Post 02

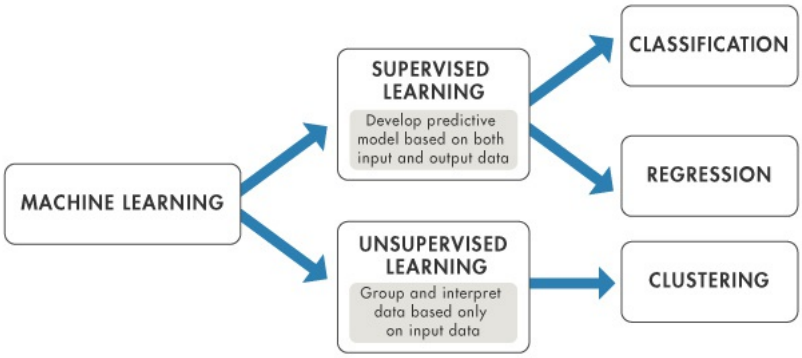
Brian Wang

November 27, 2017

Topic: Unsupervised Learning - K-Means Clustering Algorithm

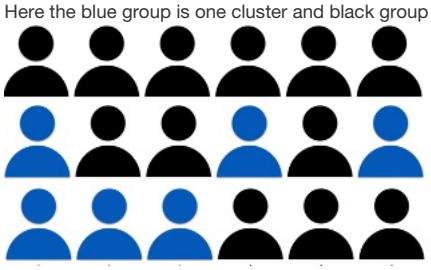
1. First at All, What the Hell is Unsupervised Learning?

Many people confused about the concept of unsupervised learning and supervised learning; unsupervised learning is primary used in finding structure in unlabeled data, supervised learning is primary used in making predictions based on labeled data[1] (more info about supervised learning, specifically on k-nearest neighbor algorithm, on my previous post).



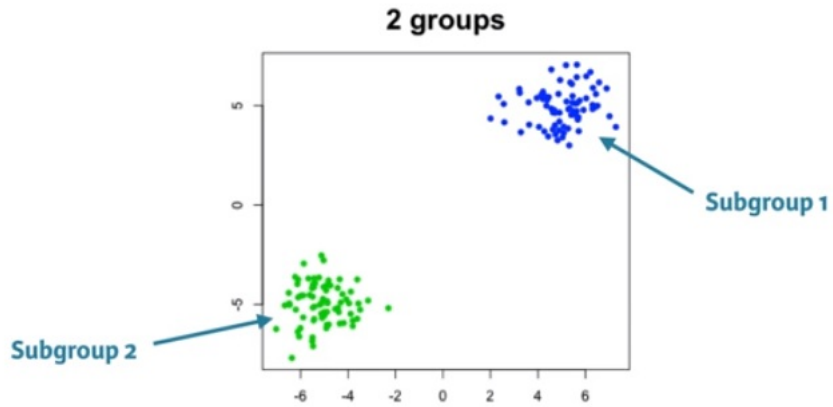
There are two type of unsupervised learning:
1. Finding homogenous subgroups within larger group-clustering
2. Finding patterns in features of the data-dimensionality reduction

In this post, we are going to talk about cluster only. An example of cluster from Cal students: A group of Cal student can be divided into different clusters by major, gender and age.
Here the blue group is one cluster and black group is another cluster, they were clustered by some feature.



2. What is K-means clustering algorithm?

K-means clustering algorithm is a method of vector quantization which is popular for cluster analysis in data mining.[2] The K-means clustering algorithm works by first assume a number of subgroups of cluster in the data, and then assign each oberservation to each subgroups.



Advantage of K-means clustering algorithm:
1. Fast, robust and easier to understand.
2. Relatively efficient.
3. Give the best result when data set is distinct or well-seperated from each other.

Disadvantage of K-means clustering algorithm:
1. The algorithm requires an intial assumption of the number of clusters. If the data is huge and complex, it is difficlt to estimate the number of clusters.
2. If the dataset has a lot overlapping, k-means clustering algorithm is usually not accurate.
3. Different representation of the data will yeild different results.
4. Selection of the initial centroids is random.

3. Function kmeans() in R

Kmeans() is the function that runs k-means clustering algorithm in R studio.

