Lecture Scribe for Machine Learning, CS 60050

Prof Aritra Hazra (January 22, 2021)

1 Summary of Last Lecture

We were trying to obtain a probably approximately correct hypothesis of the unknown function as well as determine the class of an unknown example depending on a new attribute. Even though the Joint Distribution Table has all the combinations of the attributes and their corresponding class probabilities, it will not be provided to us since the number of entries in the table is an exponential function of the number of attributes and there is data sparsity in the natural world.
The conditional independence assumption implies that x is conditionally independent of y given z if the probability of x given y and z is equal to the probability of x given z. The conditional independence assumption helps us in reducing the number of probability estimations (from exponential to linear) required in computing the probability of a new example to belong to a class, given a set of attributes. For N attributes and C classes only O(NC) unique probabilities are needed in the Naive Bayes algorithm. The Bayes rule without conditional independence:

$$P(y = y_k|x_1, x_2....x_n) = \frac{P(y = y_k)P(x_1, x_2....x_n|y = y_k)}{\sum_j P(y = y_j)P(x_1, x_2....x_n|y = y_j)}$$

The Bayes rule with conditional independence:

$$P(y = y_k|x_1, x_2....x_n) = \frac{P(y = y_k)\prod_{i=1}^{n} P(x_i|y = y_k)}{\sum_j P(y = y_j)\prod_{i=1}^{n} P(x_i|y = y_j)}$$

For training the Naive Bayes classifier, the prior probabilities of all classes as well as the conditional probabilities of all the attributes given their classes have to be computed. In order to classify a new example x, the following classification rule may be used:

$$y_{new} = \boldsymbol{argmax_{y_k}} P(y = y_k)\prod_{i=1}^{n} P(x_i^{new}|y = y_k)$$

where

$$x^{new} =< x_1^{new}, x_2^{new}, ..., x_n^{new} >$$

The Naive Bayes Classifier can used in many real life scenarious like spam mail classification and news categorization.

2 Spam filtering

Let us consider an e-mail to have a maximum of 1000 English words. Then we have 1000 attributes. On a regular basis, we use a maximum vocabulary of 50000 English words. So the attribute space is 50000 words. In the Naive Bayes Spam classifier, we make an assumption that all words have independent and identical probability distribution in the Sample Space of all e-mails.
During training of the classifier, we have to estimate the conditional probabilities of all words in the vocabulary given either of the classes "Spam" or "Not Spam". We also need to compute the prior probability of any of the classes "Spam" or "Not Spam". The prior probability of one class is 1 minus the prior probability of the other class. The Naive Bayes algorithm follows the principle of Maximum Likelihood Estimations to estimate the above probabilities.
Then when a new mail arrives, we need to use the conditional probability of all the words in the new mail with respect to both the classes "Spam" and "Not Spam" which have already been estimated during the training phase. Next we can use the classification rule to compute the posterior probabilities of the two classes "Spam" and "Not Spam" given the new e-mail.

$$P(y = Spam|mail^{new}) = P(y = Spam)\prod_{i=1}^{n} P(mail_i^{new}|y = Spam)$$

$$P(y = NotSpam|mail^{new}) = P(y = NotSpam)\prod_{i=1}^{n} P(mail_i^{new}|y = NotSpam)$$

where

$$mail^{new} =< mail_1^{new}, mail_2^{new}, ..., mail_n^{new} >$$

and n is the number of words in the newly arrived e-mail.
If P(y=Spam|mail(new))>P(y=NotSpam|mail(new)), then the new email is classified as a spam mail otherwise it is classified as a non spam mail.

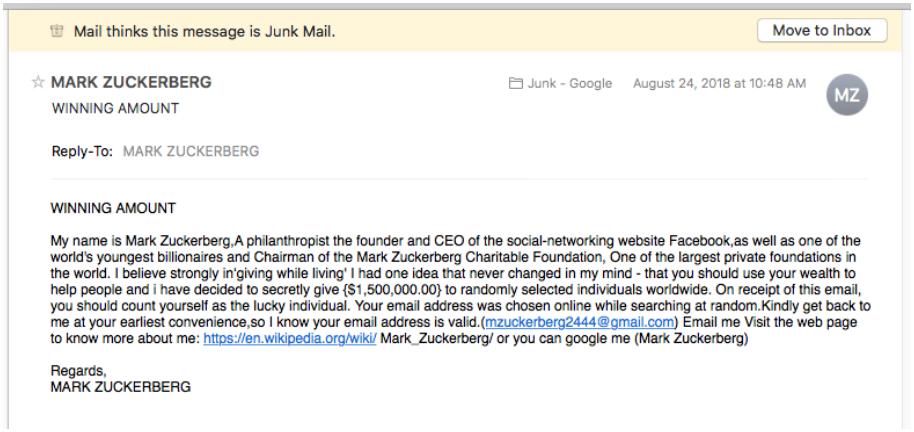

Figure 1: Email Spam example

3 Issues in Naive Bayes

The following sections explain the different issues present in the Naive Bayes algorithm:

i) **Dependence on conditional independence**:
In the spam mail classifier example, we assume that all words are conditionally independent given the class, but this may not be true for all words. For example, the word "am" can only be preceded by the word "I" and not by "You" or any other word. Therefore tha attributes "I" and "am" are not conditionally independent, given either class. When two attributes are dependent, they dampen the overall estimations and can lead us to some sort of mis-estimation. In other words, the more the presence of conditional dependence among word "attributes", the greater the chances of error in classification. Despite this shortcoming, spam filters work reasonably good with some tolerance for error.

