**MACHINE LEARNING PROJECT**

**MAC-LRN**

**DETAILED DISCUSSION OF MILESTONE 1 AND 2**

**GROUP MEMBERS:**

**AREVALO, MARK ANTHONY A.**

**NABLO, JUAN PAOLO J.**

**VIVO, ROI EMMANUEL A.**

**DATA SUBMITTED: MAR 28, 2016**

**MACHINE LEARNING PROJECT**

**Milestone #1**

**SPAMBASE**

**A. Dataset Building and Normalization**

Provide the following details:

1. Source of Dataset: UCI Machine Learning Repository
2. Description of Dataset (include the number of original features and the sample size):

This dataset contains a collection of spam emails from a set of individuals who have filed spam to the owner’s postmaster. Within the dataset, there exists a collection of non-spam emails as well. There are 4601 samples and 58 features.

1. Name, Description and type of each feature. Include here if you had to normalize the feature, the reason for its normalization and the range of values (e.g. 0 to 1).

To lessen confusion, features 1-58 have been labelled alphabetically.

48 continuous real [0,100] attributes of type **word\_freq\_WORD**

= percentage of words in the e-mail that match WORD, i.e. 100 \* (number of times the WORD appears in the e-mail) / total number of words in e-mail. A "word" in this case is any string of alphanumeric characters bounded by non-alphanumeric characters or end-of-string.

6 continuous real [0,100] attributes of type **char\_freq\_CHAR**

= percentage of characters in the e-mail that match CHAR, i.e. 100 \* (number of CHAR occurrences) / total characters in e-mail

1 continuous real [1,...] attribute of type **capital\_run\_length\_average**

= average length of uninterrupted sequences of capital letters

1 continuous integer [1,...] attribute of type **capital\_run\_length\_longest**

= length of longest uninterrupted sequence of capital letters

1 continuous integer [1,...] attribute of type **capital\_run\_length\_total**

= sum of length of uninterrupted sequences of capital letters

= total number of capital letters in the e-mail

1 nominal {0,1} class attribute of type **spam**

= denotes whether the e-mail was considered spam (1) or not (0), i.e. unsolicited commercial e-mail.

1. Processed/Normalized dataset If the dataset cannot be uploaded because of its size, provide a link to it (via Dropbox or Google drive)

\*\*\* Insert New Link if ever there’s a new one \*\*\*

1. Number of features that were removed and the reason for their removal

0 features were removed. All features are relevant.

**B. Performance of kNN, Decision Trees, Bayesian Network on Different Feature Sets**

1. Performance measures to include are Accuracy, Precision, Recall, F-Measure, AUC and Kappa statistic
2. Create one table for each measure indicated above. Use the table below as your guide. Note that you need to provide the performance when the complete feature set is used for model building as well as when different feature selection techniques are applied.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search**  **(BACKWARD ELIMINATION)** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | 90.31 | 89.57 | 91.09 | 72.09 |
| **kNN (k=3)** | 90.35 | 91.92 | 91.33 | 71.85 |
| **kNN (k=5)** | 89.94 | 92.44 | 90.87 | 72.03 |
| **kNN (k=7)** | 89.63 | 92.24 | 91.37 | 71.72 |
| **kNN (k=9)** | 89.24 | 91.91 | 90.85 | 72.03 |
| **Decision Trees** | 88.35 | 90.59 | 90.55 | 82.03 |
| **Bayesian Network (e.g. Naïve Bayes)** | 79.79 | 81.79 | 82.81 | 62.51 |
| **MLP (Training Cycles = 10)** | 91.28 |  |  | 80.72 |

**Table 1 – Accuracy (%)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search**  **(BACKWARD ELIMINATION)** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | 91.92 | 93.01 | 92.66 | 86.03 |
| **kNN (k=3)** | 91.15 | 92.91 | 91.91 | 85.14 |
| **kNN (k=5)** | 90.81 | 93.42 | 91.64 | 84.88 |
| **kNN (k=7)** | 90.49 | 93.48 | 91.98 | 84.37 |
| **kNN (k=9)** | 89.63 | 92.73 | 91.31 | 84.44 |
| **Decision Trees** | 85.81 | 89.13 | 90.14 | 84.21 |
| **Bayesian Network (e.g. Naïve Bayes)** | 96.74 | 79.65 | 95.94 | 97.36 |
| **MLP (Training Cycles = 10)** | 90.71 |  |  | 82.72 |

**Table 2 – Precision (%)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search**  **(BACKWARD ELIMINATION)** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | 92.14 | 89.53 | 92.65 | 64.42 |
| **kNN (k=3)** | 93.15 | 93.83 | 93.97 | 64.89 |
| **kNN (k=5)** | 92.83 | 94.19 | 93.47 | 65.53 |
| **kNN (k=7)** | 92.65 | 93.76 | 93.97 | 65.46 |
| **kNN (k=9)** | 93.04 | 94.05 | 93.87 | 66.03 |
| **Decision Trees** | 96.88 | 96.23 | 94.84 | 86.62 |
| **Bayesian Network (e.g. Naïve Bayes)** | 68.97 | 94.01 | 74.82 | 39.20 |
| **MLP (Training Cycles = 10)** | 87.2 |  |  | 86.26 |

