**NAME**

**COLLEGE NUMBER**

#Loads the package library for grouping charts together

> library(cluster)

# Load the library tidyverse that comes along with graph ploting packages

 > library(tidyverse)

-- Attaching packages ------------------------------------------------------ tidyverse 1.3.1 --

v ggplot2 3.3.5     v purrr   0.3.4

v tibble  3.1.2     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()

Warning messages: #Should be ignored they show the versions of packages that are readily available

1: package ‘tidyverse’ was built under R version 4.1.1

2: package ‘tidyr’ was built under R version 4.1.1

3: package ‘readr’ was built under R version 4.1.1

4: package ‘purrr’ was built under R version 4.1.1

5: package ‘dplyr’ was built under R version 4.1.1

6: package ‘stringr’ was built under R version 4.1.1

7: package ‘forcats’ was built under R version 4.1.1

>

#In order to compute the K mean clusters using the Kmeans model,

#create a variable called w and assign it a function called k which takes

#the methods of df1, and function ntart that deafines the 1st boundery 10;

#where the 10 deifines  the first 10 generated configurations

#which also takes the variable as total within clusters defined by function $tot.withinss

w <- function(k) {

  kmeans(df1, k, nstart = 10 )$tot.withinss

}

#Define the class boundery of the k series as:

k.values <- 1:15

#Next create a new variable called wss\_values then

#the assign it a function called map\_dbl, to generate vector double outputs

#using the k start variable w initially created

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

# Use the R plot function plot(), to plot the vectors that

#will take the horizontal scale as "Number of clusters K"

#and Vertical scale as "Total within-clusters sum of squares"

#Use type="b" to plot dots by default and plot 19 set of characters

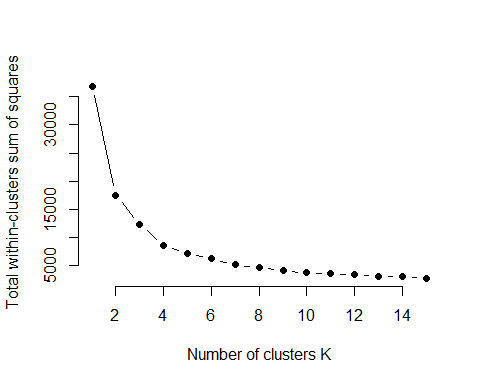
#Finally convert frames to factor columns by default using argument frame = FALSE

plot(k.values, wss\_values,

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

     xlab="Number of clusters K",

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



#Get the results

km <- kmeans(df1, centers = 10, nstart = 25)

print(km)

