The probabilistic modelling cycle - II

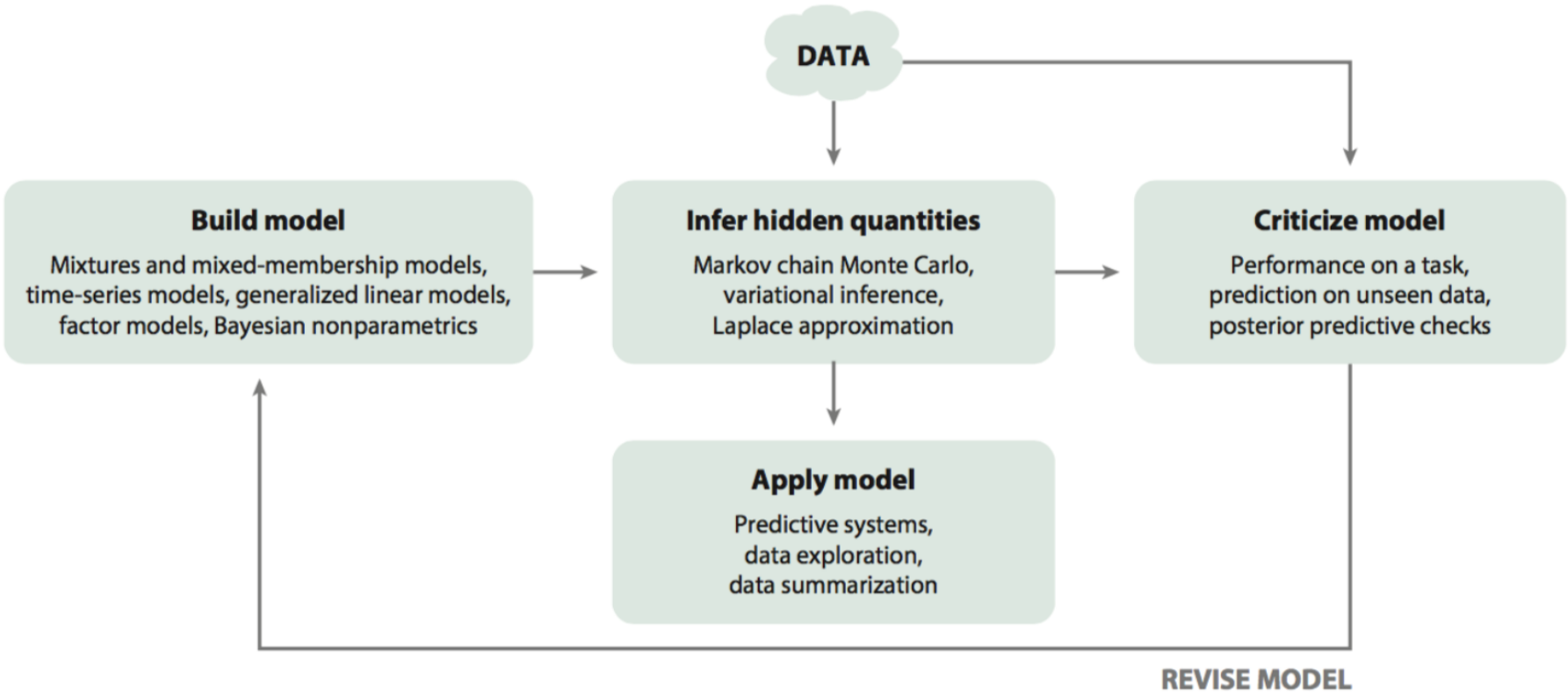


Image from: David M. Blei (2014) "Build, compute, critique, repeat: Data analysis with latent variable models." *Annual Review of Statistics and its Applications 1, 303–323.*

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, improves with experience E". (Tom Mitchell, 1997)*

Easy example: linear regression

- We have data about two variables X and Y "experience"
- 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 $\mathbf{y} = \{y_1, \ldots, y_n\}$ denoting the data and $\hat{\mathbf{y}} = \{\hat{y}_1, \ldots, \hat{y}_n\}$ denoting the model estimates

Easy example: linear regression

- We have data about two variables X and Y "experience"
- 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 $\mathbf{y} = \{y_1, \ldots, y_n\}$ denoting the data and $\hat{\mathbf{y}} = \{\hat{y}_1, \ldots, \hat{y}_n\}$ denoting the model estimates

Is this really ML???

