Case Study Samsung Data

Abel Isaias Gutierrez-Cruz

9/9/2021

Data exploring

```
names(samsungData)[1:12]
```

```
##  [1] "tBodyAcc-mean()-X" "tBodyAcc-mean()-Y" "tBodyAcc-mean()-Z"
##  [4] "tBodyAcc-std()-X"  "tBodyAcc-std()-Y"  "tBodyAcc-std()-Z"
##  [7] "tBodyAcc-mad()-X"  "tBodyAcc-mad()-Y"  "tBodyAcc-mad()-Z"
## [10] "tBodyAcc-max()-X"  "tBodyAcc-max()-Y"  "tBodyAcc-max()-Z"
```

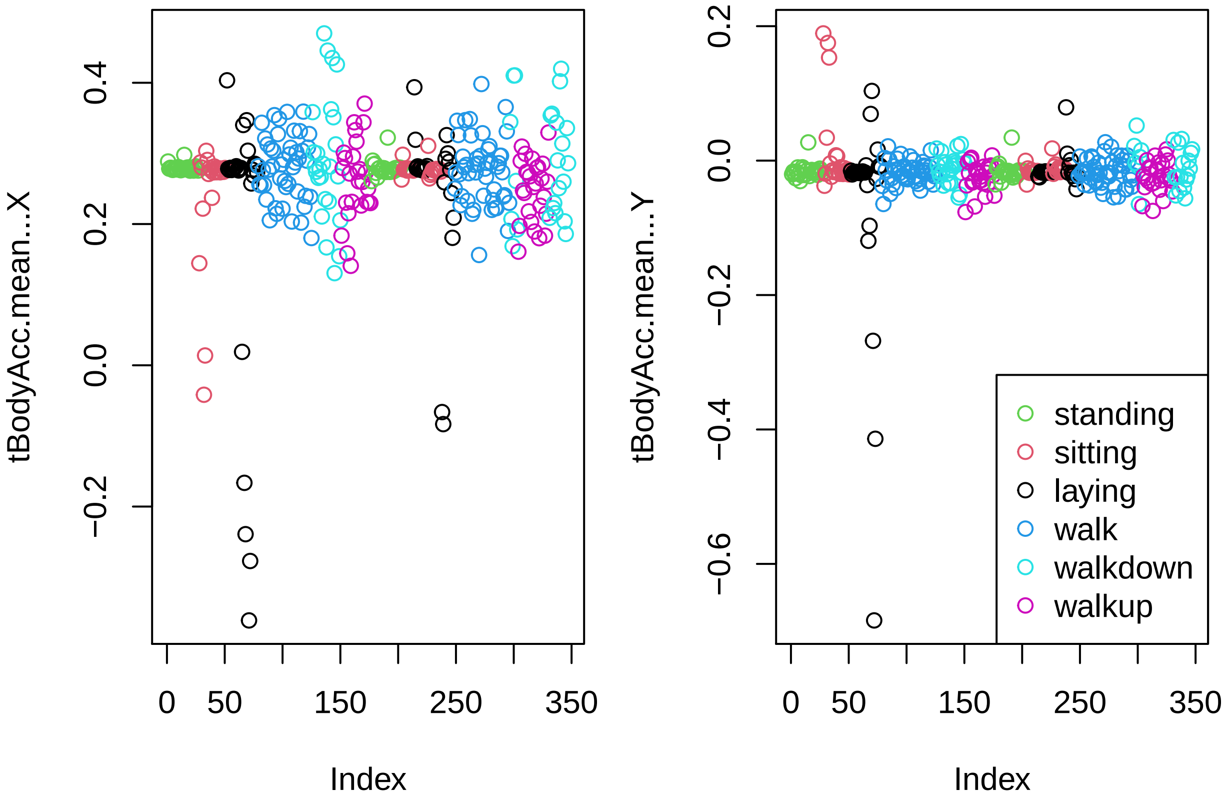
```
table(samsungData$activity)
```

```
## 
##   laying  sitting standing     walk walkdown   walkup 
##     1407     1286     1374     1226      986     1073
```