## K-means clustering with 10 clusters of sizes 36, 11, 14, 8, 15, 12, 30, 9, 7, 4

##

## Cluster means:

## List the   College      Age   Gender Year.Level      Game Game.type   Reasons  Opinion

## 1  4.777778 2.000000 1.166667   3.666667 22.361111  4.388889 34.250000 2.361111

## 2  4.818182 2.000000 1.454545   4.000000 12.000000 10.090909 12.545455 2.454545

## 3  3.785714 1.857143 1.214286   3.500000 22.285714  2.571429 20.500000 2.571429

## 4  4.625000 1.625000 1.125000   3.250000  8.500000 10.250000  2.375000 3.375000

## 5  4.000000 1.666667 1.400000   3.266667  3.333333  1.800000  9.266667 3.466667

## 6  4.416667 1.916667 1.083333   4.166667  3.833333  4.833333 23.166667 2.833333

## 7  5.100000 1.833333 1.133333   3.300000 21.833333  2.666667  7.700000 2.633333

## 8  6.111111 1.888889 1.222222   2.888889 11.000000 10.111111 33.222222 2.777778

## 9  4.571429 1.571429 1.285714   2.714286  4.000000  1.000000 37.714286 2.714286

## 10 5.500000 2.500000 1.250000   3.000000  6.500000 12.000000 40.000000 3.000000

##         GPA

## 1  3.027778

## 2  3.090909

## 3  3.428571

## 4  3.125000

## 5  3.200000

## 6  3.000000

## 7  3.633333

## 8  3.111111

## 9  3.000000

## 10 2.750000

##

## Clustering vector:

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

##   7  10   7   6   8   2   6   6   2   4   5   5   5   1   8   7   5   5   5   1

##  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40

##   6   3   7   3   5   1   5   1   3   2   1   2   1   7   5   3   1   4  10   7

##  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60

##   7   3   2   6   8   7   7   6   7   6   3   1   1   9   7   7   1   5   1   2

##  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80

##   2   1   3   5   1   5   9   2   2   7   7   1   1   1   8   1   8   7   5   7

##  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99 100

##   6   9   9   6   1   4   3   3   9   1   1   2   1   1   3   7   1   7   8   7

## 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120

##   1   5   1   6   6   7   1   1   6   1   1   7   4   4   1   8  10   5   3   7

## 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140

##   1  10   7   7   3   7   1   3   3   9   9   1   7   4   1   7   8   1   2   7

## 141 142 143 144 145 146

##   4   8   7   4   7   1

##

## Within cluster sum of squares by cluster:

##  [1]  576.11111  325.63636  602.57143  250.25000  440.66667  355.08333

##  [7] 1047.70000  149.11111   91.14286   50.50000

##  (between\_SS / total\_SS =  89.4 %)

##

## Available components:

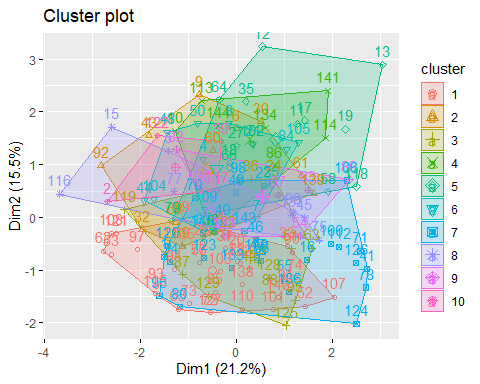
##

## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"

## [6] "betweenss"    "size"         "iter"         "ifault"

fviz\_cluster(km, data = df1)

#Conduct a plot on the dataset



#Generate the variable a and assign it the model and plot associated with it

a<-df1 %>%

  as\_tibble() %>%

  mutate(cluster = km$cluster,

         state = row.names(Game.type)) %>%

  ggplot(aes(df$Game.type, Age, color = factor(cluster), label = Game.type)) +

  geom\_text() +

  theme\_bw()+ ggtitle("Clustering between Game type and Age")+

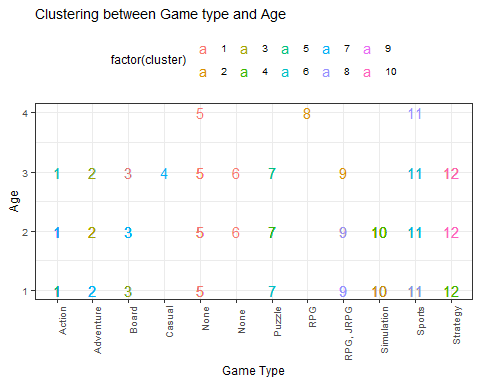
  theme(text = element\_text(size=9),

                          axis.text.x = element\_text(angle=90,hjust=1))+

  theme(legend.position="top", legend.box = "horizontal")+

  labs(y = "Age", x="Game Type")

a



b<-df1 %>%

  as\_tibble() %>%

  mutate(cluster = km$cluster,

         state = row.names(Game.type)) %>%

  ggplot(aes(df$Game.type, df$College, color = factor(cluster), label = Game.type)) +

  geom\_text()+

  theme\_bw()+ ggtitle("Clustering between Game type and College")+

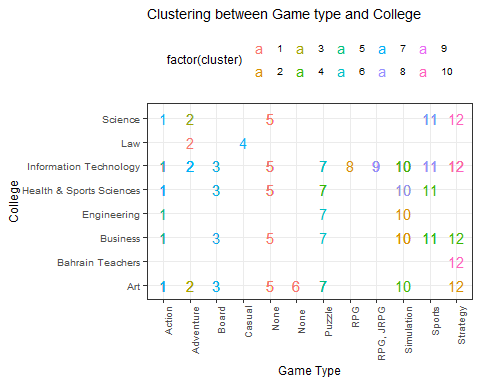
  theme(text = element\_text(size=9),

        axis.text.x = element\_text(angle=90,hjust=1))+

  theme(legend.position="top", legend.box = "horizontal")+

  labs(y = "College", x="Game Type")

b



c<-df1 %>%

  as\_tibble() %>%

  mutate(cluster = km$cluster,

         state = row.names(Game.type)) %>%

  ggplot(aes(df$Game.type, df$Year.Level, color = factor(cluster), label = Game.type)) +

