Introduction:

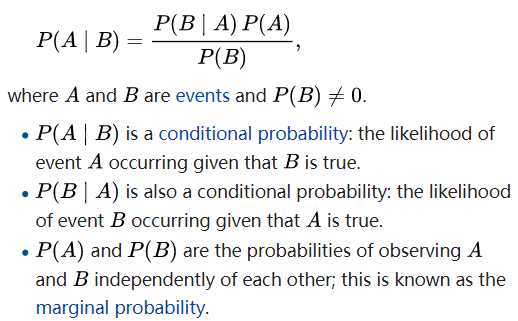
Nowadays, with the develop of the informatization, we are more convenient to get and send messages to others. However, the information explosion also brings us the outstanding disadvantage that anyone can achieve us and send the messages we are not interested in. The most common annoying problem is the spam mail. Therefore, the Naïve Bayesian Classification has been put forward to filter those spam mails based on Bayesian probability. Bayesian probability is often used in text classification for it can predict the class membership probability that a given sample belongs to a particular class.

方法解释（数学解释，算法解释）

Introductions:

Bayesian classifier is based on Bayes' theorem which has a long history. And a large number of researches has proved its surprisingly effective in txt classification. It describes the probability of an event, based on prior knowledgr of conditions that might releted to the event.

Bayes' theorem is stated mathematically as the following equation:[2]



And in the perspective of machine learning, we can take B as having a certain characteristic and take A as belonging to a specific classification. Therefore, Bayes’s theorem comes as the following expression:



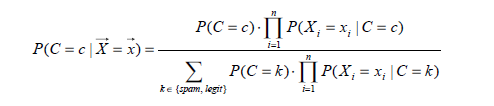
For further explanation, we consider  as the posterior probability and which predict the probability that the sample belongs to a certain class.

And  is the probability that the sample with some certain characteristic belongs to a certain class.

is the probability that the sample belongs to a certain class without knowing if it has a certain characteristic, which is called as the priori probability.

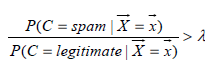
The last expression  describes the probability of the sample has a certain characteristic without knowing if it belongs to the class.

When considering filtering the spam emails, it is easy to estimate the , for you can calculate the probability of the appearance in such a training set as the collection of all the emails. However, it is difficult to calculate the without simplifying assumptions in practice. Because there are too many possible values of characteristics such as some common statements in spam email. Therefore, based on conditionally independent, assuming X1,…Xn are those expressions belonging to **X, we can make simplyfications which yields:**



In this formula, it is easy to estimate the P(Xi|C) and P(C). We can calculate P(Xi|C) by counting the frequency of Xi statements in training set as a collection of spam emails. And we can easily get the P(C) by counting the frequency of the spam mail in the collection of both the spam emails and the legitimate ones.

After obtain the probability of P(C=c|X=x), we need to set a criterion for judging whether it is a spam email or a legitimate one. When taking decision-theoretic notion of cost into consideration, we could make the reasonable assumptions that P(C=spam|X=x) is more costly than the P(C=legitimate|X=x). Because it is more severe to take legitimate emails as a spam one than taking the spam as the legitimate one, for we don’t want to miss any important information in legitimate email. Therefore, we can draw the judgement criterion as follows:



Furthermore, we could reasonably draw that  . Therefore, the email can be classified into a spam if it meets the following classification criterion:



And according to some researches done before, there are two rational scenarios to choose the value t and λ. The first scenario is done by Sahami et al. 1998, he set t to be 0.999, in which he took λ as 999. It corresponds to the scenario that the cost of taking legitimate emails as a spam one is equal to taking 999 spam emails as the legitimate ones. For the users who feels painful about losing the legitimate email, it is reasonable. The second senario is that, if we allocate the space to receive and store the spam email other than blocking them out from our mailboxes, it will be less severe to take a legitimate email to be a spam email, for we still can check them in our mailbox. Thus, it is feasible for us to set λ to be 1. So, when P(C=spam|X=x) is more than 0.5, we can believe it is a spam email.

Actually, there are many other special but important cases that we need to take notice of. However, many of them get involved with another important subject in AI &Linguistics called NLP (Neuro-Linguistic Programming). NLP is an area of computer science and artificial intelligence concerned with the interactions between computers and human (natural) languages, in particular how to program computers to fruitfully process large amounts of natural language data. In this section, I only want to consider one simple but common scenario. Sometimes, we include a specific x in P(C=spam|X=x), but when scanning the whole training set, we may find it doesn’t appear in our training set, then P(C=spam|X=x)=0. Therefore, we may draw the expression P(C=spam |X=x)=0. After finishing the classification, we may find that the spam email with expression X will get escaped from the filter into our mailbox, for the probability to think it as a spam email is 0 less than the criterion previously set. For such occasion, we use Laplace Smoothing method which is also called to be plus one smoothing. In Laplace Smoothing method, for those P(C=spam |X=x)=0 because of the 0 frequency of xi, we add the count of appearance to be 1, which will effectively avoid the problem.

For further training process, there will be new expressions adding into the spam. And the Naïve Bayesian Classification will reveal its advantage that it will sensitively give spam expressions new weights and train it again and again based on the reliable statistical methods.

伪代码

Email segement:

def textParse(bigString):

import re #导入正则表达式的库

listOfTokens=re.split(r'\W\*',bigString) #返回列表

return [tok.lower() for tok in listOfTokens if len(tok)>2]

Generate vocabulary:

def createVocabList(dataSet):

vocabSet=set([])

for docment in dataSet:

vocabSet=vocabSet| set(docment) #union of tow sets

return list(vocabSet) #convet if to list

Generate word vector

分析

* 优点

1. 对待预测样本进行预测，**过程简单速度快**(想想邮件分类的问题，预测就是分词后进行概率乘积，在log域直接做加法更快)。
2. **对于多分类问题也同样很有效**，复杂度也不会有大程度上升。
3. **在分布独立这个假设成立的情况下**，贝叶斯分类器**效果奇好**，会略胜于逻辑回归，同时我们**需要的样本量也更少一点**。
4. 对于类别类的输入特征变量，效果非常好。对于数值型变量特征，我们是默认它符合正态分布的。

* 缺点

1. 对于测试集中的一个类别变量特征，如果在训练集里没见过，直接算的话概率就是0了，预测功能就失效了。当然，我们前面的文章提过我们有一种技术叫做**『平滑』操作**，可以缓解这个问题，最常见的平滑技术是拉普拉斯估测。
2. 那个…咳咳，朴素贝叶斯算出的概率结果，比较大小还凑合，实际物理含义…恩，别太当真。
3. 朴素贝叶斯有分布独立的假设前提，而**现实生活中这些predictor很难是完全独立的**。

应用

结论

未来展望

reference