Probabilistic Graphical Models **Fall 2013**

Lecture 1 — October 9, 2013

Lecturer: Guillaume Obozinski *Scribe: Huu Dien Khue Le, Robin Bénesse*

The web page of the course: http://www.di.ens.fr/~fbach/courses/fall2013/

1.1 Introduction

1.1.1 Problem

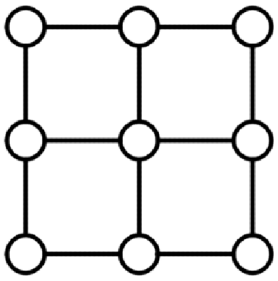
To model complex data, one is confronted with two main questions:

- How to manage the complexity of the data to be processed?
- How to infer global properties from local models?

These questions lead to 3 types of problems: the representation of data (or how to obtain a global model from a local model), the inference of the distributions (how to use the model), and the learning of the models (what are the parameters of the models?).

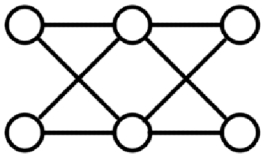
1.1.2 Examples

- **Image:** consider a 100×100 (pixels) monochromatic image. If each pixel is modelled by a discrete random variable (so there are 10000 of them), then the image can be modelled using a grid of the form:

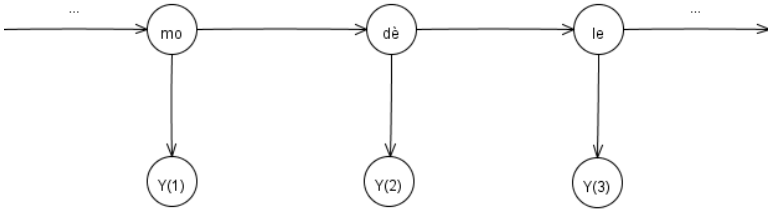


- **Bioinformatics:** consider a long sequence of 10000 ADN bases. If each base of this sequence is modelled by a discrete random variable (that, in general, can take values in $\{A, C, G, T\}$), then the sequence can be modelled by a Markov chain:

- **Finance:** consider the evolution of stock prices in discrete time, where we have values at time n. It is reasonable to postulate that the change of price of a stock at time n only depends only on its price (or the price of other stocks) at time $n-1$. For only two stocks, a possible simplified model is the following dependency graph:



- **Speech processing:** consider the syllables of a word and the way they are interpreted by a human ear or by a computer. Each syllable can be represented by a random sound. The objective is then to retrieve the word from the sequence of sounds heard or recorded. In this case, we can use a hidden Markov model:



- **Text:** consider a text with 1000000 words. The text is modelled by a vector such that each of its components equals to the number of times each keyword appears. This is usually called the "bag of words" model. This model seems to be weak, as it does not take the order of the words into account. However, it works quite well in practice. A so-called *naive Bayes classifier* can be used for classification (for example spam *vs* non spam).

It is clear that models which ignore the dependence among variables are too simple for real-world problems. On the other hand, models in which every random variable is dependent all or too many other ones are doomed both for statistical (lack of data) and computational reasons. Therefore, in practice, one has to make suitable assumptions to design models with

the right level of complexity, so that the models obtained are able to *generalize* well from a statistical point of view and lead to tractable computations from an algorithmic perspective.

1.2 Basic notations and properties

In this section we recall some basic notations and properties of random variables.

Convention: Mathematically, the probability that a random variable X takes the value x is denoted $p(X = x)$. In this document, we simply write $p(X)$ to denote a distribution over the random variable X, or $p(x)$ to denote the distribution evaluated for the particular value x. It is similar for more variables.

Fundamental rules. For two random variables X, Y we have

- Sum rule:
$$p(X) = \sum_Y p(X, Y).$$
- Product rule:
$$p(X, Y) = p(Y|X)p(X).$$

Independence. Two random variables X and Y are said to be independent if and only if

$$P(X, Y) = P(X)P(Y).$$

Conditional independence. Let X, Y, Z be random variables. We define X and Y to be conditionally independent given Z if and only if

$$P(X, Y|Z) = P(X|Z)P(Y|Z).$$

Property: If X and Y are conditionally independent given Z, then

$$P(X|Y, Z) = P(X|Z).$$

Independent and identically distributed. A set of random variables is independent and identically distributed (i.i.d.) if each random variable has the same probability distribution as the others and all are mutually independent.

