Report on Assignment 2

The accuracy, precision, recall and F1 score for the given algorithms are as follows

1] **Multinomial Naive Bayes on the Bag of words model**

In Multinomial Naïve Bayes algorithm we used Bag of words model and calculated their accuracies. Following tables shows accuracy, precision, recall, and F1 score for the given three datasets.

For Dataset1

Class	Precision	Recall	F1-score	Support
Spam	0.89	0.96	0.92	130
Ham	0.99	0.95	0.97	348
Avg/total	0.96	0.96	0.96	478

Accuracy Score: 0.9560669456066946

The overall accuracy at test set is **95.60%**

For Dataset2

Class	Precision	Recall	F1-score	Support
Spam	0.91	0.93	0.92	149
Ham	0.96	0.95	0.96	307
Avg/total	0.95	0.95	0.95	456

Accuracy Score: 0.9451754385964912

The overall accuracy at test set is **94.51%**

For Dataset3

Class	Precision	Recall	F1-score	Support
Spam	0.98	0.86	0.92	391
Ham	0.73	0.95	0.82	151
Avg/total	0.91	0.89	0.89	543

Accuracy Score: 0.8858195211786372

The overall accuracy at test set is **88.58%**

2] **Discrete Naive Bayes on the Bernoulli model**

In Discrete Naïve Bayes algorithm we used Bernoulli model and calculated their accuracies. Following tables shows accuracy, precision, recall, and F1 score for the given three datasets.

For Dataset1

Class	Precision	Recall	F1-score	Support
Spam	0.86	0.24	0.37	130
Ham	0.78	0.99	0.87	348
Avg/total	0.80	0.78	0.73	478

Accuracy Score: 0.7824267782426778

The overall accuracy at test set is **78.24%**

For Dataset2

Class	Precision	Recall	F1-score	Support
Spam	0.88	0.24	0.38	149
Ham	0.73	0.98	0.84	307
Avg/total	0.78	0.74	069	456

Accuracy Score: 0.7412280701754386

The overall accuracy at test set is **74.12%**

For Dataset3

Class	Precision	Recall	F1-score	Support
Spam	0.89	1.00	0.94	391
Ham	0.99	0.70	0.82	152
Avg/total	0.92	0.91	0.91	543

Accuracy Score: 0.9134438305709024

The overall accuracy at test set is **91.34%**

3] Logistic Regression on both Bag of words and Bernoulli models

In Logistic Regression algorithm we used both Bag of words and Bernoulli models to calculate their accuracies. In this I have tune the algorithms with alpha, Lambda, and number of iteration as parameters. For both models we use following tuning parameters in train and validation set.

Lambda = -3,-2,-1,1, 2, 3

Alpha = 0.1, 0.20, 0.29, 0.3, 0.01

Number of iteration = 900, 1000, 1500

Following tables shows accuracy, precision, recall, and F1 score for the given three datasets. Also I have specifies on which parameter the respective accuracy is achieve

I] Bag of Words

For Dataset1

Some tuning examples from training and validation set(Test Accuracy is Validation Accuracy)

For itr: 900 alpha: 0.3 Lambda: -2

Train Accuracy: 95.37037037037037 %

Test Accuracy: 84.17266187050359 %

For itr: 1500 alpha: 0.1 Lambda: 3

Train Accuracy: 25.308641975308642 %

Test Accuracy: 29.496402877697843 %

For itr: 1000 alpha: 0.29 Lambda: 2

Train Accuracy: 97.8395061728395 %

Test Accuracy: 95.68345323741008 %

Lambda = 2, alpha = 0.29, iteration = 1000

Class	Precision	Recall	F1-score	Support
Spam	0.89	0.78	0.83	130
Ham	0.92	0.97	0.94	348
Avg/total	0.91	0.91	0.91	478

Accuracy Score: 0.9142259414225942

The overall accuracy at test set is **91.42%**

For Dataset2

Some tuning examples from training and validation set(Test Accuracy is Validation Accuracy)

For itr: 1000 alpha: 0.2 Lambda: 2

Train Accuracy: 93.33333333333333 %

Test Accuracy: 91.11111111111111 %

For itr: 900 alpha: 0.3 Lambda: 1

Train Accuracy: 100.0 %

Test Accuracy: 97.77777777777 %

For itr: 1000 alpha: 0.2 Lambda: 2

Train Accuracy: 93.33333333333 %

Test Accuracy: 91.11111111111 %

Lambda = 2, alpha = 0.2, iteration = 1000

Class	Precision	Recall	F1-score	Support
Spam	0.83	0.98	0.90	149
Ham	0.99	0.91	0.95	307
Avg/total	0.94	0.93	0.93	456

Accuracy Score: 0.9298245614035088

The overall accuracy at test set is **92.98%**

For Dataset3

Some tuning examples from training and validation set (Test Accuracy is Validation Accuracy)

For itr: 900 alpha: 0.1 Lambda: 3

Train Accuracy: 98.66310160427807 %

Test Accuracy: 96.27329192546584 %

For itr: 1000 alpha: 0.2 Lambda: -3

Train Accuracy: 97.86096256684492 %

Test Accuracy: 94.40993788819875 %

For itr: 1500 alpha: 0.3 Lambda: 2

Train Accuracy: 32.887700534759354 %

Test Accuracy: 29.81366459627329 %

Lambda = 3, alpha = 0.1, iteration = 900

Class	Precision	Recall	F1-score	Support
Spam	0.96	0.91	0.94	391
Ham	0.80	0.91	0.85	152
Avg/total	0.92	0.91	0.91	543

