

Introduction to Probabilistic Models

Antonio Salmerón

Department of Mathematics
University of Almería
antonio.salmeron@ual.es

Helsinki, June 13 2022

Motivation



"I always wondered how it would be if a Superior species landed on Earth and showed us How they played chess. Now I know it."

Peter Heine Nielsen.
Chess Grand Master and Magnus Carlsen's coach

The question is,

- Can we (**humans**) learn (**interpret**) anything from it?

Motivation



"I always wondered how it would be if a Superior species landed on Earth and showed us How they played chess. Now I know it."

Peter Heine Nielsen.
Chess Grand Master and Magnus Carlsen's coach

The question is,

- Can we (**humans**) learn (**interpret**) anything from it?

Examples: Predictive medicine



What should I do to prevent players from being injured?

Examples: Self-driving cars



Can I predict an event in advance so that I can avoid it?

Examples: Image generation

A majestic oil painting of a raccoon Queen wearing red French royal gown. The painting is hanging on an ornate wall decorated with wallpaper.



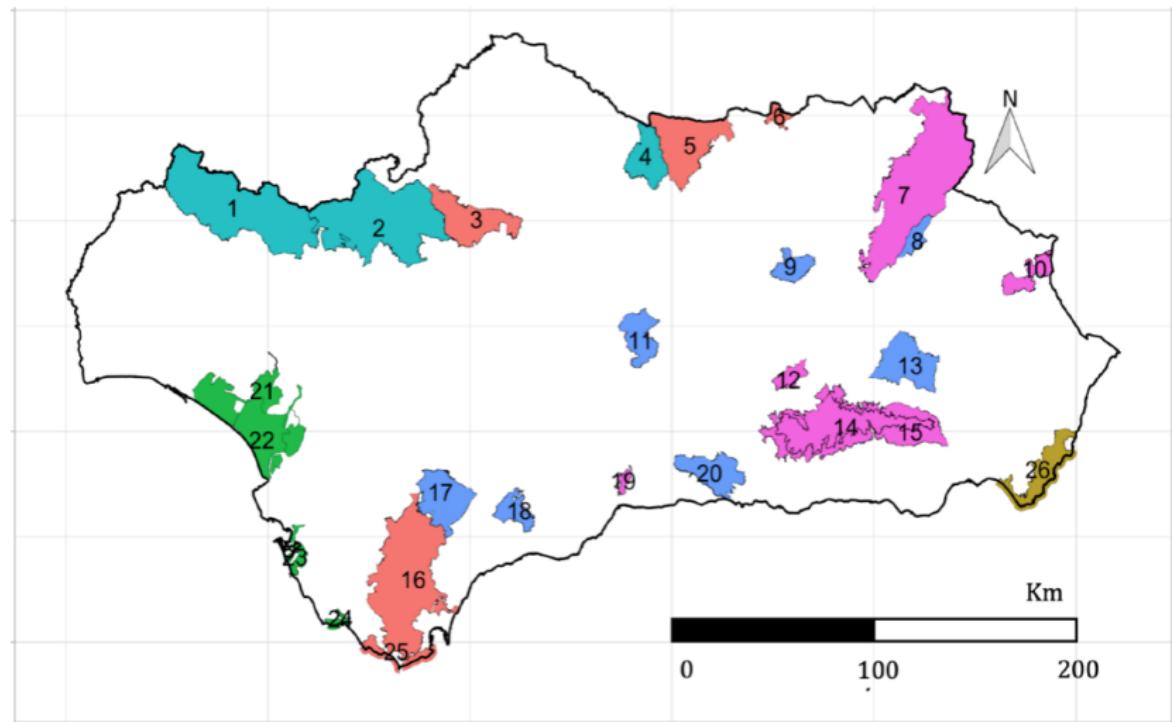
Imagen

unprecedented photorealism × deep level of language understanding

Google Research, Brain Team

Generating images from written descriptions

Examples: Land use



Monitoring protected areas

Probabilistic models

All the previous examples:

- Operate in environments where large amounts of data are available
- However, data don't cover all the possible scenarios ⇒ **UNCERTAINTY**
- Use a **probabilistic model** learnt from data
- Use inference algorithms to carry out **prediction** and **structure analysis**

Probabilistic models offer:

- Principled quantification of uncertainty
- Natural way of dealing with missing data
- **Interpretability**

Probabilistic Machine Learning

What is machine learning?

"We say that a computer program P learns from experience E with respect to some class of tasks T and a performance measure R , if its performance on the tasks in T , measured in terms of R , improve with experience E ".

(Tom Mitchell, 1997)

Easy example: linear regression

- We have data about two variables X and Y
- We want to predict the value of Y from the value of X
- To solve this task, we decide to use a linear regression model

$$\hat{y} = a + bx$$

- As performance measure, we use

$$\text{rmse}(\mathbf{y}, \hat{\mathbf{y}}) = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

with $\mathbf{y} = \{y_1, \dots, y_n\}$ denoting the data and $\hat{\mathbf{y}} = \{\hat{y}_1, \dots, \hat{y}_n\}$ denoting the model estimates

Is this really ML???

Easy example: linear regression

- We have data about two variables X and Y
- We want to predict the value of Y from the value of X
- To solve this task, we decide to use a linear regression model

$$\hat{y} = a + bx$$

- As performance measure, we use

$$\text{rmse}(\mathbf{y}, \hat{\mathbf{y}}) = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

with $\mathbf{y} = \{y_1, \dots, y_n\}$ denoting the data and $\hat{\mathbf{y}} = \{\hat{y}_1, \dots, \hat{y}_n\}$ denoting the model estimates

Is this really ML???

Easy example: linear regression

- The linear regression model we have used **IS NOT** probabilistic
- By the end of the lecture, we'll see how it can be approached from a probabilistic point of view

Uncertainty

You have probably heard about two types of uncertainty:

- **Aleatoric:** Due to randomness, i.e. the variability in the outcome of an experiment due to random effects
- **Epistemic:** Due to lack of knowledge

Example

- Assume we want to predict Y from X
- We estimate a joint distribution $p(x, y)$
- We predict Y using $p(y|x) = p(x, y)/p(x)$
- We observe $X = x$, what is our model prediction for Y ?

Uncertainty

You have probably heard about two types of uncertainty:

- **Aleatoric:** Due to randomness, i.e. the variability in the outcome of an experiment due to random effects
- **Epistemic:** Due to lack of knowledge

Example

- Assume we want to predict Y from X
- We estimate a joint distribution $p(x, y)$ [EPISTEMIC]
- We predict Y using $p(y|x) = p(x, y)/p(x)$
- We observe $X = x$, what is our model prediction for Y ? [ALEATORIC]

Uncertainty

You have probably heard about two types of uncertainty:

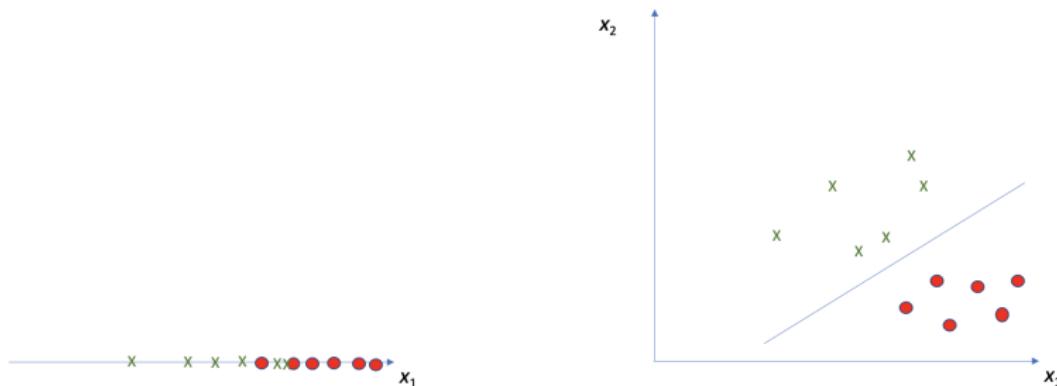
- **Aleatoric:** Due to randomness, i.e. the variability in the outcome of an experiment due to random effects
- **Epistemic:** Due to lack of knowledge

Example

- Assume we want to predict Y from X
- We estimate a joint distribution $p(x, y)$ [REDUCIBLE]
- We predict Y using $p(y|x) = p(x, y)/p(x)$
- We observe $X = x$, what is our model prediction for Y ? [IRREDUCIBLE]

Uncertainty

- Epistemic uncertainty can be reduced gathering more data, but also increasing the number of features.



