Probabilistic Modelling Principles

Yingzhen Li

Department of Computing
Imperial College London

@liyzhen2
yingzhen.li@imperial.ac.uk

October 21, 2022

Principles of probabilistic modelling

Have you ever wondered about the following questions:

- Why using ℓ_2 loss in many regression problems?
- Where does the cross-entropy loss come from?
- What is a good principle for choosing a good loss function?

Principles of probabilistic modelling

Have you ever wondered about the following questions:

- Why using ℓ_2 loss in many regression problems?
- Where does the cross-entropy loss come from?
- What is a good principle for choosing a good loss function?

Probabilistic modelling gives you good answers for all of them!

Principles of probabilistic modelling

Probabilistic modelling is about:

1. making model assumptions on **how the data is generated**
2. estimating model parameters under probabilistic principles
3. model checking using data, and repeat 1 - 3
4. using the fitted model for downstream tasks

Example: coin flipping

Imagine you'd like to predict the next coin flip result:

- Assume $x_1, x_2, ..., x_N$ are observed **independent** coin flip results using the **same** coin,
- I.e., $x_1, ..., x_N$ are sampled i.i.d. from the same **data distribution** $p_{data}(x)$
- However, we don't know $p_{data}(x)$

Example: coin flipping

Imagine you'd like to predict the next coin flip result:

Probabilistic modelling is about:

1. Assume x is sampled from $p(x|\boldsymbol{\theta})$ $\Leftarrow$ **our probabilistic model**
2. estimating $\boldsymbol{\theta}$ under probabilistic principles such as MLE, MAP, posterior inference $\Leftarrow$ **learning the model**
3. check if $p(x|\boldsymbol{\theta}^*)$ fits $p_{data}(x)$ well, and repeat 1 - 3 $\Leftarrow$ **model checking**
4. making prediction for next coin flip result using $p(x|\boldsymbol{\theta}^*)$

Example: coin flipping

Imagine you'd like to predict the next coin flip result:

Step 1: Assume x is sampled from $p(x|\boldsymbol{\theta})$

$$x = \begin{cases} 1, & \text{with probability } \boldsymbol{\theta} \\ 0, & \text{with probability } 1 - \boldsymbol{\theta} \end{cases}, \quad \boldsymbol{\theta} \in [0, 1].$$

$$\Leftrightarrow \quad p(x|\boldsymbol{\theta}) = \text{Bern}(\boldsymbol{\theta}).$$

- Likelihood of $\boldsymbol{\theta}$ given observed x: $\ell(\boldsymbol{\theta}) = p(x|\boldsymbol{\theta})$

Example: coin flipping

Imagine you'd like to predict the next coin flip result:

Step 2: estimating $\boldsymbol{\theta}$ using probabilistic principles
Here we consider **maximum likelihood estimation (MLE)**

Idea of MLE: for datapoints x sampled from $p_{data}(x)$

- We want to find $\boldsymbol{\theta}^*$ such that $p(x|\boldsymbol{\theta}^*) \approx p_{data}(x)$

Example: coin flipping

Imagine you'd like to predict the next coin flip result:

Step 2: estimating $\boldsymbol{\theta}$ using probabilistic principles
Here we consider **maximum likelihood estimation (MLE)**

Idea of MLE: for datapoints x sampled from $p_{data}(x)$

- We want to find $\boldsymbol{\theta}^*$ such that $p(x|\boldsymbol{\theta}^*) \approx p_{data}(x)$
- We need to measure the "closeness" of the two distributions $\Rightarrow$ use the KL divergence

$$\mathrm{KL}[p_{data}(x)||p(x|\boldsymbol{\theta})] = \mathbb{E}_{p_{data}(x)}\left[\log \frac{p_{data}(x)}{p(x|\boldsymbol{\theta})}\right]$$

Example: coin flipping

Imagine you'd like to predict the next coin flip result:

Step 2: estimating $\boldsymbol{\theta}$ using probabilistic principles
Here we consider **maximum likelihood estimation (MLE)**

