Data\_clust = props  
#scaling the data and finding generalized euclidean distance  
scale\_data = scale(Data\_clust)  
scale\_data

## Energy Dancebility Loudness Valence Acoustiveness  
## [1,] 1.15021860 0.18804631 0.8986926 1.23159462 0.22115398  
## [2,] 1.40011168 0.79630428 0.2877298 0.51918924 0.46134664  
## [3,] 0.83785224 0.87233653 1.4332851 0.83086660 -0.21119281  
## [4,] 1.33763841 0.41614305 0.8986926 0.83086660 -0.69157813  
## [5,] 0.83785224 -0.04005043 0.2877298 -0.41584281 -0.59550107  
## [6,] 0.96279878 0.64423979 0.2877298 0.07393588 -0.49942401  
## [7,] 0.46301261 0.79630428 0.8986926 1.32064529 -0.69157813  
## [8,] 0.33806606 -0.95243739 -0.3232331 -0.63846949 -0.35530841  
## [9,] -2.09839153 -1.25656638 -1.4687884 -1.70707755 2.86327326  
## [10,] 0.08817298 1.10043327 0.8986926 0.38561324 -0.06707721  
## [11,] 1.02527205 -0.19211492 0.8986926 -0.23774147 -0.54746254  
## [12,] 0.77537897 -0.19211492 0.2877298 -0.19321613 0.89369343  
## [13,] 0.71290569 0.94836877 0.2877298 0.47466391 0.17311545  
## [14,] 0.77537897 1.40456225 -0.3232331 0.83086660 -0.64353960  
## [15,] 0.83785224 -1.56069536 -1.4687884 1.14254395 -0.64353960  
## [16,] 0.27559279 1.02440102 -0.3232331 -0.86109617 0.26919251  
## [17,] -0.59903301 0.56820754 0.8986926 1.36517063 -0.45138547  
## [18,] 0.65043242 1.32853001 -1.4687884 -0.37131748 -0.35530841  
## [19,] 0.58795915 -0.11608268 -0.8578255 -0.68299483 -0.69157813  
## [20,] -0.59903301 1.40456225 0.8986926 1.05349328 -0.64353960  
## [21,] -0.47408647 1.17646551 -0.8578255 -0.54941882 -0.69157813  
## [22,] -0.16172011 -1.56069536 0.2877298 -0.32679214 -0.64353960  
## [23,] -0.16172011 0.64423979 0.2877298 0.96444260 -0.69157813  
## [24,] 0.65043242 0.41614305 0.2877298 0.91991727 -0.69157813  
## [25,] 1.52505822 -0.87640514 0.8986926 0.56371458 0.55742371  
## [26,] 0.77537897 -1.25656638 -0.3232331 0.96444260 -0.69157813  
## [27,] 1.21269187 -0.26814717 0.2877298 1.58779731 -0.59550107  
## [28,] 1.46258495 0.03598182 1.4332851 0.91991727 -0.59550107  
## [29,] 0.58795915 0.56820754 0.2877298 0.29656256 -0.59550107  
## [30,] 0.27559279 1.10043327 0.8986926 -0.54941882 -0.69157813  
## [31,] 0.02569971 0.41614305 2.0442479 1.18706928 -0.45138547  
## [32,] 1.15021860 0.26407856 0.8986926 1.36517063 -0.64353960  
## [33,] 0.83785224 0.79630428 1.4332851 -0.01511479 -0.30726988  
## [34,] -1.16129245 -0.34417942 -0.8578255 -0.50489348 0.17311545  
## [35,] 0.71290569 -1.10450188 0.2877298 0.11846122 0.07703838  
## [36,] -0.47408647 1.25249776 0.2877298 1.23159462 -0.30726988  
## [37,] -1.84849844 -2.70117906 0.2877298 -2.01875491 3.29562005  
## [38,] 1.27516514 0.03598182 0.8986926 1.18706928 -0.64353960  
## [39,] 1.27516514 0.56820754 -0.3232331 1.40969596 -0.59550107  
## [40,] 0.65043242 -1.10450188 0.2877298 -1.17277352 -0.54746254  
## [41,] 0.90032551 -1.02846964 0.8986926 0.87539193 1.56623288  
## [42,] -1.84849844 0.49217530 -0.8578255 -1.88517890 -0.49942401  
dist\_data = dist(scale\_data,method ="euclidean")  
dist\_data

