**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)

<https://www.dropbox.com/sh/xyv2ce9j04iuxdd/AACu0Xr_GNOyT-YPJPbFZRPva?dl=0>

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 (%)**

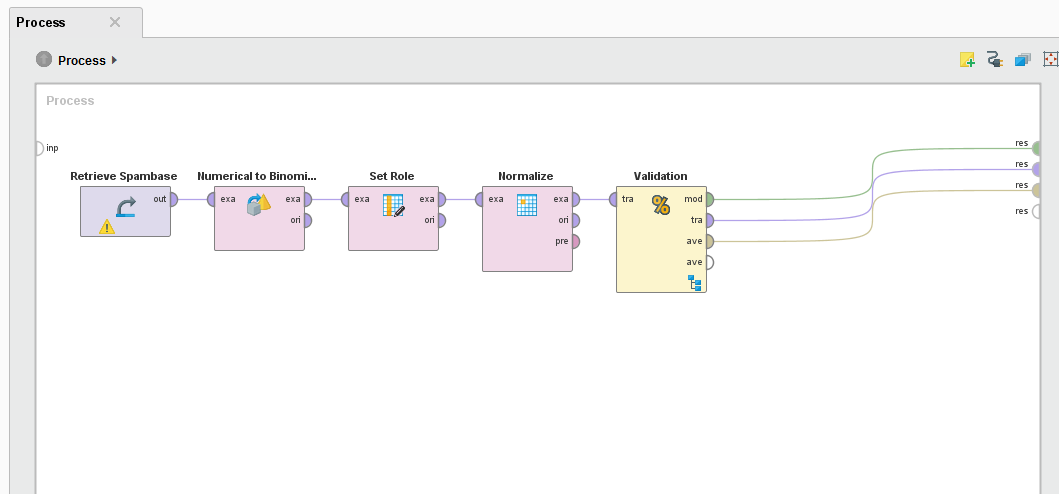
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search**  **(BACKWARD ELIMINATION)** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components 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 |

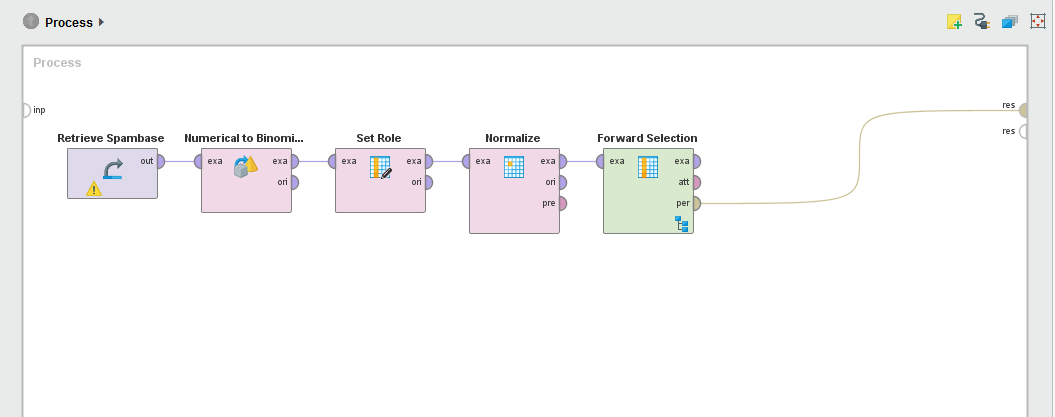
**Table 6 – Kappa Statistic**

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



**Figure 1 –** Complete Feature Set process

* 1. Feature Selection using Forward Search

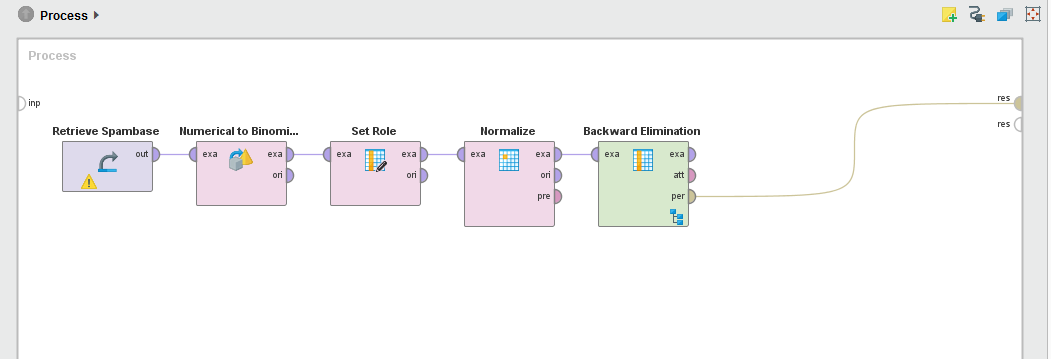


**Figure 2 –** Feature Selection using Forward Search process



**Figure 3 –** Inside Forward Selection

* 1. Feature Selection using Backward Search

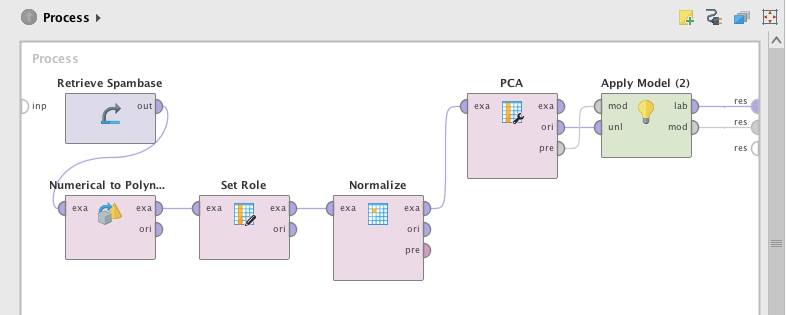


**Figure 4 –** Feature Selection using Backward Search process



**Figure 5 –** Inside Backward Elimination

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

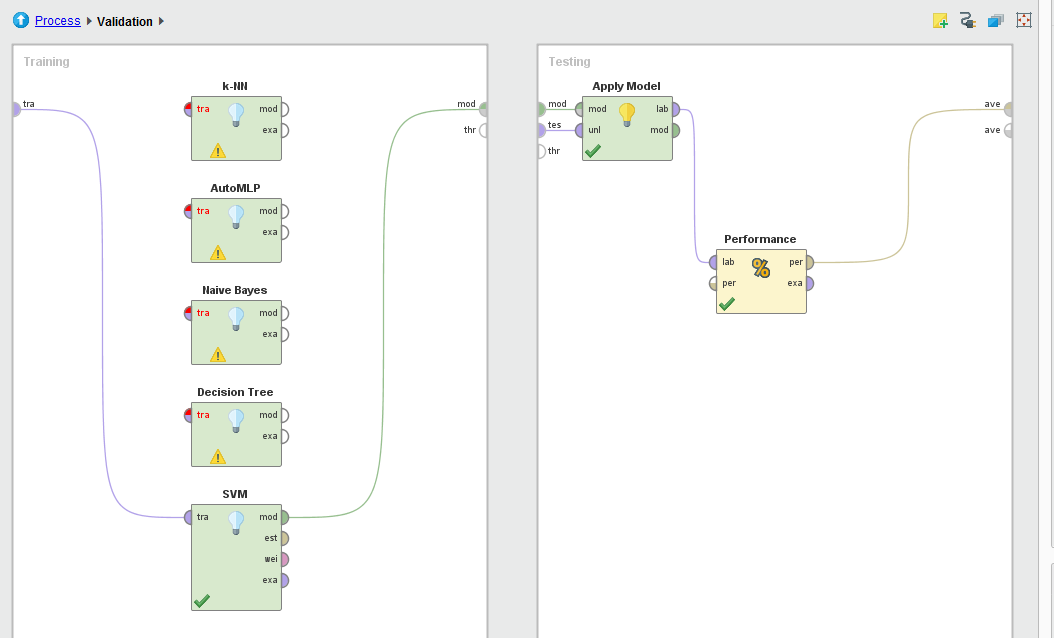


**Figure 6 –** PCA process

The top 9 features are: C, S, U, Y, AA, AD AF, AH and AN. These were done by getting the top 9 features based on the highest values from PC1, PC2, PC3 (refer to MS1 excel file).

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, SVM , Naïve Bayes, or Decision Tree. Performance is set to be binominal classification.

**C. Discussion of Results**

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

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CFS** |  |  |  |  |  |  |
|  | **Accuracy** | **Precision** | **Recall** | **F-Measure** | **AUC** | **Kappa** |
| **kNN (k=1)** | 90.31 | 91.92% | 92.14% | 92.02% | 0.5 | 0.797 |
| **kNN (k=3)** | 90.35 | 91.15% | 93.15% | 92.13% | 0.919 | 0.797 |
| **kNN (k=5)** | 89.94 | 90.81% | 92.83% | 91.79% | 0.936 | 0.788 |
| **kNN (k=7)** | 89.63 | 90.49% | 92.65% | 91.54% | 0.953 | 0.782 |
| **kNN (k=9)** | 89.24 | 89.63% | 93.04% | 91.29% | 0.954 | 0.772 |
| **Decision Trees** | 88.35 | 85.81% | 96.88% | 90.99% | 0.864 | 0.747 |
| **Bayesian Network (e.g. Naïve Bayes)** | 79.79 | 96.74% | 68.97% | 80.51% | 0.934 | 0.607 |
| **MLP (Training Cycles = 10)** | 91.28% | 90.71% | 87.20% | 88.70% | 0.959 | 0.816 |
| **SVM** | 89.78% | 93.09% | 80.08% | 86.04% | 0.962 | 0.781 |

**Table 7 – Complete Feature Set**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Forward** |  |  |  |  |  |  |
|  | **Accuracy** | **Precision** | **Recall** | **F-Measure** | **AUC** | **Kappa** |
| **kNN (k=1)** | 89.57 | 93.01% | 89.53% | 91.22% | 0.5 | 0.784 |
| **kNN (k=3)** | 91.92 | 92.91% | 93.83% | 93.36% | 0.925 | 0.83 |
| **kNN (k=5)** | 92.44 | 93.42% | 94.19% | 93.79% | 0.946 | 0.841 |
| **kNN (k=7)** | 92.24 | 93.48% | 93.76% | 93.61% | 0.969 | 0.837 |
| **kNN (k=9)** | 91.91 | 92.73% | 94.05% | 93.38% | 0.958 | 0.83 |
| **Decision Trees** | 90.59 | 89.13% | 96.23% | 92.54% | 0.891 | 0.798 |
| **Bayesian Network (e.g. Naïve Bayes)** | 81.79 | 79.65% | 94.01% | 86.22% | 0.872 | 0.599 |
| **MLP (Training Cycles = 10)** |  |  |  |  |  |  |
| **SVM** | 89.22% | 89.40% | 93.29% | 91.29% | 0.953 | 0.772 |

**Table 8 – Feature Selection using Forward Search**

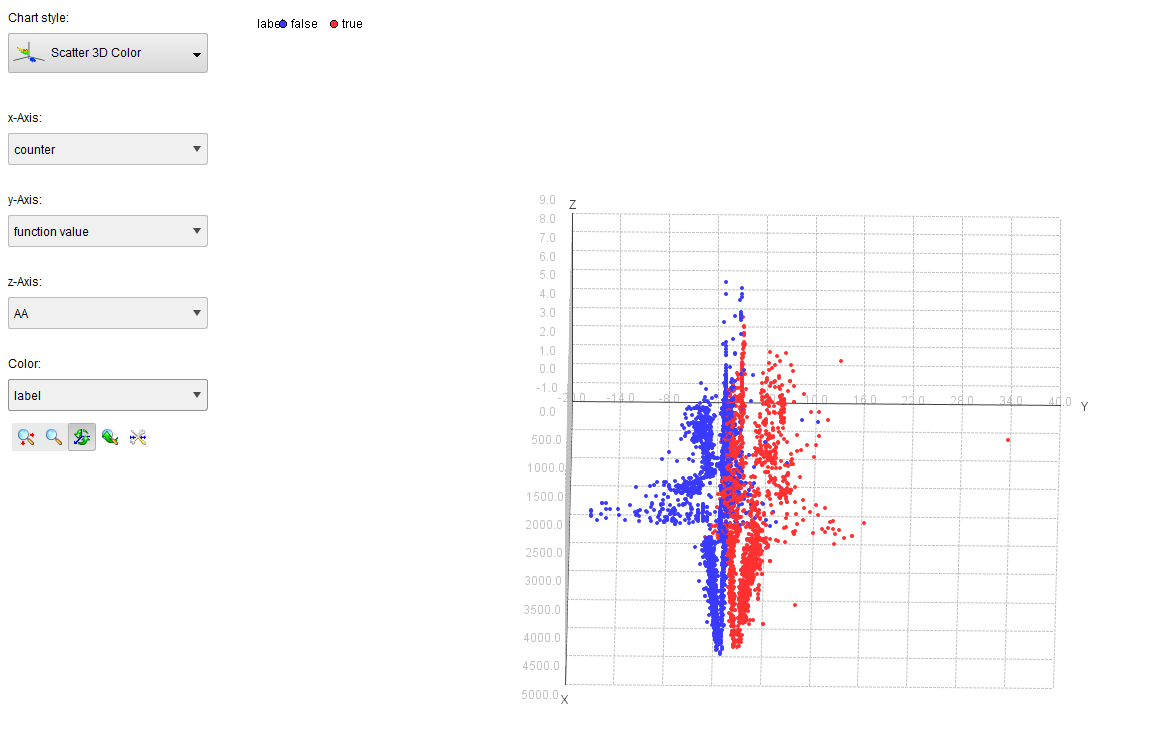
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Backward** |  |  |  |  |  |  |
|  | **Accuracy** | **Precision** | **Recall** | **F-Measure** | **AUC** | **Kappa** |
| **kNN (k=1)** | 91.09 | 92.66% | 92.65% | 92.64% | 0.5 | 0.813 |
| **kNN (k=3)** | 91.33 | 91.91% | 93.97% | 92.92% | 0.919 | 0.817 |
| **kNN (k=5)** | 90.87 | 91.64% | 93.47% | 92.54% | 0.935 | 0.808 |
| **kNN (k=7)** | 91.37 | 91.98% | 93.97% | 92.95% | 0.955 | 0.818 |
| **kNN (k=9)** | 90.85 | 91.31% | 93.87% | 92.56% | 0.958 | 0.807 |
| **Decision Trees** | 90.55 | 90.14% | 94.84% | 92.40% | 0.907 | 0.799 |
| **Bayesian Network (e.g. Naïve Bayes)** | 82.81 | 95.94% | 74.82% | 84.05% | 0.936 | 0.66 |
| **MLP (Training Cycles = 10)** |  |  |  |  |  |  |
| **SVM** | 91.28% | 91.03% | 94.98% | 92.96% | 0.815 | 0.959 |

