# Naive Bayes

Given an example input **x**, the probability of it belonging to class c is given to be



where  and , i.e., there are *K* classes in total. In Bayesian inference framework, we will assign the example into the class which maximizes the posterior, that is



In , the evidence  is independent of the class . Therefore, we can transform the optimization problem into



In a Naive Bayes Classifier (NBC), the basic assumption is conditional independency. That it, in each class , we assume the *m* features of the input are independent from each other, which implies



where is the value of the feature of one example . Now the classification result of a NBC is



The remaining problem is to determine the class prior  and the conditional distribution of each feature .

## Estimation of

In this problem, there are only two classes, for spam and or for non-spam. Therefore, it is reasonable to assume a Bernoulli distribution as the class prior, i.e., , where is probability of being a spam . Since there are enough examples in the training set, for simplicity, we just use a maximum likelihood method to estimate the parameter , which turns out to be



where and are the number of spam and non-spam mails in the training set respectively.

## Estimation of

Since our data has been binarized, each feature can have a value of 0 or 1. Supposing the probability of feature in class *c* being 1 is , i.e., , we know that each feature in each class in Bernoulli distributed: . The remaining work is to infer the parameter value with the given training set .

As the parameter is only concerned with the class , we extract a subset from with the examples labeled in class , that is



In this case, we will take the Bayesian methods to infer the unknown parameter with the Maximum A Posteriori estimation. In this framework, the unknown parameter will also be considered as a random variable with a prior probability , where because is the parameter of the above Bernoulli distribution. The posterior of given the training subset is



We assume the prior of is a Beta distribution, the conjugate prior of the Bernoulli distribution. Then we have



where beta function, *B*, is a normalization constant to ensure that the total probability integrates to 1. In practice, we should choose the two shape parameters and according to our prior knowledge. Then, for  in , if the examples in are independently identically distributed, then  is a binomial distribution with respect to the binary value of the feature. We count the number of examples in according to the feature value as follows:



Then, the likelihood  in is



The MAP estimation from is



Applying a logarithmic transformation on yields



In the prior  is parameterized by and and the likelihood is given in . Now we can insert and into the optimization objective , which leads to



Note that the last term of the above two equations in are both constants which are independent of . Therefore, combining and , the MAP estimation is now turned into



The derivative of with respect to is



By making the derivative zero, we get the maximizer to be



where is size of the subset .

In fact, from and we can also see that



which indicates the posterior of is still Beta distributed with two updated shape parameters compared with its prior. This stems from the fact that the prior we have chosen, Beta distribution, is exactly the conjugate distribution of Bernoulli and binomial distribution. To maximize the posterior in , the solution is the unique mode of Beta distribution, that is



which just leads to the same answer as .

## Bayesian testing

In the above sections 1.2 and 1.3, we have estimated the prior and the likelihood from the training set *D* based on the conditional independency assumption of a Naive Bayesian Classifier. As we have shown in equation , now that we have got  and , given a test example , its class label is predicted by the Naive Bayesian Classifier to be



which maximizes the posterior probability . The class prior and the feature likelihood in each class are



In practice, to avoid numeric issues such as underflow, we compute the logarithmic transform of the objective function in instead, that is,



# Gaussian Naive Bayes

In this question, the data are not binarized. Therefore, to build a Naive Bayes Classifier, we assume the continuous feature data are Gaussian distributed conditioned on each class, i.e.,



We use the maximum likelihood method to estimate both the class prior and the conditional feature distribution ). The estimation of is still shown in . Next, we will derive the MLE for Gaussian distribution.

## MLE of Gaussian distribution

For notational simplicity, consider *N* scalar observations  stemming from a Gaussian distribution . Assume these observations are mutually independent. Then the log-likelihood function for the *N* observations is



According to , we obtain



Inserting into leads to



whose derivative with respect to is



Equating to zero gives



Next, to facilitate the computation, let  and take the derivative of with respect to *s*. We get



By making , we can obtain the maximum likelihood estimation of the variance is



In conclusion, we can see that maximum likelihood estimation of the Gaussian distribution parameters are exactly the corresponding sample mean and sample variance.

Let’s come back to our Naive Bayes Classifier problem in . The maximum likelihood estimation for the mean and variance parameters of the class dependent feature Gaussian distribution is given by



where is a subset of examples belonging to class *c* with and is the value of the feature of . Now with the estimated distribution parameters , we can continue the Naive Bayes predication as shown in subsection 1.3.

# Logistic Regression

In Bayesian classification, the posterior probability for class assignment  is estimated via the respective conditional pdfs, which is not, in general, an easy task. On the contrary, the logistic regression method can model such posterior probabilities directly, where the distribution of data is of no interest. Therefore, logistic regression is a discriminative approach.

## Definition

For simplicity, we consider a binary classification problem. There are only two possible classes and the output (the class label) can be 0 or 1. The mode assumes that



where  is the *standard logistic function*. A bias term has been involved in the parameter vector  as the first element and the sample input has been prepended with 1 to allow the bias term, that is, . Here, *m* is the number of features. It is easy to write the posterior probability of an example x being in class 0:



We can also see that



which is known as the *logistic odds*. Supposing there are only two classes, and , the posterior probability for class predication of and can be integrated together as



## Parameter estimation

In logistic regression, given a training set , the parameter vector **w** is estimated via the maximum likelihood method. In the next, for notational conciseness, the logistic function  will be notated by a simple f. Then, the logistic regression model in becomes



The likelihood function for the training set *D* is written as



where . To facilitate the subsequent MLE, we consider the negative log-likelihood function given by



The derivative of the standard logistic function is



which implies



according to the chain rule. Now it is readily to write the gradient of  with respect to **w** as follows



The Hessian matrix of with respect to **w** is



We can express the gradient and Hessian using matrix notation as follows



where



Convexity

optimzation