Inputs of Kmeans(X, Centers, iter.max = 10, nstart = 1, algorithm = c("Hartigan-Wong", "Lloyd", "Forgy", "MacQueen"), trace=FALSE)

x is a numeric matrix of data, or an object that can be coerced to such a matrix.
centers is the number of clusters/subgroups. (You have to make the decision, must be a numeric input)
iter.max is the maximum number of iteration allowed.
nstart is the numner of the intial random centroids(mean). If nstart = 25, it will generate 25 intial random centroids and choose the best one for the alogrithm. It helps to generate better accuracy.
algorithm is just different k means clustering alogrithms. By defacult, the system will choose the better one for the given data set. More details

Outputs of kmeans(), it usually return a list of nine elements

cluster is the cluster assign to each variable. It's integer from 1 to center(the input you have selected).
center is a matrix of cluster centers.
totss is the total sum of squares.
withinss is the vector of within-cluster sum of squares, one component per cluster.
tot.withinss is the total within-cluster sum of squares, i.e. sum(withinss).
betweenss is the between-cluster sum of squares, i.e. $totss - tot.withinss$.
size is the number of points in each cluster.

4. Example of implementing kmeans()

Here we are going to use the dataset,"USAreest" in r studio.
Step 1. prepare your dataset, removing any missing and unusual data. Scale your data if necessary.

```
datasets = USArrests
### removing all missing
datasetss = na.omit(datasets)
### inspecting the data
str(datasets)
```

```
## 'data.frame':    50 obs. of  4 variables:
##  $ Murder  : num  13.2 10 8.1 8.8 9 7.9 3.3 5.9 15.4 17.4 ...
##  $ Assault : int  236 263 294 190 276 204 110 238 335 211 ...
##  $ UrbanPop: int  58 48 80 50 91 78 77 72 80 60 ...
##  $ Rape    : num  21.2 44.5 31 19.5 40.6 38.7 11.1 15.8 31.9 25.8 ...
```

```
### first five row
head(datasets,5)
```

```
##            Murder Assault UrbanPop Rape
## Alabama      13.2     236       58 21.2
## Alaska       10.0     263       48 44.5
## Arizona       8.1     294       80 31.0
## Arkansas      8.8     190       50 19.5
## California    9.0     276       91 40.6
```

```
summary(datasets)
```

```
##      Murder          Assault         UrbanPop          Rape
##  Min.   : 0.800   Min.   : 45.0   Min.   :32.00   Min.   : 7.30
##  1st Qu.: 4.075   1st Qu.:109.0   1st Qu.:54.50   1st Qu.:15.07
##  Median : 7.250   Median :159.0   Median :66.00   Median :20.10
##  Mean   : 7.788   Mean   :170.8   Mean   :65.54   Mean   :21.23
##  3rd Qu.:11.250   3rd Qu.:249.0   3rd Qu.:77.75   3rd Qu.:26.18
##  Max.   :17.400   Max.   :337.0   Max.   :91.00   Max.   :46.00
```

As you can see, the data has a wide range from 0.8 to 337. We need to rescale the data to make it more usable.

```
### scale() is a useful function that scale your data accordingly
modified_datasets = scale(datasets)
summary(modified_datasets)
```

```
##      Murder           Assault           UrbanPop            Rape
##  Min.   :-1.6044   Min.   :-1.5090   Min.   :-2.31714   Min.   :-1.4874
##  1st Qu.:-0.8525   1st Qu.:-0.7411   1st Qu.:-0.76271   1st Qu.:-0.6574
##  Median :-0.1235   Median :-0.1411   Median : 0.03178   Median :-0.1209
##  Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.00000   Mean   : 0.0000
##  3rd Qu.: 0.7949   3rd Qu.: 0.9388   3rd Qu.: 0.84354   3rd Qu.: 0.5277
##  Max.   : 2.2069   Max.   : 1.9948   Max.   : 1.75892   Max.   : 2.6444
```

```
### now our data has much narrow range from [-2,3]
```