## 1 2 3 4 5 6  
## 2 1.1708603   
## 3 1.0950441 1.4777231   
## 4 1.0396236 1.3957361 0.9871463   
## 5 1.9758168 1.7340233 1.9613067 1.5474784   
## 6 1.5735058 1.1557236 1.4268700 1.0842858 0.8561418   
## 7 1.2973973 1.7952709 0.9501200 1.0720896 2.0586139 1.4957918  
## 8 2.6986748 2.5625013 2.9739973 2.5762930 1.2501068 1.9599070  
## 9 5.8191003 5.5013338 6.1212162 6.2679625 5.1785328 5.5271193  
## 10 1.6610697 1.5760798 1.0577599 1.6179212 1.7769528 1.2768809  
## 11 1.7058294 1.7553897 1.6458957 1.2768097 0.6823146 1.0844898  
## 12 1.7722014 1.4358651 2.1722743 2.1497024 1.5146912 1.6573357  
## 13 1.3106717 0.7618631 1.2681730 1.3861479 1.5415439 0.8762610  
## 14 2.0054416 1.5657141 1.8866496 1.6698857 2.0051624 1.2570622  
## 15 3.0848438 3.2505821 3.8243615 3.1403969 2.7979625 3.0208276  
## 16 2.7091003 1.9057695 2.5530616 2.6032363 1.6638551 1.5668780  
## 17 1.9169130 2.4434909 1.6691603 2.0290375 2.4496267 2.1184829   
## 18 3.1712918 2.3217594 3.1829679 2.9099815 2.2479256 1.9672828  
## 19 2.8271397 2.3619527 2.9683587 2.4944770 1.2087702 1.6250413  
## 20 2.3063407 2.4992173 1.6941602 2.1862157 2.5857449 2.0906904  
## 21 3.2719695 2.7277458 3.0329562 2.9750598 2.1308366 2.0213101  
## 22 2.8859696 3.1513845 3.1238091 2.8056591 1.8225651 2.5115003  
## 23 1.7908158 1.9974805 1.6161939 1.6404945 1.8389716 1.4472277  
## 24 1.2670093 1.4820122 1.3394908 0.9238277 1.4271386 0.9498466  
## 25 1.3537735 1.7883113 2.1161947 1.8267958 1.9580054 2.0877219  
## 26 2.1505727 2.5505550 2.8053697 2.1505900 1.9420371 2.2025874  
## 27 1.1743495 1.8512248 1.8639050 1.1997146 2.0511233 1.7877091  
## 28 1.0819213 1.7809548 1.1159763 0.6805029 1.8688563 1.6300234  
## 29 1.5411503 1.3704414 1.3785542 1.1194328 0.9695080 0.4528566  
## 30 2.3669620 2.0497491 1.6703026 1.8711980 1.4202773 1.2161967  
## 31 1.7558928 2.5294328 1.1939651 1.7938933 2.5579595 2.2927958  
## 32 0.8782474 1.6295535 1.1071333 0.5882502 1.9333027 1.4969906  
## 33 1.6083705 1.5826196 0.8548077 1.2423668 1.5018144 1.1814760  
## 34 3.4248215 3.2102358 3.5577558 3.5270457 2.4495045 2.7547025

(kmeans5 <- kmeans(scale\_data,5,nstart = 20))

## K-means clustering with 5 clusters of sizes 66, 129, 45, 192, 166  
##   
## Cluster means:  
## Energy Dancebility Loudness Valence Acoustiveness  
## 1 -0.9047734 -0.6644365 -0.9052677 -1.2982540 -0.01394369  
## 2 -0.4978166 0.8104498 -0.5973375 0.2371954 -0.18884931  
## 3 -2.0692373 -1.1247771 -1.3482929 -1.0233663 2.83978775  
## 4 0.6780899 0.5175194 0.6759457 0.9062350 -0.27324258  
## 5 0.5232278 -0.6593010 0.4078061 -0.4389102 -0.30148210  
##   
## Clustering vector:  
## [1] 4 4 4 4 5 4 4 5 3 4 5 5 4 4 5 2 4 2 5 4 2 5 4 4 5 5 4 4 4 4 4 4 4 1 5  
## [36] 2 3 4 4 5 5 1 5 4 3 4 4 4 5 4 1 3 4 5 4 4 4 4 1 4 4 5 1 5 5 5 4 4 2 1  
## [71] 4 4 4 5 4 5 4 2 2 4 5 4 5 4 4 5 3 5 4 4 4 4 4 3 5 3 2 2 4 5 4 4 2 5 4  
## [106] 1 4 4 2 2 4 5 2 4 2 4 4 4 4 5 4 2 2 4 2 5 5 4 4 5 4 5 5 4 4 4 5 5 5 1  
## [141] 2 5 5 2 5 2 1 5 2 5 2 4 5 5 5 5 2 5 4 5 2 4 4 4 4 1 1 2 1 4 2 2 4 4 5  
## [176] 5 2 4 5 5 1 2 5 5 3 5 2 5 4 5 3 5 5 4 5 4 4 5 5 4 2 5 2 5 4 1 4 5 3 3  
## [211] 4 4 4 4 1 5 4 4 1 4 2 2 5 5 5 2 5 5 2 2 2 1 5 2 4 2 4 2 4 1 4 4 5 1 5  
## [246] 4 5 3 1 4 5 1 5 3 1 2 4 4 3 1 2 2 5 2 2 2 3 1 3 2 3 4 5 3 1 1 2 2 2 4  
## [281] 2 3 5 5 5 5 5 4 5 2 5 2 1 4 2 4 5 4 5 4 5 4 4 5 5 4 4 5 5 5 4 1 5 5 4  
## [316] 4 5 5 4 4 4 4 2 4 1 5 3 3 2 5 4 2 5 4 4 4 1 5 1 5 1 5 1 5 1 4 5 4 1 3  
## [351] 5 4 4 4 5 4 3 4 2 1 3 4 5 1 5 5 3 1 2 4 2 5 1 2 2 2 2 1 2 4 2 5 2 2 5  
## [386] 2 4 2 5 4 1 1 5 2 3 5 5 4 1 5 5 4 5 5 5 5 5 3 5 4 5 5 5 2 4 4 1 4 3 1  
## [421] 3 2 5 5 4 4 4 3 4 5 1 2 3 5 2 2 4 2 4 2 2 5 2 5 2 5 2 2 2 5 2 4 1 4 4  
## [456] 3 4 4 1 1 5 4 4 2 5 2 3 5 3 2 2 1 2 5 4 4 4 2 4 5 2 1 4 4 2 5 4 2 1 5  
## [491] 4 1 4 4 3 4 2 2 4 5 3 4 5 4 2 4 2 4 2 2 2 5 1 2 5 5 4 5 2 3 4 2 4 3 4  
## [526] 1 4 4 2 5 2 1 2 3 2 5 2 5 1 4 1 2 2 2 1 2 2 5 5 5 2 1 4 1 3 3 5 4 2 2  
## [561] 5 1 2 2 5 2 4 3 3 3 2 5 2 3 4 5 4 2 2 2 4 5 4 2 4 1 4 2 5 1 4 4 4 1 2  
## [596] 5 5 1  
##   
## Within cluster sum of squares by cluster:  
## [1] 162.1891 267.8626 208.7179 297.5688 292.4047  
## (between\_SS / total\_SS = 58.8 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss"   
## [5] "tot.withinss" "betweenss" "size" "iter"   
## [9] "ifault"

