

Probabilistic view

Machine Learning and Data Mining, 2022

Artem Maevskiy

National Research University Higher School of Economics



LAMBD A • HSE

September 10, 2022

Why probability?

- ▶ Machine learning often deals with random quantities
- ▶ Sources of uncertainty:
 - Inherent stochasticity of the system being modelled
 - Lack of information
 - Incomplete modelling (discarding information for the sake of simplicity, computability, etc.)

Probability recap



Probability

- ▶ Frequentist:
 - **relative frequency** of occurrence of an experiment's outcome, when repeating the experiment

Example: coin toss

Toss a coin N times (H – number of 'heads', T – number of 'tails')
Probability:

$$P(\text{'heads'}) = \lim_{N \rightarrow +\infty} \frac{H}{N}$$

$$P(\text{'tails'}) = \lim_{N \rightarrow +\infty} \frac{T}{N} = 1 - P(\text{'heads'})$$

Probability

- ▶ Frequentist:
 - **relative frequency** of occurrence of an experiment's outcome, when repeating the experiment
- ▶ Bayesian:
 - **degree of belief**

Example: doctor analyzes a patient and says that the patient has 40% probability of having the flu (we can't "repeat" this patient)

Random variable

- ▶ A variable that can take values randomly
- ▶ Can think of it as variable enumerating possible outcomes of a random event
 - E.g., for the coin toss:

$$x = \begin{cases} 0, & \text{'heads'} \\ 1, & \text{'tails'} \end{cases}$$

Random variable

- ▶ A variable that can take values randomly
- ▶ Can think of it as variable enumerating possible outcomes of a random event
 - E.g., for the coin toss:

$$x = \begin{cases} 0, & \text{'heads'} \\ 1, & \text{'tails'} \end{cases}$$

- A bit more complex example – number of coin tosses with 'heads' out of N tosses total:

The diagram shows the equation $n = n \in \{0, \dots, N\}$. A red arrow points from the text "random variable" to the first n in the equation. A blue arrow points from the text "possible value it may take" to the n in the set notation.

random variable $n = n \in \{0, \dots, N\}$ **possible value it may take**

Probability mass function (PMF)

- ▶ Defined for discrete variables
- ▶ Equals to probability for the variable x to take a given value x :

$$P(x = x)$$

- ▶ or just $P(x)$ – omitting the name of the variable
- ▶ Joint probability distribution – probability for several random variables to take some particular values simultaneously:

$$P(x = x, y = y) \equiv P(x, y)$$

- ▶ PMF must:
 - be defined on all possible states of the variable
 - take values in the $[0, 1]$ interval
 - sum to 1 over all possible outcomes (probability for anything to happen)

Probability density function (PDF)

- ▶ Defined for continuous variables
- ▶ Equals to:

$$p(x) = \lim_{\delta x \rightarrow 0} P(x \in (x, x + \delta x)) / \delta x$$

- ▶ PDF must:
 - be defined on all possible states of the variable
 - be ≥ 0 (can be higher than 1 though)
 - integrate to 1 over all possible outcomes (probability for anything to happen):

$$\int_{\mathbf{X}} p(x) dx = 1$$

Expectation and variance

- Expectation:

For a discrete variable

$$\mathbb{E}[x] = \sum_x xP(x)$$

For a continuous variable

$$\mathbb{E}[x] = \int_x xp(x)dx$$

- Meaning: **average outcome**

- Variance:

$$\text{Var}[x] = \mathbb{E}[(x - \mathbb{E}[x])^2] = \mathbb{E}[x^2] - (\mathbb{E}[x])^2$$

- Meaning: **spread of the outcomes**

Some distributions

- ▶ Uniform[a, b]:

$$p(x) = \frac{1}{b - a} = \textit{const}$$

- ▶ Binomial:

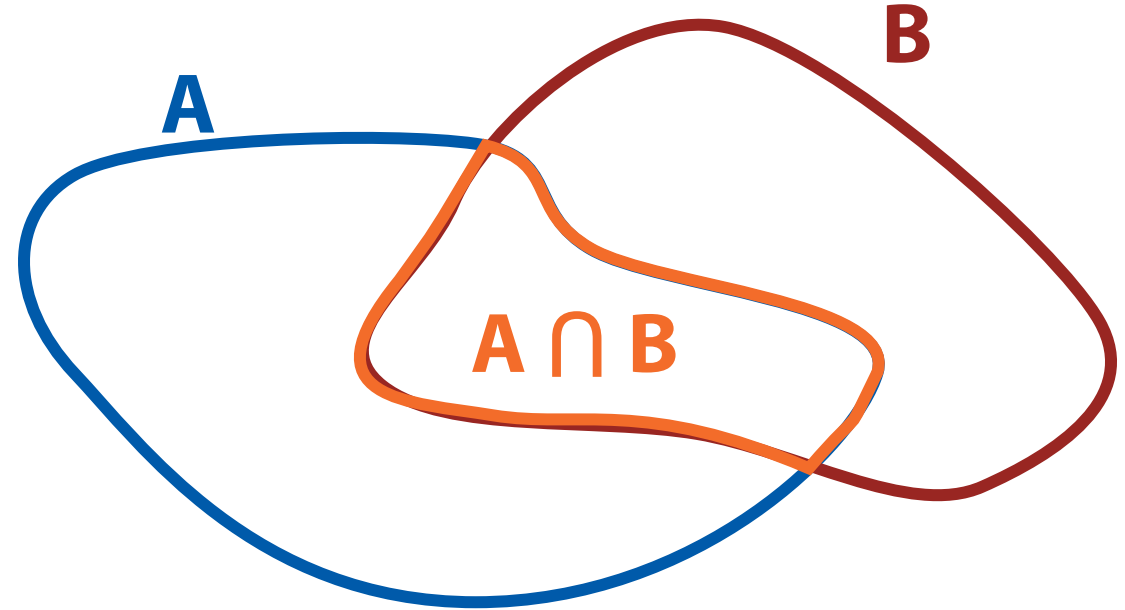
$$P(k) = \frac{n!}{k! (n - k)!} p^k (1 - p)^{n-k}$$

- ▶ Normal distribution:

$$p(x) \equiv \mathcal{N}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Conditional probability

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$



For PDF:
$$p(x|y) = \frac{p(x, y)}{p(y)}$$

– i.e. we're renormalizing $p(x, y)$ as a distribution of only x for some fixed y

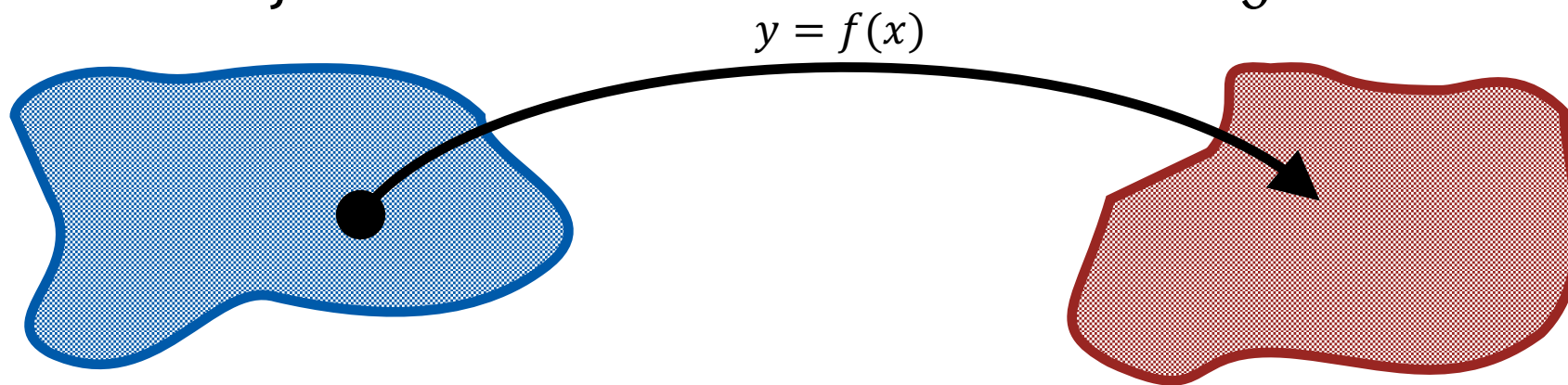
Probabilistic view on supervised learning



