# Generative Modelling

Christos Dimitrakakis

November 11, 2025

# Outline


Graphical models
- Random variables and probabilities
- Graphical model
- Exercises

Classification
- Classification: Generative modelling
- Density estimation

Algorithms for latent variable models
- Gradient algorithms
- Expectation maximisation

Exercises
- Density estimation
- Classification

Graphical models
- Random variables and probabilities
- Graphical model
- Exercises

Classification
- Classification: Generative modelling
- Density estimation

Algorithms for latent variable models
- Gradient algorithms
- Expectation maximisation

Exercises
- Density estimation
- Classification

# Notational problems

In probability theory, we are dealing with functions $P$ on sets, called measures. In statistics, we are dealing with random variables. While these two can be related, it is common to use shorthand notation in statistics. If in doubt, use the following conversion table.

## Probability theory

1. Sets $A \subset \Omega$ are events.
2. Probability measure $P$ on $\Omega$.
3. $P(A) \in [0, 1]$ with $A \subset \Omega$
4. $P(A \cap C)$ the probability $\omega \in A \cap C$
5. $P(A|B)$ the probability of $A$ if $B$ is true.
6. Marginal distribution $P(A) = \sum_{i=1}^{n} \mathbb{P}(A \cap H_i)$
7. Bayes's theorem: $P(A \mid B) = P(A \cap B)/P(B)$.

## Statistics

1. Random variables $x : \Omega \to X$.
2. The measure $P$ is implicit.
3. $\mathbb{P}(x)$ for the distribution of $x$
4. $\mathbb{P}(x, y)$ the joint distribution of $x, y$.
5. $\mathbb{P}(x|y)$ the distribution of $x$ if $y$ is given.
6. Marginal distribution $\mathbb{P}(y) = \sum_{i \in x} \mathbb{P}(x = i, y)$.
7. Bayes's theorem: $\mathbb{P}(x|y) = \mathbb{P}(x, y)/\mathbb{P}(y)$

# Probability and statistics

## Probability theory

Here, we talk a lot about probability measures $P$ on some space $\Omega$. These are functions on sets, so that

- $P(\Omega) = 1$
- $P(A) \in [0, 1]$ for any $A \subset \Omega$
- $P(A \cup B) = P(A) + P(B)$ if $A \cap B = \emptyset$.

# Statistics

In statistics, we are interested in random variables $x, y, \ldots$ which are functions $x : \Omega \to X$, t$y : \Omega \to Y$ etc.

- We write $\mathbb{P}(x)$ as a shorthand for *the probability distribution of* $x$.
- The distribution of $x$ arises in the following way:
  1. Sample $\omega$ from $P$
  2. Calculate $x(\omega)$.
- We can get a probability measure $P_x$ by measuring how often $x$ falls in various sets $A \subset X$:

$$P_x(A) \triangleq P(\{\omega : x(\omega) \in A\})$$

Here, $\{\omega : x(\omega) \in A\}$ is literally the set of random outcomes $\omega$ for which $x(\omega)$ is in the set $A$. To avoid writing this cumbersome expression, we simply write $\mathbb{P}(x)$ as a general shorthand. This allows us to also write

$$\mathbb{P}(x \in A) = P_x(A).$$

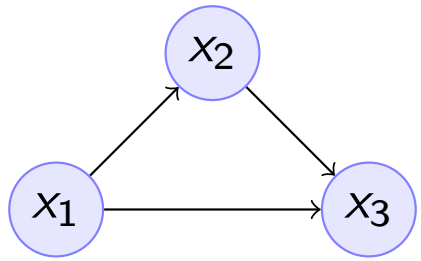
# Joint distributions, multiple variables

If you thought things were messy, wait to see what happens when you have more than one variable!

- $x : \omega \to X$, $y : \omega \to Y$.
- The measure $P_{x,y}$ has to be defined on the subsets of $X \times Y$.

$$P_{x,y}(A) = P(\{\omega : (x(\omega), y(\omega)) \in A\}$$

# Graphical models




## Directed acyclic graph

A directed acyclic graph with

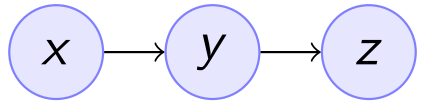
- Nodes $x_1, \ldots, x_n$
- Edges $x_i \to x_j$.

so that there are no cycles in the graph.

## Conditional independence from graphical models

- Each node of the model corresponds to a random variable
- The parent of a node are the direct dependencies of the random variable.

# Model specification




- The graphical model tells us what depends on what.
- It does not tell us how to generate data.
- We need to specify the functional relationship between variables.

## Statistics

$$x \sim f$$
$$y \mid x = a \sim g(a)$$
$$z \mid y = b \sim h(b)$$

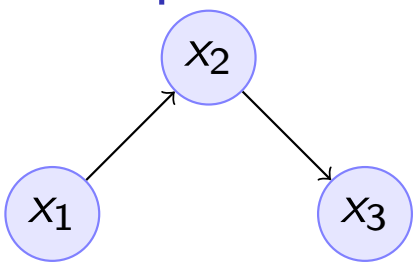
## Python

```
x = f()
y = g(x)
z = h(y)
```

# Graphical models and factorisation

- Graphical models tell us what directly depends on what
- They allow us to simplify the joint distribution of the variables
- This is formed into a product of factors, one for each variable.

