



Introduction to **Machine Learning and Data Mining** (Học máy và Khai phá dữ liệu)

Khoat Than

School of Information and Communication Technology
Hanoi University of Science and Technology

Content

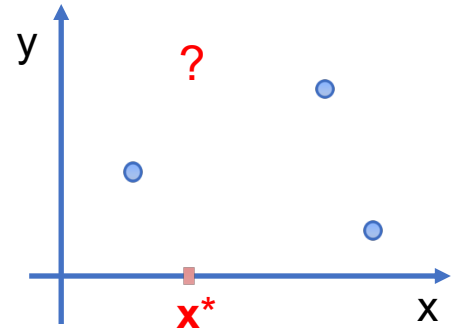
- Introduction to Machine Learning & Data Mining
- Unsupervised learning
- Supervised learning
- **Probabilistic modeling**
- Practical advice

Why probabilistic modeling?

- Inferences from data are intrinsically **uncertain**.
(suy diễn từ dữ liệu thường không chắc chắn)
- Probability theory: *model uncertainty* instead of ignoring it!
- Inference or prediction can be done by using **probabilities**.
- Applications: Machine learning, Data Mining, Computer Vision, NLP, Bioinformatics, ...
- The goal of this lecture
 - Overview about probabilistic modeling
 - Key concepts
 - Application to classification

Data

- Let $\mathbf{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_M, y_M)\}$ be a dataset with M instances.
 - Each \mathbf{x}_i is a vector in an n -dimensional space, e.g., $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{in})^T$. Each dimension represents an attribute.
 - y is the output (response), univariate
- **Prediction:** given data \mathbf{D} , what can we say about y^* at an unseen input \mathbf{x}^* ?



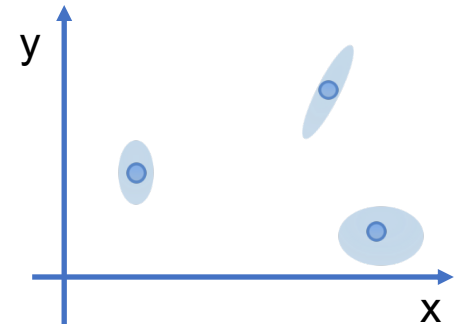
- To make predictions, we need to make **assumptions**
- A **model H (mô hình)** encodes these assumptions, and often depends on some parameters θ , e.g.,

$$y = f(x|\theta)$$

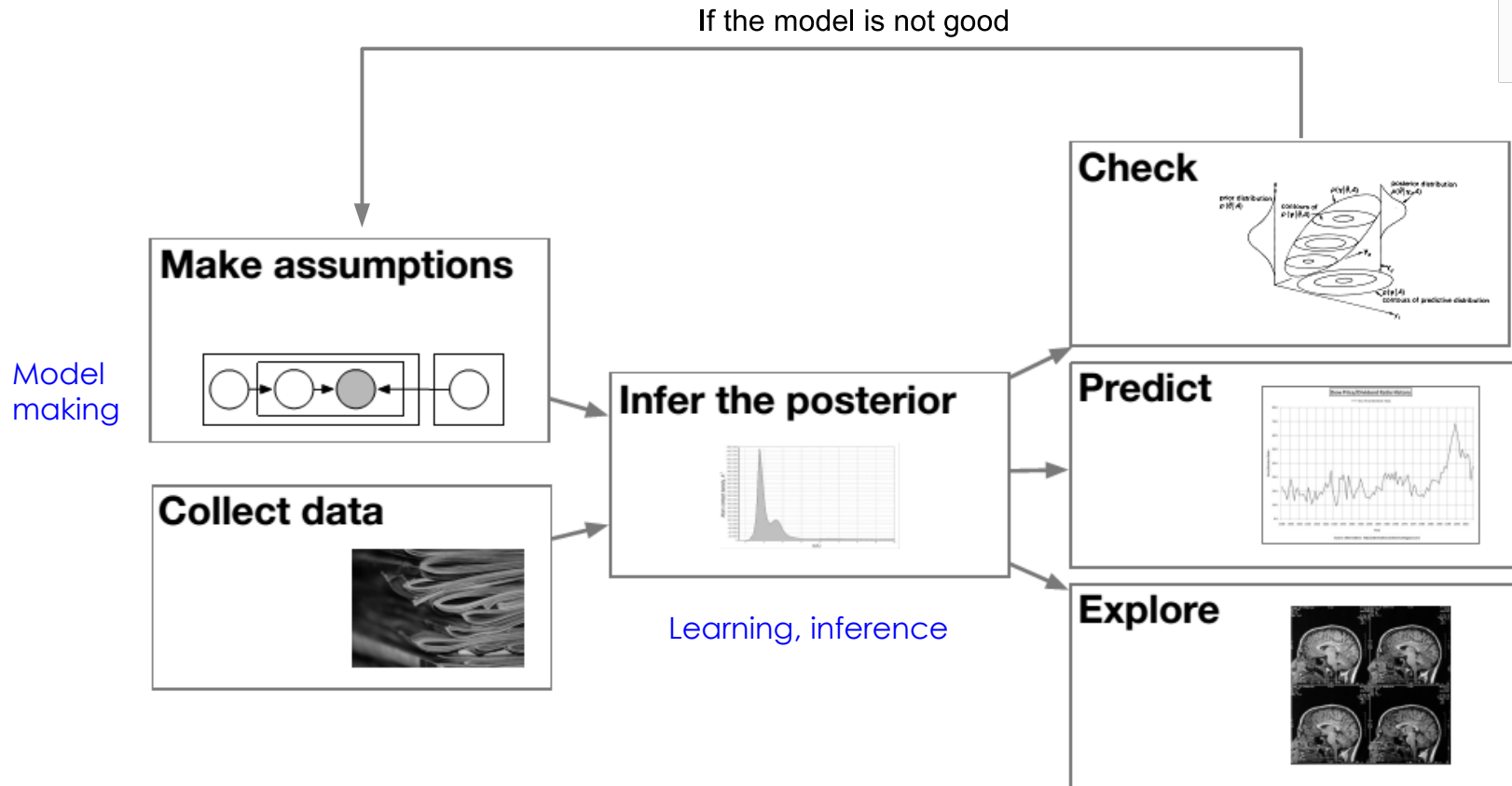
- **Learning** (estimation) is to find an H from a given \mathbf{D} .

