CSCI-B 555 MACHINE LEARNING

PROGRAMMING PROJECT 1

Naive Bayes algorithm with Maximum Likelihood and MAP Solutions

Neha Pai - nrpai@iu.edu

Introduction

Text classification aims to categorize test samples into classes. An important application of the same is sentiment analysis. Sentiment analysis is used to classify text into positive, negative or neutral samples. The following is an implementation of the Naive Bayes algorithm with maximum likelihood and MAP solutions and its evaluation using cross validation on the task of sentiment analysis for 3 datasets from three domains imdb.com, amazon.com and yelp.com.

For the implementation, we use cross validation to split data into training and testing sets. This is advantageous as all samples are used for both training and validation. Also, we use MLE and MAP for parameter estimation. This is given by the solution for a Discrete distribution (with a Dirichlet prior) for its parameters.

Experiment 1

Our motive is to run stratified cross validation to generate learning curves for Naive Bayes. The smoothing parameter is set to m=0 and m=1 for comparison. Maximum Likelihood estimate fails to predict samples in cases where the MaxL estimate is 0, meaning that the probability of the word given the class is 0. This reduces the accuracy of classification and intuitively we expect low performance for Naive Bayes using MaxL estimates.

MAP tries to solve this problem by using a smoothing parameter m=1. This smoothes out the maximum likelihood values and avoids 0/1 extreme situations. We would expect better accuracies for MAP based classification.

We test the code against 3 datasets with 1000 sentiment labelled samples. The dataset is split using stratified cross validation. For measuring the learning curve, we use subsamples of sizes 0.1N, 0.2N… N folds of cross validation and test for different samples of test and train sets. The result of the accuracies and standard deviations are plot against the training set size for visualization purposes.

Results and Conclusions

The figures represents the MaxL learning curve in blue and MAP learning curve in red. Following are the results:

- As expected, the accuracy for MAP is greater than the MaxL accuracies for all training data set sizes.
- The accuracy increases for larger training data set sizes.
- The extent of deviation is much larger for MaxL solution.

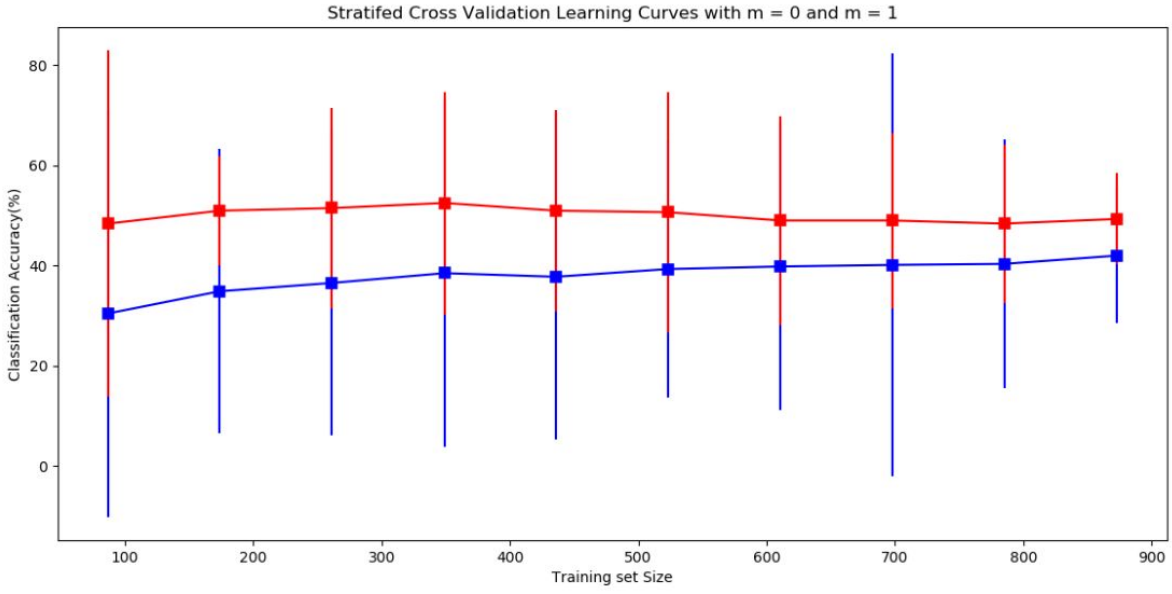

Figure 1: Learning curve for dataset - “amazon_cells_labelled.txt”

