Relationship between Least Squares Approximation and Maximum Likelihood Hypotheses

Steven Bergner*, Chris Demwell†
Lecture notes for Cmpt 882 — Machine Learning

February 19, 2004

Abstract

In these notes, a derivation of the estimate of the mean of a normal distribution forms a foundation for the discussion of the least-squares estimate as an example of the class of maximum-likelihood estimates. Finally, the Naive Bayesian Classifier is introduced.

1 Overview

This lecture is set in the context of a larger discussion; our first step is to recall that context. Thus, these notes are structured to cover the following aspects:

1. Overview of using Bayesian Statistics in machine learning
2. Learning the mean of a Gaussian distribution
3. Learning real-valued functions
4. Naive Bayes classifier

The major parts of the lecture are contained in § 2, § 3, and § 4.

The goal of this lecture is to provide an overview of various machine learning techniques. The lecture is accompanied by the book by Mitchell [2]. Most of the insights and explanations given in these notes are derived from this source, and through the teaching of Dr. Oliver Schulte of Simon Fraser University. Another valuable source of information for our summary was Tom Mitchell's online slides for the book [1].

1.1 Bayes Learning

Bayesian Learning directly manipulates probabilities, given the data observed as well as prior probabilities. It is usually concerned with the posterior probabilities of hypotheses being true. There are several approaches that we can take to distinguish between hypotheses, summarized in the following figure:

*email: sbergner@cs.sfu.cs
†email: cdemwell@cs.sfu.ca

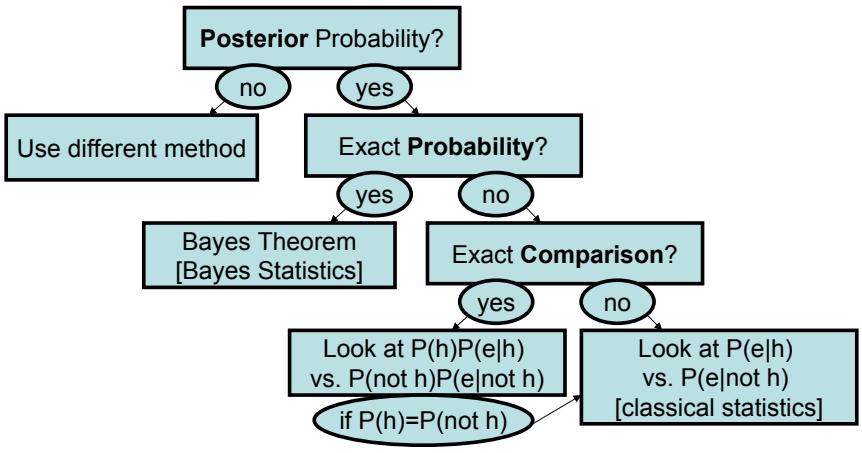

Figure 1: Various Bayesian approaches appropriate to differing requirements

If we are not concerned with posterior probabilities, then we need not use Bayesian methods and can use some frequentist[1] approach. Next, if we are concerned with exact posterior probability values, we use Bayesian statistics to find it. Otherwise, if we are interested in an exact comparison between two posterior probabilities, then we compare them; if we are not concerned with an exact comparison, we can discard the prior probabilities and work only with the conditional probabilities of the data given the hypothesis. Note that this is the same as the frequentist, or classical approach. The approaches we look at in the following notes fall into this last category. Nevertheless, they can be shown to correspond to Bayesian statistics under certain assumptions.

1.2 Quick Review of Basic Bayesian Statistics

All algorithms we are considering to be Bayesian learners are based on Bayes' theorem:

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}. \tag{1}$$

This equation relates the following probabilities:

- *posterior probability* $P(h|D)$ of hypothesis h to hold given the data D,
- *likelihood* $P(D|h)$ of observing data D given hypothesis h.
- *prior probabilities* $P(D)$ and $P(h)$ of encountering data D or respectively hypothesis h.

When we use this relationship for learning a concept we usually look for the hypothesis with the highest *maximum a posteriori* probability (MAP) for the given data. That simply means that we're looking to get the largest posterior probability for the data given the hypothesis:

$$\begin{aligned} h_{MAP} &= \underset{h \in H}{\operatorname{argmax}} P(h|D) \\ &= \underset{h \in H}{\operatorname{argmax}} \frac{P(D|h)P(h)}{P(D)} \end{aligned} \tag{2}$$

[1] In other words: Classical, not Bayesian. Concerned only with observable data, not prior probabilities.