Uncertainty

- Uncertainty appears in any step
 - Measurement uncertainty (**D**)
 - Parameter uncertainty (**θ**)
 - Uncertainty regarding the correct model (**H**)
 - Measurement uncertainty
 - Uncertainty can occur in both inputs and outputs.
 - How to represent uncertainty?
- > Probability theory



The modeling process



[Blei, 2012]

Basics of Probability Theory

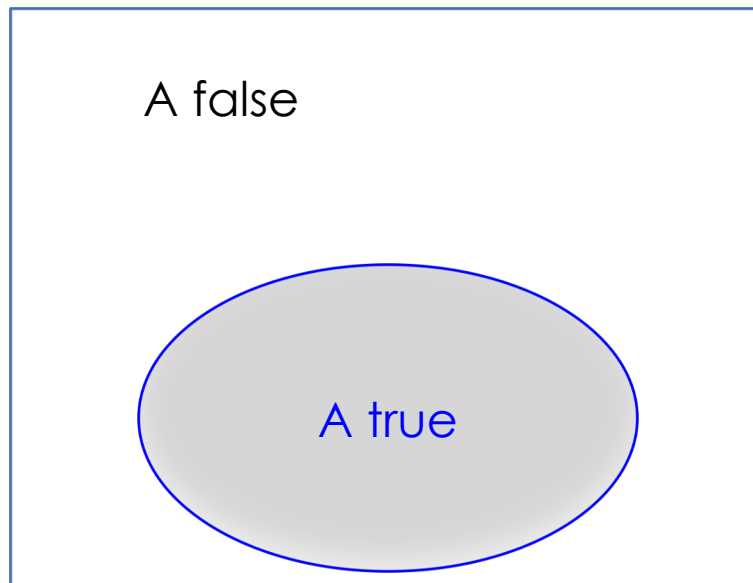
Basic concepts in Probability Theory

- Assume we do an experiment with random outcomes, e.g., tossing a die.
- *Space S of outcomes*: the set of all possible outcomes of an experiment
 - Ex: $S = \{1, 2, 3, 4, 5, 6\}$ for tossing a die
- *Event E* : a subset of the outcome space S .
 - Ex: $E = \{1\}$ the event that the die appears 1.
 - Ex: $E = \{1, 3, 5\}$ the event that the die appears odd.
- *Space W of events*: the space of all possible events
 - Ex: W contains all possible tosses
- *Random variable*: represents a random event, and has an associated probability of occurrence of that event.



Probability visualization

- **Probability** represents the likelihood/possibility that an event A occurs.
 - Denoted by $P(A)$.
- $P(A)$ is the proportion of the subspace that A is true.



← The event space
(space of all
possible outcomes
of the event A)

Binary random variables

- A binary (boolean) random variable can receive only value of either *True* or *False*.
- Some axioms:
 - $0 \leq P(A) \leq 1$
 - $P(\text{true}) = 1$
 - $P(\text{false}) = 0$
 - $P(A \text{ or } B) = P(A) + P(B) - P(A, B)$
- Some consequences:
 - $P(\text{not } A) = P(\sim A) = 1 - P(A)$
 - $P(A) = P(A, B) + P(A, \sim B)$

Multinomial random variables

- A multinomial random variable can receive one from K possible values of $\{v_1, v_2, \dots, v_k\}$.

$$P(A = v_i, A = v_j) = 0 \text{ if } i \neq j$$

$$P\left(\bigcup_{n=1}^m (A = v_n)\right) = \sum_{n=1}^m P(A = v_n)$$

$$P\left(\bigcup_{n=1}^k (A = v_n)\right) = \sum_{n=1}^k P(A = v_n) = 1$$

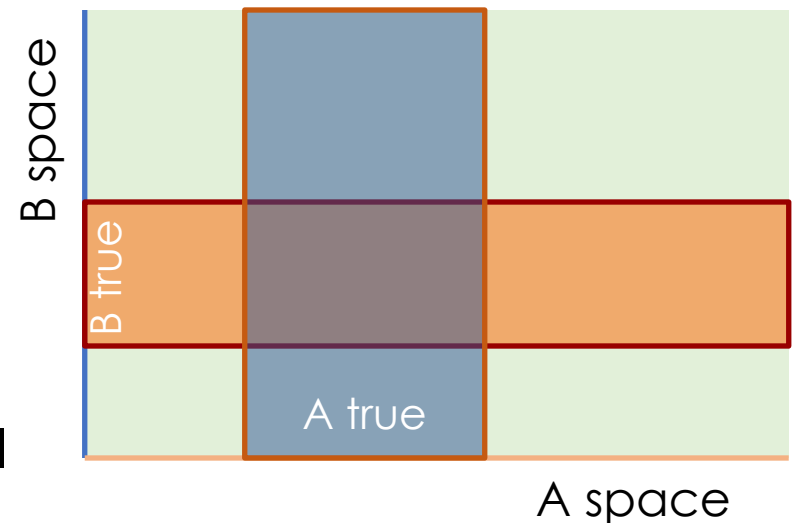
Joint probability (1)

■ Joint probability:

- The possibility of A and B that occur simultaneously.
- $P(A,B)$ is the proportion of the space in which both A and B are true.