Easy example: linear regression

- The linear regression model we have used IS NOT a probabilistic model
- We'll see later how it can be approached from a probabilistic point of view

Learning probabilistic models from data

Model (simple):

- a theoretical probability density/mass function f
  - associated with random variable X
  - having parameter θ

Learning problem:

- We assume f is known except for parameter θ
- This is denoted as $f(x; \theta)$ or $f(x \mid \theta)$
- Goal: estimate θ

Tools:

- for a sample $X_1, \ldots, X_n$ drawn from $f(x \mid \theta)$, the likelihood function is:

$$l(\theta \mid x_1, \ldots, x_n) \stackrel{\text{def}}{=} f(x_1, \ldots, x_n \mid \theta) = \prod_{i=1}^{n} f(x_i \mid \theta)$$

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

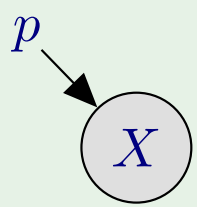
Learning parameters from data: frequentist approach

- POV: parameter θ has a fixed but unknown value

Consider tossing a (fair?) coin

Goal: estimate $p(heads)$

Frequentist POV:
probability = relative frequency
"in the long run"



After 50 trials:

$HHHTTH \ldots\ldots TH$

$\hat{p} = \frac{15}{50}$

What is underlying theoretical model $f(x \mid p)$?

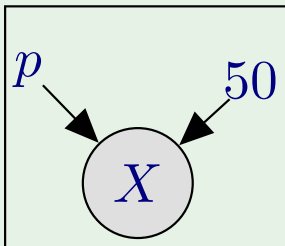
Learning parameters from data: frequentist approach

- POV: parameter θ has a fixed but unknown value

Consider tossing a (fair?) coin

Goal: estimate $p(heads)$

Frequentist POV:
probability = relative frequency
"in the long run"



After 50 trials:

$HHHTTH \ldots\ldots TH$

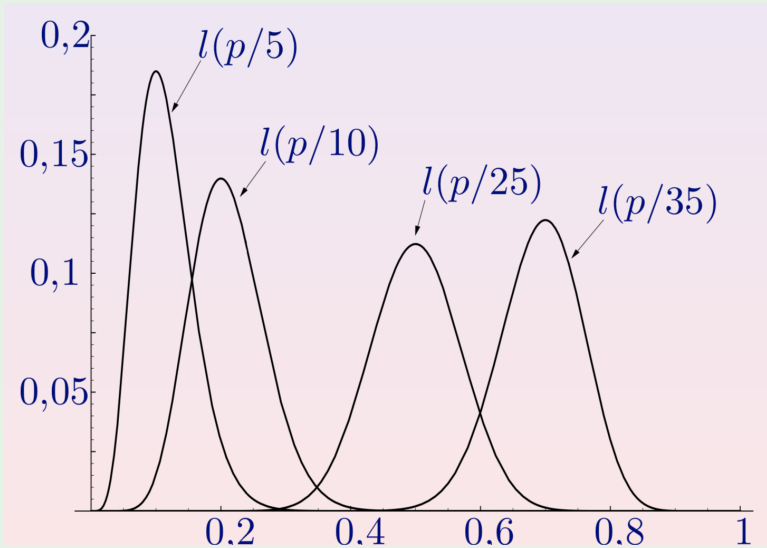
$\hat{p} = \frac{15}{50}$

What is underlying theoretical model $f(x \mid p)$?

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

The likelihood function is

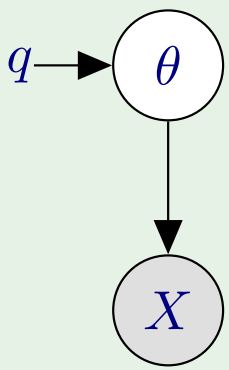
$$l(p \mid x) = \binom{50}{x} p^x (1-p)^{50-x}$$



Learning parameters from data: Bayesian approach

- POV: parameters are modelled as random variables $\rightarrow$ information about them can be included prior to observing data
- Additional tools: using Bayes' rule, the prior information is combined with the likelihood, yielding a posterior distribution
- The posterior then becomes the new prior
- As such, inferences about the parameter allow for its updating

Bayesian networks for Bayesian learning



- Random variables (and parameters) inside circles
- Grey if observable; white if hidden
- Fixed quantities without circle

Learning from data: Bayesian approach

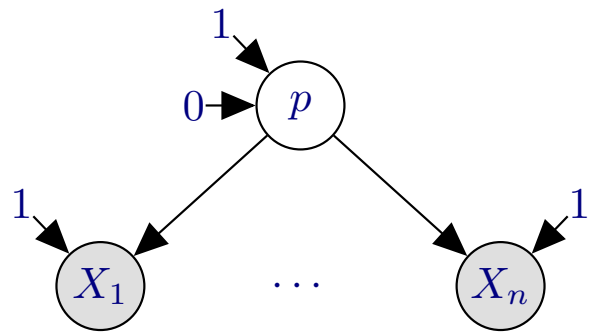
Distributions in a Bayesian model - I

For learning:

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

Learning from data: Example of Bayesian approach

- Assume a sample $X_1, X_2, \ldots, X_n \sim \mathcal{B}(1, p)$ and $p \sim \mathcal{U}(0, 1)$



- Then the likelihood and the prior are,

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

$$\pi(p) = \frac{1}{1-0} = 1, \text{ if } p \in (0, 1)$$

Learning from data: Example of Bayesian approach

Assume a sample $X_1, X_2, \ldots, X_n \sim \mathcal{B}(1, p)$ and $p \sim \mathcal{U}(0, 1)$

