Generative Graphical Models

INTELLIGENT SYSTEMS FOR PATTERN RECOGNITION (ISPR)

DAVIDE BACCIU – DIPARTIMENTO DI INFORMATICA - UNIVERSITA’ DI PISA

DAVIDE.BACCIU@UNIPI.IT

Generative Learning

- ML models that represent knowledge inferred from data under the form of probabilities
  - Probabilities can be sampled: new data can be generated
  - Supervised, unsupervised, weakly supervised learning tasks
  - Incorporate prior knowledge on data and tasks
  - Interpretable knowledge (how data is generated)
- The majority of the modern task comprises large numbers of variables
  - Modeling the joint distribution of all variables can become impractical
  - Exponential size of the parameter space
  - Computationally impractical to train and predict

The Graphical Models Framework

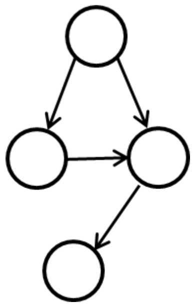
- Representation
  - Graphical models are a compact way to represent exponentially large probability distributions
  - Encode conditional independence assumptions
  - Different classes of graph structures imply different assumptions/capabilities
- Inference
  - How to query (predict with) a graphical model?
  - Probability of unknown X given observations $\boldsymbol{d}$, $P(X|\boldsymbol{d})$
  - Most likely hypothesis
- Learning
  - Find the right model parameter
  - An inference problem after all

Graphical Model Representation

A graph whose **nodes** (vertices) are **random variables** whose **edges** (links) represent **probabilistic relationships** between the variables

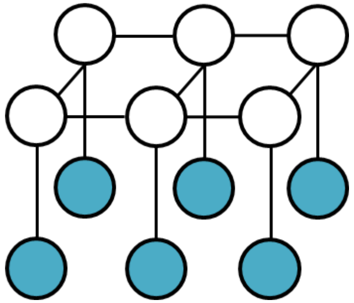
Different classes of graphs

Directed Models



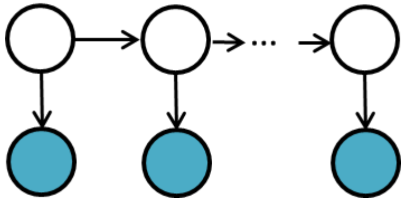
Directed edges express causal relationships

Undirected Models



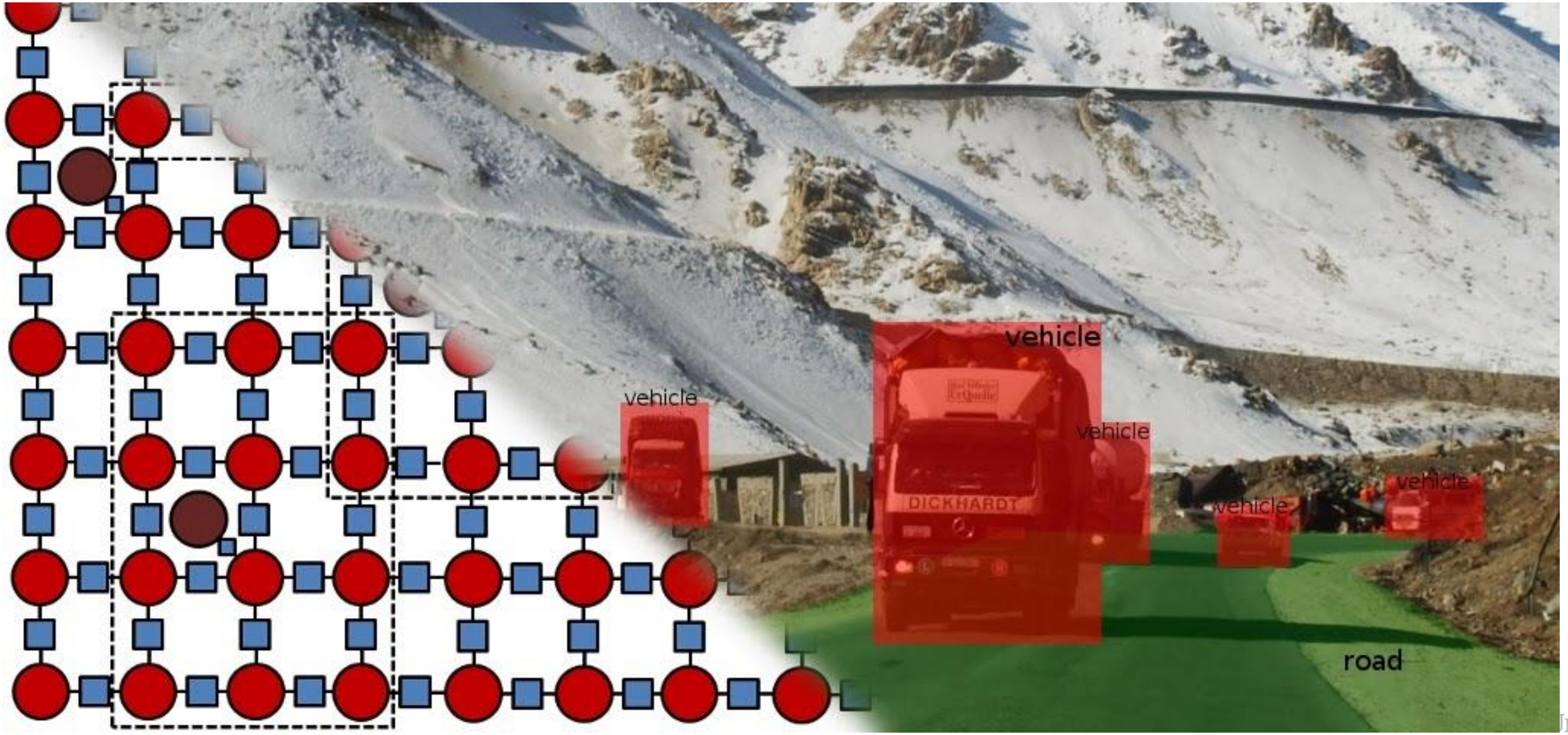
Undirected edges express soft constraints