Probabilistic graphical models

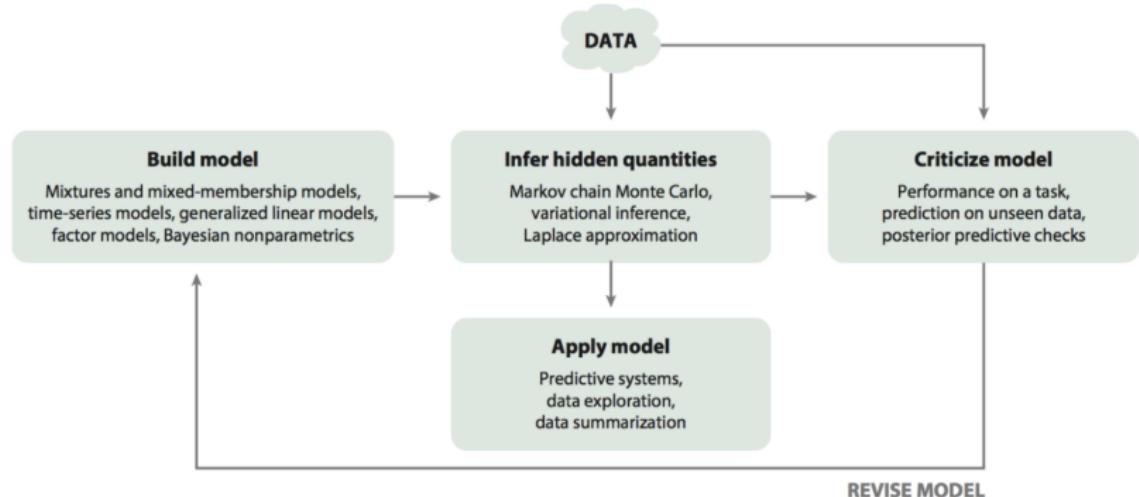
What we need from probabilistic models:

- Ability to operate in **high dimensional** spaces
- Support **efficient** inference and learning

Probabilistic graphical models offer:

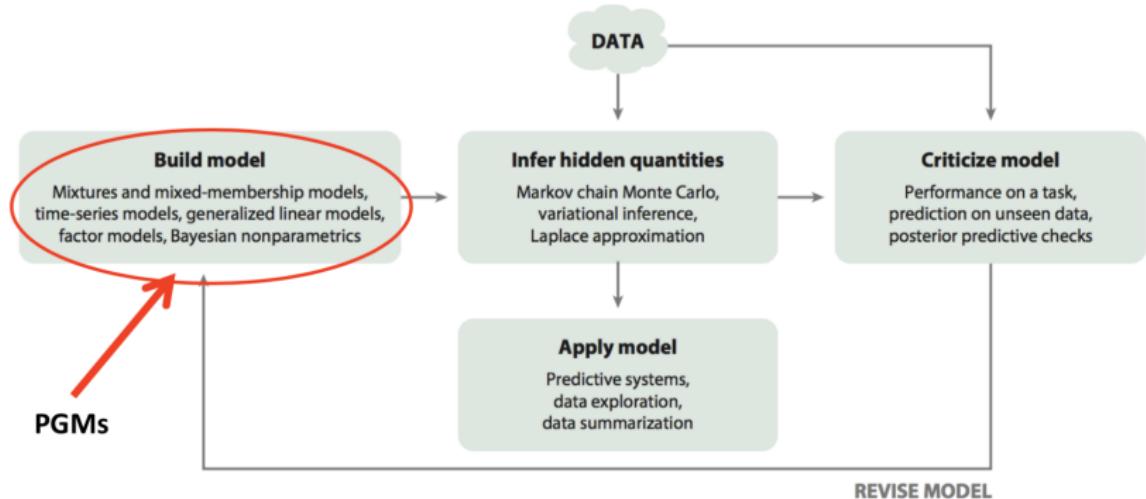
- **Structured** specification of high dimensional distributions in terms of low dimensional factors
- **Efficient** inference and learning taking advantage of the structure
- **Graphical** representation interpretable by humans

The probabilistic modeling cycle



Blei, David M. "Build, compute, critique, repeat: Data analysis with latent variable models." *Annual Review of Statistics and Its Application* 1 (2014): 203-232.

The probabilistic modeling cycle



Blei, David M. "Build, compute, critique, repeat: Data analysis with latent variable models." *Annual Review of Statistics and Its Application* 1 (2014): 203-232.

Learning probabilistic models from data: frequentist approach

- A random variable X
- A probability density/mass function f associated with X
- A parameter θ , whose value is fixed but unknown
- We assume f is known except for parameter θ , and denote this fact as $f(x; \theta)$ or $f(x|\theta)$
- For a sample X_1, \dots, X_n drawn from $f(x|\theta)$, the likelihood function is

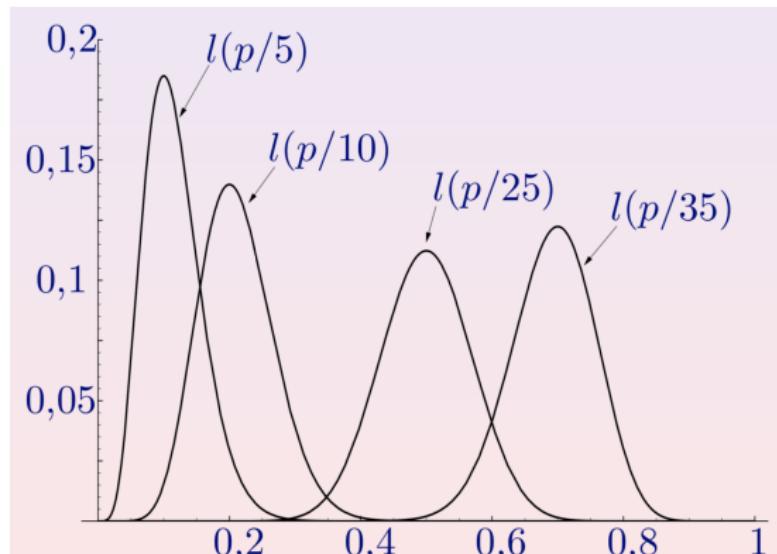
$$l(\theta|x_1, \dots, x_n) = f(x_1, \dots, x_n|\theta) = \prod_{i=1}^n f(x_i|\theta)$$

i.e. the joint density/mass of the sample regarded as a function of the parameter

Learning probabilistic models from data: frequentist approach

Assume a sample of size 1, $X \sim \mathcal{B}(50, p)$

The likelihood function is $l(p/x) = \binom{50}{x} p^x (1-p)^{50-x}$



Learning probabilistic models from data: Bayesian approach

- Parameters are modeled as random variables and information about them can be included prior to observing data
- By Bayes' rule, the prior information is combined with the likelihood, yielding a posterior distribution that is used for making inferences about the parameter