kmeans5

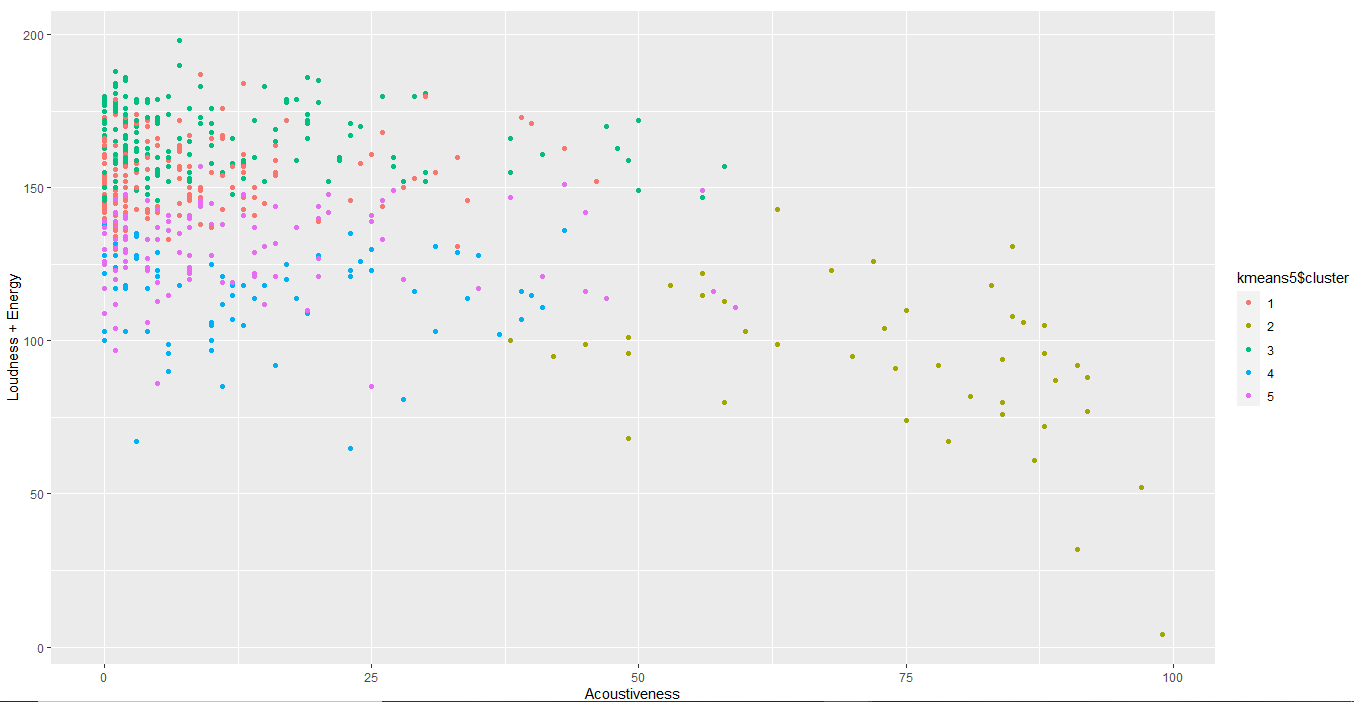
## K-means clustering with 5 clusters of sizes 66, 129, 45, 192, 166  
##   
## Cluster means:  
## Energy Dancebility Loudness Valence Acoustiveness  
## 1 -0.9047734 -0.6644365 -0.9052677 -1.2982540 -0.01394369  
## 2 -0.4978166 0.8104498 -0.5973375 0.2371954 -0.18884931  
## 3 -2.0692373 -1.1247771 -1.3482929 -1.0233663 2.83978775  
## 4 0.6780899 0.5175194 0.6759457 0.9062350 -0.27324258  
## 5 0.5232278 -0.6593010 0.4078061 -0.4389102 -0.30148210  
##   
## Clustering vector:  
## [1] 4 4 4 4 5 4 4 5 3 4 5 5 4 4 5 2 4 2 5 4 2 5 4 4 5 5 4 4 4 4 4 4 4 1 5  
## [36] 2 3 4 4 5 5 1 5 4 3 4 4 4 5 4 1 3 4 5 4 4 4 4 1 4 4 5 1 5 5 5 4 4 2 1  
## [71] 4 4 4 5 4 5 4 2 2 4 5 4 5 4 4 5 3 5 4 4 4 4 4 3 5 3 2 2 4 5 4 4 2 5 4  
## [106] 1 4 4 2 2 4 5 2 4 2 4 4 4 4 5 4 2 2 4 2 5 5 4 4 5 4 5 5 4 4 4 5 5 5 1  
## [141] 2 5 5 2 5 2 1 5 2 5 2 4 5 5 5 5 2 5 4 5 2 4 4 4 4 1 1 2 1 4 2 2 4 4 5  
## [176] 5 2 4 5 5 1 2 5 5 3 5 2 5 4 5 3 5 5 4 5 4 4 5 5 4 2 5 2 5 4 1 4 5 3 3  
## [211] 4 4 4 4 1 5 4 4 1 4 2 2 5 5 5 2 5 5 2 2 2 1 5 2 4 2 4 2 4 1 4 4 5 1 5  
## [246] 4 5 3 1 4 5 1 5 3 1 2 4 4 3 1 2 2 5 2 2 2 3 1 3 2 3 4 5 3 1 1 2 2 2 4  
## [281] 2 3 5 5 5 5 5 4 5 2 5 2 1 4 2 4 5 4 5 4 5 4 4 5 5 4 4 5 5 5 4 1 5 5 4  
## [316] 4 5 5 4 4 4 4 2 4 1 5 3 3 2 5 4 2 5 4 4 4 1 5 1 5 1 5 1 5 1 4 5 4 1 3  
## [351] 5 4 4 4 5 4 3 4 2 1 3 4 5 1 5 5 3 1 2 4 2 5 1 2 2 2 2 1 2 4 2 5 2 2 5  
## [386] 2 4 2 5 4 1 1 5 2 3 5 5 4 1 5 5 4 5 5 5 5 5 3 5 4 5 5 5 2 4 4 1 4 3 1  
## [421] 3 2 5 5 4 4 4 3 4 5 1 2 3 5 2 2 4 2 4 2 2 5 2 5 2 5 2 2 2 5 2 4 1 4 4  
## [456] 3 4 4 1 1 5 4 4 2 5 2 3 5 3 2 2 1 2 5 4 4 4 2 4 5 2 1 4 4 2 5 4 2 1 5  
## [491] 4 1 4 4 3 4 2 2 4 5 3 4 5 4 2 4 2 4 2 2 2 5 1 2 5 5 4 5 2 3 4 2 4 3 4  
## [526] 1 4 4 2 5 2 1 2 3 2 5 2 5 1 4 1 2 2 2 1 2 2 5 5 5 2 1 4 1 3 3 5 4 2 2  
## [561] 5 1 2 2 5 2 4 3 3 3 2 5 2 3 4 5 4 2 2 2 4 5 4 2 4 1 4 2 5 1 4 4 4 1 2  
## [596] 5 5 1  
##   
## Within cluster sum of squares by cluster:  
## [1] 162.1891 267.8626 208.7179 297.5688 292.4047  
## (between\_SS / total\_SS = 58.8 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss"   
## [5] "tot.withinss" "betweenss" "size" "iter"   
## [9] "ifault"

