**MACHINE LEARNING PROJECT**

**DETAILED DISCUSSION OF MILESTONE 1 AND 2**

**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 labeled 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 occurences) / 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 |

**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 |

**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 |

**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 |

**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 |

**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 |

**Table 6 – Kappa Statistic**

**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 |
| **MLP (Training Cycles= 1000)** | **Accuracy** | hi |  |  |  |
| **Precision** |  |  |  |  |
| **Recall** |  |  |  |  |
| **F-Measure** |  |  |  |  |
| **AUC** |  |  |  |  |
| **Kappa** |  |  |  |  |

**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.

**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.