L6: Parameter estimation

Introduction

Parameter estimation

Maximum likelihood

Bayesian estimation

Numerical examples

In previous lectures we showed how to build classifiers when the underlying densities are known

- Bayesian Decision Theory introduced the general formulation
- Quadratic classifiers covered the special case of unimodal Gaussian data

In most situations, however, the true distributions are unknown and must be estimated from data

- Two approaches are commonplace
  - Parameter Estimation (this lecture)
  - Non-parametric Density Estimation (the next two lectures)

Parameter estimation

- Assume a particular form for the density (e.g. Gaussian), so only the parameters (e.g., mean and variance) need to be estimated
  - Maximum Likelihood
  - Bayesian Estimation

Non-parametric density estimation

- Assume NO knowledge about the density
  - Kernel Density Estimation
  - Nearest Neighbor Rule

ML vs. Bayesian parameter estimation

Maximum Likelihood

- The parameters are assumed to be FIXED but unknown
- The ML solution seeks the solution that "best" explains the dataset X

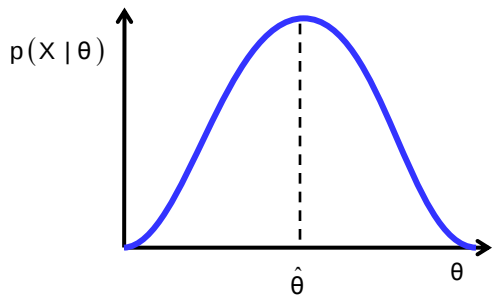
$$\hat{\theta} = argmax[p(X|\theta)]$$

Bayesian estimation

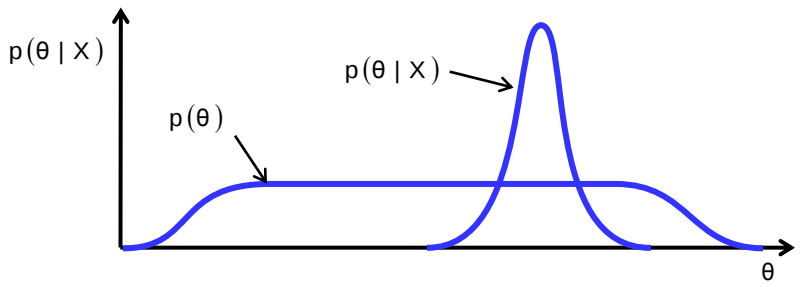
- Parameters are assumed to be random variables with some (assumed) known a priori distribution
- Bayesian methods seeks to estimate the posterior density $p(\theta|X)$
- The final density $p(x|X)$ is obtained by integrating out the parameters

$$p(x|X) = \int p(x|\theta)p(\theta|X)d\theta$$

Maximum Likelihood



Bayesian



Maximum Likelihood

Problem definition

- Assume we seek to estimate a density $p(x)$ that is known to depends on a number of parameters $\theta = [\theta_1, \theta_2, \dots \theta_M]^T$
  - For a Gaussian pdf, $\theta_1 = \mu$, $\theta_2 = \sigma$ and $p(x) = N(\mu, \sigma)$
  - To make the dependence explicit, we write $p(x|\theta)$
- Assume we have dataset $X = \{x^{(1}, x^{(2}, \dots x^{(N}\}$ drawn independently from the distribution $p(x|\theta)$ (an i.i.d. set)
  - Then we can write