■ Ex:

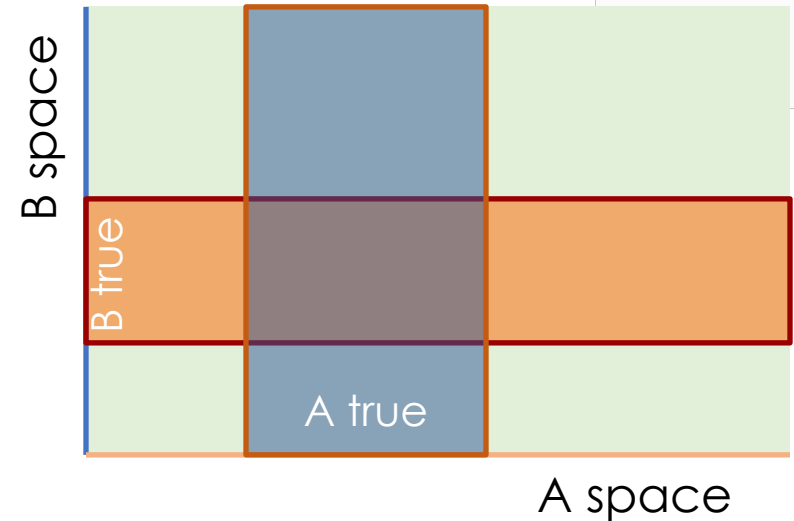
- A: I will play football tomorrow.
- B: John will not play football.
- $P(A,B)$: the probability that I will but John will not play football tomorrow.



Joint probability (2)

- Denote S_A the space of A.
- Denote S_B the space of B.
- Denote S_{AB} the space of (A, B).

$$S_{AB} = S_A \times S_B$$



- Then:

$$P(A,B) = |T_{AB}| / |S_{AB}|$$

- T_{AB} is the space in which both A and B are true.
- $|X|$ denotes the volume of the set X.

Conditional probability (1)

- Conditional probability:

- $P(A | B)$: the possibility that A happens given that B has already occurred.
- $P(A | B)$ is the proportion of the space in which A occurs, knowing that B is true.

- Ex:

- A: I will play football tomorrow.
- B: it will not rain tomorrow.
- $P(A | B)$: the probability that I will play football, provided that it will not rain tomorrow.

- What is different between joint and conditional probabilities?

Conditional probability (2)

- We have:

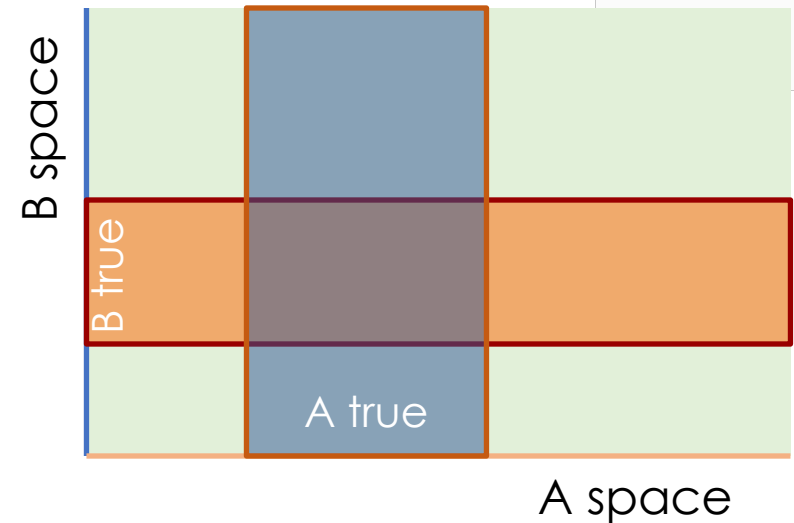
$$P(A | B) = \frac{P(A, B)}{P(B)}$$

- Some consequences:

$$P(A, B) = P(A | B) \cdot P(B)$$

$$P(A | B) + P(\sim A | B) = 1$$

$$\sum_{i=1}^k P(A = v_i | B) = 1$$

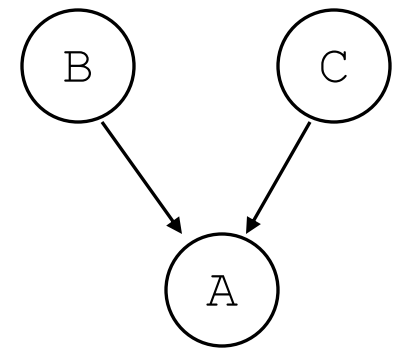


Conditional probability (3)

- $P(A | B, C)$ shows the probability of A given that B and C already has occurred.

■ Ex:

- A: I will wander over the near river tomorrow morning.
- B: it will be very nice tomorrow morning.
- C: I will wake up early tomorrow morning.
- $P(A | B, C)$: the probability that wander over the near river, provided that it will be very nice and I will wake up early tomorrow morning.



$P(A | B, C)$

Statistical independence (1)

- Two events A and B are called ***Statistically Independent*** if the the probability that A occurs does not change with respect to the occurrence of B.
 - $P(A | B) = P(A)$.
- Ex:
 - A: I will play football tomorrow.
 - B: the pacific ocean contains many fishes.
 - $P(A | B) = P(A)$: the fact that the pacific ocean contains many fishes does not affect my decision to play football tomorrow.

Statistical independence (2)

■ Assume $P(A | B) = P(A)$, we have:

- $P(\sim A | B) = P(\sim A)$
- $P(B | A) = P(B)$
- $P(A, B) = P(A) \cdot P(B)$
- $P(\sim A, B) = P(\sim A) \cdot P(B)$
- $P(A, \sim B) = P(A) \cdot P(\sim B)$
- $P(\sim A, \sim B) = P(\sim A) \cdot P(\sim B)$.

Conditional independence

- Two events A and C are called **Conditionally Independent** given B if $P(A | B, C) = P(A | B)$.
- Ex:
 - A: I will play football tomorrow.
 - B: the football match will happen in-house tomorrow.
 - C: it will not rain tomorrow.
 - $P(A | B, C) = P(A | B)$.

Some rules in probability theory

■ Chain rules:

- $P(A,B) = P(A | B).P(B) = P(B | A).P(A) = P(B,A)$
- $P(A | B) = P(A,B)/P(B) = P(B | A).P(A)/P(B)$
- $P(A,B | C) = P(A,B,C)/P(C) = P(A | B,C).P(B,C)/P(C)$
 $= P(A | B,C).P(B | C).$

■ Independence:

- $P(A | B) = P(A)$
if A and B are statistically independent.
- $P(A,B | C) = P(A | C).P(B | C)$
if A and B are statistically independent, conditioned on C.
- $P(A_1, \dots, A_n | C) = P(A_1 | C) \dots P(A_n | C)$
if A_1, \dots, A_n are statistically independent, conditioned on C.