## Example




This graphical model implies the factorisation

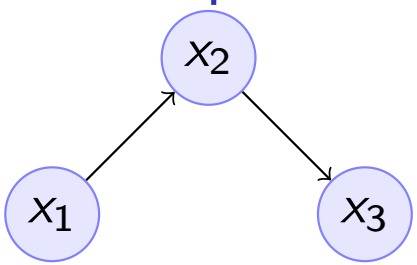
$$\mathbb{P}(x_1, x_2, x_3) = \mathbb{P}(x_3 \mid x_2, x_1)\,\mathbb{P}(x_2 \mid x_1)\,\mathbb{P}(x_1) = \mathbb{P}(x_3 \mid x_2)\,\mathbb{P}(x_2 \mid x_1)\,\mathbb{P}(x_1)$$

Notice that for each factor $\mathbb{P}(x_i \mid x_j)$, $x_j$ is the parent of $x_i$.

# Conditional independence

- Graphical models tell us what directly depends on what
- Consequently, they also specify conditional independence

## Example




$$\mathbb{P}(x_1, x_2, x_3) = \mathbb{P}(x_3 \mid x_2)\,\mathbb{P}(x_2 \mid x_1)\,\mathbb{P}(x_1)$$

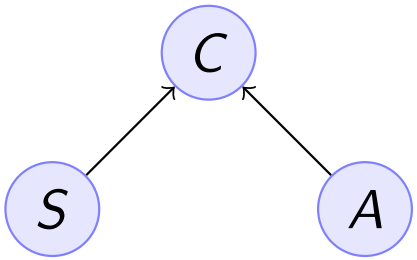
## Definition (Conditional independence)

We say that variables $x, y$ are conditionally independent given $z$ and write $x \perp\!\!\!\perp y \mid z$ if and only if

$$\mathbb{P}(x, y \mid z) = \mathbb{P}(x \mid z)\,\mathbb{P}(y \mid z)$$

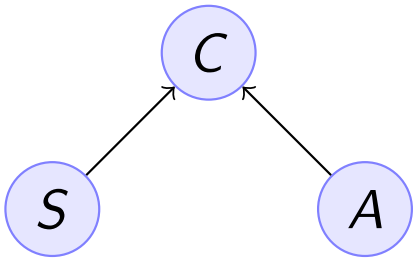
- In the above example, it holds that $x_3 \perp\!\!\!\perp x_1 \mid x_2$.

# Smoking and lung cancer



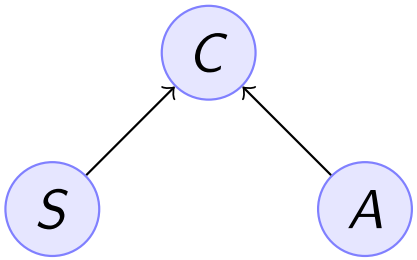

- Smoking and lung cancer graphical model.
- $S$: Smoking, $C$: cancer, $A$: asbestos exposure.

# Smoking and lung cancer




- Smoking and lung cancer graphical model.
- $S$: Smoking, $C$: cancer, $A$: asbestos exposure.
- In this graph, $A \perp\!\!\!\perp S$, but $A \not\perp\!\!\!\perp S \mid C$

# Smoking and lung cancer

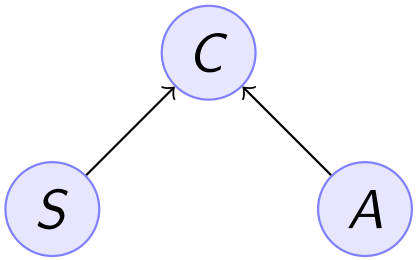



- Smoking and lung cancer graphical model.
- $S$: Smoking, $C$: cancer, $A$: asbestos exposure.
- In this graph, $A \perp\!\!\!\perp S$, but $A \not\perp\!\!\!\perp S \mid C$

## XOR example

- C = xor(S, A).

# Smoking and lung cancer

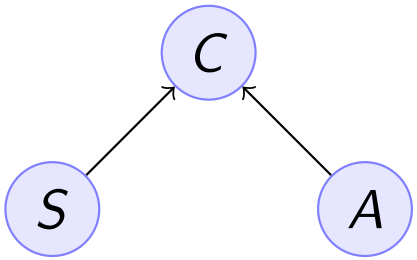



- Smoking and lung cancer graphical model.
- $S$: Smoking, $C$: cancer, $A$: asbestos exposure.
- In this graph, $A \perp\!\!\!\perp S$, but $A \not\perp\!\!\!\perp S \mid C$

## XOR example

- C = xor(S, A).
- If we know C = 1, S = 1, what is $A$?