**Table 3 – Recall (%)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search**  **(BACKWARD ELIMINATION)** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | 92.02 | 91.22 | 92.64 | 73.66 |
| **kNN (k=3)** | 92.13 | 93.36 | 92.92 | 73.63 |
| **kNN (k=5)** | 91.79 | 93.79 | 92.54 | 73.93 |
| **kNN (k=7)** | 91.54 | 93.61 | 92.95 | 73.69 |
| **kNN (k=9)** | 91.29 | 93.38 | 92.56 | 74.09 |
| **Decision Trees** | 90.99 | 92.54 | 92.40 | 85.37 |
| **Bayesian Network (e.g. Naïve Bayes)** | 80.51 | 86.22 | 84.05 | 55.85 |
| **MLP (Training Cycles = 10)** | 88.7 |  |  | 84.42 |

**Table 4 – F-Measure (%)**

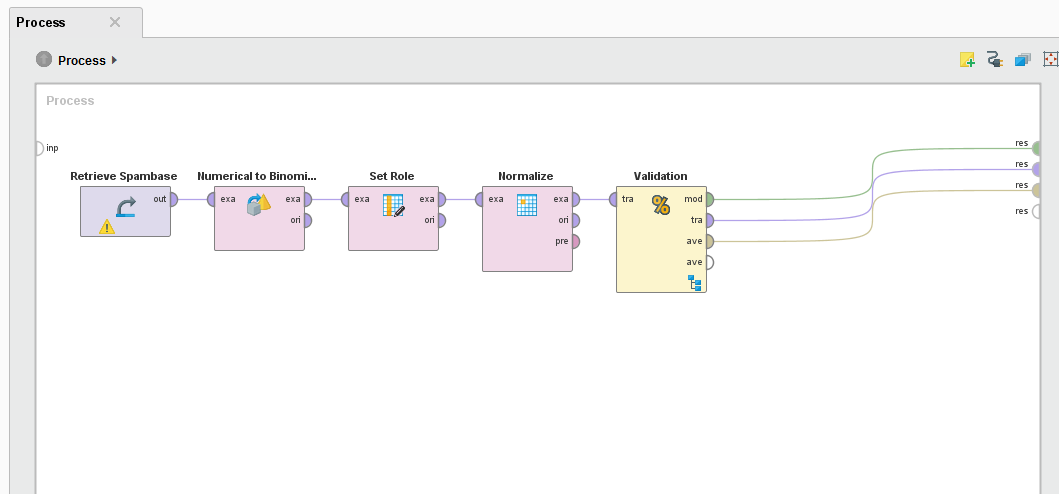
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search**  **(BACKWARD ELIMINATION)** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | 0.5 | 0.5 | 0.5 | 0.5 |
| **kNN (k=3)** | 0.919 | 0.925 | 0.919 | 0.756 |
| **kNN (k=5)** | 0.936 | 0.946 | 0.935 | 0.766 |
| **kNN (k=7)** | 0.953 | 0.969 | 0.955 | 0.771 |
| **kNN (k=9)** | 0.954 | 0.958 | 0.958 | 0.769 |
| **Decision Trees** | 0.864 | 0.891 | 0.907 | 0.853 |
| **Bayesian Network (e.g. Naïve Bayes)** | 0.934 | 0.872 | 0.936 | 0.846 |
| **MLP (Training Cycles = 10)** | 0.959 |  |  | 0.875 |

**Table 5 – AUC**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search**  **(BACKWARD ELIMINATION)** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | 0.797 | 0.784 | 0.813 | 0.453 |
| **kNN (k=3)** | 0.797 | 0.830 | 0.817 | 0.446 |
| **kNN (k=5)** | 0.788 | 0.841 | 0.808 | 0.448 |
| **kNN (k=7)** | 0.782 | 0.837 | 0.818 | 0.442 |
| **kNN (k=9)** | 0.772 | 0.830 | 0.807 | 0.447 |
| **Decision Trees** | 0.747 | 0.798 | 0.799 | 0.621 |
| **Bayesian Network (e.g. Naïve Bayes)** | 0.607 | 0.599 | 0.660 | 0.324 |
| **MLP (Training Cycles = 10)** | 0.816 |  |  | 0.592 |