ii) **Posterior probability becomes zero because of a single attribute**:
During training the Naive Bayes Classifier, some attribute (say Ai) may be missing from the training dataset examples corresponding to a given class (say Ck) and have conditional probability as zero for that class. If Ai appears in the test dataset, then the posterior probability of Ck becomes zero as the Naive Bayes rule uses the product of the attributes' conditional probabilities to compute the class posterior probabilities. Because of this, the other classes will always be selected for all test examples containing Ai even though the other attributes might point to Ck.

3.1 Formal treatment of MLE and MAP in the context of Naive Bayes

Maximum Likelihood Estimate (MLE) where D is the probability distribution of the training dataset and $|D|$ is the number of examples in the training dataset:

$$\hat{\pi_k} = \hat{P}(y = y_k) = \frac{\#\mathcal{D}\{y = y_k\}}{|\mathcal{D}|}$$

$$\theta_{ijk} = \hat{P}(x_i = x_{ij}|y = y_k) = \frac{\#\mathcal{D}\{x_i = x_{ij} \wedge y = y_k\}}{\#\mathcal{D}\{y = y_k\}}$$

Maximum a Posteriori (MAP) estimate given the prior (assumption) that the distribution is Dirichlet:

$$\hat{\pi_k} = \hat{P}(y = y_k) = \frac{\#\mathcal{D}\{y = y_k\} + (\beta_k - 1)}{|\mathcal{D}| + \sum_m (\beta_m - 1)}$$

$$\theta_{ijk} = \hat{P}(x_i = x_{ij} | y = y_k) = \frac{\#\mathcal{D}\{x_i = x_{ij} \wedge y = y_k\} + (\beta_k - 1)}{\#\mathcal{D}\{y = y_k\} + \sum_m (\beta_m - 1)}$$

MAP estimates can counter the cases when any attribute is absent in the set of training examples for a given class.
Using the above equations, any new example in the test set can be classified as follows:

$$y_{new} = \boldsymbol{argmax_{y_k}} \hat{\pi_k} \pi_{i=1}^{n} \hat{\theta_{ijk}}$$

iii) **can only deal with problems in which both attributes and classes are discrete**:
For example in image processing, we can have an image where the intensity of the pixels(attributes) may be real valued instead of integer valued. This can often occur when images from one colour scheme like Blue,Green,Red (BGR) is converted to another colour scheme like Hue,Saturation,Intensity (HSI). The issue is that we have used Naive Bayes until now only with respect to data that has both discrete attributes and discrete classes.

3.2 use of Gaussian Distribution for datasets having continuous attributes

Gaussian Naive Bayes is another formulation of the Naive Bayes algorithm where the attributes follow the gaussian distribution.
The probability density function for a variable x that follows the gaussian distribution is:

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-1/2(\frac{x-\mu}{\sigma})^2}; \mu : mean; \sigma : standard\ deviation$$

The Gaussain Naive Bayes conditional probability of attribute x(i) given class y(k) is as follows:

$$P(x_i = x | y = y_k) = \frac{1}{\sqrt{2\pi\sigma_{ik}^2}} e^{-1/2(\frac{x-\mu_{ik}}{\sigma_{ik}})^2}$$

$$\mu_{ik} : conditional\ mean; \sigma_{ik} : conditional\ standard\ deviation$$

Here only the conditional means and the conditional standard deviations need to be learned during training instead of learning the probabilities as in the case of discrete attributes. In Gaussian Naive Bayes, sometimes for simplicity we assume that the conditional standard deviation is either equal to the standard deviation of the attribute i for all i or equal to the standard deviation of class k for all k. In order to classify the test examples, we use the classification rule:

$$y^{new} = \boldsymbol{argmax_{y_k}} P(y = y_k) \pi_{i=1}^{n} N(x_i^{new}, \sigma_{ik}, \mu_{ik})$$

The Gaussian Naive Bayes has been used in detecting brain tumour from MRI scan images.

4 Decision Surfaces of different classification algorithms

4.1 Decision Tree Classifier

The Hyper-surface formed by plotting the decision boundaries of a decision tree are sharp and parallel to the axes, where each axis represents an attribute. The partitions formed from the decision surface do not form a one to one mapping with the classes. One class may have multiple partitions at different continuous locations.