# Smoking and lung cancer

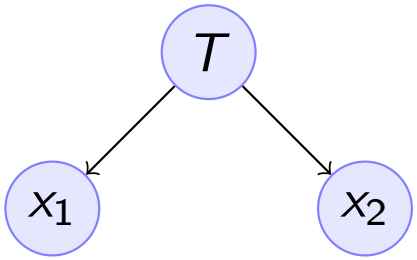



- Smoking and lung cancer graphical model.
- $S$: Smoking, $C$: cancer, $A$: asbestos exposure.
- In this graph, $A \perp\!\!\!\perp S$, but $A \not\perp\!\!\!\perp S \mid C$

## XOR example

- C = xor(S, A).
- If we know C = 1, S = 1, what is $A$?
- C explains away

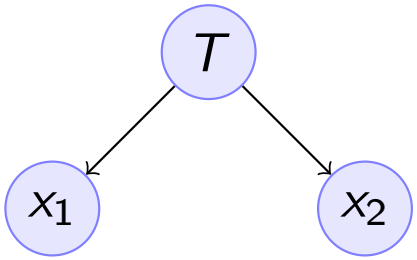
# Time of arrival at work




Time of arrival at work graphical model where $T$ is a traffic jam and $x_1$ is the time John arrives at the office and $x_2$ is the time Jane arrives at the office.

- Even though $x_1, x_2$ are not independent, they become independent once you know $T$, i.e. $x_1 \perp\!\!\!\perp x_2 \mid T$.

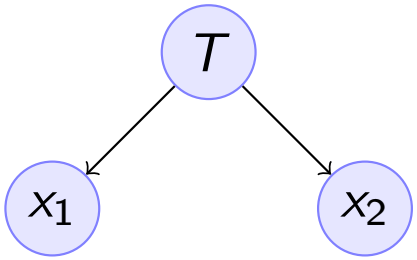
# Time of arrival at work




Time of arrival at work graphical model where $T$ is a traffic jam and $x_1$ is the time John arrives at the office and $x_2$ is the time Jane arrives at the office.

- Even though $x_1, x_2$ are not independent, they become independent once you know $T$, i.e. $x_1 \perp\!\!\!\perp x_2 \mid T$.
- Proof:

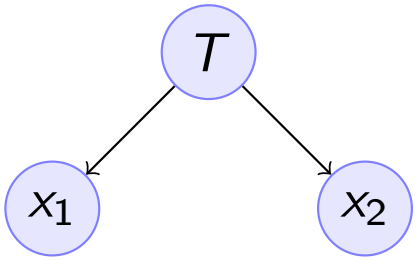
# Time of arrival at work




Time of arrival at work graphical model where $T$ is a traffic jam and $x_1$ is the time John arrives at the office and $x_2$ is the time Jane arrives at the office.

- Even though $x_1, x_2$ are not independent, they become independent once you know $T$, i.e. $x_1 \perp\!\!\!\perp x_2 \mid T$.
- Proof:
- $\mathbb{P}(x_1, x_2, T) = \mathbb{P}(x_2 \mid T)\,\mathbb{P}(x_1 \mid T)\,\mathbb{P}(T)$ from the graph.

# Time of arrival at work




Time of arrival at work graphical model where $T$ is a traffic jam and $x_1$ is the time John arrives at the office and $x_2$ is the time Jane arrives at the office.

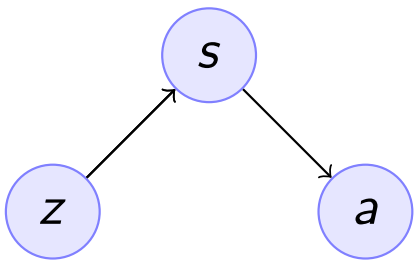
- Even though $x_1, x_2$ are not independent, they become independent once you know $T$, i.e. $x_1 \perp\!\!\!\perp x_2 \mid T$.
- Proof:
- $\mathbb{P}(x_1, x_2, T) = \mathbb{P}(x_2 \mid T)\,\mathbb{P}(x_1 \mid T)\,\mathbb{P}(T)$ from the graph.
- $\mathbb{P}(x_1, x_2, T)/\,\mathbb{P}(T) = \mathbb{P}(x_1, x_2 \mid T) = \mathbb{P}(x_2 \mid T)\,\mathbb{P}(x_1 \mid T)$. □

# School admission

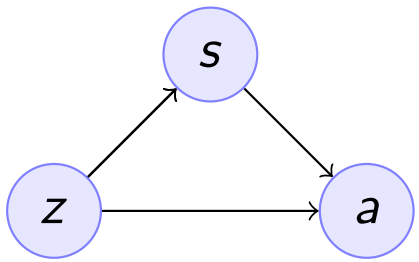
## Example

| School | Male | Female |
|---|---|---|
| A | 62 | 82 |
| B | 63 | 68 |
| C | 37 | 34 |
| D | 33 | 35 |
| E | 28 | 24 |
| F | 6 | 7 |
| Average | 50 | 27 |

- $z$: gender
- $s$: school applied to
- $a$: admission

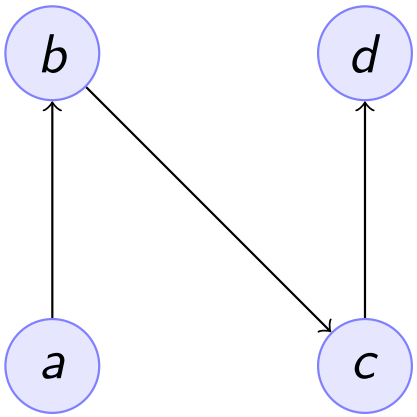



Is admission independent of gender?