When comparing different hypotheses h that are all derived from the same data D we can drop the denominator $P(D)$. This leads to the last form of Eq. 2:

$$h_{MAP} = \underset{h \in H}{\operatorname{argmax}} P(D|h)P(h) \tag{3}$$

It is possible to further simplify the above search problem by assuming that *all hypotheses are equally probable* (all $P(h_i)$ are equal and sum to one). Incorporating this into Eq. 3 yields the so-called *maximum likelihood* hypothesis:

$$h_{ML} = \underset{h \in H}{\operatorname{argmax}} P(D|h). \tag{4}$$

Thus, the maximum likelihood hypothesis h_{ML} can be understood as a special case of maximum posterior hypothesis h_{MAP} given all hypotheses are equally probable.

2 Maximum Likelihood Estimator

A good way to get an understanding of a new learning algorithm is to compare it to other ones we have already looked at before. This helps to reveal specific differences and similarities between the approaches. In a previous lecture we have already investigated the properties of FIND-S and CANDIDATE-ELIMINATION from a Bayesian perspective.[2]

Maximum likelihood estimation is a special case of Bayesian learning. Here we will consider a very simple setting to get an impression of the basic properties of this technique. The insights we get from this analysis will help us to arrive at a more general statement in the next section. Our current problem has the following properties:

- The data is given as a set of values $D = \{d_i\}$ that are drawn from a normally distributed source $\mathcal{N}(\mu, \sigma)$,
- The hypothesis space contains only normal distributions $\mathcal{N}(\mu, \sigma)$ of fixed standard deviation σ; the different μ we are taking into account as hypotheses are all equally probable,
- The samples are assumed to be taken in an independent and identically distributed manner (which we abbreviate `iid`).

The normal distribution, also known as the Gaussian distribution, is completely characterized by its mean μ and the standard deviation σ:

$$P(X = x) = \mathcal{N}(\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \tag{5}$$

Eq. 5 is the probability density function of the distribution. A plot of a normalized version of it is shown in Fig. 2. Due to the central limit theorem we can approximate the sum of many independent random variables by a normal distribution.[3] We claim without proof that this demonstrates the validity of deriving general claims from the discussion of only this distribution.

Let us imagine that we have performed some experiment - perhaps a survey, or a physics experiment. We have a number of sample of data D, and we wish to use it to characterize the result of the experiment.

[2] The major difference was in choosing priors $P(h)$ for the hypotheses.

[3] See `http://en.wikipedia.org/wiki/Central_limit_theorem` or `http://mathworld.wolfram.com/CentralLimitTheorem.html`.

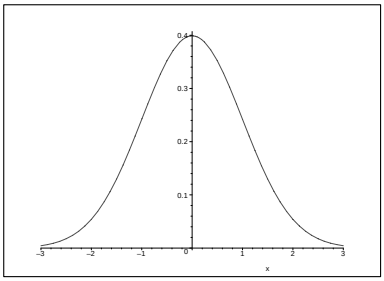

Figure 2: Gaussian distribution, normalized, with $\mu = 0$ and $\sigma = 1$ (also referred to as the standard normal distribution).

A very frequent, though often inadequate[4] way that people do that is to find the mean of the underlyind distribution. Said another way, looking at the data D we are now interested in finding the most likely value of μ:

$$P(D|\mathcal{N}(\mu,\sigma)) = \prod_{i=1}^{m} P(d_i|\mathcal{N}(\mu,\sigma)). \tag{6}$$

Assuming a fixed σ and taking the unknown μ as the hypothesis we are looking for we can write instead

$$P(D|\mu) = \prod_{i=1}^{m} P(d_i|\mu). \tag{7}$$

We can compute the probability of the hypothesis as a product because of the independence of the samples. Furthermore, since we are interested in maximizing the probability of our hypothesis, we can also consider the logarithm of the probability instead. This is a monotonous and well defined mapping for values > 0. Taking all this into account the maximum likelihood hypothesis is determined as follows:

$$\begin{aligned} h_{ML} &= \operatorname*{argmax}_{h\in H} P(D|\mathcal{N}(\mu,\sigma)) \\ h_{ML} &= \operatorname*{argmax}_{\mu} P(D|\mu) \\ &= \operatorname*{argmax}_{\mu} \prod_{i=1}^{m} P(d_i|\mu). \end{aligned} \tag{8}$$

