KAIST EE

Lecture 8: When Models Meet Data

Yi, Yung (이용)

Mathematics for Machine Learning
KAIST EE

MONTH DAY, 2021

Roadmap

KAIST EE

- Data, Models, and Learning
- Models as Functions: Empirical Risk Minimization
- Models as Probabilistic Models: Parameter Estimation (ML and MAP)
- Probabilistic Modeling and Inference
- Directed Graphical Models
- Model Selection

Roadmap

KAIST EE

- Data, Models, and Learning
- Models as Functions: Empirical Risk Minimization
- Models as Probabilistic Models: Parameter Estimation (ML and MAP)
- Probabilistic Modeling and Inference
- Directed Graphical Models
- Model Selection

Data, Models, and Learning

KAIST EE

- Three major components of a machine learning system
  1. Data: $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n), \ldots, (\boldsymbol{x}_N, y_N)\}$
  2. Models: functions or probabilistic models
  3. Learning: Training, and prediction/inference
- Good machine learning models: Perform well for unseen (untrained) data
- Machine learning algorithm: training and prediction

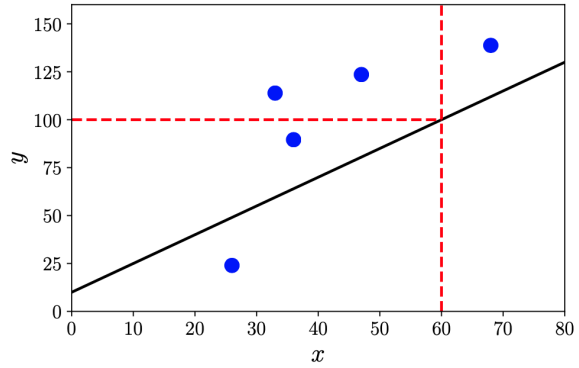
Data as Vectors

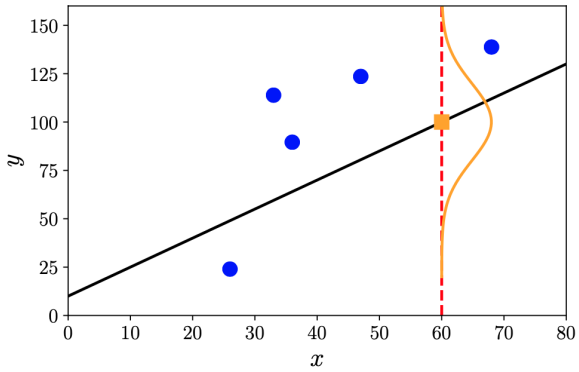
KAIST EE

- Tabular format or not, numerical or not, good feature extraction etc.
- Assume that data is given as D-dimensional vector $\boldsymbol{x}_n$ of real numbers, each called features, attributes, or covariates.
- Dataset: consisting of data points or examples $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N$
- In supervised learning, $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n), \ldots, (\boldsymbol{x}_N, y_N)\}$, where y_n is the label (or target, response variable, or annotation).
- Better representation of data as vectors
  - finding lower-dimensional approximations of the original feature vector (PCA via SVD or EVD)
  - using nonlinear higher-dimensional combinations of the original feature vector (feature map and kernel)

Models: Functions vs. Probabilistic Models

- Now, the business of constructing a predictor.
- Models as functions
  - $f : \mathbb{R}^D \mapsto \mathbb{R}$.
  - Example. $f(\boldsymbol{x}) = \boldsymbol{\theta}^\mathsf{T}\boldsymbol{x} + \theta_0$, Unknown parameter: $\boldsymbol{\theta}, \theta_0$
- Models as probabilistic models
  - model our uncertainty due to the observation process and our uncertainty in the parameters of our model
  - predictors should be able to express some sort of uncertainty via probabilistic models
  - Parameters: parameters of a chosen probabilistic model (mean and variance of Gaussian)





Learning Algorithms

KAIST EE

- Three algorithmic phases

(1) Prediction or inference: via function or probabilitic models

(2) Training or parameters estimation
  - fixed parameter assumption (non-probabilistic) or Bayesisan approach (probabilistic)
  - Non-probabilistic: e.g., empirical risk minimization
  - Probabilistic: e.g., ML (Maximum Likelihood), MAP (Maximum A Posteriori)
  - Cross-validation: Simulation of performing for unseen data
  - Regularization/Prior: Balancing models between training and unseen data

(3) Hyperparameter tuning or model selection

Roadmap

KAIST EE

- Data, Models, and Learning
- Models as Functions: Empirical Risk Minimization
- Models as Probabilistic Models: Parameter Estimation (ML and MAP)
- Probabilistic Modeling and Inference
- Directed Graphical Models
- Model Selection

