Anay Gupta
CSE 494: AI for Cyber Security
Shakarian - Friday 1 pm
February 15th, 2019

CSE 494 Homework 1

1. K-means Clustering

a).

Attribute	Attribute Type
Product Version	Categorical
Image Version	Ordinal
File Type	Categorical
Initialized Data Size	Categorical
Entry Point	Textual
Subsystem	Categorical
Linker Version	Ordinal
Code Size	Numerical (could also be ordinal maybe?)
Comments	Textual
Template	Categorical
Language Code	Categorical
Internal Name	Categorical
Shared Doc	Categorical
OS Version	Ordinal
Company Website	Textual
Character Set	Categorical
File OS	Categorical
File Flags Mask	Categorical (maybe Textual to some degree like Entry Point and therefore, might need n-grams)
Original File Name	Textual
Object File Type	Categorical

b). *Steps i and ii are run with max_iter = '300' (default)*

i. Cluster sizes for k = 4 to k = 6:

When k = 4

Cluster size of cluster 0 = **3**
Cluster size of cluster 1 = **5**
Cluster size of cluster 2 = **1**
Cluster size of cluster 3 = **284**

When k = 5

Cluster size of cluster 0 = **3**
Cluster size of cluster 1 = **1**
Cluster size of cluster 2 = **75**
Cluster size of cluster 3 = **211**
Cluster size of cluster 4 = **3**

When k = 6

Cluster size of cluster 0 = **120**
Cluster size of cluster 1 = **3**
Cluster size of cluster 2 = **1**
Cluster size of cluster 3 = **91**
Cluster size of cluster 4 = **75**
Cluster size of cluster 5 = **3**

ii. Inertia for k = 4 to k = 6:

When k = 4

Inertia = **235845462703011.66**

When k = 5

Inertia = **28180727341347.594**

When k = 6

Inertia = **26382070125474.297**

None of these inertias are scaled, hence they are huge values.

iii. Silhouette coefficient for k = 6, init = 'random', and max_iter = 100:

Silhouette Score (evaluation of K means on unscaled data) = **0.5215915759792641**

iv. Clustering output (Precision, Recall, F1) based on malware family labels:

Still using similar attributes as part iii (k = 6, init = 'random', max_iter = 100),

Precision = 107 / 293 = **0.36519**

Recall = 242 / 293 = **0.82594**
F1 = (2* 0.36519* 0.82594) / (0.36519+ 0.82594) = **0.50645**

Side Experiment problem:

- Analyzed one feature at a time to draw conclusions about which malware features are best descriptive of the families in terms of changes to precision and recall

1. Dropping all features except LinkerVersion:
   Precision = 141 / 293 = **0.48123**
   Recall = 230 / 293 = **0.78498**
   F1 score = (2* 0.48123* 0.78498) / (0.48123 + 0.78498) = **0.59667**
2. Dropping all features except Code Size:
   Precision = 110 / 293 = **0.37543**
   Recall = 186 / 293 = **0.6348**
   F1 Score = (2* 0.37543* 0.6348) / (0.37543+ 0.6348) = **0.47182**
3. Dropping all features except Initialized Data Size:
   Precision = 137 / 293 = **0.46758**
   Recall = 213 / 293 = **0.72696**
   F1 Score = (2* 0.46758* 0.72696) / (0.46758+ 0.72696) = **0.56911**

- Based on the changes in precision and recall when each malware feature (out of the chosen four) were analyzed independently, it seems that **LinkerVersion and Data Size are the more family-descriptive features.** This is because these features had higher precision/recall/f1 scores which implies that they are **getting more of the data set** (goal of recall is to "find everything") and **correctly classifying malware at a higher rate** (goal of precision is to get low false positives).
- The F1 score of K means on feature LinkerVersion is **0.59667**. The F1 score of K means on feature Initialized Data Size is **0.56911**.

Bonus Problem

Running agglomerative clustering algorithm (with single linkage, euclidean affinity and 6 clusters) on the same feature matrix obtained in question 1b:

Precision = 58 / 293 = **0.19795**
Recall = 283 / 293 = **0.96587**
F1 score = (2* 0.19795* 0.96587) / (0.19795 + 0.96587) **= 0.32856**

The main drawback in terms of interpretation when agglomerative clustering is compared with k-means clustering is visualization. Hierarchical clustering is generally applicable to a small set of data while K-means is very useful for large data sets. In

terms of interpretation, it is hard to clearly visualize the final output while using hierarchical clustering (dendrograms are useful but lots of overlap occurs to the point where it becomes hard to interpret).