Parameter Estimation for Generative Models

Ziping Zhao

School of Information Science and Technology
ShanghaiTech University, Shanghai, China

CS182: Introduction to Machine Learning (Fall 2022)
http://cs182.sist.shanghaitech.edu.cn

Ch. 4 & Ch. 5 of I2ML

Outline

Outline

Different Approaches to Supervised Learning

- Three major approaches corresponding to different model assumptions:
  - Parametric approach
  - Nonparametric approach
  - Semiparametric approach
- Current topic: parametric approach

Parametric Approach

- Assumption: data follows a distribution that obeys a parametric model, e.g., Gaussian.
- Advantage of the parametric approach: the model is fully specified by a small number of parameters θ as sufficient statistics of the distribution
- The sample $\mathcal{X} \in \{\mathbf{x}^t\}$ is assumed to be drawn (usually i.i.d.) from an underlying distribution, i.e., $\mathbf{x}^t \sim p(\mathbf{x})$
- The number of parameters, i.e., $\dim(\theta)$, is independent of the sample size $|\mathcal{X}|$.
- Parameter estimation (or density estimation): assuming some parametric form, as a kind of inductive bias, for $p(\mathbf{x} \mid \theta)$ (or $p(\mathbf{x}; \theta)$), θ is estimated using $\mathcal{X}$; it is an unsupervised learning approach
- The estimated model/distribution is then used to make a decision
- Two approaches to parameter estimation:
  - Maximum likelihood estimation: θ is a fixed point (point estimation)
  - Bayesian estimation: θ is a random variable whose prior uncertainty (represented as a prior distribution) can be incorporated.

Parametric Approach to Classification

- Recall Bayes' rule for classification:

$$P(C_i \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid C_i)P(C_i)}{\sum_{k=1}^{K} p(\mathbf{x} \mid C_k)P(C_k)}$$

$$\text{Choose } C_i \text{ if } P(C_i \mid \mathbf{x}) = \max_k P(C_k \mid \mathbf{x})$$

- To compute $P(C_i \mid \mathbf{x})$ for classification, $p(\mathbf{x} \mid C_i)$ and $P(C_i)$ have to be estimated from the sample $\mathcal{X}$.
- A classifier thus created is often called a generative classifier (as opposed to a discriminative classifier to be studied later in this course), since it specifies how to generate the data based on $p(\mathbf{x} \mid C_i)$ and $P(C_i)$.

Maximum Likelihood Estimation - I

- Maximum likelihood estimation (MLE) seeks to find θ that makes sampling $\mathcal{X}$ from $p(\mathbf{x} \mid \theta)$ as likely as possible by maximizing the likelihood of θ given the sample $\mathcal{X} = \{\mathbf{x}^t\}_{t=1}^N$.
- Likelihood of θ given $\mathcal{X}$ (with the i.i.d. assumption):

$$L(\theta \mid \mathcal{X}) = p(\mathcal{X} \mid \theta) = \prod_{t=1}^{N} p(\mathbf{x}^t \mid \theta)$$

- Log likelihood (mainly for computational simplification):

$$\mathcal{L}(\theta \mid \mathcal{X}) = \log L(\theta \mid \mathcal{X}) = \sum_{t=1}^{N} \log p(\mathbf{x}^t \mid \theta)$$

- Maximum likelihood (ML) estimate:

$$\hat{\theta} = \arg\max_{\theta} \mathcal{L}(\theta \mid \mathcal{X})$$

Maximum Likelihood Estimation - II

Local extrema of a log-transformed function is equal to those of the original function.
Proof:

- Let $f(\mathbf{x})$ be a positive function.
- Suppose that $\mathbf{x}_0$ is the local maximum of $f(\mathbf{x})$ in a neighborhood of $\mathbf{x}_0$, denoted as $B_\epsilon(\mathbf{x}_0)$. Then, for any $\mathbf{y} \in B_\epsilon(\mathbf{x}_0)$, $f(\mathbf{y}) \leq f(\mathbf{x}_0)$.
- $\log(\cdot)$ is a monotonically increasing function: if $z \leq w$, then $\log z \leq \log w$.
- So for any $\mathbf{y} \in B_\epsilon(\mathbf{x}_0)$, since $f(\mathbf{y}) \leq f(\mathbf{x}_0)$, we have $\log f(\mathbf{y}) \leq \log f(\mathbf{x}_0)$. Hence, $\mathbf{x}_0$ is also the local maximum for $\log f(\mathbf{x})$.
- The other direction can be proven by noting that the inverse of $\log(\cdot)$ is also monotonically increasing.