Idea of MLE: for datapoints x sampled from $p_{data}(x)$

- We want to find $\boldsymbol{\theta}^*$ such that $p(x|\boldsymbol{\theta}^*) \approx p_{data}(x)$
- We want this KL to be small:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathrm{KL}[p_{data}(x)||p(x|\boldsymbol{\theta})]$$

Example: coin flipping

Imagine you'd like to predict the next coin flip result:

Step 2: estimating $\boldsymbol{\theta}$ using probabilistic principles
Here we consider **maximum likelihood estimation (MLE)**

Idea of MLE: for datapoints x sampled from $p_{data}(x)$

- We want to find $\boldsymbol{\theta}^*$ such that $p(x|\boldsymbol{\theta}^*) \approx p_{data}(x)$

$$\Leftrightarrow \quad \boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{p_{data}(x)}[\log p(x|\boldsymbol{\theta})]$$

Example: coin flipping

Imagine you'd like to predict the next coin flip result:

Step 2: estimating $\boldsymbol{\theta}$ using probabilistic principles
Here we consider **maximum likelihood estimation (MLE)**

Idea of MLE: for datapoints x sampled from $p_{data}(x)$

- We want to find $\boldsymbol{\theta}^*$ such that $p(x|\boldsymbol{\theta}^*) \approx p_{data}(x)$
- Estimate using dataset $\mathcal{D} = \{x_1, ..., x_N\}$ sampled from $p_{data}(x)$:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{n=1}^{N} \log p(x_n|\boldsymbol{\theta})$$

Example: coin flipping

Imagine you'd like to predict the next coin flip result:

Step 2: estimating $\boldsymbol{\theta}$ using probabilistic principles
Here we consider **maximum likelihood estimation (MLE)**

- Estimate using dataset $\mathcal{D} = \{x_1, ..., x_N\}$ sampled from $p_{data}(x)$:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{n=1}^{N} \log p(x_n | \boldsymbol{\theta})$$

- model assumption: $p(x|\boldsymbol{\theta}) = \text{Bern}(\boldsymbol{\theta})$

$$\Rightarrow \quad \boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{n=1}^{N} x_n \log \boldsymbol{\theta} + (1 - x_n) \log(1 - \boldsymbol{\theta})$$

Example: coin flipping

Imagine you'd like to predict the next coin flip result:

Step 2: estimating $\boldsymbol{\theta}$ using probabilistic principles
Here we consider **maximum likelihood estimation (MLE)**

- Estimate using dataset $\mathcal{D} = \{x_1, ..., x_N\}$ sampled from $p_{data}(x)$:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{n=1}^{N} \log p(x_n | \boldsymbol{\theta})$$

- solution by zeroing the gradient:

$$\frac{1}{N} \sum_{n=1}^{N} x_n \boldsymbol{\theta}^{-1} - (1 - x_n)(1 - \boldsymbol{\theta})^{-1} = 0 \quad \Rightarrow \quad \boldsymbol{\theta}^* = \frac{1}{N} \sum_{n=1}^{N} x_n$$

Example: coin flipping

Imagine you'd like to predict the next coin flip result:

Step 3: check if $p(x|\boldsymbol{\theta}^*)$ fits $p_{data}(x)$ well
(We assume the model has passed here)

Example: coin flipping

Imagine you'd like to predict the next coin flip result:

Step 4: making prediction for next coin flip result using $p(x|\boldsymbol{\theta}^*)$

$$\boldsymbol{\theta}^* = \frac{1}{N} \sum_{x \in \mathcal{D}} x$$

$$\Rightarrow \quad x_{N+1} = \begin{cases} 1, & \text{with probability } \frac{1}{N} \sum_{n=1}^{N} x_n \\ 0, & \text{with probability } 1 - \frac{1}{N} \sum_{n=1}^{N} x_n \end{cases} .$$

Principles of probabilistic modelling

Datapoints $(\boldsymbol{x}, y)$ are sampled from an **unknown ground truth distribution** $p_{data}(\boldsymbol{x}, y)$

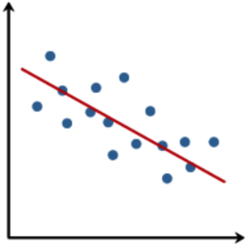
Probabilistic modelling is about (in supervised learning case):