Empirical Risk Minimization

KAIST EE

- Predictor as a function
- Given $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n), \ldots, (\boldsymbol{x}_N, y_N)\}$, estimate a predictor $f(\cdot, \boldsymbol{\theta}) : \mathbb{R}^D \mapsto \mathbb{R}$
- Find a good parameter $\boldsymbol{\theta}^*$, such that $f(\boldsymbol{x}_n, \boldsymbol{\theta}^*) = \hat{y}_n \approx y_n$, for all $n = 1, \ldots, N$
- Example. Affine function: By adding the unit feature $x^{(0)} = 1$ and θ_0, i.e.,
  $\boldsymbol{x}_n = [1, x_n^{(1)}, \ldots, x_n^{(D)}]^\mathsf{T}$, $\boldsymbol{\theta} = [\theta_0, \theta_1, \ldots, \theta_D]^\mathsf{T}$

  $$f(\boldsymbol{x}_n, \boldsymbol{\theta}) = \boldsymbol{\theta}^\mathsf{T} \boldsymbol{x}_n = \theta_0 + \sum_{d=1}^{D} \theta_d x_n^{(d)}$$
- Example. Neural network: Complex non-linear function

Loss Function

- Training set: $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n), \ldots, (\boldsymbol{x}_N, y_N)\}$, an example matrix $\boldsymbol{X} := [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]^\mathsf{T} \in \mathbb{R}^{N \times D}$, a label vector $\boldsymbol{y} := [y_1, \ldots, y_N]^\mathsf{T}$,
- Average loss, empirical risk

$$\boldsymbol{R}_{\text{emp}}(f, \boldsymbol{X}, \boldsymbol{y}) = \frac{1}{N} \sum_{n=1}^{N} \ell(y_n, \hat{y}_n)$$

- Goal: Minimizing empirical risk
- Example. The squared loss function $\ell(y_n, \hat{y}_n) = (y_n - \hat{y}_n)^2$ leads to:

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^D} \frac{1}{N} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|^2$$

- Question. Ultimgate goal: Minimizing expected risk (for unseen data) $\boldsymbol{R}_{\text{true}} = \mathbb{E}_{\boldsymbol{x},y}[\ell(y, f(\boldsymbol{x}))]$?

Overfitting and Regularization

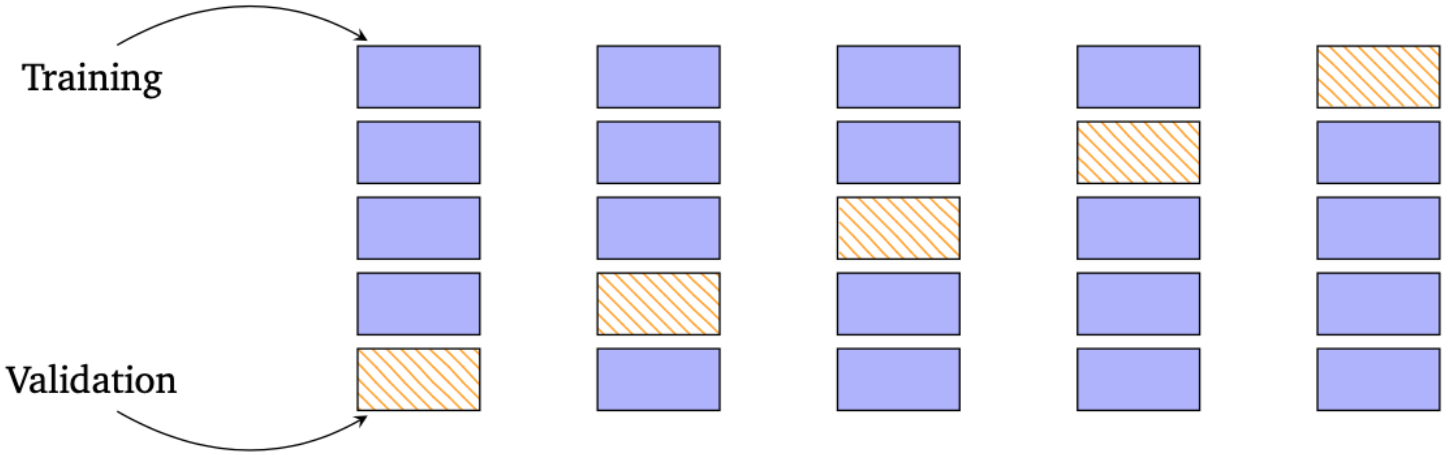
KAIST EE

- The predictor fits too closely to the training data and does not generalize well to new data
- Need to somehow bias the search for the minimizer of empirical risk by introducing a penalty term
- Regularization: Compromise between accurate solution of empirical risk minimization and the size or complexity of the solution.
- Example. Regularized Least Squares

$$\min_{\boldsymbol{\theta}\in\mathbb{R}^D} \frac{1}{N} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|^2 + \lambda \|\boldsymbol{\theta}\|^2$$