Outline

Outline

Example: Bernoulli - I

- Discrete random variable x with two possible values, $x \in \{0, 1\}$
- E.g., in binary classification, use $P(x = 1)$ to represent $P(C_1)$ (and hence $P(x = 0) = 1 - P(x = 1)$ represents $P(C_2)$).
- Probability mass function (with parameter $\theta = p = P(x = 1)$):

$$P(x \mid p) = p^x (1 - p)^{1-x}$$

- Log likelihood:

$$\mathcal{L}(p \mid \mathcal{X}) = \sum_{t=1}^{N} \left[x^t \log p + (1 - x^t) \log(1 - p) \right]$$

- ML estimate:

$$\hat{p} = \frac{1}{N} \sum_{t=1}^{N} x^t \quad \text{(sample mean)}$$

Example: Bernoulli - II

- Note that the estimate $\hat{p}$ is a function of the sample and is another random variable
- We can discuss the distribution of $\hat{p}(\mathcal{X}_i)$ given different $\mathcal{X}_i$'s sampled from the same $p(x)$
  - the variance of the distribution of $\hat{p}(\mathcal{X}_i)$ is expected to decrease as N increases; as the samples get bigger, they (and hence their averages) get more similar.

Example: Binomial

- Discrete random variable x with M possible values, $x \in \{0, 1, \ldots, M\}$
- Probability mass function (with parameter $\theta = p$):

$$P(x \mid p) = \binom{M}{x} p^x (1-p)^{M-x}$$

- Log likelihood:

$$\mathcal{L}(p \mid \mathcal{X}) = \sum_{t=1}^{N} \left[\log \binom{M}{x_t} + x^t \log p + (M - x^t) \log(1-p) \right]$$

- ML estimate:

$$\hat{p} = \frac{1}{NM} \sum_{t=1}^{N} x^t \quad \text{(sample mean)}$$

- can be viewed as NM iid Bernoulli trials

Example: Generalized Bernoulli or Multinomial - I

- Discrete random variable x with $K \geq 2$ possible values, e.g., for K classes.
- Indicator variables $x_1, \ldots, x_K$:

$$x_i = \begin{cases} 1 & \text{if outcome is state } i \\ 0 & \text{if outcome is not state } i \end{cases}$$

  with $\sum_{i=1}^{K} x_i = 1$.
- Probability mass function (with parameters $\theta = (p_1, \ldots, p_K)^T$):

$$P(x \mid \theta) = P(x_1, \ldots, x_K \mid p_1, \ldots, p_K) = \prod_{i=1}^{K} p_i^{x_i}$$

  with constraint

$$\sum_{i=1}^{K} p_i = 1$$

Example: Generalized Bernoulli or Multinomial - II

- Log likelihood:

$$\mathcal{L}(p_1, \ldots, p_K \mid \mathcal{X}) = \sum_{t=1}^{N} \sum_{i=1}^{K} x_i^t \log p_i$$

- Constrained optimization problem (with equality constraint $\sum_{i=1}^{K} p_i = 1$)
  - can be solved using the method of Lagrange multipliers (or equivalently KKT optimality conditions).
- ML estimates:

$$\hat{p}_i = \frac{1}{N} \sum_{t=1}^{N} x_i^t$$

  So $\hat{\theta} = (\hat{p}_1, \ldots, \hat{p}_K)^T$ is also the sample mean.
- Another estimation method: view it as K separate Bernoulli experiments

Example: Normal (Gaussian)

- Continuous random variable x following univariate normal (Gaussian) distribution $\mathcal{N}(\mu, \sigma^2)$ with mean μ and variance σ^2 with $\sigma > 0$.
- Probability density function (with parameters $\theta = (\mu, \sigma)^T$):

$$p(x \mid \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]$$

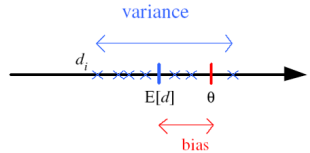
- Log likelihood:

$$\mathcal{L}(\mu, \sigma \mid \mathcal{X}) = -\frac{N}{2}\log(2\pi) - N\log\sigma - \frac{1}{2\sigma^2}\sum_{t=1}^{N}(x^t - \mu)^2$$

- ML estimates:

$$m = \frac{1}{N}\sum_{t=1}^{N} x^t, \qquad s^2 = \frac{1}{N}\sum_{t=1}^{N}(x^t - m)^2$$

Evaluating an Estimator: Bias and Variance - I



- Parameter to be estimated: θ, i.e., the true value
- Estimator based on sample $\mathcal{X}$: $d = d(\mathcal{X})$; a random variable depending on $\mathcal{X}$
- Bias: measures how much the expected value of the estimate $\mathbb{E}(d) = \mathbb{E}(d(\mathcal{X}))$ deviates from θ.

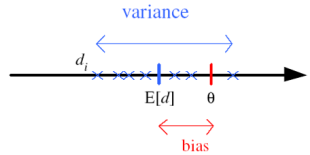
$$b_\theta(d) = \mathbb{E}(d) - \theta$$

Variance: measures how much, on average, the estimate d varies from the expected value $\mathbb{E}[d]$.

$$Var(d) = \mathbb{E}\left[(d - \mathbb{E}(d))^2\right]$$

- We would like both to be small.

Evaluating an Estimator: Bias and Variance - II



- Mean squared error (MSE) of estimator d (measures how much d is different from θ):

$$\begin{aligned} r(d,\theta) &= \mathbb{E}\left[(d-\theta)^2\right] \\ &= \mathbb{E}\left[(d-\mathbb{E}(d)+\mathbb{E}(d)-\theta)^2\right] \\ &= \dots \\ &= (\mathbb{E}(d)-\theta)^2 + \mathbb{E}\left[(d-\mathbb{E}(d))^2\right] \\ &= (b_\theta(d))^2 + Var(d) = \text{bias}^2 + \text{variance} \end{aligned}$$

Evaluating an Estimator: Unbiased Estimator

- If $b_\theta(d) = \mathbb{E}(d) - \theta = 0$ for all θ, d is an unbiased estimator of θ.
- the sample mean $m = \frac{1}{N}\sum_{t=1}^{N} x^t$ is an unbiased estimator of the mean μ
  - x^t is from some density with mean μ

$$\mathbb{E}(m) = \mathbb{E}\left(\frac{1}{N}\sum_{t=1}^{N} x^t\right) = \frac{1}{N}\sum_{t=1}^{N}\mathbb{E}(x^t) = \frac{1}{N}N\mu = \mu$$

  - $N \to \infty$, $m \to \mu$
- the normal ML estimator $s^2 = \frac{1}{N}\sum_{t=1}^{N}(x^t - m)^2$ is a biased estimator of σ^2
  - x^t is from some density with mean μ and variance σ^2

$$\mathbb{E}(s^2) = \mathbb{E}\left(\frac{1}{N}\sum_{t=1}^{N}(x^t - m)^2\right) = \mathbb{E}\left(\frac{1}{N}\sum_{t=1}^{N}(x^t)^2\right) - \mathbb{E}(m^2)$$

$$= \sigma^2 + \mu^2 - \left(\frac{\sigma^2}{N} + \mu^2\right) = \frac{N-1}{N}\sigma^2$$

  - $\frac{N}{N-1}s^2$ is an unbiased estimator

Evaluating an Estimator: Consistent Estimator

- sample mean m is a consistent estimator of μ, since $Var(m) \to 0$ as $N \to \infty$
  - x^t is from some density with variance σ^2

$$Var(m) = Var\left(\frac{1}{N}\sum_{t=1}^{N} x^t\right) = \frac{1}{N^2}\sum_{t=1}^{N} Var\left(x^t\right) = \frac{1}{N^2} N\sigma^2 = \frac{\sigma^2}{N}$$

  - as N gets larger, m deviates less from μ

Outline

Bayesian Estimation - I

- Before looking at a sample $\mathcal{X}$, we (or experts of the application) may have some prior information on the possible value range that a parameter, θ, may take.
- This information is useful and should be used, especially when the sample is small.
- The prior information does not tell us exactly what the parameter value is (otherwise we would not need the sample).
- Unlike MLE which treats θ as a fixed (but unknown) point, the Bayesian estimation approach treats it as a random variable with prior density $p(\theta)$ (i.e, our a prior uncertainty about θ, e.g. normal distribution).
- Combining the likelihood density $p(\mathcal{X} \mid \theta)$ (i.e., information from the sample $\mathcal{X}$), using Bayes' rule, we obtain posterior density of θ (i.e., uncertainty about θ after observing the sample $\mathcal{X}$):

$$p(\theta \mid \mathcal{X}) = \frac{p(\mathcal{X} \mid \theta)p(\theta)}{p(\mathcal{X})} = \frac{p(\mathcal{X} \mid \theta)p(\theta)}{\int p(\mathcal{X} \mid \theta')p(\theta')d\theta'}$$

Bayesian Estimation - II

- Full Bayesian estimation approach (through marginalization over θ):
  - Estimation of the density at any x (i.e., the probability that any sample occurs):

$$\begin{aligned} p(x \mid \mathcal{X}) &= \int p(x, \theta \mid \mathcal{X}) d\theta \\ &= \int p(x \mid \theta, \mathcal{X}) p(\theta \mid \mathcal{X}) d\theta \\ &= \int p(x \mid \theta) p(\theta \mid \mathcal{X}) d\theta \end{aligned}$$

    taking an average over predictions using all θ, weighted by their probabilities ($p(x \mid \theta, \mathcal{X}) = p(x \mid \theta)$ because once we know θ, the sufficient statistics, we know everything about the distribution)
  - Prediction, as in regression, based on a function $g(x \mid \theta)$:

$$y = \int g(x \mid \theta) p(\theta \mid \mathcal{X}) d\theta$$

Computational Considerations

- Evaluating the integrals (for marginalization over θ) may be computationally difficult, except in cases where the posterior $p(\theta \mid \mathcal{X})$ has a nice form.
- So the full Bayesian estimation approach is sometimes replaced by other methods for computational considerations.
- Maximum a posteriori (MAP) estimation - mode of the posterior density:

$$\theta_{\text{MAP}} = \arg\max_{\theta} p(\theta \mid \mathcal{X}) = \arg\max_{\theta} p(\mathcal{X} \mid \theta) p(\theta)$$

$$p(x \mid \mathcal{X}) = p(x \mid \theta_{\text{MAP}}) \qquad y = g(x \mid \theta_{\text{MAP}})$$

  - Maximum likelihood (ML) estimation – MAP with uniform/flat prior:

$$\theta_{\text{ML}} = \arg\max_{\theta} p(\mathcal{X} \mid \theta)$$

- Bayes' estimator – expectation w.r.t. posterior density:

$$\theta_{\text{Bayes'}} = \mathbb{E}(\theta \mid \mathcal{X}) = \int \theta p(\theta \mid \mathcal{X}) d\theta$$

$$p(x \mid \mathcal{X}) = p(x \mid \theta_{\text{Bayes'}}) \qquad y = g(x \mid \theta_{\text{Bayes'}})$$

Bayes' Estimator

- Bayes' estimator – expectation w.r.t. posterior density:

$$\theta_{\text{Bayes'}} = \mathbb{E}(\theta \mid \mathcal{X}) = \int \theta p(\theta \mid \mathcal{X}) d\theta$$

- The reason for taking the expected value is that the best estimate of a random variable is its mean.
  - Let us say θ is the variable we want to predict with $\mathbb{E}(\theta) = \mu$. For any estimate c (a constant value), we have the mean squared error (MSE)

$$\begin{aligned}\mathbb{E}[(\theta - c)^2] &= \mathbb{E}[(\theta - \mu + \mu - c)^2] \\ &= \mathbb{E}[(\theta - \mu)^2] + (\mu - c)^2\end{aligned}$$

    which is minimum if c is taken as μ.
  - Strictly speaking, $\mathbb{E}(\theta \mid \mathcal{X})$ is the Bayes' estimator under the squared error loss function, i.e., a minimum mean square error (MMSE) estimator.
- In the case of a normal density, the mode is the mean.
  - If $p(\theta \mid \mathcal{X})$ is normal, then $\theta_{\text{MAP}} = \theta_{\text{Bayes'}}$.

Example - I

- Bayesian estimation with known μ_0, σ_0, and σ:

$$x^t \sim \mathcal{N}(\theta, \sigma^2),\ \theta \sim \mathcal{N}(\mu_0, \sigma_0^2)$$

- ML estimator:

$$\theta_{\mathrm{ML}} = \frac{1}{N}\sum_{t=1}^{N} x^t = m \quad \text{(sample mean)}$$

- The $p(\theta \mid \mathcal{X})$ is normal.

$$\begin{aligned} p(\theta \mid \mathcal{X}) &\propto p(\mathcal{X} \mid \theta)p(\theta) \\ &\propto \frac{1}{(2\pi)^{\frac{N}{2}}\,\sigma^N} \exp\left[-\frac{\sum_{t=1}^{N}(x^t-\theta)^2}{2\sigma^2}\right] \times \frac{1}{(2\pi)^{\frac{1}{2}}\,\sigma_0} \exp\left[-\frac{(\theta-\mu_0)^2}{2\sigma_0^2}\right] \\ &\propto \exp\left[-\frac{\sum_{t=1}^{N}(\theta-x^t)^2}{2\sigma^2}\right] \exp\left[-\frac{(\theta-\mu_0)^2}{2\sigma_0^2}\right] \\ &\propto \exp\left[-\frac{\sigma_0^2\sum_{t=1}^{N}(\theta-x^t)^2 + \sigma^2(\theta-\mu_0)^2}{2\sigma_0^2\sigma^2}\right] \propto \exp\left[-\frac{\left(\theta - \frac{\sigma_0^2\sum_{i=1}^{N} x_i + \sigma^2\mu_0}{N\sigma_0^2+\sigma^2}\right)^2}{2C}\right] \end{aligned}$$