Problem setup

\mathcal{X} — a set of objects

\mathcal{Y} — a set of targets



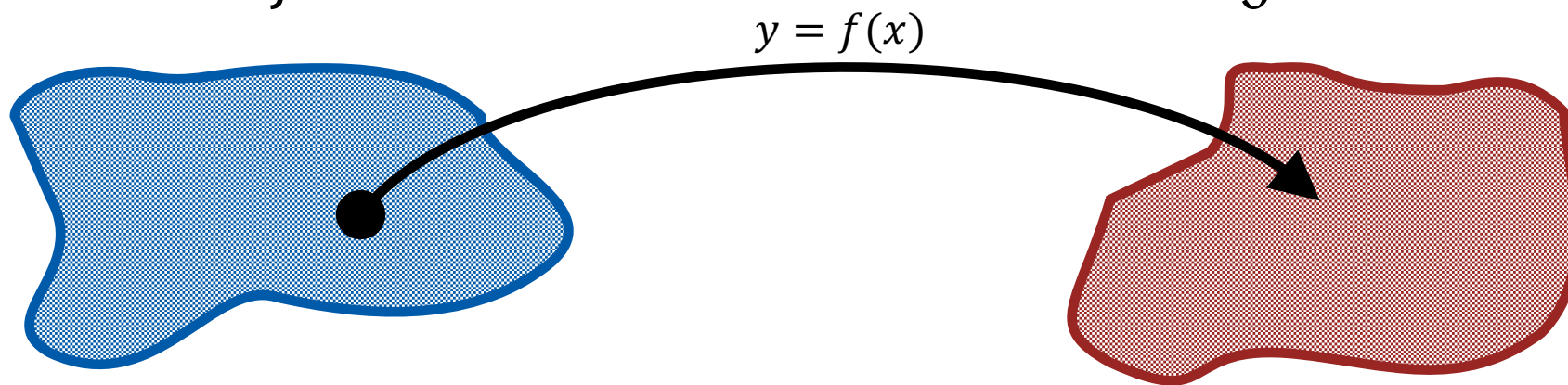
A dataset: $D = \{(x_i, y_i) : i = 1, 2, \dots, N\}$

$$x_i \in \mathcal{X}, \quad y_i = f(x_i) \in \mathcal{Y}$$

Problem setup

\mathcal{X} — a set of objects

\mathcal{Y} — a set of targets



A dataset: $D = \{(x_i, y_i) : i = 1, 2, \dots, N\}$

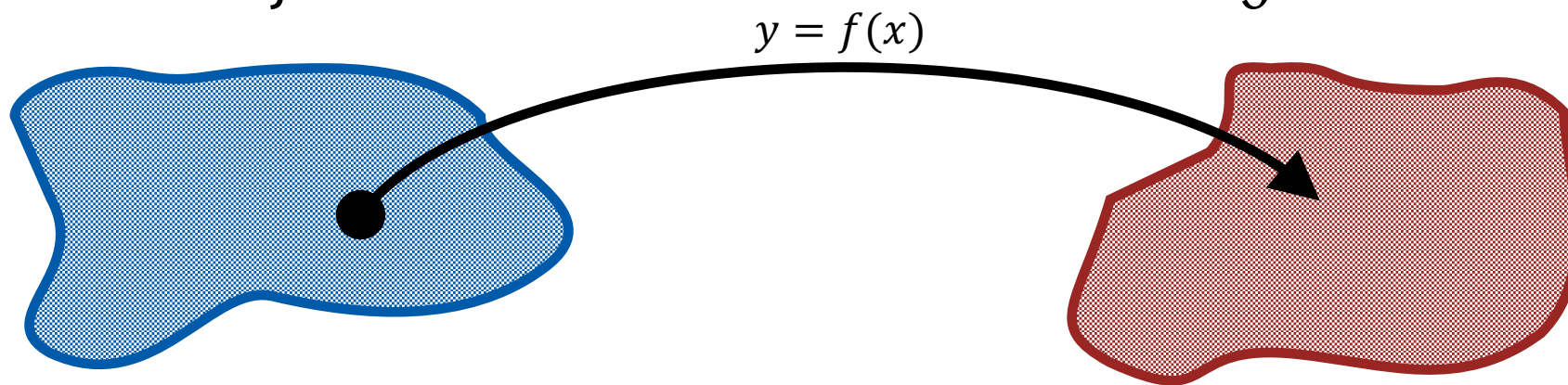
$$x_i \in \mathcal{X}, \quad y_i = f(x_i) \in \mathcal{Y}$$