  geom\_text()+

  theme\_bw()+ ggtitle("Clustering between Game type and Year Level")+

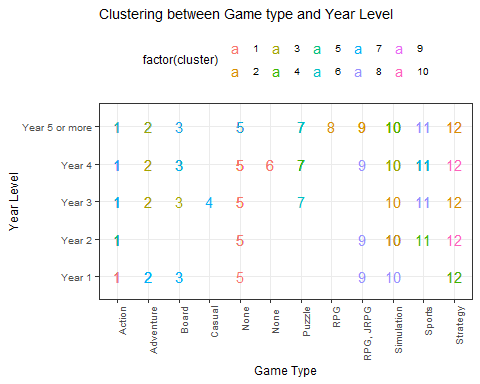
  theme(text = element\_text(size=9),

        axis.text.x = element\_text(angle=90,hjust=1))+

  theme(legend.position="top", legend.box = "horizontal")+

  labs(y = "Year Level", x="Game Type")

c



d<-df1 %>%

  as\_tibble() %>%

  mutate(cluster = km$cluster,

         state = row.names(Game.type)) %>%

  ggplot(aes(df$Game.type, df$GPA, color = factor(cluster), label = Game.type)) +

  geom\_text()+

  theme\_bw()+ ggtitle("Clustering between Game type and GPA")+

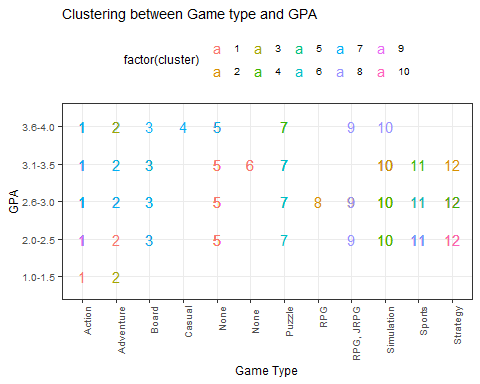
  theme(text = element\_text(size=9),

        axis.text.x = element\_text(angle=90,hjust=1))+

  theme(legend.position="top", legend.box = "horizontal")+

  labs(y = "GPA", x="Game Type")

d



library(gridExtra)

##

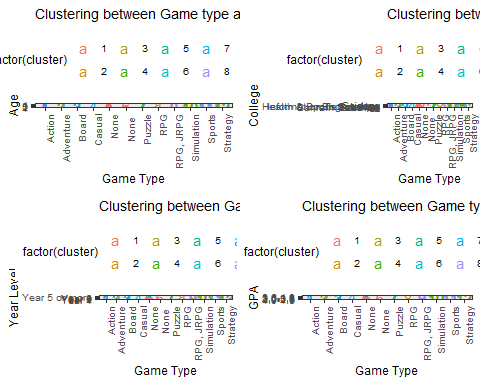
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':

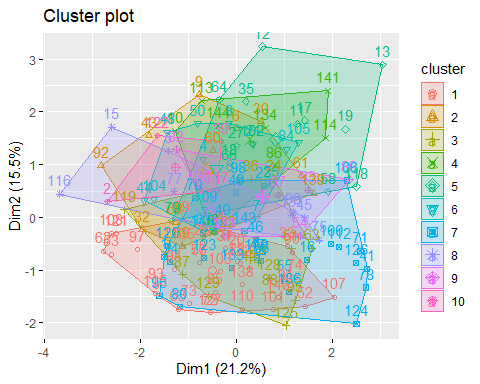
##

##     combine

grid.arrange(a,b,c,d, nrow = 2)



fviz\_cluster(km, data = df1)



#Principal Component analysis

df1.pca <- prcomp(df1, center = TRUE,scale. = TRUE)

summary(df1.pca)

## Importance of components:

##                           PC1    PC2    PC3    PC4    PC5     PC6     PC7

## Standard deviation     1.3808 1.1792 1.0771 1.0153 0.9986 0.93148 0.84361

## Proportion of Variance 0.2118 0.1545 0.1289 0.1145 0.1108 0.09641 0.07907

## Cumulative Proportion  0.2118 0.3664 0.4953 0.6098 0.7206 0.81701 0.89609

##                            PC8     PC9

## Standard deviation     0.78323 0.56724

## Proportion of Variance 0.06816 0.03575

## Cumulative Proportion  0.96425 1.00000

#library(devtools)