  - $\|\boldsymbol{\theta}\|^2$: regularizer, λ: regularization parameter

Cross-Validation for Generalization Performance



Roadmap

KAIST EE

- Data, Models, and Learning
- Models as Functions: Empirical Risk Minimization
- Models as Probabilistic Models: Parameter Estimation (ML and MAP)
- Probabilistic Modeling and Inference
- Directed Graphical Models
- Model Selection

MLE (Maximum Likelihood Estimation): Concepts

KAIST EE

- Idea: define a function of the parameters called likelihood function.
- Negative log-likelihood for data $\boldsymbol{x}$ and a family of probability densities $\mathbb{P}(\boldsymbol{x} \mid \boldsymbol{\theta})$ parameterized by $\boldsymbol{\theta}$:

$$\mathcal{L}_{\boldsymbol{x}}(\boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\theta}) := -\log \mathbb{P}(\boldsymbol{x} \mid \boldsymbol{\theta})$$

  - $\mathcal{L}(\boldsymbol{\theta})$: how likely a particular setting of $\boldsymbol{\theta}$ is for the observations $\boldsymbol{x}$.

- MLE: Find $\boldsymbol{\theta}$ such that $\mathcal{L}(\boldsymbol{\theta})$ is minimized (i.e., likelihood is maximized)

MLE: Supervised Learning

KAIST EE

- The set of iid examples $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_N, y_N)$
- $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ and $\mathcal{Y} = \{y_1, \ldots, y_N\}$
- Negative log-likelihood

$$\mathcal{L}(\boldsymbol{\theta}) = -\log \mathbb{P}(\mathcal{Y} \mid \mathcal{X}, \boldsymbol{\theta}) = \sum_{n=1}^{N} \log \mathbb{P}(y_n \mid \boldsymbol{x}_n, \boldsymbol{\theta})$$

- Example. Assume independent Gaussian noise $\mathcal{N}(0, \sigma^2)$ and linear model $y_n = \boldsymbol{x}_n^{\mathsf{T}}\boldsymbol{\theta}$ for prediction. Then, $Y_n|(\boldsymbol{x}_n, \boldsymbol{\theta}) \sim \mathcal{N}(\boldsymbol{x}_n^{\mathsf{T}}\boldsymbol{\theta}, \sigma^2)$.

$$\mathcal{L}(\boldsymbol{\theta}) = -\sum_{n=1}^{N} \log \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_n - \boldsymbol{x}_n^{\mathsf{T}}\boldsymbol{\theta})^2}{2\sigma^2}\right) = \frac{1}{2\sigma^2}\sum_{n=1}^{N}(y_n - \boldsymbol{x}_n^{\mathsf{T}}\boldsymbol{\theta})^2 - \sum_{n=1}^{N} \log \frac{1}{\sqrt{2\pi\sigma^2}}$$

MAP (Maximum A Posteriori)

KAIST EE

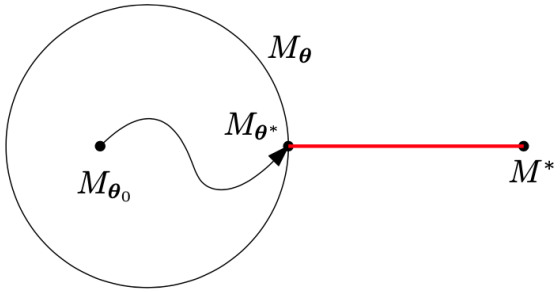
- What if we have some prior knowledge about $\boldsymbol{\theta}$? Then, how should we change our knowledge about $\boldsymbol{\theta}$ after observing data $\boldsymbol{x}$?
- Compute a posteriori distribution (using Bayes' Theorem) and find $\boldsymbol{\theta}$ that maximizes the distribution:

$$\max_{\boldsymbol{\theta}} \mathbb{P}(\boldsymbol{\theta} \mid \boldsymbol{x}) = \max_{\boldsymbol{\theta}} \frac{\mathbb{P}(\boldsymbol{x} \mid \boldsymbol{\theta})\mathbb{P}(\boldsymbol{\theta})}{\mathbb{P}(\boldsymbol{x})} \Longleftrightarrow \min_{\boldsymbol{\theta}} \Big(-\log \mathbb{P}(\boldsymbol{\theta} \mid \boldsymbol{x}) \Big)$$