1. Assuming the output y given $\boldsymbol{x}$ is sampled from

$$p(y|\boldsymbol{x}, \boldsymbol{\theta})$$

2. estimating $\boldsymbol{\theta}$ under probabilistic principles
   such as MLE, MAP, posterior inference
3. check if $p(y|\boldsymbol{x}, \boldsymbol{\theta}^*)$ fits $p_{data}(\boldsymbol{x}, y)$ well, and repeat 1 - 3
4. using $p(y|\boldsymbol{x}, \boldsymbol{\theta}^*)$ for predictions

Probabilistic modelling: linear regression

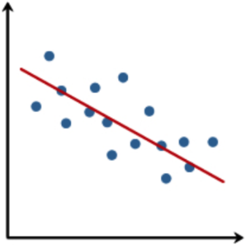


Linear regression

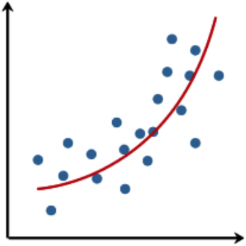
$$f(\boldsymbol{x}, \boldsymbol{\theta}) = \boldsymbol{\theta}^\top \boldsymbol{x},$$

$$y = f(\boldsymbol{x}, \boldsymbol{\theta}) + \epsilon, \epsilon \sim \mathcal{N}(0, \sigma^2)$$

Probabilistic modelling: linear regression



$\Rightarrow$



Linear regression

$$f(\boldsymbol{x}, \boldsymbol{\theta}) = \boldsymbol{\theta}^\top \boldsymbol{x},$$

$$y = f(\boldsymbol{x}, \boldsymbol{\theta}) + \epsilon, \epsilon \sim \mathcal{N}(0, \sigma^2)$$

Non-linear regression

$$f(\boldsymbol{x}, \boldsymbol{\theta}) = \boldsymbol{\theta}^\top \phi(\boldsymbol{x})$$

$$y = f(\boldsymbol{x}, \boldsymbol{\theta}) + \epsilon, \epsilon \sim \mathcal{N}(0, \sigma^2)$$

Probabilistic modelling: linear regression

Step 1: making assumptions about the output generation process

$$y = \boldsymbol{\theta}^\top \phi(\boldsymbol{x}) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2)$$

- $\boldsymbol{\theta}$ is the model parameter
- $\phi(\boldsymbol{x})$ is a pre-defined feature mapping (e.g., polynomial features)

Probabilistic modelling: linear regression

Step 1: making assumptions about the output generation process

$$y = \boldsymbol{\theta}^\top \phi(\boldsymbol{x}) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2)$$

- $\boldsymbol{\theta}$ is the model parameter
- $\phi(\boldsymbol{x})$ is a pre-defined feature mapping (e.g., polynomial features)

Probabilistic formulation:

- The distribution of y given $\boldsymbol{x}$ under model assumption:

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

- Likelihood of $\boldsymbol{\theta}$ given observed data $(\boldsymbol{x}, y)$:

$$\ell(\boldsymbol{\theta}) = p(y|\boldsymbol{x}, \boldsymbol{\theta})$$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**

Idea of MLE: for datapoints $(\boldsymbol{x}, y)$ sampled from $p_{data}(\boldsymbol{x}, y)$

- We want to find $\boldsymbol{\theta}^*$ such that $p(y|\boldsymbol{x}, \boldsymbol{\theta}^*) \approx p_{data}(y|\boldsymbol{x})$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**

Idea of MLE: for datapoints $(\boldsymbol{x}, y)$ sampled from $p_{data}(\boldsymbol{x}, y)$

- We want to find $\boldsymbol{\theta}^*$ such that $p(y|\boldsymbol{x}, \boldsymbol{\theta}^*) \approx p_{data}(y|\boldsymbol{x})$
- We need to measure the "closeness" of the two distributions $\Rightarrow$ use the KL divergence

$$\mathrm{KL}[p_{data}(y|\boldsymbol{x})||p(y|\boldsymbol{x}, \boldsymbol{\theta})] = \mathbb{E}_{p_{data}(y|\boldsymbol{x})}\left[\log \frac{p_{data}(y|\boldsymbol{x})}{p(y|\boldsymbol{x}, \boldsymbol{\theta})}\right]$$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**