Bayes formula. For two random variables X, Y we have

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}.$$

(Note that Bayes formula is not a Bayesian formula in the sense of Bayesian statistics).

1.3 Statistical models

Definition 1.1 (Statistical model) *A (parametric) statistical model $\mathcal{P}_\Theta$ is a collection of probability distributions (or a collection of probability density functions[1]) defined on the same space and parameterized by parameters θ belonging to a set $\Theta \subset \mathbb{R}^p$. Formally:*

$$\mathcal{P}_\Theta = \{p_\theta(\cdot) \mid \theta \in \Theta\}.$$

1.3.1 Bernoulli model

Consider a binary random variable X that can take the value 0 or 1. If $p(X = 1)$ is parametrized by $\theta \in [0, 1]$:

$$\begin{cases} \mathbb{P}(X = 1) = \theta \\ \mathbb{P}(X = 0) = 1 - \theta \end{cases}$$

then a probability distribution of the Bernoulli model can be written as

$$p(X = x; \theta) = \theta^x (1 - \theta)^{1-x} \tag{1.1}$$

and we can write

$$X \sim \text{Ber}(\theta). \tag{1.2}$$

The Bernoulli model is the collection of these distributions for $\theta \in \Theta = [0, 1]$.

1.3.2 Binomial model

A binomial random variable $\text{Bin}(\theta, N)$ is defined as the value of the sum of n i.i.d. Bernoulli r.v. with parameter θ. The distribution of a binomial random variable N is

$$\mathbb{P}(N = k) = \binom{n}{k} \theta^k (1 - \theta)^{n-k}.$$

The set Θ is the same as for the Bernoulli model.

1.3.3 Multinomial model

Consider a discrete random variable C that can take one of K possible values $\{1, 2, \ldots, K\}$. The random variable C can be represented by a K-dimensional random variable $X = (X_1, X_2, \ldots, X_K)^T$ for which the event $\{C = k\}$ corresponds to the event

$$\{X_k = 1 \text{ and } X_l = 0, \forall l \neq k\}.$$

[1] In which case, they are all defined with respect to the same base measure, such as the Lebesgue measure in $\mathbb{R}^d$

If we parametrize $\mathbb{P}(C = k)$ by a parameter $\pi_k \in [0, 1]$, then by definition we also have

$$\mathbb{P}(X_k = 1) = \pi_k \quad \forall k = 1, 2, \ldots, K,$$

with $\sum_{k=1}^{K} \pi_k = 1$. The probability distribution over $\mathbf{x} = (x_1, \ldots, x_k)$ can be written as

$$p(\mathbf{x}; \boldsymbol{\pi}) = \prod_{k=1}^{K} \pi_k^{x_k} \tag{1.3}$$

where $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_K)^T$. We will denote $\mathcal{M}(1, \pi_1, \ldots, \pi_K)$ such a discrete distribution. The corresponding set of parameters is $\Theta = \{\boldsymbol{\pi} \in \mathbb{R}^+ \mid \sum_{k=1}^{K} \pi = 1\}$.

Now if we consider n independent observations of a $\mathcal{M}(1, \boldsymbol{\pi})$ multinomial random variable X, and we denote by N_k the number of observations for which $x_k = 1$, then the joint distribution of $N_1, N_2, \ldots, N_K$ is called a multinomial $\mathcal{M}(n, \boldsymbol{\pi})$ distribution. It takes the form:

$$p(n_1, n_2, \ldots, n_K; \boldsymbol{\pi}, n) = \frac{n!}{n_1! n_2! \ldots n_K!} \prod_{k=1}^{K} \pi_k^{n_k} \tag{1.4}$$

and we can write

$$(N_1, \ldots, N_K) \sim \mathcal{M}(N, \pi_1, \pi_2, \ldots, \pi_K). \tag{1.5}$$

The multinomial $\mathcal{M}(n, \boldsymbol{\pi})$ is to the $\mathcal{M}(1, \boldsymbol{\pi})$ distribution, as the binomial distribution is to the Bernoulli distribution. In the rest of this course, when we will talk about multinomial distributions, we will always refer to a $\mathcal{M}(1, \boldsymbol{\pi})$ distribution.