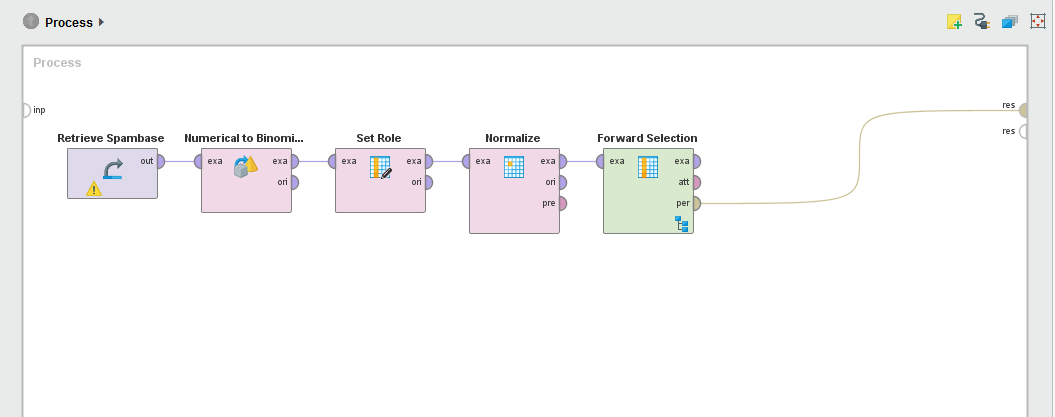
**Table 6 – Kappa Statistic**

1. **Processes**
   1. Complete Feature Set

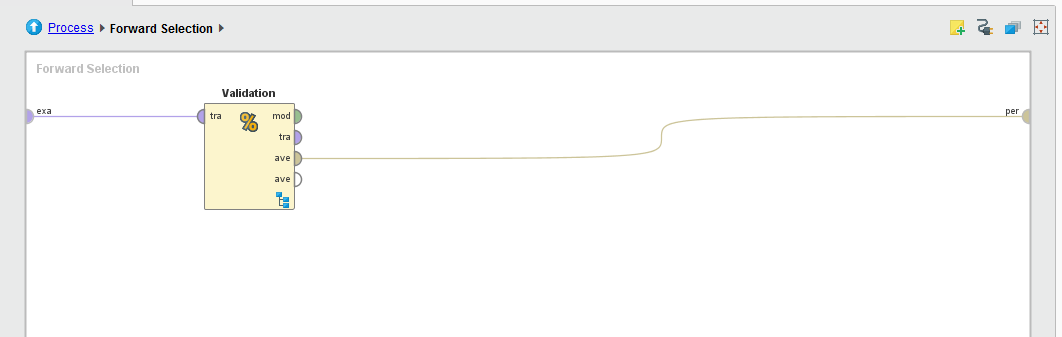


**Figure 1 –** Complete Feature Set process

* 1. Feature Selection using Forward Search

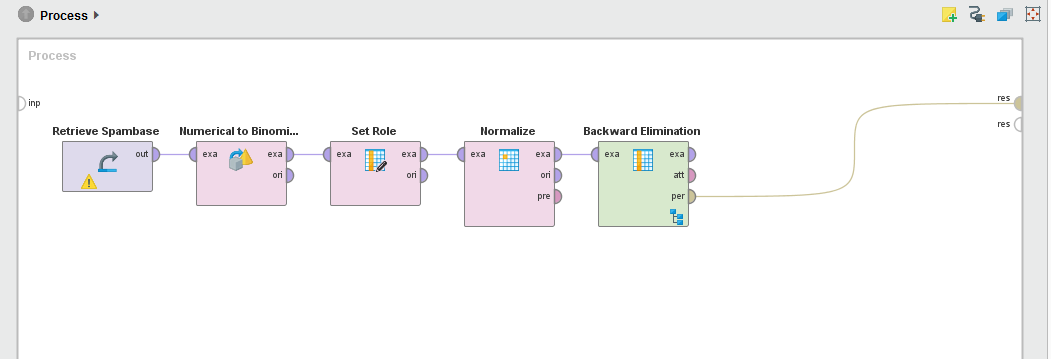


**Figure 2 –** Feature Selection using Forward Search process

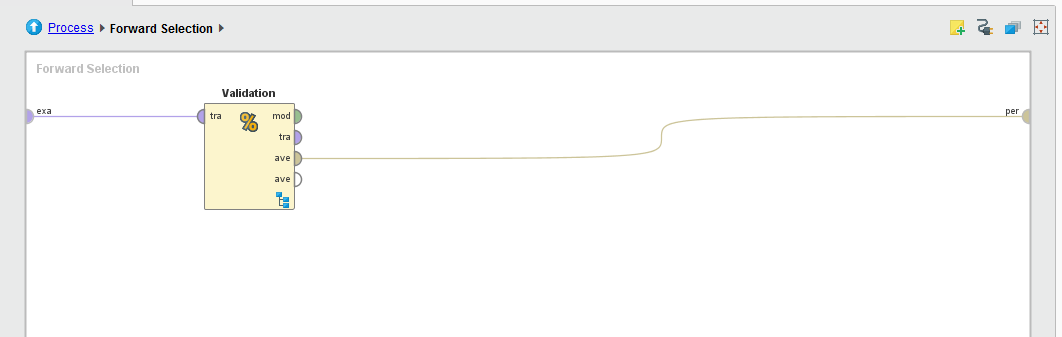


**Figure 3 –** Inside Forward Selection

* 1. Feature Selection using Backward Search



**Figure 4 –** Feature Selection using Backward Search process



**Figure 5 –** Inside Backward Elimination

* 1. Top 9 features based on 3 Components of Principal Components Analysis (PCA)

