An Introduction to Bayes theorem

Inspired by machinelearningmastery blog on Bayes theorem

It provides a principled way of calculating conditional probability without using joint probability.

- Marginal probability

  Probability of an event that is independent of other random variables.
  E.g. P(A)

- Joint probability

  Probability of two(or more) simultaneous events from two(or more) respective dependent random variable.
  E.g. P(A and B)==P(A,B)

- Conditional probability

  Probability of an event given the occurrence of another event.
  E.g. P(A/B)

  Joint probability using conditional probability
  P(A and B)=P(A/B)*P(B)

  Conditional probability using joint probability
  P(A/B)=P(A and B)/P(B)

  Alternate way of calculating conditional probability
  P(A/B)=(P(B/A)*P(A))/P(B)

  This is done either when joint probability is difficult to calculate or when reverse conditional probability is available or easy to calculate.

We can write P(B) as,
P(B)=P(B/A)*P(A)+P(B/not A)*P(not A)
Where,
P(not A)=1-P(A) and
P(B/not A)=1-P(not B/not A)

This alternate form of calculating conditional probability is called **Bayes' rule** or **Bayes' theorem**.

Intuitive meaning of terms in Bayes theorem

What is the probability that there is fire given that there is smoke?

P(fire/smoke)=[P(smoke/fire)*P(fire)]/P(smoke)
Where,
P(fire/smoke)=posterior probability
P(smoke/fire):likelihood
P(fire)=prior probability
P(smoke)=evidence

Bayes theorem applied to a medical diagnostic test

Take medical diagnostic test as a binary classification problem where the predictions are made by the *diagnostic test(model)* and the classes are *cancer* and *non-cancer*.

Defining this classification using confusion matrix.

	Positive class(PC) (cancer=True)	Negative class(NC) (non-cancer=False)
Positive prediction(PP) (test=positive)	True Positive(TP)	False Positive(FP)
Negative prediction(NP) (test=negative)	False Negative(FN)	True Negative(TN)

P(test=positive/cancer=true)=TP/(TP+FN)=TPR
P(test=positive/cancer=false)=FP/(FP+TN)=FPR
P(test=negative/cancer=true)=FN/(TP+FN)=FNR
P(test=negative/cancer=false)=TN/(FP+TN)=TNR

Intuitively,the denominator takes on the sum of either the positive class column probabilities or the negative class column probabilities of the confusion matrix as per the case.

Vanilla python code for Bayes with explanation

Bayes Theorem implementation for a Cancer diagnostic test

```
def bayes(p_b_given_a,p_a,p_b_given_not_a):
    p_b=(p_b_given_a*p_a)+(p_b_given_not_a*(1-p_a))    #p_b is the probability of test being positive
    p_a_given_b=(p_b_given_a*p_a)/p_b
    return p_a_given_b
```

```
#calculate the probability that a randomly selected patient has cancer given that the test diagnosed positive

#That is,calculate P(cancer=True/test=positive)==p_a_given_b==posterior probability

#We have,

p_b_given_a=0.85                                         #sensitivity=P(test=positive/cancer=true)
p_a=0.0002                                               #P(cancer=true)==prior probability
p_b_given_not_a=0.05                                     #P(test=positive/cancer=false)

p_a_given_b=bayes(p_b_given_a,p_a,p_b_given_not_a)

print('P(cancer=True/test=positive)= %.3f%%'%(p_a_given_b*100))
```

```
P(cancer=True/test=positive)= 0.339%
```

Here,*posterior probability* is defined as the probability that a randomly selected patient has cancer given the test diagnosed him as positive(has cancer),that is P(cancer=true/test=positive).

By Bayes theorem,

P(cancer=true/test=positive)

=

[P(test=positive/cancer=true)*P(cancer=true)]/P(test=positive)

=

(TPR*PC)/PP

This turns out to be equal to the precision of the given confusion matrix.

Precision=TP/(TP+FP)

Here,a question arises:

Why not directly calculate precision rather than applying Bayes' theorem?

This is because we don't have the confusion matrix for a population of people both with and without cancer that have been tested and have not been tested. Instead, all we have is some priors and probabilities about our population and our test.