Product and sum rules

- Consider x and y are discrete random variables. Their domains are X and Y respectively

- **Product rule:**

$$P(x, y) = P(x|y)P(y)$$

- **Sum rule**

$$P(x) = \sum_{y \in Y} P(x, y)$$

- The summation (tổng) should be integration (tích phân) if y is continuous
(tổng sẽ được thay bằng tích phân nếu biến y liên tục)

Bayes' rule

$$P(\theta|\mathbf{D}) = \frac{P(\mathbf{D}|\theta)P(\theta)}{P(\mathbf{D})}$$

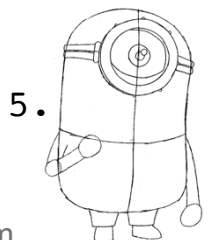
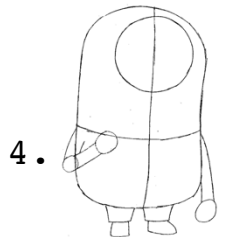
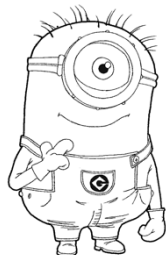
- $P(\theta)$: *prior probability* (xác suất tiên nghiệm) of the variable θ .
 - Our uncertainty about θ before observing data.
- $P(\mathbf{D})$: prior probability that we can observe data \mathbf{D} .
- $P(\mathbf{D} | \theta)$: probability (*likelihood*) that we can observe data \mathbf{D} provided that θ is known.
- $P(\theta | \mathbf{D})$: *posterior probability* (xác suất hậu nghiệm) of θ if we already have observed data \mathbf{D} .
 - Bayesian approach bases on this quantity.

Probabilistic models

Model, inference, learning

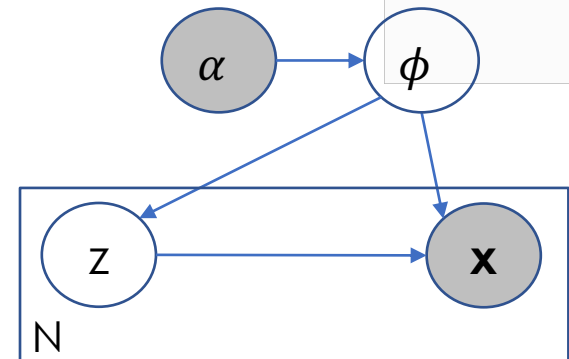
Probabilistic model

- ❑ Our assumption on how the data were generated
(giả thuyết của chúng ta về quá trình dữ liệu đã được sinh ra như thế nào)
- ❑ Example: **how a sentence is generated?**
 - ❖ We assume our brain does as follow:
 - ❖ *First choose the topic of the sentence*
 - ❖ *Generate the words one-by-one to form the sentence*
- ❑ **How will TIM be drawn?**



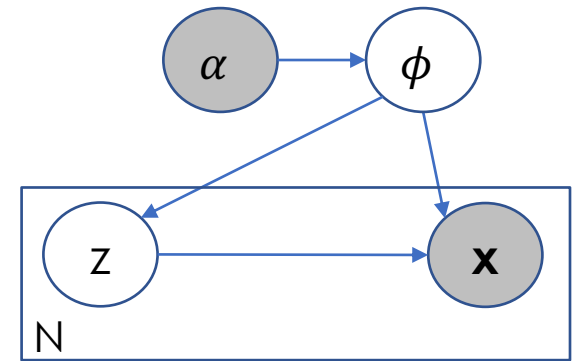
Probabilistic model

- A model sometimes consists of
 - ❖ **Observed variable** (e.g., x) which models the observation (data instance) (biến quan sát được)
 - ❖ **Hidden variable** which describes the hidden things (e.g., z, ϕ) (biến ẩn)
 - ❖ **Local variable** (e.g., z, x) which associates with one data instance
 - ❖ **Global variable** (e.g., ϕ) which is shared across the data instances, and is the representative of the model
 - ❖ **Relations** between the variables
- Each variable follows some probability distribution (mỗi biến tuân theo một phân bố xác suất nào đó)



Different types of models

- **Probabilistic graphical model (PGM):** Graph + Probability Theory (mô hình đồ thị xác suất)
 - Each vertex represents a random variable, grey circle means “observed”, white circle means “latent”
 - Each edge represents the conditional dependence between two variables
 - *Directed graphical model:* each edge has a direction
 - *Undirected graphical model:* no direction in the edges
- Latent variable model: a PGM which has at least one latent variable
- Bayesian model: a PGM which has a prior distribution on its parameter

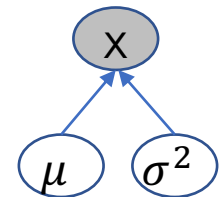
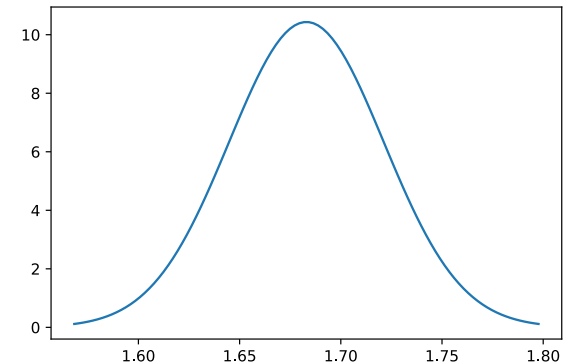


Univariate normal distribution

- We wish to model the height of a person
 - We had collected a dataset from 10 people in Hanoi:
 $\mathbf{D}=\{1.6, 1.7, 1.65, 1.63, 1.75, 1.71, 1.68, 1.72, 1.77, 1.62\}$
- Let x denote the random variable that represents the height of a person
- **Assumption:** x follows a Normal distribution (Gaussian) with the following *probability density function* (PDF)