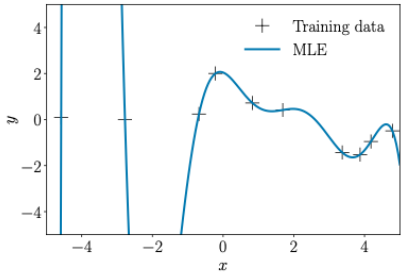
  - In finding the optimal $\boldsymbol{\theta}$, $\mathbb{P}(\boldsymbol{x})$ can be ignored
- ML and MAP: Bridging the non-probabilistic and probabilistic worlds as it explicitly acknowledges the need for a prior distribution, yet producing a point estimate (one single parameter return).
- We later see the full parameter distributions

Model Fitting

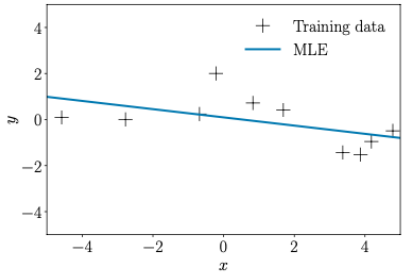
- Model class $M_{\boldsymbol{\theta}}$ vs. Right model M^*



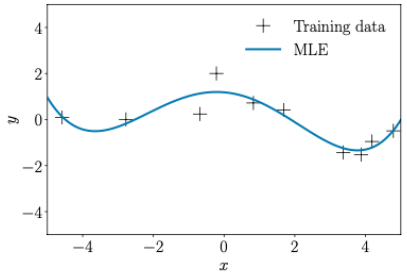
- Overfitting vs. Underfitting vs. Good fitting



(a) Overfitting



(b) Underfitting.



(c) Fitting well.

Roadmap

KAIST EE

- Data, Models, and Learning
- Models as Functions: Empirical Risk Minimization
- Models as Probabilistic Models: Parameter Estimation (ML and MAP)
- Probabilistic Modeling and Inference
- Directed Graphical Models
- Model Selection

Modeling Generative Process and Probabilistic Models

KAIST EE

- Many machine learning tasks: prediction of future events and decision making
- Often build (probabilistic) models that describe the generative process that generates the observed data
- In probabilistic modeling, the joint distribution $\mathbb{P}(\boldsymbol{x}, \boldsymbol{\theta})$ of the observed variables $\boldsymbol{x}$ and the hidden parameters $\boldsymbol{\theta}$ encapsulate the key infomration
  - the prior and likelihood, marginal distribution, and the posterior
- Essentially, if we know the joint distribution, we know all about its probabilistic model

Bayesian Inference

KAIST EE

- Earlier, two ways of estimating the parameter $\boldsymbol{\theta}$: ML and MAP. Essentially, it is solving an optimization problem to get a single best value $\boldsymbol{\theta}^*$. $\Longrightarrow$ Prediction through $\mathbb{P}(\boldsymbol{x} \mid \boldsymbol{\theta}^*)$.
- Rather than just a likelihood, having the full posterior distribution can be useful. For a data set $\mathcal{X}$, a parameter prior $\mathbb{P}(\boldsymbol{\theta})$, and a likelihood function, the posterior is:

$$\mathbb{P}(\boldsymbol{\theta} \mid \mathcal{X}) = \frac{\mathbb{P}(\mathcal{X} \mid \boldsymbol{\theta})\mathbb{P}(\boldsymbol{\theta})}{\mathbb{P}(\mathcal{X})}, \quad \mathbb{P}(\mathcal{X}) = \int \mathbb{P}(\mathcal{X} \mid \boldsymbol{\theta})\mathbb{P}(\boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\theta}$$

- Question. Examples of prediction using the posterior distribution?
- Bayesian inference requires to solving integration, which is often challenging. In particuluar, a conjugate prior is not chosen, the integration is not analytically tractable.
- Approximation technqiues: MCMC (Markov Chain Monte Carlo), Laplance approximation, variational inference, expectation propagation

Bayesian Prediction

Given the data set $\mathcal{X}$, we want to predict A, we want to do the following prediction:

$$\mathbb{P}(A \mid \mathcal{X})$$

- ML: Easy (high), Exact (low)

$$\mathbb{P}(A \mid \mathcal{X}) \approx \mathbb{P}(A \mid \boldsymbol{\theta}), \quad \boldsymbol{\theta} = \arg\max \mathbb{P}(\mathcal{X} \mid \boldsymbol{\theta})$$

