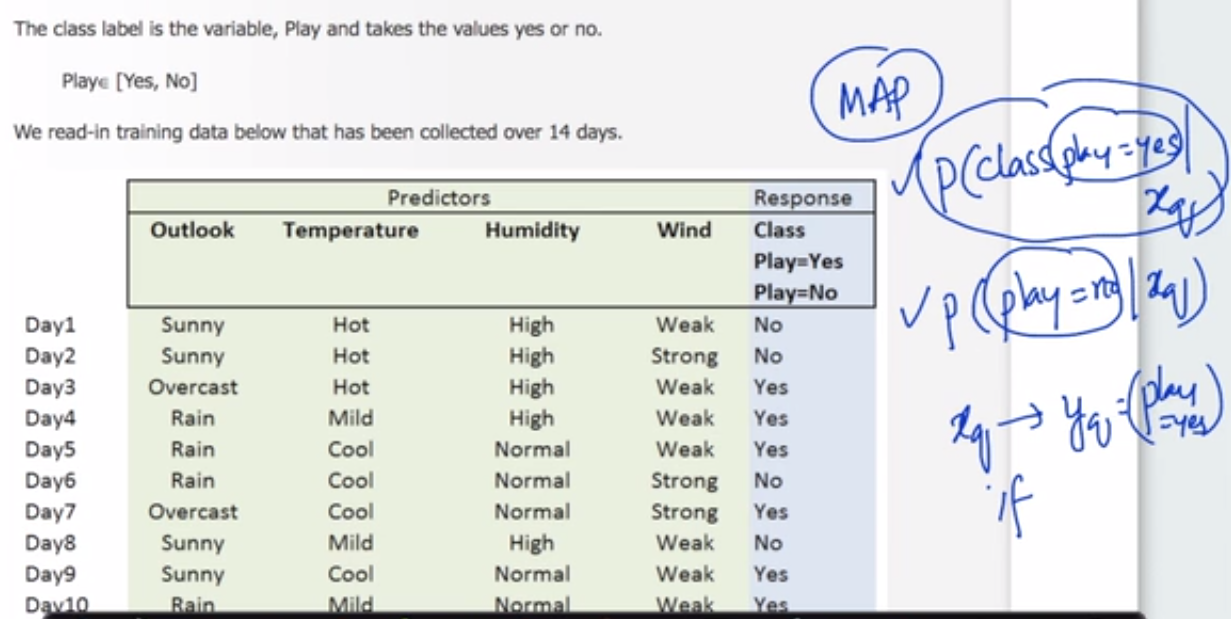
**Now lets take an real life binary classification example to understand Naïve Bayes more clearly.**

**We are taking a toy dataset in which class label is “Can Play” or “Cannot Play”**

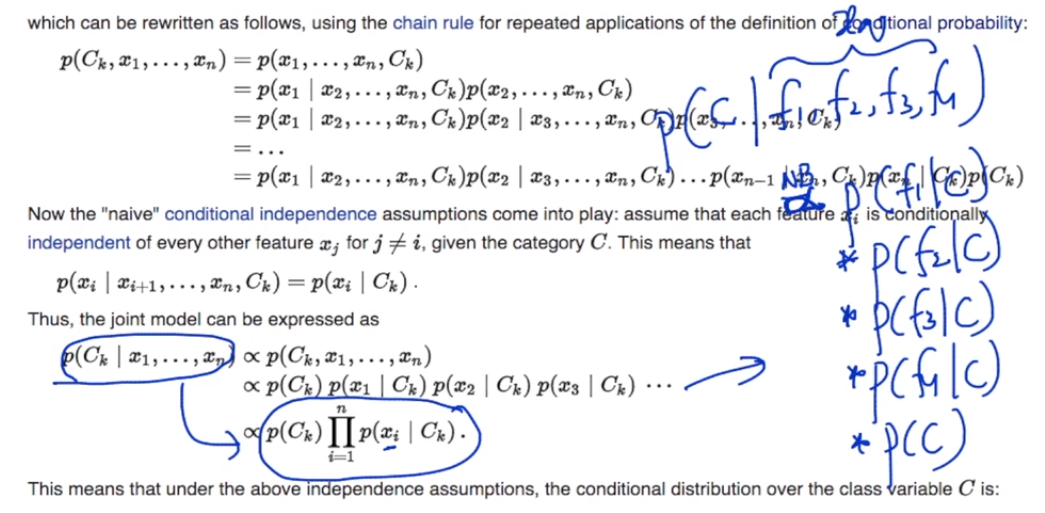
Given features are all categorical features on the basis of which we have to decide is it good time to play or not.