kmeans5$centers

## Energy Dancebility Loudness Valence Acoustiveness  
## 1 -0.9047734 -0.6644365 -0.9052677 -1.2982540 -0.01394369  
## 2 -0.4978166 0.8104498 -0.5973375 0.2371954 -0.18884931  
## 3 -2.0692373 -1.1247771 -1.3482929 -1.0233663 2.83978775  
## 4 0.6780899 0.5175194 0.6759457 0.9062350 -0.27324258  
## 5 0.5232278 -0.6593010 0.4078061 -0.4389102 -0.30148210

kmeans5$cluster <- as.factor(kmeans5$cluster)

ggplot(Data\_clust, aes(Acoustiveness,Loudness+Energy,color = kmeans5$cluster)) + geom\_point()



#To validate our assumption we took the help of the nbclust function t find optimal no. of clusters  
nb\_clust = NbClust(Data\_num, distance="euclidean", method = 'kmeans')

## \*\*\* : The Hubert index is a graphical method of determining the number of clusters.  
## In the plot of Hubert index, we seek a significant knee that corresponds to a   
## significant increase of the value of the measure i.e the significant peak in Hubert  
## index second differences plot.   
##

## \*\*\* : The D index is a graphical method of determining the number of clusters.   
## In the plot of D index, we seek a significant knee (the significant peak in Dindex  
## second differences plot) that corresponds to a significant increase of the value of  
## the measure.   
##   
## \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   
## \* Among all indices:   
## \* 9 proposed 2 as the best number of clusters   
## \* 2 proposed 3 as the best number of clusters   
## \* 8 proposed 4 as the best number of clusters   
## \* 1 proposed 6 as the best number of clusters   
## \* 1 proposed 8 as the best number of clusters   
## \* 2 proposed 10 as the best number of clusters   
## \* 1 proposed 15 as the best number of clusters   
##   
## \*\*\*\*\* Conclusion \*\*\*\*\*   
##   
## \* According to the majority rule, the best number of clusters is 2   
##   
##   
## \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