Learning probabilistic models from data: Bayesian approach

Distributions in a Bayesian model

- The prior distribution, $\pi(\theta)$
- The joint distribution of (X, θ) , $\psi(x, \theta) = f(x|\theta)\pi(\theta)$
- The prior predictive distribution of X , $m(x) = \int_{\theta} f(x|\theta)\pi(\theta) d\theta$
- The posterior distribution of θ ,

$$\pi(\theta|x) = \frac{f(x|\theta)\pi(\theta)}{\int_{\theta} f(x|\theta)\pi(\theta) d\theta}$$

- The predictive distribution given $\mathbf{x} = \{x_1, \dots, x_n\}$:

$$f(x_{n+1}|\mathbf{x}) = \int_{\theta} f(x_{n+1}|\theta, \mathbf{x})\pi(\theta|\mathbf{x})d\theta = \int_{\theta} f(x_{n+1}|\theta)\pi(\theta|\mathbf{x})d\theta$$

Learning probabilistic models from data: Bayesian approach. Example

- Assume a sample $X_1, X_2, \dots, X_n \sim \mathcal{B}(1, p)$ and $p \sim \mathcal{U}(0, 1)$
- Then the likelihood and the prior are,

$$f(x_1, \dots, x_n | p) = p^{\sum x_i} (1-p)^{n - \sum x_i}, \quad \text{with } x_i = 0, 1; \quad p \in (0, 1),$$
$$\pi(p) = 1, \quad \text{if } p \in (0, 1)$$

- The posterior distribution is

$$\begin{aligned}\pi(p|x_1, \dots, x_n) &= \frac{f(x_1, \dots, x_n | p) \pi(p)}{\int_0^1 f(x_1, \dots, x_n | p) \pi(p) dp} = \frac{p^{\sum x_i} (1-p)^{n - \sum x_i}}{\int_0^1 p^{\sum x_i} (1-p)^{n - \sum x_i} dp} \\ &= \frac{\Gamma(n+2)}{\Gamma(\sum x_i + 1) \Gamma(n - \sum x_i + 1)} p^{\sum x_i} (1-p)^{n - \sum x_i}\end{aligned}$$

which corresponds to $\boxed{\text{Be}\left(\sum x_i + 1, n - \sum x_i + 1\right)}$

Very easy to compute for some models

Learning probabilistic models from data: Bayesian approach. Example

- Assume a sample $X_1, X_2, \dots, X_n \sim \mathcal{B}(1, p)$ and $p \sim \mathcal{U}(0, 1)$
- Then the likelihood and the prior are,

$$f(x_1, \dots, x_n | p) = p^{\sum x_i} (1-p)^{n-\sum x_i}, \quad \text{with } x_i = 0, 1; \quad p \in (0, 1),$$
$$\pi(p) = 1, \quad \text{if } p \in (0, 1)$$

- The posterior distribution is

$$\begin{aligned}\pi(p|x_1, \dots, x_n) &= \frac{f(x_1, \dots, x_n | p)\pi(p)}{\int_0^1 f(x_1, \dots, x_n | p)\pi(p) dp} = \frac{p^{\sum x_i} (1-p)^{n-\sum x_i}}{\int_0^1 p^{\sum x_i} (1-p)^{n-\sum x_i} dp} \\ &= \frac{\Gamma(n+2)}{\Gamma(\sum x_i + 1)\Gamma(n - \sum x_i + 1)} p^{\sum x_i} (1-p)^{n-\sum x_i}\end{aligned}$$

which corresponds to $\boxed{\text{Be}\left(\sum x_i + 1, n - \sum x_i + 1\right)}$

Very easy to compute for some models

Learning probabilistic models from data: Bayesian approach. Example

Likelihood	Prior	Posterior
$\mathcal{B}(1, \theta)$	$Be(\alpha, \beta)$	$Be(\alpha + \sum_{i=1}^n x_i, \beta + n - \sum_{i=1}^n x_i)$
$\mathcal{NB}(r, \theta)$	$Be(\alpha, \beta)$	$Be(\alpha + rn, \beta - nr + \sum_{i=1}^n x_i)$
$\mathcal{G}(\theta)$	$Be(\alpha, \beta)$	$Be(\alpha + n, \beta + \sum_{i=1}^n x_i)$
$\mathcal{MN}(n, \theta_1, \dots, \theta_k)$	$Dir(\alpha_1, \dots, \alpha_k)$	$Dir(\alpha_1 + x_1, \dots, \alpha_k + x_k)$
$P(\theta)$	$\Gamma(\alpha, \beta)$	$\Gamma(\alpha + \sum_{i=1}^n x_i, \beta + n)$
$Exp(\theta)$	$\Gamma(\alpha, \beta)$	$\Gamma(\alpha + n, \beta + \sum_{i=1}^n x_i)$
$\mathcal{N}(\mu, \tau)$	$\mathcal{N}(\mu_0, \tau_0)$	$\mathcal{N}(\frac{\tau_0\mu_0 + n\tau\bar{x}}{\tau_0 + n\tau}, \tau_0 + n\tau)$
$\mathcal{N}(\underline{\mu}, \tau)$	$\Gamma(\alpha_0, \beta_0)$	$\Gamma(\alpha_0 + \frac{n}{2}, \beta_0 + \frac{1}{2} \sum_{i=1}^n (x_i - \mu)^2)$

Learning probabilistic models from data: Bayesian approach. Example

- The prior predictive distribution is

$$\begin{aligned}m(x) &= \int_0^1 p^x (1-p)^{1-x} dp \\&= \frac{\Gamma(x+1)\Gamma(2-x)}{\Gamma(3)} \int_0^1 \frac{\Gamma(3)}{\Gamma(x+1)\Gamma(2-x)} p^x (1-p)^{1-x} dp \\&= \frac{x!(1-x)!}{2} = \boxed{\frac{1}{2}} \quad \text{with } x = 0, 1\end{aligned}$$

- The (posterior) predictive distribution is

$$\begin{aligned}f(x|x_1, \dots, x_n) &= \int_0^1 p^x (1-p)^{1-x} \frac{\Gamma(n+2)}{\Gamma(\sum x_i + 1)\Gamma(n - \sum x_i + 1)} p^{\sum x_i} (1-p)^{n - \sum x_i} dp \\&= \frac{\Gamma(n+2)}{\Gamma(\sum x_i + 1)\Gamma(n - \sum x_i + 1)} \int_0^1 p^{x + \sum x_i} (1-p)^{n+1-(x+\sum x_i)} dp \\&= \frac{\Gamma(n+2)}{\Gamma(\sum x_i + 1)\Gamma(n - \sum x_i + 1)} \frac{\Gamma(x+1 + \sum x_i)\Gamma(n+2 - (x + \sum x_i))}{\Gamma(n+3)}\end{aligned}$$

Learning probabilistic models from data: Bayesian approach

- The method above is known as *fully Bayesian* approach
- Sometimes, we don't need to compute the denominator of the posterior distribution, in which case θ can be estimated as

$$\hat{\theta} = \arg \max_{\theta} \log f(x_1, \dots, x_n | \theta) + \log \pi(\theta)$$