View the behavior of the average acceleration

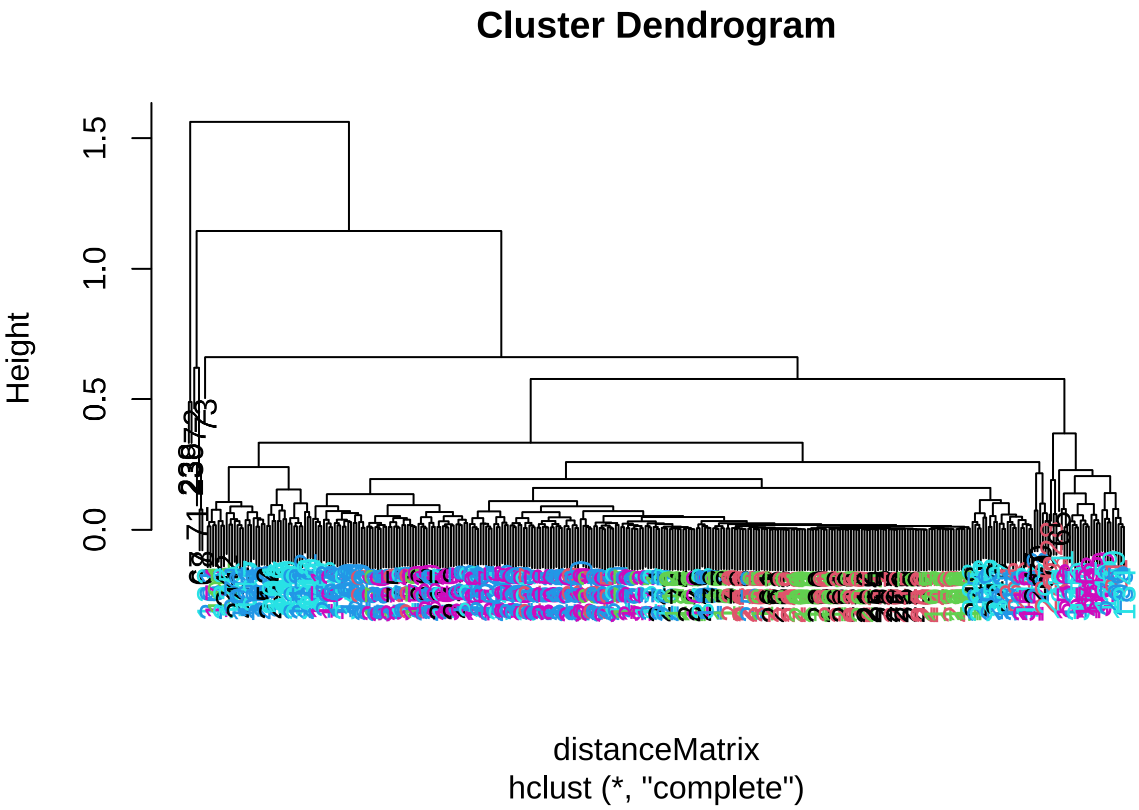
Plotting average acceleration for frist subject

```
par(mfrow = c(1, 2), mar = c(5, 4, 1, 1))
samsungData <- transform(samsungData, activity = factor(activity))
sub1 <- subset(samsungData, subject == 1)
plot(sub1[, 1], col = sub1$activity, ylab = names(sub1)[1])
plot(sub1[, 2], col = sub1$activity, ylab = names(sub1)[2])
legend("bottomrigh", legend = unique(sub1$activity), col = unique(sub1$activity),
       pch = 1)
```



Clustering based just on average acceleration

```
source("myplclust.R")
distanceMatrix <- dist(sub1[, 1:3])
hclustering <- hclust(distanceMatrix)
myplclust(hclustering, lab.col = unclass(sub1$activity))
```