- Recall that the likelihood and the prior are:

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

- The posterior distribution is

$$\pi(p|x_1, \ldots, x_n) = \frac{f(x_1, \ldots, x_n|p)\pi(p)}{\int_0^1 f(x_1, \ldots, x_n|p)\pi(p)\, dp} = \frac{p^{\sum x_i}(1-p)^{n-\sum x_i}}{\int_0^1 p^{\sum x_i}(1-p)^{n-\sum x_i}\, dp}$$

Learning from data: Example of Bayesian approach

Pattern matching: the Beta distribution $Be(\alpha, \beta)$

$$f(p) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}; \qquad \int_0^1 f(p)\, dp = 1$$

$$\int_0^1 p^{\sum x_i}(1-p)^{n-\sum x_i}\, dp =$$

$$= \int_0^1 \frac{\Gamma(\sum x_i+1)\Gamma(n-\sum x_i+1)}{\Gamma(n+2)} \frac{\Gamma(n+2)}{\Gamma(\sum x_i+1)\Gamma(n-\sum x_i+1)} p^{\sum x_i}(1-p)^{n-\sum x_i}\, dp$$

$$= \frac{\Gamma(\sum x_i+1)\Gamma(n-\sum x_i+1)}{\Gamma(n+2)} \int_0^1 \frac{\Gamma(n+2)}{\Gamma(\sum x_i+1)\Gamma(n-\sum x_i+1)} p^{\sum x_i}(1-p)^{n-\sum x_i}\, dp$$

$$= \frac{\Gamma(\sum x_i+1)\Gamma(n-\sum x_i+1)}{\Gamma(n+2)} \cdot 1$$

Learning from data: Example of Bayesian approach

Assume a sample $X_1, X_2, \ldots, X_n \sim \mathcal{B}(1,p)$ and $p \sim \mathcal{U}(0,1) = Be(1,1)$

- Then the likelihood and the prior are,

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

- The posterior distribution is

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

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

Learning from data: Example of Bayesian approach

Assume a sample $X_1, X_2, \ldots, X_n \sim \mathcal{B}(1,p)$ and $p \sim \mathcal{U}(0,1) = Be(1,1)$

- Then the likelihood and the prior are,

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

- The posterior distribution is

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

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

Very easy to compute for some models

Conjugate priors and likelihoods

Prior and likelihood are called conjugate, if prior and posterior are from same family.

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

Learning from data: Bayesian approach

Distributions in a Bayesian model - II

For validation and use:

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

- The (posterior) predictive distribution given $\boldsymbol{x} = \{x_1, \ldots, x_n\}$:

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

Example Bayesian approach, continued

- The prior predictive distribution is

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

- The (posterior) predictive distribution is

$$f(x|x_1, \ldots, x_n) =$$

$$= \int_0^1 p^x (1-p)^{1-x} \frac{\Gamma(n+2)}{\Gamma(\sum x_i + 1)\Gamma(n - \sum x_i + 1)} p^{\sum x_i} (1-p)^{n - \sum x_i} dp$$

$$= \frac{\Gamma(n+2)}{\Gamma(\sum x_i + 1)\Gamma(n - \sum x_i + 1)} \int_0^1 p^{x + \sum x_i} (1-p)^{n+1-(x+\sum x_i)} dp$$

$$= \frac{\Gamma(n+2)}{\Gamma(\sum x_i + 1)\Gamma(n - \sum x_i + 1)} \frac{\Gamma(x + 1 + \sum x_i)\Gamma(n + 2 - (x + \sum x_i))}{\Gamma(n+3)}$$

Learning 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

$$\begin{aligned}\hat{\theta} &= \arg\max_{\theta} f(x_1, \ldots, x_n, \theta) \\ &= \arg\max_{\theta} f(x_1, \ldots, x_n|\theta)\pi(\theta) \\ &= \arg\max_{\theta} \{\log f(x_1, \ldots, x_n|\theta) + \log \pi(\theta)\}\end{aligned}$$

  known as the MAP (Maximum A Posteriori) estimator

- Note that we could also choose

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

  which is actually the MLE (Maximum Likelihood Estimator)

Some simple examples

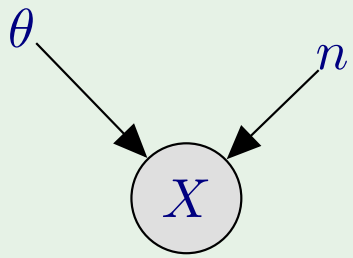
Tossing a coin

- X: result of n coin tosses with some $p(heads)$
- random variables?
- fixed quantities?
- hidden variables?
- coin is possibly biased towards tails