1.3.4 Gaussian models

The Gaussian distribution is also known as the normal distribution. In the case of a scalar variable X, the Gaussian distribution can be written in the form

$$\mathcal{N}(x; \mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{1.6}$$

where μ is the mean and σ^2 is the variance. For a d-dimensional vector $\mathbf{x}$, the multivariate Gaussian distribution takes the form

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{d/2}} \frac{1}{|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu})\right) \tag{1.7}$$

where $\boldsymbol{\mu}$ is a d-dimensional vector, $\boldsymbol{\Sigma}$ is a $d \times d$ symmetric positive definite matrix , and $|\boldsymbol{\Sigma}|$ denotes the determinant of $\boldsymbol{\Sigma}$. It is a well-known property that the parameter $\boldsymbol{\mu}$ is equal to the expectation of X and that the matrix $\boldsymbol{\Sigma}$ is the covariance matrix of X, which means that $\boldsymbol{\Sigma}_{ij} = \mathbb{E}[(X_i - \mu_i)(X_j - \mu_j)]$.

1.4 Parameter estimation by maximum likelihood

1.4.1 Definition

Maximum likelihood estimation (MLE) is a method of estimating the parameters of a statistical model. Suppose we have a sample $x_1, x_2, \ldots, x_n$ of n independent and identically distributed observations, coming from a distribution $p(x_1, x_2, \ldots, x_n; \theta)$ where θ is an unknown parameter (both x_i and θ can be vectors). As the name suggests, the MLE finds the parameter $\hat{\theta}$ under which the data $x_1, x_2, \ldots, x_n$ are most likely:

$$\hat{\theta} = \operatorname*{argmax}_{\theta} p(x_1, x_2, \ldots, x_n; \theta) \tag{1.8}$$

The probability on the right-hand side in the above equation can be seen as a function of θ and can be denoted by $\mathcal{L}(\theta)$:

$$\mathcal{L}(\theta) = p(x_1, x_2, \ldots, x_n; \theta) \tag{1.9}$$

This function is called the *likelihood*.
As $x_1, x_2, \ldots, x_n$ are independent and identically distributed, we have

$$\mathcal{L}(\theta) = \prod_{i=1}^{n} p(x_i; \theta) \tag{1.10}$$

In practice it is often more convenient to work with the logarithm of the likelihood function, called the *log-likelihood*:

$$\ell(\theta) = \log \mathcal{L}(\theta) = \log \prod_{i=1}^{n} p(x_i; \theta) \tag{1.11}$$

$$= \sum_{i=1}^{n} \log p(x_i; \theta) \tag{1.12}$$

Next, we will apply this method for the models presented previously. **We assume that all the observations are independent and identically distributed** in all of the remainder of this lecture.

1.4.2 MLE for the Bernoulli model

Consider n observations $x_1, x_2, \ldots, x_n$ of a binary random variable X following a Bernoulli distribution $\mathrm{Ber}(\theta)$. From (1.12) and (1.1) we have

$$\begin{aligned} \ell(\theta) &= \sum_{i=1}^{n} \log p(x_i; \theta) \\ &= \sum_{i=1}^{n} \log \theta^{x_i} (1-\theta)^{1-x_i} \\ &= N \log(\theta) + (n - N) \log(1-\theta) \end{aligned}$$

where $N = \sum_{i=1}^{n} x_i$.
As $\ell(\theta)$ is strictly concave, it has a unique maximizer, and since the function is in addition differentiable, its maximizer $\hat{\theta}$ is the zero of its gradient $\nabla\ell(\theta)$:

$$\nabla\ell(\theta) = \frac{\partial}{\partial\theta}\ell(\theta) = \frac{N}{\theta} - \frac{n-N}{1-\theta}.$$

It is easy to show that $\nabla\ell(\theta) = 0 \iff \theta = \frac{N}{n}$. Therefore we have

$$\hat{\theta} = \frac{N}{n} = \frac{x_1 + x_2 + \cdots + x_n}{n}. \tag{1.13}$$

1.4.3 MLE for the multinomial model

Consider N observations $X_1, X_2, \ldots, X_N$ of a discrete random variable X following a multinomial distribution $\mathcal{M}(1, \boldsymbol{\pi})$, where $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_K)^T$. We denote $\mathbf{x}_i$ $(i = 1, 2, \ldots, N)$ the K-dimensional vectors of 0s and 1s representing X_i, as presented in Section 1.3.3.
From (1.12) and (1.3) we have