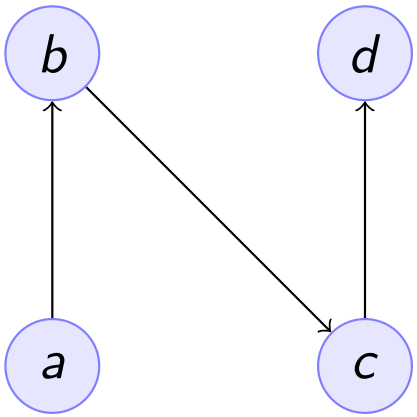



How about here?

# What is the model for this graph?

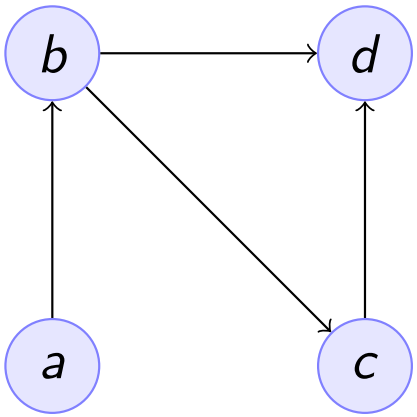


# What is the model for this graph?

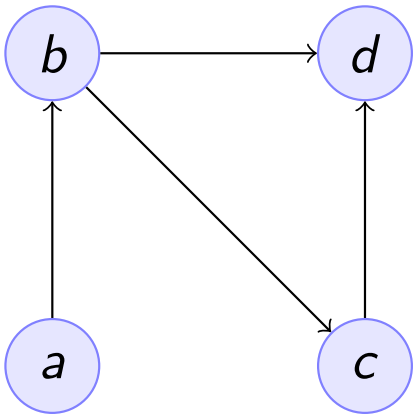



$$\mathbb{P}(a, b, c, d) = \mathbb{P}(d|c)\,\mathbb{P}(c|b)\,\mathbb{P}(b|a)\,\mathbb{P}(a)$$

# What is the model for this graph?

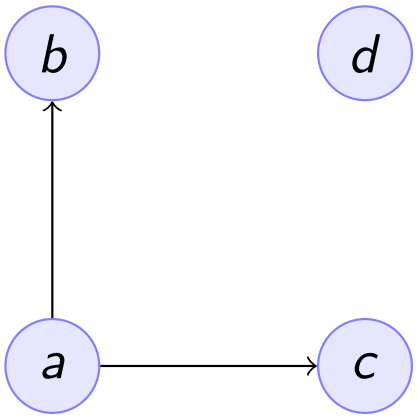


# What is the model for this graph?

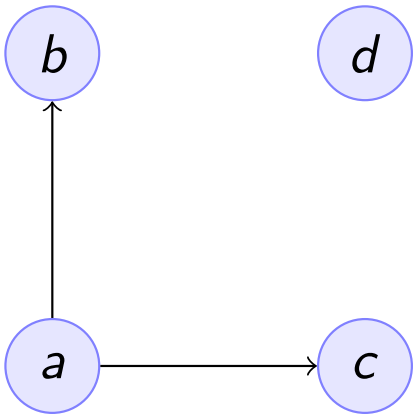



$$\mathbb{P}(a, b, c, d) = \mathbb{P}(d|b, c)\,\mathbb{P}(c|b)\,\mathbb{P}(b|a)\,\mathbb{P}(a)$$

# What is the model for this graph?

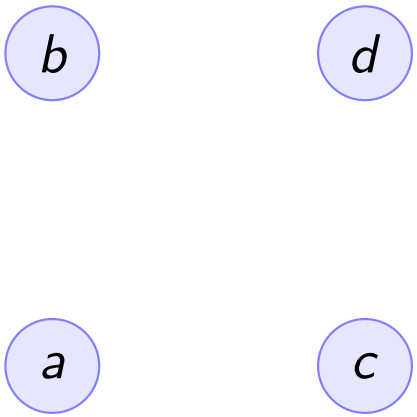


# What is the model for this graph?



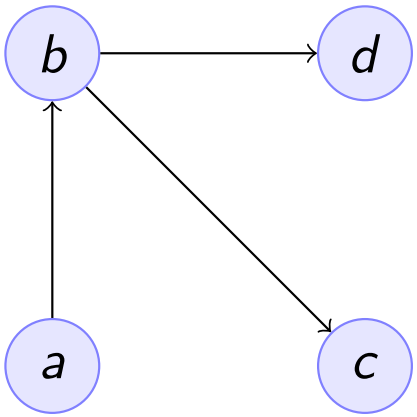

$$\mathbb{P}(a, b, c, d) = \mathbb{P}(d)\,\mathbb{P}(c|a)\,\mathbb{P}(b|a)\,\mathbb{P}(a)$$