- MAP: Easy (mid), Exact (mid)

$$\mathbb{P}(A \mid \mathcal{X}) \approx \mathbb{P}(A \mid \boldsymbol{\theta}), \quad \boldsymbol{\theta} = \arg\max \mathbb{P}(\boldsymbol{\theta} \mid \mathcal{X})$$

- Fully Bayesian: Easy (low), Exact (high)

  - predictive inference, use of posterior predictive distribution, bayesian prediction

$$\mathbb{P}(A \mid \mathcal{X}) = \int \mathbb{P}(A \mid \boldsymbol{\theta}) \mathbb{P}(\boldsymbol{\theta} \mid \mathcal{X}) \mathrm{d}\boldsymbol{\theta}$$

Latent-Variable Models (1)

KAIST EE

- Including latent variables in the model: Contributing to the interpretability of the model
- Examples: PCA for dimensionality reduction, Gaussian mixture models for density estimation
- Offers data generation process through parameters: $\mathbb{P}(\boldsymbol{x} \mid \boldsymbol{\theta}, \boldsymbol{z})$, $\mathbb{P}(\boldsymbol{z})$
- Marginalization over the latent variables, which allows parameter estimation by ML and MAP (using the prior $\mathbb{P}(\boldsymbol{\theta})$)

$$\mathbb{P}(\boldsymbol{x} \mid \boldsymbol{\theta}) = \int \mathbb{P}(\boldsymbol{x} \mid \boldsymbol{\theta}, \boldsymbol{z}) \mathbb{P}(\boldsymbol{z}) \, \mathrm{d}\boldsymbol{z}$$

- We can compute a posterior on the latent variables, but marginalizing over both $\boldsymbol{z}$ and $\boldsymbol{\theta}$ is hard:

$$\mathbb{P}(\boldsymbol{z} \mid \mathcal{X}) = \frac{\mathbb{P}(\mathcal{X} \mid \boldsymbol{z}) \mathbb{P}(\boldsymbol{z})}{\mathbb{P}(X)}, \quad \mathbb{P}(\mathcal{X} \mid \boldsymbol{z}) = \int \mathbb{P}(\mathcal{X} \mid \boldsymbol{z}, \boldsymbol{\theta}) \mathbb{P}(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta}$$

Latent-Variable Models (2)

KAIST EE

- Instead, it is easier to compute the latent-variable posterior, but conditioned on the model parameters, i.e.,

$$\mathbb{P}(\boldsymbol{z} \mid \mathcal{X}, \boldsymbol{\theta}) = \frac{\mathbb{P}(\mathcal{X} \mid \boldsymbol{z}, \boldsymbol{\theta})\mathbb{P}(\boldsymbol{z})}{\mathbb{P}(\mathcal{X} \mid \boldsymbol{\theta})}$$

- Question. How do we use the posteriors $\mathbb{P}(\boldsymbol{z} \mid \mathcal{X})$ or $\mathbb{P}(\boldsymbol{z} \mid \mathcal{X}, \boldsymbol{\theta})$ in practice? Any examples?

Roadmap

KAIST EE

- Data, Models, and Learning
- Models as Functions: Empirical Risk Minimization
- Models as Probabilistic Models: Parameter Estimation (ML and MAP)
- Probabilistic Modeling and Inference
- Directed Graphical Models
- Model Selection

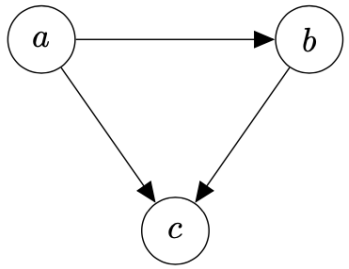
Graphical Models

KAIST EE

- Joint distribution of a probabilistic model: key quantity of interest, but quite complicated without structural properties
- However, there exist relations of independence, conditional independence among random variables.
- (Probabilistic) graphical models: Roughly speaking, a graph of random variables.
  - Simple ways to visualize the structure of the model
  - Insights into the structural properties, e.g., conditional independence
  - Computations for inference and learning can be expressed in terms of graphical manipulations

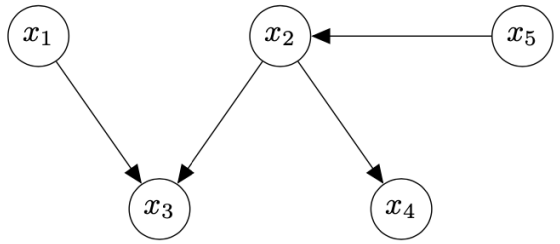
Graph Semantics

$$\mathbb{P}(a, b, c) = \mathbb{P}(c|a, b)\mathbb{P}(b|a)\mathbb{P}(a)$$



(a) Fully connected.