Dynamic Models

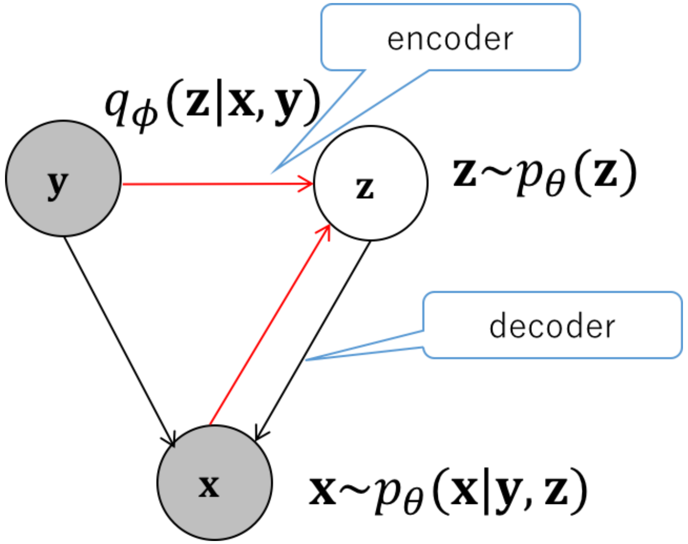


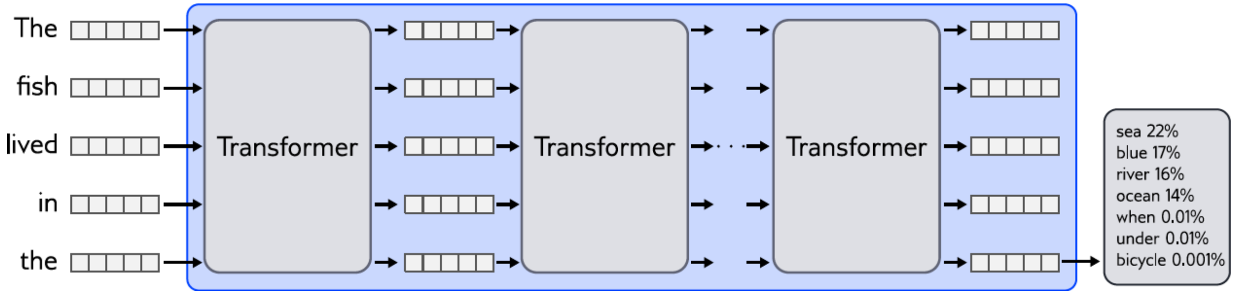
Structure changes to reflect dynamic processes