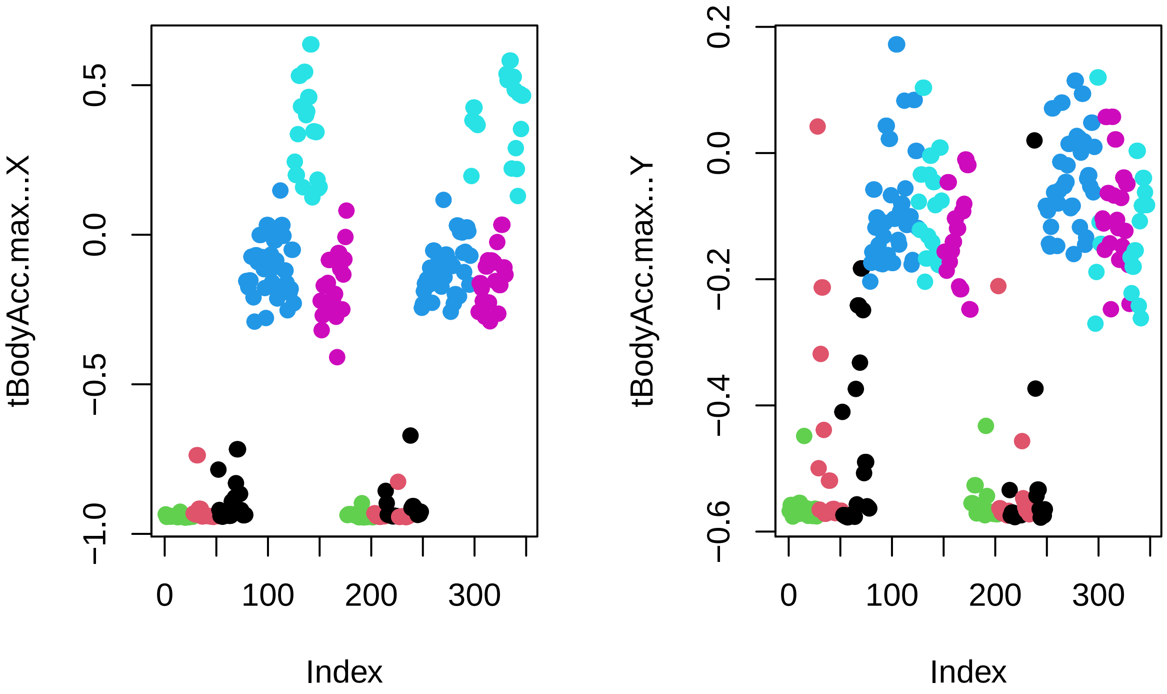


The first three variables seem not be informative, so we'll take the columns 11 and 10 that content the maximum acceleration for the first subject