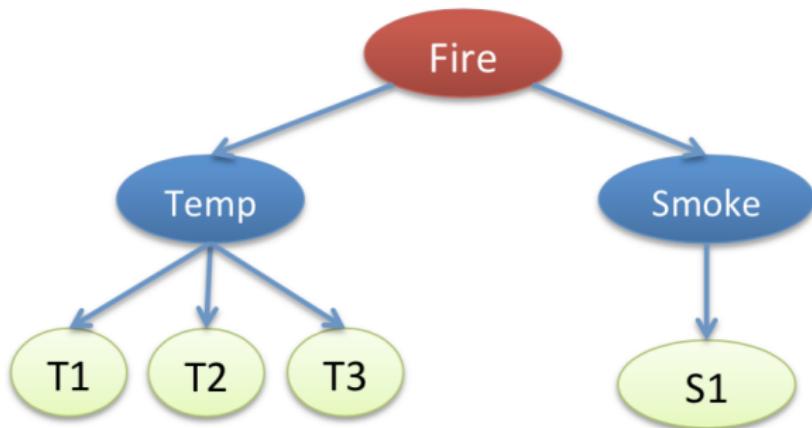
known as the **MAP** estimator

- Note that we could also choose

$$\hat{\theta} = \arg \max_{\theta} \log f(x_1, \dots, x_n | \theta)$$

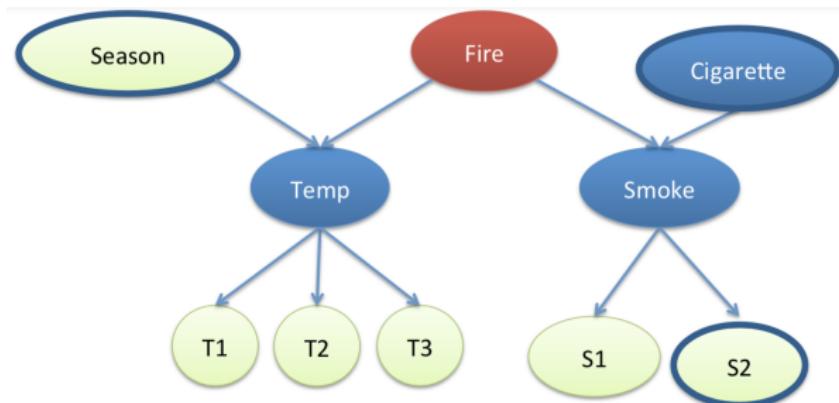
which is actually the **MLE**

Bayesian networks



$$p(f, t, s, t_1, t_2, t_3, s_1) = p(t_1|t)p(t_2|t)p(t_3|t)p(s_1|s)p(t|f)p(s|f)p(f)$$

Bayesian networks: modular structure



$$p(\text{se}, f, c, t, s, t_1, t_2, t_3, s_1) = p(t_1|t)p(t_2|t)p(t_3|t)p(s_1|s)p(s_2|s) \\ p(t|se, f)p(s|f, c)p(se)p(f)p(c)$$

Bayesian networks: definition

A Bayesian network over random variables X_1, \dots, X_n consists of

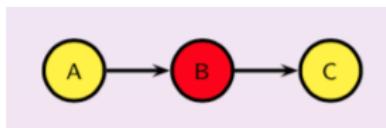
- A DAG $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = \{X_1, \dots, X_n\}$
- A set of local conditional distributions $\mathcal{P} = \{p(X_i|pa(X_i)), X_i \in \mathcal{V}\}$ where $pa(X_i)$ denotes the parents of X_i according to \mathcal{E}

Every Bayesian network encodes a joint distribution factorized as

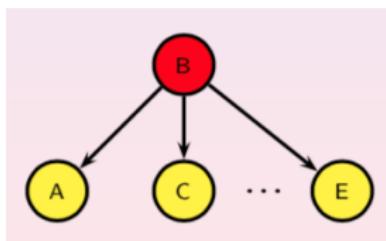
$$p(X_1, \dots, X_n) = \prod_{i=1}^n p(X_i|pa(X_i))$$

