Description:

This experiment is about implementing the Gaussian Naïve Bayes and Logistic regression to classify the data. In this algorithm, training and testing are done on 'Spambase' dataset. The spambase dataset contains 4601 sample example of spam and not spam emails. There is a 58 number of attributes (57 features and 1 class label). The spam-base dataset is divided into training set and test set, each has the same portion of the positive and negative examples in the entire set. Using this train set and test set, the data is classified using Gaussian naïve bayes and logistic regression classification techniques. The results are shown for each experiment below.

Experiment [1]: Classification with naïve bayes

In this experiment, the first, prior probability of each class, 1 (spam) and 0 (not-spam) is calculated in the training dataset. The mean and standard deviation for each of the 57 features is calculated in the training set of the values given each class. To avoid the problem of any standard deviation equal to zero, added the small value epsilon 0.0001 to each standard deviation computed. Using this mean and standard deviation, we calculated the probability density function for each feature to classify the test data. After that, we used the log of the product on all features and the prior probability of that class and predicted the class which has a maximum value.

Results:

Training dataset: class ({0.0: 1400, 1.0: 901})
Prior probability: class ({ 0.0 : 0.60843, 1.0 : 0.39156 })
Precision: 0.684834
Recall: 0.950657
Accuracy: 0.8069565
Confusion Matrix: [[989 399]
[45 867]]

Here, we are assuming that all features are independent of all others. From the confusion matrix, we can clearly observe that Type-2 (False Negative samples) error is more than the Type-1(False Positive samples) error. Due to this, we got higher Recall and smaller precision value. The accuracy of the test set is 80.69% using naïve Bayes classification, where we got the maximum accuracy of 92.43% for the number of feature =45 with feature selection technique in SVM. Thus, SVM based classification is much better than naïve bayes classification on this dataset. This is because of attributes, here the attributes are not independent of each other. Naïve bayes does not do well on this problem in spite of the independence assumption of attributes. Naïve bayes perform better when attributes are independent and class overlapping is smaller. Thus, for this dataset, if attributes are independent of each other then naïve bayes might have performed well for spambase dataset.

Experiment [2]: Classification with logistic regression

In this experiment, after reading the dataset from the file, the data is divided into attributes and labels. The model_selection library from scikit-learn library contains the test_train_split method to divide the dataset into training and testing data. The training and testing of spambase dataset are done using scikit-learn package linear-model. LogistRegression class is used to classification. In

this model, the probabilities describing the possible outcomes of a single trial are modeled using a logistic function. This implementation can fit binary, One-vs- Rest, or multinomial logistic regression with optional L2 or L1 regularization.

The parameter used for this experiment:

Default parameters:

```
class sklearn.linear_model. LogisticRegression (penalty='l2', dual=False, tol=0.0001, C=1.0,
fit_intercept=True, intercept_scaling=1, class_weight=None, random_state=None, solver='warn', max_iter=100,
multi_class='warn', verbose=0, warm_start=False, n_jobs=None)
```

Source:http://scikitlearn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html#sklearn.linear_model.LogisticRegression

Changed Parameter:

1. random_state=0 : The seed of the pseudo random number generator to use when shuffling the data.
2. Solver='lbfgs' : Algorithm used for the optimization problem.
3. Max_iter=3000 : Maximum number of iteration required by the solver to converge.
4. Multi_class='multinomial' : The loss minimized is the multinomial loss fit across the entire probability distribution

Results:

Precision: 0.91383219
Recall: 0.8837719
Accuracy: 0.92086956
Confusion Matrix: [[1312 76]
[106 806]]

From the confusion matrix, we can clearly observe that Type-2 (False Negative samples) error is less than the Type-1(False Positive samples) error. Due to this, we got smaller Recall and higher precision value. For this experiment, we got accuracy 92.08% using logistic regression which is almost same compare to the SVM (92.43%). For this dataset, logistic regression classification performs better than naïve bayes. Logistic regression and SVM are discriminative models where naïve bayes is a generative model. Generative model perform well when we have small dataset but for the spambase dataset, we have 2300 samples for testing. Thus, bigger dataset like spambase, discriminative models perform well compare to generative model.