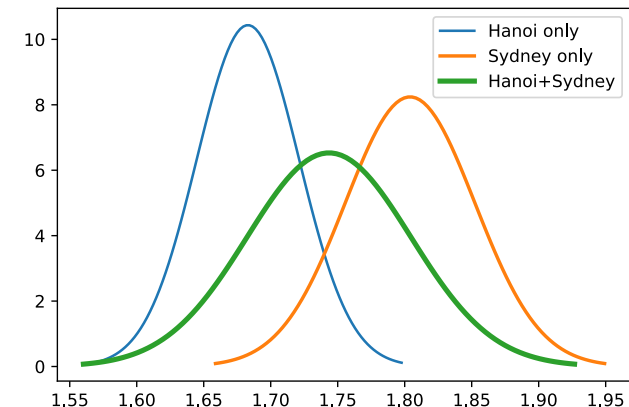
$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}$$

- where $\{\mu, \sigma^2\}$ are the mean and variance
- Note:
 - $\mathcal{N}(x|\mu, \sigma^2)$ represents the class of normal distributions
 - This class is parameterized by $\theta = (\mu, \sigma^2)$
- To make prediction/inference, we need to know specific values of $\{\mu, \sigma^2\}$



Univariate Gaussian mixture model (1)

- We wish to model the height of a person
 - We had collected a dataset from 10 people in Hanoi + 10 people in Sydney
 $\mathbf{D} = \{1.6, 1.7, 1.65, 1.63, 1.75, 1.71, 1.68, 1.72, 1.77, 1.62, 1.75, 1.80, 1.85, 1.65, 1.91, 1.78, 1.88, 1.79, 1.82, 1.81\}$
- Let x denote the random variable that represents the height
- If we use Normal distribution:
 - Blue curve models the height in Hanoi
 - Orange curve models the height in Sydney
 - Green curve models the whole \mathbf{D}
- Univariate Gaussian does not model well the underlying distribution
 - Mixture model?
(mô hình hỗn hợp)



Univariate Gaussian mixture model (2)

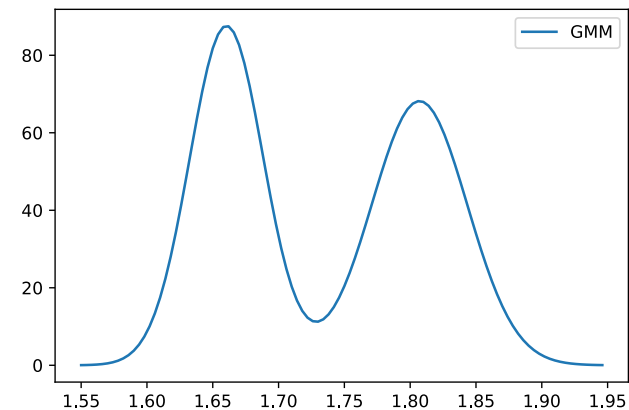
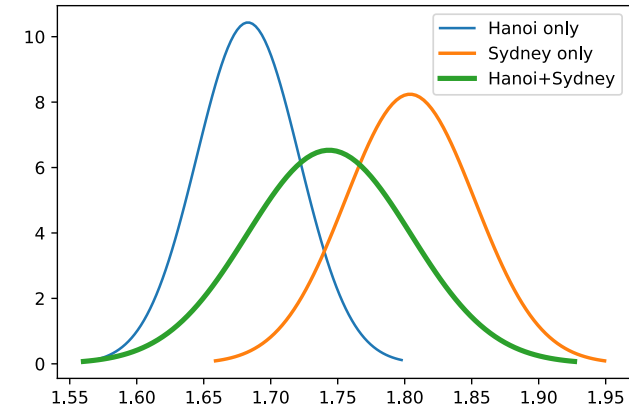
- **Assumption:** the data are generated from two different Gaussians, and each instance is generated from one of the two Gaussians.

Generative process:

- ❖ *Pick the component index: $z \sim \text{Multinomial}(z|\phi)$*
- ❖ *Generate sample $x \sim \text{Normal}(x | \mu_z, \sigma_z^2)$*
- This is **Gaussian mixture model** (GMM) (mô hình hỗn hợp Gauss)
 - (μ_1, σ_1^2) represents the first Gaussian
 - (μ_2, σ_2^2) represents the second Gaussian
 - $\phi \in [0,1]$ is the parameter of the Multinomial distribution, $P(z = 1 | \phi) = \phi = 1 - P(z = 2 | \phi)$
- Density of the GMM:

$$\phi \mathcal{N}(x | \mu_1, \sigma_1^2) + (1 - \phi) \mathcal{N}(x | \mu_2, \sigma_2^2)$$

Note: “ \sim ” means “follows” (tuân theo)



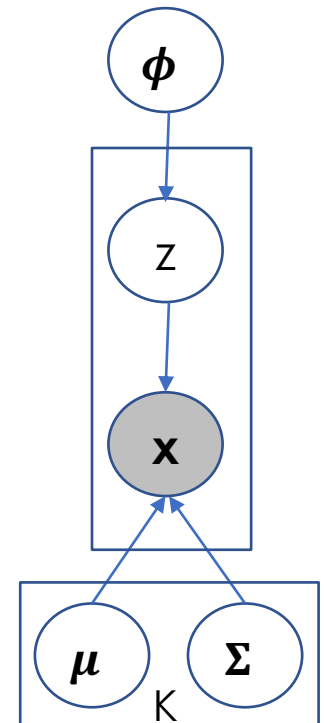
GMM: Multivariate case

- ❑ Consider the case each \mathbf{x} belongs to the n -dimensional space.
- ❑ GMM: we assume that the data are samples from K different Gaussian distributions.
- ❑ Each instance \mathbf{x} is generated from one of those K Gaussians by the following **generative process**:
 - ❖ Take the component index $z \sim \text{Multinomial}(z|\boldsymbol{\phi})$
 - ❖ Generate $\mathbf{x} \sim \text{Normal}(\mathbf{x} | \boldsymbol{\mu}_z, \boldsymbol{\Sigma}_z)$
- ❑ The density function is

$$p(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\phi}) = \sum_{k=1}^K \phi_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- ❑ $\boldsymbol{\phi} = (\phi_1, \dots, \phi_K)$ represents the weights of the Gaussians
- ❑ Each multivariate Gaussian has density

$$\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{\det(2\pi\boldsymbol{\Sigma})}} \exp \left[-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right]$$