# Draw the graph for this model



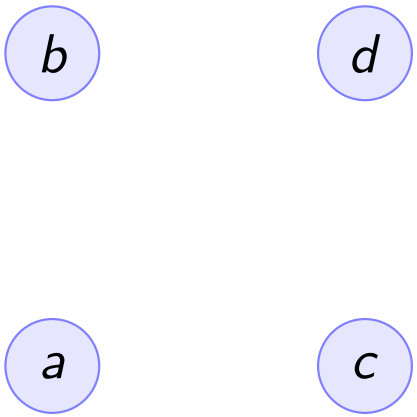

$$P(a, b, c, d) = P(a)P(b|a)P(c|b)P(d|b)$$

# Draw the graph for this model



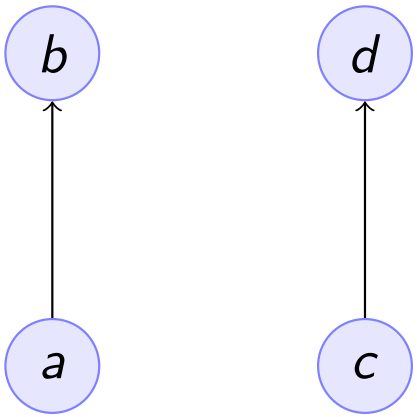

$$P(a, b, c, d) = P(a)P(b|a)P(c|b)P(d|b)$$

# Draw the graph for this model



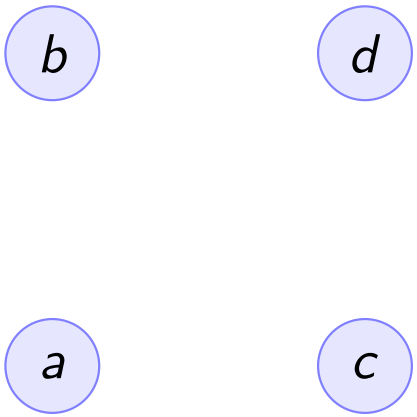

$$P(a, b, c, d) = P(a)P(b|a)P(d|c)P(c)$$

# Draw the graph for this model



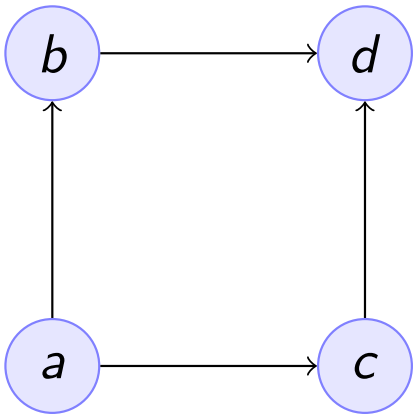

$$P(a, b, c, d) = P(a)P(b|a)P(d|c)P(c)$$

# Draw the graph for this model




$$P(a, b, c, d) = P(a)P(b|a)P(c|a)P(d|b, c)$$

# Draw the graph for this model




$$P(a, b, c, d) = P(a)P(b|a)P(c|a)P(d|b, c)$$

# Conditional independence (general)

- Consider variables $x_1, \ldots, x_n$.
- Let $B, D$ be subsets of $[n]$, and
- $\boldsymbol{x}_B \triangleq (x_i)_{i \in B}$ be the variables with indices in $B$.
- $\boldsymbol{x}_{-j} \triangleq (x_i)_{i \neq i}$ all the variables apart from $x_j$.

## Definition (Conditional independence)

We say $x_i$ is conditionally independent of $\boldsymbol{x}_B$ given $\boldsymbol{x}_D$ and write

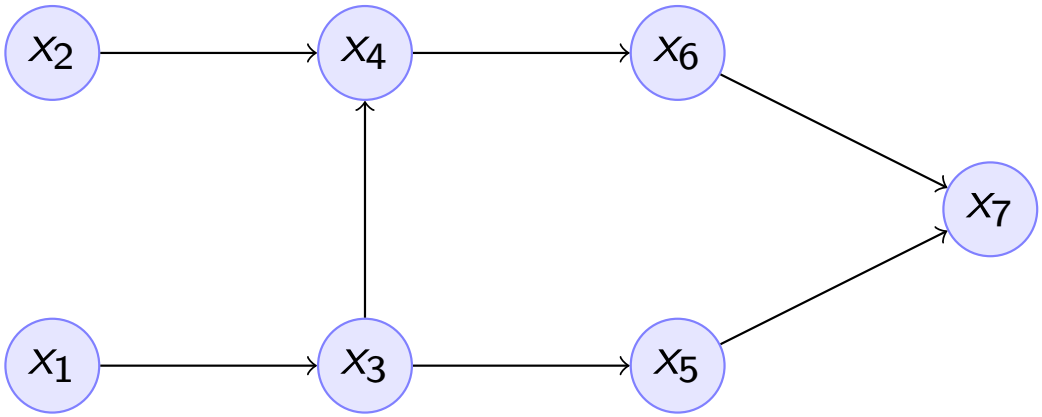
$$x_i \perp\!\!\!\perp \boldsymbol{x}_B \mid \boldsymbol{x}_D$$

if and only if:

$$\mathbb{P}(x_i, \boldsymbol{x}_B \mid \boldsymbol{x}_D) = \mathbb{P}(x_i \mid \boldsymbol{x}_D)\, \mathbb{P}(\boldsymbol{x}_B \mid \boldsymbol{x}_D).$$

- For this to hold in graphical model, $D$ must separate $i$ from $B$ in the graph.