Interpreting Bayesian network structures: d -separation

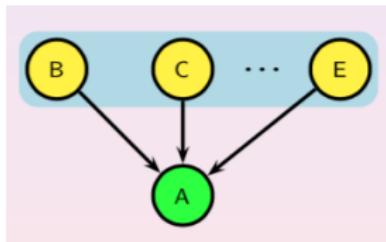
- Serial connection



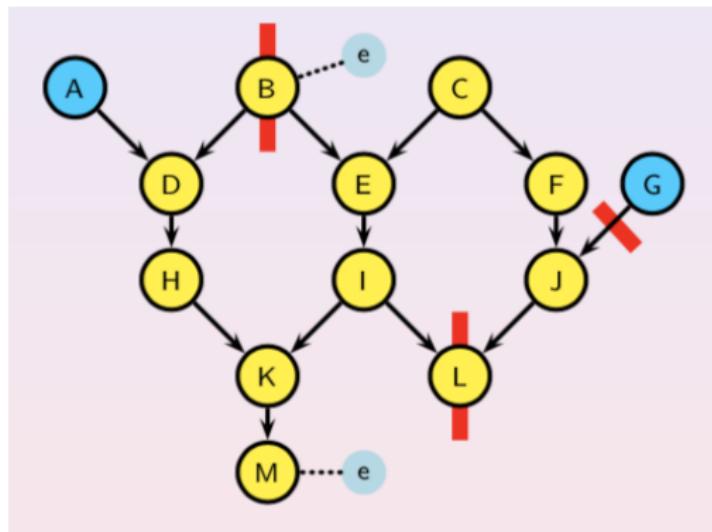
- Diverging connection



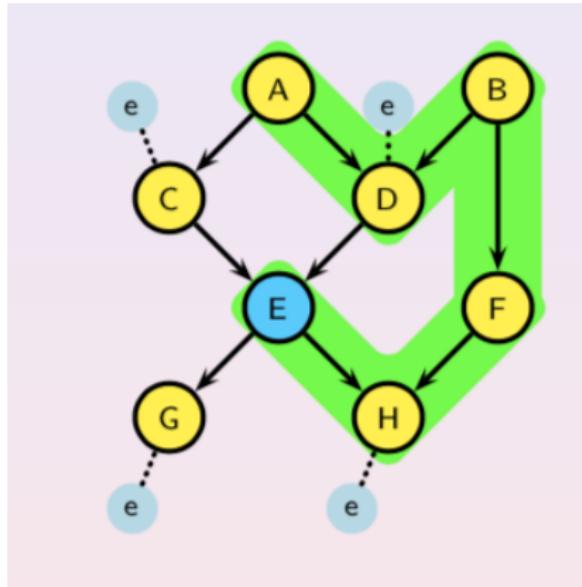
- Converging connection



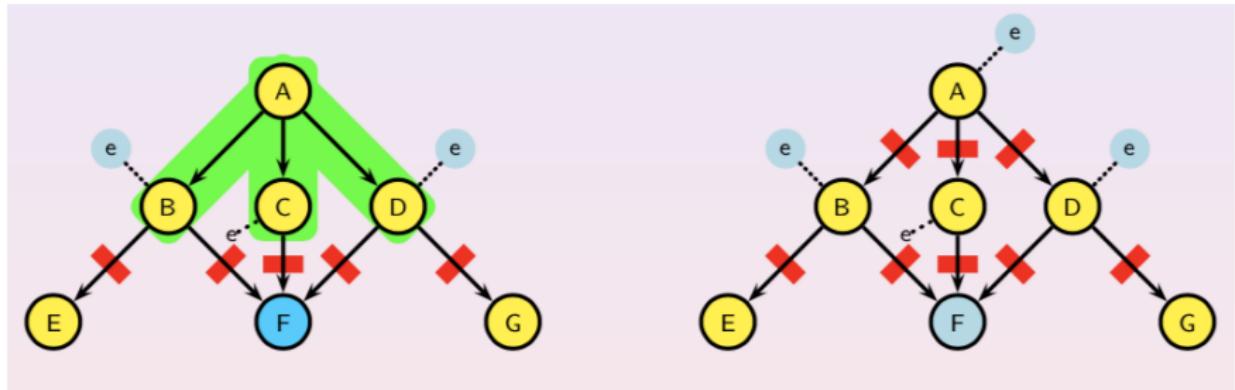
d -separation example



d -separation example



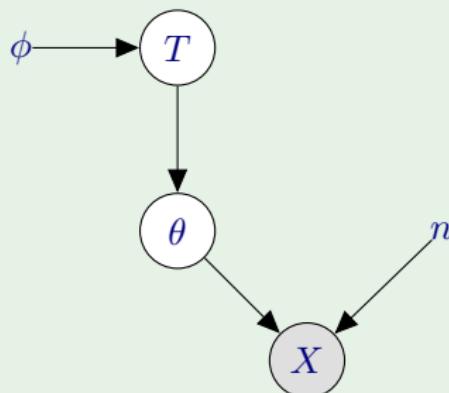
d -separation example



Some simple examples

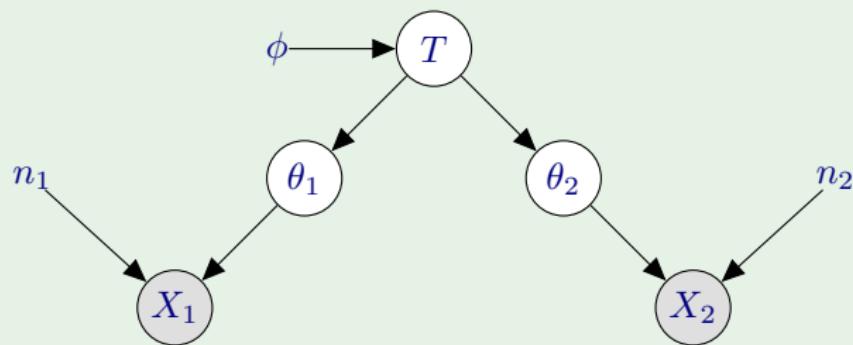
- Random variables (and parameters) inside circles
- Fixed quantities, without circle

Tossing a coin



Some simple examples

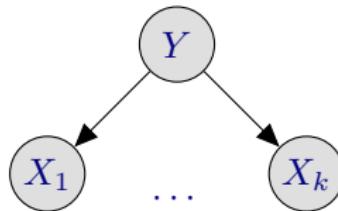
Two coins



Some simple examples

Naive Bayes

- Predicting the value of a categorical variable, Y from a set of features X_1, \dots, X_k



$$p(y|x_1, \dots, x_k) \propto p(x_1, \dots, x_k|y)p(y) = p(y) \prod_{i=1}^k p(x_i|y)$$

Generative and discriminative models

Predicting Y from X

Generative model

- Learn $p(x, y) = p(x|y)p(y)$ from data
- Compute $p(y|x)$ using Bayes rule

- Naive Bayes, Bayesian networks in general, ...
- Can be used to generate synthetic data
- Higher asymptotic error but reached more quickly

Discriminative model

- Estimate $p(y|x)$ directly from data

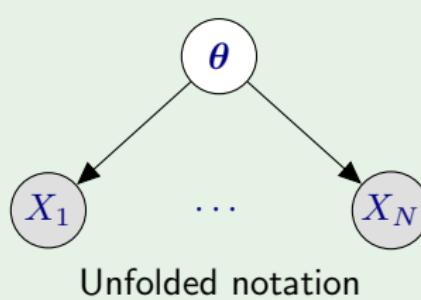
- Logistic regression, NNs, ...
- Lower asymptotic error but reached more slowly

Plate notation

The idea is to avoid **repeated substructures**

Example: independent data points

- Assume the elements in a sample X_1, \dots, X_N are independent if the parameter θ is known



Unfolded notation

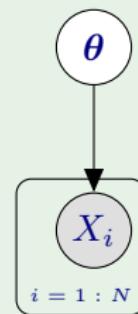


Plate notation

Plate notation

Example: linear regression

- $Y_i|\{\mathbf{w}, \mathbf{x}_i\} = \mathbf{w}^\top \mathbf{x}_i + \epsilon_i$ with $\mathbf{x}_i = [1, x_i]^\top$
- $\epsilon_i \sim \mathcal{N}(0, 1/\gamma)$ with γ known
- $\mathbf{w} \sim \mathcal{N}(\boldsymbol{\mu}_0 = \mathbf{0}, \boldsymbol{\Sigma}_0 = \mathbf{I}_{2 \times 2})$

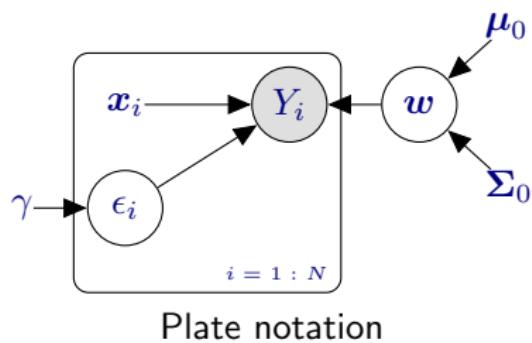
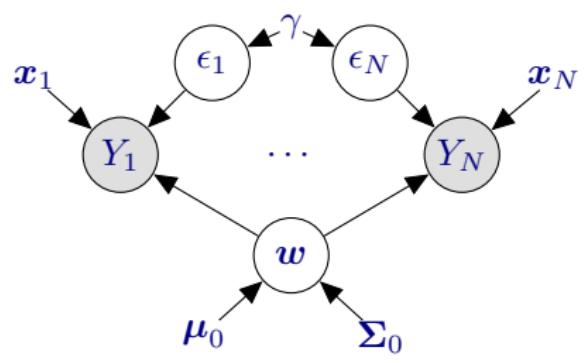


Plate notation



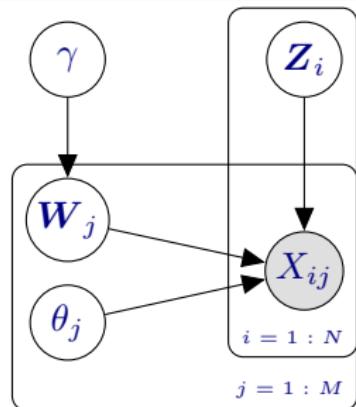
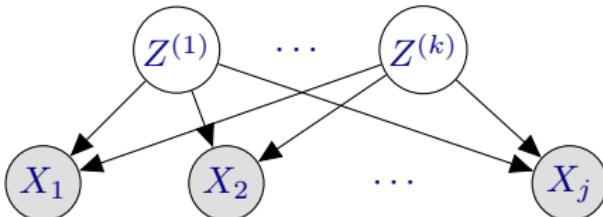
Unfolded notation

Latent variable models

Example: factor analysis model

- In general, **latent variable models** are regarded as probabilistic models where some of the variables cannot be observed
- FA summarizes a high-dimensional observation \mathbf{X} of correlated variables by a smaller set of factors \mathbf{Z} assumed independent a priori
- Model formulation:
 - $\mathbf{Z}_i \sim \mathcal{N}(\boldsymbol{\mu}_0 = \mathbf{0}, \boldsymbol{\Sigma}_0 = \mathbf{I})$, $i = 1, \dots, N$
 - $X_{i,j} | \{\mathbf{z}_i, \mathbf{w}_j, \theta_j\} \sim \mathcal{N}(\mathbf{w}_j^\top \mathbf{z}_i, 1/\theta_j)$, $i = 1, \dots, N, j = 1, \dots, M$

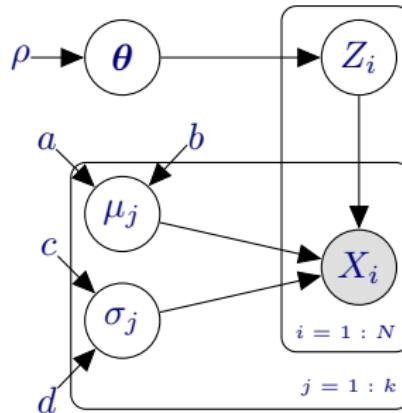
Local model



Latent variable models

Example: mixture of Gaussians

- Popular model for **clustering**
- Model formulation:
 - k Gaussians with frequencies $\boldsymbol{\theta} = (\theta_1, \dots, \theta_k)$
 - N observations generated by
 - ★ $Z_i \sim \text{Multinom}(\boldsymbol{\theta})$, $i = 1, \dots, N$
 - ★ $X_i|z_i \sim \mathcal{N}(\mu_i, \sigma_i)$, $i = 1, \dots, N$
 - ★ Bayesian setting: priors on the parameters