**Table 9 – Feature Selection using Backward Elimination**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **PCA** |  |  |  |  |  |  |
|  | **Accuracy** | **Precision** | **Recall** | **F-Measure** | **AUC** | **Kappa** |
| **kNN (k=1)** | 72.09 | 86.03% | 64.42% | 73.66% | 0.5 | 0.453 |
| **kNN (k=3)** | 71.85 | 85.14% | 64.89% | 73.63% | 0.756 | 0.446 |
| **kNN (k=5)** | 72.03 | 84.88% | 65.53% | 73.93% | 0.766 | 0.448 |
| **kNN (k=7)** | 71.72 | 84.37% | 65.46% | 73.69% | 0.771 | 0.442 |
| **kNN (k=9)** | 72.03 | 84.44% | 66.03% | 74.09% | 0.769 | 0.447 |
| **Decision Trees** | 82.03 | 84.21% | 86.62% | 85.37% | 0.853 | 0.621 |
| **Bayesian Network (e.g. Naïve Bayes)** | 62.51 | 97.36% | 39.20% | 55.85% | 0.846 | 0.324 |
| **MLP (Training Cycles = 10)** | 80.72% | 82.72% | 86.26% | 84.42% | 0.875 | 0.592 |
| **SVM** | 68.46% | 77.99% | 56.62% | 65.35% | 0.757 | 0.378 |

**Table 10 - Top 9 features based on 3 Components of PCA**

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. MLP, Decision Trees and Bayesian Network did not give greater results compared to kNN.



**Figure 8 – SVM Results**

Based on the SVM results above, true and false nodes are visible. The color blue indicates that the node is not a spam while the red nodes illustrate a spam instance.

**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

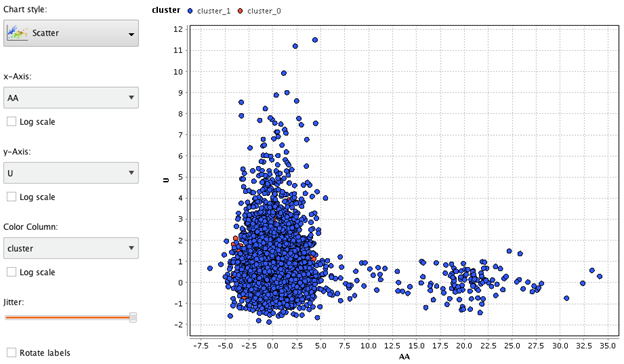


Figure 1: Clustering – k=2 – Scatter Plot

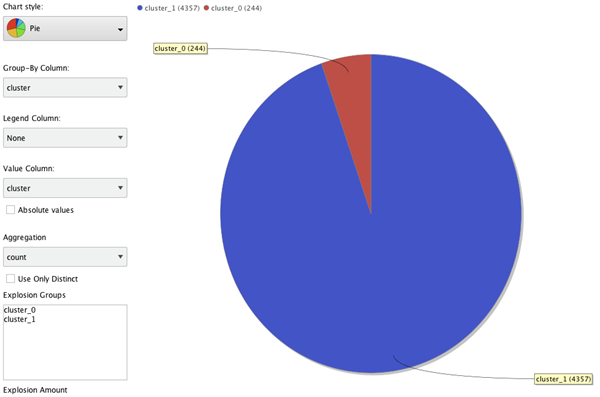


Figure 2: Clustering – k=2 – Pie Chart

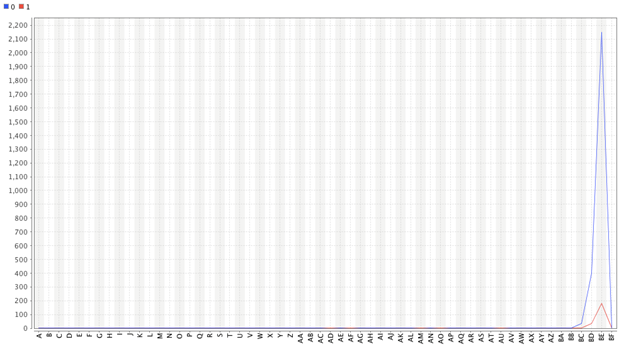


Figure 3: Plot View

Max value for Centroid 0 and 1 is feature BD.

b) K=20

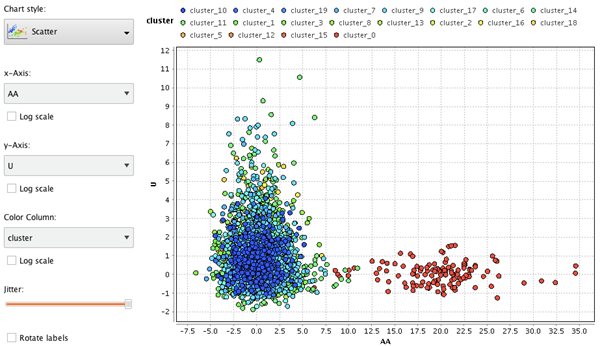


Figure 4: Clustering – k=20 – Scatter Plot

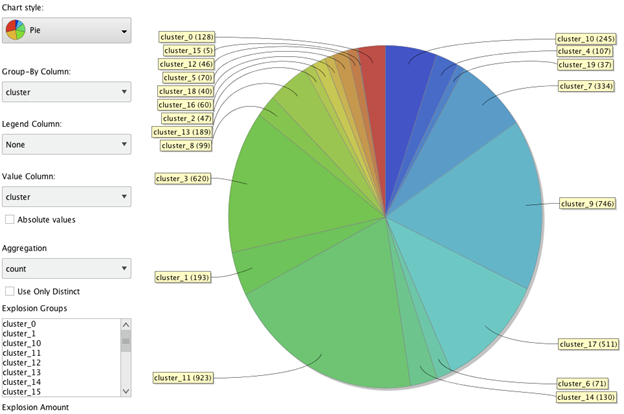


Figure 5: Clustering – k=20 – Pie Chart



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)

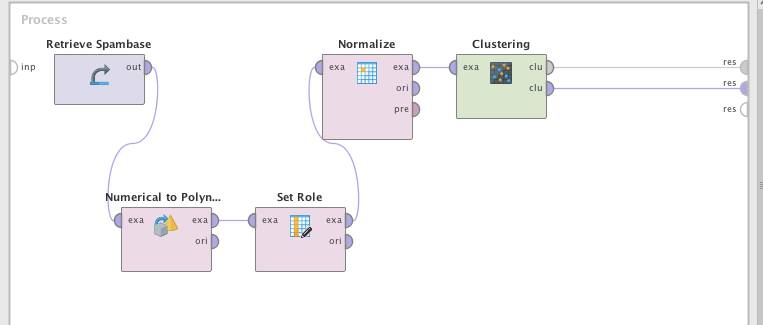
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Cluster** | **Number of samples** | **Cluster** | **Number of samples** | **Cluster** | **Number of samples** |
| 0 | 128 | 8 | 99 | 16 | 60 |
| 1 | 193 | 9 | 746 | 17 | 511 |
| 2 | 47 | 10 | 245 | 18 | 40 |
| 3 | 620 | 11 | 923 | 19 | 37 |
| 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



**Process 1 -** Clustering

5. 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

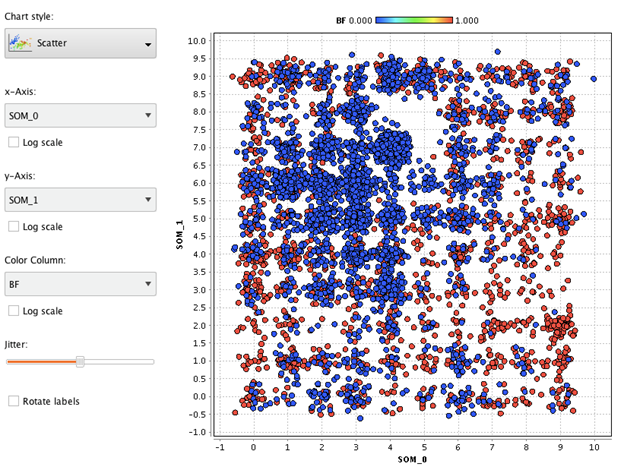


Figure 8: Scatter Plot -1,000 Training Rounds

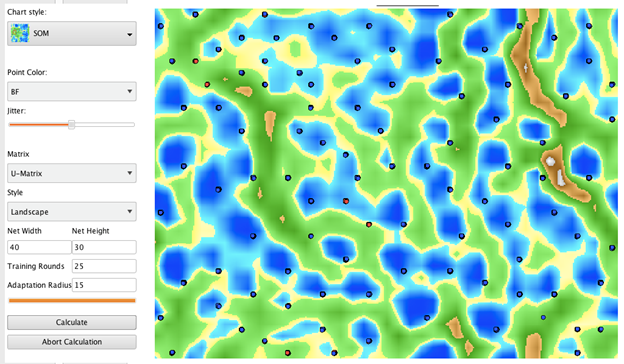


Figure 9: SOM Plot - 1,000 Training Rounds

b. 10,000 Training Rounds

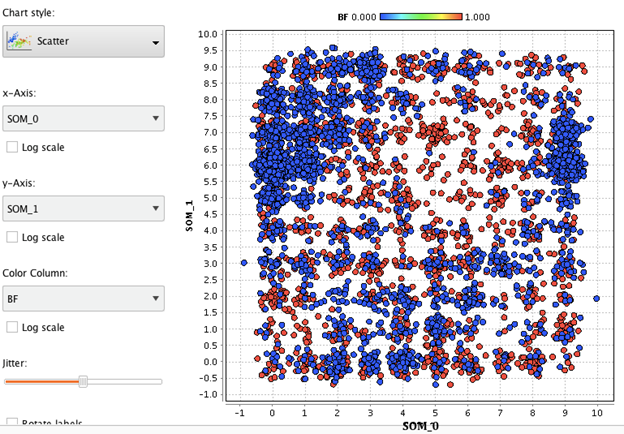


Figure 10: Scatter Plot - 10,000 Training Rounds

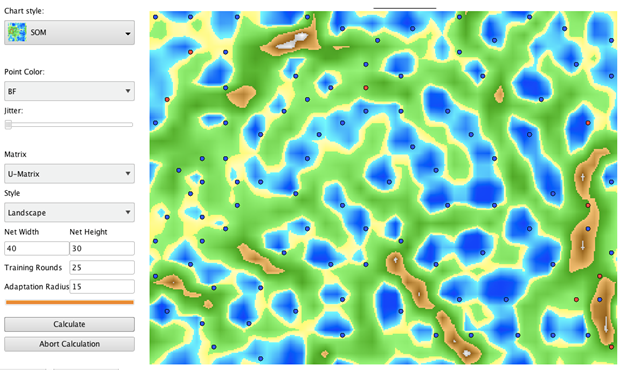


Figure 11: SOM - 10,000 Training Rounds

c. 15,000 Training Rounds

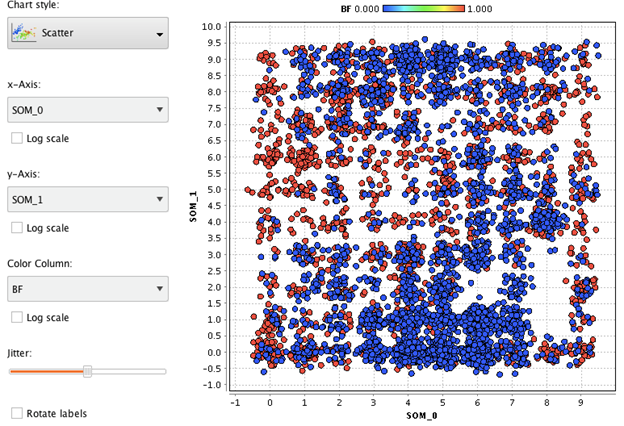


Figure 12: Scatter Plot - 15,000 Training Rounds

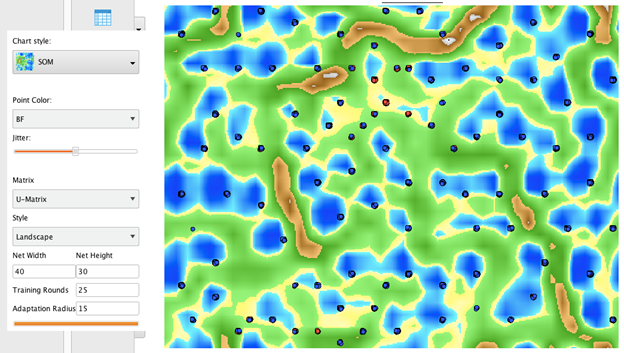
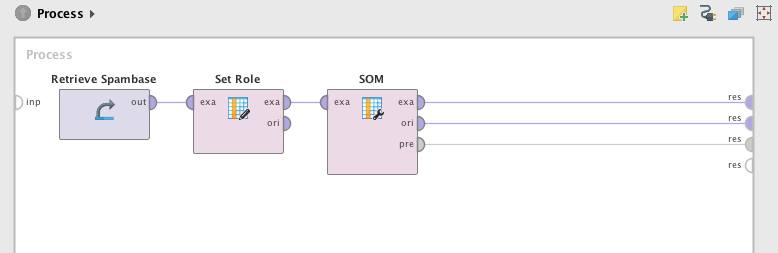


Figure 13: SOM - 15,000 Training Rounds

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



**Process 2 -** SOM

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.