nb\_clust

## $All.index  
## KL CH Hartigan CCC Scott Marriot TrCovW  
## 2 5.3459 179.3297 75.3766 56.1026 3829.165 8.603133e+41 58272462097  
## 3 0.2856 138.4604 87.0761 33.8910 4201.822 1.038003e+42 45365031712  
## 4 2.0534 134.6145 55.1959 32.1655 4777.885 7.042283e+41 33253803097  
## 5 0.6527 123.9324 61.3638 32.2060 5076.744 6.675573e+41 29563036522  
## 6 2.8979 121.4731 35.0111 32.6460 5306.351 6.547867e+41 21226652533  
## 7 0.9009 112.8585 33.2984 32.0953 5480.079 6.665369e+41 17811320916  
## 8 1.0121 106.7621 30.9208 31.8175 5660.363 6.439886e+41 15201455326  
## 9 2.1497 102.0046 21.9362 31.6813 5836.477 6.071298e+41 13152384646  
## 10 7.9626 96.3211 14.9572 31.2358 6007.672 5.629462e+41 12004628169  
## 11 0.0811 90.2362 22.6243 30.5508 6126.232 5.586612e+41 11916052845  
## 12 2.2671 87.1027 16.1458 30.4657 6300.023 4.971763e+41 10549607032  
## 13 0.4526 83.2472 20.5650 30.1151 6404.077 4.903047e+41 10032125808  
## 14 1.4850 80.9882 16.6940 30.1285 6558.340 4.393420e+41 9387626937  
## 15 1.0567 78.4110 15.7136 29.9979 6657.297 4.274277e+41 8757460252  
## TraceW Friedman Rubin Cindex DB Silhouette Duda Pseudot2  
## 2 1589169.3 301.3525 33.8525 0.2470 1.7153 0.2742 1.5463 -131.0656  
## 3 1410750.6 333.0064 38.1338 0.2471 2.0069 0.1433 1.9614 -231.3512  
## 4 1230649.5 331.1899 43.7146 0.2889 1.5045 0.2257 1.2040 -39.1414  
## 5 1126017.2 333.4215 47.7766 0.2889 1.6756 0.1538 1.1054 -17.0646  
## 6 1020423.6 372.2858 52.7206 0.2798 1.5559 0.1669 1.5912 -66.8810  
## 7 963445.0 396.5807 55.8385 0.2713 1.6256 0.1477 1.4828 -51.4444  
## 8 912057.5 448.2672 58.9846 0.3287 1.6067 0.1489 1.4769 -61.6751  
## 9 866638.6 455.5597 62.0758 0.3207 1.6153 0.1461 1.4585 -39.2941  
## 10 835521.2 464.2882 64.3877 0.3493 1.6406 0.1463 1.7726 -51.8654  
## 11 814794.9 478.6988 66.0256 0.3227 1.5870 0.1432 1.7556 -60.2545  
## 12 784556.4 482.9828 68.5704 0.3339 1.6456 0.1419 1.4011 -38.9332  
## 13 763519.4 502.1824 70.4596 0.3313 1.6750 0.1357 1.5328 -33.0229  
## 14 737590.3 501.5172 72.9366 0.3247 1.6600 0.1398 1.6691 -56.5250  
## 15 717091.8 531.6740 75.0215 0.3290 1.6721 0.1335 1.3933 -30.2022  
## Beale Ratkowsky Ball Ptbiserial Frey McClain Dunn Hubert  
## 2 -1.8555 0.2332 794584.67 0.4533 2.7603 0.4451 0.0644 0  
## 3 -2.5674 0.2395 470250.21 0.3573 -0.8841 1.2506 0.0364 0  
## 4 -0.8889 0.2677 307662.36 0.4985 2.7917 0.9745 0.0715 0  
## 5 -0.5008 0.2590 225203.45 0.3802 -0.1707 2.1328 0.0662 0  
## 6 -1.9515 0.2473 170070.59 0.4107 0.9709 2.1215 0.0503 0  
## 7 -1.7079 0.2323 137635.00 0.3757 0.0659 2.8381 0.0491 0  
## 8 -1.6933 0.2199 114007.18 0.3813 0.8241 2.9375 0.0751 0  
## 9 -1.6448 0.2120 96293.18 0.3530 -0.0418 3.6926 0.0609 0  
## 10 -2.2795 0.2067 83552.12 0.3598 0.5527 3.7059 0.0881 0  
## 11 -2.2515 0.1983 74072.27 0.3483 0.2852 4.1047 0.0633 0  
## 12 -1.4973 0.1968 65379.70 0.3392 1.4791 4.6196 0.0691 0  
## 13 -1.8158 0.1907 58732.26 0.3165 0.0595 5.4192 0.0629 0  
## 14 -2.0909 0.1883 52685.02 0.3184 0.6919 5.5541 0.0646 0  
## 15 -1.4711 0.1826 47806.12 0.3043 0.3738 6.2122 0.0691 0  
## SDindex Dindex SDbw  
## 