$$\mathbb{P}(x_1, x_2, x_3, x_4, x_5) =$$
$$\mathbb{P}(x_1)\mathbb{P}(x_5)\mathbb{P}(x_2|x_5)\mathbb{P}(x_3|x_1, x_2)\mathbb{P}(x_4|x_2)$$

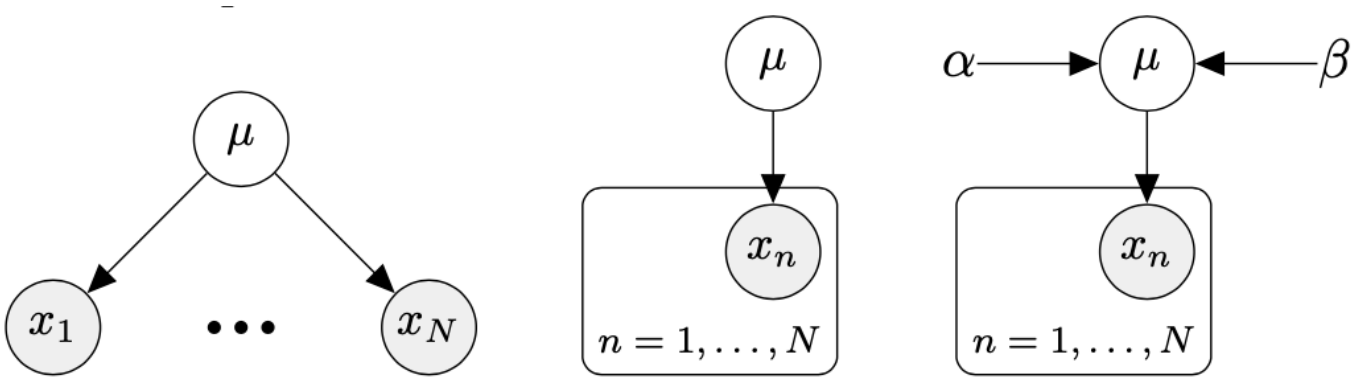


(b) Not fully connected.

- Nodes: random variables
- Directed edge for direct dependence: b directly depends on a: $a \to b$
- Graph layout: factorization of the joint distribution

$$\mathbb{P}(x_1, \ldots, x_K) = \prod_{k=1}^{K} \mathbb{P}(x_k \mid \mathrm{Pa}_k), \quad \mathrm{Pa}_k \text{ are the parent nodes of } x_k.$$

Example: N coin-flip experiments



(a) Version with x_n explicit. (b) Version with plate notation. (c) Hyperparameters α and β on the latent μ.

- Shaded nodes: observables, μ: probability of head, a (latent) random variable
- Joint distribution

$$\mathbb{P}(x_1, \ldots, x_N \mid \mu) = \prod_{n=1}^{N} \mathbb{P}(x_n \mid \mu)$$

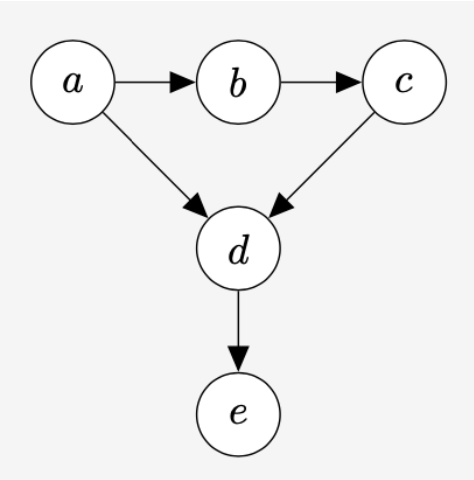
Conditional Independence and d-Separation

KAIST EE

- Question. How can we see conditional independence in the directed graphical models? For example, $\mathcal{A} \perp\!\!\!\perp \mathcal{B} \mid \mathcal{C}$?
- d-separation
  - All possible trails[1] from any node $\mathcal{A}$ to any node in $\mathcal{B}$
  - Any such path is blocked if it includes any node such that either of the following is true:
    - The arrows on the path meet either head to tail or tail to tail at the node, and the node is in $\mathcal{C}$
    - The arrows meet head to head at the node, and neither the node nor any of its descendants is in $\mathcal{C}$
  - If all the paths are blocked, then $\mathcal{A}$ is d-separated from $\mathcal{B}$ by $\mathcal{C}$.
  - If d-separated, $\mathcal{A} \perp\!\!\!\perp \mathcal{B} \mid \mathcal{C}$