.

**MACHINE LEARNING PROJECT**

**Milestone #1**

**Diabetic Retinopathy**

**A. Dataset Building and Normalization**

1. **Source of Dataset**

* UCI Machine Learning Repository – Center for Machine Learning and Intelligent Systems http://archive.ics.uci.edu/ml/datasets/Diabetic+Retinopathy+Debrecen+Data+Set
* Dr. Balint Antal, Department of Computer Graphics and Image Processing Faculty of Informatics, University of Debrecen, 4010, Debrecen, POB 12, Hungary antal.balint '@' inf.unideb.hu
* Dr. Andras Hajdu, Department of Computer Graphics and Image Processing Faculty of Informatics, University of Debrecen, 4010, Debrecen, POB 12, Hungary hajdu.andras '@' inf.unideb.hu

1. **Description of Dataset**

Number of original features: 20

Sample size: 1151

This dataset contains features extracted from the Messidor image set to predict whether an image contains signs of diabetic retinopathy or not. All features represent either a detected lesion, a descriptive feature of a anatomical part or an image-level descriptor. The underlying method image analysis and feature extraction as well as our classification technique is described in Balint Antal, Andras Hajdu: An ensemble-based system for automatic screening of diabetic retinopathy, Knowledge-Based Systems 60 (April 2014), 20-27.

1. **Name, Description and type of each feature**

A Result of Quality Assessment-----------------------------------------------------------------------------------Real = [0 - 1]

B Result of Pre-screening------------------------------------------------------------------------------------------ Real = [0 - 1]

C-H Results of MA detection----------------------------------------------------------------------------------------- Real = [0 - 1]

I-P Results of MA detection for exudates------------------------------------------------------------------------ Real = [0 - 1]

Q Euclidean distance of the center of the macula and the center of the optic disc----------------- Real = [0 - 1]

R The diameter of the optic disc--------------------------------------------------------------------------------- Real = [0 - 1]

S The binary result of the AM/FM-based classification----------------------------------------------------- Real = [0 - 1]

T Class label--------------------------------------------------------------------------------------------Binomial ⊆ [false, true]

The attributes were set from Numerical to Binomial, from Nominal to Numerical, and then normalized to attain a linear and more robust relationship.

1. **Processed/Normalized dataset**

https://www.dropbox.com/s/ukzgzmqoz1hm9ay/Milestone-1-diabetic-retinopathy-normalized.xlsx?dl=0

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

None

**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** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | 61.25 % | 68.20 % | 63.34 % | 67.16 % |
| **kNN (k=3)** | 62.29 % | 71.42 % | 64.47 % | 64.64 % |
| **kNN (k=5)** | 63.16 % | 72.20 % | 66.20 % | 65.68 % |
| **kNN (k=7)** | 64.73 % | 70.89 % | 68.03 % | 67.33 % |
| **kNN (k=9)** | 63.94 % | 70.89 % | 65.94 % | 68.20 % |
| **Decision Trees** | 53.69 % | 54.39 % | 54.39 % | 53.26 % |
| **Bayesian Network**  **(e.g. Naïve Bayes)** | 60.30 % | 65.68 % | 65.07 % | 59.77 % |
| **MLP**  **(Training Cycles = 10)** | 70.46% | 73.41% | 72.72% | 69.5% |

**Table 1 – Accuracy**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | 0.225 | 0.365 | 0.267 | 0.341 |
| **kNN (k=3)** | 0.247 | 0.429 | 0.291 | 0.292 |
| **kNN (k=5)** | 0.265 | 0.446 | 0.326 | 0.315 |
| **kNN (k=7)** | 0.296 | 0.420 | 0.362 | 0.348 |
| **kNN (k=9)** | 0.281 | 0.422 | 0.321 | 0.366 |
| **Decision Trees** | 0.069 | 0.030 | 0.093 | 0.017 |
| **Bayesian Network**  **(e.g. Naïve Bayes)** | 0.233 | 0.322 | 0.313 | 0.201 |
| **MLP**  **(Training Cycles = 10)** | 0.416 | 0.47 | 0.458 | 0.396 |

**Table 2 – Kappa**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | 0.500 | 0.500 | 0.500 | 0.300 |
| **kNN (k=3)** | 0.660 | 0.736 | 0.675 | 0.684 |
| **kNN (k=5)** | 0.680 | 0.771 | 0.703 | 0.713 |
| **kNN (k=7)** | 0.694 | 0.761 | 0.717 | 0.725 |
| **kNN (k=9)** | 0.696 | 0.767 | 0.708 | 0.731 |
| **Decision Trees** | 0.549 | 0.548 | 0.565 | 0.529 |
| **Bayesian Network**  **(e.g. Naïve Bayes)** | 0.689 | 0.678 | 0.700 | 0.652 |
| **MLP**  **(Training Cycles = 10)** | 0.797 | 0.8 | 0.804 | 0.776 |

**Table 3 – AUC**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | 64.40 % | 71.60 % | 66.67 % | 69.19 % |
| **kNN (k=3)** | 65.83 % | 75.09 % | 68.23 % | 67.59 % |
| **kNN (k=5)** | 66.91 % | 76.89 % | 70.48 % | 69.22 % |
| **kNN (k=7)** | 68.60 % | 75.75 % | 71.97 % | 71.10 % |
| **kNN (k=9)** | 67.88 % | 76.85 % | 70.02 % | 72.15 % |
| **Decision Trees** | 56.27 % | 53.80 % | 58.24 % | 53.54 % |
| **Bayesian Network**  **(e.g. Naïve Bayes)** | 81.56 % | 73.28 % | 74.59 % | 64.45 % |
| **MLP**  **(Training Cycles = 10)** | 79.9% | 80.98% | 78.90 | 79.09% |

**Table 4 – Precision**

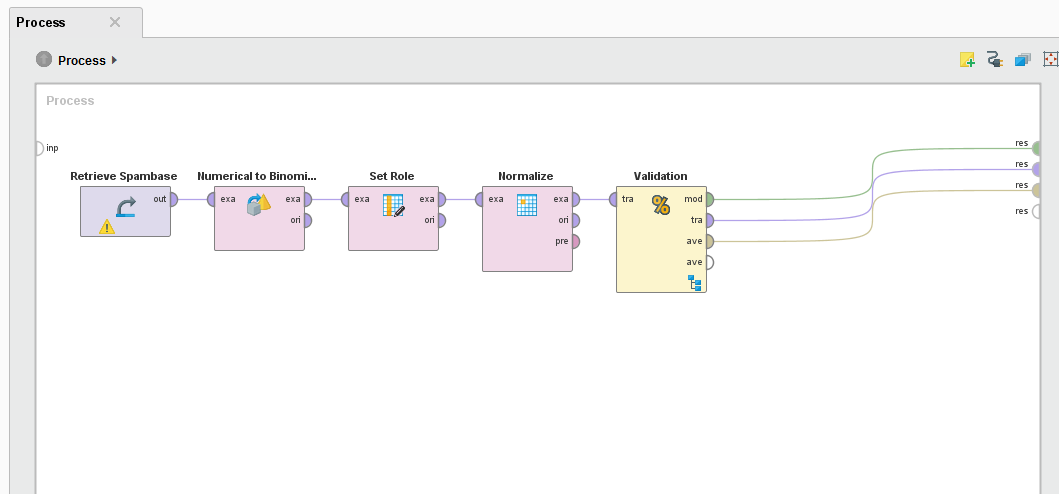
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | 60.39 % | 66.45 % | 61.87 % | 68.74 % |
| **kNN (k=3)** | 60.23 % | 69.07 % | 61.87 % | 64.16 % |
| **kNN (k=5)** | 60.56 % | 68.09 % | 62.52 % | 63.67 % |
| **kNN (k=7)** | 61.87 % | 66.45 % | 65.14 % | 64.81 % |
| **kNN (k=9)** | 60.88 % | 64.65 % | 62.68 % | 65.20 % |
| **Decision Trees** | 57.28 % | 99.67 % | 49.75 % | 90.34 % |
| **Bayesian Network**  **(e.g. Naïve Bayes)** | 32.57 % | 55.65 % | 51.88 % | 54.01 % |
| **MLP**  **(Training Cycles = 10)** | 60.24% | 67.1% | 66.95% | 59.23% |

**Table 5 – Recall**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Complete Feature Set** | **Feature Selection using Forward Search** | **Feature Selection using Backward Search** | **Top 9 features (with higher coefficients) based on 3 Components of Principal Components Analysis (PCA)** |
| **kNN (k=1)** | 62.33 % | 68.93 % | 64.18 % | 68.97 % |
| **kNN (k=3)** | 62.91 % | 71.95 % | 64.89 % | 65.83 % |
| **kNN (k=5)** | 63.57 % | 72.22 % | 66.26 % | 66.33 % |
| **kNN (k=7)** | 65.06 % | 70.79 % | 68.38 % | 67.81 % |
| **kNN (k=9)** | 64.19 % | 70.22 % | 66.15 % | 68.56 % |
| **Decision Trees** | 56.77 % | 69.88 % | 53.66 % | 67.23 % |
| **Bayesian Network**  **(e.g. Naïve Bayes)** | 46.55 % | 63.26 % | 61.20 % | 58.77 % |
| **MLP**  **(Training Cycles = 10)** | 68.21% | 72.66 | 72.14 | 66.66% |