Inference in Bayesian networks

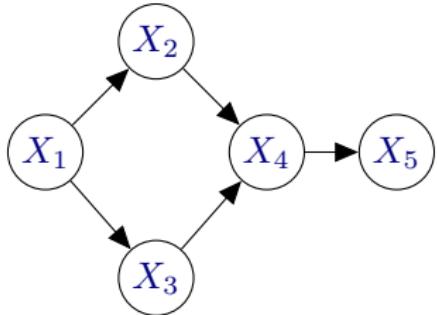
Assume a Bayesian network over variables $\mathbf{X} = \{X_1, \dots, X_n\}$

$$\left. \begin{array}{c} \text{Bayesian network} \\ + \\ \text{Evidence } (\mathbf{X}_E) \end{array} \right\} \Rightarrow P(\mathbf{X}_I | \mathbf{X}_E)?$$

Inference methods

- Exact
 - Brute force: compute $P(\mathbf{X}, \mathbf{X}_E)$ and marginalize out $\mathbf{X} \setminus \mathbf{X}_I$
 - Take advantage of the network structure
- Approximate
 - Sampling
 - Deterministic

Exact inference: Variable elimination



- We are interested in X_5
- All variables are discrete
- $E = \emptyset$

$$\begin{aligned} p(x_5) &= \sum_{x_1, \dots, x_4} p(x_1, x_2, x_3, x_4, x_5) \\ &= \sum_{x_1, \dots, x_4} p(x_1)p(x_2|x_1)p(x_3|x_1)p(x_4|x_2, x_3)p(x_5|x_4) \\ &= \sum_{x_2, \dots, x_4} \sum_{x_1} p(x_1)p(x_2|x_1)p(x_3|x_1)p(x_4|x_2, x_3)p(x_5|x_4) \\ &= \sum_{x_2, \dots, x_4} p(x_4|x_2, x_3)p(x_5|x_4) \left[\sum_{x_1} p(x_1)p(x_2|x_1)p(x_3|x_1) \right] \\ &= \sum_{x_2, \dots, x_4} p(x_4|x_2, x_3)p(x_5|x_4) h(x_2, x_3) \end{aligned}$$

We have reached a similar problem as initially, but with **one variable less**.

Variable elimination algorithm

Input

\mathcal{P} : Conditional distributions in the network

\mathbf{X} : Variables in the network

X_I : Target variable

\mathbf{X}_E : Observed variables

\mathbf{Y} : All variables in \mathbf{X} except X_I ($\mathbf{Y} = \mathbf{X} \setminus \{X_I\}$).

- ① Restrict all the distributions in \mathcal{P} to the evidence $\mathbf{X}_E = \mathbf{x}_E$
- ② For each $Y \in \mathbf{Y}$,
 - ① Let \mathcal{P}_Y be the set of distributions in \mathcal{P} that contain Y .
 - ② $q := \prod_{p \in \mathcal{P}_Y} p$.
 - ③ $r := \sum_y q$.
 - ④ $\mathcal{P} := \mathcal{P} \setminus \mathcal{P}_Y \cup \{r\}$.
- ③ $p(x_I) = \prod_{p \in \mathcal{P}} p$.
- ④ Normalize p

Variable elimination algorithm

Input

\mathcal{P} : Conditional distributions in the network

\mathbf{X} : Variables in the network

X_I : Target variable

\mathbf{X}_E : Observed variables

\mathbf{Y} : All variables in \mathbf{X} except X_I ($\mathbf{Y} = \mathbf{X} \setminus \{X_I\}$).

- ① Restrict all the distributions in \mathcal{P} to the evidence $\mathbf{X}_E = \mathbf{x}_E$
- ② For each $Y \in \mathbf{Y}$
 - ① Let \mathcal{P}_Y be the set of distributions in \mathcal{P} that contain Y
 - ② $q := \prod_{p \in \mathcal{P}_Y} p \implies \text{COMPLEXITY}$
 - ③ $r := \sum_y q$
 - ④ $\mathcal{P} := \mathcal{P} \setminus \mathcal{P}_Y \cup \{r\}$
- ③ $p(x_I) = \prod_{p \in \mathcal{P}} p$
- ④ Normalize p

Considerations about exact inference

- Product of functions raises complexity
 - Exponentially in the case of **discrete** variables
- Complexity also depends on the **elimination order**
- Representation of densities turns out to be relevant
 - **Closed-form** solutions to product and marginalization are preferable

Monte Carlo inference algorithms

- A Bayesian network is a representation of a joint probability distribution over \mathbf{X} \Rightarrow it describes some **population** consisting of all the possible configurations of \mathbf{X}
- If the entire population was available, the **inference problem** could be solved exactly, basically by **counting cases**
- **Problem:** Population size can be huge or even infinite.
- **Monte Carlo** methods operate by drawing an artificial **sample** from it using some random mechanism
- The sample (**much smaller than the population**), is used to estimate the distribution of each variable of interest.

Key issues in a Monte Carlo inference algorithm:

- ① The sampling mechanism
- ② The functions (estimators) which compute the probabilities from the sample

Monte Carlo inference algorithms

- A Bayesian network is a representation of a joint probability distribution over \mathbf{X} \Rightarrow it describes some **population** consisting of all the possible configurations of \mathbf{X}
- If the entire population was available, the **inference problem** could be solved exactly, basically by **counting cases**
- **Problem:** Population size can be huge or even infinite.
- **Monte Carlo** methods operate by drawing an artificial **sample** from it using some random mechanism
- The sample (**much smaller than the population**), is used to estimate the distribution of each variable of interest.

Key issues in a Monte Carlo inference algorithm:

- ① The sampling mechanism
- ② The functions (estimators) which compute the probabilities from the sample

Importance sampling. General setting

- Assume we have a random variable X with density $p(x)$
- Importance sampling is a technique designed for estimating the expected value of a function $f(X)$. It is based on the following transformation:

$$\mathbb{E}_p[f(x)] = \int f(x)p(x)dx = \int \frac{p(x)}{p^*(x)}f(x)p^*(x)dx = \mathbb{E}_{p^*}\left[\frac{p(x)}{p^*(x)}f(x)\right],$$

where p^* is a density function called the sampling or proposal distribution, s.t. $p^*(x) > 0$ whenever $p(x) > 0$.

- Therefore, $\mathbb{E}_p[f(x)]$ can be estimated by drawing a sample $x^{(1)}, \dots, x^{(m)}$ from p^* and computing

$$\hat{\mathbb{E}}_p[f(x)] = \frac{1}{m} \sum_{j=1}^m \frac{p(x^{(j)})}{p^*(x^{(j)})} f(x^{(j)}),$$

which is specially convenient if p^* is easier to handle than p

Importance sampling. General setting

- Assume we have a random variable X with density $p(x)$
- Importance sampling is a technique designed for estimating the expected value of a function $f(X)$. It is based on the following transformation:

$$\mathbb{E}_p[f(x)] = \int f(x)p(x)dx = \int \frac{p(x)}{p^*(x)}f(x)p^*(x)dx = \mathbb{E}_{p^*}\left[\frac{p(x)}{p^*(x)}f(x)\right],$$

where p^* is a density function called the sampling or proposal distribution, s.t. $p^*(x) > 0$ whenever $p(x) > 0$.