# More complex example




In this example, we have:

$$x_7 \perp\!\!\!\perp x_1, x_2 \mid x_3, x_4$$

and

$$x_7 \perp\!\!\!\perp x_3 \mid x_4, x_5$$

Graphical models
Random variables and probabilities
Graphical model
Exercises

Classification
Classification: Generative modelling
Density estimation

Algorithms for latent variable models
Gradient algorithms
Expectation maximisation

Exercises
Density estimation
Classification

Graphical models

Classification

Classification: Generative modelling

Density estimation

Algorithms for latent variable models

Exercises

# Generative modelling

## General idea

- Data $(x_t, y_t)$.
- Need to model $P(y|x)$.
- Model the complete data distribution: $P(x|y)$, $P(x)$, $P(y)$.
- Calculate $P(y|x) = \frac{P(x|y)P(y)}{P(x)}$.

## Examples

- Naive Bayes classifier.
- Gaussian mixture model.
- Large language models.

## Modelling the data distribution in classification

- Need to estimate the density $P(x|y)$ for each class $y$.
- Need to estimate $P(y)$.

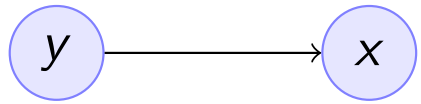
# The basic graphical model

## A discriminative classification model

Here $P(y|x)$ is given directly.

## A generative classification model

Here $P(y|x) = P(x|y)P(y)/P(x)$.



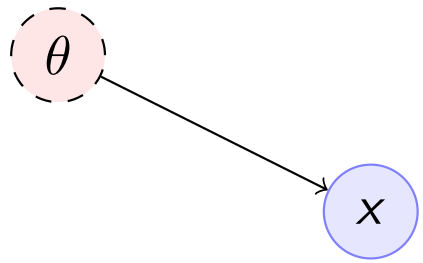

## An unsupervised generative model

Here we just have $P(x)$.

# Adding parameters to the graphical model

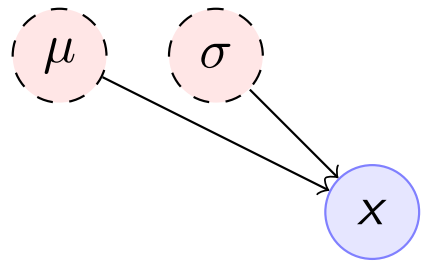
- We can also see the parameters of the distribution as (random) variables.
- We can put those random variables in the graphical model as well.
- Since the parameters are not observed, we denote them with dashed circles.
- They are a type of latent or hidden variable.

## A Bernoulli RV




Here, $x|\theta \sim \text{Bernoulli}(\theta)$

## A normally distributed variable




Here $x|\mu, \sigma \sim \text{Normal}(\mu, \sigma^2)$

# Classification: Naive Bayes Classifier

- Data $(x, y)$
- $x \in X$
- $y \in Y \subset \mathbb{N}$, $N_i$: amount of data from class $i$.

# Classification: Naive Bayes Classifier

- Data $(x, y)$
- $x \in X$
- $y \in Y \subset \mathbb{N}$, $N_i$: amount of data from class $i$.

## Separately model each class

- Assume each class data comes from a different normal distribution
- $x|y = i \sim \mathrm{Normal}(\mu_i, \sigma_i I)$
- For each class, calculate
  - Empirical mean $\hat{\mu}_i = \sum_{t:y_t=i} x_t / N_i$
  - Empirical variance $\hat{\sigma}_i$.

# Classification: Naive Bayes Classifier

- Data $(x, y)$
- $x \in X$
- $y \in Y \subset \mathbb{N}$, $N_i$: amount of data from class $i$.

## Separately model each class

- Assume each class data comes from a different normal distribution
- $x|y = i \sim \mathrm{Normal}(\mu_i, \sigma_i I)$
- For each class, calculate
  - Empirical mean $\hat{\mu}_i = \sum_{t:y_t=i} x_t / N_i$
  - Empirical variance $\hat{\sigma}_i$.

## Decision rule

Use Bayes's theorem:

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

choosing the $y$ with largest posterior $P(y|x)$.