Step 2 make your assumption on number of centers(subgroup)
We can just assume there are three centers(regions, west, east and mid).
You can always change the number of centers and make any nunbers you want.

```
center3 = kmeans(modified_datasets,center =3, nstart = 25)
### inspecting our outputs
str(center3)
```

```
## List of 9
##  $ cluster     : Named int [1:50] 2 2 2 1 2 2 1 1 2 2 ...
##   ..- attr(*, "names")= chr [1:50] "Alabama" "Alaska" "Arizona" "Arkansas" ...
##  $ centers     : num [1:3, 1:4] -0.447 1.005 -0.962 -0.347 1.014 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:3] "1" "2" "3"
##   .. ..$ : chr [1:4] "Murder" "Assault" "UrbanPop" "Rape"
##  $ totss       : num 196
##  $ withinss    : num [1:3] 19.6 46.7 12
##  $ tot.withinss: num 78.3
##  $ betweenss   : num 118
##  $ size        : int [1:3] 17 20 13
##  $ iter        : int 2
##  $ ifault      : int 0
##  - attr(*, "class")= chr "kmeans"
```

```
### just to give you better understanding of the output
### each state is labeled with a specfic cluster number
head(center3$cluster,10)
```

```
##     Alabama      Alaska     Arizona    Arkansas  California    Colorado
##           2           2           2           1           2           2
## Connecticut    Delaware     Florida     Georgia
##           1           1           2           2
```

```
### cluster matrix
center3$centers
```

```
##       Murder    Assault   UrbanPop       Rape
## 1 -0.4469795 -0.3465138  0.4788049 -0.2571398
## 2  1.0049340  1.0138274  0.1975853  0.8469650
## 3 -0.9615407 -1.1066010 -0.9301069 -0.9667633
```

As you can see, cluster1 has the lowest number of murder, assault, urbanpop and rape. Follow by cluster2 and last is cluster3. Cluster1 can be thought as the safest area, cluster2 is relatively safer than cluster3.

Step 3 visualize your cluster
You need to install "factoextra" package if you have not installed.

```
library(factoextra)
```

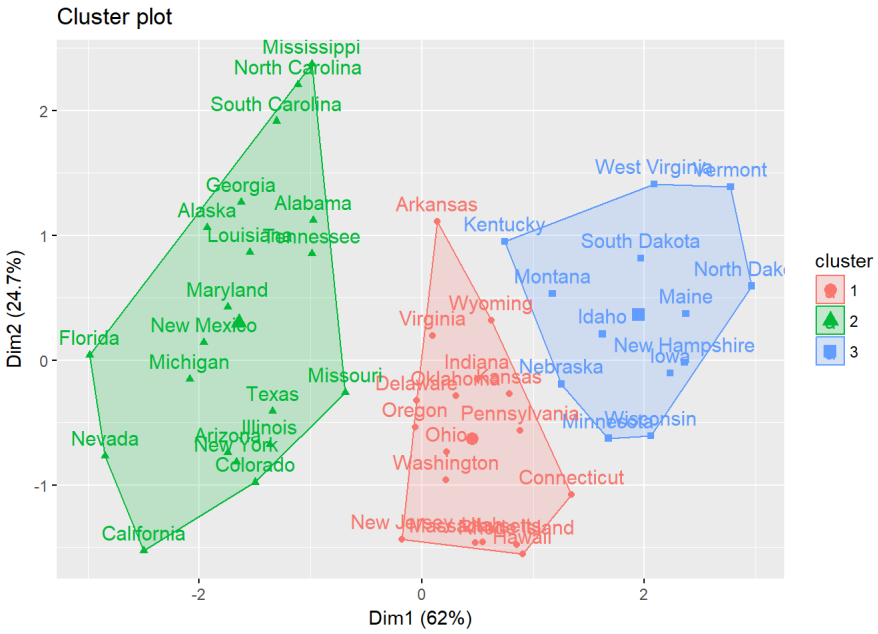
```
## Warning: package 'factoextra' was built under R version 3.4.3
```

```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 3.4.2
```

```
## Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13EFCZ
```

```
###fviz_cluster(object, data), object is like the method of partition, in another words, how do you to seperate yo
u cluster.
fviz_cluster(center3, data = modified_datasets)
```