P(Class(Play = Yes)| xq) or P(Class(Play = No)| xq) .

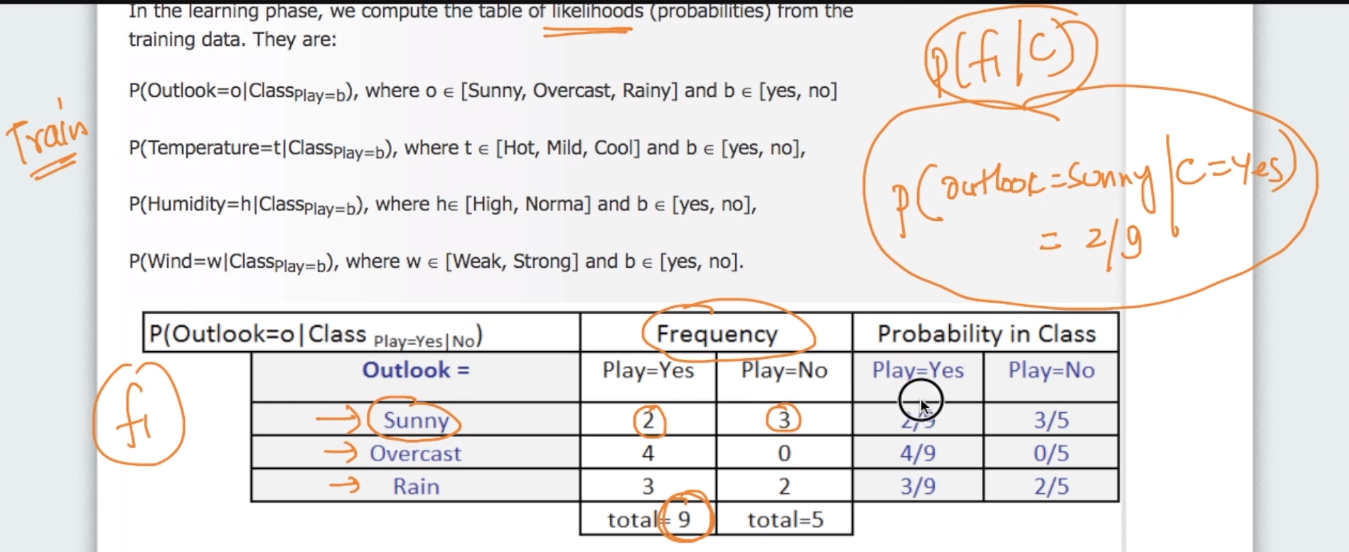
Given x q are categorical features (Outlook, Temperature, Humidity , Wind)

So this is same as what we learnt in our Naïve Bayes algo explanation.



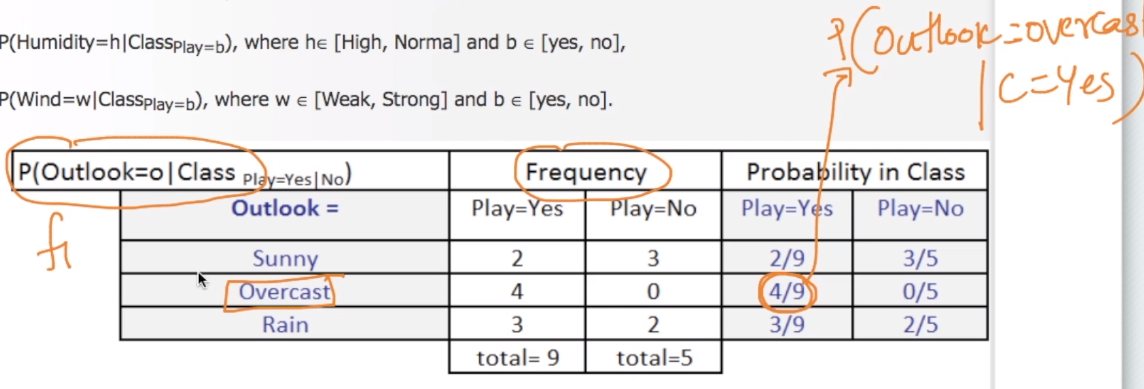
P(C | f1,f2,f3,f4) is proportional to multiplication of probability of each feature for given class label and probability of class label.