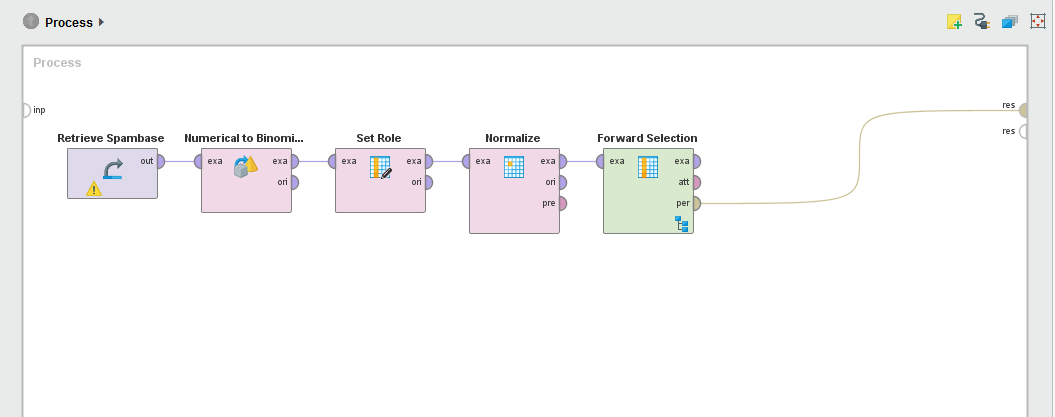
**Table 6 – F-Measure**

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



**Figure 1 –** Complete Feature Set process

* 1. Feature Selection using Forward Search

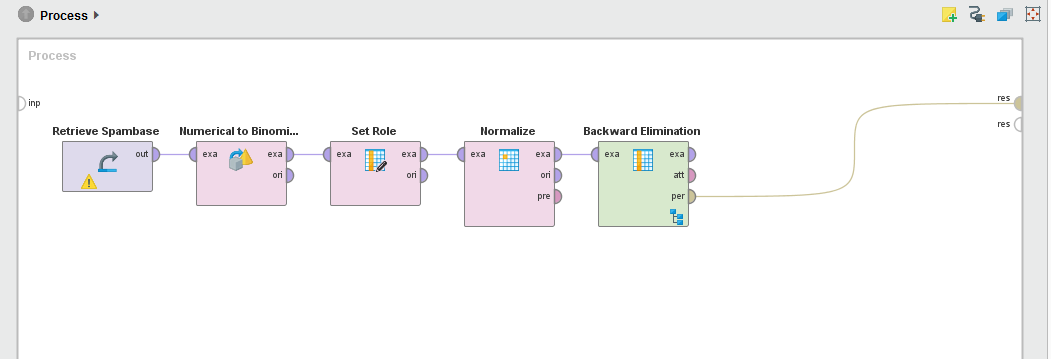


**Figure 2 –** Feature Selection using Forward Search process



**Figure 3 –** Inside Forward Selection

* 1. Feature Selection using Backward Search

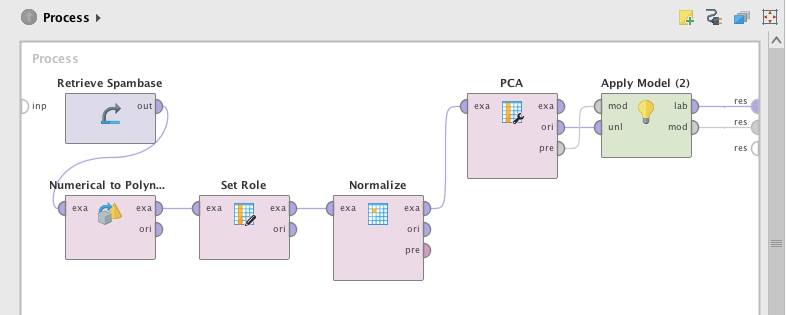


**Figure 4 –** Feature Selection using Backward Search process



**Figure 5 –** Inside Backward Elimination

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

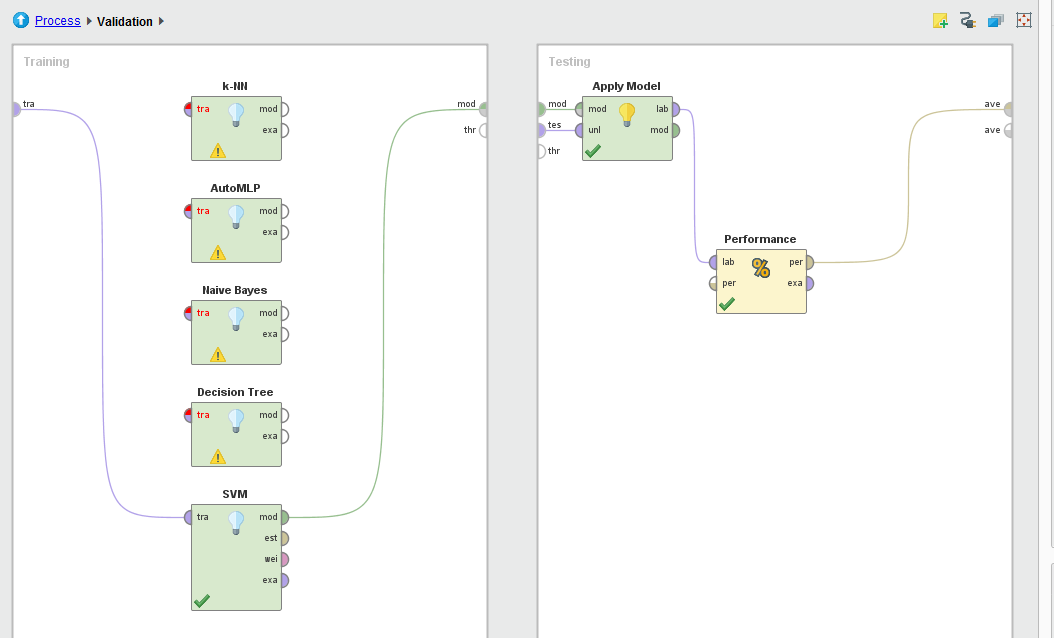


**Figure 6 –** PCA process

The top 9 features are:B, C, D, E, F, G, H, I, and S. These were done by getting the top 9 features based on the highest values from PC1, PC2, PC3 (refer to MS1 excel file).

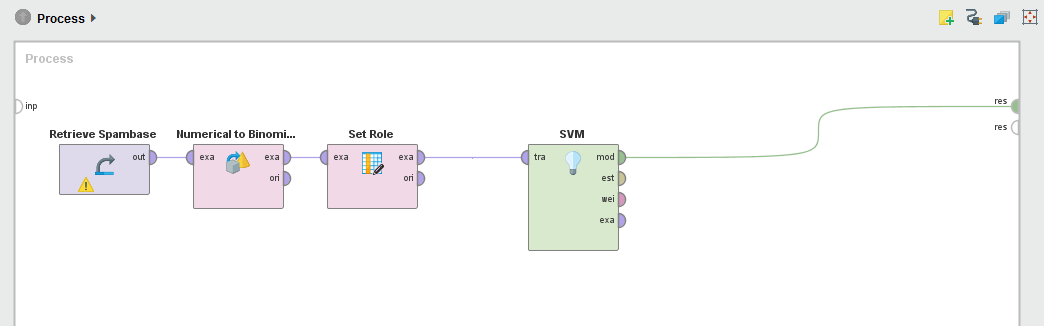
The resulting dataset with only top 9 features was used at the Complete Feature Set process as well. The results of this were placed under this category.

* 1. Validation



**Figure 7 –** Inside Validation

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

****

**Figure 8 – SVM Process**

**C. Discussion of Results**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CFS** |  |  |  |  |  |  |
|  | **Accuracy** | **Precision** | **Recall** | **F-Measure** | **AUC** | **Kappa** |
| **kNN (k=1)** | 61.25% | 64.40% | 60.39% | 62.33% | 0.5 | 0.225 |
| **kNN (k=3)** | 62.29% | 65.83% | 60.23% | 62.91% | 0.66 | 0.247 |
| **kNN (k=5)** | 63.16% | 66.91% | 60.56% | 63.57% | 0.68 | 0.265 |
| **kNN (k=7)** | 64.73% | 68.60% | 61.87% | 65.06% | 0.694 | 0.296 |
| **kNN (k=9)** | 63.94% | 67.88% | 60.88% | 64.19% | 0.696 | 0.281 |
| **Decision Trees** | 53.69% | 56.27% | 57.28% | 56.77% | 0.549 | 0.069 |
| **Bayesian Network (e.g. Naïve Bayes)** | 60.30% | 81.56% | 32.57% | 46.55% | 0.689 | 0.233 |
| **MLP (Training Cycles = 10)** | 70.46% | 79.90% | 60.24% | 68.21% | 0.797 | 0.416 |
| **SVM** |  |  |  |  |  |  |

**Table 7 – Complete Feature Set**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Forward** |  |  |  |  |  |  |
|  | **Accuracy** | **Precision** | **Recall** | **F-Measure** | **AUC** | **Kappa** |
| **kNN (k=1)** | 68.20% | 71.60% | 66.45% | 68.93% | 0.5 | 0.365 |
| **kNN (k=3)** | 71.42% | 75.09% | 69.07% | 71.95% | 0.736 | 0.429 |
| **kNN (k=5)** | 72.20% | 76.89% | 68.09% | 72.22% | 0.771 | 0.446 |
| **kNN (k=7)** | 70.89% | 75.75% | 66.45% | 70.79% | 0.761 | 0.42 |
| **kNN (k=9)** | 70.89% | 76.85% | 64.65% | 70.22% | 0.767 | 0.422 |
| **Decision Trees** | 54.39% | 53.80% | 99.67% | 69.88% | 0.548 | 0.03 |
| **Bayesian Network (e.g. Naïve Bayes)** | 65.68% | 73.28% | 55.65% | 63.26% | 0.678 | 0.322 |
| **MLP (Training Cycles = 10)** | 73.41% | 80.98% | 67.10% | 72.66 | 0.8 | 0.47 |
| **SVM** |  |  |  |  |  |  |

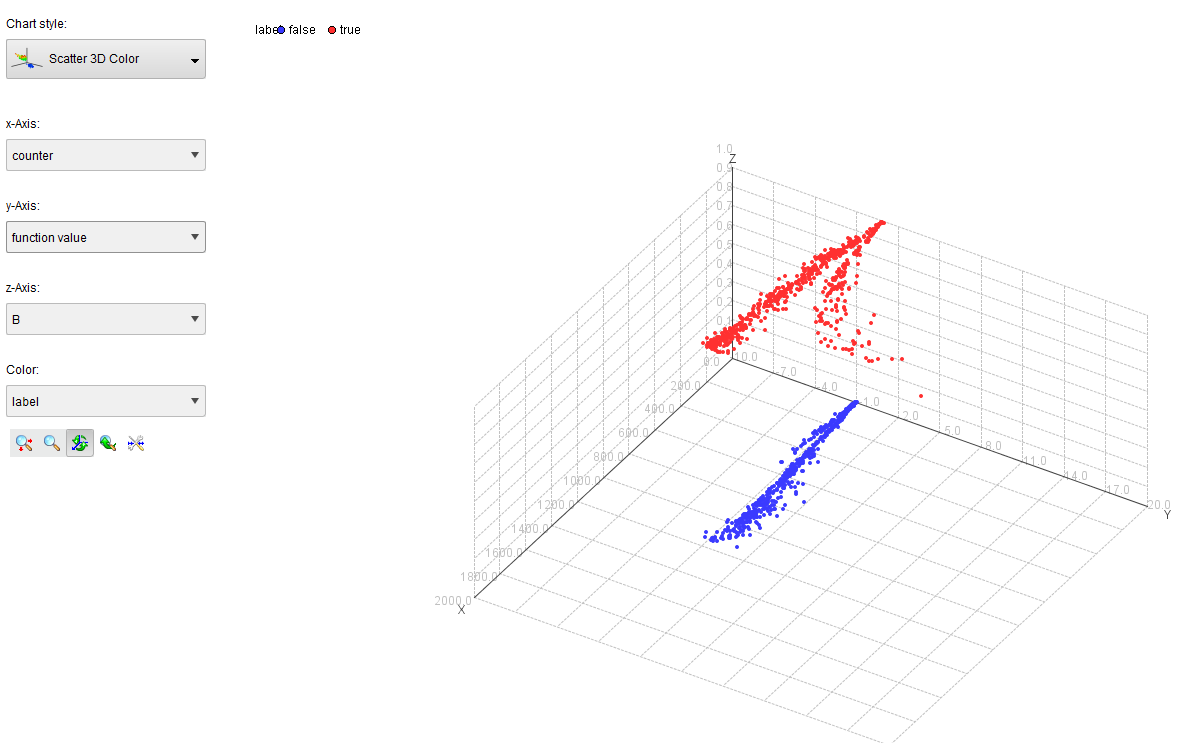
**Table 8 – Feature Selection using Forward Search**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Backward** |  |  |  |  |  |  |
|  | **Accuracy** | **Precision** | **Recall** | **F-Measure** | **AUC** | **Kappa** |
| **kNN (k=1)** | 63.34% | 66.67% | 61.87% | 64.18% | 0.5 | 0.267 |
| **kNN (k=3)** | 64.47% | 68.23% | 61.87% | 64.89% | 0.675 | 0.291 |
| **kNN (k=5)** | 66.20% | 70.48% | 62.52% | 66.26% | 0.703 | 0.326 |
| **kNN (k=7)** | 68.03% | 71.97% | 65.14% | 68.38% | 0.717 | 0.362 |
| **kNN (k=9)** | 65.94% | 70.02% | 62.68% | 66.15% | 0.708 | 0.321 |
| **Decision Trees** | 54.39% | 58.24% | 49.75% | 53.66% | 0.565 | 0.093 |
| **Bayesian Network (e.g. Naïve Bayes)** | 65.07% | 74.59% | 51.88% | 61.20% | 0.7 | 0.313 |
| **MLP (Training Cycles = 10)** | 72.72% | 78.9% | 66.95% | 72.14% | 0.804 | 0.458 |
| **SVM** |  |  |  |  |  |  |

**Table 9 – Feature Selection using Backward Elimination**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **PCA** |  |  |  |  |  |  |
|  | **Accuracy** | **Precision** | **Recall** | **F-Measure** | **AUC** | **Kappa** |
| **kNN (k=1)** | 67.16% | 69.19% | 68.74% | 68.97% | 0.3 | 0.341 |
| **kNN (k=3)** | 64.64% | 67.59% | 64.16% | 65.83% | 0.684 | 0.292 |
| **kNN (k=5)** | 65.68% | 69.22% | 63.67% | 66.33% | 0.713 | 0.315 |
| **kNN (k=7)** | 67.33% | 71.10% | 64.81% | 67.81% | 0.725 | 0.348 |
| **kNN (k=9)** | 68.20% | 72.15% | 65.20% | 68.56% | 0.731 | 0.366 |
| **Decision Trees** | 53.26% | 53.54% | 90.34% | 67.23% | 0.529 | 0.017 |
| **Bayesian Network (e.g. Naïve Bayes)** | 59.77% | 64.45% | 54.01% | 58.77% | 0.652 | 0.201 |
| **MLP (Training Cycles = 10)** | 69.50% | 79.09% | 59.23% | 66.66% | 0.776 | 0.396 |
| **SVM** |  |  |  |  |  |  |

**Table 10 – Top 9 features based on 3 Components of PCA**

****

**Figure 8 – SVM Results**

Based on the SVM results above, true and false nodes are visible. The color blue indicates that the node is negative of diabetic retinopathy while the red nodes illustrates otherwise.

Based on the results, the following characteristics were observed:

* In the Complete Feature Set, MLP was the best classifier.
* In the Feature Selection Forward Search, MLP was the best classifier.
* In the Feature Selection Backward Search, MLP was the best classifier.
* In PCA, MLP was the best classifier.
* This shows that overall, MLP was the best classifier.

**D. Problems Encountered**

SVM were not finished due to lack of time. The processes were not able to finish on time.