View the behavior of the max acceleration

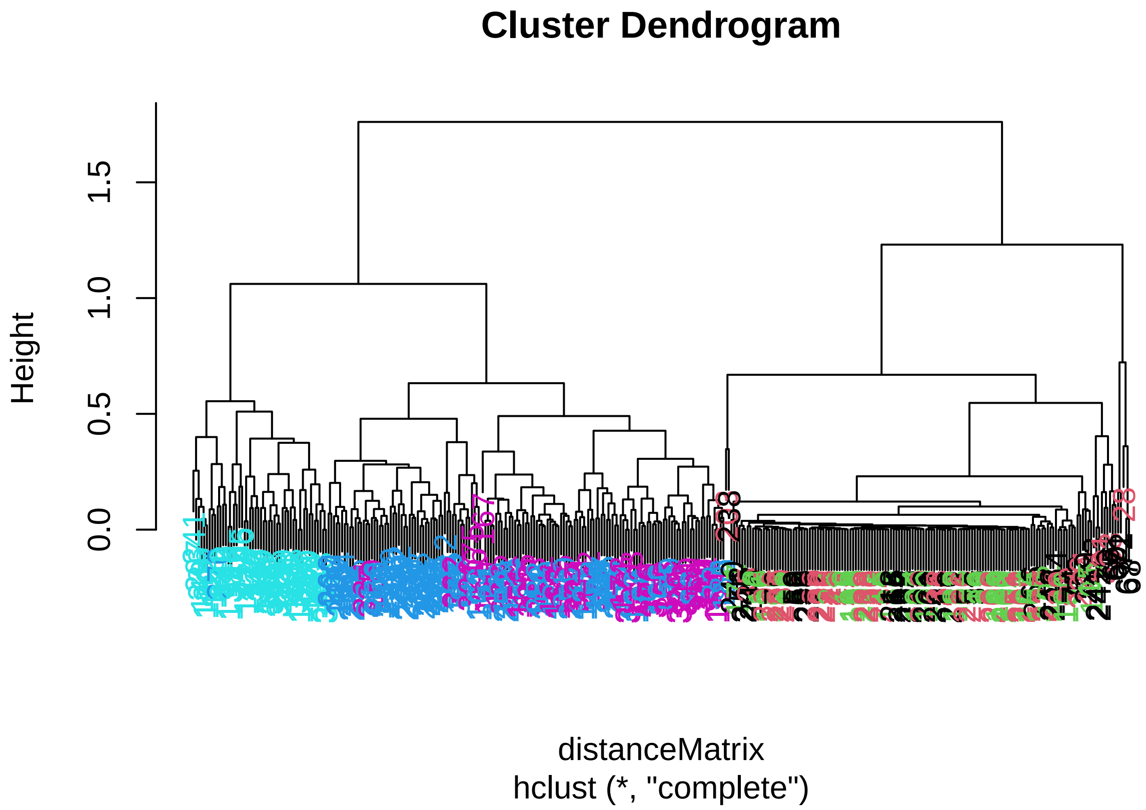
Plotting max acceleration for the first subject

```
par(mfrow = c(1, 2))
plot(sub1[, 10], pch = 19, col = sub1$activity, ylab = names(sub1)[10])
plot(sub1[, 11], pch = 19, col = sub1$activity, ylab = names(sub1)[11])
```



Clustering based on maximum acceleration

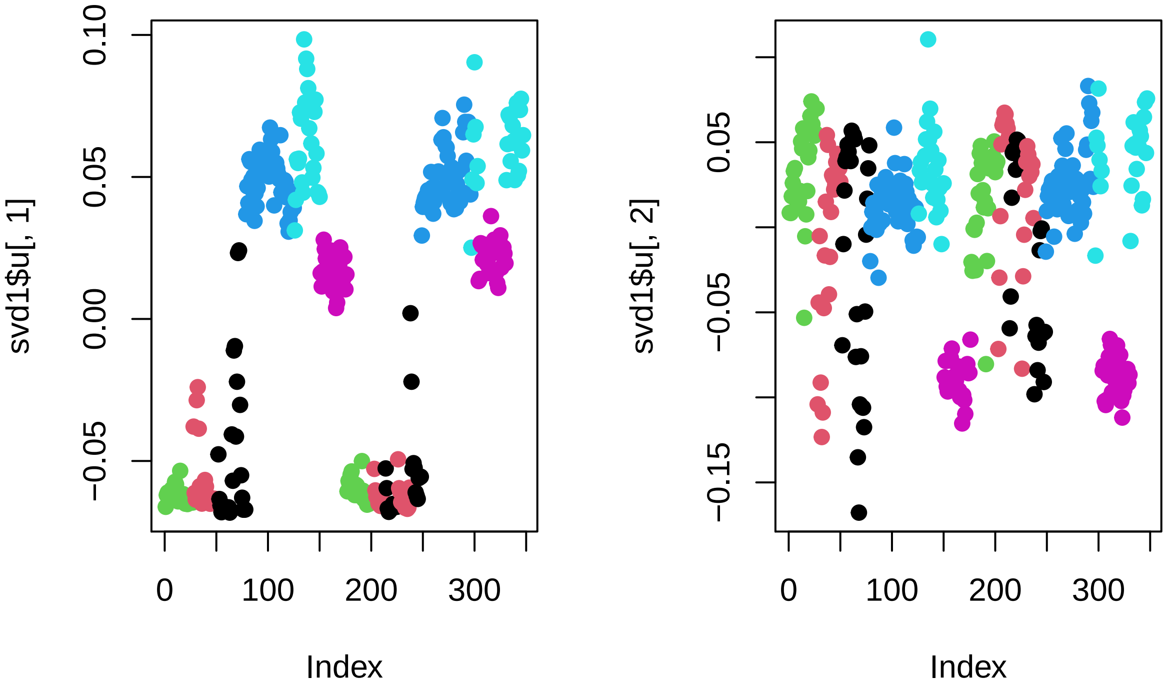
```
distanceMatrix <- dist(sub1[, 10:12])
hclustering <- hclust(distanceMatrix)
myplclust(hclustering, lab.col = unclass(sub1$activity))
```