Idea of MLE: for datapoints $(\boldsymbol{x}, y)$ sampled from $p_{data}(\boldsymbol{x}, y)$

- We want to find $\boldsymbol{\theta}^*$ such that $p(y|\boldsymbol{x}, \boldsymbol{\theta}^*) \approx p_{data}(y|\boldsymbol{x})$
- We need to measure the "closeness" of the two distributions $\Rightarrow$ use the KL divergence

$$\mathrm{KL}[p_{data}(y|\boldsymbol{x})||p(y|\boldsymbol{x}, \boldsymbol{\theta})] = \mathbb{E}_{p_{data}(y|\boldsymbol{x})}\left[\log \frac{p_{data}(y|\boldsymbol{x})}{p(y|\boldsymbol{x}, \boldsymbol{\theta})}\right]$$

- We want this KL to be small for all $\boldsymbol{x}$ sampled from $p_{data}(\boldsymbol{x})$

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathbb{E}_{p_{data}(\boldsymbol{x})}[\mathrm{KL}[p_{data}(y|\boldsymbol{x})||p(y|\boldsymbol{x}, \boldsymbol{\theta})]]$$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**

Idea of MLE: for datapoints $(\boldsymbol{x}, y)$ sampled from $p_{data}(\boldsymbol{x}, y)$

- We want to find $\boldsymbol{\theta}^*$ such that $p(y|\boldsymbol{x}, \boldsymbol{\theta}^*) \approx p_{data}(y|\boldsymbol{x})$
- We need to measure the "closeness" of the two distributions $\Rightarrow$ use the KL divergence

$$\mathrm{KL}[p_{data}(y|\boldsymbol{x})||p(y|\boldsymbol{x}, \boldsymbol{\theta})] = \mathbb{E}_{p_{data}(y|\boldsymbol{x})}\left[\log \frac{p_{data}(y|\boldsymbol{x})}{p(y|\boldsymbol{x}, \boldsymbol{\theta})}\right]$$

- We want this KL to be small for all $\boldsymbol{x}$ sampled from $p_{data}(\boldsymbol{x})$

$$\Leftrightarrow \quad \boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{p_{data}(\boldsymbol{x}, \mathbf{y})}[\log p(y|\boldsymbol{x}, \boldsymbol{\theta})]$$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**

Idea of MLE: for datapoints $(\boldsymbol{x}, y)$ sampled from $p_{data}(\boldsymbol{x}, y)$

- We want to find $\boldsymbol{\theta}^*$ such that $p(y|\boldsymbol{x}, \boldsymbol{\theta}^*) \approx p_{data}(y|\boldsymbol{x})$
- We need to measure the "closeness" of the two distributions $\Rightarrow$ use the KL divergence

$$\mathrm{KL}[p_{data}(y|\boldsymbol{x})||p(y|\boldsymbol{x}, \boldsymbol{\theta})] = \mathbb{E}_{p_{data}(y|\boldsymbol{x})}\left[\log \frac{p_{data}(y|\boldsymbol{x})}{p(y|\boldsymbol{x}, \boldsymbol{\theta})}\right]$$

- We want this KL to be small for all $\boldsymbol{x}$ sampled from $p_{data}(\boldsymbol{x})$
  Estimate using dataset $\mathcal{D} = \{(\boldsymbol{x}_n, y_n)\}_{n=1}^N$ from $p_{data}(\boldsymbol{x}, y)$:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \log p(y_n|\boldsymbol{x}_n, \boldsymbol{\theta})$$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**
MLE: find $\boldsymbol{\theta}^*$ by

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \log p(y_n | \boldsymbol{x}_n, \boldsymbol{\theta})$$

- We assumed the probabilistic model to be

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

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**
MLE: find $\boldsymbol{\theta}^*$ by

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \log p(y_n | \boldsymbol{x}_n, \boldsymbol{\theta})$$