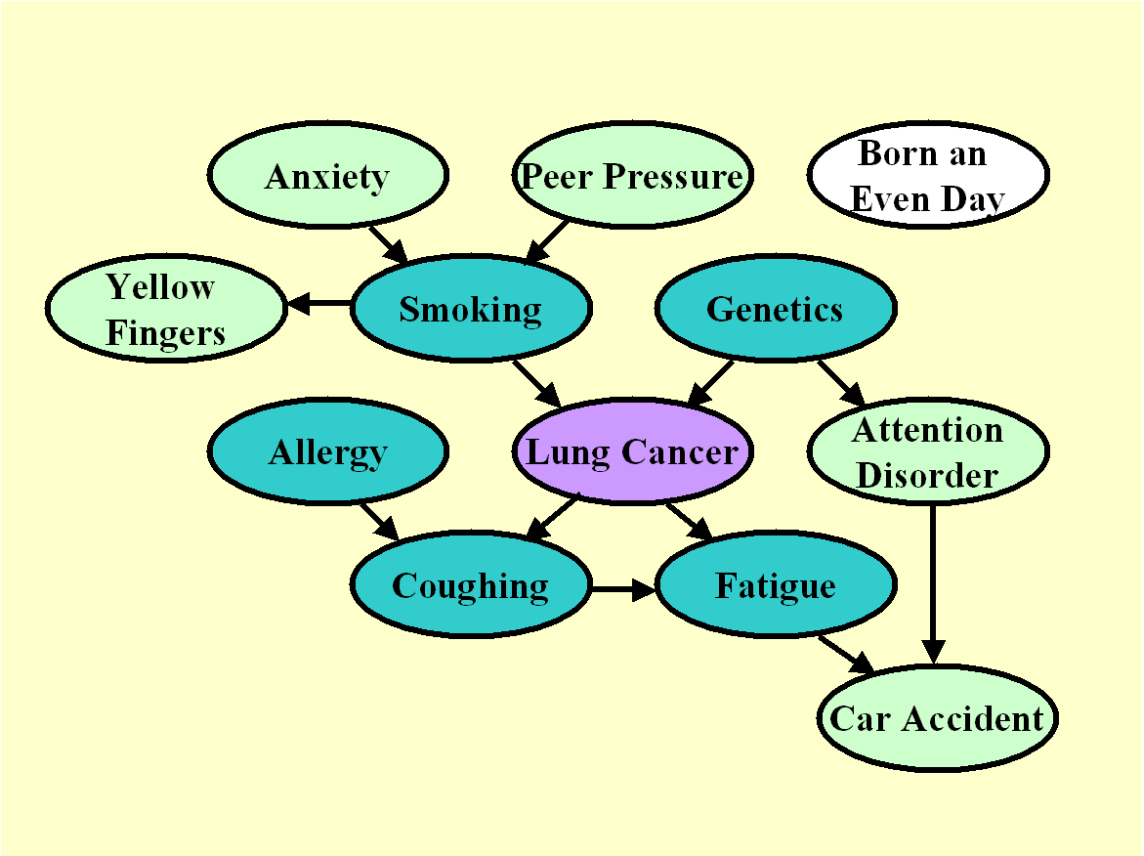
[1] paths that ignore the direction of the arrows

Example



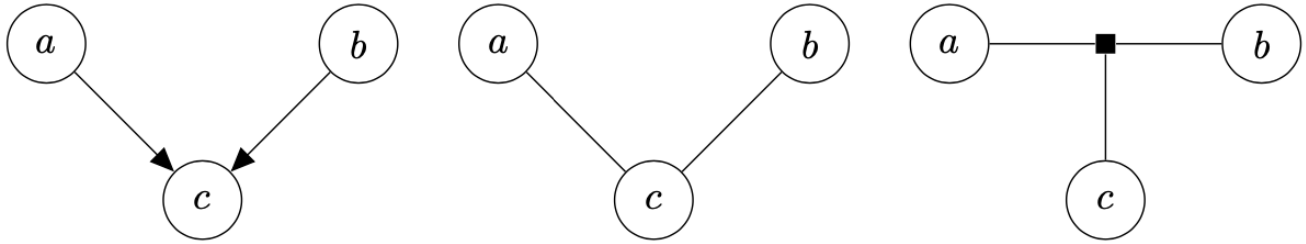
- $b \perp\!\!\!\perp d \mid a, c$
- $a \perp\!\!\!\perp c \mid b$
- $b \not\perp\!\!\!\perp d \mid c$
- $a \not\perp\!\!\!\perp c \mid b, e$

