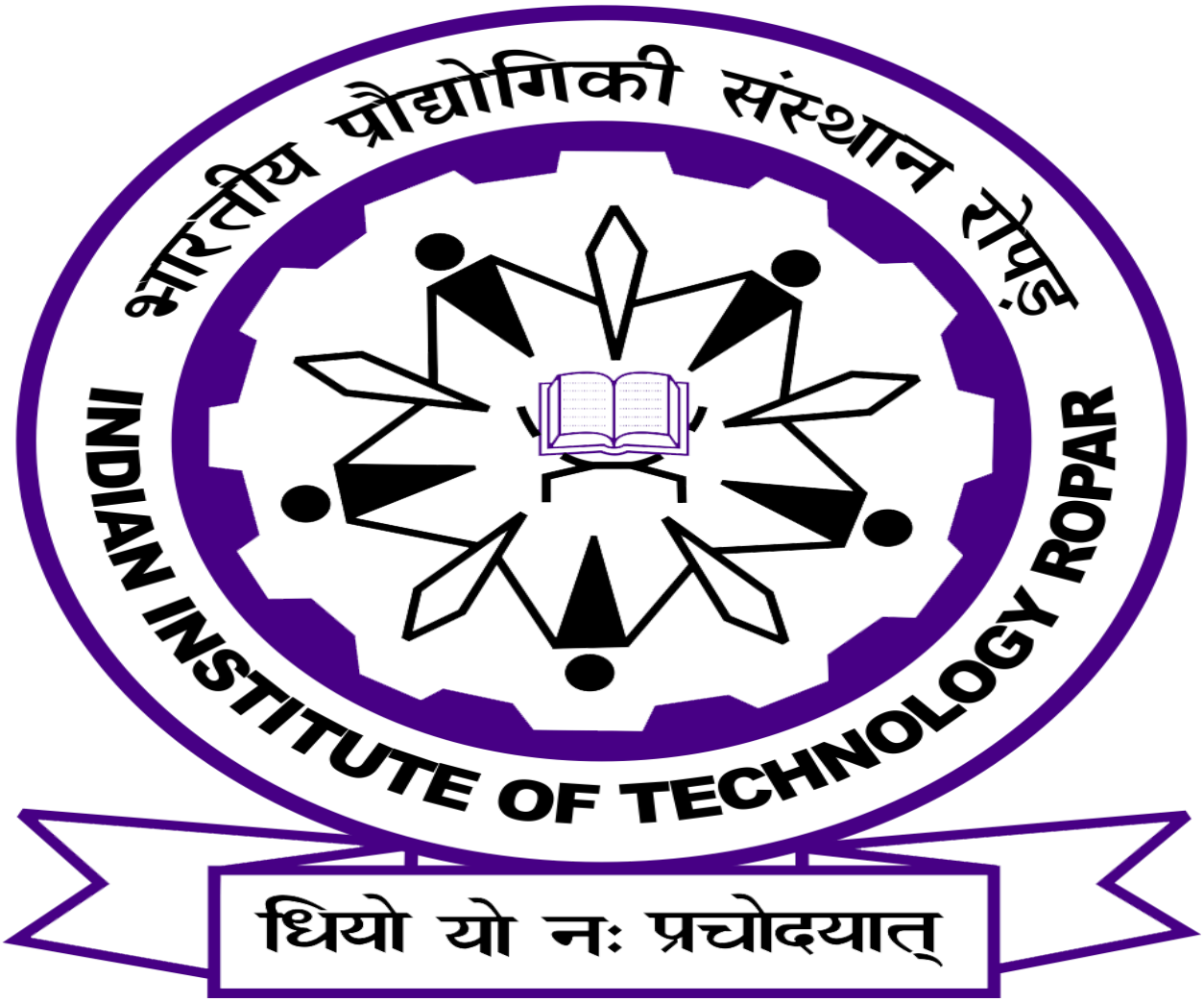
Naive Bayes & Clustering

- Lab 4 Report -



Prasad Kshirsagar

2016csb1041

Computer Science & Engineering

Naive Bayes Classifier

INTRODUCTION

The aim of the first part of this lab was to classify the test email as either **Spam** or **Ham** by using Naive Bayes Classifiers. In machine learning **naive Bayes classifiers** are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the features.