PGM: some well-known models

- Gaussian mixture model (GMM)
 - Modeling real-valued data
- Latent Dirichlet allocation (LDA)
 - Modeling the topics hidden in textual data
- Hidden Markov model (HMM)
 - Modeling time-series, i.e., data with time stamps or sequential nature
- Conditional Random Field (CRF)
 - for structured prediction

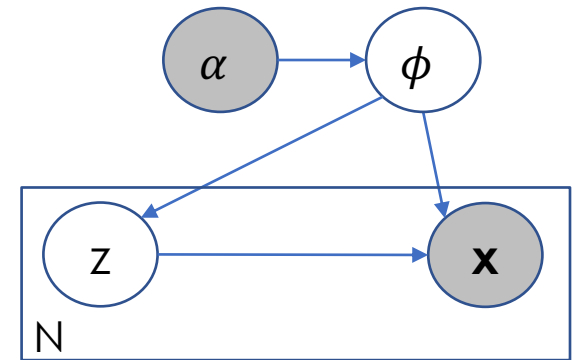
Probabilistic model: two problems

□ **Inference** for a given instance \mathbf{x}_n

- ❖ Recovery of the local variable (e.g., z_n), or
- ❖ The distribution of the local variables (e.g., $P(z_n, \mathbf{x}_n | \phi)$)
- ❖ Example: for GMM, we want to know z_n indicating which Gaussian did generate \mathbf{x}_n

□ **Learning (estimation)**

- ❖ Given a training dataset, estimate the joint distribution of the variables
 - ❖ E.g., estimate $P(\phi, z_1, \dots, z_n, \mathbf{x}_1, \dots, \mathbf{x}_n | \alpha)$
 - ❖ E.g., estimate $P(\mathbf{x}_1, \dots, \mathbf{x}_n | \alpha)$
 - ❖ E.g., estimate α
- ❖ Inference of local variables is often needed



Inference and Learning

MLE, MAP

Some inference approaches (1)

- Let D be the data, and h be a hypothesis
 - hypothesis: unknown parameter, hidden variables, ...
- **Maximum Likelihood Estimation (MLE, cực đại hoá khả năng)**

$$h^* = \arg \max_{h \in \mathbf{H}} P(D|h)$$

- Finds h^* (in the hypothesis space \mathbf{H}) that maximizes the likelihood of the data.
 - *Other words: MLE makes inference about the model that is most likely to have generated the data.*
- **Bayesian inference** (suy diễn Bayes) considers the transformation of our prior knowledge $P(h)$, through the data D , into the posterior knowledge $P(h|D)$.
 - Remember the Bayes' rule: $P(h|D) = P(D|h)P(h)/P(D)$. So

$$P(h|D) \propto P(D|h) * P(h)$$

(Posterior \propto Likelihood * Prior)

Some inference approaches (2)

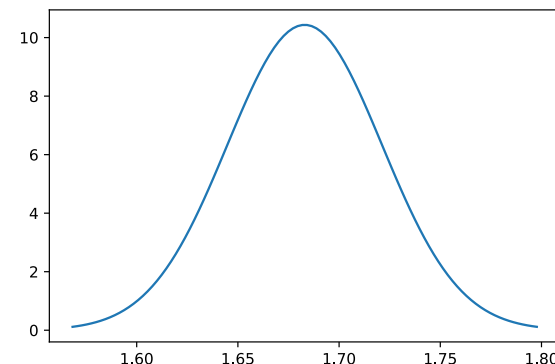
- In some cases, we may know the prior distribution of h .
- **Maximum a Posterior Estimation (MAP, cực đại hoá hậu nghiệm)**

$$\begin{aligned} h^* &= \arg \max_{h \in \mathbf{H}} P(h|\mathbf{D}) = \arg \max_{h \in \mathbf{H}} P(\mathbf{D}|h) P(h)/P(\mathbf{D}) \\ &= \arg \max_{h \in \mathbf{H}} P(\mathbf{D}|h) P(h) \end{aligned}$$

- Finds h^* that maximizes the posterior probability of h .
- MAP finds a point (posterior mode), not a distribution → point estimation
- MLE is a special case of MAP, when using uniform prior over h .
- *Full Bayesian inference* tries to estimate the full posterior distribution $P(h|\mathbf{D})$, not just a point h^* .
- Note:
 - MLE, MAP, or full Bayesian approaches can be applied to both learning and inference.

MLE: Gaussian example (1)

- We wish to model the height of a person, using the dataset $\mathbf{D} = \{1.6, 1.7, 1.65, 1.63, 1.75, 1.71, 1.68, 1.72, 1.77, 1.62\}$
 - Let x be the random variable representing the height of a person.
 - Model: assume that x follows a Gaussian distribution with **unknown** mean μ and variance σ^2
 - **Learning:** estimate (μ, σ) from the given data $\mathbf{D} = \{x_1, \dots, x_{10}\}$.
- Let $f(x|\mu, \sigma)$ be the density function of the Gaussian family, parameterized by (μ, σ) .
 - $f(x_n|\mu, \sigma)$ is the likelihood of instance x_n .
 - $f(\mathbf{D}|\mu, \sigma)$ is the likelihood function of \mathbf{D} .
- Using MLE, we will find
$$(\mu_*, \sigma_*) = \arg \max_{\mu, \sigma} f(\mathbf{D}|\mu, \sigma)$$



MLE: Gaussian example (2)

- **i.i.d assumption:** we assume that the data are independent and identically distributed (dữ liệu được sinh ra một cách độc lập)

□ As a result, we have $P(\mathbf{D}|\mu, \sigma) = P(x_1, \dots, x_{10}|\mu, \sigma) = \prod_{i=1}^{10} P(x_i|\mu, \sigma)$