Some simple examples

Tossing a coin

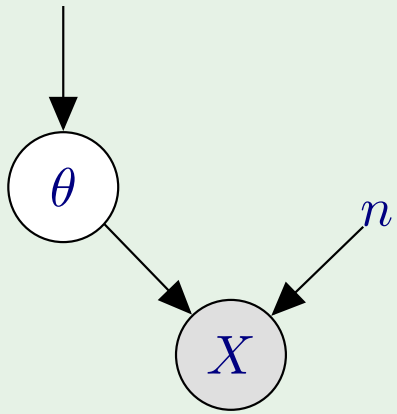
- X: result of n coin tosses with some $p(heads)$
- random variables?
- fixed quantities?
- hidden variables?
- coin is possibly biased towards tails



Some simple examples

Tossing a coin

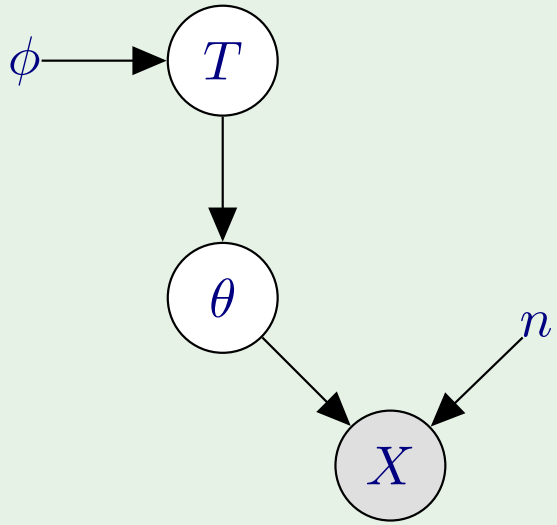
- X: result of n coin tosses with some $p(heads)$
- random variables?
- fixed quantities?
- hidden variables?
- coin is possibly biased towards tails



Some simple examples

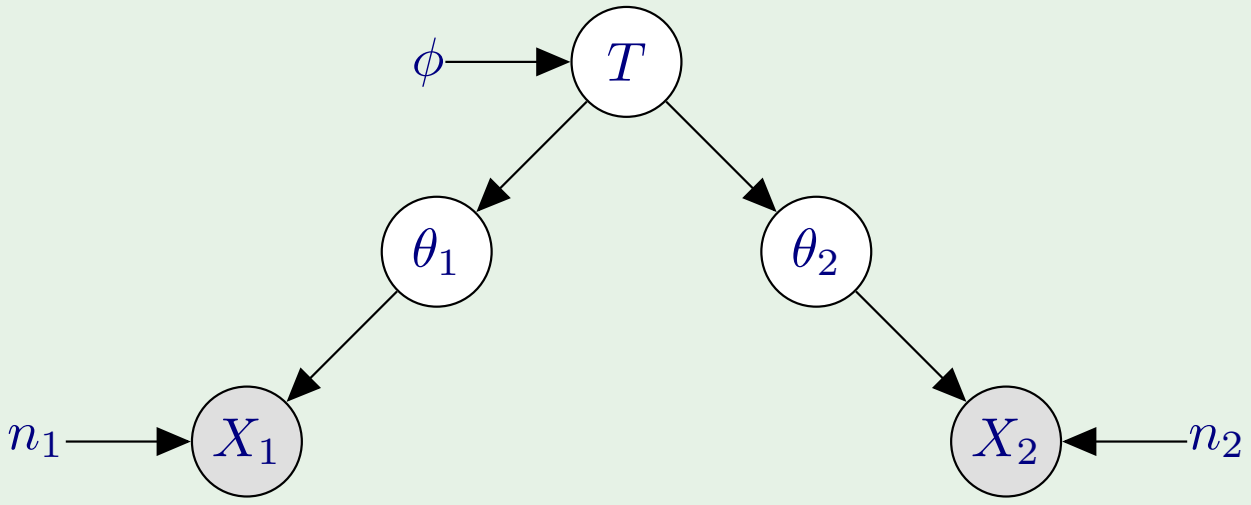
Tossing a biased(?) coin

- X: result of n coin tosses with some $p(heads)$
- random variables?
- fixed quantities?
- hidden variables?
- coin is possibly biased towards tails



Some simple examples

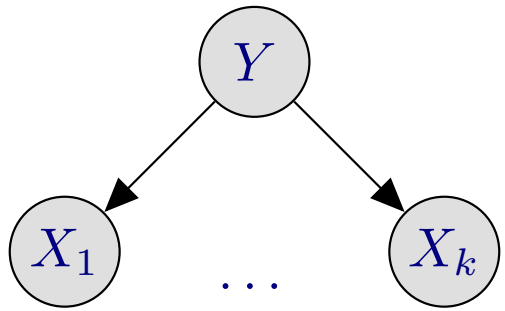
Two coins



Some simple examples

Naive Bayes

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



$$p(y \mid x_1, \ldots, x_k) \overset{\text{Bayes}}{=} \frac{p(x_1, \ldots, x_k \mid y)p(y)}{p(x_1, \ldots, x_k)}$$