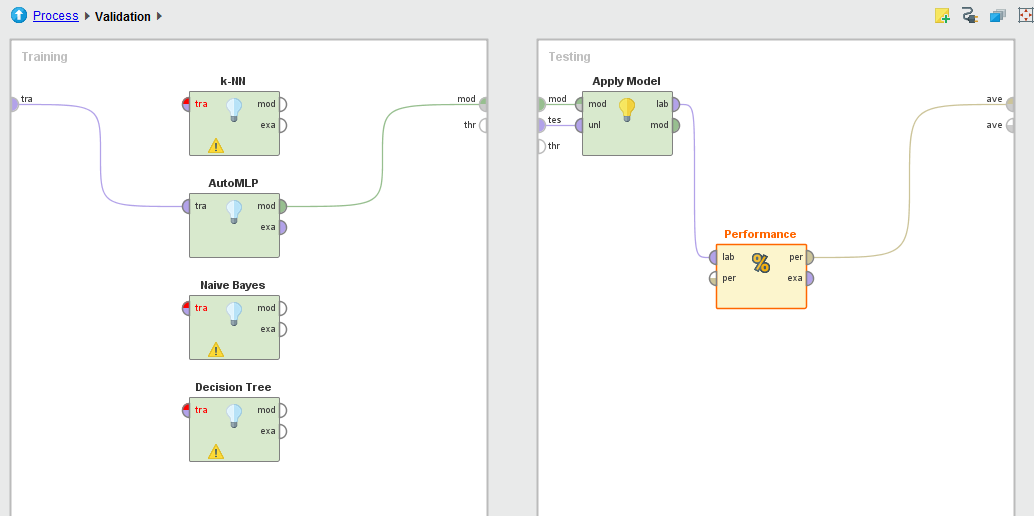
\*INSERT PROCESS\*

**Figure 6 –** PCA process

The top 9 features are: \*\*\*

The resulting dataset with only top 9 features was used at the Complete Feature Set process as well. The results of this were placed under this category.

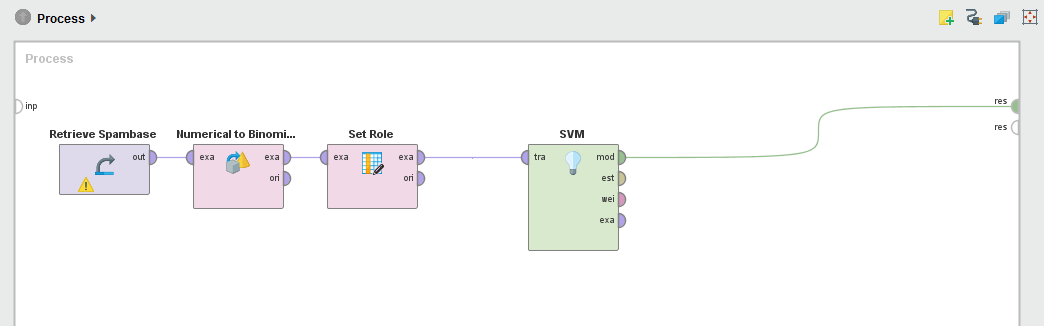
* 1. Validation



**Figure 7 –** Inside Validation

Validation was set to have the one of the classifiers: k-NN, autoMLP, Naïve Bayes, or Decision Tree. Performance is set to be binominal classification.