$$p(X|\theta) = \Pi_{k=1}^{N} p\left(x^{(k}|\theta\right)$$

  - The ML estimate of θ is the value that maximizes the likelihood $p(X|\theta)$

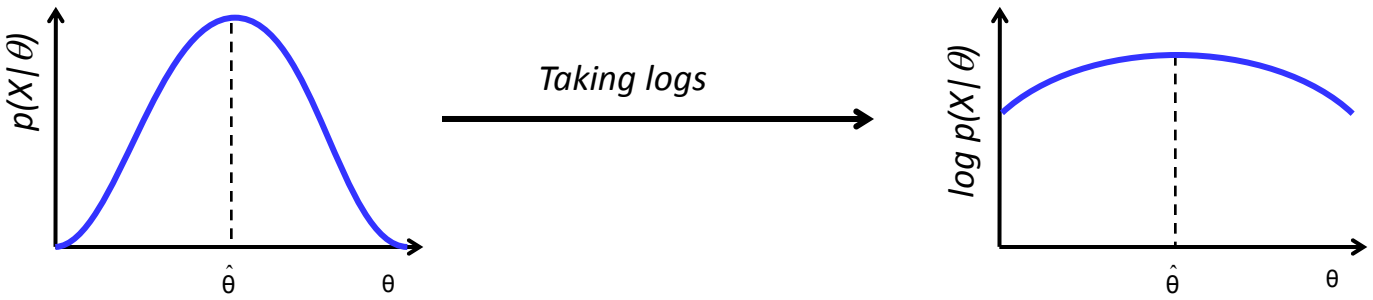
$$\hat{\theta} = argmax[p(X|\theta)]$$

  - This corresponds to the intuitive idea of choosing the value of θ that is most likely to give rise to the data

For convenience, we will work with the log likelihood

– Because the log is a monotonic function, then:

$$\hat{\theta} = argmax[p(X|\theta)] = argmax[\log p(X|\theta)]$$



– Hence, the ML estimate of θ can be written as:

$$\hat{\theta} = argmax\left[\log \Pi_{k=1}^{N} p\left(x^{(k}|\theta\right)\right] = argmax\left[\Sigma_{k=1}^{N} \log p\left(x^{(k}|\theta\right)\right]$$

- This simplifies the problem, since now we have to maximize a sum of terms rather than a long product of terms
- An added advantage of taking logs will become very clear when the distribution is Gaussian

Example: Gaussian case, μ unknown

Problem statement

- Assume a dataset $X = \{x^{(1}, x^{(2}, \dots x^{(N}\}$ and a density of the form $p(x) = N(\mu, \sigma)$ where σ is known
- What is the ML estimate of the mean?

$$\theta = \mu \Rightarrow \hat{\theta} = \arg max \Sigma_{k=1}^{N} log p\left(x^{(k} | \theta\right) =$$

$$= \arg max \Sigma_{k=1}^{N} log \left(\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2} \left(x^{(k} - \mu\right)^2 \right) \right) =$$

$$= \arg max \Sigma_{k=1}^{N} \left[log\left(\frac{1}{\sqrt{2\pi}\sigma} \right) - \frac{1}{2\sigma^2} \left(x^{(k} - \mu\right)^2 \right]$$

- The maxima of a function are defined by the zeros of its derivative

$$\frac{\partial \Sigma_{k=1}^{N} log p\left(x^{(k} | \theta\right)}{\partial \theta} = \frac{\partial}{\partial \theta} \Sigma_{k=1}^{N} log p(\cdot) = 0 \Rightarrow$$

$$\mu = \frac{1}{N} \Sigma_{k=1}^{N} x^{(k}$$

- So the ML estimate of the mean is the average value of the training data, a very intuitive result!

Example: Gaussian case, both μ and σ unknown

A more general case when neither μ nor σ is known

- Fortunately, the problem can be solved in the same fashion
- The derivative becomes a gradient since we have two variables

$$\hat{\theta} = \begin{bmatrix} \theta_1 = \mu \\ \theta_2 = \sigma^2 \end{bmatrix} \Rightarrow \nabla_\theta = \begin{bmatrix} \frac{\partial}{\partial \theta_1} \Sigma_{k=1}^{N} logp\left(x^{(k}|\theta\right) \\ \frac{\partial}{\partial \theta_2} \Sigma_{k=1}^{N} logp\left(x^{(k}|\theta\right) \end{bmatrix} = \Sigma_{k=1}^{N} \begin{bmatrix} \frac{1}{\theta_2}\left(x^{(k} - \theta_1\right) \\ -\frac{1}{2\theta_2} + \frac{\left(x^{(k} - \theta_1\right)^2}{2\theta_2^2} \end{bmatrix} = 0$$

- Solving for θ_1 and θ_2 yields

$$\hat{\theta}_1 = \frac{1}{N}\Sigma_{k=1}^{N} x^{(k}; \quad \hat{\theta}_2 = \frac{1}{N}\Sigma_{k=1}^{N}\left(x^{(k} - \hat{\theta}_1\right)^2$$