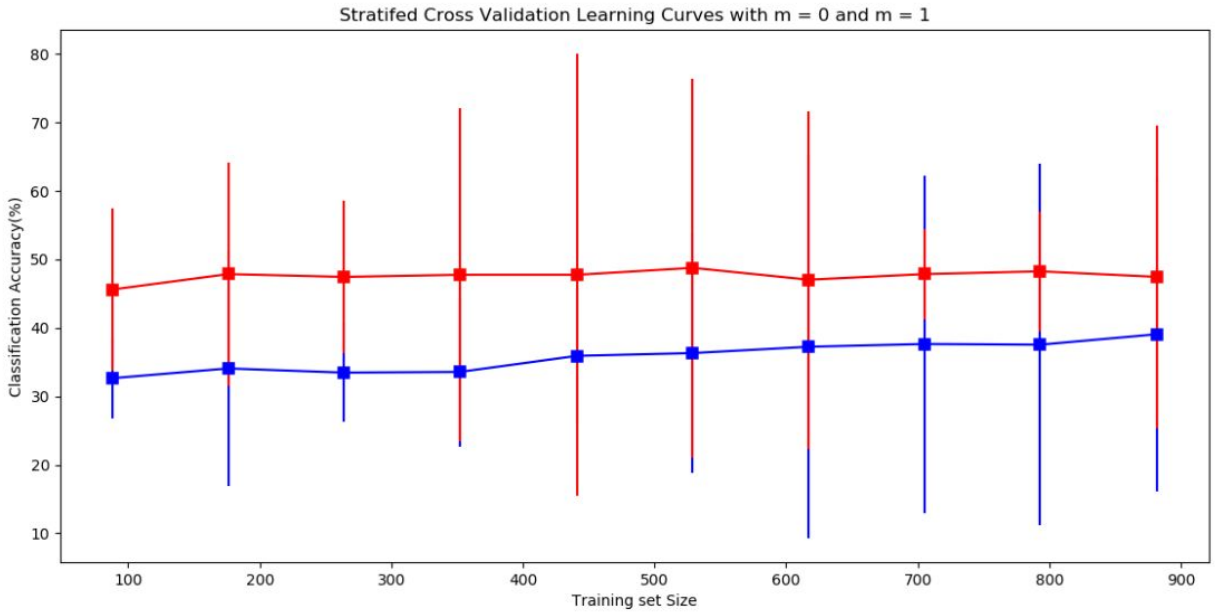

Figure 2: Learning curve for dataset - “yelp_labelled.txt”

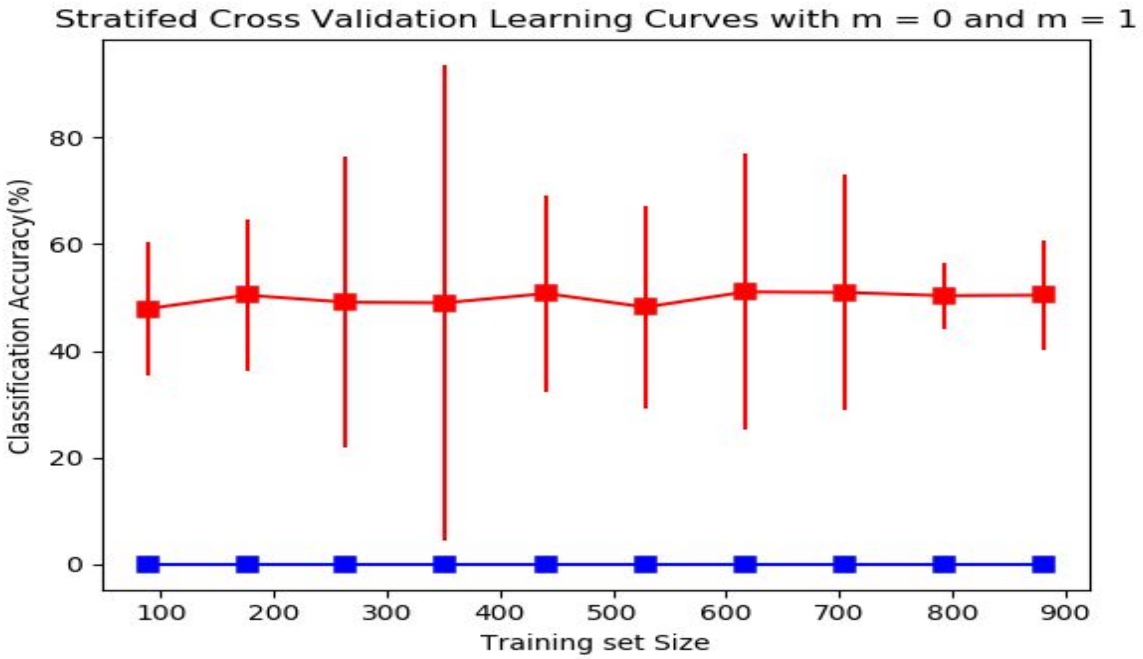

Figure 3: Learning curve for dataset - “imdb_labelled.txt”

We can conclude that for better accuracy of the model, we can use a larger data set and train the model using a smoothing parameter m=1 while using the naive bayes approach.