- Using this assumption, MLE will be

$$(\mu_*, \sigma_*) = \arg \max_{\mu, \sigma} \prod_{i=1}^{10} f(x_i|\mu, \sigma) = \arg \max_{\mu, \sigma} \prod_{i=1}^{10} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x_i-\mu)^2}$$

$$= \arg \max_{\mu, \sigma} \log \prod_{i=1}^{10} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x_i-\mu)^2}$$

Log trick,
 $\log \stackrel{\text{def}}{=} \ln$

$$= \arg \max_{\mu, \sigma} \sum_{i=1}^{10} \left(-\frac{1}{2\sigma^2} (x_i - \mu)^2 - \log \sqrt{2\pi\sigma^2} \right)$$

- Using gradients (w.r.t μ, σ), we can find

$$\mu_* = \frac{1}{10} \sum_{i=1}^{10} x_i = 1.683, \quad \sigma_*^2 = \frac{1}{10} \sum_{i=1}^{10} (x_i - \mu_*)^2 \approx 0.0015$$

MAP: Gaussian Naïve Bayes (1)

- Consider the classification problem
 - Training data $\mathbf{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_M, y_M)\}$ with M instances, C classes.
 - Each \mathbf{x}_i is a vector in the n -dimensional space \mathbb{R}^n , e.g., $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{in})^T$.
- *Model assumption:* we assume there are C different distributions generating the data in \mathbf{D} , and the data with label c are generated from a Gaussian distribution parameterized by $(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$
 - $\boldsymbol{\mu}_c$ is the mean vector, $\boldsymbol{\Sigma}_c$ is the covariance matrix of size $n \times n$.
- *Learning:* we consider $P(\boldsymbol{\mu}, \boldsymbol{\Sigma}, c | \mathbf{D})$, where $(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = (\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\mu}_C, \boldsymbol{\Sigma}_C)$

$$(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*) \stackrel{\text{def}}{=} \arg \max_{\boldsymbol{\mu}, \boldsymbol{\Sigma}, c} P(\boldsymbol{\mu}, \boldsymbol{\Sigma}, c | \mathbf{D}) = \arg \max_{\boldsymbol{\mu}, \boldsymbol{\Sigma}, c} P(\mathbf{D} | \boldsymbol{\mu}, \boldsymbol{\Sigma}, c) P(c)$$

Bayes' rule,
removing $P(\mathbf{D})$,
assuming uniform
prior over $\boldsymbol{\mu}, \boldsymbol{\Sigma}$

- We estimate $P(c)$ to be the proportion of class c in \mathbf{D} :
 $P(c) = |\mathbf{D}_c| / |\mathbf{D}|$ where \mathbf{D}_c contains all instances with label c in \mathbf{D} .

- Since the C classes are independent, we can do learning for each class

$$(\boldsymbol{\mu}_{c*}, \boldsymbol{\Sigma}_{c*}) \stackrel{\text{def}}{=} \arg \max_{\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c} P(\mathbf{D}_c | \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c) P(c) = \arg \max_{\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c} P(\mathbf{D}_c | \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$$

MAP: Gaussian Naïve Bayes (2)

- Assuming the samples are i.i.d, we have

$$\begin{aligned}
 (\boldsymbol{\mu}_{c*}, \boldsymbol{\Sigma}_{c*}) &= \arg \max_{\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c} \prod_{\mathbf{x} \in D_c} P(\mathbf{x} | \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c) = \arg \max_{\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c} \sum_{\mathbf{x} \in D_c} \log P(\mathbf{x} | \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c) \\
 &= \arg \max_{\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c} \sum_{\mathbf{x} \in D_c} \log \left[\frac{1}{\sqrt{\det(2\pi \boldsymbol{\Sigma}_c)}} \exp \left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_c)^T \boldsymbol{\Sigma}_c^{-1} (\mathbf{x} - \boldsymbol{\mu}_c) \right) \right] \\
 &= \arg \max_{\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c} \sum_{\mathbf{x} \in D_c} -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_c)^T \boldsymbol{\Sigma}_c^{-1} (\mathbf{x} - \boldsymbol{\mu}_c) - \log \sqrt{\det(2\pi \boldsymbol{\Sigma}_c)}
 \end{aligned}$$

- Using gradients (w.r.t $\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c$), we can arrive at


$$\boldsymbol{\mu}_{c*} = \frac{1}{|D_c|} \sum_{\mathbf{x} \in D_c} \mathbf{x}, \quad \boldsymbol{\Sigma}_{c*} = \frac{1}{|D_c|} \sum_{\mathbf{x} \in D_c} (\mathbf{x} - \boldsymbol{\mu}_{c*})(\mathbf{x} - \boldsymbol{\mu}_{c*})^T$$

- So, after training we obtain the $(\boldsymbol{\mu}_{c*}, \boldsymbol{\Sigma}_{c*}, P(c))$ for each class c .

MAP: Gaussian Naïve Bayes (3)

- Trained model: $(\boldsymbol{\mu}_{c*}, \boldsymbol{\Sigma}_{c*}, P(c))$ for each class c
- Prediction for a new instance \mathbf{z} by finding the class label that gives the highest posterior probability:

Bayes' rule


$$\begin{aligned} c_z &= \arg \max_{c \in \{1, \dots, C\}} P(c | \mathbf{z}, \boldsymbol{\mu}_{c*}, \boldsymbol{\Sigma}_{c*}) = \arg \max_{c \in \{1, \dots, C\}} P(\mathbf{z} | \boldsymbol{\mu}_{c*}, \boldsymbol{\Sigma}_{c*}, c) P(c) \\ &= \arg \max_{c \in \{1, \dots, C\}} \log P(\mathbf{z} | \boldsymbol{\mu}_{c*}, \boldsymbol{\Sigma}_{c*}, c) + \log P(c) \\ &= \arg \max_{c \in \{1, \dots, C\}} -\frac{1}{2} (\mathbf{z} - \boldsymbol{\mu}_{c*})^T \boldsymbol{\Sigma}_{c*}^{-1} (\mathbf{z} - \boldsymbol{\mu}_{c*}) - \log \sqrt{\det(2\pi \boldsymbol{\Sigma}_{c*})} + \log P(c) \end{aligned}$$

- If using MLE, we do not need to use/estimate the prior $P(c)$.

