K-Means Clustering of Forest Types Using R

Eugenie Shin

Computing and Graphics in Applied Statistics

17 April 2020

Abstract

For this project, we demonstrate the k-means clustering method on the Forest Type Mapping data set. The data set contains integers representing spectral information in the green, red, and near infrared wavelengths, obtained from various forested areas using ASTER satellite imagery. We attempt to classify the data into four possible types of forest: ‘Sugi’, ‘Hinoki’, ‘Mixed Deciduous’, and ‘Other’.

Methods and Materials

The first column of the Forest Types data set indicates the type of forest (‘s’ for Sugi, ‘h’ for Hinoki, ‘d’ for Mixed Deciduous, ‘o’ for Other). Since we already know that the data has these four types of forest, we know in advance to expect k = 4 clusters. We will look at attributes b1 to b9 (columns 2 to 10), indicating ASTER image bands containing spectral information for each forest, to see whether the data can be classified.

The k-means clustering method works by first randomly assigning each data point to a cluster, finding the centroid of each cluster, then reassigning data points so that every data point is closer to the centroid of its new cluster than that of any other cluster. Then, it recomputes the centroid of the resulting cluster and repeats the process.

We load the Forest Type data set into R:

```
> forest <- read.csv("/Users/Eugenie/Documents/class
material/2019-2020/stat 486 computing applied
stat/project/ForestTypes/training.csv")
 > head(forest)
  class b1 b2 b3  b4 b5  b6  b7 b8 b9 pred_minus_obs_H_b1
pred_minus_obs_H_b2 pred_minus_obs_H_b3
1    d  39 36 57  91 59 101  93 27 60               75.70
14.86               40.35
2    h  84 30 57 112 51  98  92 26 62               30.58
20.42               39.83
3    s  53 25 49  99 51  93  84 26 58               63.20
26.70               49.28
```

```
4    s  59 26 49 103 47  92  82 25 56              55.54
24.50               47.90
5    d  57 49 66 103 64 106 114 28 59              59.44
2.62                32.02
6    h  85 28 56 120 52  98 101 27 65              35.14
23.43               42.29
  pred_minus_obs_H_b4 pred_minus_obs_H_b5 pred_minus_obs_H_b6
pred_minus_obs_H_b7 pred_minus_obs_H_b8
1                7.97              -32.92              -38.92
-14.94                4.47
2              -16.74              -24.92              -36.33
-15.67                8.16
3                3.25              -24.89              -30.38
-3.60                4.15
4               -6.20              -20.98              -30.28
-5.03                7.77
5               -1.33              -37.99              -43.57
-34.25                1.83
6              -16.58              -25.43              -34.14
-17.45                1.58
  pred_minus_obs_H_b9 pred_minus_obs_S_b1 pred_minus_obs_S_b2
pred_minus_obs_S_b3 pred_minus_obs_S_b4
1               -2.36              -18.41               -1.88
-6.43              -21.03
2               -2.26              -16.27               -1.95
-6.25              -18.79
3               -1.46              -15.92               -1.79
-4.64              -17.73
4                2.68              -13.77               -2.53
-6.34              -22.03
5               -2.94              -21.74               -1.64
-4.62              -23.74
```

```
6                -10.28               -26.18                -1.89
-5.89               -34.92
  pred_minus_obs_S_b5 pred_minus_obs_S_b6 pred_minus_obs_S_b7
pred_minus_obs_S_b8 pred_minus_obs_S_b9
1                 -1.60                -6.18               -22.50
-5.20                -7.86
2                 -1.99                -6.18               -23.41
-8.87               -10.83
3                 -0.48                -4.69               -19.97
-4.10                -7.07
4                 -2.34                -6.60               -27.10
-7.99               -10.81
5                 -0.85                -5.50               -22.83
-2.74                -5.84
6                 -1.89                -8.05               -29.72
-1.94                -4.94
```

Use the kmeans() function (we are only interested in columns 2 to 10 for the purposes of this project).