Now what we will do in out training stage is try to calculate this probability of likelihoods(by using training data) above by forming some easy tables to make

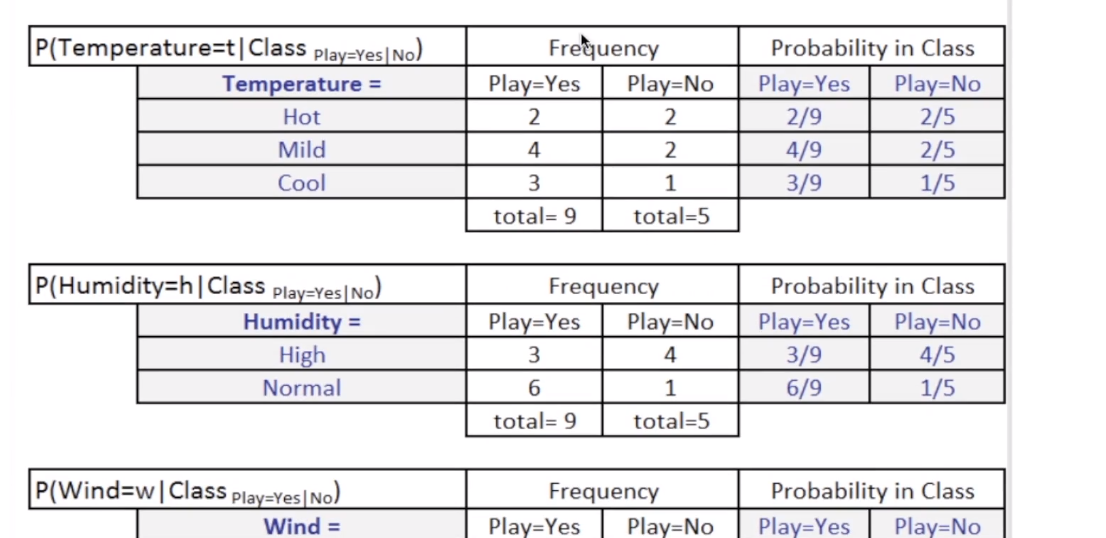


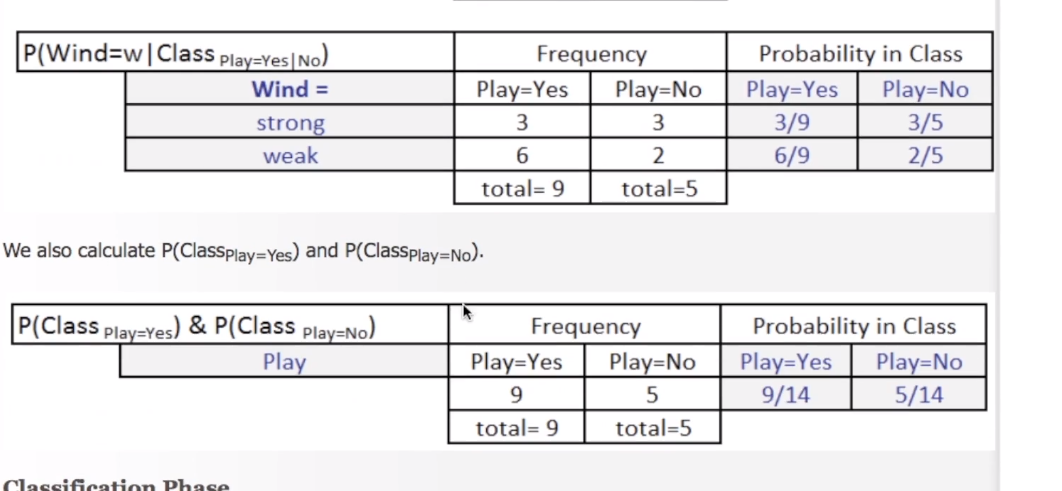
All the data above helps us answer easy question like shown above P(Outlook = Sunny | C = Yes) = 2/9