- Therefore, $\mathbb{E}_p[f(x)]$ can be estimated by drawing a sample $x^{(1)}, \dots, x^{(m)}$ from p^* and computing

$$\hat{\mathbb{E}}_p[f(x)] = \frac{1}{m} \sum_{j=1}^m \frac{p(x^{(j)})}{p^*(x^{(j)})} f(x^{(j)}),$$

which is specially convenient if p^* is easier to handle than p

Importance sampling in probabilistic inference

- Consider a rather general setting, with continuous target variable X_i , observed variables \mathbf{X}_E , unobserved discrete variables \mathbf{X}_D and unobserved continuous variables \mathbf{X}_C
- Assume we are interested in the probability of the variable taking values in a given interval (a, b) . This amounts to computing

$$p(X_i \in (a, b) | \mathbf{x}_E) = \frac{\int_a^b \sum_{\mathbf{x}_D \in \Omega_{\mathbf{X}_D}} \int_{\mathbf{x}_C \in \Omega_{\mathbf{X}_C}} p(x_i, \mathbf{x}_C, \mathbf{x}_D, \mathbf{x}_E) d\mathbf{x}_C dx_i}{\sum_{\mathbf{x}_{D_i} \in \Omega_{\mathbf{X}_{D_i}}} \int_{\mathbf{x}_{C_i} \in \Omega_{\mathbf{X}_{C_i}}} p(\mathbf{x}_{C_i}, \mathbf{x}_{D_i}, \mathbf{x}_E) d\mathbf{x}_{C_i}}$$

Importance sampling in probabilistic inference

Importance sampling can be applied to probabilistic inference in BNs taking into account that we can write $p(X_i \in (a, b) | \mathbf{x}_E)$ as

$$\begin{aligned} p(X_i \in (a, b) | \mathbf{x}_E) &= \mathbb{E}_{p(x_i | \mathbf{x}_E)} [\mathbf{1}_{(a,b)}(x_i)] \\ &= \mathbb{E}_{p^*} \left[\frac{p(x_i | \mathbf{x}_E)}{p^*(x_i)} \mathbf{1}_{(a,b)}(x_i) \right] \\ &= \frac{1}{p(\mathbf{x}_E)} \mathbb{E}_{p^*} \left[\frac{p(x_i, \mathbf{x}_E)}{p^*(x_i)} \mathbf{1}_{(a,b)}(x_i) \right], \end{aligned}$$

where $\mathbf{1}_{(a,b)}$ is the indicator function on interval (a, b) .

Importance sampling in probabilistic inference

Denoting $g(x_i) = p(x_i, \mathbf{x}_E)$,

$$\begin{aligned}\hat{p}(X_i \in (a, b) | \mathbf{x}_E) &= \frac{1}{p(\mathbf{x}_E)} \hat{\mathbb{E}}_{p^*} \left[\frac{g(x_i)}{p^*(x_i)} \mathbf{1}_{(a,b)}(x_i) \right] \\ &= \frac{1}{p(\mathbf{x}_E)} \frac{1}{m} \sum_{j=1}^m \frac{g(x_i^{(j)})}{p^*(x_i^{(j)})} \mathbf{1}_{(a,b)}(x_i^{(j)})\end{aligned}$$

The normalizing constant, $p(\mathbf{x}_E)$, can be estimated using **the same sample** $x_i^{(1)}, \dots, x_i^{(m)}$, just by summing all the *weights*:

$$\hat{p}(\mathbf{x}_E) = \sum_{j=1}^m \frac{g(x_i^{(j)})}{p^*(x_i^{(j)})}$$

Importance sampling in probabilistic inference

- The crucial step of Importance Sampling is the choice of the proposal distribution
- A careless choice can even lead to discarding most of the samples

Example

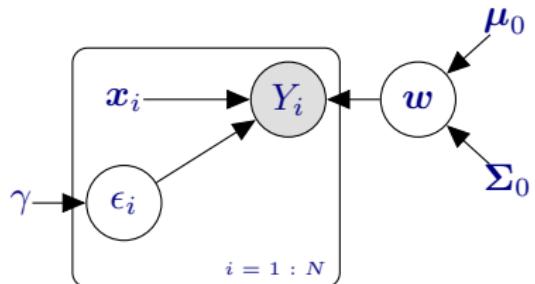
Consider binary variables X_1 and X_2 with $P(X_1 = 0) = 0.9$, $P(X_2 = 0|X_1 = 0) = 0$ and $P(X_2 = 0|X_1 = 1) = 0.5$. Assume X_2 is observed to be equal to 0. We have freedom to choose p^* .

- We choose p^* to be uniform $\implies > 50\%$ of the samples are discarded (they have weight equal to 0)
- We choose p^* to be $P(X_1)$ $\implies > 90\%$ of the samples are discarded

Importance sampling. Example

Linear regression

- $Y_i | \{w, x_i\} = w^\top x_i + \epsilon_i$ with $x_i = [1, x_i]^\top$
- $\epsilon_i \sim \mathcal{N}(0, 1/\gamma)$ with γ known
- $w \sim \mathcal{N}(\mu_0 = \mathbf{0}, \Sigma_0 = \mathbf{I}_{2 \times 2})$



- We have to compute

$$p(w|x) \sim \mathcal{N}(\mu, \Sigma)$$

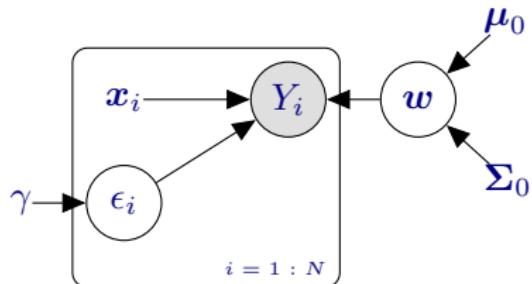
- $\mu = (\mathbb{E}[w_0], \mathbb{E}[w_1])$
- $\Sigma = \begin{pmatrix} \sigma_{w_0}^2 & \sigma_{w_0, w_1} \\ \sigma_{w_0, w_1} & \sigma_{w_1}^2 \end{pmatrix}$

- We have to sample ϵ_i and w from p^*
- We can assume, for instance p^* to be the product of three independent standard normals for ϵ_i , w_0 and w_1

Importance sampling. Example

Linear regression

- $Y_i | \{w, x_i\} = w^\top x_i + \epsilon_i$ with $x_i = [1, x_i]^\top$
- $\epsilon_i \sim \mathcal{N}(0, 1/\gamma)$ with γ known
- $w \sim \mathcal{N}(\mu_0 = \mathbf{0}, \Sigma_0 = \mathbf{I}_{2 \times 2})$



- We have to compute

$$p(w|x) \sim \mathcal{N}(\mu, \Sigma)$$

- $\mu = (\mathbb{E}[w_0], \mathbb{E}[w_1])$
- $\Sigma = \begin{pmatrix} \sigma_{w_0}^2 & \sigma_{w_0, w_1} \\ \sigma_{w_0, w_1} & \sigma_{w_1}^2 \end{pmatrix}$

- We have to sample ϵ_i and w from p^*
- We can assume, for instance p^* to be the product of three independent standard normals for ϵ_i , w_0 and w_1

Importance sampling. Example

Taking into account that

- $\sigma_{w_k}^2 = \mathbb{E}[w_k^2] - (\mathbb{E}[w_k])^2, k = 0, 1$
- $\sigma_{w_0, w_1} = \mathbb{E}[w_0 w_1] - \mathbb{E}[w_0] \mathbb{E}[w_1]$

it turns out that what we have to estimate to solve the linear regression problem is

