Data 630: Machine Learning Summer 2023

Assignment # 5– K-means Clustering

Simon A Hochmuth

Email: shochmuth3@student.umgc.edu

Professor: Ami Gates

Introduction

Objective:

The analysis conducted on the satellite dataset had the goal of using K-means clustering to understand how the features group together. Using these groups, the goal is to figure out if there is a relationship between each of the features, and how these relationships can help predict and understand soil type. Once the model is made the overall objective is to analyze each cluster to see how the data is segmented to understand soil attributes, where a strategy can be made to identify soil types better. The goal of this analysis is to provide a way to categorize and understand soil attributes in each 3x3 neighborhood provided in the dataset.

Problem Domain:

The National Aeronautics and Space Administration (NASA) has created the Landsat program to play a critical role in understanding and managing Earths resources critical for sustaining human life. Figure 1 shows the various Landsat missions that NASA has had since 1972. The goal of the images taken by the various satellites over the years is to continuously observe and show natural or human induced change on the global land surface. The increase in global land use has consequences in areas like climate change, ecosystem destruction, carbon cycling, resource management, human health, or the global economy, which is why NASA has shown interest in documenting and recording data in the Landsat program. (NASA 2022).

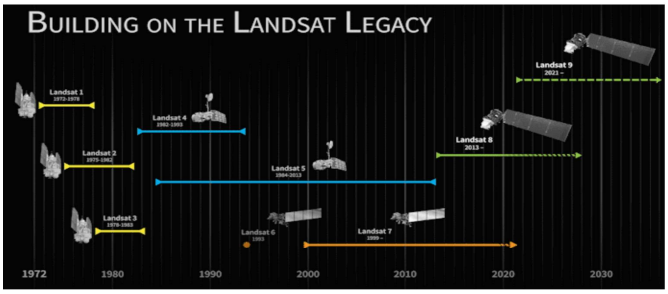

Figure 1: Image showing all landsat projects (NASA 2022).

Method Rationale:

As discussed, the goal of the Landsat program is to continuously observe and document the global land surface to understand change that has occurred. To do this, a K-means clustering model will be used to group and segment the data using unsupervised machine learning. After the data is preprocessed, the K-means model will output clusters with the data segmented into groups. The next step will be to analyze each cluster to understand if these groupings sufficiently segment each of the 6 soil types in a way that can be used to predict future images.

Analysis:

Data:

The dataset was pulled from the UCI machine learning repository, and the data was gathered from NASA's Landsat program, which has been around since 1972. Each observation collects data that corresponds to a 3x3 subsection within an 82x100 area. Each observation consists of images collected in red, green, and infrared spectral bands. The data contains a target variable called class, which is a number code that describes the central pixel in each 3x3 neighborhood image. The target variable code relates to either red soil, cotton crop, grey soil, damp grey soil, soil with vegetation stubble, or very damp grey soil (Srinivasan. 1993).

Exploratory Analysis:

The data contains 37 features with 4435 observations gathered from pixel values in satellite images. Since the data was based on satellite images, there was little point in exploring the skew of each variable since pictures were not provided to compare against. However, the data seemed to follow a bell curve for most distributions, as shown in figure 2. This makes sense

since the Landsat program is focused on making the center point of these images one of the six soil types, which in turn will cause the image to have the largest density in the middle.

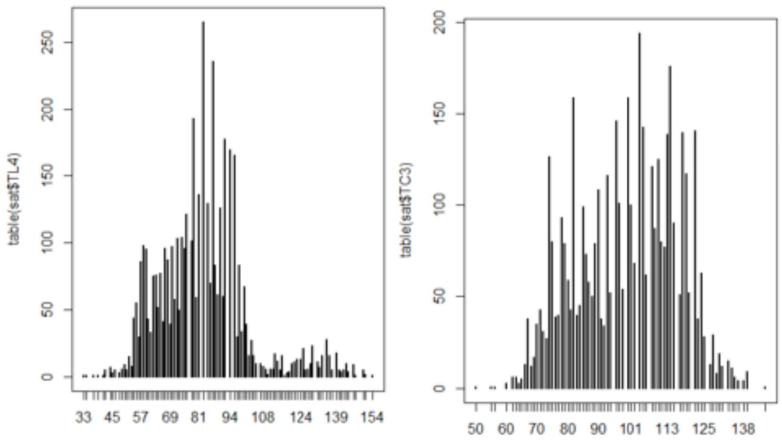

Figure 2: histogram of variables TC3 and TL4.

The target variable was also analyzed to understand which of the soil attributes were most prevalent in the data. For the most part the data was evently distributed, with a skew towards the 1st , 3rd, and 7th class as shown in figure 3 below.