**Milestone 2**

**Diabetic Retinopathy**

**A. k-Means Clustering**

**a.) K=2**

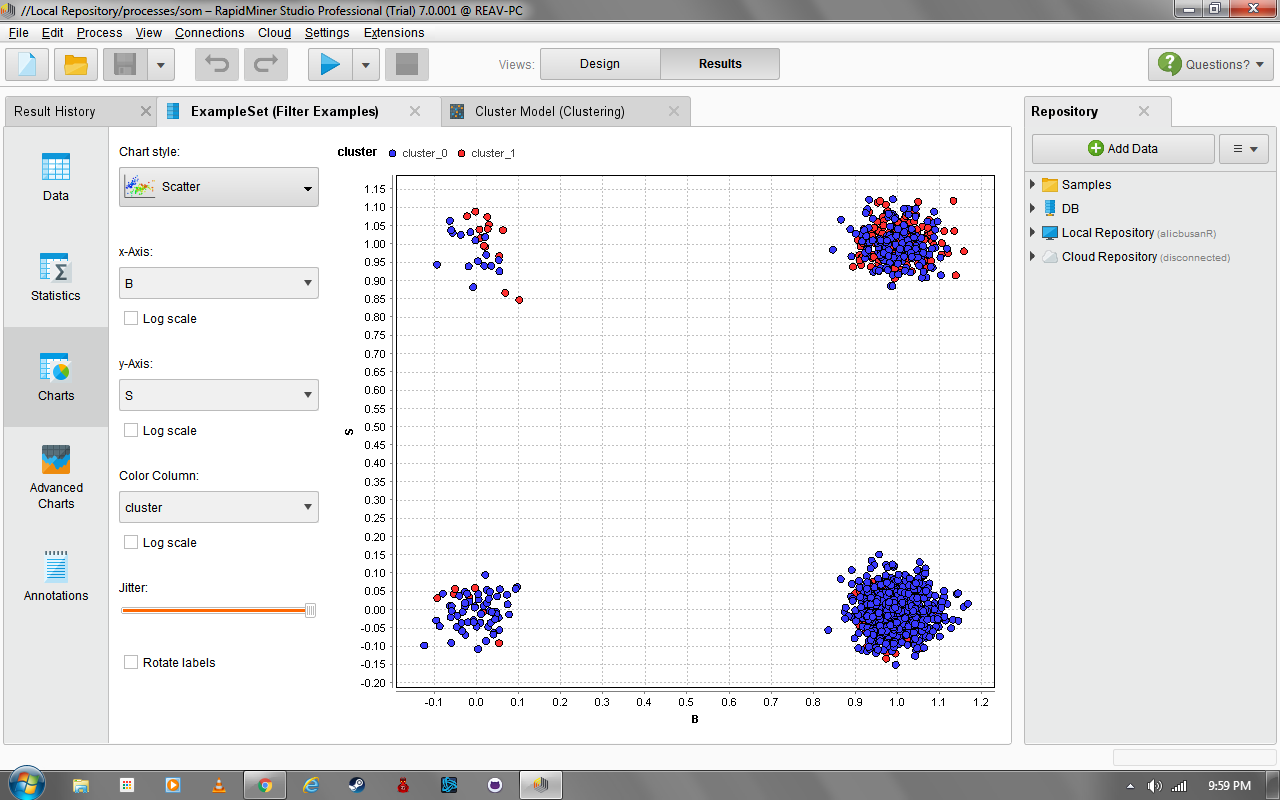


Figure 14: Cluster Plot (x=B,y=S)

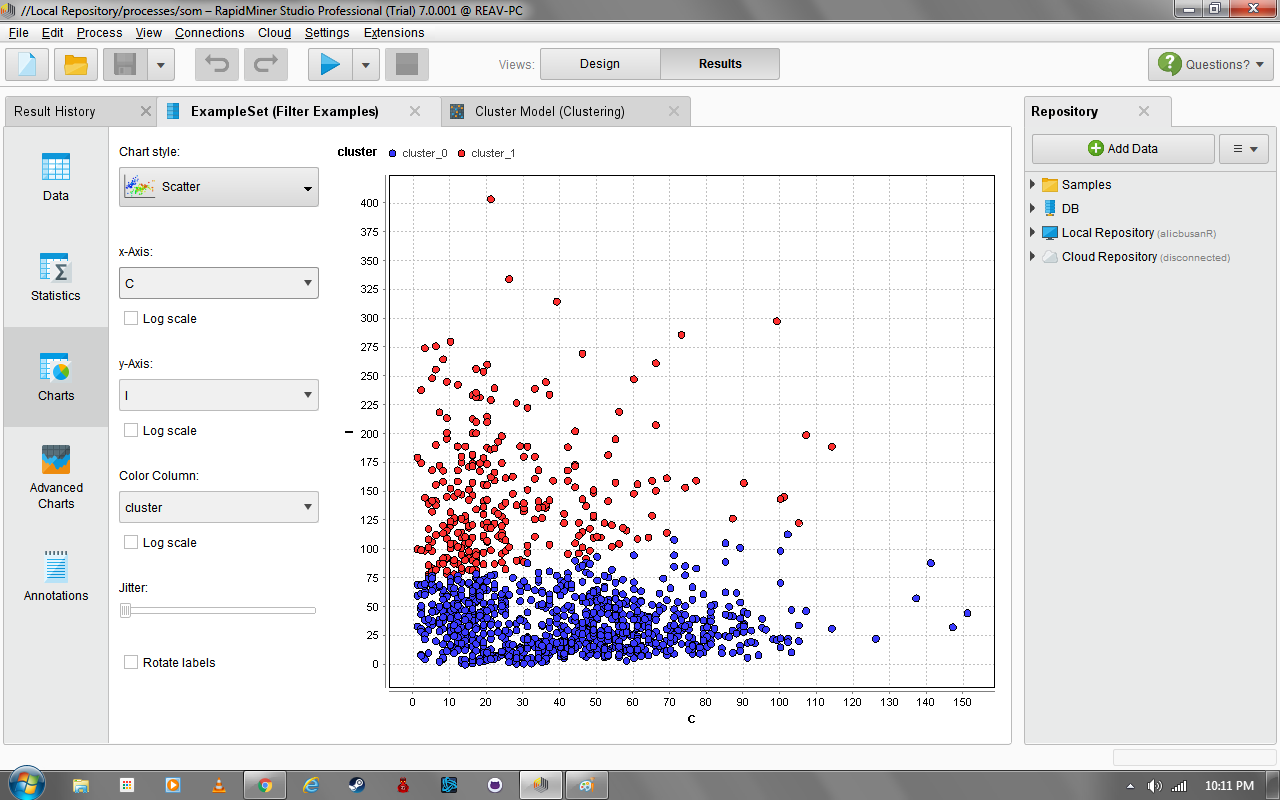


Figure 15: Cluster Plot (x=C, y=I)

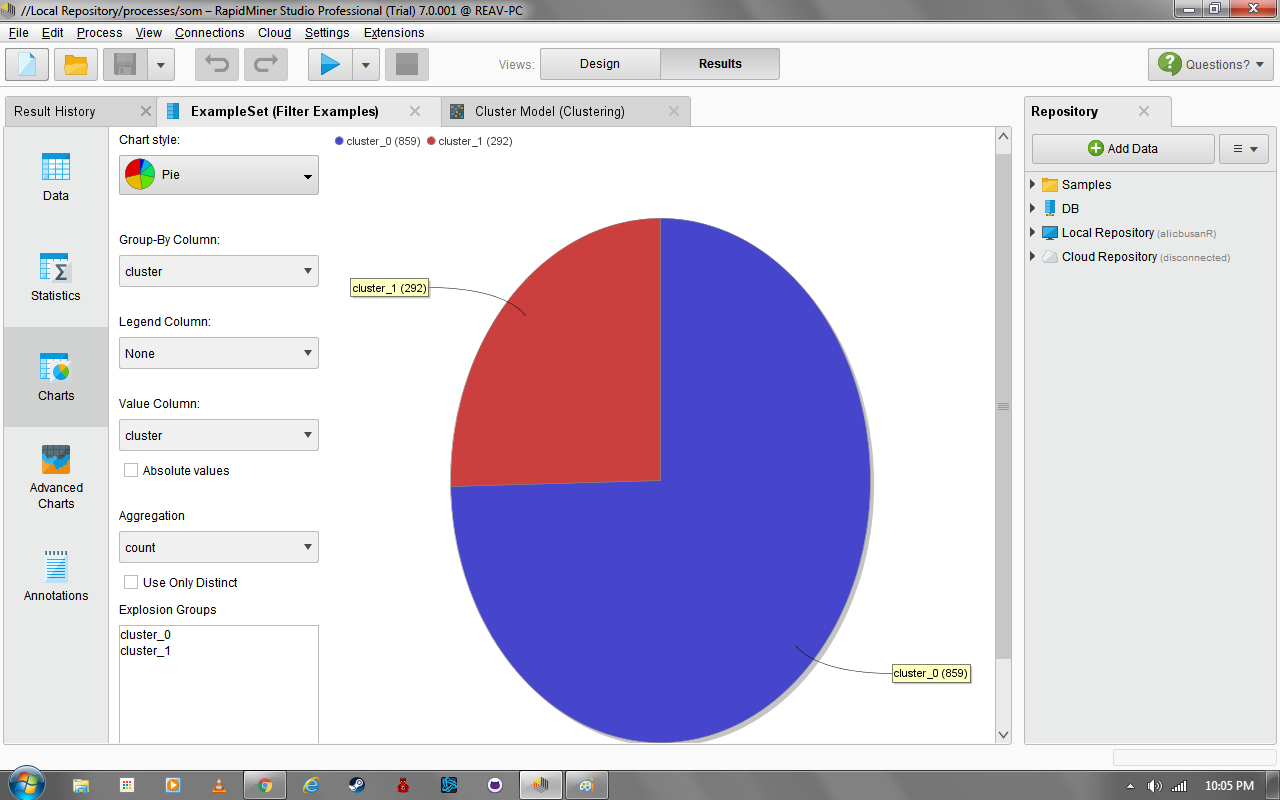


Figure 16: Pie Chart



Figure 17: Plot View

1. Clusters

a. Based on Figure 17, cluster 1 has the feature I as the most dominant feature in the cluster. As for cluster 0, it has feature C.

b. The clusters are based on the features I and C. Clusters 0 and 1 are a mix of both output classes/labels (diabetes or no diabetes).

2. Number of Samples per Cluster

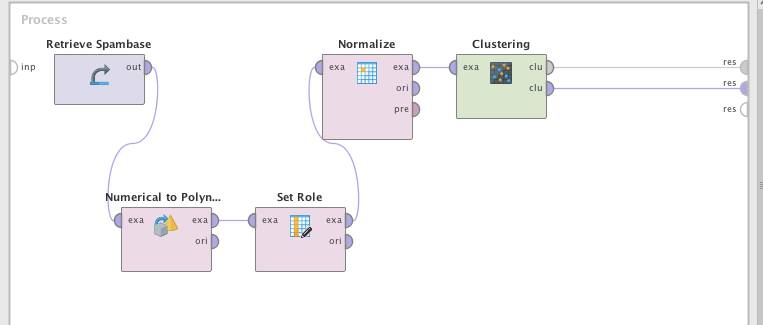
2 clusters (cluster\_0 - cluster 1)

|  |  |
| --- | --- |
| Cluster | Number of samples |
| 0 | 292 |
| 1 | 859 |

3. Cluster Explanation

There are only 2 possible values for the class, which are 1 and 0. These are Boolean values to indicate if a patient has signs of diabetic retinopathy (disease of the retina that results in impairment or loss of vision) or not. This results to using clustering with k = 2. Since the clusters are distinguishable from one another, this is a logical reason to choose k as 2.

4. Process



**Process 1 -** Clustering

5. 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 results, k-means clustering successfully grouped the samples into two clusters. However, the centroids of these 2 clusters are similar that at some point some samples cannot be differentiated from each other.

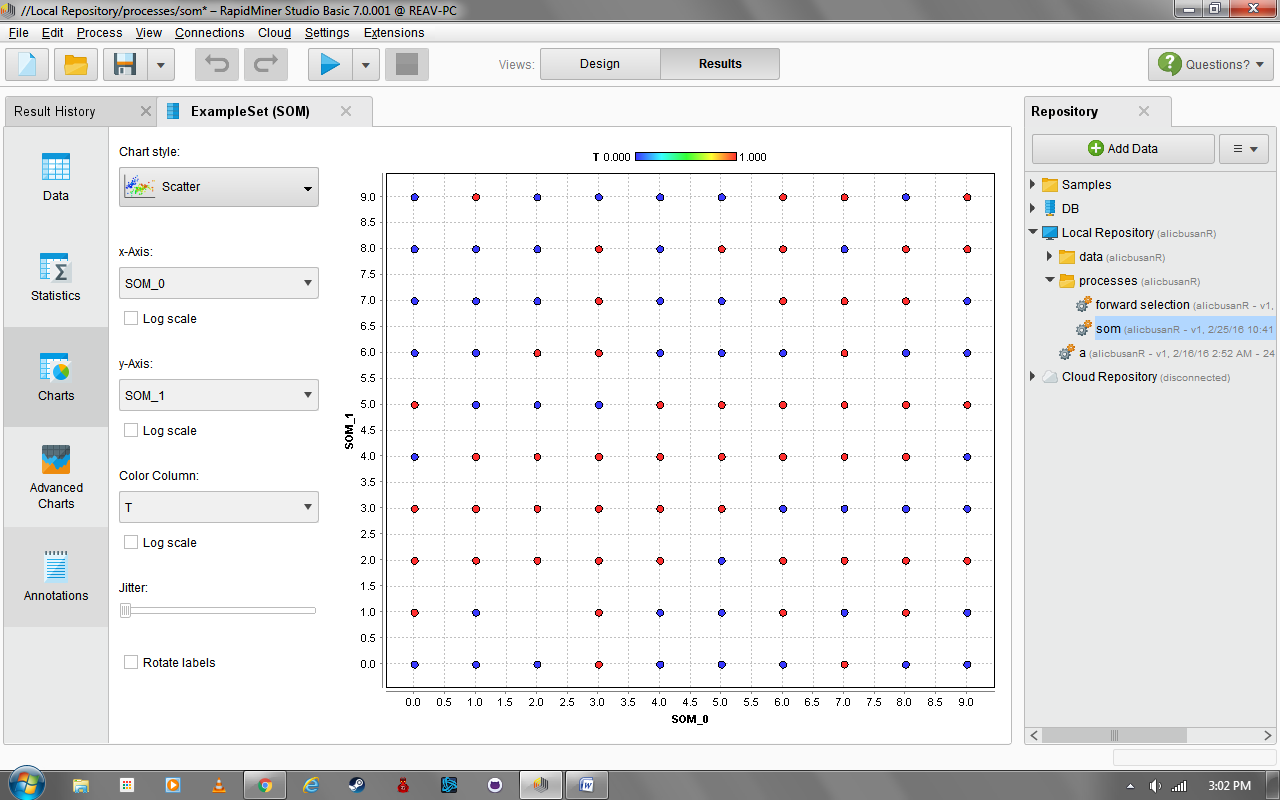
**B. SOM**

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.