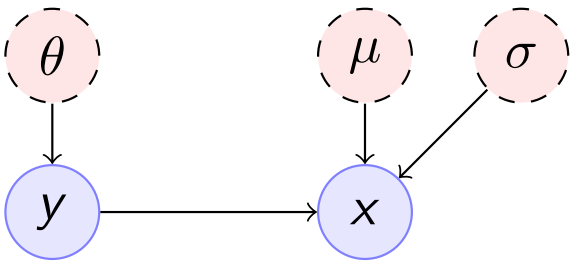
- $P(x|y = i) \propto \exp(-\|\hat{\mu}_i - x\|^2/\hat{\sigma}_i^2)$

# Graphical model for the Naive Bayes Classifier

## When $x \in \mathbb{R}$

Assume $k$ classes, then

- $\mu = (\mu_1, \ldots, \mu_k)$
- $\sigma = (\sigma_1, \ldots, \sigma_k)$
- $\theta = (\theta_1, \ldots, \theta_k)$




- $y \mid \theta \sim \text{Mult}(\theta)$
- $x \mid y, \mu, \sigma \sim \text{Normal}(\mu_y, \sigma_y^2)$

# Density estimation

The simplest type of generative model is just modelling the distribution of $x$. There are a number of models for this.

## Parametric models

- Fixed histograms
- Gaussian Mixtures

## Non-parametric models

- Variable-bin histograms
- Infinite Gaussian Mixture Model
- Kernel methods

# Histograms

## Fixed histogram

- Hyper-Parameters: number of bins
- Parameters: Number of points in each bin.

## Variable histogram

- Hyper-parameters: Rule for constructing bins
- Generally $\sqrt{n}$ points in each bin.

# Gaussian Mixture Model

## Hyperparameters:

- Number of Gaussian $k$.

## Parameters:

- Multinomial distribution $\beta$ over Gaussians
- For each Gaussian $i$, center $\mu_i$, covariance matrix $\Sigma_i$.

## Algorithms:

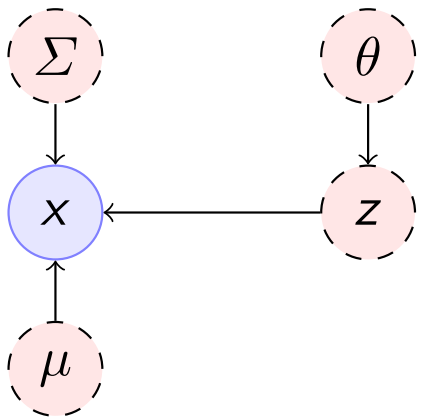
- Expectation Maximisation
- Gradient Ascent
- Variational Bayesian Inference (with appropriate prior)

# Details of Gaussian mixture models

## Model. For each point $x_t$:

- $z_t \mid \theta \sim \mathrm{Mult}(\theta_i)$, $\theta \in \stackrel{\geq}{+}^k$
- $x_t | z_t = i \sim \mathrm{Normal}(\mu_i, \Sigma_i)$.
- $\mathrm{Mult}(\theta)$ is multinomial

$$\mathbb{P}(z_t = i \mid \theta) = \theta_i$$




- $\mathrm{Normal}(\mu, \Sigma)$ is multivariate Gaussian

$$p(x \mid \mu, \Sigma) \propto \exp(-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu))$$

- The generating distribution is

$$p(x|\theta, \mu, \Sigma) = \sum_{z \in [k]} p(x \mid \mu_z, \Sigma_z) P(z \mid \theta).$$

# Applications of Gaussian mixture models

- Density estimation
- Clustering
- Used as part of a more complex model.

Graphical models
- Random variables and probabilities
- Graphical model
- Exercises

Classification
- Classification: Generative modelling
- Density estimation

Algorithms for latent variable models
- Gradient algorithms
- Expectation maximisation

Exercises
- Density estimation
- Classification

# Gradient ascent

In the following we use $\theta$ for all the parameters of the Gaussian mixture model, with $x = (x_1, \ldots, x_T)$ and $z = (z_1, \ldots, z_T)$

## Objective function

One way to estimate $\theta$ is through maximising the likelihood $L(\theta) = P(x|\theta)$

## Marginalisation over latent variable

However, we need to marginalise over all values $z$

$$L(\theta) = \sum_z P(z, x|\theta)$$

For $T$ data points and $k$ different values of $z_t$, there are $k^T$ vectors $z$ to sum over.

## Gradient ascent

If we can calculate the gradient of $L$, we can use gradient ascent to update our parameters:

$$\theta^{(n+1)} = \theta^{(n)} + \alpha \nabla_\theta L(\theta).$$

# Gradient calculation

Here we use the log trick: $\nabla \ln f(\theta) = \nabla f(\theta)/f(\theta)$.

$$\nabla_\theta L(\theta) = \sum_z \nabla_\theta P(z, x \mid \theta) \tag{1}$$

$$= \sum_z P(z, x \mid \theta) \nabla_\theta \ln P(z, x \mid \theta) \tag{2}$$

$$= \sum_z P(x \mid z, \theta) P(z \mid \theta) \nabla_\theta \ln P(z, x \mid \theta) \tag{3}$$

$$\approx \frac{1}{m} \sum_{i=1}^{m} P(x \mid z^{(i)}, \theta) \nabla_\theta \ln P(z^{(i)}, x \mid \theta) \qquad z^{(i)} \sim P(z \mid \theta) \tag{4}$$

The final approximates the sum with the sample mean, sampling $z^{(i)}$ from the distribution. Hence, we can implement the following algorithm

- For $i = 1, \ldots, m$: $z^{(i)} \sim P(z \mid \theta^{(n)})$
- $d^{(n)} = \frac{1}{m} \sum_{i=1}^{m} P(x \mid z^{(i)}, \theta) \nabla_\theta \ln P(z^{(i)}, x \mid \theta^{(n)})$
- $\theta^{(n+1)} = \theta^n + \alpha d^{(n)}$.