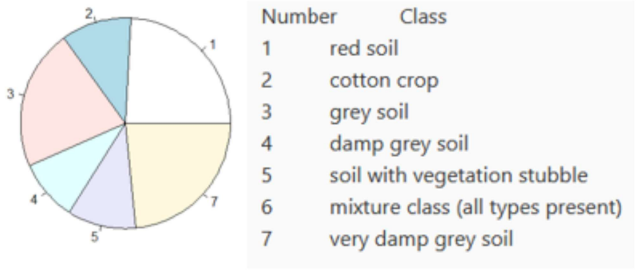

Figure 3: Pie chart of the target class with legend provided by Srinivasan.

Preprocessing:

Due to the lack of insight into the data, it was decided not to clean any of the variables. The variables had no missing values, as shown in figure 4, which meant nothing had to be dropped or filled. Lastly, a new dataset was made without the target variable because it was not

necessary for the K-means clustering model, however the model would be compared against the original target variable to understand how the model segmented each class.

TL1	TL2	TL3	TL4	TC1	TC2	TC3	TC4	TR1	TR2	TR3	TR4	CL1	CL2	CL3	CL4
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CC1	CC2	CC3	CC4	CR1	CR2	CR3	CR4	BL1	BL2	BL3	BL4	BC1	BC2	BC3	BC4
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BR1	BR2	BR3	BR4	class											
0	0	0	0	0											

Figure 4: count of missing values in the dataset.

Algorithm Intuition:

The K-means model is an unsupervised learning method that uses the process of clustering to find groups such that similar objects are in the same group. This isn't a prediction method but uses similarity and relationships to create these groups. It is then up to the Data Scientist to analyze and understand each grouping to find the underlying patterns in the data. The goal is to use this clustering technique to explore the data, not to create a predictive model. The K-means model creates these clusters by reading in the number of clusters an analyst wants equal to *k*. The model then picks *k* random points within the data as centroids and assigns records to the closest cluster, then it recalculates the centroid based on the mean value of all points. It repeats this process, until the centroid does not change (UMGC. 2023). The next step will be for the analyst to look at each cluster and try to understand the underlying patterns within each grouping.

Cluster Development & Anomaly Detection:

An elbow plot was created that plotted the sum of squares across various amounts of clusters. The plot shows a steep decline in the sum of squares which forms a sharp angle called

the elbow, which was used to determine the optimal number of clusters to use for this model. In this case the optimal amount is shown in figure 5 as three clusters, however there is a second point where the plot tapers out at six clusters. It was decided to use three clusters to start, but some further exploration in the outputs at six clusters could be done as well.

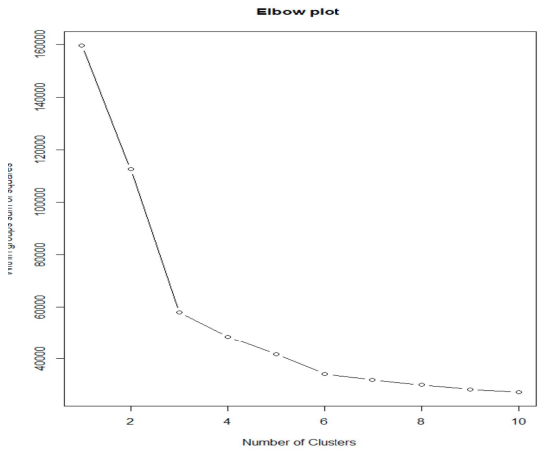

Figure 5: shows elbow plot for the Landsat data.

Anomalies can be detected by locating the instances with the largest distance from the cluster center. By running the formulas in figure 6, an array of distances is captured and is sorted to find the largest values in the variable outlier. The variable then outputs the top 10 anomalies with the largest distances from the center. Future work would include diving into each of these anomalies to understand how to better clean, understand, and capture data in the future.

```
#7. Anomaly detection
centers <- kc$centers[kc$cluster, ]
#head(centers, 15)
distances <- sqrt(rowSums((newsat - centers)^2))
distances
outliers <- order(distances, decreasing=T)[1:10]
outliers
sat[outliers,]
```

```
> outliers <- order(distances, decreasing=T)[1:10]
> outliers
 [1] 1181  761 1221 1117  995 1277  268  580 3573 200
>
```

Figure 6: Formulas to detect anomalies and the output of the top 10 outliers.

Result:

Output and Model Properties:

As discussed, the elbow method found that the best number of clusters for this model was *k* = 3 with a possibility of using *k=6*. After some analysis it was found that *k=6* provided less rows with similar objects in one group, as shown in figure 7 below. For that reason, *k=3* was kept as the number of clusters used by the model. However, both matrices in figure 7 show that within the clusters that some classes share similar relationships and attributes, which is why there is overlap across the clusters.