This clustering is useful, but is hard identify each internal cluster from the external clusters of moving and not moving

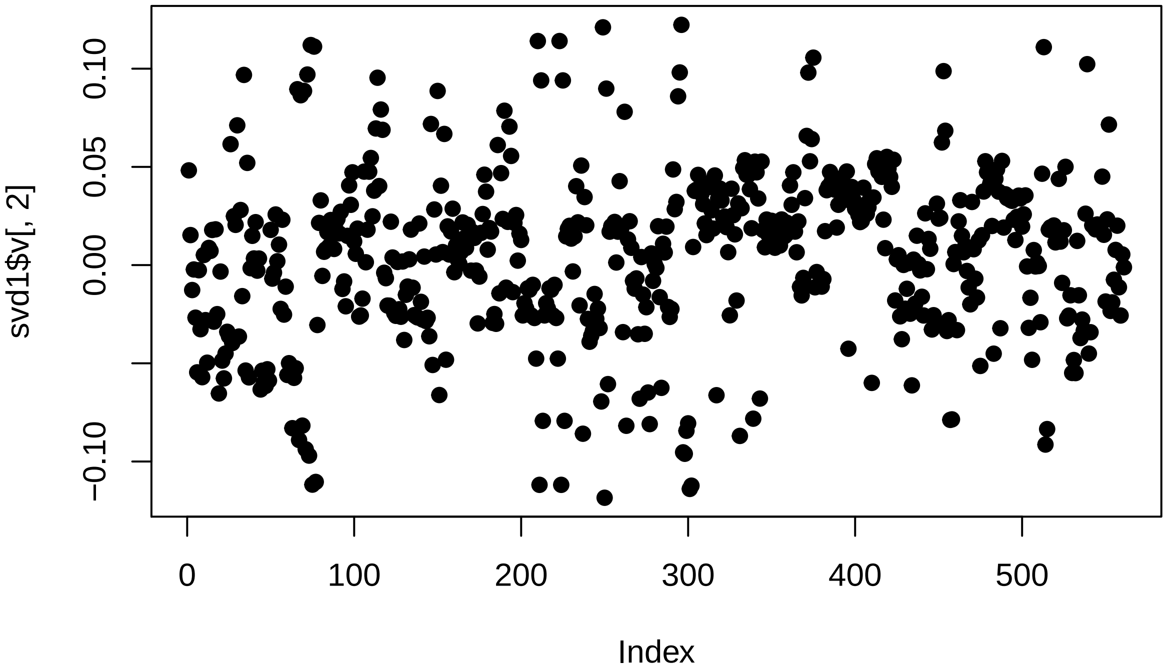
Singular Value Decomposition

```
svd1 = svd(scale(sub1[, -c(562, 563)]))
par(mfrow = c(1, 2))
plot(svd1$u[, 1], col = sub1$activity, pch = 19)
plot(svd1$u[, 2], col = sub1$activity, pch = 19)
```