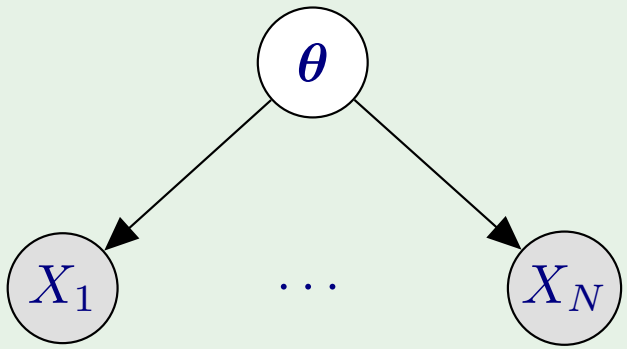
$$\propto \quad p(x_1, \ldots, x_k \mid y)p(y) = p(y) \prod_{i=1}^{k} p(x_i \mid y)$$

Plate notation

The idea is to avoid repeated substructures

Example: independent data points

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



Unfolded notation

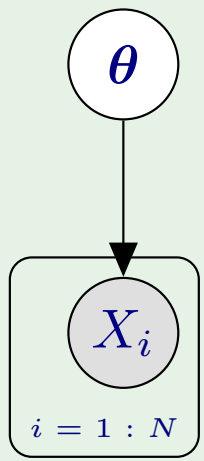


Plate notation

Plate notation: linear regression revisited

Example: linear regression (fully probabilistic)

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

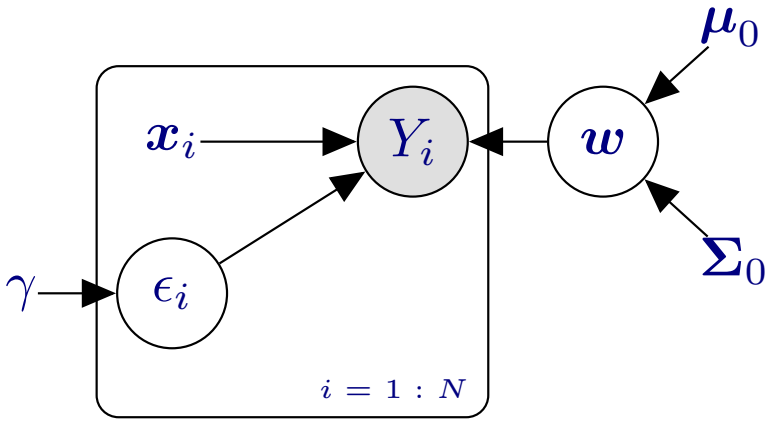
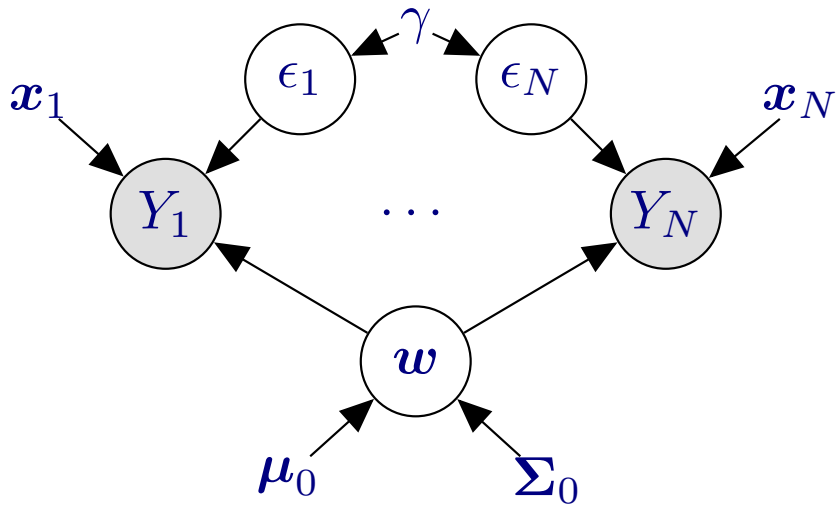