	1	2	3	4	5	6
1	635	20	16	0	383	18
2	0	4	4	386	85	0
3	9	72	1	0	2	877
4	0	313	21	0	13	68
5	21	30	307	0	112	0
7	0	325	698	0	3	12

	1	2	3
1	1	322	749
2	420	48	11
3	0	7	954
4	0	226	189
5	1	432	37
7	0	960	78

Figure 7: Two correlation matrices with k=6 and k=3 clusters

The output of the model also produced the cluster plot shown in figure 8 below. Where 3 clusters were created from the original data. As discussed, there was some overlap between the clusters, which is also shown in the plot below.

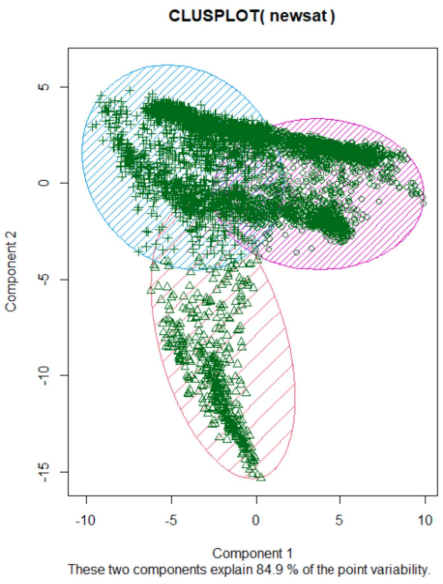

Figure 8: Cluster plot of the k-means model

Evaluation:

As discussed, it was decided to use k=3 clusters when creating this unsupervised learning model. Figure 7 shows that classes 2, 3, 5, and 7 were successful in having many of the values in one cluster, while classes 1 and 4 had values split between the 2nd and 3rd clusters. When the highest number of each row was added together it was shown that 749+ 420+954+226+432+ 960= 3741 out of 4435 observations are clustered in agreement (UMGC. 2023). This comes down to roughly 84% of the total observations clustered in agreement across the 3 clusters. This is a very good showing that the model can segment four out six of the total classes, with a very high percentage of the observations in agreement. When looking into the clusters in figure 7, it is shown that cluster 1 has 420 class 2 values with very little overlap from other classes, while clusters 2 and 3 had a lot of overlap from the rest of the classes. This shows that the K-means model can successfully segment class 2 but finds a lot of relationships across the rest of the classes. An example is that 749 class 1 values and 954 class 3 values are in cluster 3, showing that class 1

and 3 have a connection across the observations. Overall, the model did a very good job at segmenting the data, however future work will require diving deeper into each of the clusters to understand why these values overlapped in the clusters.

Conclusion:

Summary:

The objective of segmenting and categorizing the 3x3 images was successful, with 84% of the total observations in agreement. The data was segmented into *k=3* clusters, where cluster 1 had very little overlap, while clusters 2 and 3 did. The overlap between multiple classes in clusters 2 and 3 will require future work to understand the relationships between each of the classes involved. Overall, the model was successful in segmenting classes 2,3,5, and 7 while it struggled with classes 1 and 4. The objective of the analysis was successful, with 84% cluster agreement. However, future work is necessary to understand the relationships between the classes that overlapped in clusters 2 and 3 and why classes 1 and 4 were not fully segmented by the model.

Limitations:

The first limitation came within the data itself. It was only a small subset of what was gathered by NASA, which means a lot of insight could be missing due to lack of information. The reason for this was because NASA did not want the images to be reconstructed, likely because of the sensitivity of the matter. Since the K-means model is largely an exploratory method, missing critical information like the original pictures can block the analyst from key insights.

Improvement Areas:

An area of improvement is to investigate each of the anomalies discussed previously. The code in figure 6 outputs ten outliers that can be analyzed to understand how each outlier is creat-

ed. From there the process of capturing the data could be fixed, or the data could be cleaned to fix some of the issues that may come from the outliers in the data. Another area of improvement is to dive deeper into the data itself. Since NASA made it so the images could not be recreated, there is a lot of missing information from the analyst. By looking at each image, and comparing it to the clusters and outliers, the analyst can start to pick out some of the similarities and relationships that the K-means model found.

References:

Srinivasan, A. (1993, February 12). Landsat Satellite Data Set. UCI Machine Learning Repository. https://archive.ics.uci.edu/dataset/146/statlog+landsat+satellite

NASA. (2022, March 24). Landsat 9. Landsat Science | A joint NASA/USGS Earth observation program. https://landsat.gsfc.nasa.gov/satellites/landsat-9/

UMGC Data 630. (2023). Unsupervised Learning Clustering https://learn.umgc.edu/d2l/common/dialogs/quickLink/quickLink.d2l?ou=770634&type=coursfile&fileId=Lecture+Slides%2fUnsupervised+Learning+Clustering.pptx]. UMGC DATA 630.