Similarly we can some other questions also like



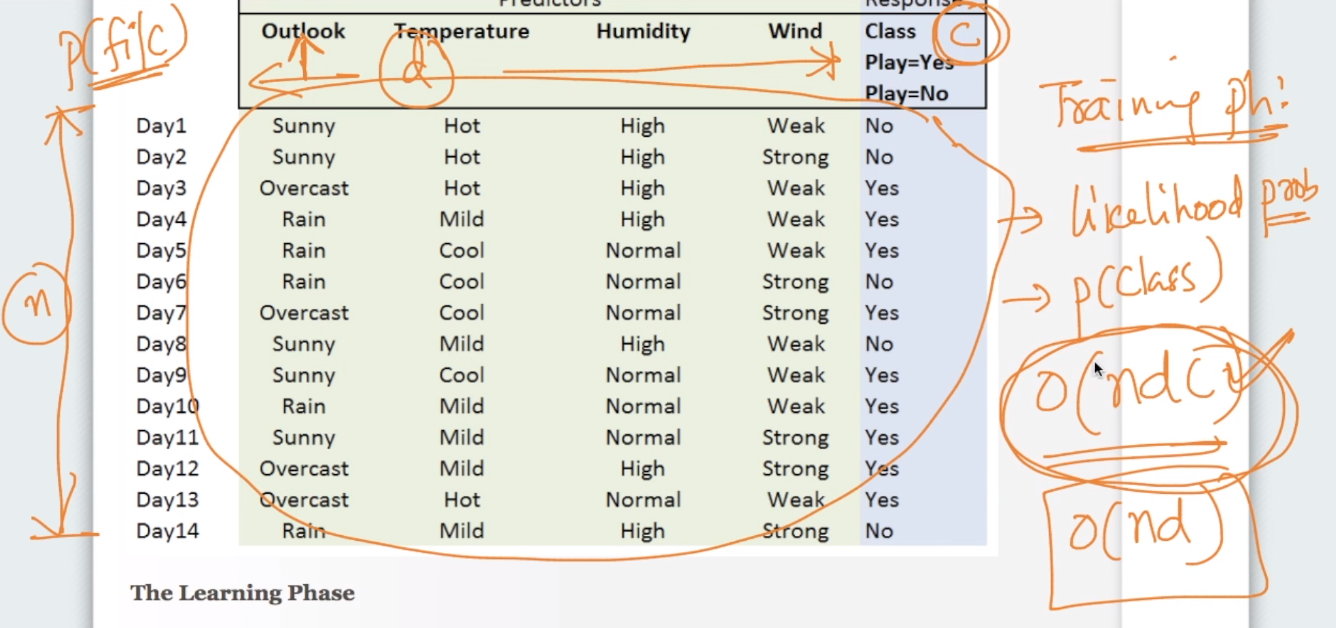
We do same thing for all the feature i.e., F1, F2, F3, F4





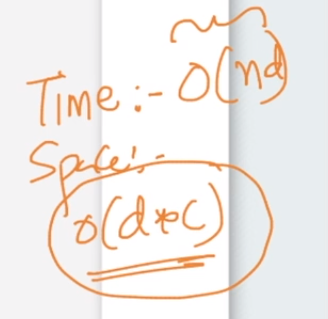
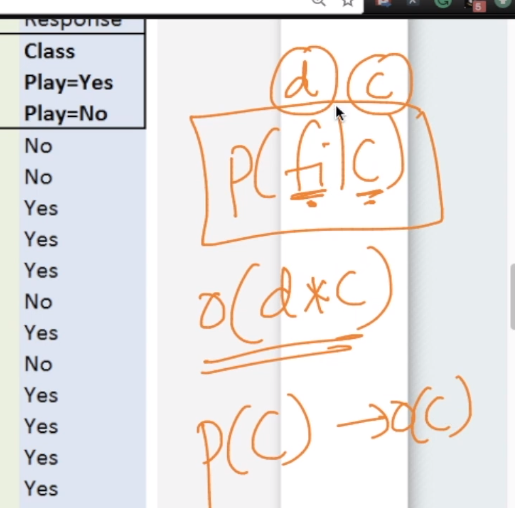
For all the categorical features we need to do this but we will see how to do this for Numerical feature.