2 0.0986 47.6936 1.2421  
## 3 0.1068 44.9709 1.3984  
## 4 0.0875 42.5334 0.9640  
## 5 0.0946 40.3459 0.7546  
## 6 0.1013 38.8089 0.6189  
## 7 0.1040 37.5770 0.5528  
## 8 0.1015 36.8551 0.5283  
## 9 0.1014 35.8684 0.4874  
## 10 0.0960 35.2885 0.4696  
## 11 0.1065 34.9389 0.4637  
## 12 0.1043 34.1707 0.4421  
## 13 0.1068 33.6281 0.4212  
## 14 0.1069 33.1544 0.4140  
## 15 0.1112 32.6511 0.3936  
##   
## $All.CriticalValues  
## CritValue\_Duda CritValue\_PseudoT2 Fvalue\_Beale  
## 2 0.8009 92.2383 1  
## 3 0.7777 134.8898 1  
## 4 0.7899 61.4499 1  
## 5 0.8034 43.8029 1  
## 6 0.8009 44.7518 1  
## 7 0.7894 42.1538 1  
## 8 0.7869 51.7315 1  
## 9 0.7695 37.4437 1  
## 10 0.7664 36.2680 1  
## 11 0.7680 42.2963 1  
## 12 0.7664 41.4491 1  
## 13 0.7578 30.3618 1  
## 14 0.7476 47.6112 1  
## 15 0.7429 37.0307 1  
##   
## $Best.nc  
## KL CH Hartigan CCC Scott Marriot  
## Number\_clusters 10.0000 2.0000 4.0000 2.0000 4.0000 4.000000e+00  
## Value\_Index 7.9626 179.3297 31.8802 56.1026 576.0629 2.971041e+41  
## TrCovW TraceW Friedman Rubin Cindex DB  
## Number\_clusters 3 4.00 8.0000 6.000 2.000 4.0000  
## Value\_Index 12907430385 75468.95 51.6865 -1.826 0.247 1.5045  
## Silhouette Duda PseudoT2 Beale Ratkowsky Ball  
## Number\_clusters 2.0000 2.0000 2.0000 2.0000 4.0000 3.0  
## Value\_Index 0.2742 1.5463 -131.0656 -1.8555 0.2677 324334.5  
## PtBiserial Frey McClain Dunn Hubert SDindex Dindex  
## Number\_clusters 4.0000 2.0000 2.0000 10.0000 0 4.0000 0  
## Value\_Index 0.4985 2.7603 0.4451 0.0881 0 0.0875 0  
## SDbw  
## Number\_clusters 15.0000  
## Value\_Index 0.3936  
##   
## $Best.partition  
## [1] 2 1 2 1 2 2 2 2 1 2 2 2 2 2 2 1 2 1 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 1 1  
## [36] 2 1 2 1 1 2 1 1 2 1 2 2 2 1 2 1 1 2 2 2 2 2 1 2 2 2 1 1 2 1 1 2 2 1 2  
## [71] 1 1 2 2 2 1 2 2 2 2 1 2 2 2 2 1 1 2 2 2 1 2 2 1 1 1 2 2 2 1 2 2 2 1 2  
## [106] 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1  
## [141] 1 2 2 2 2 2 1 2 1 2 2 2 1 2 2 2 1 2 2 2 1 2 2 2 2 1 1 2 2 2 1 1 2 2 2  
## [176] 2 2 2 2 2 1 1 2 2 1 1 1 2 2 2 1 2 1 2 1 2 2 2 2 2 1 1 2 2 2 1 2 1 1 2  
## [211] 2 2 2 2 2 1 2 2 1 2 2 2 1 1 1 2 1 2 2 2 1 1 2 2 2 2 2 2 2 1 2 2 2 2 1  
## [246] 2 2 1 1 2 2 2 2 1 1 1 2 2 2 1 2 1 2 2 2 1 1 1 1 1 1 2 2 2 1 1 2 2 2 2  
## [281] 2 1 2 2 1 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2  
## [316] 2 1 2 2 2 2 2 2 2 1 2 2 2 1 1 2 2 1 2 2 2 1 2 1 2 1 2 1 2 2 2 2 2 1 1  
## [351] 1 2 2 2 2 2 1 2 1 1 1 2 2 2 1 2 2 2 1 2 2 2 1 2 2 1 2 1 2 2 2 2 1 2 2  
## [386] 2 2 2 1 2 2 1 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 1 2 2 2 1 2 2 2 2 1 2 1 1  
## [421] 2 2 2 2 2 2 2 1 1 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 1 2 2 2 2 2 1 1 1 2 1  
## [456] 1 2 2 1 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 1 2 2 2 2  
## [491] 2 2 2 2 1 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 1 2  
## [526] 1 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 1 2 2 1 2 2 2 1 2 2 1 1 2 2 2 1  
## [561] 2 2 2 1 2 1 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2  
## [596] 1 2 2