Plate notation



Unfolded notation

- Underlying model: $\boxed{y_i = w_0 + w_1 x_i + \epsilon_i}$

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

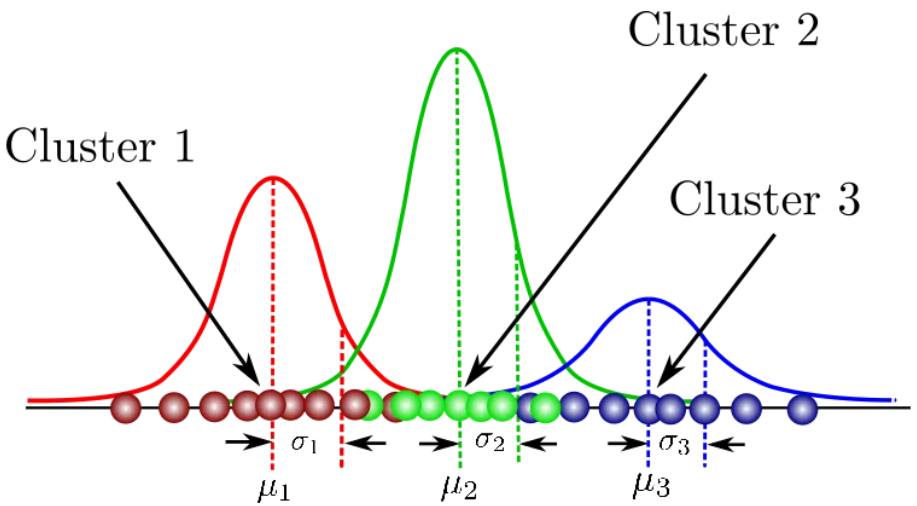
Latent variable models

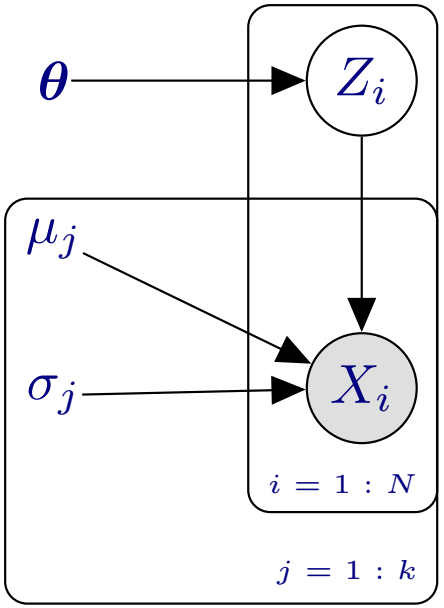
In general, latent variable models are regarded as probabilistic models where some of the variables cannot be observed.

Example: mixture of Gaussians; popular model for clustering

Model formulation:

- k Gaussians with frequencies $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_k)$ (sum to 1)
- N observations generated by
  - $Z_i \sim \text{Multinom}(\boldsymbol{\theta})$, $i = 1, \ldots, N$ (indicates which Gaussian/cluster)
  - $X_i | z_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$, $i = 1, \ldots, N$





https://towardsdatascience.com/gaussian-mixture-models-explained-6986aaf5a95

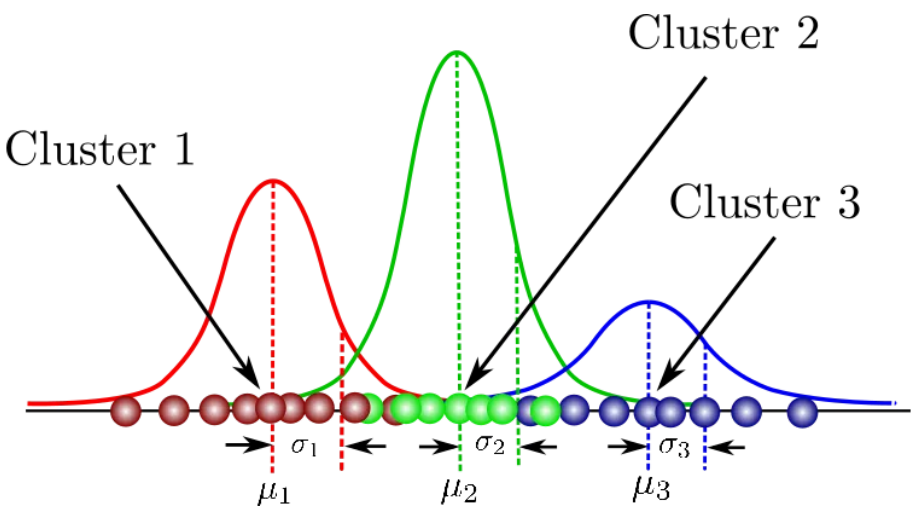
Latent variable models

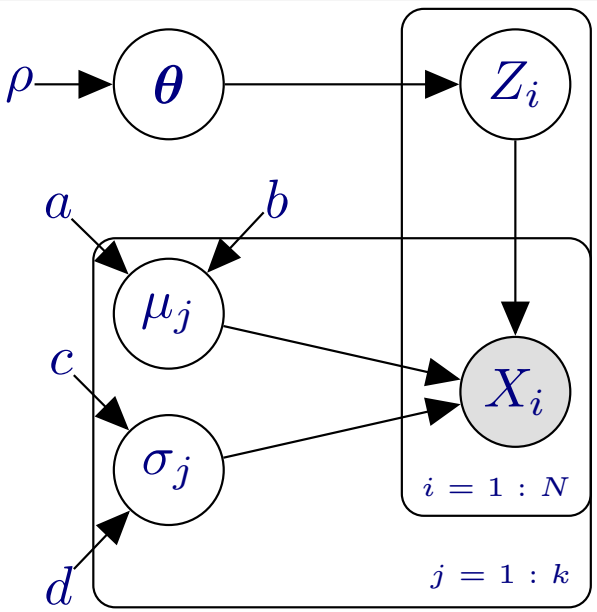
In general, latent variable models are regarded as probabilistic models where some of the variables cannot be observed.

