Machine Problem #2
COT 5610 - Machine Learning
Fall 2014
Due 09-18-2014

Edward Aymerich

1 Problem description

A Naive Gaussian Bayes (NGB) classifier algorithm was implemented and applied to the MNIST data set. This data set consist of 60,000 training examples of handwritten digits, and 10,000 test examples.

In order to test the NGB classifier, three test protocols were used:

- Test Protocol 1: using all 50,000 training examples from training set for training, and all 10,000 test examples from test set for testing. To avoid division by zero, a small value ϵ was added to the estimated variance. Different values of ϵ were used to test the impact of ϵ on performance.
- Test Protocol 2: the training set was divided into a training subset (with 50,000 examples), and a validation set (with 10,000 examples). The validation set was used to find the optimal value for α_d among the values {1%, 2%, 4%, 8%, 16%}, and the optimal value was used to test the model against the original test set.
- Test Protocol 3: only the training set was used in this protocol. The NGB algorithm was tested using 5-fold cross-validation and 10-fold cross-validation on the training set.

2 NGB implementation

The NGB algorithm was implemented on C++11, using OpenMP directives to accelerate the experiment runs.

For each dimension of feature vector (corresponding to each pixel in the MNIST data set) the mean and variance was estimated.

3 Results

This section presents the error rates resulting from applying the implemented NGB algorithm on the MNIST data set, using the test protocols described on the Problem Description. The implemented NGB algorithm was compiled using GCC 4.9.1 on a Windows 7 64 bit platform, and the experiments were executed using a Intel Core i5 3470 processor running at 3.2GHz.

3.1 Test Protocol 1

The best value of ϵ was determined by testing different values and observing the obtained error rate, as shown on figure 1.

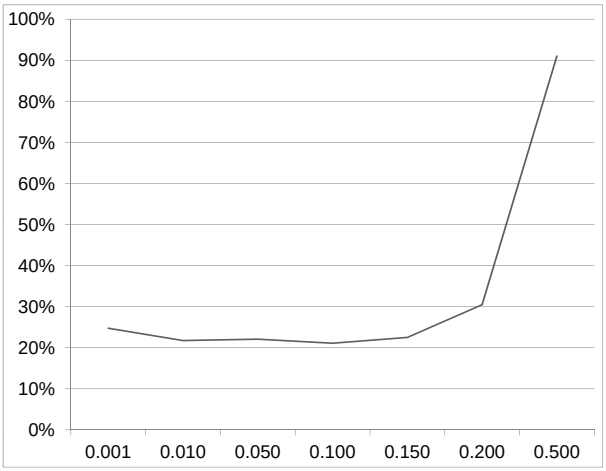

Figure 1: Error rate for Test Protocol 1, for values of ϵ ranging from 0.001 to 0.5.

The lowest error rate happens when $\epsilon = 0.1$, with a value of 21.09%. A second test was conducted to test values of ϵ around 0.1, and the results are shown on figure 2.

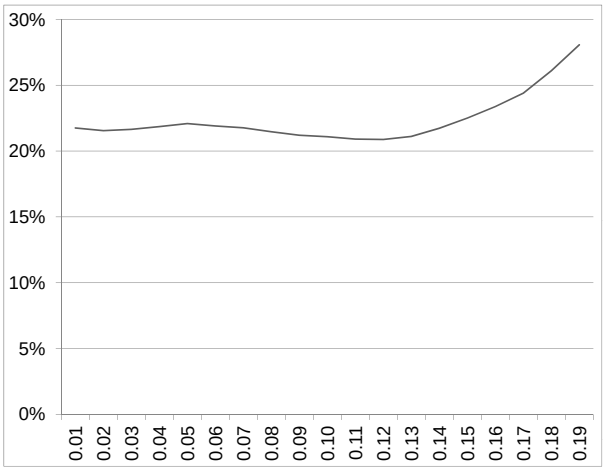

Figure 2: Error rate for Test Protocol 1, for values of ϵ ranging from 0.001 to 0.5.

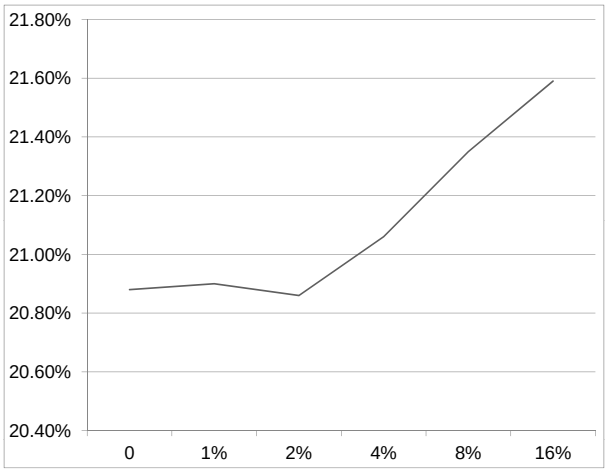

Figure 3: Error rate for Test Protocol 1, for MLE ($\alpha_d = 0$) and MAP (α_d={1%, 2%, 4%, 8%, 16%}).

The error rate reaches the minimum when $\epsilon = 0.12$, with a value of 20.88%. This value of ϵ was used in all subsequent experiments. The full results of the tests for different values of ϵ are shown in table 1.