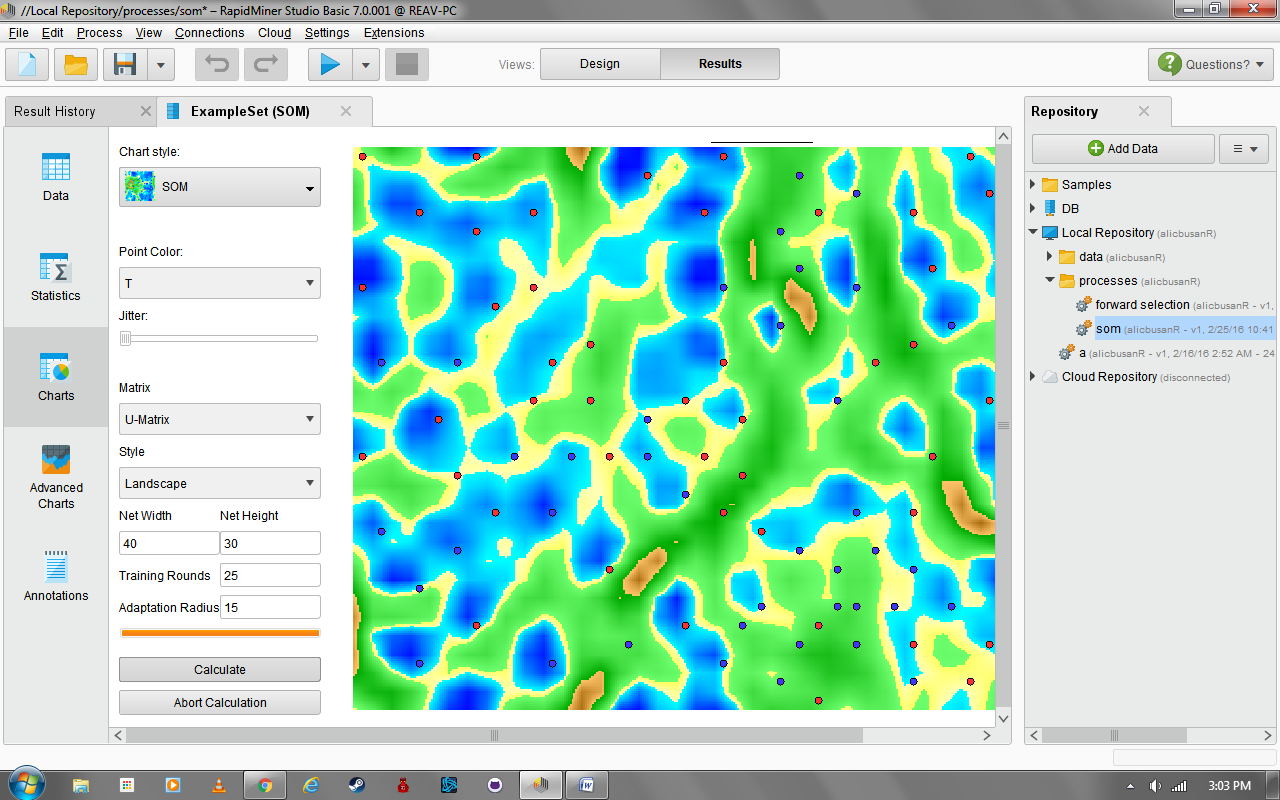
1. 1,000 Training Rounds c. 15,000 Training Rounds
2. 10,000 Training Rounds

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

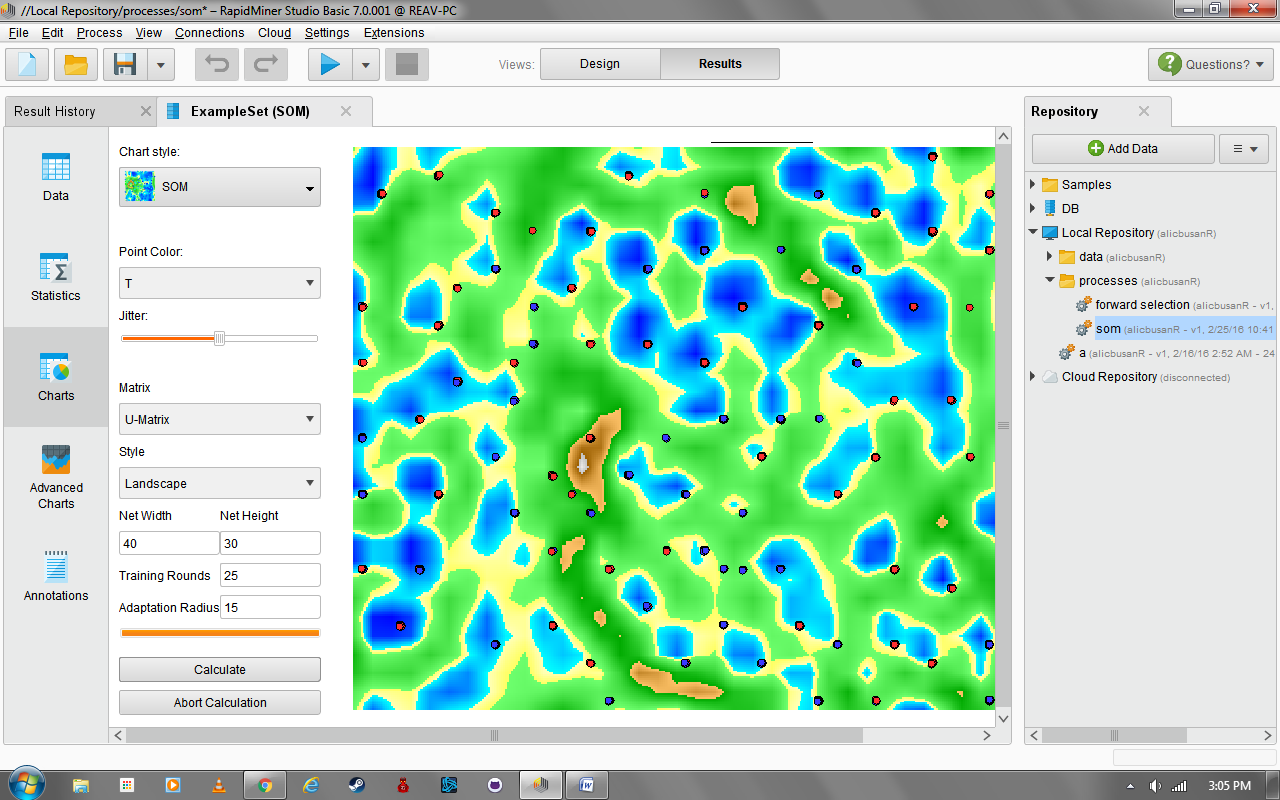
1. 1,000 Training Rounds



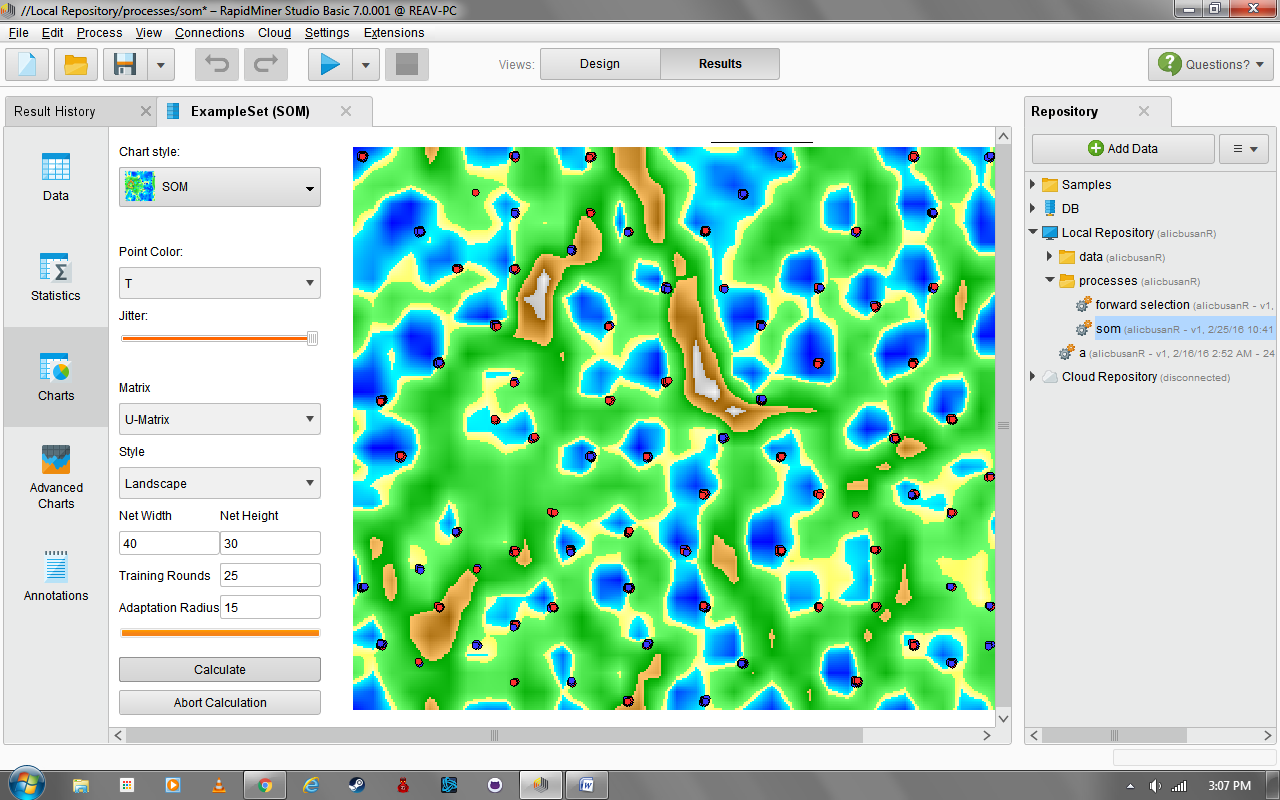
**Figure 18: Scatter Plot - 1,000 Training Rounds - Minimum Jitter**