MAP: Multinomial Naïve Bayes (1)

- Consider the text classification problem (dữ liệu có thuộc tính rời rạc)
 - Training data $\mathbf{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_M, y_M)\}$ with M documents, C classes.
 - TF: each document \mathbf{x}_i is represented by a vector of V dimensions, e.g., $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{iV})^T$, each x_{ij} is the *frequency* of term j in document \mathbf{x}_i
- *Model assumption*: we assume there are C different distributions generating the data in \mathbf{D} , and the data with label c are generated from a multinomial distribution which is parameterized by θ_c and has probability mass function

$$f(x_1, \dots, x_V | \theta_{c1}, \dots, \theta_{cV}) = \frac{\Gamma(\sum_{j=1}^V x_j + 1)}{\prod_{j=1}^V \Gamma(x_j + 1)} \prod_{k=1}^V \theta_{ck}^{x_k}$$

- $\theta_{cj} = P(x = j | \theta_{cj})$ is the probability that term index $j \in \{1, \dots, V\}$ appears, satisfying $\sum_{k=1}^V \theta_{ck} = 1$. Γ is the gamma function.
- *Learning*: we can do similarly with Gaussian Naïve Bayes to estimate $\theta_c = (\theta_{c1}, \dots, \theta_{cV})$ and $P(c)$ for each class c .

MAP: Multinomial Naïve Bayes (2)

- Trained model: $(\boldsymbol{\theta}_{c*}, P(c))$ for each class c

- Prediction for a new instance $\mathbf{z} = (z_1, \dots, z_V)^T$ by

$$\begin{aligned} c_z &= \arg \max_{c \in \{1, \dots, C\}} P(c | \mathbf{z}, \boldsymbol{\theta}_{c*}) = \arg \max_{c \in \{1, \dots, C\}} P(\mathbf{z} | \boldsymbol{\theta}_{c*}, c) P(c) \\ &= \arg \max_{c \in \{1, \dots, C\}} \log P(\mathbf{z} | \boldsymbol{\theta}_{c*}) + \log P(c) \end{aligned} \quad (\text{MNB.1})$$

$$= \arg \max_{c \in \{1, \dots, C\}} \log \frac{\Gamma(\sum_{j=1}^V z_j + 1)}{\prod_{j=1}^V \Gamma(z_j + 1)} \prod_{k=1}^V \theta_{ck*}^{z_k} + \log P(c)$$

$$= \arg \max_{c \in \{1, \dots, C\}} \log \prod_{k=1}^V \theta_{ck*}^{z_k} + \log P(c)$$

$$= \arg \max_{c \in \{1, \dots, C\}} \log \prod_{k=1}^V P(z_k | \theta_{ck*}) + \log P(c) \quad (\text{MNB.2})$$

- The label that gives the highest posterior probability
- Note: we essentially assume that *the attributes are conditionally independent*, as shown in equations (MNB.1) and (MNB.2).
(ta ngầm giả thuyết rằng các thuộc tính độc lập với nhau)

GMM revisit

- Consider learning GMM, with K Gaussian distributions, from the training data $\mathbf{D} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M\}$.

- The density function is $p(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\phi}) = \sum_{k=1}^K \phi_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$

- $\boldsymbol{\phi} = (\phi_1, \dots, \phi_K)$ represents the weights of the Gaussians

- Each multivariate Gaussian has density

$$\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{\sqrt{\det(2\pi\boldsymbol{\Sigma}_k)}} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1}(\mathbf{x} - \boldsymbol{\mu}_k)\right]$$

- MLE tries to maximize the following log-likelihood function

$$L(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\phi}) = \sum_{i=1}^M \log \sum_{k=1}^K \phi_k \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- We cannot find a closed-form solution!

- Approximation and iterative algorithms are needed.

Difficult situations

- No closed-form solution for the learning/inference problem?
(không tìm được ngay công thức nghiệm)
 - The examples before are easy cases, as we can find solutions in a closed form by using gradient.
 - Many models (e.g., GMM) do not admit a closed-form solution.
- No explicit expression of the density/mass function?
(không có công thức tường minh để tính toán)
- Intractable inference (bài toán suy diễn không khả thi)
 - Inference in many probabilistic models is NP-hard.
[Sontag & Roy, 2011; Tosh & Dasgupta, 2019]

Reference

- Blei, David M., Alp Kucukelbir, and Jon D. McAuliffe. "Variational inference: A review for statisticians." *Journal of the American Statistical Association* 112, no. 518 (2017): 859-877.
- Blundell, Charles, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. "Weight Uncertainty in Neural Network." In *International Conference on Machine Learning (ICML)*, pp. 1613-1622. 2015.
- Gal, Yarin, and Zoubin Ghahramani. "Dropout as a bayesian approximation: Representing model uncertainty in deep learning." In *International Conference on Machine Learning*, pp. 1050-1059. 2016.
- Ghahramani, Zoubin. "Probabilistic machine learning and artificial intelligence." *Nature* 521, no. 7553 (2015): 452-459.
- Kingma, Diederik P., and Max Welling. "Auto-encoding variational bayes." In *International Conference on Learning Representations (ICLR)*, 2014.
- Jordan, Michael I., and Tom M. Mitchell. "Machine learning: Trends, perspectives, and prospects." *Science* 349, no. 6245 (2015): 255-260.
- Tosh, Christopher, and Sanjoy Dasgupta. "The Relative Complexity of Maximum Likelihood Estimation, MAP Estimation, and Sampling." In *Proceedings of the 32nd Conference on Learning Theory*, in PMLR 99:2993-3035, 2019.
- Sontag, David, and Daniel Roy, "Complexity of inference in latent dirichlet allocation" in: *Proceedings of Advances in Neural Information Processing System*, 2011.