# Naïve Bayes Classifier for Spam Filter of Binary Dataset

## **Theory:**

For this spam dataset is multivariate binary dataset, I choose Bernoulli Naïve Bayes Algorithm to predict the probability. Here is the equation:

, *xi* ∊ (0,1)  {\displaystyle p\_{ki}}, *pki* is the probability of class *Ck* given the value of *xi* {\displaystyle C\_{k}}

**Predict:**

for test data set:

[0 0 0 1 0 1 1 1 1 1 1 1 0 0 1 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 1 1 1

]

P(X|S) = p(X1=0|S).p(X2=0|S).p(X3=0|S ). p(X4=1|S). p(X5=0|S) … .p(X57=1|S). p(S)

P(X|NS) = p(X1=0|NS).p(X2=0|NS).p(X3=0|NS ).p(X4=1|NS).p(X5=0|NS) … . p(X57=1|NS). P(NS)

If p(X|S) > p(X|NS), it’s Spam (Pi = 1);

If p(X|S) < p(X|NS), it’s not Spam (Pi = 0).

**Accuracy:**

Get the predict result of all test data Pi , and the actual result of test data Ti

If Pi == Ti , R+1 (R means right)

Count the total number of test data set N. So the accuracy = (R/N)\*100%

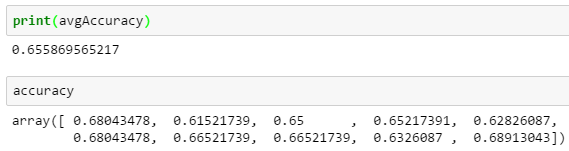
**For 10 folds’ cross validation:**

Loop the upper step 10 times, get the average accuracy of these ten times.

## **Result:**

After 10 times’ cross validation, I get the average accuracy is **65.56%**

Randomlize(shuffle) it before split to trainset and testset:

Code: data\_Spam=data\_Spam.sample(frac=1).reset\_index(drop=True) 

Not randomlize:

