Mathematics for Machine Learning

— Linear Regression: Problem Formulation & Parameter Estimation

Joseph Chuang-Chieh Lin

Department of Computer Science & Information Engineering,
Tamkang University

Fall 2023

Credits for the resource

- The slides are based on the textbooks:
  - *Marc Peter Deisenroth, A. Aldo Faisal, and Cheng Soon Ong: Mathematics for Machine Learning. Cambridge University Press. 2020.*
  - *Howard Anton, Chris Rorres, Anton Kaul: Elementary Linear Algebra. Wiley. 2019.*
- We could partially refer to the monograph:
  Francesco Orabona: A Modern Introduction to Online Learning.
  https://arxiv.org/abs/1912.13213

Outline

1. Introduction
2. Problem Formulation
3. Parameter Estimation
    - Maximum Likelihood Estimation (MLE)
    - Overfitting in Linear Regression
    - Maximum A Posteriori Estimation (MAP)
    - MAP Estimation as Regularization

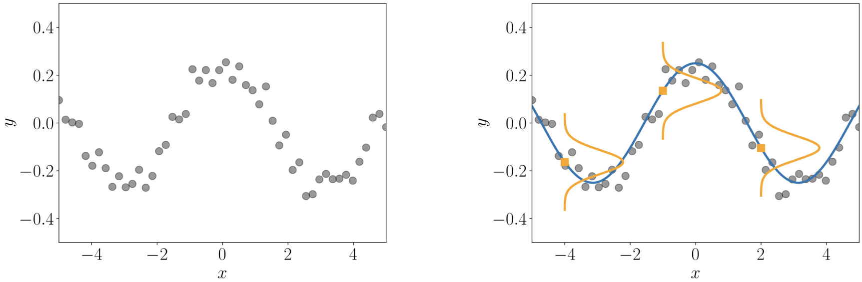
Linear Regression

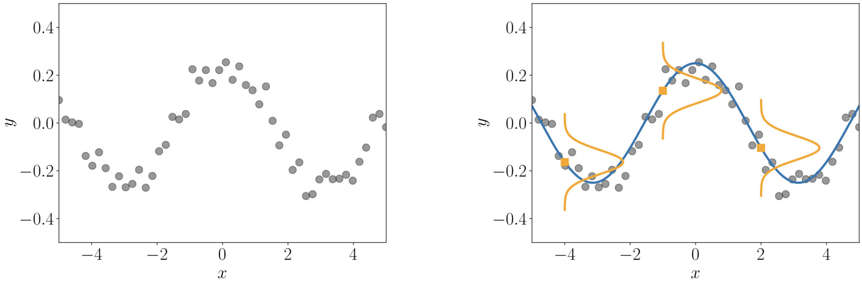
Aim

Find (or Infer) a function $f : \mathbb{R}^D \mapsto \mathbb{R}$ which maps input $\mathbf{x} \in \mathbb{R}^D$ to the corresponding function values $f(\mathbf{x}) \in \mathbb{R}$.

- And we hope f to generalize well to unseen input.

- Training input: $\{\mathbf{x}_i\}_{i=1}^N$
- Assume the noisy observations $\{y_i\}_{i=1}^N$ for $y_i = f(\mathbf{x}_i) + \epsilon$, an i.i.d. random variable ϵ.
  - Consider zero-mean Gaussian noise throughout our discussions.


0.4
0.2
0.0
−0.2
−0.4
y
−4
−2
0
2
4
x
0.4
0.2
0.0
−0.2
−0.4
y
−4
−2
0
2
4
x



Applications of regression:

- Time series analysis, Reinforcement learning, Optimization, Computer games, Classification algorithms, etc.

Problems Involved in Regression

- Choice of the model and the parametrization.
  - Function classes, particular parametrization (e.g., degree of the polynomial)
- Finding good parameters
  - Loss minimization w.r.t. different loss functions.
- Overfitting and model selection
- Relationship b/w loss functions and parameter priors.
  - Probabilistic models.
- Uncertainty modeling.
  - We have limited amount of data.
  - Equip model predictions with confidence bounds.

Outline

Problem Formulation

- Because of observing noise, we adopt a probabilistic approach to explicitly model the noise using a likelihood function.
- **Focus:** a regression problem with the likelihood function:

$$p(y \mid \mathbf{x}) = \mathcal{N}(y \mid f(\mathbf{x}), \sigma^2).$$

  - $\mathbf{x} \in \mathbb{R}^D$: inputs.
  - $y \in \mathbb{R}$: noisy function values (targets).
  - The relationship between $\mathbf{x}$ and y:

$$y = f(\mathbf{x}) + \epsilon,$$

  for $\epsilon \sim \mathcal{N}(0, \sigma^2)$.

An Example of Linear Regression

- An example of linear regression:

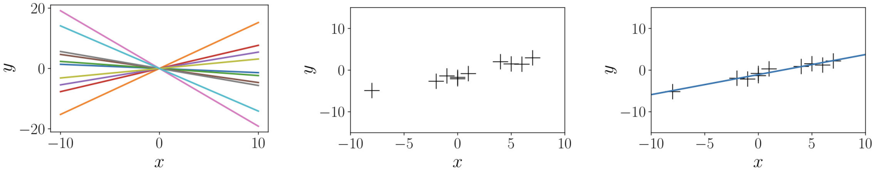
$$p(y \mid \mathbf{x}, \boldsymbol{\theta}) = \mathcal{N}(y \mid \mathbf{x}^\top \boldsymbol{\theta}, \sigma^2).$$

$\iff$

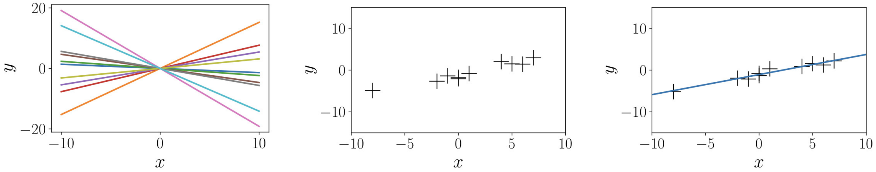
$$y = \mathbf{x}^\top \boldsymbol{\theta} + \epsilon,$$

for $\epsilon \sim \mathcal{N}(0, \sigma^2)$.

  - $\boldsymbol{\theta} \in \mathbb{R}^D$: the parameters we seek.
  - ϵ: the only source of uncertainty.



- "Linear": linear in the parameters.



- "Linear": linear in the parameters.
- Hence, $y = \phi^\top(\mathbf{x})\boldsymbol{\theta}$ is also regarded as a linear regression (ϕ can be nonlinear) .

Outline

The Likelihood

- Given a training set $\mathcal{D} := \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, $\mathbf{x}_i \in \mathbb{R}^D$ and $y_i \in \mathbb{R}$ for $i = 1, \ldots, N$.
- By the independence of the input, the likelihood factorizes:

$$
\begin{aligned}
p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) &= p(y_1, \ldots, y_N \mid \mathbf{x}_1, \ldots, \mathbf{x}_N, \boldsymbol{\theta}) \\
&= \prod_{i=1}^{N} p(y_i \mid \mathbf{x}_i, \boldsymbol{\theta}) = \prod_{i=1}^{N} \mathcal{N}(y_i \mid \mathbf{x}_i^\top \boldsymbol{\theta}, \sigma^2).
\end{aligned}
$$

The likelihood and the factors $p(y_i \mid \mathbf{x}_i, \boldsymbol{\theta})$ are Gaussian due to the noise distribution.

The Likelihood

- Given a training set $\mathcal{D} := \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, $\mathbf{x}_i \in \mathbb{R}^D$ and $y_i \in \mathbb{R}$ for $i = 1, \ldots, N$.
- By the independence of the input, the likelihood factorizes:

$$\begin{aligned} p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) &= p(y_1, \ldots, y_N \mid \mathbf{x}_1, \ldots, \mathbf{x}_N, \boldsymbol{\theta}) \\ &= \prod_{i=1}^{N} p(y_i \mid \mathbf{x}_i, \boldsymbol{\theta}) = \prod_{i=1}^{N} \mathcal{N}(y_i \mid \mathbf{x}_i^\top \boldsymbol{\theta}, \sigma^2). \end{aligned}$$

  The likelihood and the factors $p(y_i \mid \mathbf{x}_i, \boldsymbol{\theta})$ are Gaussian due to the noise distribution.
- **Goal:** Find optimal parameters $\boldsymbol{\theta}^* \in \mathbb{R}^D$.

The Likelihood

- Given a training set $\mathcal{D} := \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, $\mathbf{x}_i \in \mathbb{R}^D$ and $y_i \in \mathbb{R}$ for $i = 1, \ldots, N$.
- By the independence of the input, the likelihood factorizes:

$$\begin{aligned} p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) &= p(y_1, \ldots, y_N \mid \mathbf{x}_1, \ldots, \mathbf{x}_N, \boldsymbol{\theta}) \\ &= \prod_{i=1}^{N} p(y_i \mid \mathbf{x}_i, \boldsymbol{\theta}) = \prod_{i=1}^{N} \mathcal{N}(y_i \mid \mathbf{x}_i^\top \boldsymbol{\theta}, \sigma^2). \end{aligned}$$

  The likelihood and the factors $p(y_i \mid \mathbf{x}_i, \boldsymbol{\theta})$ are Gaussian due to the noise distribution.

- **Goal:** Find optimal parameters $\boldsymbol{\theta}^* \in \mathbb{R}^D$.
- Then we can make predictions for an arbitrary test input $\mathbf{x}_*$ and get target y_* with $p(y_* \mid \mathbf{x}_*, \boldsymbol{\theta}^*) = \mathcal{N}(y_* \mid \mathbf{x}_*^\top \boldsymbol{\theta}^*, \sigma^2)$.

Outline

Maximum Likelihood Estimation (MLE)

Find parameters $\boldsymbol{\theta}_{ML}$

$$\boldsymbol{\theta}_{ML} \in \arg\max_{\boldsymbol{\theta}} p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}).$$

Note:

- The likelihood $p(y \mid \mathbf{x}, \boldsymbol{\theta})$ is NOT a probability distribution of $\boldsymbol{\theta}$.

Maximum Likelihood Estimation (MLE)

Find parameters $\boldsymbol{\theta}_{ML}$

$$\boldsymbol{\theta}_{ML} \in \arg\max_{\boldsymbol{\theta}} p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}).$$

Note:

- The likelihood $p(y \mid \mathbf{x}, \boldsymbol{\theta})$ is NOT a probability distribution of $\boldsymbol{\theta}$. It's a function of $\boldsymbol{\theta}$ (might not be integrable w.r.t $\boldsymbol{\theta}$).
- However, it's a normalized probability distribution in y.

How to find the desired θ_{ML}?

1. Perform gradient ascent (or descent).

How to find the desired θ_{ML}?

1. Perform gradient ascent (or descent).
2. For linear regression, we can directly have a closed-form solution.

How to find the desired θ_{ML}?

1. Perform gradient ascent (or descent).
2. For linear regression, we can directly have a closed-form solution.
3. In practice, we do not maximize the likelihood directly. Instead, we apply the negative log-likelihood.
    - It does not suffer from numerical underflow.
    - The differentiation rules become simpler.

Maximize likelihood $\Leftrightarrow$ Minimize negative log-likelihood

The negative log-likelihood

$$-\log p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) = -\log \prod_{i=1}^{N} p(y_i \mid \mathbf{x}_i, \boldsymbol{\theta}) = -\sum_{i=1}^{N} \log p(y_i \mid \mathbf{x}_i, \boldsymbol{\theta}).$$

⋆ **Note:** the independence assumption on the training set applies here.

Maximize likelihood ⇔ Minimize negative log-likelihood

The negative log-likelihood

$$-\log p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) = -\log \prod_{i=1}^{N} p(y_i \mid \mathbf{x}_i, \boldsymbol{\theta}) = -\sum_{i=1}^{N} \log p(y_i \mid \mathbf{x}_i, \boldsymbol{\theta}).$$

⋆ **Note:** the independence assumption on the training set applies here.

$$\log p(y_i \mid \mathbf{x}_i, \boldsymbol{\theta}) = -\frac{1}{2\sigma^2}(y_i - \mathbf{x}^\top \boldsymbol{\theta})^2 + \text{constant}_{\text{indepent of } \boldsymbol{\theta}}.$$

Ignoring the constant terms, we obtain

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) &:= \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - \mathbf{x}_i^\top \boldsymbol{\theta})^2 \\ &= \frac{1}{2\sigma^2}(\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta})^\top(\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta}) = \frac{1}{2\sigma^2}\|\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta}\|^2,\end{aligned}$$

where $\boldsymbol{X} := [\mathbf{x}_1, \ldots, \mathbf{x}_N]^\top \in \mathbb{R}^{N\times D}$ and $\mathbf{y} := [y_1, \ldots, y_N]^\top \in \mathbb{R}^N$.

Ignoring the constant terms, we obtain

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) &:= \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - \mathbf{x}_i^\top\boldsymbol{\theta})^2 \\ &= \frac{1}{2\sigma^2}(\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta})^\top(\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta}) = \frac{1}{2\sigma^2}\|\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta}\|^2,\end{aligned}$$

where $\boldsymbol{X} := [\mathbf{x}_1, \ldots, \mathbf{x}_N]^\top \in \mathbb{R}^{N\times D}$ and $\mathbf{y} := [y_1, \ldots, y_N]^\top \in \mathbb{R}^N$.

To get $\boldsymbol{\theta}$, we need to solve $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = \mathbf{0}^\top$

Ignoring the constant terms, we obtain

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) &:= \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - \mathbf{x}_i^\top \boldsymbol{\theta})^2 \\ &= \frac{1}{2\sigma^2}(\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta})^\top(\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta}) = \frac{1}{2\sigma^2}\|\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta}\|^2,\end{aligned}$$

where $\boldsymbol{X} := [\mathbf{x}_1, \ldots, \mathbf{x}_N]^\top \in \mathbb{R}^{N\times D}$ and $\mathbf{y} := [y_1, \ldots, y_N]^\top \in \mathbb{R}^N$.

To get $\boldsymbol{\theta}$, we need to solve $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = \mathbf{0}^\top$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = \mathbf{0}^\top \iff \boldsymbol{\theta}_{ML}^\top \boldsymbol{X}^\top \boldsymbol{X} = \mathbf{y}^\top \boldsymbol{X}$$

Ignoring the constant terms, we obtain

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) &:= \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - \mathbf{x}_i^\top \boldsymbol{\theta})^2 \\ &= \frac{1}{2\sigma^2}(\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta})^\top(\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta}) = \frac{1}{2\sigma^2}\|\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta}\|^2,\end{aligned}$$

where $\boldsymbol{X} := [\mathbf{x}_1, \ldots, \mathbf{x}_N]^\top \in \mathbb{R}^{N\times D}$ and $\mathbf{y} := [y_1, \ldots, y_N]^\top \in \mathbb{R}^N$.

To get $\boldsymbol{\theta}$, we need to solve $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = \mathbf{0}^\top$

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = \mathbf{0}^\top \quad &\Longleftrightarrow \quad \boldsymbol{\theta}_{ML}^\top \boldsymbol{X}^\top \boldsymbol{X} = \mathbf{y}^\top \boldsymbol{X} \\ &\Longleftrightarrow \quad \boldsymbol{\theta}_{ML}^\top = \mathbf{y}^\top \boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X})^{-1}\end{aligned}$$

Ignoring the constant terms, we obtain

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) &:= \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - \mathbf{x}_i^\top \boldsymbol{\theta})^2 \\ &= \frac{1}{2\sigma^2}(\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta})^\top(\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta}) = \frac{1}{2\sigma^2}\|\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta}\|^2,\end{aligned}$$

where $\boldsymbol{X} := [\mathbf{x}_1, \ldots, \mathbf{x}_N]^\top \in \mathbb{R}^{N\times D}$ and $\mathbf{y} := [y_1, \ldots, y_N]^\top \in \mathbb{R}^N$.

To get $\boldsymbol{\theta}$, we need to solve $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = \mathbf{0}^\top$

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = \mathbf{0}^\top &\iff \boldsymbol{\theta}_{ML}^\top \boldsymbol{X}^\top \boldsymbol{X} = \mathbf{y}^\top \boldsymbol{X} \\ &\iff \boldsymbol{\theta}_{ML}^\top = \mathbf{y}^\top \boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X})^{-1} \\ &\iff \boldsymbol{\theta}_{ML} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1}\boldsymbol{X}^\top \mathbf{y}.\end{aligned}$$

- ⋆ We use the positive definite property of $\boldsymbol{X}^\top \boldsymbol{X}$ if $\operatorname{rank}(\boldsymbol{X}) = D$.

Remark

- We can get a global minimum because the Hessian $\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}) = \boldsymbol{X}^\top\boldsymbol{X}$ is positive definite (for full rank $\boldsymbol{X}$?).

MLE with Features

- Note that "linear" regression is linear in the "parameters".
- We can perform an arbitrary nonlinear transformation $\phi(\mathbf{x})$ of the input $\mathbf{x}$, and then linearly combine these components.

MLE with Features

- Note that "linear" regression is linear in the "parameters".
- We can perform an arbitrary nonlinear transformation $\phi(\mathbf{x})$ of the input $\mathbf{x}$, and then linearly combine these components.
- The corresponding linear regression turns out to be:

$$p(y \mid \mathbf{x}, \boldsymbol{\theta}) = \mathcal{N}(y \mid \boldsymbol{\phi}^\top(\mathbf{x})\boldsymbol{\theta}, \sigma^2).$$

$\Longleftrightarrow$

$$y = \boldsymbol{\phi}^\top(\mathbf{x})\boldsymbol{\theta} + \epsilon$$

MLE with Features

- Note that "linear" regression is linear in the "parameters".
- We can perform an arbitrary nonlinear transformation $\phi(\mathbf{x})$ of the input $\mathbf{x}$, and then linearly combine these components.
- The corresponding linear regression turns out to be:

$$p(y \mid \mathbf{x}, \boldsymbol{\theta}) = \mathcal{N}(y \mid \boldsymbol{\phi}^\top(\mathbf{x})\boldsymbol{\theta}, \sigma^2).$$

$\iff$

$$y = \boldsymbol{\phi}^\top(\mathbf{x})\boldsymbol{\theta} + \epsilon = \sum_{k=0}^{K-1} \theta_k \phi_k(\mathbf{x}) + \epsilon$$

- $\boldsymbol{\phi} : \mathbb{R}^D \mapsto \mathbb{R}$ is a (nonlinear) transformation of the input $\mathbf{x}$
- $\phi_k : \mathbb{R}^D \mapsto \mathbb{R}$: the kth feature vector of $\boldsymbol{\phi}$.

Polynomial Regression

Consider a regression problem $y = \boldsymbol{\phi}^\top(\mathbf{x})\boldsymbol{\theta} + \epsilon$, for $x \in \mathbb{R}$ and $\boldsymbol{\theta} \in \mathbb{R}^K$. A polynomial tranformation of $\mathbf{x}$ is often used as

$$\boldsymbol{\phi}(x) = \begin{bmatrix} \phi_0(x) \\ \phi_1(x) \\ \vdots \\ \phi_{K-1}(x) \end{bmatrix} = \begin{bmatrix} 1 \\ x \\ x^2 \\ \vdots \\ x^{K-1} \end{bmatrix} \in \mathbb{R}^K.$$

- We lift the original one-dimensional input space into a K-dimensional feature space.
- We can model polynomials of degree $\leq K-1$ as $f(x) = \sum_{k=1}^{K-1} \theta_k x^k = \boldsymbol{\phi}^\top(x)\boldsymbol{\theta}$, for $\boldsymbol{\theta} = [\theta_0, \ldots, \theta_{K-1}]^\top \in \mathbb{R}^K$ which contains the linear parameters θ_k.

For $\mathbf{x}_i \in \mathbb{R}^D$

We can also define a feature matrix as

$$\boldsymbol{\Phi} := \begin{bmatrix} \phi^\top(\mathbf{x}_1) \\ \vdots \\ \phi^\top(\mathbf{x}_N) \end{bmatrix} = \begin{bmatrix} \phi_0(\mathbf{x}_1) & \cdots & \phi_{K-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \cdots & \phi_{K-1}(\mathbf{x}_2) \\ \vdots & & \vdots \\ \phi_0(\mathbf{x}_N) & \cdots & \phi_{K-1}(\mathbf{x}_N) \end{bmatrix} \in \mathbb{R}^{N \times K},$$

where $\Phi_{ij} = \phi_j(\mathbf{x}_i)$ and $\phi_j : \mathbb{R}^D \mapsto \mathbb{R}$.

Feature Matrix for Second-Order Polynomials

$$\boldsymbol{\Phi} := \begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & x_N & x_N^2 \end{bmatrix}.$$

With the feature matrix $\boldsymbol{\Phi}$:

$$\boldsymbol{\Phi} := \begin{bmatrix} \phi^\top(\mathbf{x}_1) \\ \vdots \\ \phi^\top(\mathbf{x}_N) \end{bmatrix} = \begin{bmatrix} \phi_0(\mathbf{x}_1) & \cdots & \phi_{K-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \cdots & \phi_{K-1}(\mathbf{x}_2) \\ \vdots & & \vdots \\ \phi_0(\mathbf{x}_N) & \cdots & \phi_{K-1}(\mathbf{x}_N) \end{bmatrix} \in \mathbb{R}^{N\times K},$$

The negative log-likelihood can be written as

$$-\log p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) = \frac{1}{2\sigma^2}(\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta})^\top(\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta}) + \text{constant}.$$

- Replacing $\boldsymbol{X}$ by $\boldsymbol{\Phi}$.
- Both of them are independent of $\boldsymbol{\theta}$.

[1]Requring $\text{rank}(\boldsymbol{\Phi}) = K$

With the feature matrix $\boldsymbol{\Phi}$:

$$\boldsymbol{\Phi} := \begin{bmatrix} \phi^\top(\mathbf{x}_1) \\ \vdots \\ \phi^\top(\mathbf{x}_N) \end{bmatrix} = \begin{bmatrix} \phi_0(\mathbf{x}_1) & \cdots & \phi_{K-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \cdots & \phi_{K-1}(\mathbf{x}_2) \\ \vdots & & \vdots \\ \phi_0(\mathbf{x}_N) & \cdots & \phi_{K-1}(\mathbf{x}_N) \end{bmatrix} \in \mathbb{R}^{N \times K},$$

The negative log-likelihood can be written as

$$-\log p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) = \frac{1}{2\sigma^2}(\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta})^\top(\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta}) + \text{constant}.$$

- Replacing $\boldsymbol{X}$ by $\boldsymbol{\Phi}$.
- Both of them are independent of $\boldsymbol{\theta}$.
- Similarly, we have[1]

$$\boldsymbol{\theta}_{ML} = (\boldsymbol{\Phi}^\top\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^\top\mathbf{y}.$$

[1] Requring $\mathrm{rank}(\boldsymbol{\Phi}) = K$

Estimating the Noise Variance (1/2)

- We can also use the principle of MLE to obtain that for σ^2_{ML} for the noise variance.

Estimating the Noise Variance (1/2)

- We can also use the principle of MLE to obtain that for σ^2_{ML} for the noise variance.
- Write down the log-likelihood:

$$\begin{aligned}
\log p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}, \sigma^2) &= \sum_{i=1}^{N} \log \mathcal{N}(y_i \mid \boldsymbol{\phi}^\top(\mathbf{x}_i)\boldsymbol{\theta}, \sigma^2) \\
&= \sum_{i=1}^{N} \left(-\frac{1}{2}\log(2\pi) - \frac{1}{2}\log\sigma^2 - \frac{1}{2\sigma^2}(y_i - \boldsymbol{\phi}^\top(\mathbf{x}_i)\boldsymbol{\theta})^2 \right) \\
&= -\frac{N}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - \boldsymbol{\phi}^\top(\mathbf{x}_i)\boldsymbol{\theta})^2 + \text{constant}
\end{aligned}$$

Estimating the Noise Variance (1/2)

- We can also use the principle of MLE to obtain that for σ^2_{ML} for the noise variance.
- Write down the log-likelihood:

$$
\begin{aligned}
\log p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}, \sigma^2) &= \sum_{i=1}^{N} \log \mathcal{N}(y_i \mid \boldsymbol{\phi}^\top(\mathbf{x}_i)\boldsymbol{\theta}, \sigma^2) \\
&= \sum_{i=1}^{N} \left(-\frac{1}{2}\log(2\pi) - \frac{1}{2}\log\sigma^2 - \frac{1}{2\sigma^2}(y_i - \boldsymbol{\phi}^\top(\mathbf{x}_i)\boldsymbol{\theta})^2 \right) \\
&= -\frac{N}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_i - \boldsymbol{\phi}^\top(\mathbf{x}_i)\boldsymbol{\theta})^2 + \text{constant}
\end{aligned}
$$

Let $s := \sum_{i=1}^{N}(y_i - \boldsymbol{\phi}^\top(\mathbf{x}_i)\boldsymbol{\theta})^2$.

Estimating the Noise Variance (2/2)

- The partial derivative w.r.t. σ^2:

$$\begin{aligned} \frac{\partial \log p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}, \sigma^2)}{\partial \sigma^2} &= -\frac{N}{2\sigma^2} + \frac{1}{\sigma^4} s = 0 \\ \iff \quad \frac{N}{2\sigma^2} &= \frac{s}{2\sigma^4}. \end{aligned}$$

Thus, $\sigma^2_{ML} = \frac{s}{N} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \boldsymbol{\phi}^\top(\mathbf{x}_i)\boldsymbol{\theta})^2$.

Outline

Evaluating the Quality of the Model

- We can evaluate the quality of the model by computing the error/loss.
- Given that σ^2 is not a free model parameter, we can ignore that term by scaling by $1/\sigma^2$ and derive a squared-error function $\|\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta}\|^2$.

Evaluating the Quality of the Model

- We can evaluate the quality of the model by computing the error/loss.
- Given that σ^2 is not a free model parameter, we can ignore that term by scaling by $1/\sigma^2$ and derive a squared-error function $\|\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta}\|^2$.
- To compare the errors of datasets with different sizes and the same scale, we often use the root-mean squared error (RMSE):

Evaluating the Quality of the Model

- We can evaluate the quality of the model by computing the error/loss.
- Given that σ^2 is not a free model parameter, we can ignore that term by scaling by $1/\sigma^2$ and derive a squared-error function $\|\mathbf{y} - \boldsymbol{\Phi\theta}\|^2$.
- To compare the errors of datasets with different sizes and the same scale, we often use the root-mean squared error (RMSE):

$$\sqrt{\frac{1}{N}\|\mathbf{y} - \boldsymbol{\Phi\theta}\|^2} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \phi^\top(\mathbf{x}_i)\boldsymbol{\theta})^2}$$

Evaluating the Quality of the Model

- We can evaluate the quality of the model by computing the error/loss.
- Given that σ^2 is not a free model parameter, we can ignore that term by scaling by $1/\sigma^2$ and derive a squared-error function $\|\mathbf{y} - \mathbf{\Phi}\boldsymbol{\theta}\|^2$.
- To compare the errors of datasets with different sizes and the same scale, we often use the root-mean squared error (RMSE):

$$\sqrt{\frac{1}{N}\|\mathbf{y} - \mathbf{\Phi}\boldsymbol{\theta}\|^2} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \phi^\top(\mathbf{x}_i)\boldsymbol{\theta})^2}$$

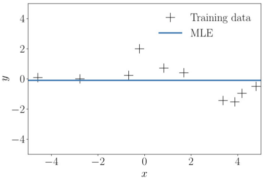
- Model selection:

Evaluating the Quality of the Model

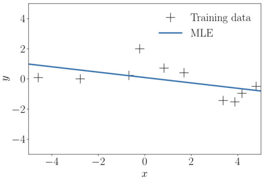
- We can evaluate the quality of the model by computing the error/loss.
- Given that σ^2 is not a free model parameter, we can ignore that term by scaling by $1/\sigma^2$ and derive a squared-error function $\|\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta}\|^2$.
- To compare the errors of datasets with different sizes and the same scale, we often use the root-mean squared error (RMSE):

$$\sqrt{\frac{1}{N}\|\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta}\|^2} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \boldsymbol{\phi}^\top(\mathbf{x}_i)\boldsymbol{\theta})^2}$$

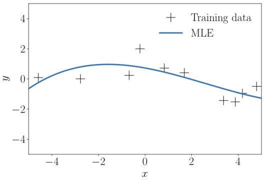
- Model selection: determine the best degree of the polynomial.
  - Brute-force searching and enumerate all reasonable polynomial degrees M.



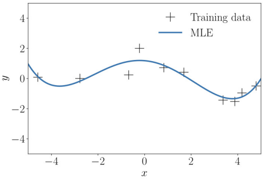
(a) $M = 0$



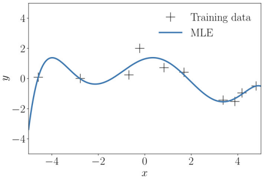
(b) $M = 1$



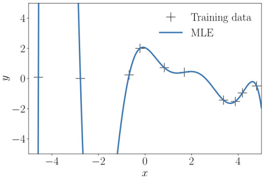
(c) $M = 3$



(d) $M = 4$

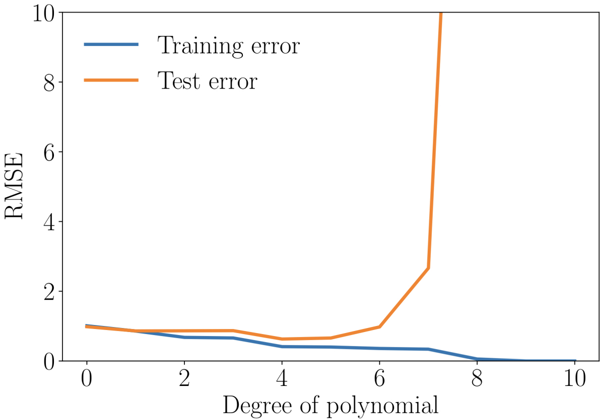


(e) $M = 6$



(f) $M = 9$

Goal: a good generalization by making *accurate* predictions for new unseen data.


Training error
Test error
RMSE
Degree of polynomial
10
8
6
4
2
0
0
2
4
6
8
10

Outline

Motivation

- MLE is prone to overfitting.
- **Experience:** The parameter values becomes relatively large when the model is overfitting.

Motivation

- MLE is prone to overfitting.
- **Experience:** The parameter values becomes relatively large when the model is overfitting.
- To mitigate the effect of huge parameter values, we place a prior distribution $p(\boldsymbol{\theta})$ on the parameters.

Motivation

- MLE is prone to overfitting.
- **Experience:** The parameter values becomes relatively large when the model is overfitting.
- To mitigate the effect of huge parameter values, we place a prior distribution $p(\boldsymbol{\theta})$ on the parameters.
- **Rough idea:** Encode the parameter values that are plausible before seeing any data.
  - For example, a Gaussian prior $p(\boldsymbol{\theta}) = \mathcal{N}(\mathbf{0}, \boldsymbol{I})$.

Maximum a Posteriori Estimation (1/5)

- Once a dataset $(\mathcal{X}, \mathcal{Y})$ is available, we week parameters that maximize the posterior distribution $p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y})$ instead of maximizing the likelihood.

$$p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y}) = \frac{p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\mathcal{Y} \mid \mathcal{X})}.$$

Maximum a Posteriori Estimation (1/5)

- Once a dataset $(\mathcal{X}, \mathcal{Y})$ is available, we week parameters that maximize the posterior distribution $p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y})$ instead of maximizing the likelihood.

$$p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y}) = \frac{p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\mathcal{Y} \mid \mathcal{X})}.$$

  - The prior will have an effect on the parameter vector.

Maximum a Posteriori Estimation (1/5)

- Once a dataset $(\mathcal{X}, \mathcal{Y})$ is available, we week parameters that maximize the posterior distribution $p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y})$ instead of maximizing the likelihood.

$$p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y}) = \frac{p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\mathcal{Y} \mid \mathcal{X})}.$$

  - The prior will have an effect on the parameter vector.
  - $\boldsymbol{\theta}_{MAP}$: the maximizer of the above posterior (i.e., the MAP estimate).

Maximum a Posteriori Estimation (2/5)

The log-transformation of the posterior:

$$\log p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y}) = \log p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) + \text{constant}$$

The constant is independent of $\boldsymbol{\theta}$.

We can see that the MAP estimate is a compromise between the prior and the data-dependent likelihood.

Maximum a Posteriori Estimation (2/5)

The log-transformation of the posterior:

$$\log p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y}) = \log p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) + \text{constant}$$

The constant is independent of $\boldsymbol{\theta}$.

We can see that the MAP estimate is a compromise between the prior and the data-dependent likelihood.

We minimize the negative log-posterior w.r.t. θ:

$$\boldsymbol{\theta}_{MAP} \in \arg\min_{\boldsymbol{\theta}} \{-\log p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) - \log p(\boldsymbol{\theta})\}.$$

Maximum a Posteriori Estimation (3/5)

$$\boldsymbol{\theta}_{MAP} \in \arg\min_{\boldsymbol{\theta}} \{-\log p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) - \log p(\boldsymbol{\theta})\}.$$

The gradient:

$$-\frac{\mathrm{d} \log p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y})}{\mathrm{d}\boldsymbol{\theta}} = -\frac{\mathrm{d} \log p(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta})}{\mathrm{d}\boldsymbol{\theta}} - \frac{\mathrm{d} \log p(\boldsymbol{\theta})}{\mathrm{d}\boldsymbol{\theta}}.$$

Assume the Gaussian prior $p(\boldsymbol{\theta}) = \mathcal{N}(\mathbf{0}, b^2 \boldsymbol{I})$. We have

$$-\log p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y}) = \frac{1}{2\sigma^2}(\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta})^\top (\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta}) + \frac{1}{2b^2}\boldsymbol{\theta}^\top \boldsymbol{\theta} + \text{constant}$$

Maximum a Posteriori Estimation (4/5)

$$-\log p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y}) = \frac{1}{2\sigma^2}(\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta})^\top(\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta}) + \frac{1}{2b^2}\boldsymbol{\theta}^\top\boldsymbol{\theta} + \text{constant}$$

Hence, the gradient of the log-posterior w.r.t. $\boldsymbol{\theta}$ is

$$-\frac{\mathrm{d}\log p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y})}{\mathrm{d}\boldsymbol{\theta}} = \frac{1}{\sigma^2}(\boldsymbol{\theta}^\top\boldsymbol{\Phi}^\top\boldsymbol{\Phi} - \mathbf{y}^\top\boldsymbol{\Phi}) + \frac{1}{b^2}\boldsymbol{\theta}^\top.$$

Setting the gradient to $\mathbf{0}^\top$ to get $\boldsymbol{\theta}_{MAP}$:

Maximum a Posteriori Estimation (5/5)

$$
\begin{aligned}
& \frac{1}{\sigma^2}(\boldsymbol{\theta}^\top \boldsymbol{\Phi}^\top \boldsymbol{\Phi} - \mathbf{y}^\top \boldsymbol{\Phi}) + \frac{1}{b^2}\boldsymbol{\theta}^\top = \mathbf{0}^\top \\
\iff \quad & \boldsymbol{\theta}^\top \left(\frac{1}{\sigma^2}\boldsymbol{\Phi}^\top \boldsymbol{\Phi} + \frac{1}{b^2}\boldsymbol{I} \right) - \frac{1}{\sigma^2}\mathbf{y}^\top \boldsymbol{\Phi} = \mathbf{0}^\top \\
\iff \quad & \boldsymbol{\theta}^\top \left(\boldsymbol{\Phi}^\top \boldsymbol{\Phi} + \frac{\sigma^2}{b^2}\boldsymbol{I} \right) = \mathbf{y}^\top \boldsymbol{\Phi} \\
\iff \quad & \boldsymbol{\theta}^\top = \mathbf{y}^\top \boldsymbol{\Phi} \left(\boldsymbol{\Phi}^\top \boldsymbol{\Phi} + \frac{\sigma^2}{b^2}\boldsymbol{I} \right)^{-1} .
\end{aligned}
$$