So this is in general what we do in training stage of this algo

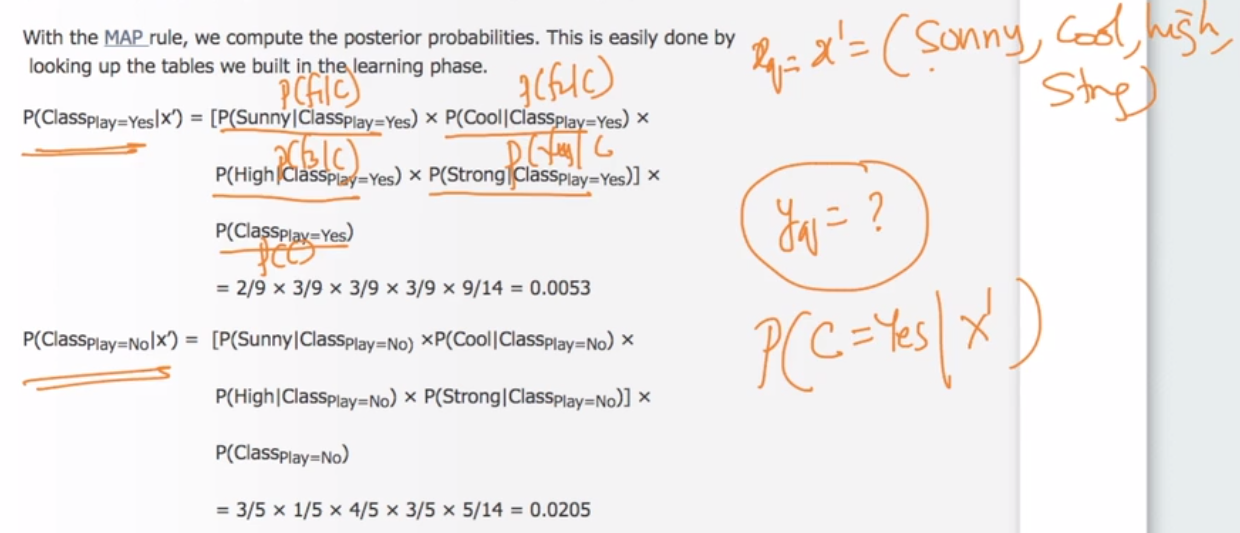


We calculate each likelihood probability and class probability . And time complexity for this when applied without any modification if O(ndc) which can be reduced to O(nd) by some modifications or even O(n) where n is no. of points in training data.

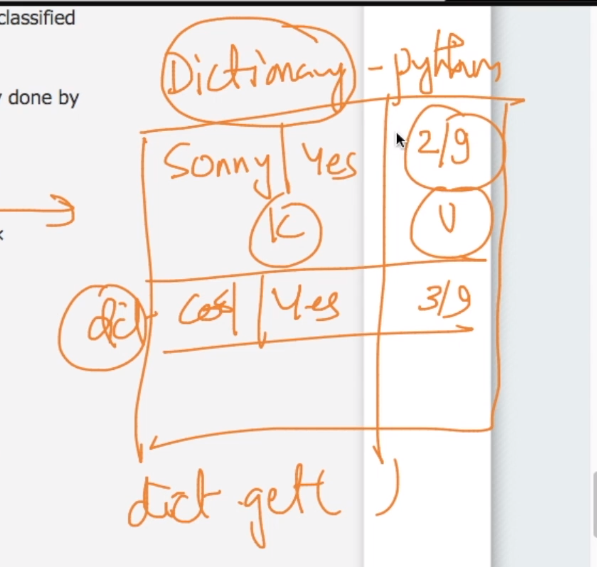
Now coming to space complexity we need to store