Maximizing the logarithm of the probability is equivalent and turns the product into a sum

$$\begin{aligned} h_{ML} &= \operatorname*{argmax}_{\mu} \ln \prod_{i=1}^{m} P(d_i|\mu) \\ &= \operatorname*{argmax}_{\mu} \sum_{i=1}^{m} \ln P(d_i|\mu) \\ &= \operatorname*{argmax}_{\mu} \sum_{i=1}^{m} \ln \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{d_i-\mu)}{\sigma}\right)^2}. \end{aligned}$$

[4]The mean provides us with a simple measure of central tendency, that is very suceptible to outlying data and provides no information about the shape of the distribution it describes. Unfortunately, it is frequently held up as a very descriptive measure to those who might be confused by more detailed statistics

For fixed σ we can drop the first factor in the expression. It does not affect the maximization because it is constant and positive. In addition to that the natural logarithm and the exponential function cancel each other out.

$$\begin{aligned} h_{ML} &= \underset{\mu}{\operatorname{argmax}} \sum_{i=1}^{m} \ln e^{-\frac{1}{2}\left(\frac{d_i-\mu)}{\sigma}\right)^2} \\ &= \underset{\mu}{\operatorname{argmax}} \sum_{i=1}^{m} -\frac{1}{2}\left(\frac{d_i-\mu)}{\sigma}\right)^2 \\ &= \underset{\mu}{\operatorname{argmax}} \sum_{i=1}^{m} -\frac{1}{2\sigma^2}(d_i-\mu)^2 \\ &= \underset{\mu}{\operatorname{argmin}} \sum_{i=1}^{m} (d_i-\mu)^2 \end{aligned} \tag{9}$$

In the last step we have removed the constant negative factor changing the problem to become a minimization. We can now take the derivative of the target function with respect to μ and set it to zero.

$$\begin{aligned} \Rightarrow 0 &= \sum_{i=1}^{m} -2(d_i-\mu) \\ 0 &= -\sum_{i=1}^{m} d_i\mu + m\mu \\ \mu &= \frac{1}{m}\sum_{i=1}^{m} d_i \end{aligned} \tag{10}$$

This is also called the empirical average of the distribution D. *Theorem:* The maximum likelihood estimate of the mean μ of a Gaussian distribution X is the empirical average $\bar{X}$.

3 Approximating real valued functions

A common technique used when fitting an expected function to a set of data samples is to *minimize the squared error* between the approximating function and the given data. In this section we are going to show that this is equivalent to finding a maximum likelihood hypothesis under certain assumptions.

In the previous section we have considered a Gaussian distributed set of values. Now we are looking at a real valued function $f(x)$ instead. Our data are samples of this function disturbed by noise. Thus, one sample can be modeled as $d_i = f(x_i) + e_i$. The noise e_i is $\mathcal{N}(0, \sigma)$ distributed and independent between samples. It is also possible to think of the noise as being caused by influences that were not taken into account when forming the hypothesis space. In that case we interpret the noise as 'things we don't yet understand'. Of course we still have to assume that these 'things' have a random behavior similar to $\mathcal{N}(0, \sigma)$. Fig. 3 illustrates the samples d_i as thick dots that are distributed around an actual (concept) function f. The learned hypothesis h_{ML} should converge to the correct concept as the number of given samples is increased.

The derivation of the maximum likelihood hypothesis for this setting is very similar to the one we already did in § 2. What we did there can be understood as learning a constant function $f(x_i) = \mu$.

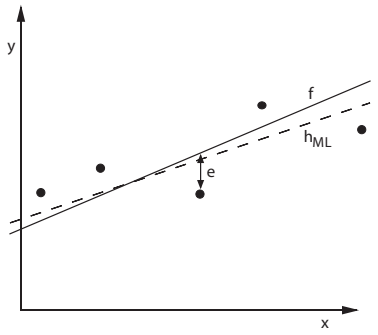

Figure 3: Gaussian disturbed samples drawn from an underlying (linear) function f. The dotted line shows a least squared error approximation that is obtained from a maximum likelihood hypothesis. (Image is taken from [1]).