Generative Models in Machine Vision



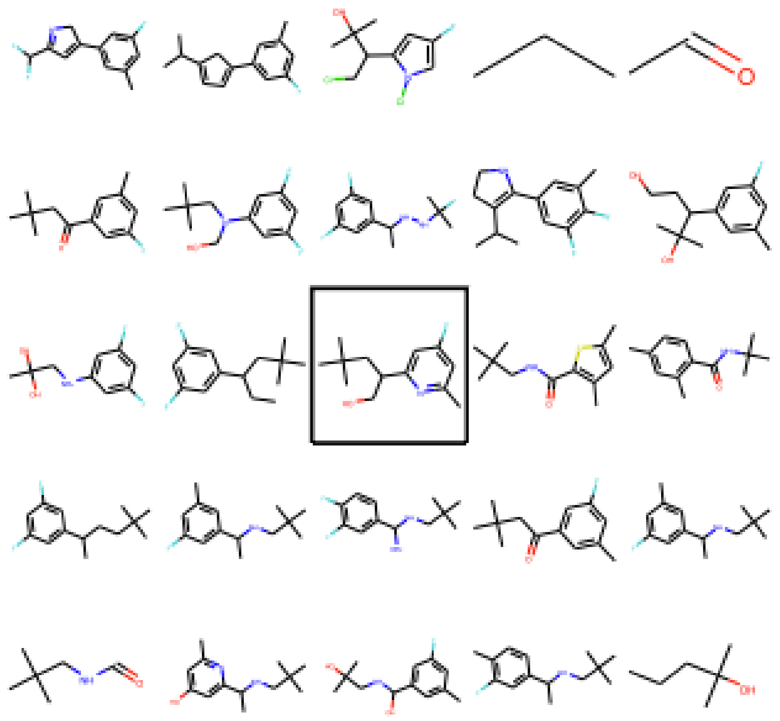
Generative Models in Deep Learning





Probabilistic (generative) learning necessary to understand Generative Deep Learning

Generate New Knowledge



Complex data can be generated if your model is powerful enough to capture its distribution

Probabilistic Models Module

Lesson 1-2	Introduction: Directed and Undirected Graphical Models
Lesson 3-4	Bayesian Networks and Conditional Independence
Lesson 5-6	Causality and structure learning
Lesson 7-9	Dynamic GM: Hidden Markov Model
Lesson 10-11	Undirected GM: Markov Random Fields
Lesson 12	Bayesian Learning: Approximated Inference
Lesson 13	Bayesian Learning: Latent Variable Models
Lesson 14	Bayesian Learning: Sampling Methods
Lesson 15	Bridging Neural and Generative: Boltzmann Machines

Lecture Outline

- Introduction
- A probabilistic refresher
  - Probability theory
  - Conditional independence
- Inference and learning in generative models
- Graphical Models
  - Directed and Undirected Representation
- Conclusions

Content spanning also next lecture

Module content is fully covered by David Barber's book

Probability Refresher

Random Variables

- A Random Variable (RV) is a function describing the outcome of a random process by assigning unique values to all possible outcomes of the experiment
- A RV models an attribute of our data (e.g. age, speech sample,...)
- Use uppercase to denote a RV, e.g. X, and lowercase to denote a value (observation), e.g. x
- A discrete (categorical) RV is defined on a finite or countable list of values Ω
- A continuous RV can take infinitely many values

Probability Functions

- Discrete Random Variables
  - A probability function $P(X = x) \in [0,1]$ measures the probability of a RV X attaining the value x
  - Subject to sum-rule $\sum_{x \in \Omega} P(X = x) = 1$
- Continuous Random Variables
  - A density function $p(t)$ describes the relative likelihood of a RV to take on a value t
  - Subject to sum-rule $\int_{\Omega}^{t} p(t)dt = 1$
  - Defines a probability distribution, e.g. $P(X \leq x) = \int_{-\infty}^{x} p(t)dt$
- Shorthand $P(x)$ for $P(X = x)$ or $P(X \leq x)$