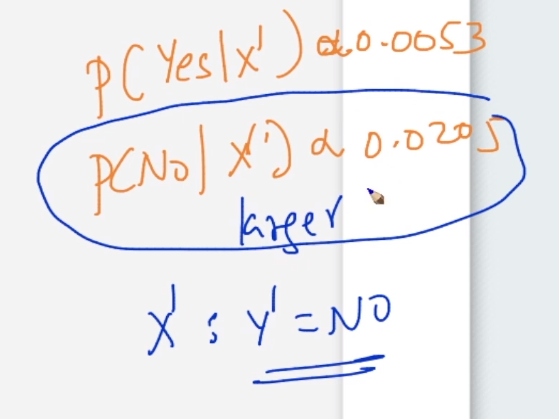
Now coming to classification in training stage we compute



All the likelihoods which we calculated can easily be stored in a python dictionary and we can use those values whenever needed.

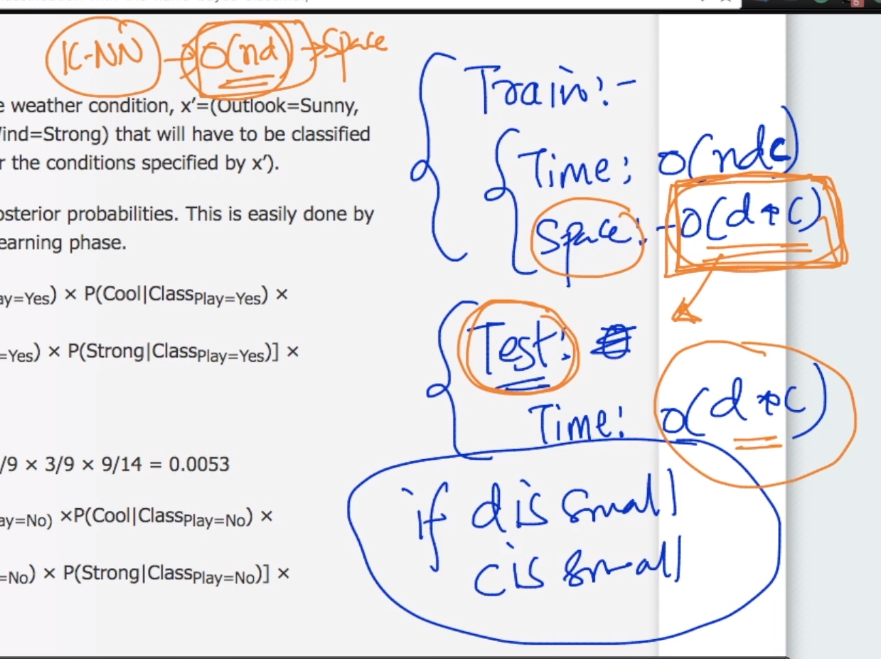


When we replace this values in our calculation above we get our final probabilities for class for given set of feature.



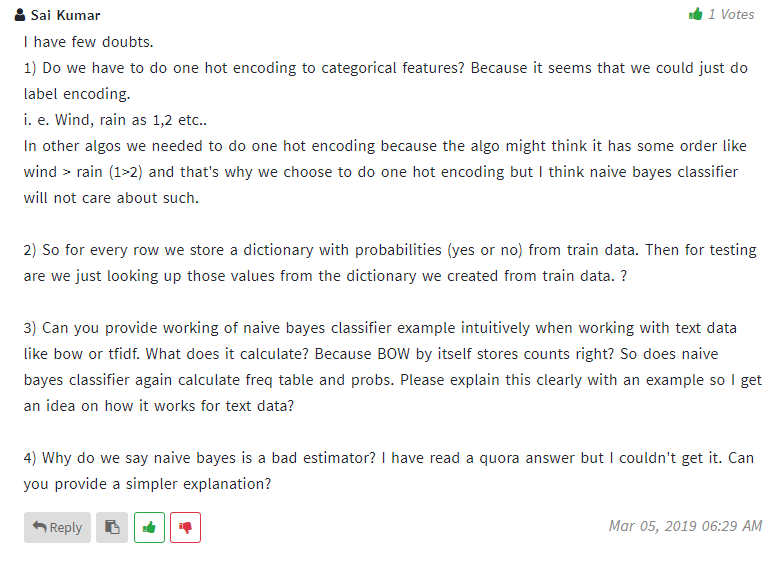
And since value for Class label “NO” is larger so we will take it as our class label to given set of feature.