Example: mixture of Gaussians; popular model for clustering

Model formulation:

- k Gaussians with frequencies $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_k)$ (sum to 1)
- N observations generated by
  - $Z_i \sim \text{Multinom}(\boldsymbol{\theta})$, $i = 1, \ldots, N$ (indicates which Gaussian/cluster)
  - $X_i | z_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$, $i = 1, \ldots, N$
  - Bayesian setting: priors on the parameters



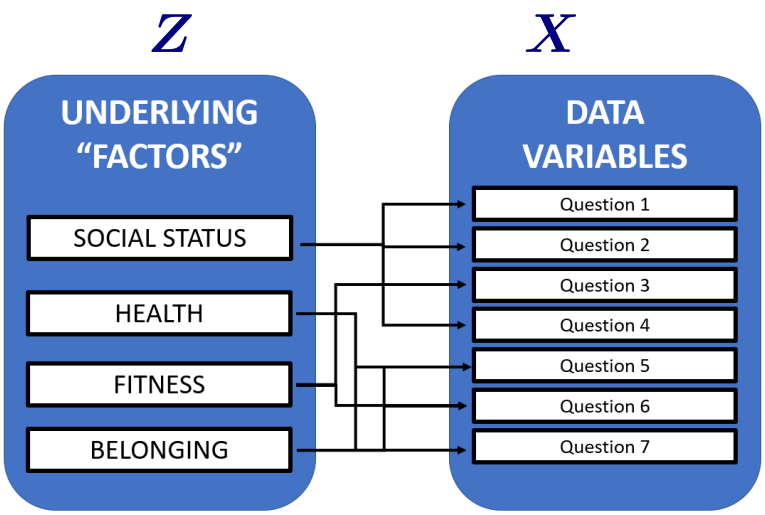


https://towardsdatascience.com/gaussian-mixture-models-explained-6986aaf5a95

Latent variable models

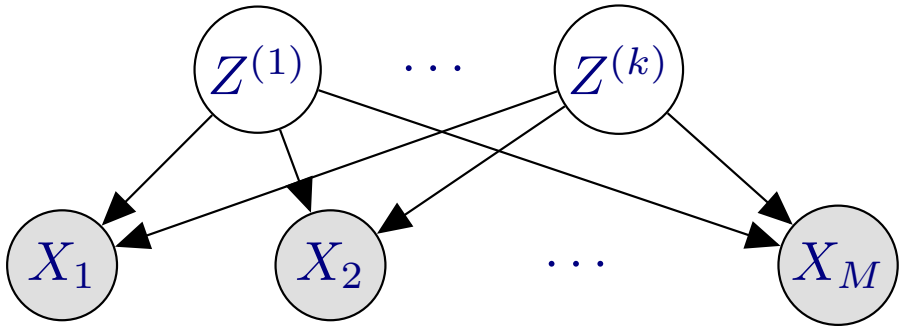
Example: factor analysis (FA) model

FA summarizes a high-dimensional observation $\boldsymbol{X}$ of M correlated variables by a smaller set of factors $\boldsymbol{Z}$ assumed independent a priori.



https://golden.com/wiki/Factor_analysis-RWGG9

Local model for single observation $\boldsymbol{X}$



$$x_1 = w_{1,1} \cdot z^{(1)} + \ldots + w_{1,k} \cdot z^{(k)} + \theta_1$$

$$\vdots$$

$$x_j = w_{j,1} \cdot z^{(1)} + \ldots + w_{j,k} \cdot z^{(k)} + \theta_j$$

$$\vdots$$

$$x_M = w_{M,1} \cdot z^{(1)} + \ldots + w_{M,k} \cdot z^{(k)} + \theta_M$$

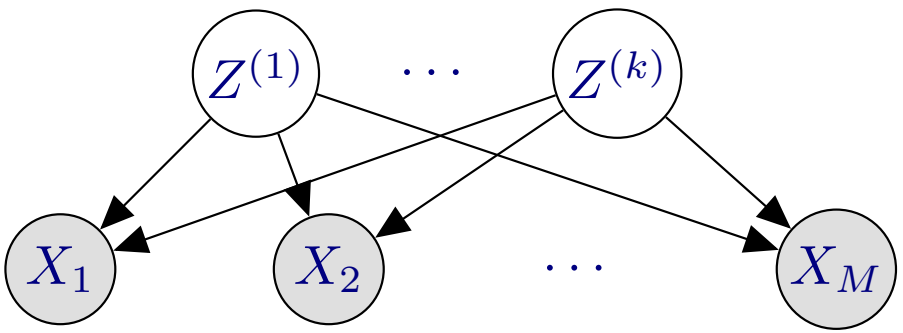
Latent variable models

Example: factor analysis (FA) model

FA summarizes a high-dimensional observation $\boldsymbol{X}$ of M correlated variables by a smaller set of factors $\boldsymbol{Z}$ assumed independent a priori.