* 1. SVM



**Figure 8 –** SVM process

**C. Discussion of Results**

Discuss the results, determine which classifier is better and explain the reasons for choosing such classifier.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Classifier** | **Performance Measures** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search (Backward Elimination)** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | **Accuracy** | 90.31% | 89.57% | 91.09% | 72.09% |
| **Precision** | 91.92% | 93.01% | 92.66% | 86.03% |
| **Recall** | 92.14% | 89.53% | 92.65% | 64.42% |
| **F-Measure** | 92.02% | 91.22% | 92.64% | 73.66% |
| **AUC** | 0.5 | 0.5 | 0.5 | 0.5 |
| **Kappa** | 0.797 | 0.784 | 0.813 | 0.453 |
| **kNN (k=3)** | **Accuracy** | 90.35% | 91.92% | 91.33% | 71.85% |
| **Precision** | 91.15% | 92.91% | 91.91% | 85.14% |
| **Recall** | 93.15% | 93.83% | 93.97% | 64.89% |
| **F-Measure** | 92.13% | 93.36% | 92.92% | 73.63% |
| **AUC** | 0.919 | 0.925 | 0.919 | 0.756 |
| **Kappa** | 0.797 | 0.83 | 0.817 | 0.446 |
| **kNN (k=5)** | **Accuracy** | 89.94% | 92.44% | 90.87% | 72.03% |
| **Precision** | 90.81% | 93.42% | 91.64% | 84.88% |
| **Recall** | 92.83% | 94.19% | 93.47% | 65.53% |
| **F-Measure** | 91.79% | 93.79% | 92.54% | 73.93% |
| **AUC** | 0.936 | 0.946 | 0.935 | 0.766 |
| **Kappa** | 0.788 | 0.841 | 0.808 | 0.448 |
| **kNN (k=7)** | **Accuracy** | 89.63% | 92.24% | 91.37% | 71.72% |
| **Precision** | 90.49% | 93.48% | 91.98% | 84.37% |
| **Recall** | 92.65% | 93.76% | 93.97% | 65.46% |
| **F-Measure** | 91.54% | 93.61% | 92.95% | 73.69% |
| **AUC** | 0.953 | 0.969 | 0.955 | 0.771 |
| **Kappa** | 0.782 | 0.837 | 0.818 | 0.442 |
| **kNN (k=9)** | **Accuracy** | 89.24% | 91.91% | 90.85% | 72.03% |
| **Precision** | 89.63% | 92.73% | 91.31% | 84.44% |
| **Recall** | 93.04% | 94.05% | 93.87% | 66.03% |
| **F-Measure** | 91.29% | 93.38% | 92.56% | 74.09% |
| **AUC** | 0.954 | 0.958 | 0.958 | 0.769 |
| **Kappa** | 0.772 | 0.83 | 0.807 | 0.447 |
| **Decision Trees** | **Accuracy** | 88.35% | 90.59% | 90.55% | 82.03% |
| **Precision** | 85.81% | 89.13% | 90.14% | 84.21% |
| **Recall** | 96.88% | 96.23% | 94.84% | 86.62% |
| **F-Measure** | 90.99% | 92.54% | 92.40% | 85.37% |
| **AUC** | 0.864 | 0.891 | 0.907 | 0.853 |
| **Kappa** | 0.747 | 0.798 | 0.799 | 0.621 |
| **Bayesian Network (e.g. Naïve Bayes)** | **Accuracy** | 79.79% | 81.79% | 82.81% | 62.51% |
| **Precision** | 96.74% | 79.65% | 95.94% | 97.36% |
| **Recall** | 68.97% | 94.01% | 74.82% | 39.20% |
| **F-Measure** | 80.51% | 86.22% | 84.05% | 55.85% |
| **AUC** | 0.934 | 0.872 | 0.936 | 0.846 |
| **Kappa** | 0.607 | 0.599 | 0.66 | 0.324 |
| **MLP (Training Cycles= 10)** | **Accuracy** | 91.28% |  |  | 80.72% |
| **Precision** | 90.71% |  |  | 82.72% |
| **Recall** | 87.20% |  |  | 86.26% |
| **F-Measure** | 88.70% |  |  | 84.42% |
| **AUC** | 0.959 |  |  | 0.875 |
| **Kappa** | 0.816 |  |  | 0.592 |

**Table 7 – All Performance Measures**

By comparing the performance measures of Complete Feature Set, Feature Selection using Forward and Backward Search, and PCA, Feature Selection using Forward Search with kNN (k=5) is found to be the best classifier. Feature Selection using Forward Search has greater performance and more consistent values compared to other processes. For Forward Search, as k is increased, the performance values becomes greater. But when k=7, the values decreased. Decision Trees and Bayesian Network did not give greater results compared to kNN.

\*\*\* INSERT SVM RESULTS HERE \*\*\*

**D. Problems Encountered**

By using the previous dataset (Water Quality Complaints), Part B of the Milestone 1 results to very low percentages in its measures. The group have found this unacceptable. In order to solve this problem, the group have attempted to modify the dataset by cleaning it. Unfortunately, this method did not work. Since there was limited time, the group has decided to look for another dataset (Spambase).

The group has encountered lengthy process run time, especially with k-NN and MLP. There was no solution to this so the group just had to wait for the process to finish.

MLP has not been finished for the forward and backward selection. Processing time has passed almost 2 days and the results were not available before the deadline of this paper.