- We assumed the probabilistic model to be

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

$$\Rightarrow \boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \log \mathcal{N}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x}), \sigma^2)$$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**
MLE: find $\boldsymbol{\theta}^*$ by

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \log p(y_n | \boldsymbol{x}_n, \boldsymbol{\theta})$$

- We assumed the probabilistic model to be

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

$$\begin{aligned}
\Rightarrow \boldsymbol{\theta}^* &= \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \log \mathcal{N}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x}), \sigma^2) \\
&= \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} -\frac{1}{2\sigma^2} ||y_n - \boldsymbol{\theta}^\top \phi(\boldsymbol{x}_n)||_2^2 + \text{const}
\end{aligned}$$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**
MLE: find $\boldsymbol{\theta}^*$ by

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \log p(y_n | \boldsymbol{x}_n, \boldsymbol{\theta})$$

- We assumed the probabilistic model to be

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

$$\begin{aligned}
\Rightarrow \boldsymbol{\theta}^* &= \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \log \mathcal{N}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x}), \sigma^2) \\
&= \arg\max_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} -\frac{1}{2\sigma^2} ||y_n - \boldsymbol{\theta}^\top \phi(\boldsymbol{x}_n)||_2^2 + \text{const} \\
&= \arg\min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \frac{1}{2\sigma^2} ||y_n - \boldsymbol{\theta}^\top \phi(\boldsymbol{x}_n)||_2^2
\end{aligned}$$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**

$$\arg\min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \frac{1}{2\sigma^2} ||y_n - \boldsymbol{\theta}^\top \phi(\boldsymbol{x}_n)||_2^2$$

Writing the objective in matrix form:
$\boldsymbol{\Phi} = (\phi(x_1), ..., \phi(x_N))^\top, \mathbf{y} = (y_1, ..., y_N)^\top$

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}), \quad L(\boldsymbol{\theta}) = \frac{1}{2\sigma^2} ||\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta}||_2^2$$

- Gradient of the loss $\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta})$:

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**

$$\arg\min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \frac{1}{2\sigma^2} ||y_n - \boldsymbol{\theta}^\top \phi(\boldsymbol{x}_n)||_2^2$$

Writing the objective in matrix form:
$\boldsymbol{\Phi} = (\phi(x_1), ..., \phi(x_N))^\top, \mathbf{y} = (y_1, ..., y_N)^\top$

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}), \quad L(\boldsymbol{\theta}) = \frac{1}{2\sigma^2} ||\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta}||_2^2$$

- Gradient of the loss $\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta})$:

$$\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) = \frac{1}{\sigma^2} \boldsymbol{\Phi}^\top (\boldsymbol{\Phi}\boldsymbol{\theta} - \mathbf{y})$$

- Setting $\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) = 0$:

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**

$$\arg\min_{\boldsymbol{\theta}} \frac{1}{N} \sum_{(\boldsymbol{x}_n, y_n) \in \mathcal{D}} \frac{1}{2\sigma^2} ||y_n - \boldsymbol{\theta}^\top \phi(\boldsymbol{x}_n)||_2^2$$

Writing the objective in matrix form:
$\boldsymbol{\Phi} = (\phi(x_1), ..., \phi(x_N))^\top, \mathbf{y} = (y_1, ..., y_N)^\top$

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}), \quad L(\boldsymbol{\theta}) = \frac{1}{2\sigma^2} ||\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta}||_2^2$$

- Gradient of the loss $\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta})$:

$$\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) = \frac{1}{\sigma^2} \boldsymbol{\Phi}^\top (\boldsymbol{\Phi}\boldsymbol{\theta} - \mathbf{y})$$

- Setting $\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) = 0$:

$$\Rightarrow \frac{1}{\sigma^2} \boldsymbol{\Phi}^\top \boldsymbol{\Phi}\boldsymbol{\theta}^* = \frac{1}{\sigma^2} \boldsymbol{\Phi}^\top \mathbf{y} \quad \Rightarrow \boldsymbol{\theta}^* = (\boldsymbol{\Phi}^\top \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^\top \mathbf{y}$$