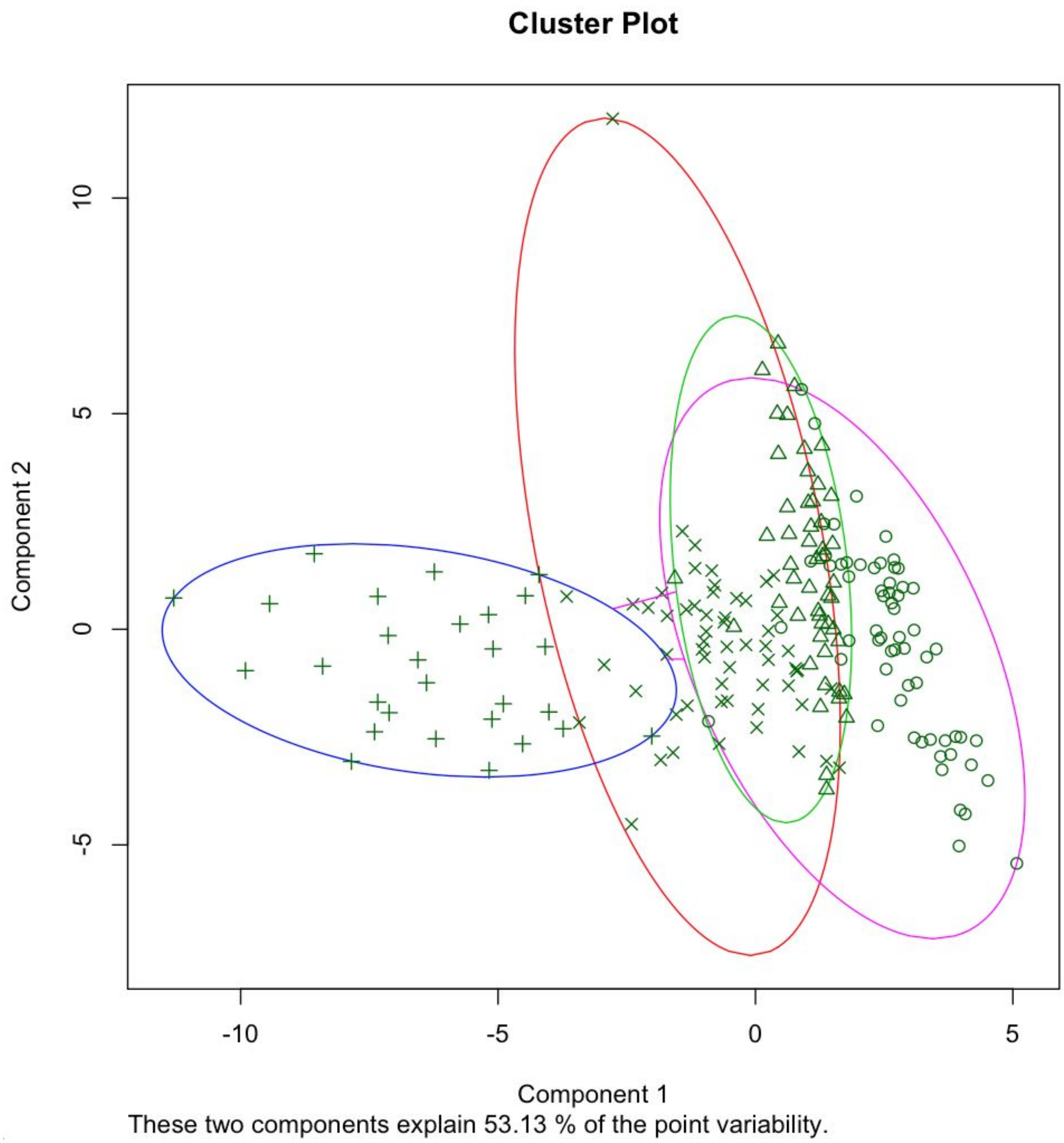
```
> forestCluster <- kmeans(forest[,2:10], 4, nstart=20)
> forestCluster
K-means clustering with 4 clusters of sizes 62, 50, 28, 58

Cluster means:
        b1       b2       b3        b4       b5        b6       b7
b8       b9
1 56.54839 28.46774 51.54839  92.61290 49.62903  91.58065 75.16129
24.50000 55.50000
2 78.34000 29.90000 55.54000 114.48000 50.86000  96.18000 99.26000
25.30000 60.40000
3 67.39286 73.10714 95.25000 106.60714 80.14286 121.03571 93.17857
45.85714 79.75000
```

```
4 54.37931 48.53448 68.41379  97.03448 64.91379 104.36207 98.39655
27.81034 58.74138

Clustering vector:
  [1] 4 2 1 1 4 2 1 4 1 3 1 3 4 4 1 4 3 4 1 2 3 4 1 2 4 2 3 4 4 4 1 1
3 2 4 4 3 2 3 4 4 4 1 1 2 4 4 3 4 2 4 1 1 2 2 1 1 4 1 2 4 4 4 3 1
 [66] 1 2 4 4 2 3 3 3 1 2 1 4 4 2 1 4 4 3 2 3 2 4 4 1 3 3 1 3 3 4 1 3
2 3 1 4 3 2 2 2 2 2 1 1 2 4 4 4 4 2 3 1 2 1 1 1 1 4 4 2 1 2 1 2 2
[131] 1 1 3 1 2 1 3 1 4 1 1 2 1 2 1 4 4 1 2 4 1 2 1 4 4 4 1 2 4 1 3 4
3 1 4 2 1 1 3 4 1 2 2 2 1 1 2 4 2 4 1 1 2 1 2 1 4 4 2 1 1 1 2 4 4
[196] 2 2 2

Within cluster sum of squares by cluster:
[1] 17825.81 10624.14 62926.46 33071.97
 (between_SS / total_SS =  62.8 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"
"tot.withinss" "betweenss"    "size"         "iter"
[9] "ifault"
```