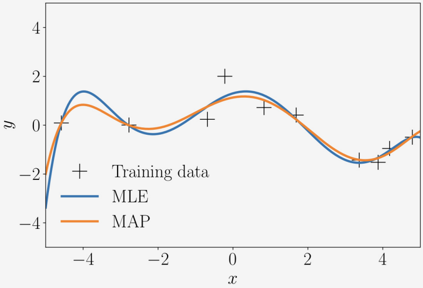
Finally, we have

Maximum a Posteriori Estimation (5/5)

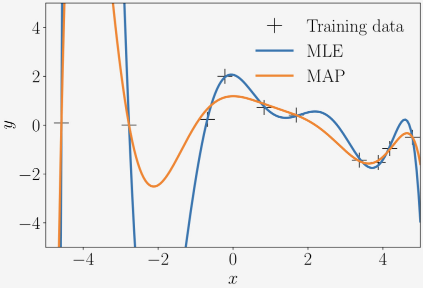
$$
\begin{aligned}
& \frac{1}{\sigma^2}(\boldsymbol{\theta}^\top \boldsymbol{\Phi}^\top \boldsymbol{\Phi} - \mathbf{y}^\top \boldsymbol{\Phi}) + \frac{1}{b^2}\boldsymbol{\theta}^\top = \mathbf{0}^\top \\
\iff \quad & \boldsymbol{\theta}^\top \left(\frac{1}{\sigma^2}\boldsymbol{\Phi}^\top \boldsymbol{\Phi} + \frac{1}{b^2}\boldsymbol{I} \right) - \frac{1}{\sigma^2}\mathbf{y}^\top \boldsymbol{\Phi} = \mathbf{0}^\top \\
\iff \quad & \boldsymbol{\theta}^\top \left(\boldsymbol{\Phi}^\top \boldsymbol{\Phi} + \frac{\sigma^2}{b^2}\boldsymbol{I} \right) = \mathbf{y}^\top \boldsymbol{\Phi} \\
\iff \quad & \boldsymbol{\theta}^\top = \mathbf{y}^\top \boldsymbol{\Phi} \left(\boldsymbol{\Phi}^\top \boldsymbol{\Phi} + \frac{\sigma^2}{b^2}\boldsymbol{I} \right)^{-1} .
\end{aligned}
$$

Finally, we have

$$
\boldsymbol{\theta}_{MAP} = \left(\boldsymbol{\Phi}^\top \boldsymbol{\Phi} + \frac{\sigma^2}{b^2}\boldsymbol{I} \right)^{-1} \boldsymbol{\Phi}^\top \mathbf{y}.
$$



(a) Polynomials of degree 6.



(b) Polynomials of degree 8.

Outline

Motivation (I)

- Mitigate the effect of overfitting by penalizing the amplitude of the parameters by means of regularization.
- Consider the regularized least squares:

$$\underbrace{\|\mathbf{y} - \mathbf{\Phi}\boldsymbol{\theta}\|^2}_{\text{for fitting data}} + \underbrace{\lambda\|\boldsymbol{\theta}\|_2^2}_{\text{regularizer}}$$

for the regularization parameter $\lambda \geq 0$.

- The 2-norm $\|\cdot\|_2$ can be replaced by other types of norm.

Motivation (II)

- The regularizer $\lambda\|\boldsymbol{\theta}\|_2^2$ can be seen as a negative log-Gaussian prior.
- The Gaussian prior $p(\boldsymbol{\theta}) = \mathcal{N}(\mathbf{0}, b^2\boldsymbol{I})$, so the negative log-Gaussian prior is

$$-\log p(\boldsymbol{\theta}) = \frac{1}{2b^2}\|\boldsymbol{\theta}\|_2^2 + \text{constant}$$

  hence we have $\lambda = \frac{1}{2b^2}$.

Minimizing the regularized least-squares loss function yields

$$\boldsymbol{\theta}_{RLS} = (\boldsymbol{\Phi}^\top \boldsymbol{\Phi} + \lambda \boldsymbol{I})^{-1} \boldsymbol{\Phi}^\top \mathbf{y}.$$

Minimizing the regularized least-squares loss function yields

$$\boldsymbol{\theta}_{RLS} = (\boldsymbol{\Phi}^\top \boldsymbol{\Phi} + \lambda \boldsymbol{I})^{-1} \boldsymbol{\Phi}^\top \mathbf{y}.$$

This is identical to the MAP estimate for $\lambda = \frac{\sigma^2}{b^2}$.

- σ^2: the noise variance
- b^2: the variance of the Gaussian prior $p(\boldsymbol{\theta}) = \mathcal{N}(\mathbf{0}, b^2 \boldsymbol{I})$.

Discussions