**MAC-LRN**

**MILESTONE 2**

Analyze your 2 datasets by applying k-means clustering and self-organizing maps techniques. Have a detailed discussion on each dataset and be guided by the following:

**Data set 1: Spambase**

**A. k-Means Clustering**

a) K=2

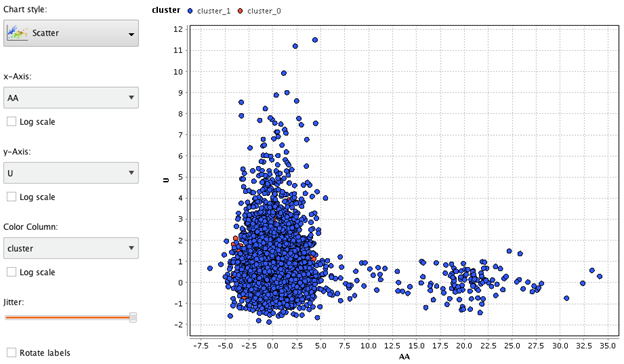


Figure 1: Clustering – k=2 – Scatter Plot

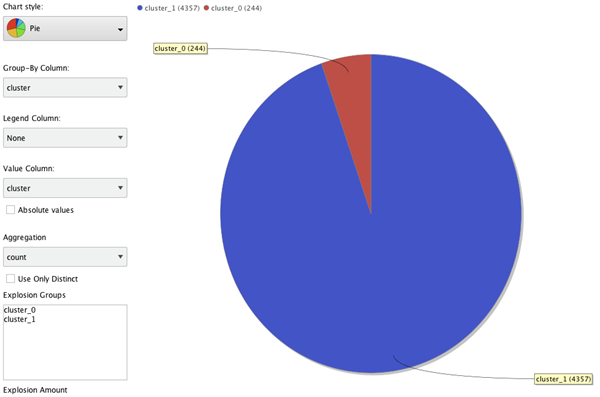


Figure 2: Clustering – k=2 – Pie Chart

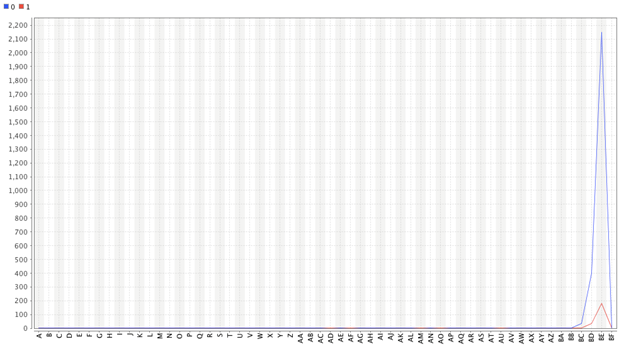


Figure 3: Plot View

Max value for Centroid 0 and 1 is feature BD.

b) K=20

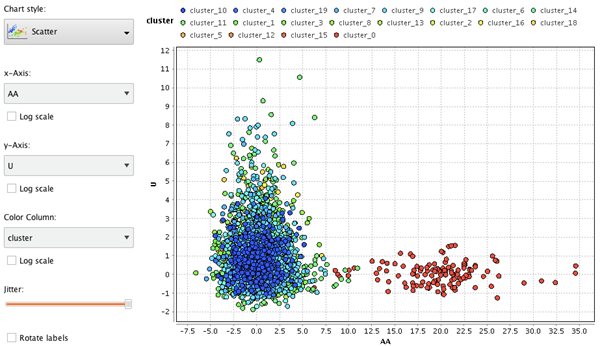


Figure 4: Clustering – k=20 – Scatter Plot

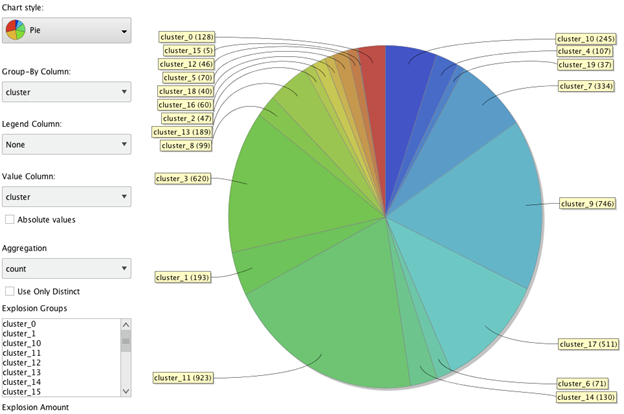


Figure 5: Clustering – k=20 – Pie Chart

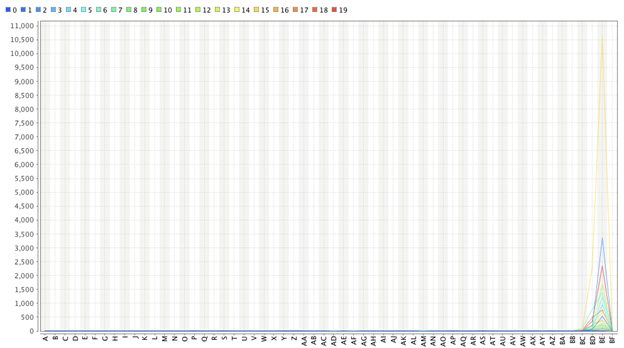


Figure 6: K=20 Plot View

**K=20**

1. Find the best set of clusters by performing several experiments. Once the number of clusters have been finalized, describe each cluster according to a) its distinct features and b) the kind of samples (or class distribution) it has. Compare the different clusters.

a. Based on the data, cluster 0 from the spam mail indicator feature dictates that this is supposedly not spam because the contradicts to the set of clusters which are related to cluster 10, and cluster 0 means that one sample is a spam letter. On the other hand, the remaining clusters would indicate that these are not spam letters.