Example - II

- MAP and Bayes' estimator:

$$\theta_{\text{MAP}} = \theta_{\text{Bayes'}} = \mathbb{E}(\theta \mid \mathcal{X}) = \frac{N/\sigma^2}{N/\sigma^2 + 1/\sigma_0^2} m + \frac{1/\sigma_0^2}{N/\sigma^2 + 1/\sigma_0^2} \mu_0$$
$$= \frac{N\sigma_0^2}{N\sigma_0^2 + \sigma^2} m + \frac{\sigma^2}{N\sigma_0^2 + \sigma^2} \mu_0$$

weighted average of the sample mean m and the prior mean μ_0, with weights being inversely proportional to their variances

  - As N increases, the Bayes' estimator gets closer to the sample average, using more the information provided by the sample.
  - When σ_0^2 is small (we have little prior uncertainty regarding the correct value of θ), or when N is small, our prior guess μ_0 has a higher effect.

Bayesian Estimation

- Both MAP and Bayes' estimators reduce the whole posterior density to a single point and lose information unless the posterior is unimodal and makes a narrow peak around these points.
- With computation getting cheaper, we can use a Monte Carlo approach that generates samples from the posterior density.
- There also are many approximation methods one can use to evaluate the full integral in the full Bayesian estimation approach.

Outline

Classification with Discriminant Functions

- In Bayes' decision rule for classification, the discriminant function for of class C_i is

$$p(x \mid C_i)P(C_i) \quad \text{or} \quad \log\left[p(x \mid C_i)P(C_i)\right]$$

- Assume Gaussian density for each class:

$$p(x \mid C_i) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left[-\frac{(x-\mu_i)^2}{2\sigma_i^2}\right]$$

- Discriminant functions:

$$\begin{aligned} g_i(x) &= \log\left[p(x \mid C_i)P(C_i)\right] \\ &= \log p(x \mid C_i) + \log P(C_i) \\ &= -\frac{1}{2}\log 2\pi - \log \sigma_i - \frac{(x-\mu_i)^2}{2\sigma_i^2} + \log P(C_i) \end{aligned}$$

Discriminant Functions based on ML Estimates

- Samples $\mathcal{X} = \{(x^t, \mathbf{r}^t)\}_{t=1}^N$ where

$$x^t \in \mathbb{R} \qquad \mathbf{r} \in \{0,1\}^K \text{ with } r_i^t = \begin{cases} 1 & \text{if } x^t \text{ belongs to } C_i \\ 0 & \text{if } x^t \text{ belongs to } C_k, k \neq i \end{cases}$$

- ML estimates:

$$\hat{P}(C_i) = \frac{1}{N}\sum_{t=1}^{N} r_i^t \text{ (generalized Bernoulli density)}$$

$$m_i = \frac{\sum_{t=1}^N x^t r_i^t}{\sum_{t=1}^N r_i^t} \qquad s_i^2 = \frac{\sum_{t=1}^N (x^t - m_i)^2 r_i^t}{\sum_{t=1}^N r_i^t} \text{ (normal density)}$$

- Discriminant functions (with constant term $-\frac{1}{2}\log 2\pi$ dropped):

$$g_i(x) = -\log s_i - \frac{(x-m_i)^2}{2s_i^2} + \log \hat{P}(C_i)$$

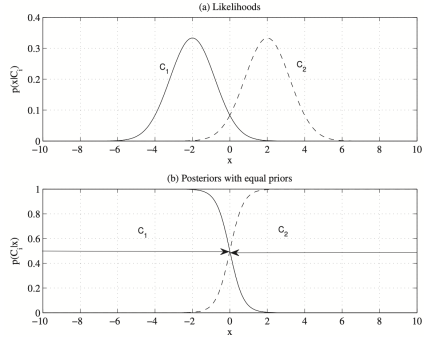
- Decision threshold between two adjacent classes C_i and C_j $(i \neq j)$: $g_i(x) = g_j(x)$