**Figure 19: SOM - 1,000 Training Rounds - Minimum Jitter**

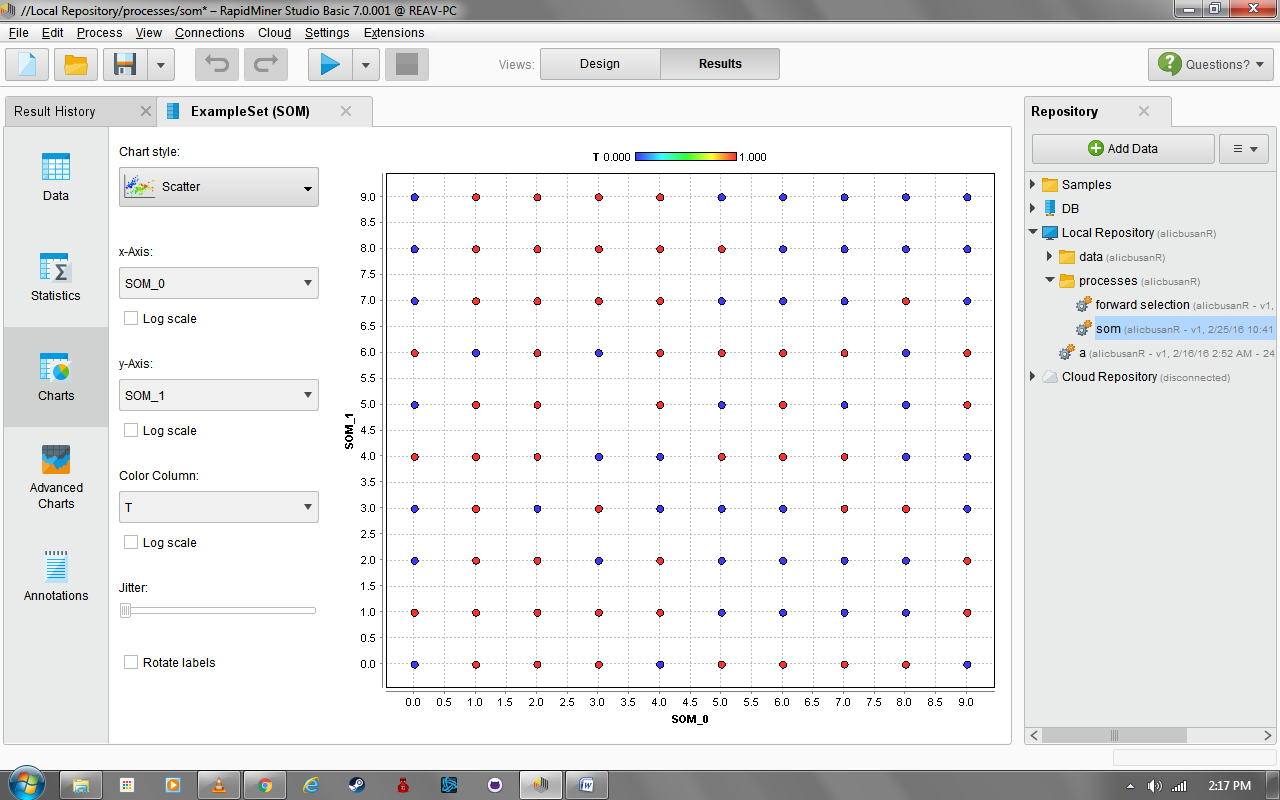


**Figure 20: SOM - 1,000 Training Rounds - Medium Jitter**

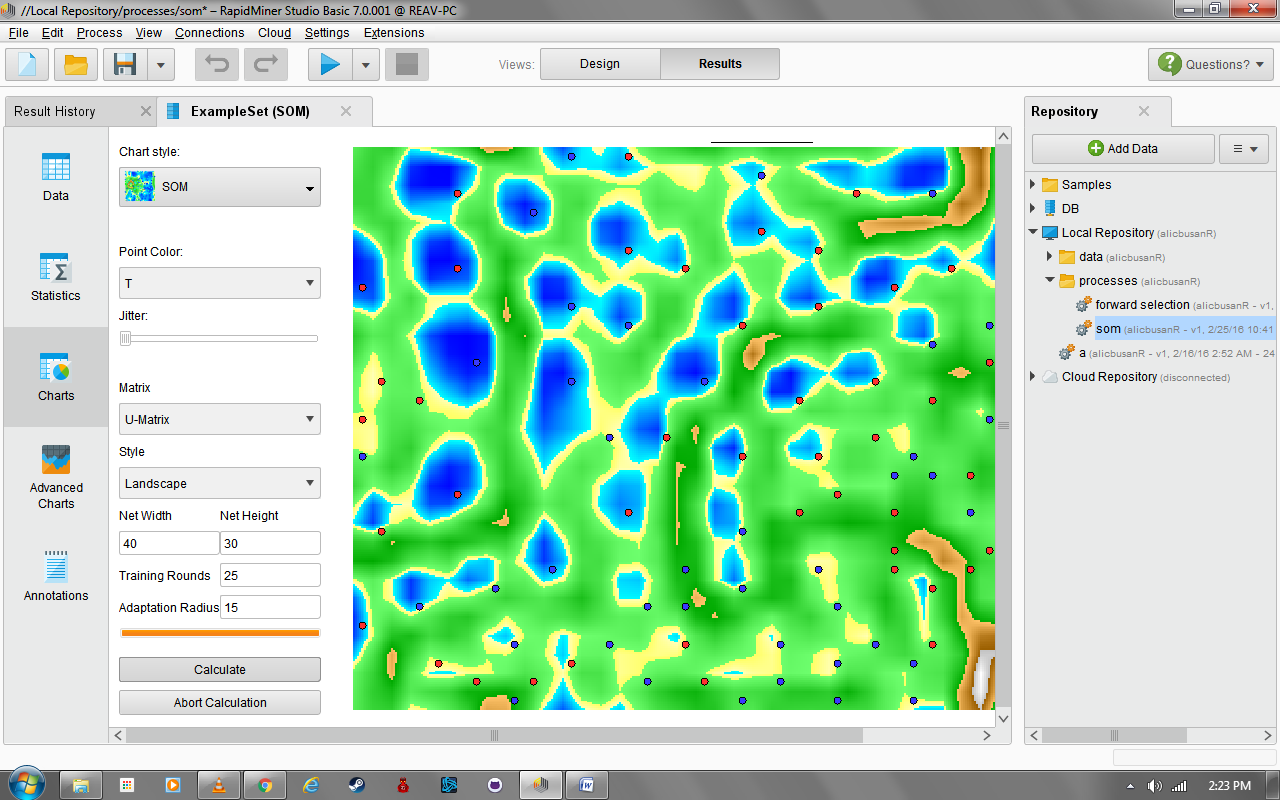


**Figure 21: SOM - 1,000 Training Rounds - Maximum Jitter**

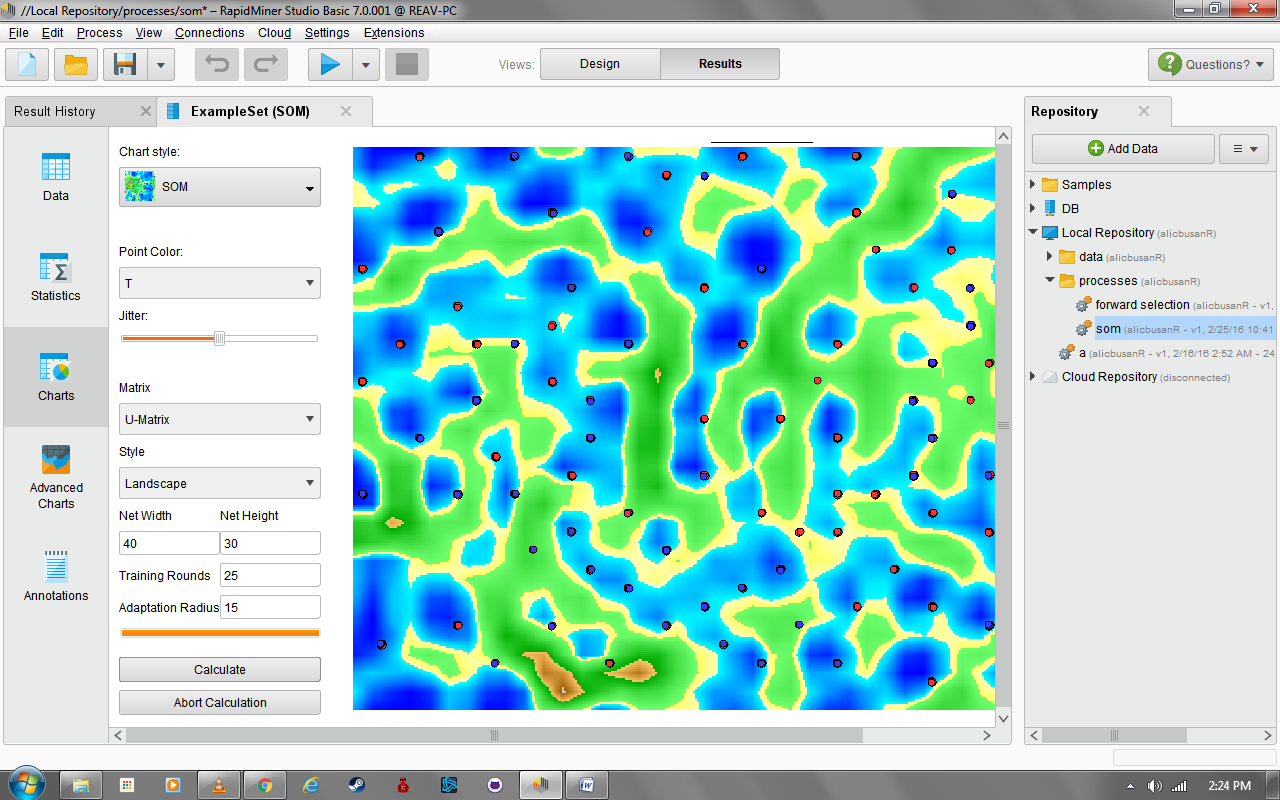
1. 10,000 Training Rounds



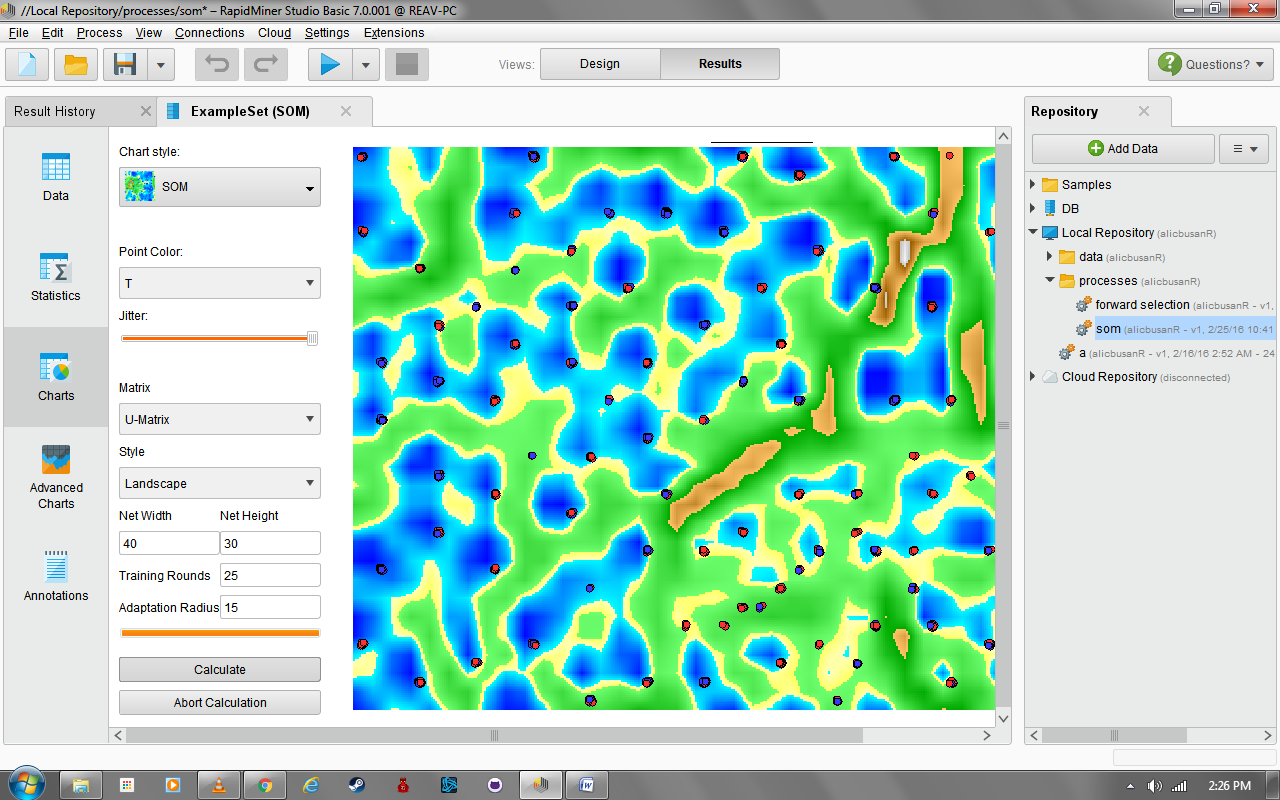
**Figure 22: Scatter Plot - 10,000 Training Rounds - Minimum Jitter**