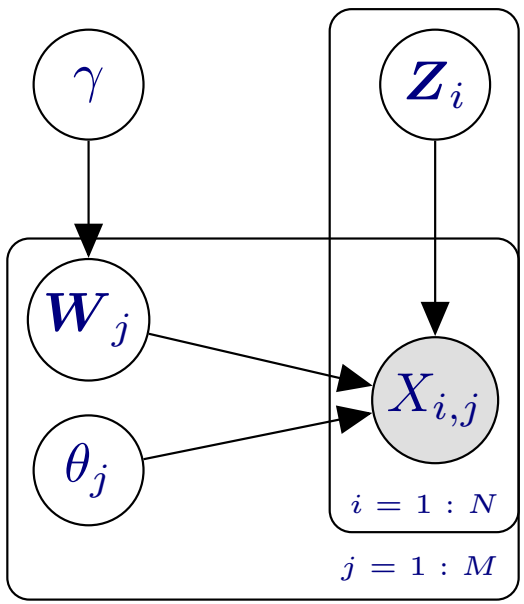
- Model formulation:
  - N observations $\boldsymbol{X}_i$ generated by:
    - ⋆ $\boldsymbol{Z}_i \sim \mathcal{N}(\boldsymbol{\mu}_0 = \mathbf{0}, \boldsymbol{\Sigma}_0 = \mathbf{I})$, $i = 1, \ldots, N$
    - ⋆ $X_{i,j} | \{\boldsymbol{z}_i, \boldsymbol{w}_j, \theta_j\} \sim \mathcal{N}(\boldsymbol{w}_j^\intercal \boldsymbol{z}_i, 1/\theta_j)$, $i = 1, \ldots, N$, $j = 1, \ldots, M$
    - ⋆ $\boldsymbol{W}_j \sim \mathcal{N}(\mathbf{0}, 1/\gamma)$

Local model for single observation $\boldsymbol{X}_i$



$$x_{i,1} = w_{1,1} \cdot z_i{}^{(1)} + \ldots + w_{1,k} \cdot z_i{}^{(k)} + \theta_1$$

etc.



Exact inference

We've already seen Variable Elimination as an example:

$$p(x_5) \quad = \quad \sum_{x_2,\ldots,x_4} p(x_4|x_2,x_3)p(x_5|x_4)h(x_2,x_3)$$

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

Approximate inference

- sampling: Monte Carlo techniques, e.g. importance sampling, MCMC
  - accurate with enough samples
  - sampling can be computationally demanding

- deterministic, e.g. variational approaches
  - uses analytical approximations to the posterior
  - some techniques scale well

Monte Carlo inference algorithms

- A Bayesian network is a representation of a joint probability distribution over $\boldsymbol{X} \Rightarrow$ it describes some population consisting of all the possible configurations of $\boldsymbol{X}$
- 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 the population using some random mechanism
- The sample (much smaller than the population), is used to estimate the distribution of each variable of interest.

Monte Carlo inference algorithms

- A Bayesian network is a representation of a joint probability distribution over $\boldsymbol{X} \Rightarrow$ it describes some population consisting of all the possible configurations of $\boldsymbol{X}$
- 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 the population 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 $x^{(1)}, \ldots, x^{(m)}$ 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 $x^{(1)}, \ldots, x^{(m)}$ from p^* and computing

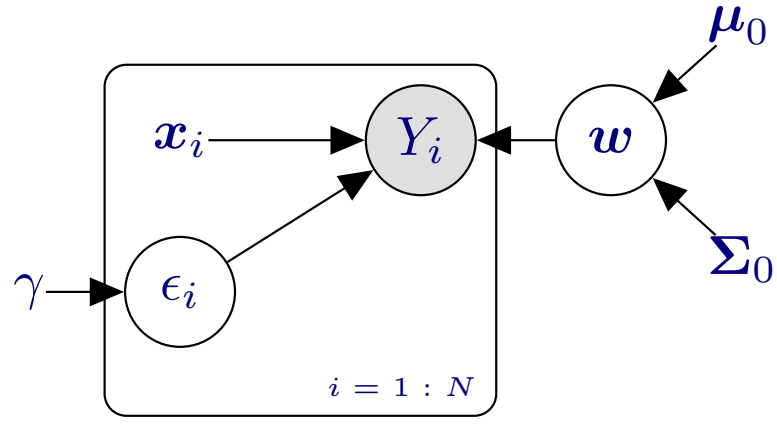
$$\hat{\mathbb{E}}_p[f(x)] = \frac{1}{m} \sum_{j=1}^{m} \frac{p(x^{(j)})}{p^*(x^{(j)})} f(x^{(j)}),$$

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

Importance sampling. Example

Linear regression

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



- We have to compute

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

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

- We have to sample ϵ_i and $\boldsymbol{w}$ 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

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

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. $p(\boldsymbol{w}|\boldsymbol{x})$

Importance sampling. Example

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

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

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

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

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

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

conclude

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 important, approximate inference is needed
- Interpretability is a key issue

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 important, approximate inference is needed
- Interpretability is a key issue