# for 2 clusters  
(kmeans2 <- kmeans(scale\_data,2,nstart = 10))

## K-means clustering with 2 clusters of sizes 110, 488  
##   
## Cluster means:  
## Energy Dancebility Loudness Valence Acoustiveness  
## 1 -1.4855855 -0.7070606 -1.240372 -1.0258399 1.3138122  
## 2 0.3348656 0.1593784 0.279592 0.2312344 -0.2961462  
##   
## Clustering vector:  
## [1] 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2  
## [36] 2 1 2 2 2 2 1 2 2 1 2 2 2 2 2 1 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 1  
## [71] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2  
## [106] 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2  
## [141] 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 1 2 1 2 2 2 2 2 2  
## [176] 2 2 2 2 2 1 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 1  
## [211] 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 1 2 2 2 2 2  
## [246] 2 2 1 2 2 2 1 2 1 2 2 2 2 1 1 1 1 2 1 2 1 1 1 1 2 1 2 2 1 1 1 1 2 2 2  
## [281] 1 1 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2  
## [316] 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 1 2 1 2 1 2 1 2 2 2 2 2 1 1  
## [351] 2 2 2 2 2 2 1 2 2 1 1 2 2 1 2 2 1 1 1 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2  
## [386] 2 2 2 2 2 1 2 2 2 1 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 2 1 1  
## [421] 1 2 2 2 2 2 2 1 2 2 1 2 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2  
## [456] 1 2 2 1 2 1 2 2 2 2 2 1 2 1 2 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2  
## [491] 2 1 2 2 1 2 1 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 1 2  
## [526] 2 2 2 1 2 1 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 1 2 1 1 1 2 2 2 2  
## [561] 2 2 2 2 2 2 2 1 1 1 2 2 2 1 2 2 2 2 2 2 2 2 2 1 2 1 2 1 2 1 2 2 2 2 2  
## [596] 2 2 1  
##   
## Within cluster sum of squares by cluster:  
## [1] 648.3949 1389.8219  
## (between\_SS / total\_SS = 31.7 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss"   
## [5] "tot.withinss" "betweenss" "size" "iter"   
## [9] "ifault"

kmeans2

## K-means clustering with 2 clusters of sizes 110, 488  
##   
## Cluster means:  
## Energy Dancebility Loudness Valence Acoustiveness  
## 1 -1.4855855 -0.7070606 -1.240372 -1.0258399 1.3138122  
## 2 0.3348656 0.1593784 0.279592 0.2312344 -0.2961462  
##   
## Clustering vector:  
## [1] 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2  
## [36] 2 1 2 2 2 2 1 2 2 1 2 2 2 2 2 1 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 1  
## [71] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2  
## [106] 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2  
## [141] 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 1 2 1 2 2 2 2 2 2  
## [176] 2 2 2 2 2 1 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 1  
## [211] 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 1 2 2 2 2 2  
## [246] 2 2 1 2 2 2 1 2 1 2 2 2 2 1 1 1 1 2 1 2 1 1 1 1 2 1 2 2 1 1 1 1 2 2 2  
## [281] 1 1 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2  
## [316] 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 1 2 1 2 1 2 1 2 2 2 2 2 1 1  
## [351] 2 2 2 2 2 2 1 2 2 1 1 2 2 1 2 2 1 1 1 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2  
## [386] 2 2 2 2 2 1 2 2 2 1 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 2 1 1  
## [421] 1 2 2 2 2 2 2 1 2 2 1 2 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2  
## [456] 1 2 2 1 2 1 2 2 2 2 2 1 2 1 2 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2  
## [491] 2 1 2 2 1 2 1 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 1 2  
## [526] 2 2 2 1 2 1 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 1 2 1 1 1 2 2 2 2  
## [561] 2 2 2 2 2 2 2 1 1 1 2 2 2 1 2 2 2 2 2 2 2 2 2 1 2 1 2 1 2 1 2 2 2 2 2  
## [596] 2 2 1  
##   
## Within cluster sum of squares by cluster:  
## [1] 648.3949 1389.8219  
## (between\_SS / total\_SS = 31.7 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss"   
## [5] "tot.withinss" "betweenss" "size" "iter"   
## [9] "ifault"

kmeans2$cluster <- as.factor(kmeans2$cluster)  
  
kmeans2$centers

## Energy Dancebility Loudness Valence Acoustiveness  
## 1 -1.4855855 -0.7070606 -1.240372 -1.0258399 1.3138122  
## 2 0.3348656 0.1593784 0.279592 0.2312344 -0.2961462