Experiment 2

Our motive is to run stratified cross validation for Naive Bayes with smoothing parameter m= 0.1, 0.2, 0.3..., 0.9,1,2,3... 9. We want to visualize the effect of smoothing on the accuracy and standard deviation of the model. Adding a smoothing parameter to our calculations increases the accuracy as it handles the cases where we are trying to classify words which we have not encountered in our training data set. We would expect the accuracy to improve for some increase in smoothing parameter value.

Results and Conclusions

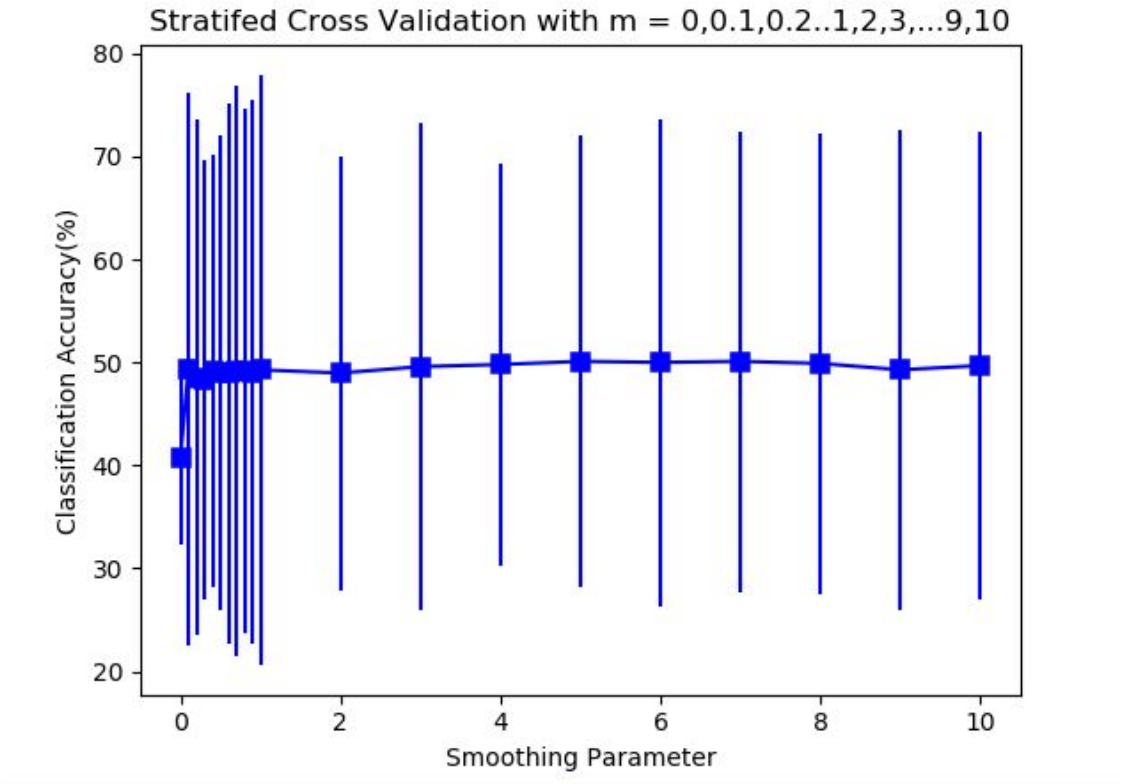

Figure 4: Stratified cross validation with m= 0.1, 0.2, 0.3..., 0.9,1,2,3... 9 for dataset - "amazon_cells_labelled.txt"

Following are the results:

- m=0 gives a low accuracy for all datasets tested.
- Standard deviation for m<1 is greater and accuracies are more spread out. For greater m, accuracies are less varied.
- Accuracy slightly improves for increasing smoothing parameter upto a limit.

We can conclude that adding a smoothing parameter has a positive effect on the overall accuracy of the model.