  - Therefore, the ML of the variance is the sample variance of the dataset, again a very pleasing result

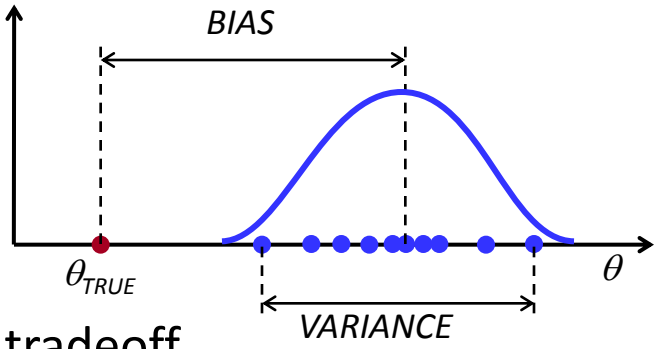
- Similarly, it can be shown that the ML estimates for the multivariate Gaussian are the sample mean vector and sample covariance matrix

$$\hat{\mu} = \frac{1}{N}\Sigma_{k=1}^{N} x^{(k}; \quad \hat{\Sigma} = \frac{1}{N}\Sigma_{k=1}^{N}\left(x^{(k} - \hat{\mu}\right)\left(x^{(k} - \hat{\mu}\right)^T$$

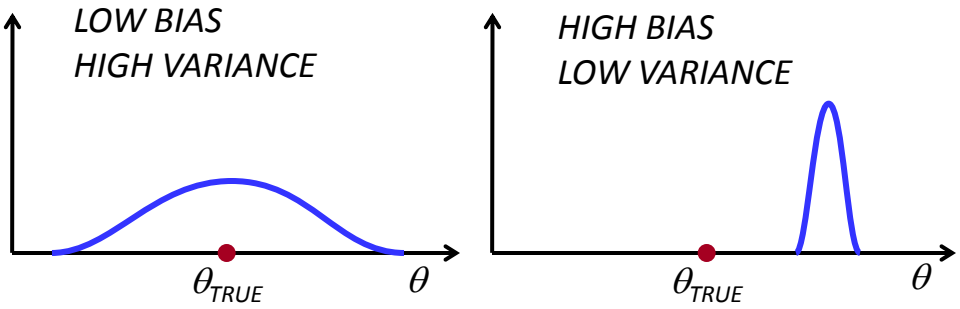
Bias and variance

How good are these estimates?

- Two measures of "goodness" are used for statistical estimates
- **BIAS**: how close is the estimate to the true value?
- **VARIANCE**: how much does it change for different datasets?



- The bias-variance tradeoff
  - In most cases, you can only decrease one of them at the expose of the other



What is the bias of the ML estimate of the mean?

$$E[\hat{\mu}] = E\left[\frac{1}{N}\Sigma_{k=1}^{N} x^{(k}\right] = \frac{1}{N}\Sigma_{k=1}^{N} E\left[x^{(k}\right] = \mu$$

- Therefore the mean is an unbiased estimate

What is the bias of the ML estimate of the variance?

$$E[\hat{\sigma}^2] = E\left[\frac{1}{N}\Sigma_{k=1}^{N}\left(x^{(k} - \hat{\mu}\right)^2\right] = \frac{N-1}{N}\sigma^2 \neq \sigma^2$$

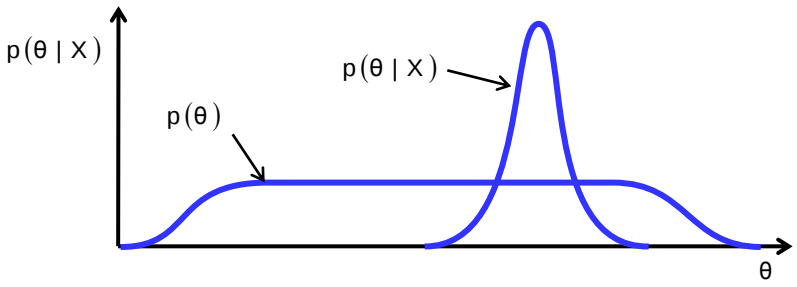
- Thus, the ML estimate of variance is BIASED
  - This is because the ML estimate of variance uses $\hat{\mu}$ instead of μ
- How "bad" is this bias?
  - For $N \to \infty$ the bias becomes zero asymptotically
  - The bias is only noticeable when we have very few samples, in which case we should not be doing statistics in the first place!
- Notice that MATLAB uses an unbiased estimate of the covariance