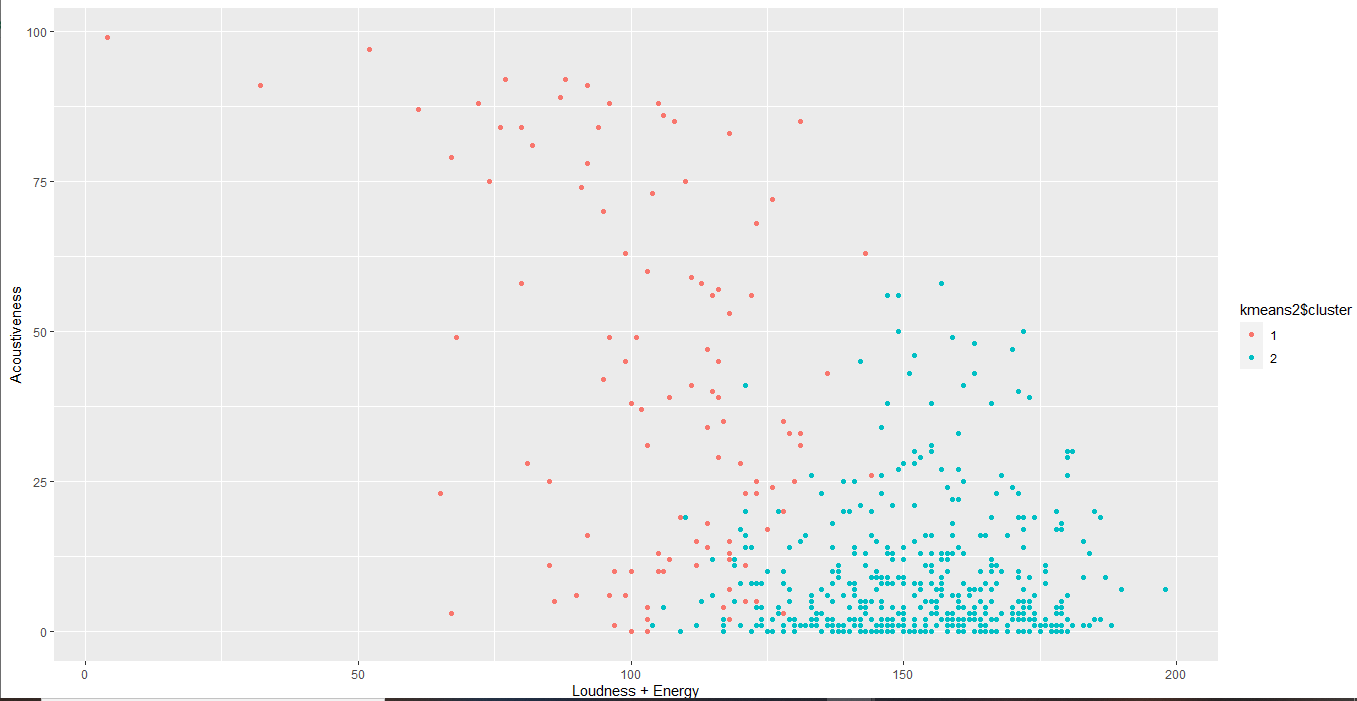
kmeans2$withinss

## [1] 648.3949 1389.8219

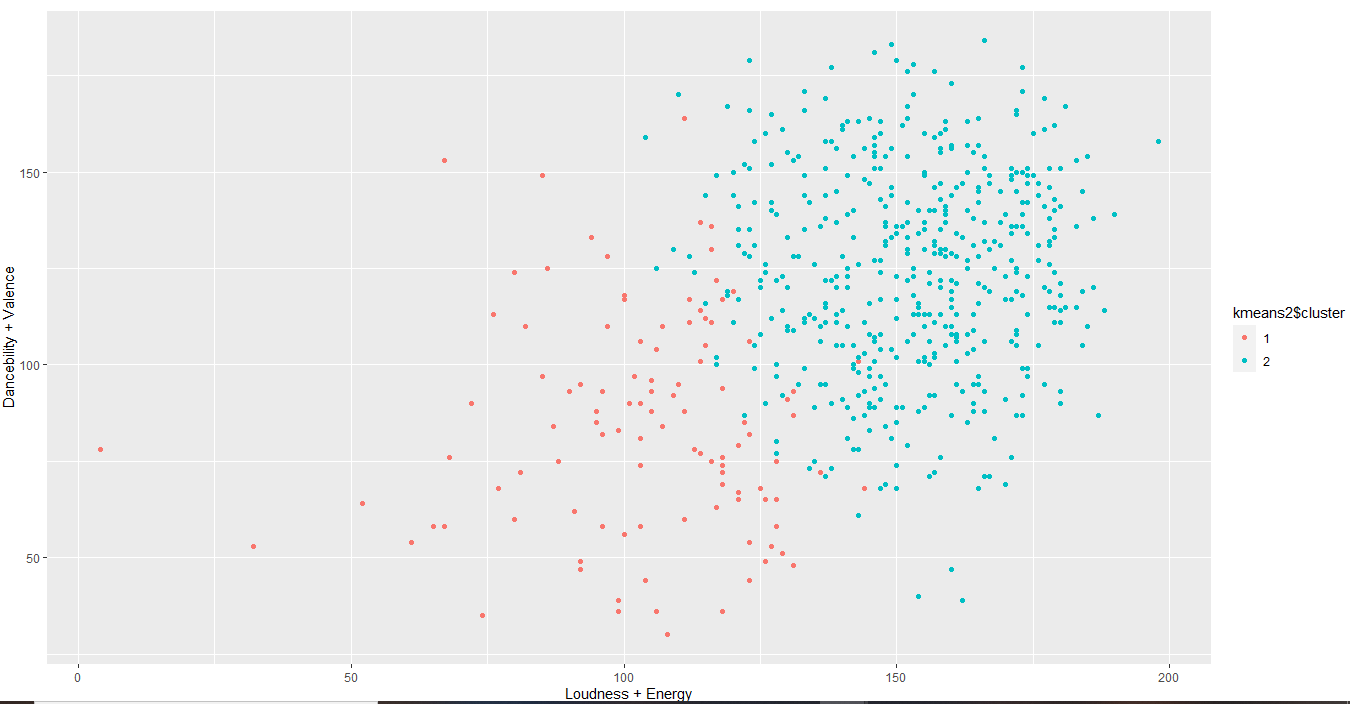
kmeans2$size

## [1] 110 488

ggplot(Data\_clust, aes(Loudness+Energy,Acoustiveness,color = kmeans2$cluster)) + geom\_point()



ggplot(Data\_clust, aes(Loudness+Energy,Dancebility+Valence,color = kmeans2$cluster)) + geom\_point()



ggplot(Data\_clust, aes(Acoustiveness,Dancebility+Valence,color = kmeans2$cluster)) + geom\_point()

