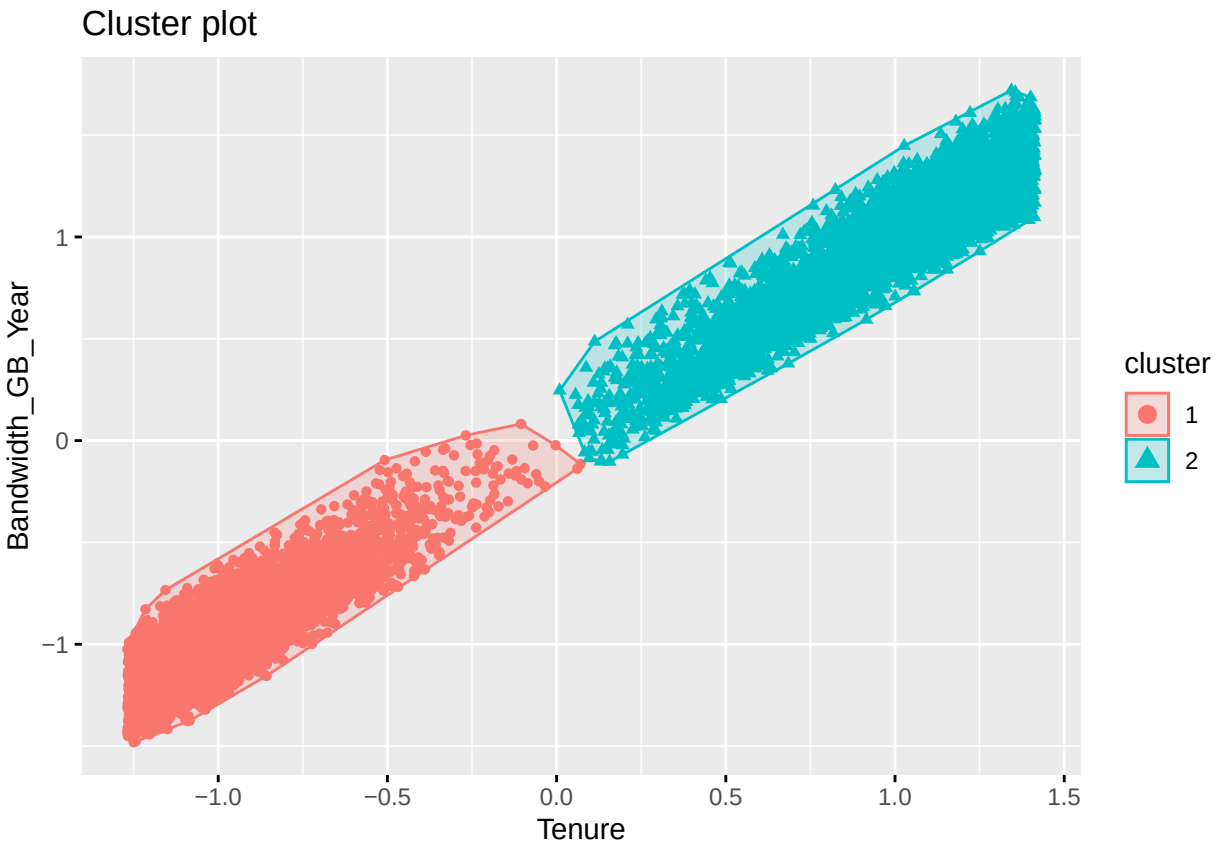

```
clusters2Final <- kmeans(dfFinal, centers=2, n=50)
clusters2Final$centers
```

```
##      Tenure Bandwidth_GB_Year
## 1 0.1143167         0.1648935
## 2 0.8304326         0.7597666
```

```
clusters2Final$betweenss / clusters2Final$totss
```

```
## [1] 0.9161668
```

```
fviz_cluster(object=clusters2Final, data=dfFinal, geom = "point")
```


Cluster plot
Bandwidth_GB_Year
1
0
−1
−1.0
−0.5
0.0
0.5
1.0
1.5
Tenure
cluster
1
2