Case I: Equal Priors and Variances

- Simplified discriminant functions: $g_i(x) = -(x - m_i)^2$
  - the quadratic term x^2 can be reduced since it is common in all discriminants, leading to a linear discriminant
- Classification rule (a.k.a. nearest centroid (mean) classifier):

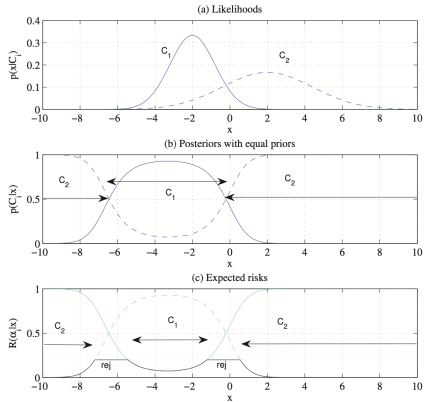
$$\text{Choose } C_i \text{ if } |x - m_i| = \min_k |x - m_k|$$

- Likelihood functions and posterior densities:



Case II: Equal Priors but Different Variances

- Simplified discriminant functions: $g_i(x) = -\log s_i - \frac{(x-m_i)^2}{2s_i^2}$
- Likelihood functions, posterior densities, and expected risks (for reject with $\lambda = 0.2$):



Case III: Different Priors

- Discriminant functions:

$$g_i(x) = -\log s_i - \frac{(x - m_i)^2}{2s_i^2} + \log \hat{P}(C_i)$$

  - a quadratic concave function in x

- If the priors are different, this has the effect of moving the threshold of decision toward the mean of the less likely class.

Discriminant Functions Based on Bayesian Estimates

- We have used the ML estimators for the parameters.
- If we have some prior information about them, for example, for the means, we can use a Bayesian estimate of $p(x \mid C_i)$ with prior on μ_i.

Remarks

Gaussianity

- When x is continuous, $p(x \mid C_i)$ cannot be immediately assumed as Gaussian densities. The classification algorithm—that is, the threshold points—will be wrong if the densities are not Gaussian.

Generative vs. Discriminative Classifiers

- This likelihood-based approach is a generative approach to classification where we use data to estimate the densities separately, calculate posterior densities using Bayes' rule, and then get the discriminant.
- We will discuss the discriminative approach (discriminant-based approach) in later lectures where we bypass the estimation of densities and directly estimate the discriminants.

Outline

Additive Parametric Model

- Functional relationship between numeric output (dependent variable) and input (independent variable) expressed in an additive form:

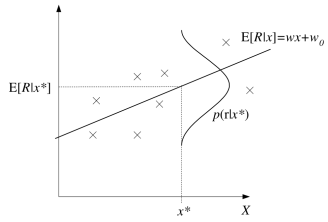
$$r = f(x) + \epsilon$$

  where $f(x)$ is an unknown but deterministic regression function and ϵ is random noise independent of the input.
- Parametric modeling:
  - $f(x) \approx$ estimator $g(x \mid \theta)$ (defined up to a set of parameters θ)
  - $\epsilon \sim \mathcal{N}(0, \sigma^2)$

Maximum Likelihood Estimation - I

- Conditional probability of output given input:

$$p(r \mid x, \theta) \sim \mathcal{N}(g(x \mid \theta), \sigma^2)$$



- We again use maximum likelihood to learn the parameters θ
- Joint density:

$$p(r, x \mid \theta) = p(r \mid x, \theta)p(x \mid \theta)$$

where $p(x)$ is the input density.

Maximum Likelihood Estimation - II

- Log likelihood of θ given an i.i.d. sample $\mathcal{X} = \{(x^t, r^t)\}_{t=1}^N$:

$$\begin{aligned}
\mathcal{L}(\theta \mid \mathcal{X}) &= \log \prod_{t=1}^{N} p(x^t, r^t \mid \theta) \\
&= \sum_{t=1}^{N} \log p(r^t \mid x^t, \theta) + \sum_{t=1}^{N} \log p(x^t \mid \theta) \\
&= \sum_{t=1}^{N} \log p(r^t \mid x^t, \theta) + \sum_{t=1}^{N} \log p(x^t) \\
&= \sum_{t=1}^{N} \log p(r^t \mid x^t, \theta) + \text{const.}
\end{aligned}$$

- We can ignore the second term since it does not depend on our estimator

Maximum Likelihood Estimation - III

- We have

$$\mathcal{L}(\theta \mid \mathcal{X}) = \sum_{t=1}^{N} \log \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{\left[r^t - g(x^t \mid \theta)\right]^2}{2\sigma^2}\right] + \text{const.}$$

$$= -\frac{1}{2\sigma^2} \sum_{t=1}^{N} \left[r^t - g(x^t \mid \theta)\right]^2 + \text{const.}$$

- Maximizing $\mathcal{L}(\theta \mid \mathcal{X})$ is equivalent to minimizing the following error function:

$$E(\theta \mid \mathcal{X}) = \frac{1}{2} \sum_{t=1}^{N} \left[r^t - g(x^t \mid \theta)\right]^2$$

  which is the most frequently used squared error function
- So the ML estimate of θ under Gaussian error assumption is also the least squares estimate.