#install\_github("vqv/ggbiplot")

library(ggbiplot)

## Loading required package: 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

## Loading required package: scales

##

## Attaching package: 'scales'

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

##

##     discard

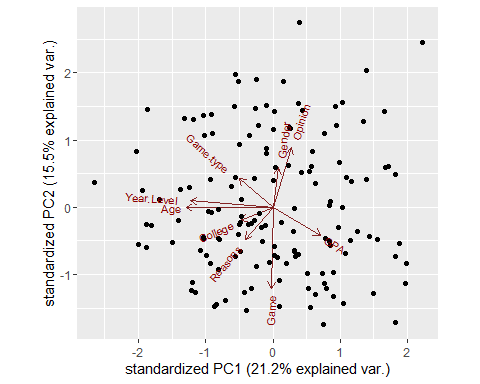
## The following object is masked from 'package:readr':

##

##     col\_factor

## Loading required package: grid

ggbiplot(df1.pca)



table(df$Game.type)

##

##     Action  Adventure      Board     Casual       None      None      Puzzle

##         47         10         11          1         31          2         10

##        RPG  RPG, JRPG Simulation     Sports   Strategy

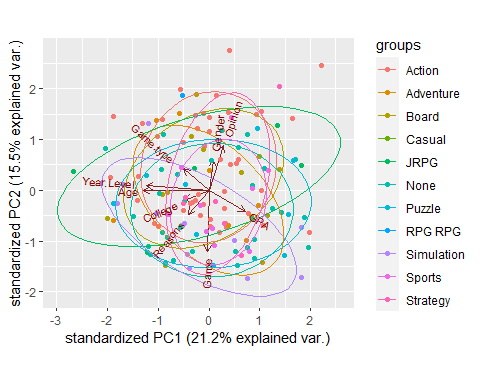
##          1          5         11         10          7

df1.gametype <- c(rep("Action", 47), rep("Adventure",10), rep("Board", 11),

                    rep("Casual",1), rep("None", 31),rep("None", 2), rep("Puzzle",10), rep("RPG RPG", 1),

                    rep("JRPG", 5),rep("Simulation", 11),rep("Sports", 10),rep("Strategy", 7))

ggbiplot(df1.pca,ellipse=TRUE,  labels=rownames(df$Game.type), groups=df1.gametype)



#Regression Analysis

model<-lm(Game.type~.,df1)

summary.aov(model)

##              Df Sum Sq Mean Sq F value  Pr(>F)

## College       1   15.2   15.17   1.146 0.28627

## Age           1   44.9   44.92   3.393 0.06763 .

## Gender        1    0.2    0.22   0.017 0.89794

## Year.Level    1    0.2    0.19   0.014 0.90511

## Game          1   93.1   93.09   7.032 0.00895 \*\*

## Reasons       1   10.3   10.27   0.776 0.37994

## Opinion       1   11.0   10.98   0.829 0.36403

## GPA           1    0.4    0.39   0.029 0.86451

## Residuals   137 1813.8   13.24

## ---

## Signif. codes:  0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(model)

##

## Call:

## lm(formula = Game.type ~ ., data = df1)

##

## Residuals:

##     Min      1Q  Median      3Q     Max

## -6.4820 -2.6640 -0.3397  2.0754  7.8528

##

## Coefficients:

##             Estimate Std. Error t value Pr(>|t|)

## (Intercept)  4.88836    2.29878   2.126  0.03525 \*

## College      0.13027    0.14585   0.893  0.37331

## Age          0.76571    0.55520   1.379  0.17010

## Gender      -0.37401    0.75185  -0.497  0.61967

## Year.Level  -0.01647    0.31730  -0.052  0.95868

## Game        -0.10682    0.03771  -2.833  0.00531 \*\*

## Reasons      0.01971    0.02482   0.794  0.42847

## Opinion     -0.22367    0.24820  -0.901  0.36909

## GPA          0.05300    0.31004   0.171  0.86451

## ---

## Signif. codes:  0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

##

## Residual standard error: 3.639 on 137 degrees of freedom

## Multiple R-squared:  0.0881, Adjusted R-squared:  0.03485

## F-statistic: 1.655 on 8 and 137 DF,  p-value: 0.1151

par(mfrow=c(2,2))

plot(model)