Joint and Conditional Probabilities

If a discrete random process is described by a set of RVs $X_1, \ldots, X_N$, then the joint probability writes

$$P(X_1 = x_1, \ldots, X_N = x_n) = P(x_1 \wedge \cdots \wedge x_n)$$

The joint conditional probability of $x_1, \ldots, x_n$ given y

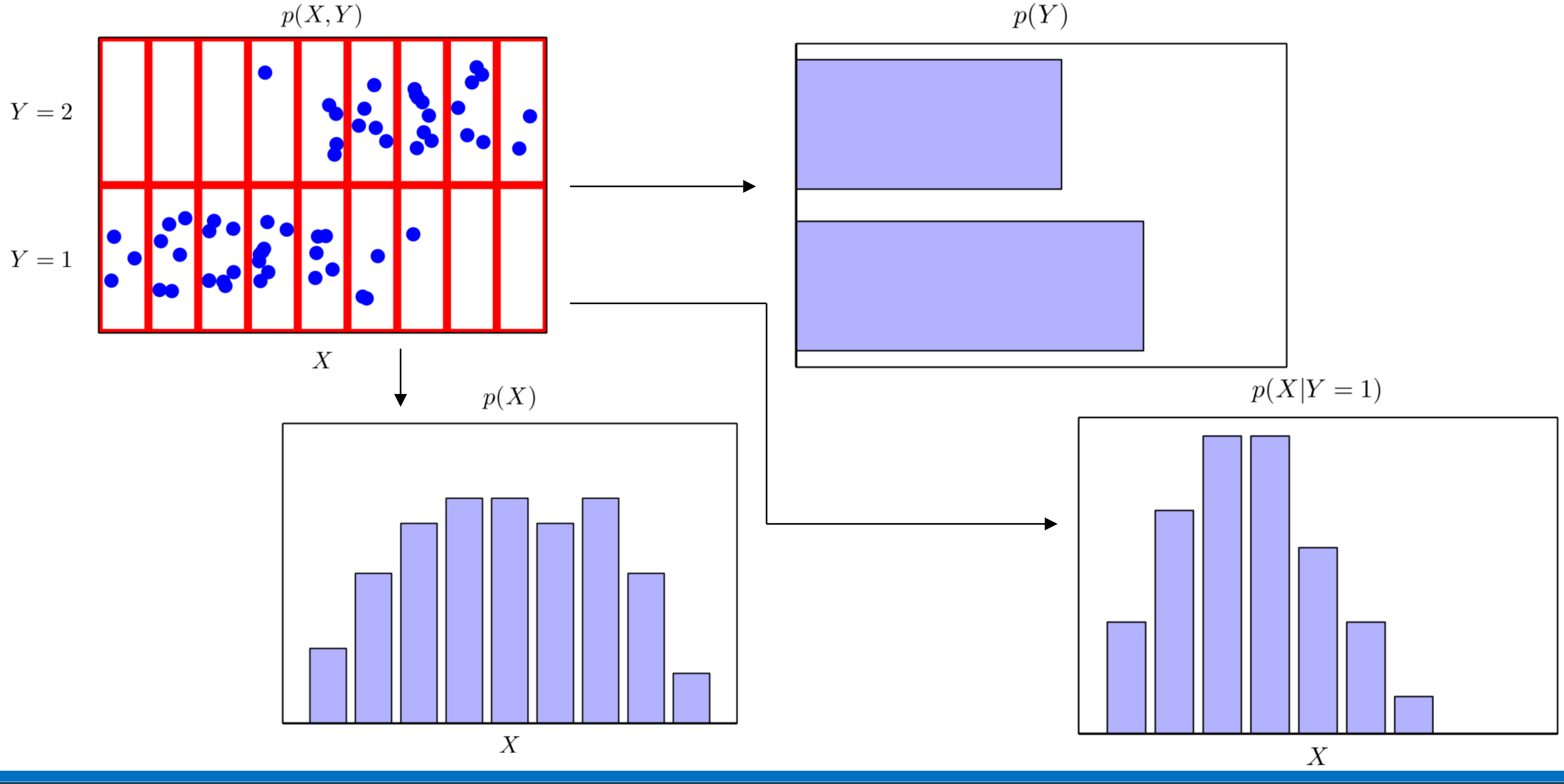
$$P(x_1, \ldots, x_n | y)$$

measures the effect of the realization of an event y on the occurrence of $x_1, \ldots, x_n$