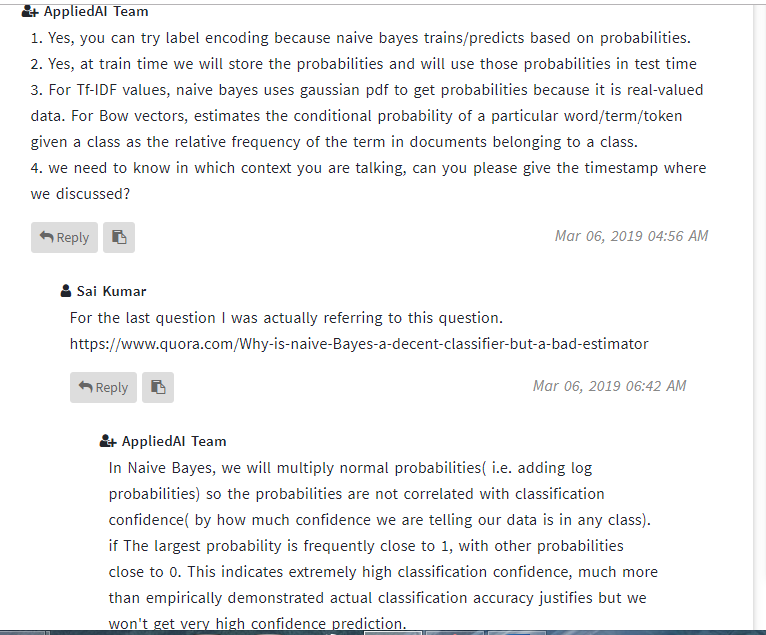
SO time and space complexity for test stage is very low as we just need to go to dictionary for every and value and don’t need to traverse to our whole data.



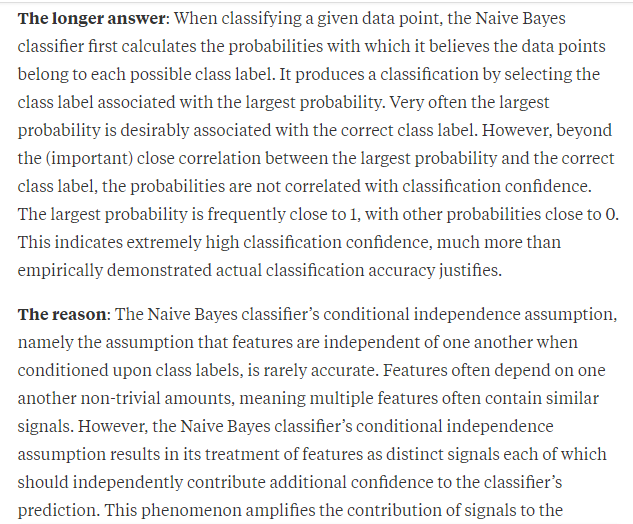
As in case of KNN space complexity was quite large whereas here in Naïve Bayes it is faster and space and time complexity is also very low at run time. Less the value of d and c less the space complexity.

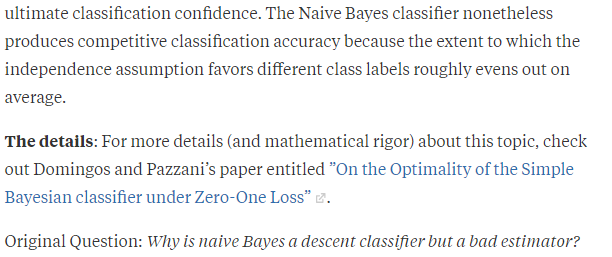
Comments:





If not understood clearly check below comments from quora about last question.





QUES: KNN doesn’t have any training phase , how?