Probabilistic modelling: linear regression

Step 3: check if $p(y|\boldsymbol{x}, \boldsymbol{\theta}^*)$ fits $p_{data}(y|\boldsymbol{x})$ well

Typical approaches:

- Cross validation
- Model selection with marginal likelihood

Probabilistic modelling: linear regression

Step 3: check if $p(y|\boldsymbol{x}, \boldsymbol{\theta}^*)$ fits $p_{data}(y|\boldsymbol{x})$ well

Typical approaches:

- Cross validation
- Model selection with marginal likelihood

If model fit is bad:

- Try another set of features $\phi'(\boldsymbol{x}) \neq \phi(\boldsymbol{x})$
- Use other classes of models other than linear regression

Probabilistic modelling: linear regression

Step 4: using $p(y|\boldsymbol{x}, \boldsymbol{\theta}^*)$ to make predictions
Assume new test input $\boldsymbol{x}_{test}$:

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

$$\Rightarrow \quad p(y_{test}|\boldsymbol{x}_{test}, \boldsymbol{\theta}^*) = \mathcal{N}(\mathbf{y}^\top \mathbf{\Phi} (\mathbf{\Phi}^\top \mathbf{\Phi})^{-1} \phi(\boldsymbol{x}_{test}), \sigma^2)$$

Probabilistic modelling: logistic regression

Step 1: making assumptions about the output generation process

$$y = \begin{cases} 1, & \text{with probability } \rho \\ 0, & \text{with probability } 1-\rho \end{cases} \quad , \quad \rho = sigmoid(\boldsymbol{\theta}^\top \phi(\boldsymbol{x}))$$

Probabilistic formulation:

- The distribution of y given $\boldsymbol{x}$ under model assumption:

$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \text{Bern}(sigmoid(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})))$$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**
MLE: find $\boldsymbol{\theta}^*$ by

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{x},\mathbf{y}) \in \mathcal{D}} \log p(y|\boldsymbol{x}, \boldsymbol{\theta})$$

- We assumed the probabilistic model to be

$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \mathrm{Bern}(sigmoid(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})))$$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**
MLE: find $\boldsymbol{\theta}^*$ by

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{x},\mathbf{y})\in\mathcal{D}} \log p(y|\boldsymbol{x}, \boldsymbol{\theta})$$

- We assumed the probabilistic model to be

$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \text{Bern}(sigmoid(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})))$$

$$\Rightarrow \boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{x},\mathbf{y})\in\mathcal{D}} \log \text{Bern}(sigmoid(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})))$$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**
MLE: find $\boldsymbol{\theta}^*$ by

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{x},\mathbf{y})\in\mathcal{D}} \log p(y|\boldsymbol{x}, \boldsymbol{\theta})$$

- We assumed the probabilistic model to be

$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \text{Bern}(sigmoid(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})))$$

$$\Rightarrow \boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{x},\mathbf{y})\in\mathcal{D}} \log \text{Bern}(sigmoid(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})))$$

$$= \arg\max_{\boldsymbol{\theta}} \frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{x},\mathbf{y})\in\mathcal{D}} y \log \hat{y}(\boldsymbol{x}; \boldsymbol{\theta}) + (1-y)\log(1-\hat{y}(\boldsymbol{x}; \boldsymbol{\theta})),$$

$$\hat{y}(\boldsymbol{x}; \boldsymbol{\theta}) = sigmoid(\boldsymbol{\theta}^\top \phi(\boldsymbol{x}))$$

Probabilistic modelling: linear regression

Step 2: estimating $\boldsymbol{\theta}$ using **maximum likelihood estimation (MLE)**

$$\arg\max_{\boldsymbol{\theta}} L(\boldsymbol{\theta}), \quad L(\boldsymbol{\theta}) = \frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{x},\mathbf{y})\in\mathcal{D}} y \log \hat{y}(\boldsymbol{x};\boldsymbol{\theta}) + (1-y)\log(1-\hat{y}(\boldsymbol{x};\boldsymbol{\theta})),$$

$$\hat{y}(\boldsymbol{x};\boldsymbol{\theta}) = sigmoid(\boldsymbol{\theta}^\top \phi(\boldsymbol{x}))$$