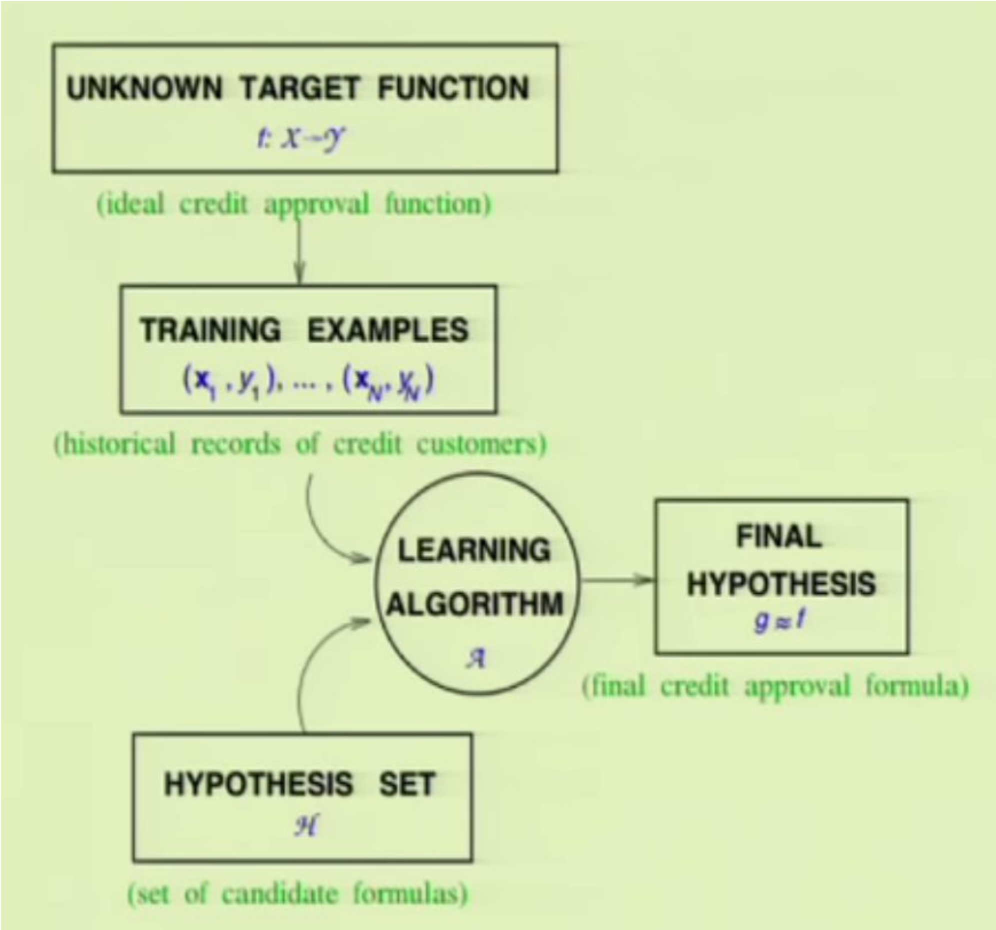
Applications of Bayes' theorem

- Modelling Hypotheses

  Machine learning model is a hypothesis function that is trained with several examples to approach an ideal target function that maps input and output correctly.

  Essence of machine learning

  Let me show you an image that broadly describes the components of learning.



Source:[Introductory machine learning course by Yaser Abu-Mostafa,Caltech]

The image represents how a learning algorithm choose one hypothesis from a set of hypothesis functions given the training examples.

It represents a *credit approval problem* wherein historical records of credit customers behave as training data and the algorithm chooses the hypothesis function that is most consistent with the unknown target function.

This target function perfectly maps input with output and we are trying to get a hypothesis that closely approximates this target function.

Bayes' theorem helps find out the probability of each hypothesis holding given the data.

P(h/D)=(P(D/h)*P(h))/P(D)

Breaking the above equation down, it says that the probability of a given hypothesis holding or being true given some observed data can be calculated as the probability of observing the data given the hypothesis multiplied by the probability of the hypothesis being true regardless of the data, divided by the probability of observing the data regardless of the hypothesis.

Our task is to find the maximum probable hypothesis given the data which is termed as Maximum a Posteriori(MAP).We can determine the MAP hypothesis by using Bayes theorem to calculate the posterior probability of each candidate hypothesis.

Fitting algorithms like linear regression for predicting a numerical value, and logistic regression for binary classification can be framed and solved under the MAP probabilistic framework.

- Classification

Classification is the predictive modelling of assigning a class to the input data. Bayes' theorem can be used to find the probability of each class given the input data.

$$P(class/data)=[P(data/class)*P(class)]/P(data)$$

Where P(class|data) is the probability of class given the input data.

This calculation can be performed for each class in the problem and the class that is assigned the largest probability can be selected and assigned to the input data.

However,full Bayes' theorem cannot be applied for classification unless we have enough data to try different combination of values on the data given a class to calculate P(data/class).As such, the direct application of Bayes Theorem also becomes intractable, especially as the number of variables or features (n) increases.