$$\hat{\Sigma}_{UNBIAS} = \frac{1}{N-1}\Sigma_{k=1}^{N}\left(x^{(k} - \hat{\mu}\right)\left(x^{(k} - \hat{\mu}\right)^T$$

Bayesian estimation

In the Bayesian approach, our uncertainty about the parameters is represented by a pdf

- Before we observe the data, the parameters are described by a prior density $p(\theta)$ which is typically very broad to reflect the fact that we know little about its true value
- Once we obtain data, we make use of Bayes theorem to find the posterior $p(\theta|X)$
  - Ideally we want the data to sharpen the posterior $p(\theta|X)$, that is, reduce our uncertainty about the parameters



- Remember, though, that our goal is to estimate $p(x)$ or, more exactly, $p(x|X)$, the density given the evidence provided by the dataset X

Let us derive the expression of a Bayesian estimate

- From the definition of conditional probability

$$p(x,\theta|X) = p(x|\theta,X)p(\theta|X)$$

- $P(x|\theta,X)$ is independent of X since knowledge of θ completely specifies the (parametric) density. Therefore

$$p(x,\theta|X) = p(x|\theta)p(\theta|X)$$

- and, using the theorem of total probability we can integrate θ out:

$$p(x|X) = \int p(x|\theta)p(\theta|X)d\theta$$

  - The only unknown in this expression is $p(\theta|X)$; using Bayes rule

$$p(\theta|X) = \frac{p(X|\theta)p(\theta)}{p(X)} = \frac{p(X|\theta)p(\theta)}{\int p(X|\theta)p(\theta)d\theta}$$

  - Where $p(X|\theta)$ can be computed using the i.i.d. assumption

$$p(X|\theta) = \prod_{k=1}^{N} p\left(x^{(k}|\theta\right)$$

  - NOTE: The last three expressions suggest a procedure to estimate $p(x|X)$. This is not to say that integration of these expressions is easy!

Example

- Assume a univariate density where our random variable x is generated from a normal distribution with known standard deviation
- Our goal is to find the mean μ of the distribution given some i.i.d. data points $X = \{x^{(1}, x^{(2}, \dots x^{(N}\}$
- To capture our knowledge about $\theta = \mu$, we assume that it also follows a normal density with mean μ_0 and standard deviation σ_0

$$p_0(\theta) = \frac{1}{\sqrt{2\pi}\sigma_0} e^{-\frac{1}{2\sigma_0^2}(\theta-\mu_0)^2}$$

- We use Bayes rule to develop an expression for the posterior $p(\theta|X)$

$$p(\theta|X) = \frac{p(X|\theta)p(\theta)}{p(X)} = \frac{p_0(\theta)}{p(X)} \Pi_{k=1}^{N} p\left(x^{(k}|\theta\right) =$$

$$\frac{1}{\sqrt{2\pi}\sigma_0} \mathrm{e}^{-\frac{1}{2\sigma_0^2}(\theta-\mu_0)^2} \frac{1}{p(X)} \prod_{k=1}^{N} \left[\frac{1}{\sqrt{2\pi}\sigma} \mathrm{e}^{-\frac{1}{2\sigma^2}\left(x^{(k}-\theta\right)^2}\right]$$

[Bishop, 1995]

- To understand how Bayesian estimation changes the posterior as more data becomes available, we will find the maximum of $p(\theta|X)$
- The partial derivative with respect to $\theta = \mu$ is

$$\frac{\partial}{\partial\theta}\log p(\theta|X) = 0 \Rightarrow \frac{\partial}{\partial\mu}\left[-\frac{1}{2\sigma_0^2}(\mu-\mu_0)^2 - \Sigma_{k=1}^{N}\frac{1}{2\sigma^2}\left(x^{(k}-\mu\right)^2\right] = 0$$

