CS6375.001 Project 1 Write up
Pranith Mullapudi PXM220087

Results for the different models:
Multinomial with BoW representation

Dataset	Accuracy	Precision	Recall	F1 Score
1	0.940789	0.891026	0.932886	0.911475
2	0.945607	0.856164	0.961538	0.905797
4	0.793738	1.000000	0.713555	0.832836
avg	0.893378	0.91573	0.869326	0.883369

Bernoulli with Bernoulli representation

Dataset	Accuracy	Precision	Recall	F1 Score
1	0.730263	0.861111	0.208054	0.335135
2	0.776151	0.896552	0.200000	0.327044
4	0.917127	0.896789	1.000000	0.945586
avg	0.807847	0.884817	0.469351	0.535921

Logistic Regression with BoW representation

Dataset	Accuracy	Precision	Recall	F1 Score
1	0.951754	0.894410	0.966443	0.929032
2	0.956067	0.916031	0.923077	0.919540
4	0.961326	0.949029	1.000000	0.973848
avg	0.956382	0.919823	0.963173	0.940806

Logistic Regression with Bernoulli representation

Dataset	Accuracy	Precision	Recall	F1 Score
1	0.958333	0.927632	0.946309	0.936877
2	0.943515	0.918699	0.869231	0.893281
4	0.966851	0.955990	1.000000	0.977500
avg	0.956233	0.934107	0.938513	0.935886

When implementing Logistic Regression, I designed the lambda to be tuned per training as can be seen by the console outputs when choosing LR in my code. I set an array of 5-6 possible values as a “lambda space” and performed training with them for 1000 iterations each, selecting the one with the best accuracy at the end. This training was done using a 70-30 split on the training set as specified in the instructions. For learning rate and iteration count, I manually went through the combinations using a set of possible values for each and checking each permutation’s performance. I ended up with a learning rate = .1 and iteration count = 5000. These gave the best results while still running in a reasonable time. On my system the 10,000 total iterations (1,000 for each possible lambda and 5,000 with the final one) takes a little under 30 seconds to run on the largest dataset (4). I did try some counts larger but decided they took too long.

1. Comparing the different models, Logistic Regression tended to perform better overall, with both better accuracies and F1 Scores almost across the board. This makes sense when looking at the diagram presented in lecture about which models to use when, as this use case is an example where the Naive Bayes assumption doesn’t hold very well. Intuitively, all emails have their own topics spam or not so the occurrences of words within them is not conditionally independent on a word to word basis.

2. Both versions of Logistic Regression performed very similarly, with it hard to pick a clear winner. Going off of the tables LR with the Bag of Words representation seems to be the best. One thing to be noted is that the only real difference between the two is that LR with BoW had better Recall but worse Precision. This also makes sense intuitively since spam emails are more likely to have many repeating words since they aren’t likely to be as coherent/formatted as regular emails. If anything it’s surprising that there isn’t a larger gap between their performances

3. It did not, though was somewhat close in performance. The real issues come into light on the largest dataset (4) where the gap in performance widens drastically with the NB model being too selective (high precision but low recall). This is since LR makes fewer assumptions and is thus safer the larger the dataset is.

4.Once again the NB version was worse, although while multinomial was somewhat comparable, the Bernoulli NB performed much worse. Interestingly Bernoulli has the opposite results compared to multinomial with poor performance on the smaller datasets but pretty decent on the large one. But even with that it performed worse in every single evaluation metric compared to LR with a bernoulli distribution, with especially poor performance on the smaller datasets (1 and 2) of accuracies in the 70%s which were the lowest in general for any model and representation. As mentioned with question 2, this is likely because the bernoulli assumption that the number of appearances doesn’t matter is wrong. That on top of the Naive Bayes assumption itself not holding leads this model to perform as the worst of the bunch.