Linear Regression - I

- Linear regression function:

$$g(x^t \mid w_0, w_1) = w_1 x^t + w_0$$

- In a "linear model", g is a linear function of the parameter θ but not necessarily of the data x^t.
- Error function:

$$E(w_0, w_1 \mid \mathcal{X}) = \frac{1}{2}\sum_{t=1}^{N}(r^t - w_1 x^t - w_0)^2$$

- Setting the derivatives of $E(w_0, w_1 \mid \mathcal{X})$ w.r.t. w_0, w_1 to 0 gives two equations

$$\sum_{t=1}^{N} r^t = N w_0 + \sum_{t=1}^{N} x^t w_1$$

$$\sum_{t=1}^{N} r^t x^t = \sum_{t=1}^{N} x^t w_0 + \sum_{t=1}^{N} \left(x^t\right)^2 w_1$$

Linear Regression - II

- Linear system of equations in vector-matrix form:

$$\mathbf{A}\mathbf{w} = \mathbf{y}$$

where

$$\mathbf{A} = \begin{bmatrix} N & \sum_t x^t \\ \sum_t x^t & \sum_t (x^t)^2 \end{bmatrix}$$

$$\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}$$

$$\mathbf{y} = \begin{bmatrix} \sum_t r^t \\ \sum_t r^t x^t \end{bmatrix}$$

- Least squares estimate (assuming that $\mathbf{A}$ is invertible):

$$\hat{\mathbf{w}} = \mathbf{A}^{-1}\mathbf{y}$$

- Linear models can be solved algebraically in closed form, while many non-linear models need to be solved numerically.

Polynomial Regression

- Polynomial regression function is a polynomial in x of order k:

$$g(x^t \mid w_0, w_1, \ldots, w_k) = w_k (x^t)^k + \cdots + w_2 (x^t)^2 + w_1 x^t + w_0$$

- The model is still linear with respect to the parameters.
  - As before, in the least squares estimate, taking the derivatives, we get $k+1$ equations in $k+1$ unknowns.
- An alternative approach for the least squares estimate (assuming that $\mathbf{D}^T\mathbf{D}$ is invertible):

$$\hat{\mathbf{w}} = (\underbrace{\mathbf{D}^T\mathbf{D}}_{\mathbf{A}})^{-1} \underbrace{\mathbf{D}^T\mathbf{r}}_{\mathbf{y}}$$

where

$$\mathbf{D} = \begin{bmatrix} 1 & x^1 & (x^1)^2 & \cdots & (x^1)^k \\ 1 & x^2 & (x^2)^2 & \cdots & (x^2)^k \\ \vdots & & & & \vdots \\ 1 & x^N & (x^N)^2 & \cdots & (x^N)^k \end{bmatrix} \qquad \mathbf{r} = \begin{bmatrix} r^1 \\ r^2 \\ \vdots \\ r^N \end{bmatrix}$$

Other Error Measures

- Squared error:
$$E(\theta \mid \mathcal{X}) = \frac{1}{2}\sum_t \left[r^t - g(x^t \mid \theta)\right]^2$$
- Relative squared error (RSE):
$$E(\theta \mid \mathcal{X}) = \frac{\sum_t \left[r^t - g(x^t \mid \theta)\right]^2}{\sum_t (r^t - \bar{r})^2}$$
- Absolute error:
$$E(\theta \mid \mathcal{X}) = \sum_t \left|r^t - g(x^t \mid \theta)\right|$$
- ϵ-insensitive error (for support vector machines to be covered later):
$$E(\theta \mid \mathcal{X}) = \sum_t \mathbf{1}(|r^t - g(x^t \mid \theta)| > \epsilon)(|r^t - g(x^t \mid \theta)| - \epsilon)$$

Outline

Tuning Model Complexity

- Remember that for best generalization, we should adjust the complexity of our learner model to the complexity of the data.
- In polynomial regression, the complexity parameter is the order of the fitted polynomial.
- Therefore we need to find a way to choose the best order that minimizes the generalization error, that is, tune the complexity of the model to best fit the complexity of the function inherent in the data.