Example in Healthcare



Source: `http://www.causality.inf.ethz.ch/data/LUCAS.html`

Three Types of Graphical Models



(a) Directed graphical model (b) Undirected graphical model (c) Factor graph

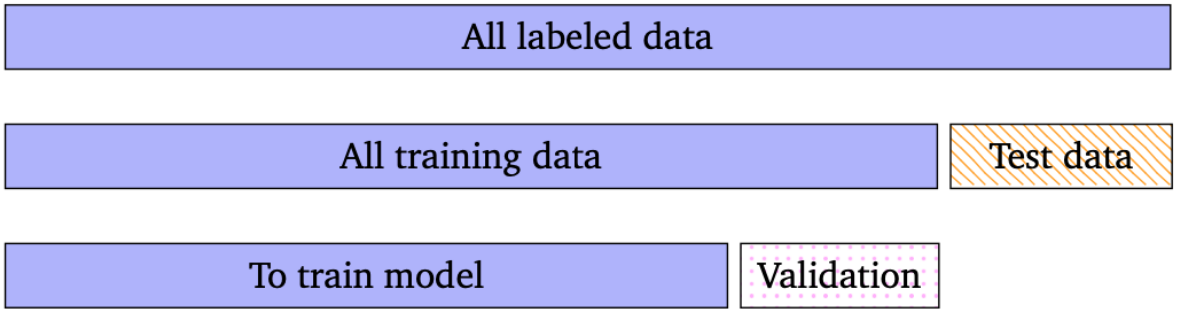
- Directed graphical models (or Bayesian Networks)
- Undirected graphical models (Markov Random Fields)
- Factor graphs

Roadmap

KAIST EE

- Data, Models, and Learning
- Models as Functions: Empirical Risk Minimization
- Models as Probabilistic Models: Parameter Estimation (ML and MAP)
- Probabilistic Modeling and Inference
- Directed Graphical Models
- Model Selection

Nested Cross-Validation



- Model selection
  - Tradeoff between model complexity and data fit
  - Occam's razor. Find the simplest model that explains the data resonably well.
- Test set: estimate the generalization performance
- Validation set: choose the best model

Bayesian Model Selection

KAIST EE

- A set of models $\boldsymbol{M} = \{M_1, \ldots, M_k\}$, where each M_k has $\boldsymbol{\theta}_k$ parameters. A prior $\mathbb{P}(M)$ on each model $M \in \boldsymbol{M}$.

$$M_k \sim \mathbb{P}(M), \quad \boldsymbol{\theta}_k \sim \mathbb{P}(\boldsymbol{\theta} \mid M_k), \quad \mathcal{D} \sim \mathbb{P}(\mathcal{D} \mid \boldsymbol{\theta}_k)$$

- Posterior distribution $\mathbb{P}(M_k \mid \mathcal{D}) \propto \mathbb{P}(M_k)\mathbb{P}(\mathcal{D} \mid M_k)$, where we have the following model evidence or marginal likelihood:

$$\mathbb{P}(\mathcal{D} \mid M_k) = \int \mathbb{P}(\mathcal{D} \mid \boldsymbol{\theta}_k)\mathbb{P}(\boldsymbol{\theta}_k \mid M_k)\mathrm{d}\boldsymbol{\theta}_k \quad (***)$$

- MAP for the model: $M^* = \arg\max_{M_k} \mathbb{P}(M_k \mid \mathcal{D})$
- With the uniform model prior (i.e., $\mathbb{P}(M_k) = 1/k$), the MAP estimate equals to maximization of model evidence.

Bayes Factors for Model Comparison

KAIST EE

- Compare two probabilistic models M_1 and M_2:

$$\text{(Posterior odds)} = \frac{\mathbb{P}(M_1 \mid \mathcal{D})}{\mathbb{P}(M_2 \mid \mathcal{D})} = \frac{\frac{\mathbb{P}(\mathcal{D}|M_1)\mathbb{P}(M_1)}{\mathbb{P}(\mathcal{D})}}{\frac{\mathbb{P}(\mathcal{D}|M_2)\mathbb{P}(M_2)}{\mathbb{P}(\mathcal{D})}} = \underbrace{\frac{\mathbb{P}(M_1)}{\mathbb{P}(M_2)}}_{\text{Prior odds}} \underbrace{\frac{\mathbb{P}(\mathcal{D} \mid M_1)}{\mathbb{P}(\mathcal{D} \mid M_2)}}_{\text{Bayes factor}}$$

- $\mathbb{P}(\mathcal{D} \mid M_k)$: How well the data is predicted by the model M_k
- With the uniform model prior, the prior odds $= 1$
- Computation of Bayes factor requires the complex integration (***) in the previous slide. In this case, we rely on some approximations such as MCMC (Markov Chain Monte Carlo).

Summary

KAIST EE

-

Questions?

Review Questions

1)