HYPOTHESIS

It is a classification technique based on **Bayes**' Theorem with an assumption of independence among predictors. In simple terms, a **Naive Bayes classifier** assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

ALGORITHM USED

1. By **bayes' rule** , We have :

$$p(C_k \mid \mathbf{x}) = \frac{p(C_k)\, p(\mathbf{x} \mid C_k)}{p(\mathbf{x})}$$

   Where,

   p(Ck / x) = posterior probability of class given instance x

   p(Ck) = Prior probability of class Ck

   p(x / Ck) = Likelihood probability of x given Ck

2. **Vocabulary** = All distinct words & tokens in I , where I is total number of words that occur in examples.
3. **Define :**

   Ik = Examples for which target label is Ck

P(Ck) = |Ik| / |I|

n = total number of words inn Ik

4. For each word Xj in Vocabulary,

Nj = total number of times word Xj occurs in Ik

P(Xj / Ck) = (Nj + (m * p)) / (n + |Vocabulary|)

Where,

P = prior estimate of probability

m = equivalent sample size

OBSERVATIONS

1. Prior probability of both classes :

   p(spam) = 0.4804

   p(ham) = 0.5196

2. |Vocabulary| = 996

   Default m = 996, p = 1/996

3. 5 Most frequently words indicative of Spam email are :

1. enron

2. a

3. the

4. corp

5. to

4. 5 Most frequently words indicative of Ham email are :

1.

aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

2. enron

3. the

4. to

5. a

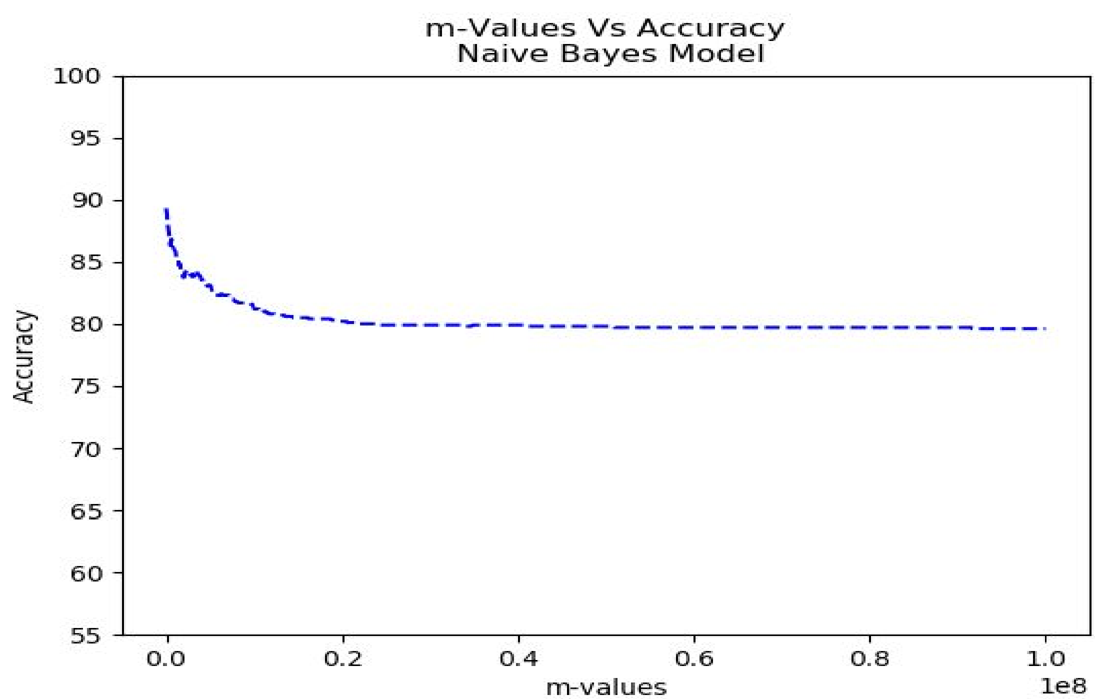
5. Accuracy of the model for default value of m : 89.3 %

6. Difficulty for finding posterior probabilities arises when the test attribute is not present in Vocabulary. Since we multiply probabilities & it may become 0 , resulting final product may be 0. Thus, we propose adding (m * p) term in numerator of bayes' rule, **so that probability of such attribute will be p and not 0.**

7. Beating our Naive bayes Classifier :

We can beat this classifier by choosing such test instances , so that all of its attributes are not in vocabulary. It will put our classifier in confused state.

8. Effect of varying m on Accuracy :



K-Means Clustering

INTRODUCTION

The aim of this part of the lab was to apply clustering to MNIST-Handwritten-Digits dataset. We did this by performing K-means clustering on the given MNIST images dataset. All the images are of dimension 20X20. We also need to check the effect of varying number of clusters on accuracy of the model. We have also built confusion matrix for each of them models.