4.2 Candidate Elimination algorithm

Since the candidate elimination algorithm does not allow disjunction which is allowed in decision trees, the hyper-surface formed by plotting the decision boundaries in candidiate elimination creates partitions such that the partitions are continuous and form a one to one mapping with the classes, while still being parllel to the axes.

4.3 Naive Bayes Classifier

Given a binary classification problem, the Naive Bayes classifier tries to check the greater of the posterior probabilities of the two classes given the attributes belonging to an example in the test set. Therefore,

$$P(y=1|x_1...x_n) \lessgtr P(y=0|x_1...x_n)$$

$$=> \frac{P(y=1|x_1...x_n)}{P(y=0|x_1...x_n)} \lessgtr 1$$

$$=> \frac{P(y=1)\pi_{i=1}^{n} P(x_i|y=1)}{P(y=0)\pi_{i=1}^{n} P(x_i|y=0)} \lessgtr 1$$

Applying logarithm on both sides of the inequality,

$$ln[\frac{P(y=1)}{P(y=0)}] + \sum_{i=1}^{n} ln[\frac{P(x_i|y=1)}{P(x_i|y=0)}] \lessgtr 0$$

Prove the following statement: If x(i) = 0,1, then the threshold (><0) is a linear function of x(i)'s.
Hints:

i) The first term in the log scale inequality is a constant.

$$ii) P(x_i=0|y=1) = 1 - P(x_i=1|y=1)$$

Because of the above statement, the Naive Bayes Classifier is also called the log linear classifier since the examples plotted in the log scale make the hypothesis space a hyperplane.

4.4 Gaussian Naive Bayes Classifier

An example of a problem which can be solved using the Gaussian Naive Bayes Classifier: P(Student is good in sports | Height, Marks in ML Exam). The attributes (Height and Marks in ML Exam) are real valued. The conditional probabilities of all the attributes with respect to the classes (good in sports and bad in sports) follow the gaussian probabilty distribution. The probability distribution surface is a 3D surface but it can be visualized in 2 dimensions using contours.
We can plot the gaussian distributions of the conditional probabilities of the students' ML marks along the y axis given the two classes. Similarly, we can plot the gaussian distributions of the conditional probabilites of the students' height along the x axis given the two classes. Then we can join the peaks of the gaussians using the third dimension or via contours, thereby obtaining the products of the conditional probabilities. It has been seen that the line of intersection of the hills forming the contours is straight if the hills have the same standard deviation, thus forming a linear decision boundary. The line of intersection of the hills forming the contours is curved if the hills have different standard deviations, thus making the decision boundary nonlinear.

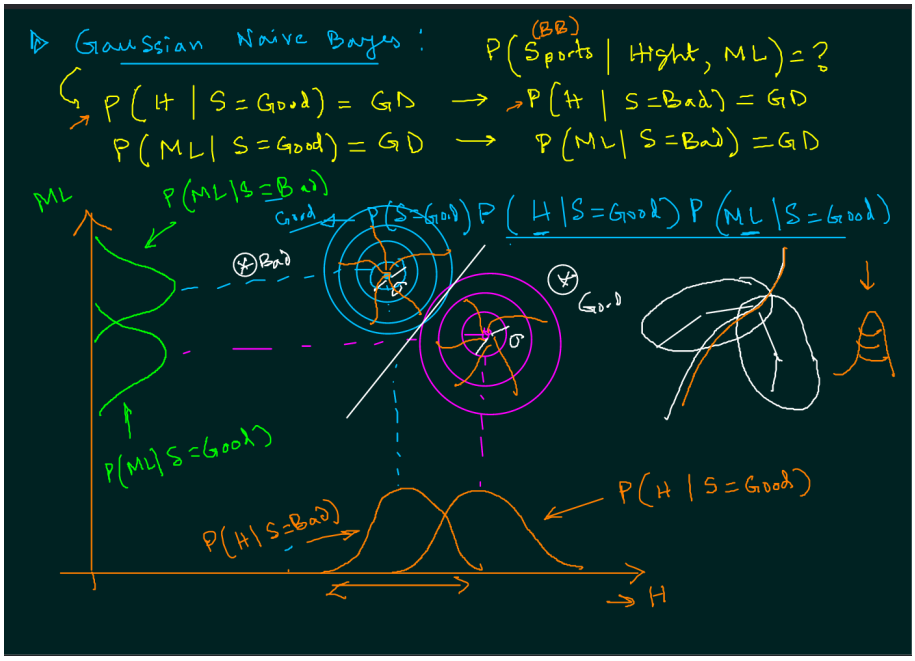

Figure 2: Gaussian Naive Bayes Classifier visualization

Prepared by: Dibya Kanti Haldar (20CD92R01)