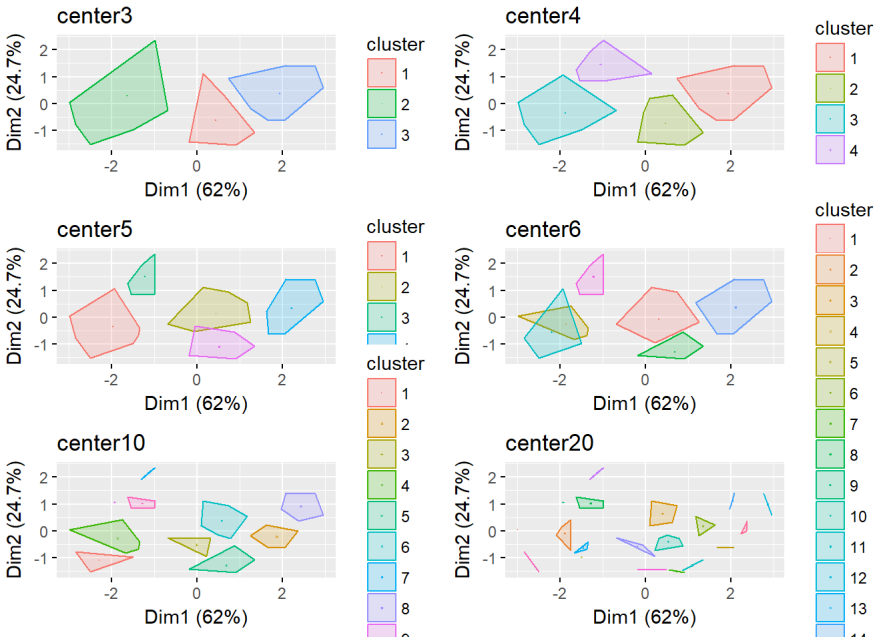


From the plot, we can clearly see there are three non-overlapped cluster.
Let do some experitment, we increase the number of center.

```
center4 = kmeans(modified_datasets, centers = 4, nstart = 25)
center5 = kmeans(modified_datasets, centers = 5, nstart = 25)
center6 = kmeans(modified_datasets,center = 6, nstart = 25)
center10 = kmeans(modified_datasets,center = 10, nstart = 25)
center15 = kmeans(modified_datasets, center =20, nstart = 25)

### plots to compare

plotcenter3 = fviz_cluster(center3,  data = datasets, geom ="points") + ggtitle("center3")
plotcenter4 = fviz_cluster(center4,  data =datasets, geom ="points") + ggtitle("center4")
plotcenter5 = fviz_cluster(center5,  data =datasets, geom ="points") + ggtitle("center5")
plotcenter6 = fviz_cluster(center6,  data =datasets, geom ="points") + ggtitle("center6")
plotcenter10 = fviz_cluster(center10,  data =datasets, geom ="points") + ggtitle("center10")
plotcenter15 = fviz_cluster(center15,  data =datasets, geom ="points") + ggtitle("center20")

### in order to display multiple grpah as the same, you need this package "gridExtra" and function grid.arrange()

library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 3.4.3
```

```
grid.arrange(plotcenter3, plotcenter4, plotcenter5,plotcenter6,plotcenter10, plotcenter15, nrow = 3)
```



As we increase the number of the center, the cluster area gets smaller and smaller, which makes sense because there are less points in the cluster.From all 6 different center, which one do you think is the most suitable for 50 states criminal region-grouping in USA?

Take Home Message/Conclusion

Whenever you are dealing with unsupervised machine learning and you need to identify clusters within a dataset. If the variables in the given dataset are somehow distinct, k-means clustering algorithm will be a powerful and efficient tool to identify clusters.Indeed, k means clustering algorithm is one of the most popular methods for clustering data.
I hope this post can give you some insight of unsupervised machine learning and k-means cluster algorithm.

Thank you for reading

Reference

http://blog.galvanize.com/introduction-k-means-cluster-analysis/\

https://www.slideshare.net/DarshakMehta6/k-means-clustering-algorithm-63472248\

https://www.datascience.com/blog/k-means-clustering\

https://www.r-bloggers.com/k-means-clustering-in-r/\

http://stanford.edu/~cpiech/cs221/handouts/kmeans.html\

http://www.sthda.com/english/rpkgs/factoextra/reference/fviz_cluster.html\

https://stats.stackexchange.com/questions/263374/clusters-and-data-visualisation-in-r\

https://www.rdocumentation.org/packages/factoextra/versions/1.0.5/topics/fviz_cluster\

at.ethz.ch/R-manual/R-devel/library/stats/html/kmeans.html\