PRE-PROCESSING

We have read 'data.txt' and created the input matrix X from this file. Dimensions of X are 5000X400. Similarly, we have created label matrix Y from 'label.txt' file. The dimensions of Y are 5000X10. Since there are only 10 classes (digits from 0 to 9).

OBSERVATIONS

1. **K = 10 :**

   Each column of this column represents column of this cluster.

2	5	6	8	7	4	7	1	0	3

   Classification Accuracy : 55.33 %

   On each run capacity is around 54-58 %. This low accuracy can be justified since some of the digits are not not even labelled. As shown above, 9 is not labelled. Thus, all of those digits are misclassified. However some of the digits have majority in more than one cluster, so majority of those digits are correctly classified & therefore rest all the digits are misclassified.

   As we can see from the confusion matrix below, instances corresponding to digit 9 are classified as either digit 4 or 7. This shows that classifier got confused & misclassified these instances. Almost all the instances belonging to digit 1 are correctly classified.

Confusion Matrix :

			A	C	T	U	A	L			
		0	1	2	3	4	5	6	7	8	9
P	0	455	0	9	2	0	19	8	2	5	3
R	1	5	497	89	51	53	156	77	66	71	42
E	2	2	0	322	15	8	1	7	2	6	1
D	3	14	1	33	281	0	144	2	1	118	8
I	4	2	1	7	14	272	36	9	159	15	230
C	5	25	1	15	10	18	190	30	5	40	6
T	6	11	0	23	2	8	12	390	1	3	2
E	7	1	0	4	4	159	13	0	269	18	212
D	8	10	1	13	131	0	119	7	0	264	2
	9	0	0	0	0	0	0	0	0	0	0

2. K = 15 :

Each column of this column represents column of this cluster.

3	7	3	6	6	4	0	0	2	1	1	8	9	8	5

Classification Accuracy : 67.39 %

As the number of clusters have increased and are more than actual number of distinct digits. Some of the digits which where previously classified in a single cluster are now divided into multiple clusters, as shown below.

Digits	**number of clusters for k=10**	**number of clusters for k=15**
0	1	2
1	1	2

8	1	2
3	1	2
6	1	2

Confusion Matrix :

			A	C	T	U	A	L			
		0	1	2	3	4	5	6	7	8	9
P	0	446	0	7	2	0	12	7	1	3	2
R	1	0	494	77	33	23	12	23	41	40	16
E	2	0	0	322	7	5	1	1	0	3	1
D	3	15	0	28	363	0	160	2	1	60	5
I	4	0	0	6	0	271	4	0	6	8	69
C	5	15	1	8	10	15	196	11	7	13	11
T	6	15	0	48	3	13	6	450	1	2	10
E	7	0	2	2	2	28	2	0	256	4	168
D	8	6	1	16	70	0	93	3	1	364	9
	9	2	2	5	13	146	20	1	177	4	206

On increasing the number of clusters some of the digits get split as the digits which were initially together now partitioned but have majority in the new clusters. But not all the digits get split & some of the digits are not even labelled.

3. K = 5 :

Each column of this column represents column of this cluster.

1	0	6	3	7

Classification Accuracy : 43.09 %

Confusion Matrix :

			A	C	T	U	A	L			
		0	1	2	3	4	5	6	7	8	9
P	0	424	0	3	3	0	9	7	2	2	2
R	1	3	495	92	59	41	163	69	63	166	55
E	2	0	0	0	0	0	0	0	0	0	0
D	3	40	3	56	408	0	246	10	0	272	11
I	4	0	0	0	0	0	0	0	0	0	0
C	5	0	0	0	0	0	0	0	0	0	0
T	6	27	0	338	7	34	14	409	3	23	7
E	7	6	2	11	23	425	68	5	432	37	425
D	8	0	0	0	0	0	0	0	0	0	0
	9	0	0	0	0	0	0	0	0	0	0

We can see that, on decreasing the number of clusters accuracy decreases. As the number of clusters are less than actual classes, So majority of instances are misclassified. Due to this some clusters might get combined.

Digits like 3,5 and 8 get merged to the single clusters because they do not good classification and classifier got confused while classifying them. Same is the case for digits 4,7 and 9.

Digits 7 and 1 do not get combined. This is because digits 7 & 1 do not have much confusion and also they have less overlapping. So, they remain separated.