- Gradient of the loss $\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta})$:

$$\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) = \frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{x},\mathbf{y})\in\mathcal{D}} [y - \hat{y}(\boldsymbol{x};\boldsymbol{\theta})]\phi(\boldsymbol{x})$$

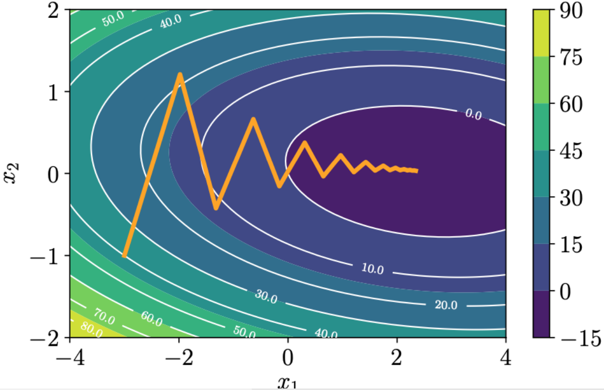
No analytic solutions!

Gradient descent based optimisation

Algorithm: Gradient Descent (gradient ascent in MLE case)

Define **starting point** $\boldsymbol{\theta}_0$, sequence of **step sizes** γ_t, set $t \leftarrow 0$.

1. Set $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \gamma_t \nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}_t)$, $t \leftarrow t + 1$
2. Repeat 1 until stopping criterion.



Probabilistic modelling: logistic regression

Step 3: check if $p(y|\boldsymbol{x}, \boldsymbol{\theta}^*)$ fits $p_{data}(y|\boldsymbol{x})$ well

Typical approaches:

- Cross validation
- Model selection with marginal likelihood

If model fit is bad:

- Try another set of features $\phi'(\boldsymbol{x}) \neq \phi(\boldsymbol{x})$
- Use other classes of models other than logistic regression

Probabilistic modelling: logistic regression

Step 4: using $p(y|\boldsymbol{x}, \boldsymbol{\theta}^*)$ to make predictions
Assume new test input $\boldsymbol{x}_{test}$:
$\boldsymbol{\theta}^*$ obtained by gradient descent

$$\Rightarrow \quad p(y_{test}|\boldsymbol{x}_{test}, \boldsymbol{\theta}^*) = \text{Bern}(Sigmoid((\boldsymbol{\theta}^*)^\top \phi(\boldsymbol{x}_{test})))$$

Probabilistic modelling & MLE: summary

Have you ever wondered about the following questions:

- Why using ℓ_2 loss in many regression problems?
  A: We assume the model to be $p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x}), \sigma^2)$, and fit $\boldsymbol{\theta}$ using MLE

Probabilistic modelling & MLE: summary

Have you ever wondered about the following questions:

- Why using ℓ_2 loss in many regression problems?
  A: We assume the model to be $p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x}), \sigma^2)$, and fit $\boldsymbol{\theta}$ using MLE
- Where does the cross-entropy loss come from?
  A: It comes from MLE, and in binary classification using $p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \text{Bern}(sigmoid(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})))$

Probabilistic modelling & MLE: summary

Have you ever wondered about the following questions:

- Why using ℓ_2 loss in many regression problems?
  A: We assume the model to be $p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}^\top \phi(\boldsymbol{x}), \sigma^2)$, and fit $\boldsymbol{\theta}$ using MLE
- Where does the cross-entropy loss come from?
  A: It comes from MLE, and in binary classification using $p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \text{Bern}(sigmoid(\boldsymbol{\theta}^\top \phi(\boldsymbol{x})))$
- What is a good principle for choosing a good loss function?
  A: Build a probabilistic model for the data generation process, and fit the parameters using MLE (or MAP, posterior inference)

Exercises

Finish relevant exercises in the exercise sheet
Next lecture: convergence of gradient descent
Pre-requisite knowledge: Eigen-decomposition
(See e.g., `https://youtu.be/xgZ8oK9Wxzg` or search relevant videos from 3Blue1Brown)