A conditional distribution $P(x|y)$ is actually a family of distributions

- For each y, there is a distribution $P(x|y)$

Probabilities Visually



Chain Rule

Definition (Product Rule a.k.a. Chain Rule)

$$P(x_1, \dots, x_i, \dots, x_n | y) = \prod_{i=1}^{N} P(x_i \,\Big|\, x_1, \dots, x_{i-1}, y)$$

Definition (Marginalization)

Using the sum and product rules together yield to the complete probability

$$P(X_1 = x_1) = \sum_{x_2} P(X_1 = x_1 | X_2 = x_2) P(X_2 = x_2)$$

Bayes Rule (a ML interpretation)

Given hypothesis $h_i \in H$ and observations $\boldsymbol{d}$

$$P(h_i|\boldsymbol{d}) = \frac{P(\boldsymbol{d}|h_i)P(h_i)}{P(\boldsymbol{d})} = \frac{P(\boldsymbol{d}|h_i)P(h_i)}{\sum_j P(\boldsymbol{d}|h_j)P(h_j)}$$

- $P(h_i)$ is the prior probability of h_i
- $P(\boldsymbol{d}|h_i)$ is the conditional probability of observing $\boldsymbol{d}$ given that hypothesis h_i is true (likelihood).
- $P(\boldsymbol{d})$ is the marginal probability of $\boldsymbol{d}$
- $P(h_i|\boldsymbol{d})$ is the posterior probability that hypothesis is true given the data and the previous belief about the hypothesis

Independence and Conditional Independence

- Two RV X and Y are independent if knowledge about X does not change the uncertainty about Y and vice versa

$$I(X,Y) \Leftrightarrow P(X,Y) = P(X|Y)P(Y)$$
$$= P(Y|X)P(X) = P(X)P(Y)$$

- Two RV X and Y are conditionally independent given Z if the realization of X and Y is an independent event of their conditional probability distribution given Z

$$I(X,Y|Z) \Leftrightarrow P(X,Y|Z) = P(X|Y,Z)P(Y|Z)$$
$$= P(Y|X,Z)P(X|Z) = P(X|Z)P(Y|Z)$$

- Shorthand $X \perp Y$ for $I(X,Y)$ and $X \perp Y|Z$ for $I(X,Y|Z)$

Wrapping Up....

- We know how to represent the world and the observations
  - Random Variables $\Longrightarrow X_1, \ldots, X_N$
  - Joint Probability Distribution $\Longrightarrow P(X_1 = x_1, \ldots, X_N = x_n)$
- We have rules for manipulating the probabilistic knowledge
  - Sum-Product
  - Marginalization
  - Bayes
  - Conditional Independence
- In this context, learning is about discovering the values for $P(X_1 = x_1, \ldots, X_N = x_n)$

Inference and learning with probabilities

Inference and Learning in Probabilistic Models

Inference - How can one determine the distribution of the values of one/several RV, given the observed values of others?

$$P(graduate|exam_1, \ldots, exam_n)$$

Machine Learning view - Given a set of observations (data) $\boldsymbol{d}$ and a set of hypotheses $\{h_i\}_i^K = 1$, how can I use them to predict the distribution of a RV X?

Learning - A very specific inference problem!

- Given a set of observations $\boldsymbol{d}$ and a probabilistic model of a given structure, how do I find the parameters θ of its distribution?
- Amounts to determining the best hypothesis h_θ regulated by a (set of) parameters θ

UNIVERSITÀ DI PISA

3 Approaches to Inference

Bayesian Consider all hypotheses weighted by their probabilities

$$P(X|\boldsymbol{d}) = \sum_i P(X|h_i)P(h_i\,|\boldsymbol{d})$$

MAP Infer X from $P(X|h_{MAP})$ where h_{MAP} is the Maximum a-Posteriori hypothesis given $\boldsymbol{d}$

$$h_{MAP} = \arg\max_{h \in H} P(h|\boldsymbol{d}) = \arg\max_{h \in H} P(\boldsymbol{d}|h)P(h)$$