Accuracy Score: 0.9116022099447514

The overall accuracy at test set is **91.16%**

II] Bernoulli model

Note : Tuning for this model is also done in similar manner as shown in Bag of Words Model

For Dataset1

Lambda = -3, alpha = 0.1, iteration = 900

Class	Precision	Recall	F1-score	Support
Spam	0.86	0.85	0.86	130
Ham	0.95	0.95	0.95	348
Avg/total	0.92	0.92	0.92	478

Accuracy Score: 0.9225941422594143

The overall accuracy at test set is **92.25%**

For Dataset2

Lambda = 2, alpha = 0.3, iteration = 1000

Class	Precision	Recall	F1-score	Support
Spam	0.86	0.98	0.92	149
Ham	0.99	0.93	0.96	307
Avg/total	0.95	0.94	0.94	456

Accuracy Score: 0.9429824561403509

The overall accuracy at test set is **94.29%**

For Dataset3

Lambda = 3, alpha = 0.01, iteration = 1500

Class	Precision	Recall	F1-score	Support
Spam	0.97	1.00	0.99	391
Ham	0.99	0.93	0.96	152
Avg/total	0.98	0.98	098	543

Accuracy Score: 0.9797421731123389

The overall accuracy at test set is **97.97%**

4] SGDClassier on both Bag of words and Bernoulli models

In SGDClassifier I have used both Bag of words and Bernoulli models to calculate their accuracies. In this I have tune the algorithms with loss, alpha, penalty and number of iteration as parameters. For both models I used following tuning parameters in train and validation set.

Loss = hinge, log, squared_hinge, perceptron.

Alpha = 0.001, 0.01, 0.1

Penalty = "l2", "l1"

Number of iteration = 5, 10, 20, 50, 100, 1000

I have calculated the best accuracies for both models with respective iterations and specified below.

I] Bag of words:

For Dataset1

 Number of iteration = 1000

The best accuracy at test set is **94.56%**

For Dataset3

Number of iteration = 150

The best accuracy at test set is **92.81%**

For Dataset2

Number of iteration = 1000

The best accuracy at test set is **94.29%**

II] Bernoulli Models

For Dataset1

Number of iteration = 90

The best accuracy at test set is **93.93%**

For Dataset2

Number of iteration = 150

The best accuracy at test set is **95.17%**

For Dataset3

Number of iteration = 60

The best accuracy at test set is **94.84%**

1] Which data representation and algorithm combination yields the best performance (measured in terms of the accuracy, precision, recall and F1 score) and why?

Ans: Logistic Regression with Bernoulli data models yields the best result in this experiment. As we know that Logistic regression performs well in boolean case. In this model the Logistic regression predict the probabilities by using direct functional form. Because of all the features are independent in Logistic Regression, the algorithm assign the weights to features to define its value in overall prediction of class. This weights are generated by decreasing the difference of two features, which helps the algorithm to generate an accurate weights by using sigmoid function. This function works well in discrete data, which leads for decreasing errors to improve the performance of this algorithm.

2] Does Multinomial Naive Bayes perform better (again performance is measured in terms of the accuracy, precision, recall and F1 score) than LR and SGDClassiffier on the Bag of words representation? Explain your yes/no answer.

Ans: No, in my experiment Multinomial Naïve Bayes has not performed well as compared to LR and SGD. This is because, Multinomial Naïve Bayes considers features as conditionally independent and find the conditional probability of features depending upon the frequencies of words in the email. And by these probabilities we check how much the word in test dataset have role in classifying the results. Where as in LR and SGD perform gradient ascent or stochastic gradient ascent to find the weights. These weights decides whether the word will help for classifying examples.

3] Does Discrete Naive Bayes perform better (again performance is measured in terms of the accuracy, precision, recall and F1 score) than LR and SGDClassifier on the Bernoulli representation? Explain your yes/no answer.

Ans: No, It seems that LR and SGD are performing really well than Discrete Naïve Bayes on the Bernoulli representation. Because, as Logistic Regression and SGD are discriminative model, i.e. they did not generates the data but use the datasets to generates weights in

order to decide the role of a word during classification of examples. They use Batch wise updates of weights, Batch 1 in case of stochastic gradient updates, which not only helps to increases speed but also improves the performance of the algorithm. Also Naïve bayes has a higher bias but lower variance. The LR and SGD also uses the cost function to reduce the error so that the weights will help to decide the role of the words in the best way it possible.

4] Does your LR implementation outperform the SGDClassifier (again performance is measured in terms of the accuracy, precision, recall and F1 score) or is the difference in performance minor? Explain your yes/no answer?

Ans: According to my experiment, it seems that at Bag of Word model the Logistic Regression perform similar to less of SGD. This is due to SGD uses stochastic gradient descent considers only one data point at a time during updating weights, which increases the speed and performance of the algorithm. But during Bernoulli model it seems that Logistic Regression doing a great job than SGD. Clearly here, the bath updates working properly along with a nonlinear sigmoid function which helps to classify the examples at its best way.