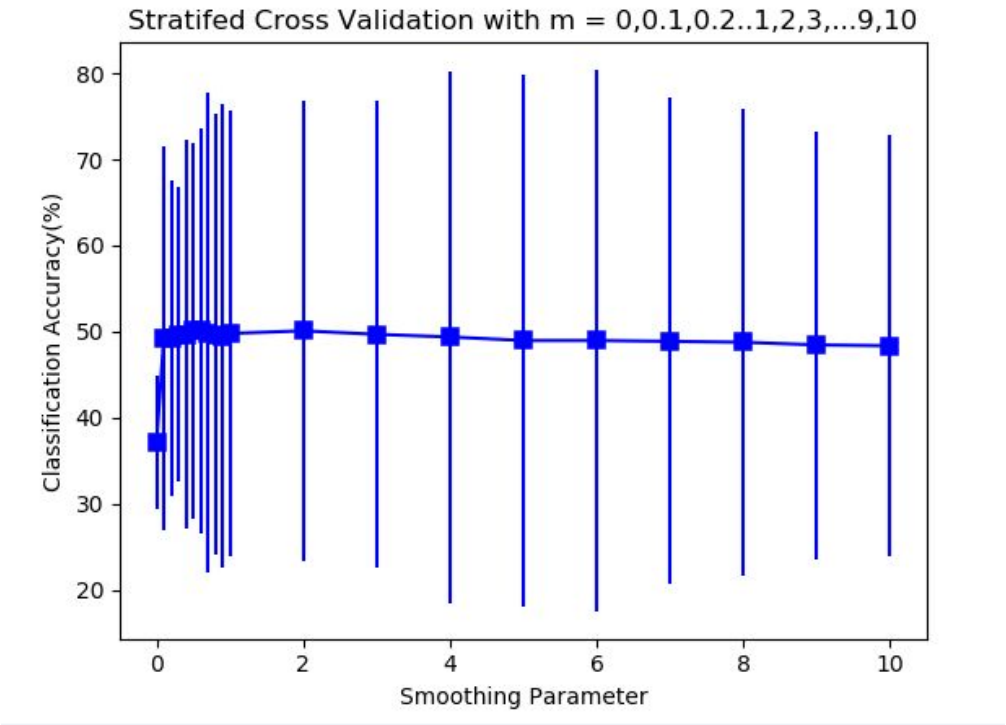

Figure 5: Stratified cross validation with m= 0.1, 0.2, 0.3..., 0.9,1,2,3... 9 for dataset - "yelp_labelled.txt"

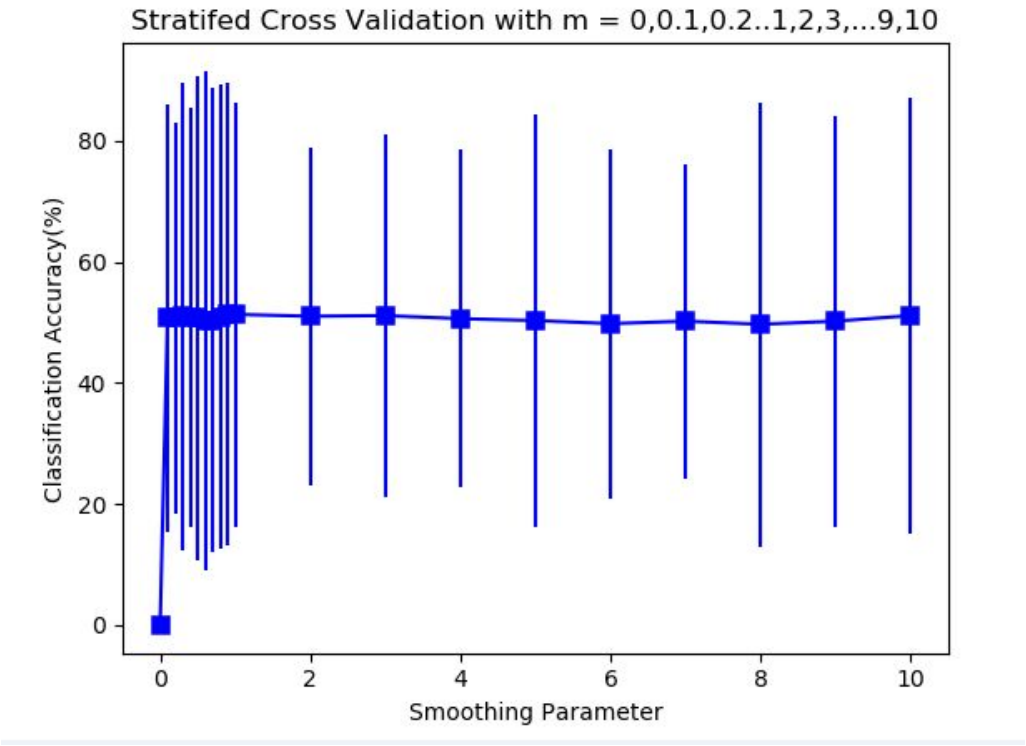

Figure 6: Stratified cross validation with m= 0.1, 0.2, 0.3..., 0.9,1,2,3... 9 for dataset - "imdb_labelled.txt"