ϵ	error rate (%)	ϵ	error rate (%)
0.001	24.75	0.11	20.91
0.01	21.75	0.12	20.88
0.02	21.55	0.13	21.11
0.03	21.65	0.14	21.73
0.04	21.86	0.15	22.51
0.05	22.09	0.16	23.38
0.06	21.91	0.17	24.40
0.07	21.77	0.18	26.10
0.08	21.47	0.19	28.07
0.09	21.20	0.20	30.48
0.10	21.09	0.50	90.99

Table 1: Error rates for different values of ϵ.

Two solutions were used to estimate the prior distribution P(Y): Maximum Likelihood Estimation (MLE) and Maximum A Posterior parameter estimation (MAP). For MAP, different values of α_d were used. The results are shown on figure 3.

When the prior is calculated using MAP with $\alpha_d = 2\%$, the error rate reaches a minimum value of 20.86%. The results of Test Protocol 1 are shown on table 2.

Prior estimation	α_d (%)	error rate (%)
MLE	n/a	20.88
MAP	1	20.90
	2	20.86
	4	21.06
	8	21.35
	16	21.59

Table 2: Results obtained from Test Protocol 1.

3.2 Test Protocol 2

In Test Protocol 2, first we tune the value of parameter α_d using a validation set, composed of 10,000 examples extracted from the original training set. The remaining 50,000 examples are used as the training subset to tune α_d.

The error rates for different values of α_d under Test Protocol 2 can be observed on figure 4.

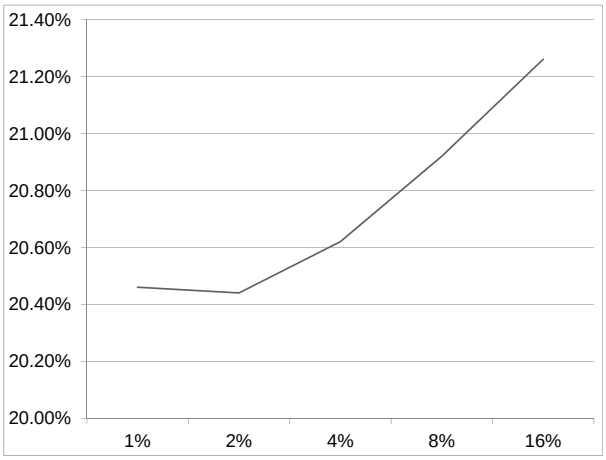

Figure 4: Error rate for Test Protocol 2, for MAP (α_d={1%, 2%, 4%, 8%, 16%}).

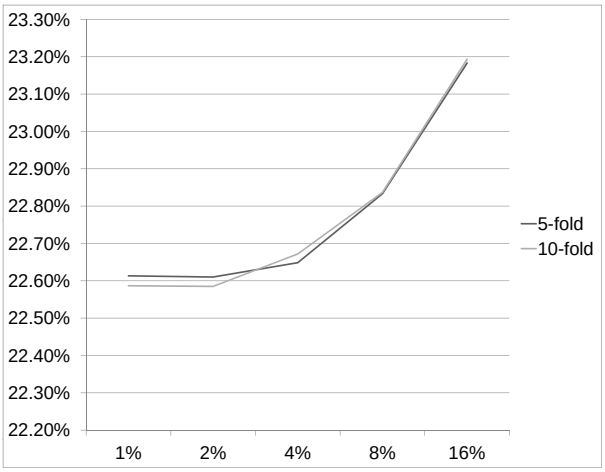

Figure 5: Error rate for Test Protocol 3, for MAP (α_d={1%, 2%, 4%, 8%, 16%}).

The minimum error rate of 20.44% is obtained when $\alpha_d = 2\%$. Using this value on the test set, an error rate of 20.77% was attained. All the results obtained from Test Protocol 2 are shown on table 3.

set	α_d (%)	error rate (%)
validation	1	20.46
	2	20.44
	4	20.62
	8	20.92
	16	21.26
test	2	20.77

Table 3: Results obtained from Test Protocol 2.

3.3 Test Protocol 3

In test protocol 3, a 5-fold cross-validation and a 10-fold cross-validation was done over the original training set. The average error rates for both cross validations using different values of α_d can be observed on figure 5.

Both cross-validations show a similar behavior. For the 5-fold cross-validation, the minimum average error rate is 22.61% using $\alpha_d = 2\%$. For the 10-fold cross-validation, the minimum average error is lower, with a value of 22.59%, also using $\alpha_d = 2\%$.

All the results obtained from Test Protocol 2 are shown on table 4.

α_d (%)	5-fold average error rate (%)	10-fold average error rate (%)
1	22.61	22.59
2	22.61	22.59
4	22.65	22.67
8	22.83	22.84
16	23.18	23.19

Table 4: Results obtained from Test Protocol 3, for MAP (α_d={1%, 2%, 4%, 8%, 16%}).

4 Conclusions

Compared with the KNN algorithm implemented for Machine Problem 1, the implemented NGB algorithm performs worse, with an error rate of 20.77% (in the best result) against a 2.88% error rate for the best case of KNN. This could be due to the fact that Naive Gaussian Bayes classifier assumes that all dimensions in the feature vector are independent, when in fact for the MNIST data set there is a strong correlation of each dimension, given that these are in fact pixels, and it is easy to see that each pixel has to relate to its neighboring pixels in order to form the picture of a hand written digit. In this sense, it is possible that the KNN algorithms retains some correlation between the pixels in a better way that NGB does.

During the experiments, no over fitting effect was observed, and changing the parameter α_d had very little effect on the error rate. It can be noticed that higher values of α_d (8% and 16%) also increased the error rate, but only for less than 1%.