- $\mathbb{E}[w_0], \mathbb{E}[w_1], \mathbb{E}[w_0^2], \mathbb{E}[w_1^2]$ and $\mathbb{E}[w_0 w_1]$
- **Note:** all expectations are taken w.r.t. $p(\mathbf{w}|\mathbf{x})$

Importance sampling. Example

- ① Generate a sample $\mathbf{z}_j = (\epsilon_i^{(j)}, w_0^{(j)}, w_1^{(j)}, y_i^{(j)}), j = 1, \dots, M$ from p^*
- ② Compute the estimations:

$$\hat{\mathbb{E}}[w_0] = \frac{1}{\hat{p}(\mathbf{x})} \frac{1}{M} \sum_{j=1}^M \frac{g(\mathbf{z}_j)}{p^*(\mathbf{z}_j)} w_0^{(j)}$$

$$\hat{\mathbb{E}}[w_1] = \frac{1}{\hat{p}(\mathbf{x})} \frac{1}{M} \sum_{j=1}^M \frac{g(\mathbf{z}_j)}{p^*(\mathbf{z}_j)} w_1^{(j)}$$

$$\hat{\mathbb{E}}[w_0^2] = \frac{1}{\hat{p}(\mathbf{x})} \frac{1}{M} \sum_{j=1}^M \frac{g(\mathbf{z}_j)}{p^*(\mathbf{z}_j)} w_0^{(j)2}$$

$$\hat{\mathbb{E}}[w_1^2] = \frac{1}{\hat{p}(\mathbf{x})} \frac{1}{M} \sum_{j=1}^M \frac{g(\mathbf{z}_j)}{p^*(\mathbf{z}_j)} w_1^{(j)2}$$

$$\hat{\mathbb{E}}[w_0 w_1] = \frac{1}{\hat{p}(\mathbf{x})} \frac{1}{M} \sum_{j=1}^M \frac{g(\mathbf{z}_j)}{p^*(\mathbf{z}_j)} w_0^{(j)} w_1^{(j)}$$

Markov Chain Monte Carlo (MCMC)

- We start with an initial configuration of the variables in the network
- A new item in the sample is generated **conditional on the previous configuration**
- The elements of the **sample** are **no longer independent**, instead they form a **Markov chain**

Gibbs sampling

A new item in the sample is generated by simulating each variable at a time conditional on the value of the other variables sampled so far:

- Assume $\mathbf{X} = \{X_1, \dots, X_n\}$
- Let $\mathbf{x}^{(i)} = (x_1^{(i)}, \dots, x_n^{(i)})$ be the current configuration
- A new configuration, $\mathbf{x}^{(i+1)}$ is generated by sampling each variable X_k , $k = 1, \dots, n$ with

$$p(x_j^{(i+1)} | x_1^{(i+1)}, \dots, x_{j-1}^{(i+1)}, x_{j+1}^{(i)}, \dots, x_n^{(i)})$$

Markov Chain Monte Carlo (MCMC)

- We start with an initial configuration of the variables in the network
- A new item in the sample is generated **conditional on the previous configuration**
- The elements of the **sample** are **no longer independent**, instead they form a **Markov chain**

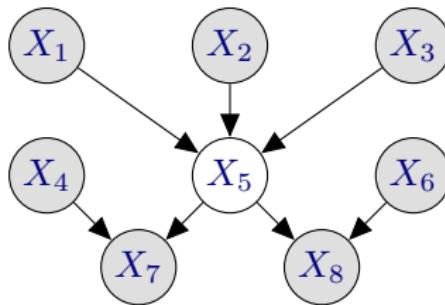
Gibbs sampling

A new item in the sample is generated by simulating each variable at a time conditional on the value of the other variables sampled so far:

- Assume $\mathbf{X} = \{X_1, \dots, X_n\}$
- Let $\mathbf{x}^{(i)} = (x_1^{(i)}, \dots, x_n^{(i)})$ be the current configuration
- A new configuration, $\mathbf{x}^{(i+1)}$ is generated by sampling each variable X_k , $k = 1, \dots, n$ with

$$p(x_j^{(i+1)} | x_1^{(i+1)}, \dots, x_{j-1}^{(i+1)}, x_{j+1}^{(i)}, \dots, x_n^{(i)})$$

Markov Chain Monte Carlo (MCMC)



$$p^*(X_5) \propto p(X_5|X_1, X_2, X_3)p(X_7|X_4, X_5)p(X_8|X_5, X_6)$$

Problems

- Generating an initial configuration with positive probability can be hard
- The dependence between elements of the sample can make it converge slowly

Markov Chain Monte Carlo (MCMC)

Example



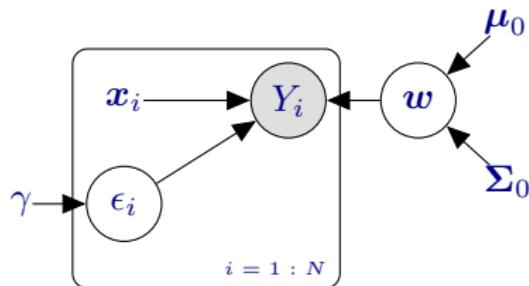
$$p(x_2|x_1) = p(\bar{x}_2|\bar{x}_1) = \delta \approx 1, \quad p(x_1) = 0.5$$

- Assume $X_1 = x_1 \Rightarrow p(x_2|w_{x_2}) = p(x_2|x_1) = \delta$
- If we draw $X_2 = x_2$, then X_1 is sampled from

$$\begin{aligned} p(x_1|w_{x_1}) &= p(x_1|x_2) = \frac{p(x_2|x_1)p(x_1)}{p(x_2|x_1)p(x_1) + p(x_2|\bar{x}_1)p(\bar{x}_1)} \\ &= \frac{\delta \times 0.5}{\delta \times 0.5 + (1 - \delta) \times 0.5} = \delta \end{aligned}$$

- Hence, we obtain $(X_1 = x_1, X_2 = x_2)$ with probability ≈ 1 , and, if a variable ever changed its value, the other one would change as well, thus appearing many times the configuration $(X_1 = \bar{x}_1, X_2 = \bar{x}_2)$.

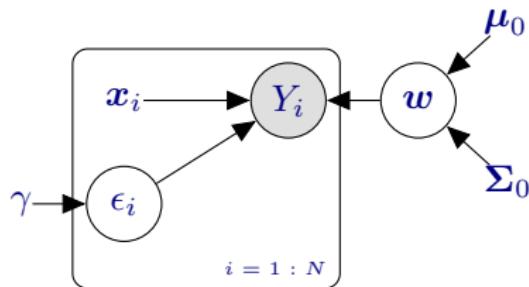
Gibbs sampling example: Linear regression



- ➊ Start with an initial random configuration
- ➋ Simulate a sample by changing coordinates one at a time
- ➌ Estimate $\mathbb{E}[w_0], \mathbb{E}[w_1], \mathbb{E}[w_0^2], \mathbb{E}[w_1^2]$ and $\mathbb{E}[w_0 w_1]$ using sample mean estimators

You still need to be able to compute the distribution of each variable given its Markov blanket

Gibbs sampling example: Linear regression



- ➊ Start with an initial random configuration
- ➋ Simulate a sample by changing coordinates one at a time
- ➌ Estimate $\mathbb{E}[w_0], \mathbb{E}[w_1], \mathbb{E}[w_0^2], \mathbb{E}[w_1^2]$ and $\mathbb{E}[w_0 w_1]$ using sample mean estimators

You still need to be able to compute the distribution of each variable given its Markov blanket

Conclusions

- PGMs provide a well founded way of handling uncertainty
- From a Bayesian point of view, inference and learning are connected tasks
- If scalability is on the focus, approximate inference is needed
- Interpretability is a key issue

THANK YOU!!!



antonio.salmeron@ual.es



@ascerdan



<http://www.ual.es/personal/asalmero>