ML Assuming uniform priors $P(h_i) = P(h_j)$, yields the Maximum Likelihood (ML) estimate $P(X|h_{ML})$

$$h_{ML} = \arg\max_{h \in H} P(\boldsymbol{d}|h)$$

Considerations About Bayesian Inference

- The Bayesian approach is optimal but poses computational and analytical tractability issues

$$P(X|\boldsymbol{d}) = \int_H P(X|h)P(h|\boldsymbol{d})dh$$

- ML and MAP are point estimates of the Bayesian since they infer based only on one most likely hypothesis
- MAP and Bayesian predictions become closer as more data gets available
- MAP is a regularization of the ML estimation
  - Hypothesis prior $P(h)$ embodies trade-off between complexity and degree of fit
  - Well-suited to working with small datasets and/or large parameter spaces

Regularization

- $P(h)$ introduces preference across hypotheses
- Penalize complexity
  - Complex hypotheses have a lower prior probability
  - Hypothesis prior embodies trade-off between complexity and degree of fit
- MAP hypothesis h_{MAP}

$$\max_h P(\boldsymbol{d}|h)P(h) \equiv \min_h - \log_2(P(\boldsymbol{d}|h)) - \log_2 P(h)$$

Number of bits required to specify h → $\log_2 P(h)$

- MAP $\Longrightarrow$ choosing the hypothesis that provides maximum compression
- MAP is a regularization of the ML estimation

UNIVERSITÀ DI PISA

Maximum-Likelihood (ML) Learning

Find the model θ that is most likely to have generated the data $\boldsymbol{d}$

$$\theta_{ML} = \arg\max_{\theta \in \Theta} P(\boldsymbol{d}|\theta)$$

from a family of parameterized distributions $P(x|\theta)$.

Optimization problem that considers the Likelihood function

$$\mathcal{L}(\theta|x) = P(x|\theta)$$

to be a function of θ.

Can be addressed by solving

$$\frac{\partial \mathcal{L}(\theta|x)}{\partial \theta} = 0$$

Learning assuming that all RV X are visible, as in Naïve Bayes

UNIVERSITÀ DI PISA

ML Learning with Hidden Variables

What if my probabilistic models contains both

- Observed random variables **X** (i.e. for which we have training data)
- Unobserved (hidden/latent) variables **Z** (e.g. data clusters)

ML learning can still be used to estimate model parameters

- The Expectation-Maximization algorithm which optimizes the complete likelihood

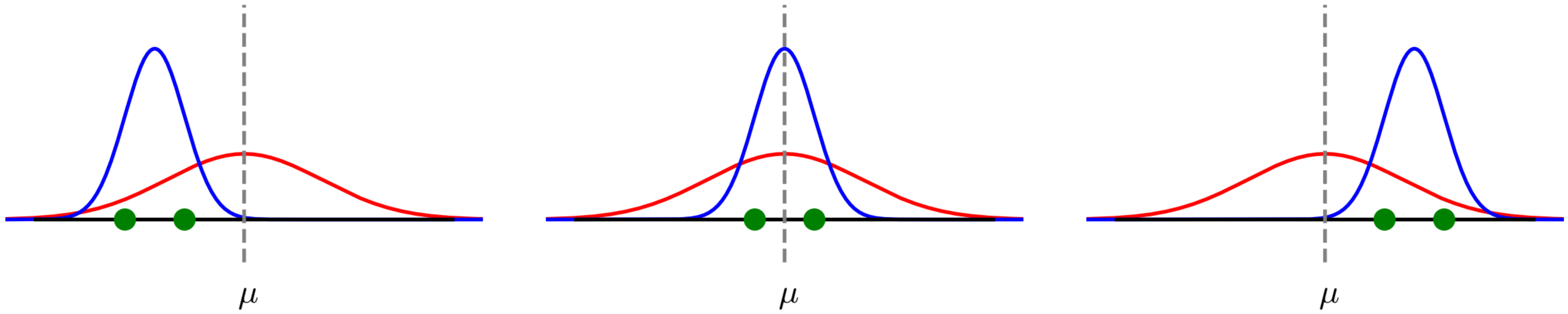
$$\mathcal{L}_c(\theta|\boldsymbol{X},\boldsymbol{Z}) = P(\boldsymbol{X},\boldsymbol{Z}|\theta) = P(\boldsymbol{Z}|\boldsymbol{X},\theta)P(\boldsymbol{X}|\theta)$$