# A lower bound on the likelihood

For any distribution $G(z)$, and specifically for $G(z) = P(z|x, \theta^{(k)})$:

$$\begin{aligned}
\ln P(x|\theta) &= \sum_z G(z) \ln P(x|\theta) = \sum_z G(z) \ln[P(x, z|\theta)/P(z|x, \theta)] \\
&= \sum_z G(z)[\ln P(x, z|\theta) - \ln P(z|x, \theta)] \\
&= \sum_z G(z) \ln P(x, z|\theta) - \sum_z G(z) \ln P(z|x, \theta) \\
&= \sum_z P(z|x, \theta^{(k)}) \ln P(x, z|\theta) - \sum_z P(z|x, \theta^{(k)}) \ln P(z|x, \theta) \\
&\geq \sum_z P(z|x, \theta^{(k)}) \ln P(x, z|\theta) - \sum_z P(z|x, \theta^{(k)}) \ln P(z|x, \theta^{(k)}) \\
&= Q(\theta \mid \theta^{(k)}) + \mathbb{H}(z \mid x, \theta^{(k)}),
\end{aligned}$$

where

$$\mathbb{H}(z \mid x, \theta^{(k)}) = \sum_z P(z \mid x, \theta^{(k)}) \ln P(z \mid x, \theta^{(k)})$$

is the entropy of $z$ for a fixed $x, \theta^{(k)}$. As this is not negative, $\ln P(x|\theta) \geq Q(\theta \mid \theta^{(k)})$.

# Some information theory

Information theory notation can be a bit confusing. Sometimes we talk about random variables $\omega$, and sometimes about probability measures $P$. This is context-dependent.

## Entropy

For a random variable $\omega$ under distribution $P$, we denote the entropy as

$$\mathbb{H}_P(\omega) \equiv \mathbb{H}(P) \equiv \mathbb{H}(\omega) = \sum_{\omega \in \Omega} P(\omega) \ln P(\omega).$$

## KL Divergence

For two probabilities $P, Q$ over random outcomes in the same space $\Omega$, we define

$$D_{KL}(P \| Q) = \sum_{\omega \in \Omega} P(\omega) \ln \frac{P(\omega)}{Q(\omega)}$$

## The Gibbs Inequality

$D_{KL}(P \| Q) \geq 0$, or $\sum_x \ln P(x) P(x) \geq \sum_x \ln Q(x) P(x)$.

# EM Algorithm (Dempster et al, 1977)

- Initial parameter $\boldsymbol{\beta}^{(0)}$, observed data $x$
- For $k = 0, 1, \ldots$

– Expectation step:

$$Q(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(k)}) \triangleq \mathbb{E}_{z \sim P(z|x,\boldsymbol{\beta}^{(k)})}[\ln P(x, z|\boldsymbol{\beta})] = \sum_{z} [\ln P(x, z|\boldsymbol{\beta})] P(z \mid x, \boldsymbol{\beta}^{(k)})$$

– Maximisation step:

$$\boldsymbol{\beta}^{(k+1)} = \arg\max_{\boldsymbol{\beta}} Q(\boldsymbol{\beta}, \boldsymbol{\beta}^{(k)}).$$

See *Expectation-Maximization as lower bound maximization, Minka, 1998*

# Minorise-Maximise

EM can be seen as a version of the minorise-maximise algorithm

- $f(\boldsymbol{\beta})$: Target function to maximise
- $Q(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)})$: surrogate function

## $Q$ Minorizes $f$

This means surrogate is always a lower bound so that

$$f(\boldsymbol{\beta}) \geq Q(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)}), \qquad f(\boldsymbol{\beta}^{(k)}) = Q(\boldsymbol{\beta}^{(k)}|\boldsymbol{\beta}^{(k)}),$$

## Algorithm

- Calculate: $Q(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)})$
- Optimise: $\boldsymbol{\beta}^{(k+1)} = \arg\max_{\boldsymbol{\beta}} Q(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)})$.

Graphical models
- Random variables and probabilities
- Graphical model
- Exercises

Classification
- Classification: Generative modelling
- Density estimation

Algorithms for latent variable models
- Gradient algorithms
- Expectation maximisation

Exercises
- Density estimation
- Classification

# GMM versus histogram

- Generate some data $x$ from an arbitrary distribution in $\mathbb{R}$.
- Fit the data with a histogram for varying numbers of bins
- Fit a GMM with varying numbers of Gaussians
- What is the best fit? How can you measure it?

# GMM Classifier

## Base class: sklearn GaussianMixtureModel

- *fit()* only works for Density Estimaiton
- *predict()* only predicts cluster labels

## Problem

- Create a GMMClassifier class
- *fit()* should take X, y, arguments
- *predict()* should predict class labels
- Hint: Use $predict_{proba}()$ and multiple GMM models