Bias and Variance - I

- A sample $\mathcal{X} = \{(x^t, r^t)\}_{t=1}^N$ is drawn from some unknown joint pdf $p(x, r)$.
- Using this sample $\mathcal{X}$, we construct our estimate $g(\cdot)$.
- Expected squared error of estimator $g(\cdot)$ at x over $p(x, r)$:

$$\mathbb{E}\Big[(r - g(x))^2 \mid x\Big] = \underbrace{\mathbb{E}\Big[(r - \mathbb{E}[r \mid x])^2 \mid x\Big]}_{\text{noise}} + \underbrace{\Big(\mathbb{E}[r \mid x] - g(x)\Big)^2}_{\text{squared error}}$$

Only the second term depends on the estimator $g(\cdot)$.

- The first term is the variance of r given x; it does not depend on $g(\cdot)$ or $\mathcal{X}$. It is the variance of noise added, σ^2. This is the part of error that can never be removed, no matter what estimator we use.
- The second term quantifies how much $g(x)$ deviates from the regression function, $f(x) = \mathbb{E}[r \mid x]$. It depends on the estimator $g(\cdot)$ and the training set $\mathcal{X}$.

Bias and Variance - II

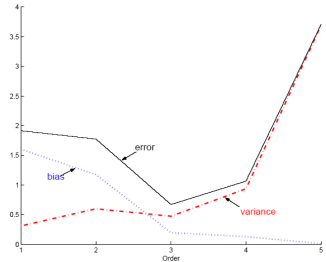
- It may be the case that for one sample, $g(x)$ may be a very good fit; and for some other sample, it may make a bad fit.
- We average over different samples $\mathcal{X}$ to quantify how well an estimator $g(\cdot)$ is, giving the expected value (average over samples $\mathcal{X}$, all of size N and drawn from the same joint density $p(x, r)$):

$$\begin{aligned}&\mathbb{E}_{\mathcal{X}}\left[(\mathbb{E}[r \mid x] - g(x))^2 \mid x\right] \\ &= \underbrace{\left(\mathbb{E}[r \mid x] - \mathbb{E}_{\mathcal{X}}[g(x)]\right)^2}_{\text{bias}^2} + \underbrace{\mathbb{E}_{\mathcal{X}}\left[(g(x) - \mathbb{E}_{\mathcal{X}}[g(x)])^2\right]}_{\text{variance}}\end{aligned}$$

  - bias measures how much $g(x)$ is wrong disregarding the effect of varying samples
  - variance measures how much $g(x)$ fluctuate around the expected value, $\mathbb{E}_{\mathcal{X}}[g(x)]$, as the sample varies
  - we want both to be small

Bias/Variance Dilemma

- As model complexity increases (e.g., order of polynomial increases):
  - bias decreases (better fit to data)
  - variance increases (fit varies more with data)
  - called the bias/variance dilemma which is true for any machine learning system
- If there is bias, this indicates that our model class does not contain the solution; this is underfitting.
- If there is variance, the model class is too general and also learns the noise; this is overfitting.
- Optimal model: best tradeoff between bias and variance.



- Model selection: search for optimal or suboptimal model.

Common Model Selection Methods - I

- Cross-validation: Measure generalization accuracy by testing on data unused during model training.
- Regularization: Penalize complex models by minimizing an augmented error function:

$$E' = \text{error on data} + \lambda \cdot \text{model complexity}$$

The regularization parameter λ, which can be determined by cross-validation, controls the weight of penalty.

Common Model Selection Methods - II

- Bayesian model selection is used when there exists prior knowledge about the appropriate class of approximating functions, represented as a prior distribution over models, $p(\text{model})$.
- Posterior distribution over models given the data:

$$p(\text{model} \mid \text{data}) = \frac{p(\text{data} \mid \text{model})p(\text{model})}{p(\text{data})}$$

- From the posterior distribution, we may:
  - choose the model with the highest posterior probability, or
  - choose multiple models with high posterior probabilities, or
  - use all models weighted by their posterior probabilities.
- Taking the log of the above equation

$$\log p(\text{model} \mid \text{data}) = \log p(\text{data} \mid \text{model}) + \log p(\text{model}) - \text{const.}$$

- Regularization may be seen as a Bayesian approach in which the "prior" favors simpler models.

Common Model Selection Methods - III

- Cross-validation is different from the other methods for model selection in that it makes no prior assumption about the model or parameters.
- If there is a large enough validation dataset, cross-validation is the best approach.
- When the data sample is small, the other models become useful.