In the previous plot seems to be separating out the magenta color from all the other clusters so we can try to find the maximum contributor

```
plot(svd1$v[, 2], pch = 19)
```

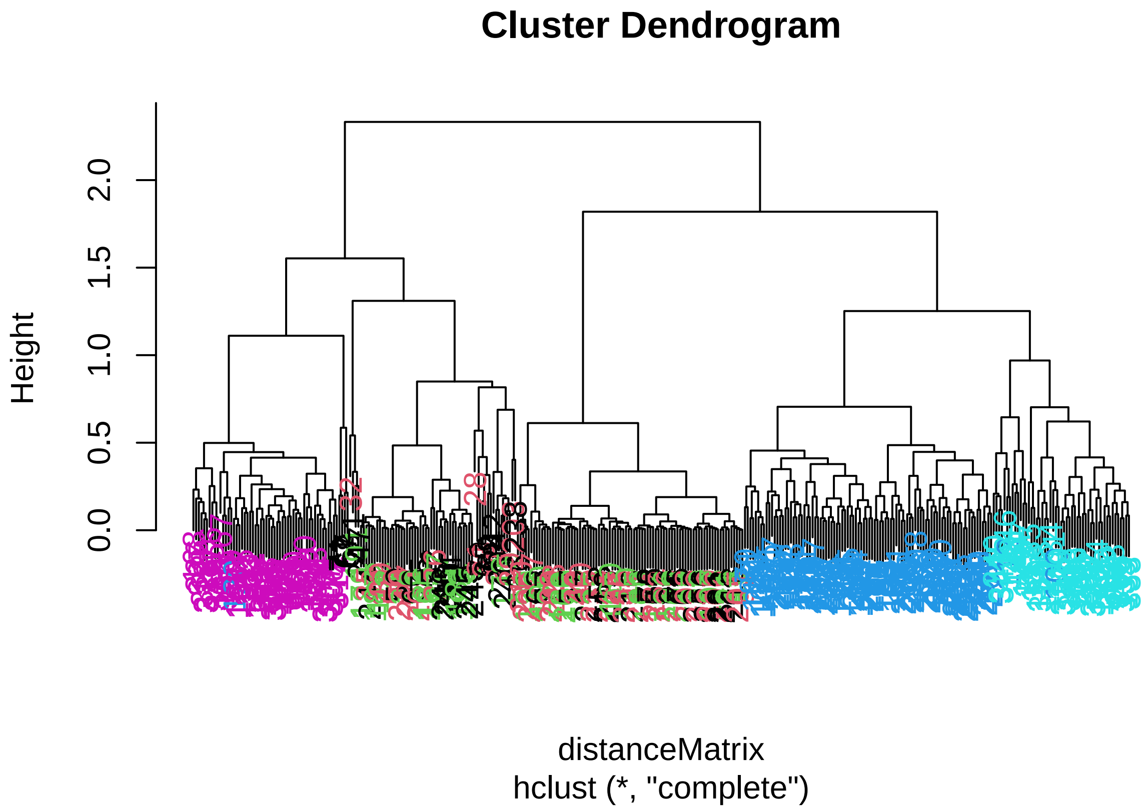


To do this, we can use the `which.max` function to figure out which of the 500 feature contributes most of the variations across observations

```
maxContrib <- which.max(svd1$v[, 2])
names(samsungData)[maxContrib]
```

```
## [1] "fBodyAcc.meanFreq...Z"
```

```
distanceMatrix <- dist(sub1[, c(10:12, maxContrib)])
hclustering <- hclust(distanceMatrix)
myplclust(hclustering, lab.col = unclass(sub1$activity))
```