To visualize the data, use clusplot() in the “cluster” package. The function uses PCA and the first two principal components that explain the data.

```
> forestCluster$cluster <- as.factor(forestCluster$cluster)
> clusplot(forest, forestCluster$cluster, main = 'Cluster Plot',
color=TRUE)
```

Results



Cluster Plot
10
5
0
-5
-10
-5
0
5
Component 2
Component 1
These two components explain 53.13 % of the point variability.

To see how the clustering algorithm grouped the data points:

```
> table(forestCluster$cluster, forest$class)

     d  h  o  s
  1  3  3  1 55
  2  1 45  1  3
  3  1  0 27  0
  4 49  0  8  1
```

From this output, we can see that most of the data points in the ‘Sugi’ (s) class got grouped into cluster 1, most of the data points in ‘Hinoki’ (h) into cluster 2, most of the data points in ‘Other’ (o) in cluster 3, and most of the data points in ‘Deciduous’ (d) into cluster 4.

Discussion

The clustering found by our analysis seems to correspond to the data points’ actual classification. Each cluster contains a majority of the points of one forest type. Points in the ‘Hinoki’ class, for example, were almost all grouped into cluster 2; only three out of 48 points were wrongly grouped into cluster 1. However, the algorithm failed to correctly classify every single one of the data points. This suggests that, while values of b1 to b9 are correlated with certain types of forests, the classes of the data may not be completely distinct. In the visual representation of the clusters using clusplot(), we see that the clusters (the areas inside each colored oval) greatly overlap with one another (though the plot is limited to two components, whereas our data has more than two, which might explain more of the groupings). Overall, the outcome is enough to show that a clear pattern or grouping exists within the data.

Literature Cited

Johnson, B., Tateishi, R., Xie, Z., 2012. Using geographically-weighted variables for image classification. Remote Sensing Letters, 3 (6), 491-499. (http://archive.ics.uci.edu/ml/datasets/Forest+type+mapping)

Kodali, Teja. "K Means Clustering in R." *R-bloggers,* 28 Dec. 2015, https://www.r-bloggers.com/k-means-clustering-in-r/.

Jaiswal, Sejal. "K Means Clustering in R Tutorial." *Datacamp,* 14 March 2018, https://www.datacamp.com/community/tutorials/k-means-clustering-r.