The process was done by testing numerous K values. Though the results were not appropriate if the values were set to 1-10. Once K was set to have 20, the clusters were more noticeable. The features of every cluster are almost the same. The feature BD is dominant for all clusters.

b. On the data, cluster 0 indicates that these samples are spam letters, different with the rest of the samples in the other clusters. This means that the remaining clusters are not spam letters. In one of the features most data for cluster 0 have similar results in the AA feature. One sample indicator is that a feature labeled the AA feature indicates that an email is a spam when the keyword “WORD” appears in the body more than 10 times. Most samples in cluster 0 have values ranging from 10 to 30 in the AA feature.

2. Show the number of clusters as well as the number of samples in each cluster.

20 clusters (cluster\_0 – cluster\_19)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Cluster** | **Number of samples** | **Cluster** | **Number of samples** | **Cluster** | **Number of samples** |
| 0 | 128 | 8 | 99 | 16 | 60 |
| 1 | 193 | 9 | 746 | 17 | 511 |
| 2 | 47 | 10 | 245 | 18 | 40 |
| 3 | 620 | 11 | 923 | 19 | 37 |
| 4 | 107 | 12 | 46 |  |  |
| 5 | 70 | 13 | 189 |  |  |
| 6 | 71 | 14 | 130 |  |  |
| 7 | 334 | 15 | 5 |  |  |

Figure 7: 20 clusters (cluster\_0 – cluster\_19)

3. Explain the reasons for choosing such a number of clusters. You can apply the objective function (mentioned in the slides) to assess the quality of each cluster. You can mention the results of your previous experiments that may help justify for the choice of k clusters.

Starting with a number of 1 to 10 for the number of clusters did not show appropriate results. It only displayed 1 major cluster with a similar color. Choosing the k as 20 displayed several clusters that had different colors from each other. Also, even though there are different clusters, only 1 cluster was different from the others based from color shades. This solution gave a result that could be used for clustering two types of samples easily. The solution may become better if k is greater than 20.

4. Process

\*\*\*INSERT PROCESS HERE \*\*\*

4. Summary and Conclusion of the experiments

The k-means clustering is a process that groups samples together with similar characteristics from other clusters with different qualities. Based on the data recorded for Milestone 1, the top 2 features for PCA was referred to for the x and y axis for the Cluster Plot. The charts above have been generated by doing this process. Based on the data gathered, the clusters are more similar than dissimilar. Because of this issue, the class is difficult to distinguish.

**B. SOM**

1. Train a 10x10 SOM map for as many iterations/cycles as possible in order to generate a good and specialized set of weights for each node. Include the generated map in the document.

1. 1,000 Training Rounds c. 15,000 Training Rounds
2. 10,000 Training Rounds

2. Make sure that the SOM map is labelled or color-coded (based on labels). A map is well-organized if nodes having the same label are close to each other.