This is now substituted with a more general $\mu_i = h(x_i) = f(x_i)$ for each data value d_i. Substituting the constant hypothesis μ by this new $h(x_i)$ in Eq. 9 yields

$$\begin{aligned} h_{ML} &= \underset{h \in H}{\operatorname{argmax}} P(D|h(x_i)) \\ &\vdots \\ h_{ML} &= \underset{h \in H}{\operatorname{argmin}} \sum_{i=1}^{m} (d_i - h(x_i))^2 \end{aligned} \tag{11}$$

The omitting dots in this derivation stand for the steps between Eq. 8 and Eq. 9. But the substitution of μ by $h(x_i)$ is really all that happens. The rest remains the same. The result in Eq. 11 is the least squared error approximation for our function that we were looking for. All that took us there were a hypothesis function h and the assumption of Gaussian distributed noise. This setting reduces to the squared difference that is found in the exponent of the Gaussian distribution.

Let us briefly recall what we have started with. All possible functions $h(x_i)$ that we can fit to the data are equally probable. For that reason it was enough to look at the likelihood of h given the data D. Assuming Gaussian noise in our sampling we end up minimizing the log-likelihood of the Gaussian noise. This turns out to be equivalent to minimizing the sum of squared differences between $h(x_i)$ and d_i with respect to different hypothesis functions h. Therefore we can state that least squares approximation can be seen as finding a maximum likelihood hypothesis given the above assumptions.

4 Naive Bayes Classifier

Now let us consider the learning task as the discovery of a target function $f\colon X \rightarrow V$ where each $x \in X$ is described by attributes:

$$x = \langle a_1, a_2, \ldots, a_n \rangle \tag{12}$$

Now given the foundation we've talked about above, we can describe the *Maximum A Priori* (MAP) value of $f(x)$:

$$\begin{aligned} v_{MAP} &= \underset{v_j \in V}{\operatorname{argmax}} P(v_j|a_1, a_2, \ldots, a_n) \\ &= \underset{v_j \in V}{\operatorname{argmax}} \frac{P(a_1, a_2, \ldots, a_n|v_j)P(v_j)}{P(a_1, a_2, \ldots, a_n)} \\ &\quad \text{and if we assume all hypotheses are equally likely, we have} \\ &= \underset{v_j \in V}{\operatorname{argmax}} P(a_1, a_2, \ldots, a_n|v_j)P(v_j) \end{aligned} \tag{13}$$

Equation 13 suggests that we are looking for the hypothesis that maximizes the posterior probability of the hypothesis given the data. Calculating this for any real problem would be intractable, as it would take exponential time in the hypothesis size! Clearly we need some kind of shortcut, so we make a simplifying assumption, called the *Naive Bayes assumption*:

$$P(a_1, a_2, \ldots, a_n|v_j) = \prod_i P(a_i|v_j) \tag{14}$$

In other words, that all attributes are independent of each other, and dependent only on the class (v_j) they are in. For any practical measurement, this assumption is preposterous: we cannot presume, for example, that a lake's temperature is independent of the season - they clearly are dependent! However, it has been demonstrated that *the assumption does not significantly affect the classifier's results.* If we do not need to estimate the actual posterior probability, but only need to decide which class to place a sample in, we are therefore justified in making this simplification which results in the Naive Bayes Classifier:

$$v_{NB} = \underset{v_j \in V}{\operatorname{argmax}} P(v_j) \prod_i P(a_i|v_j) \tag{15}$$

This results in a logarithmic improvement in time complexity, resulting in a linear-time algorithm. The Naive Bayes Classifier is one of the most practical classifiers available. It has been shown, under some circumstances, to be as practical to use as decision trees, neural networks, and nearest-neighbour learning. It is reasonable to use it when one has a moderate or large body of training data available, and when the data attributes are largely independent of each other.

In future classes, we will explore the Naive Bayes Classifier in further depth, and we will consider the case of text classification using the Naive Bayes Classifier. This is commonly done in text indexing and spam detection software.

References

[1] Tom M. Mitchell. Machine learning — Online slides. `http://www-2.cs.cmu.edu/~tom/mlbook-chapter-slides.html`.

[2] Tom M. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.