K means clustering

Is useful set `nstart` with some specify value to find the optimal solution in place from select those point aleatory

```
kClust <- kmeans(sub1[, -c(562, 563)], centers = 6)
table(kClust$cluster, sub1$activity)
```

```
##
##     laying sitting standing walk walkdown walkup
##   1     24      33       46    0        0      0
##   2      0       0        0    0        0     53
##   3     16      12        7    0        0      0
##   4     10       2        0    0        0      0
##   5      0       0        0   95        0      0
##   6      0       0        0    0       49      0
```

If you see the result, you can see k-means here had a little bit trouble separating out also the laying, sitting and standing

If we try with another value of `nstart` we get a better result:

```
kClust <- kmeans(sub1[, -c(562, 563)], center = 6, nstart = 100)
table(kClust$cluster, sub1$activity)
```

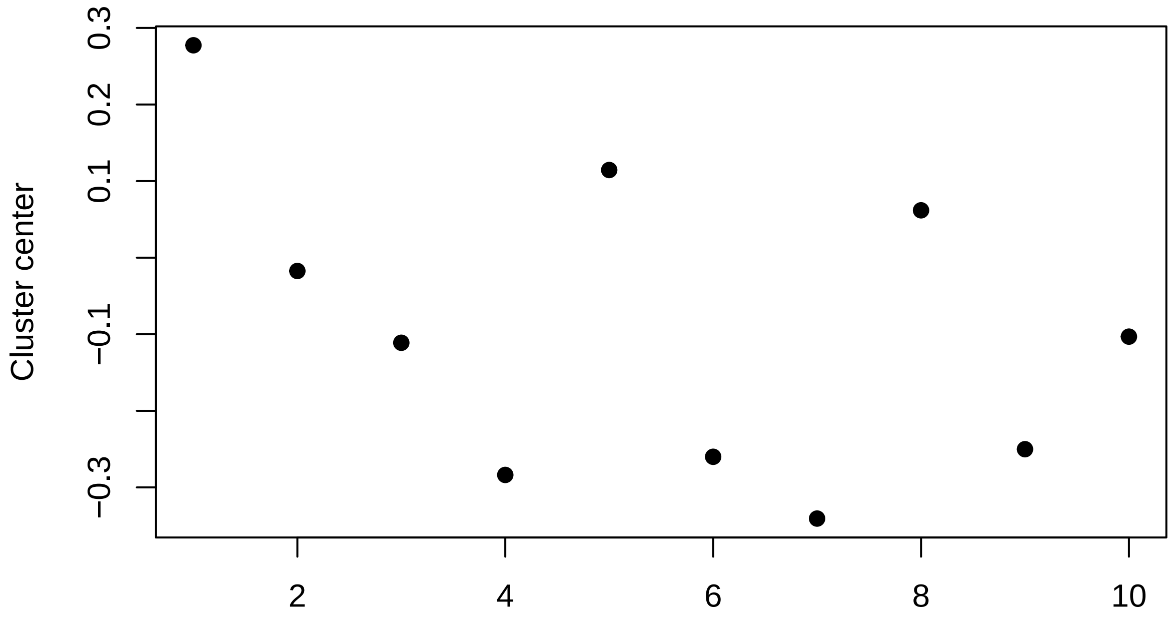
```
##
##     laying sitting standing walk walkdown walkup
##   1      0       0        0   95        0      0
##   2     29       0          0    0         0      0
##   3      3       0          0    0         0     53
##   4      0       0          0    0        49      0
##   5      0      37         51    0         0      0
##   6     18      10          2    0         0      0
```

```
par(nfrow = c(1, 1))
```

Cluster 1 variable centers (laying)

```
## Warning in par(nfrow = c(1, 1)): "nfrow" is not a graphical parameter
```

```
plot(kClust$center[1, 1:10], pch = 19, ylab = "Cluster center", xlab = "")
```



So one of the things that you can do by looking at the cluster centers is to see well what features seem to have interesting values that kind of drive the location to that center And, which could give you a hint, in terms of what features will be most useful for predicting that activity.