– There's some **underlying probability distribution** $p(x, y)$

Problem setup

\mathcal{X} — a set of objects

\mathcal{Y} — a set of targets



A dataset: $D = \{(x_i, y_i) : i = 1, 2, \dots, N\}$

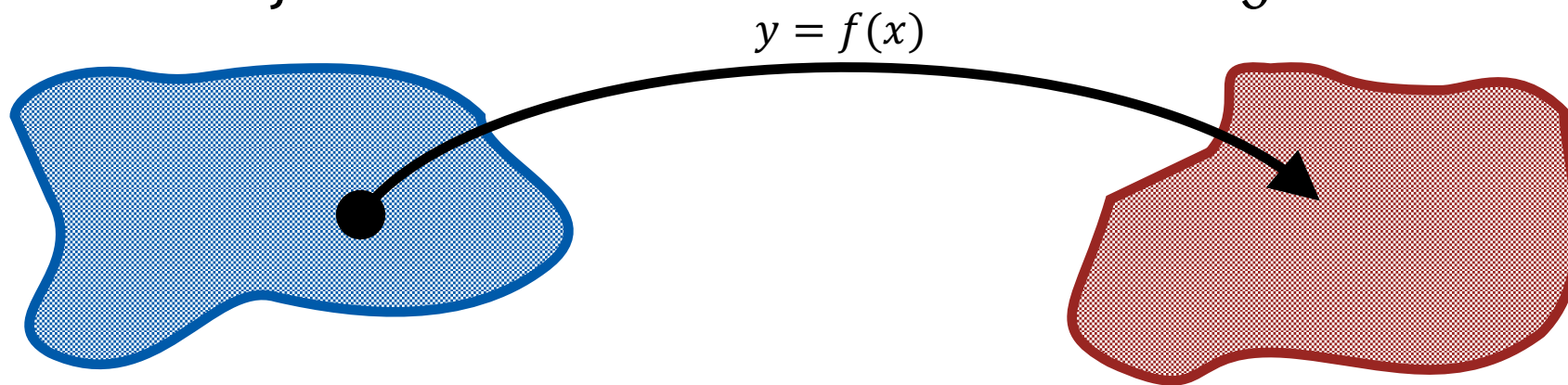
$$x_i \in \mathcal{X}, \quad y_i = f(x_i) \in \mathcal{Y}$$

- There's some **underlying probability distribution** $p(x, y)$
- (x_i, y_i) are drawn from $p(x, y)$, **independently** for each i

Problem setup

\mathcal{X} — a set of objects

\mathcal{Y} — a set of targets




A dataset: $D = \{(x_i, y_i) : i = 1, 2, \dots, N\}$

$$x_i \in \mathcal{X}, \quad y_i = f(x_i) \in \mathcal{Y}$$

- There's some **underlying probability distribution** $p(x, y)$
- (x_i, y_i) are drawn from $p(x, y)$, **independently** for each i
- Can also say that for a given x_i , the target y_i is drawn from $p(y|x)$

Deterministic and stochastic components

- ▶ With this view, we can separate deterministic and stochastic parts of the true mapping:

$$y|x = f(x) + \varepsilon(x)$$


Deterministic part (expectation)

$$f(x) = \mathbb{E}[y|x] \equiv \int y \cdot p(y|x) dy$$

Random part (noise)

$$\varepsilon(x) = y|x - \mathbb{E}[y|x]$$

Probabilistic model

Let's make an assumption about data:

$$y|x = f(x) + \varepsilon$$

Assume that **label noise** is **normally distributed**:

$$\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$$

Probabilistic model

Let's make an assumption about data:

$$y|x = f(x) + \varepsilon$$

Assume that **label noise** is **normally distributed**:

$$\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$$

This means, that labels are also normally distributed, for a given point x :

$$y|x \sim \mathcal{N}(f(x), \sigma_\varepsilon^2)$$

Probabilistic model

Let's make an assumption about data:

$$y|x = f(x) + \varepsilon$$

Assume that **label noise** is **normally distributed**:

$$\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$$

This means, that labels are also normally distributed, for a given point x :

$$y|x \sim \mathcal{N}(f(x), \sigma_\varepsilon^2)$$

We want our model $\hat{f}_\theta(x)$ to fit the true dependence $f(x)$, i.e. we **define a probabilistic model**:

$$y|x \sim \mathcal{N}(\hat{f}_\theta(x), \sigma_\varepsilon^2)$$

Probabilistic model

Our model can be fitted with the **maximum likelihood** approach:

$$L = \prod_{i=1 \dots N} \mathcal{N}(y_i | \hat{f}_{\theta}(x_i), \sigma_{\varepsilon}^2) \rightarrow \max_{\theta}$$