- A 2-step iterative process

$$\theta^{(k+1)} = \arg\max_{\theta} \sum_{\boldsymbol{z}} P\left(\boldsymbol{Z} = \boldsymbol{z} \middle| \boldsymbol{X}, \theta^{(k)}\right) \log \mathcal{L}_c(\theta|\boldsymbol{X},\boldsymbol{Z} = \boldsymbol{z})$$

We will see EM in action in HMMs

Bias of ML Learning



Graphical Models

Joint Probabilities and Exponential Complexity

Discrete Joint Probability Distribution as a Table

X_1	…	X_i	…	X_n	$P(X_1, \dots, X_n)$
X_1'	…	X_i'	…	X_n'	$P(X_1', \dots, X_n')$
X_1^l	…	X_i^l	…	X_n^l	$P(X_1^l, \dots, X_n^l)$

- Describes $P(X_1, \dots, X_n)$ for all the RV instantiations
- For n binary RV X_i the table has 2^n entries!

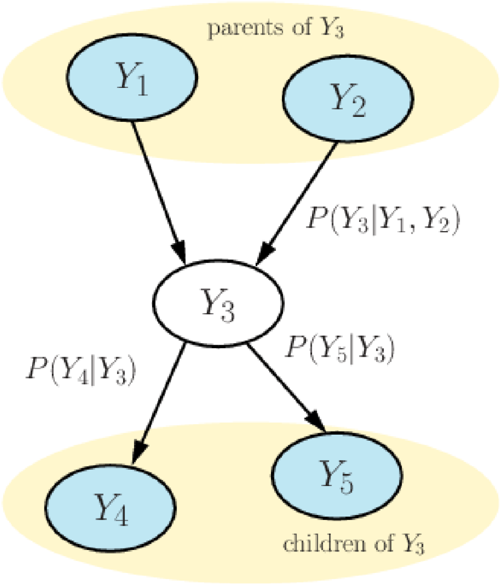
Any probability can be obtained from the **Joint Probability Distribution** $P(X_1, \dots, X_n)$ by **marginalization** but again at an exponential cost (e.g. 2^{n-1} for a marginal distribution from binary RV).

UNIVERSITÀ DI PISA

Graphical Models

- Compact graphical representation for exponentially large joint distributions
- Simplifies marginalization and inference algorithms
- Allow to incorporate prior knowledge concerning causal relationships and associations between RV
  - Directed Graphical Models a.k.a. Bayesian Networks
  - Undirected Graphical Models a.k.a. Markov Random Fields

A Sneak Peak of Next Lecture



Bayesian networks

- Directed Acyclic Graphs (DAGs) with nodes representing variables
- Edges describing conditional independence relationships

Conditional Probabilities local to each node describe the probability distribution given its parents and providing a factorization of the joint distribution

$$P(Y_1, \ldots, Y_N) = \prod_{i=1}^{N} P(Y_i \mid pa(Y_i))$$

Generative Models in Code

- PyMC3 - Bayesian statistics and probabilistic ML; gradient-based Markov chain Monte Carlo variational inference (Python, Theano)
- Edward - Bayesian statistics and ML, deep learning, and probabilistic programming (Python, TensorFlow)
- Pyro - Deep probabilistic programming (Python, PyTorch)
- TensorFlow Probability - Combine probabilistic models and deep learning with GPU/TPU support (Python)
- PyStruct - Markov Random Field models in Python (some of them)
- Pgmpy - Python package for Probabilistic Graphical Models
- Stan - Probabilistic programming language for statistical inference (native C++, PyStan package)

Take Home Messages

- Generative models as a gateway for next-gen deep learning
- Everything is an inference problem, including learning
- Directed graphical models
  - Represent asymmetric (causal) relationships between RV and conditional probabilities in compact way
- Undirected graphical models
  - Represent bi-directional relationships (e.g. constraints)

Important Note

Next Wednesday lecture (05/03/2025) is canceled due to Student's General Assembly. Will be recovered eventually (TBD)

Next 2 Lectures (04-06/03/2025)

Conditional independence: representation and learning

- Bayesian Networks
- Markov properties in Bayesian Networks
- Conditional independence as a graph-theoretic concept
- Conditional independence in undirected models
- Learning conditional independence relationships from data

Guest lectures by Riccardo Massidda