- which, after some algebraic manipulation, becomes

$$\mu_N = \underbrace{\frac{\sigma^2}{\sigma^2+N\sigma_0^2}\mu_0}_{PRIOR} + \underbrace{\frac{N\sigma_0^2}{\sigma^2+N\sigma_0^2}\frac{1}{N}\Sigma_{k=1}^{N}x^{(k}}_{ML}$$

  - Therefore, as N increases, the estimate of the mean μ_N moves from the initial prior μ_0 to the ML solution

- Similarly, the standard deviation σ_N can be found to be

$$\frac{1}{\sigma_N^2} = \frac{N}{\sigma^2} + \frac{1}{\sigma_0^2}$$

[Bishop, 1995]

Example

Assume that the true mean of the distribution $p(x)$ is $\mu = 0.8$ with standard deviation $\sigma = 0.3$

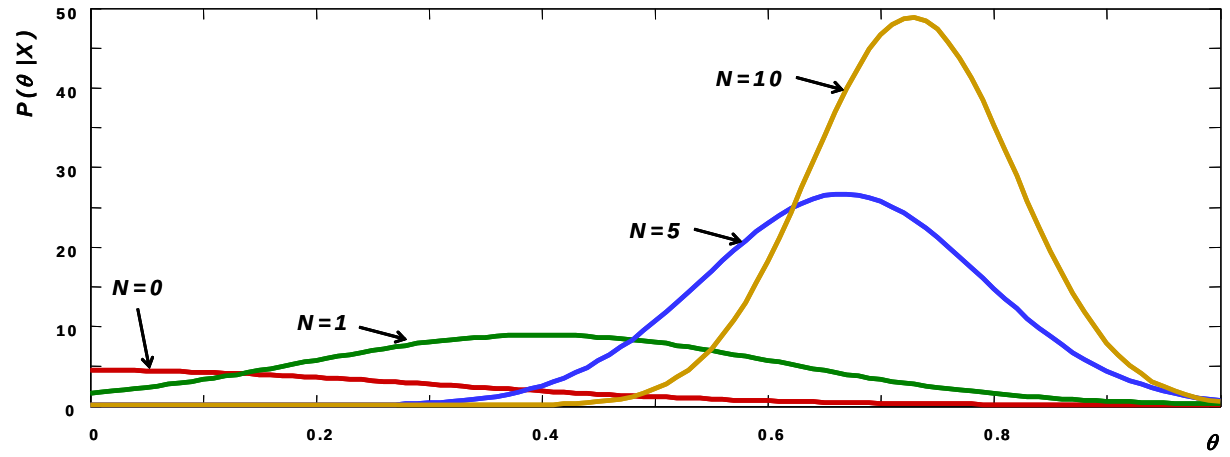
- In reality we would not know the true mean; we are just "playing God"

– We generate a number of examples from this distribution

– To capture our lack of knowledge about the mean, we assume a normal prior $p_0(\theta_0)$, with $\mu_0 = 0.0$ and $\sigma_0 = 0.3$

– The figure below shows the posterior $p(\mu|X)$

- As N increases, the estimate μ_N approaches its true value ($\mu = 0.8$) and the spread σ_N (or uncertainty in the estimate) decreases



ML vs. Bayesian estimation

What is the relationship between these two estimates?

- By definition, $p(X|\theta)$ peaks at the ML estimate
- If this peak is relatively sharp and the prior is broad, then the integral below will be dominated by the region around the ML estimate

$$p(x|X) = \int p(x|\theta)p(\theta|X)d\theta \cong p\left(x|\hat{\theta}\right)\underbrace{\int p(\theta|X)d\theta}_{=1} = p\left(x|\hat{\theta}\right)$$

  - Therefore, the Bayesian estimate will approximate the ML solution
- As we have seen in the previous example, when the number of available data increases, the posterior $p(\theta|X)$ tends to sharpen
  - Thus, the Bayesian estimate of $p(x)$ will approach the ML solution as $N \to \infty$
  - In practice, only when we have a limited number of observations will the two approaches yield different results