"The observed data is most probable"

Probabilistic model

Our model can be fitted with the **maximum likelihood** approach:

$$L = \prod_{i=1 \dots N} \mathcal{N}(y_i | \hat{f}_\theta(x_i), \sigma_\varepsilon^2) \rightarrow \max_{\theta}$$



"The observed data is most probable"

Max. likelihood = min. negative log likelihood

$$-\log L = - \sum_{i=1 \dots N} \log \mathcal{N}(y_i | \hat{f}_\theta(x_i), \sigma_\varepsilon^2)$$

Probabilistic model

Our model can be fitted with the **maximum likelihood** approach:

$$L = \prod_{i=1 \dots N} \mathcal{N}(y_i | \hat{f}_{\theta}(x_i), \sigma_{\varepsilon}^2) \rightarrow \max_{\theta}$$

"The observed data is most probable"



Max. likelihood = min. negative log likelihood

$$\begin{aligned} -\log L &= - \sum_{i=1 \dots N} \log \mathcal{N}(y_i | \hat{f}_{\theta}(x_i), \sigma_{\varepsilon}^2) \\ &= - \sum_{i=1 \dots N} \left[\log \exp \left(- \frac{(y_i - \hat{f}_{\theta}(x_i))^2}{2\sigma_{\varepsilon}^2} \right) - \log \sqrt{2\pi\sigma_{\varepsilon}^2} \right] \end{aligned}$$

Probabilistic model

Our model can be fitted with the **maximum likelihood** approach:

$$L = \prod_{i=1 \dots N} \mathcal{N}(y_i | \hat{f}_{\theta}(x_i), \sigma_{\varepsilon}^2) \rightarrow \max_{\theta}$$

"The observed data is most probable"



Max. likelihood = min. negative log likelihood

$$\begin{aligned} -\log L &= - \sum_{i=1 \dots N} \log \mathcal{N}(y_i | \hat{f}_{\theta}(x_i), \sigma_{\varepsilon}^2) \\ &= - \sum_{i=1 \dots N} \left[\log \exp \left(- \frac{(y_i - \hat{f}_{\theta}(x_i))^2}{2\sigma_{\varepsilon}^2} \right) - \log \sqrt{2\pi\sigma_{\varepsilon}^2} \right] \\ &= C \cdot \sum_{i=1 \dots N} (y_i - \hat{f}_{\theta}(x_i))^2 + const \rightarrow \min_{\theta} \end{aligned}$$

Probabilistic model

Our model can be fitted with the **maximum likelihood** approach:

$$L = \prod_{i=1 \dots N} \mathcal{N}(y_i | \hat{f}_\theta(x_i), \sigma_\varepsilon^2) \rightarrow \max_{\theta}$$

"The observed data is most probable"

Max. likelihood = min. negative log likelihood

$$-\log L = - \sum_{i=1 \dots N} \log \mathcal{N}(y_i | \hat{f}_\theta(x_i), \sigma_\varepsilon^2)$$

$$= - \sum_{i=1 \dots N} \left[\log \exp \left(- \frac{(y_i - \hat{f}_\theta(x_i))^2}{2\sigma_\varepsilon^2} \right) - \log \sqrt{2\pi\sigma_\varepsilon^2} \right]$$

**MSE loss \Leftrightarrow Prob. model
with normal label noise!**

$$= C \cdot \sum_{i=1 \dots N} (y_i - \hat{f}_\theta(x_i))^2 + const \rightarrow \min_{\theta}$$

Summary

- ▶ Machine Learning often deals with **randomness** (intrinsic, lack of information, incomplete modelling)
- ▶ Supervised learning problems can be posed in the probabilistic context
- ▶ The mapping between features and labels can be decomposed into **deterministic** and **stochastic** parts
- ▶ There's a **probabilistic model** behind the loss function
- ▶ Food for thought: what probabilistic model would correspond to minimizing MAE loss: $\frac{1}{N} \sum_i |y_i - \hat{f}(x_i)|$?

Thank you!



amaevskij@hse.ru



SiLiKhon

Artem Maevskiy