a. 1,000 Training Rounds

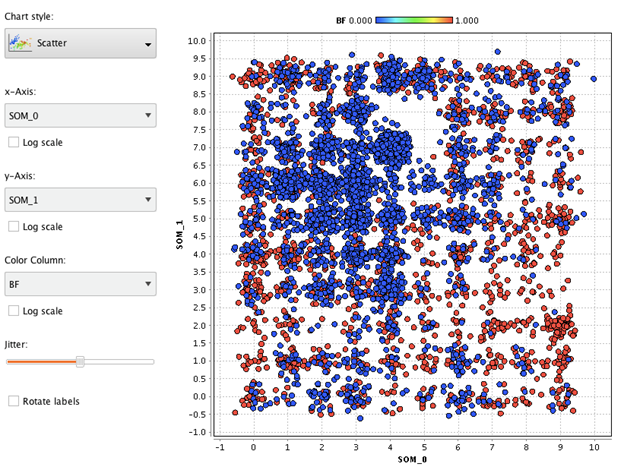


Figure 8: Scatter Plot -1,000 Training Rounds

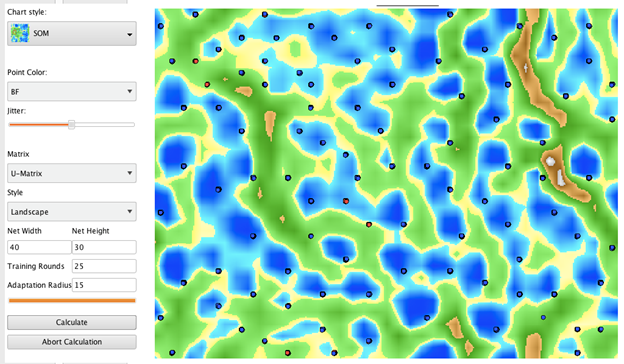


Figure 9: SOM Plot - 1,000 Training Rounds

b. 10,000 Training Rounds

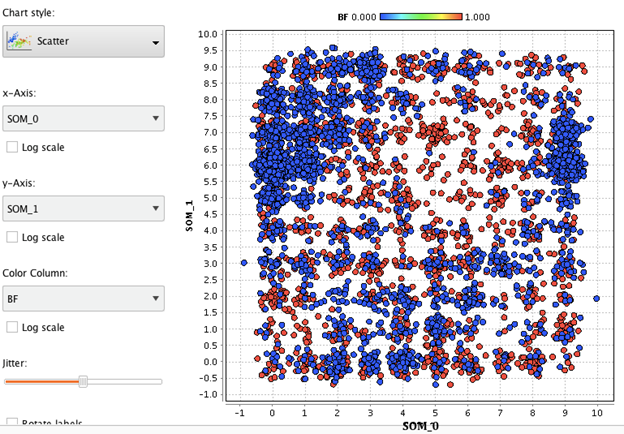


Figure 10: Scatter Plot - 10,000 Training Rounds

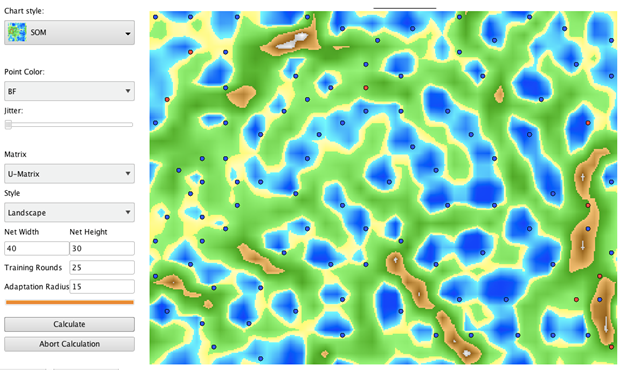


Figure 11: SOM - 10,000 Training Rounds

c. 15,000 Training Rounds

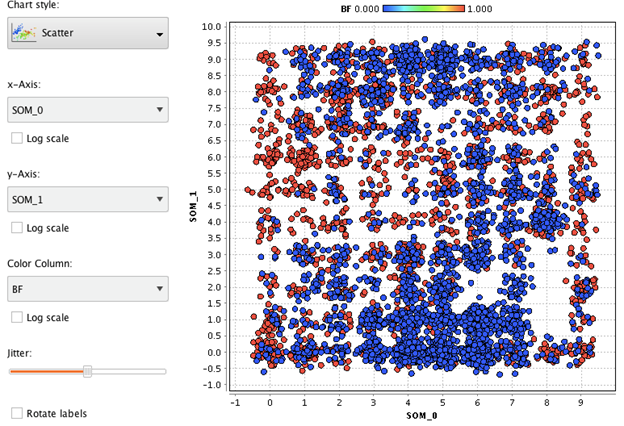


Figure 12: Scatter Plot - 15,000 Training Rounds

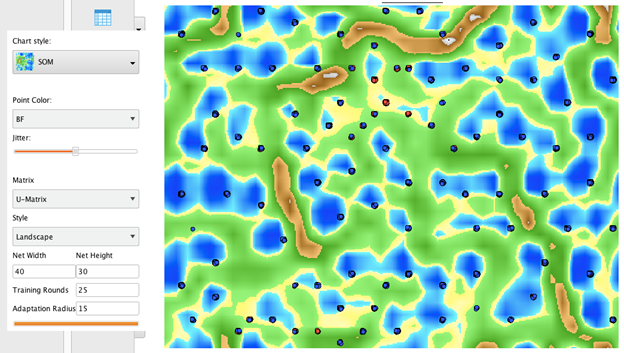


Figure 13: SOM - 15,000 Training Rounds

3. Interpret the results of the map. Analyze the kind of samples that were assigned on nodes that are close in the map. For example, determine the kind of samples that were assigned in the upper left corner nodes of the map. Have some analyses on several areas of the map.

Based on all the maps, it is difficult to distinguish what kind of samples are assigned to the parts of the map especially with no labels on it. Just like the clustering process, the maps are unorganized though the map with 1000 training rounds is more “clean” compared to the others.

4. Process

\*\*\* INSERT PROCESS HERE \*\*\*

5. Summary and Conclusion of the experiments

Based on the results, Using SOM with different training rounds such as 1000, 10000, and 15000 on the dataset did not organize the map. However, when compared to all the outputs, SOM with 1000 training rounds look more organized than the other training rounds.

Even with unsupervised learning, the dataset was still unorganized due to similarities of the data in the features. The data can still be used for analysis, but it will not be as accurate for predictions.