**Figure 23: SOM - 10,000 Training Rounds - Minimum Jitter**

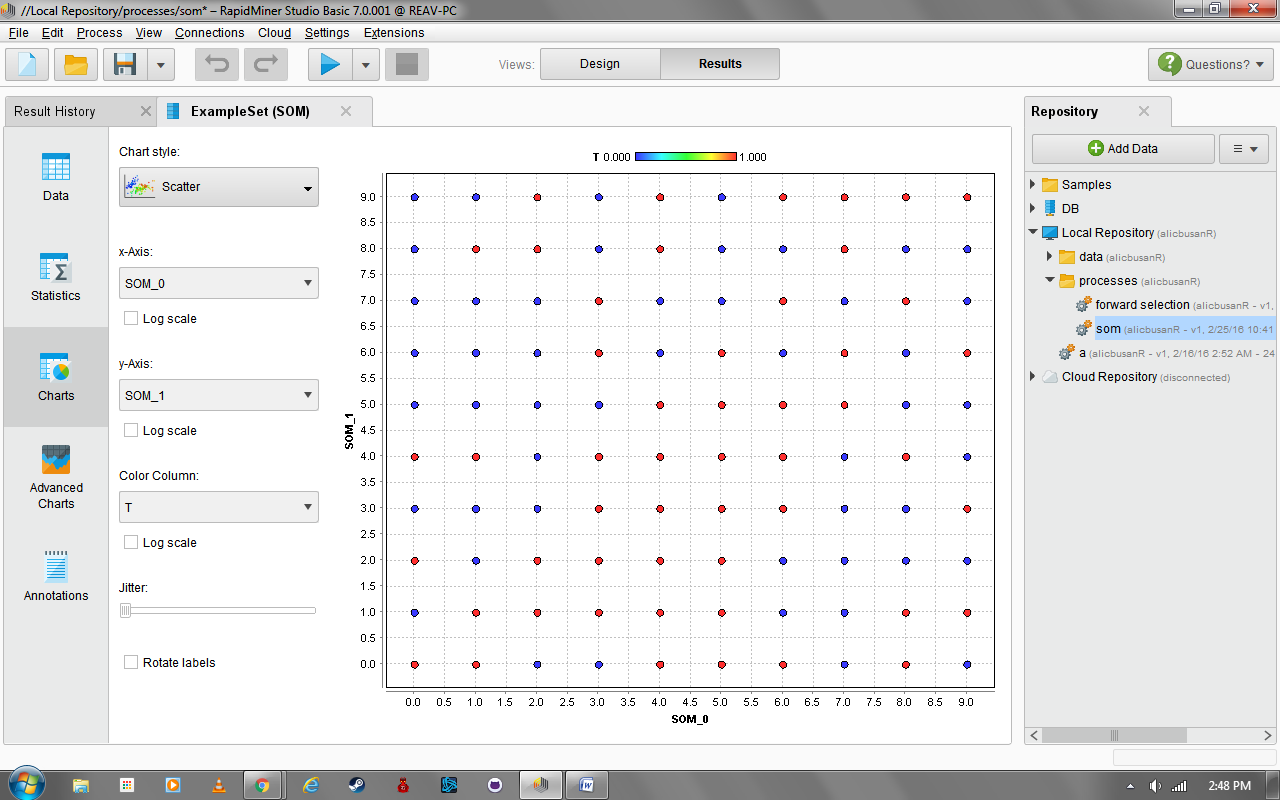


**Figure 24: SOM - 10,000 Training Rounds - Medium Jitter**

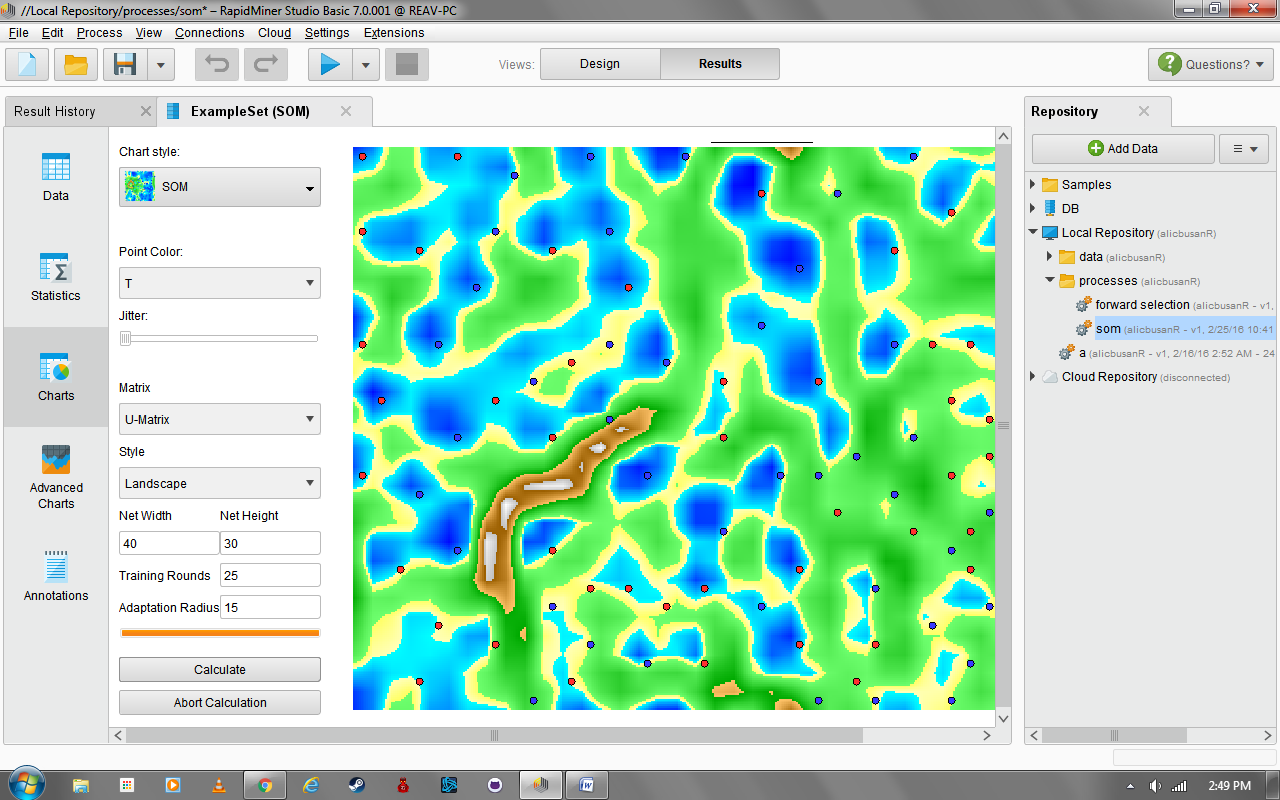


**Figure 25: SOM - 10,000 Training Rounds - Maximum Jitter**

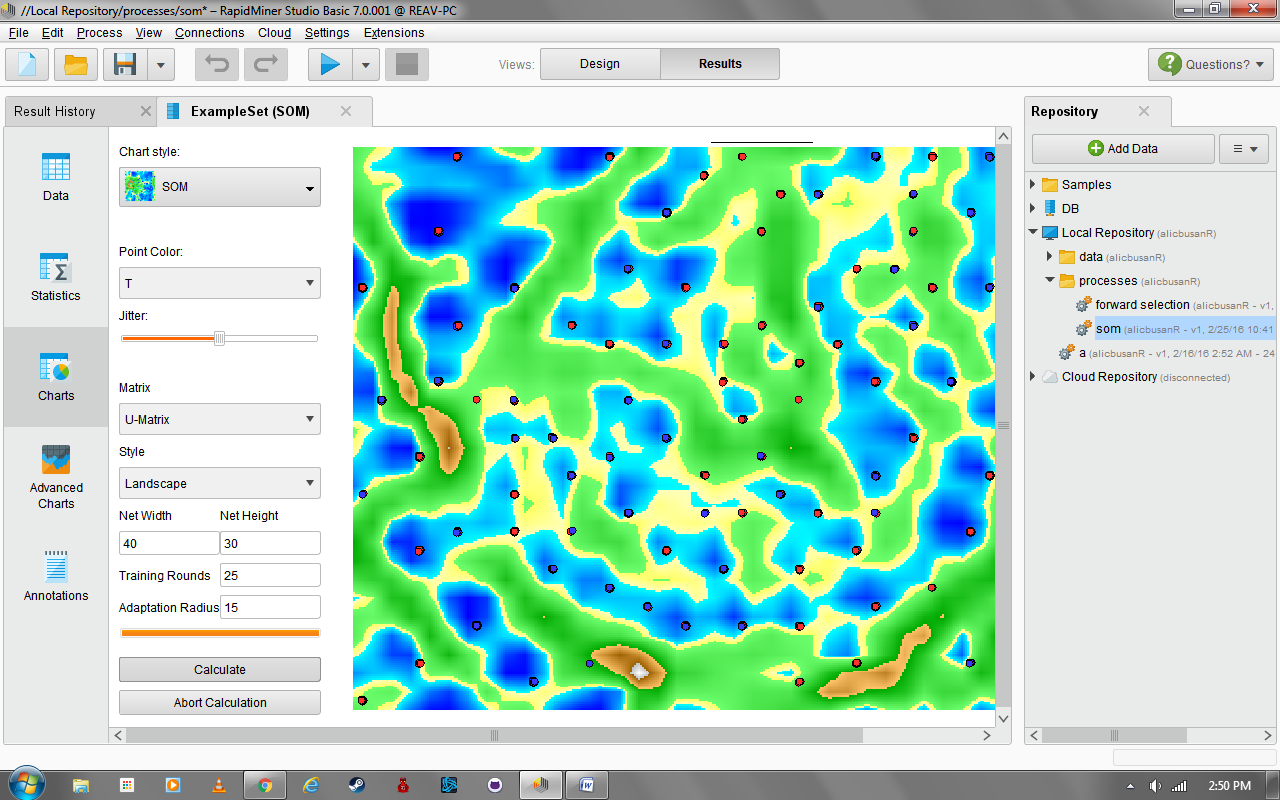
1. 15,000 Training Rounds



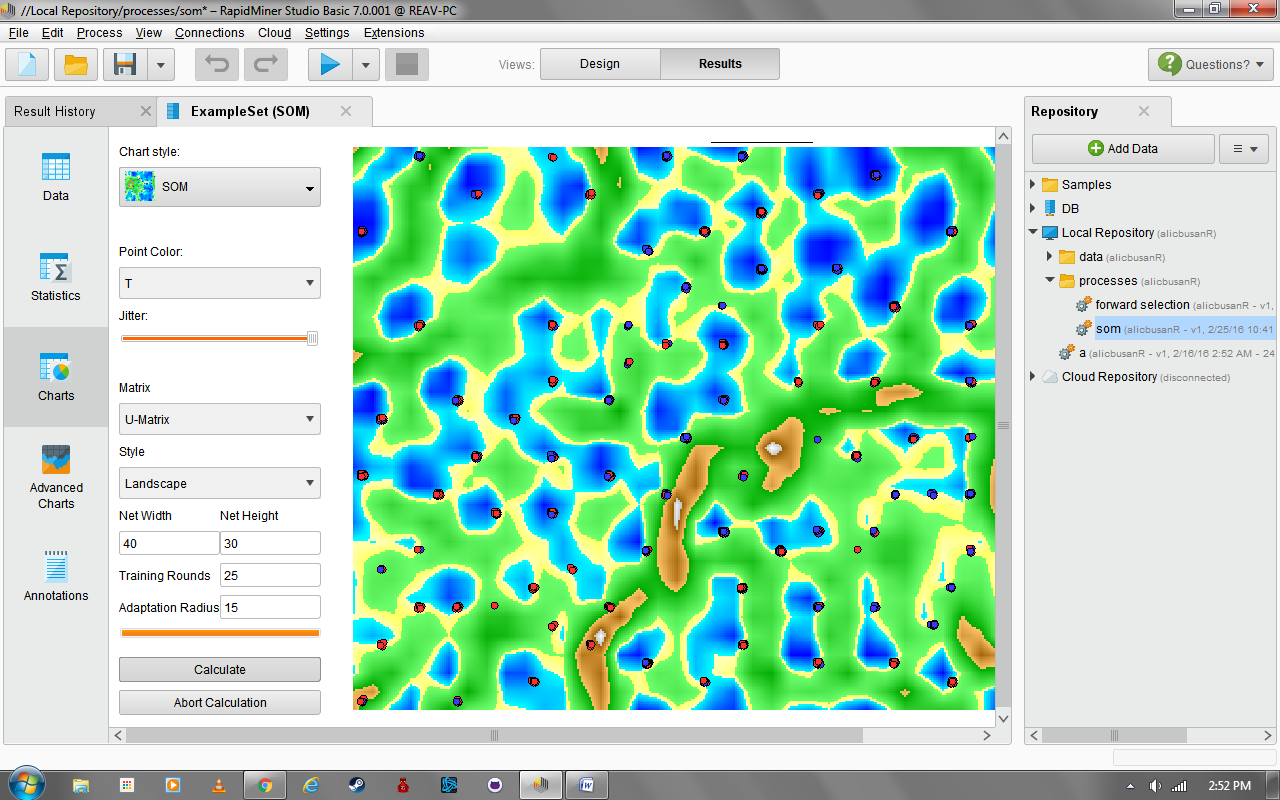
**Figure 26: Scatter Plot - 15,000 Training Rounds - Minimum Jitter**



**Figure 27: SOM - 15,000 Training Rounds - Minimum Jitter**

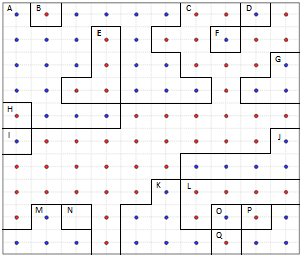


**Figure 28: SOM - 15,000 Training Rounds - Medium Jitter**

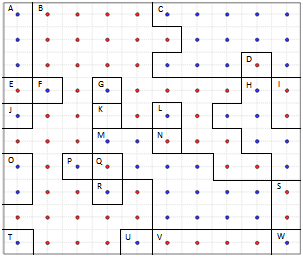


**Figure 29: SOM - 15,000 Training Rounds - Maximum Jitter**

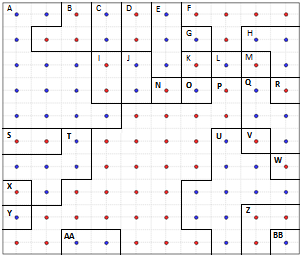
The scatter plot blocks (figures 30-32) were bordered to simplify the image and emphasize the nodes. Every block grouped together with the same color was marked with letters to easily identify each node from the rest.



**Figure 30: Simplified Scatter Plot - 1,000 Training Rounds Simplified**



**Figure 31: Simplified Scatter Plot - 10,000 Training Rounds Simplified**

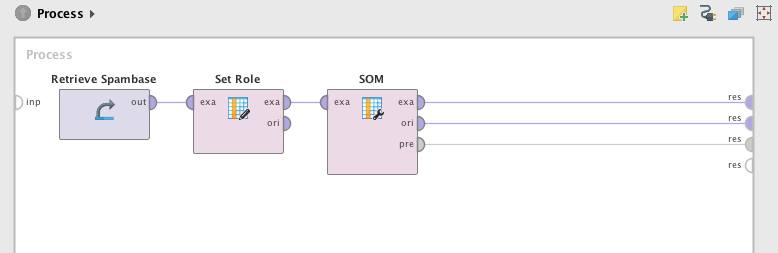


**Figure 32: Simplified Scatter Plot - 15,000 Training Rounds Simplified**

3. Map Interpretation

Based on the results, taking the SOM map of dataset 2 with 1000, 10000, and 15000 training rounds produced a varying number of nodes. With 1000 training rounds, the results displayed 17 distinct nodes with nodes A, B, E, H, J, K, L, O, P, Q, R that assumed to have a label of ‘0’ () and the rest having a label of ‘1’ (). At 10000, 23 nodes were produced. And finally at 15000 training rounds, 28 nodes were retrieved.

4. Process



**Process 2 -** SOM

5. Summary and Conclusion of the experiments

All of the information here were taken at jitter set to a minimum producing a 10x10 SOM map with clear, specific, and distinct nodes that allows the classification of each sample more precise and accurate. Although the actual relationship between each node were not discussed, this still shows a limited display of the SOM of diabetes.

**Spambase Summary and Conclusions**

**Milestone 1**

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. MLP, Decision Trees and Bayesian Network did not give greater results compared to kNN.

**Clustering**

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. 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.

**SOM**

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. 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.

**Diabetes Summary and Conclusions**

**Milestone 1**

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 = 7) is found to be the best classifier. Feature Selection using Backward Search has greater performance and more consistent values compared to other processes. However, using Forward Selection and PCA, though k = 7 did not perform the best, overall, k = 7 sure came close having greater performance values.

Based on the results, the following characteristics were observed:

1. In the Complete Feature Set, kNN (k=7) was the best classifier.
2. In the Feature Selection Forward Search, kNN (k=5) was the best classifier.
3. In the Feature Selection Backward Search, kNN (k=7) was the best classifier.
4. In PCA, kNN (k=9) was the best classifier.

This shows that overall, kNN (k=7) was the best classifier.

**Clustering**

There are only 2 possible values for the class, which are 1 and 0. These are boolean values to indicate of a patient has diabetes or not. This results to making k=2. Since the clusters are distinguishable from one another, this is a good reason to choose k as 2.

The k-means clustering is a process that groups samples together with similar characteristics from other clusters with different qualities. Based on the results, k-means clustering successfully grouped the samples into two clusters. However, the centroids of these 2 clusters are similar that at some point some samples cannot be differentiated from each other.

**SOM**

Based on the results, taking the SOM map of dataset 2 with 1000, 10000, and 15000 training rounds produced a varying number of nodes. With 1000 training rounds, the results displayed 17 distinct nodes. At 10000, 23 nodes were produced. And finally at 15000 training rounds, 28 nodes were retrieved. All of the information here were taken at jitter set to a minimum producing a 10x10 SOM map with clear, specific, and distinct nodes that allows the classification of each sample more precise and accurate. Although the actual relationship between each node were not discussed, this still shows a limited display of the SOM of diabetes.