$$\begin{aligned}
\ell(\boldsymbol{\pi}) &= \sum_{i=1}^{N} \log p(\mathbf{x}_i;\, \boldsymbol{\pi}) \\
&= \sum_{i=1}^{N} \log \left(\prod_{k=1}^{K} \pi_k^{x_{ik}} \right) \\
&= \sum_{i=1}^{N} \sum_{k=1}^{K} x_{ik} \log \pi_k \\
&= \sum_{k=1}^{K} n_k \log \pi_k
\end{aligned}$$

where $n_k = \sum_{i=1}^{N} x_{ik}$ (n_k is therefore the number of observations of $x_k = 1$).
We need to maximize this quantity subject to the constraint $\sum_{k=1}^{K} \pi_k = 1$.

Brief review on Lagrange duality

Lagrangian. Consider the following convex optimization problem:

$$\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}), \text{ subject to } \mathbf{A}\mathbf{x} = \mathbf{b} \tag{1.14}$$

where f is a convex function, $\mathcal{X} \subset \mathbb{R}^p$ is a convex set included in the domain[2] of f, $\mathbf{A} \in \mathbb{R}^{n\times p}, \mathbf{b} \in \mathbb{R}^n$.
The *Lagrangian* associated with this optimization problem is defined as

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T(\mathbf{A}\mathbf{x} - \mathbf{b}) \tag{1.15}$$

[2]The domain of a function is the set on which the function is finite.

The vector $\boldsymbol{\lambda} \in \mathbb{R}^n$ is called the *Lagrange multiplier vector*.

Lagrange dual function. The *Lagrange dual function* is defined as

$$g(\boldsymbol{\lambda}) = \min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}) \tag{1.16}$$

The problem of maximizing $g(\boldsymbol{\lambda})$ with respect to $\boldsymbol{\lambda}$ is known as the *Lagrange dual problem*.

Max-min inequality. For any $f : \mathbb{R}^n \times \mathbb{R}^m$ and any $w \in \mathbb{R}^n$ and $z \in \mathbb{R}^m$, we have

$$f(w,z) \leq \max_{z \in Z} f(w,z) \Longrightarrow \min_{w \in W} f(w,z) \leq \min_{w \in W} \max_{z \in Z} f(w,z) \tag{1.17}$$

$$\Longrightarrow \max_{z \in Z} \min_{w \in W} f(w,z) \leq \min_{w \in W} \max_{z \in Z} f(w,z). \tag{1.18}$$

The last inequality is known as the *max-min inequality*.

Duality. It is easy to show that

$$\max_{\boldsymbol{\lambda}} L(\mathbf{x}, \boldsymbol{\lambda}) = \begin{cases} f(\mathbf{x}) & \text{if } \mathbf{Ax} = \mathbf{b} \\ +\infty & \text{otherwise.} \end{cases}$$

Which gives us

$$\min_{\mathbf{x}} f(\mathbf{x}) = \min_{\mathbf{x}} \max_{\boldsymbol{\lambda}} L(\mathbf{x}, \boldsymbol{\lambda}) \tag{1.19}$$

Now from (1.16), (1.18) and (1.19) we have

$$\max_{\boldsymbol{\lambda}} g(\boldsymbol{\lambda}) = \max_{\boldsymbol{\lambda}} \min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}) \leq \min_{\mathbf{x}} \max_{\boldsymbol{\lambda}} L(\mathbf{x}, \boldsymbol{\lambda}) = \min_{\mathbf{x}} f(\mathbf{x}) \tag{1.20}$$

This inequality says that the optimal value d^* of the Lagrange dual problem always lower-bounds the optimal value p^* of the original problem. This property is called the *weak duality*. If the equality $d^* = p^*$ holds, then we say that the *strong duality* holds. Strong duality means that the order of the minimization over $\mathbf{x}$ and the maximization over $\boldsymbol{\lambda}$ can be switched without affecting the result.

Slater's constraint qualification lemma. If there exists an $\mathbf{x}$ in the relative interior of $\mathcal{X} \cap \{\mathbf{Ax} = \mathbf{b}\}$ then strong duality holds. (Note that by definition $\mathcal{X}$ is included in the domain of f so that if $\mathbf{x} \in \mathbf{X}$ then $f(\mathbf{x}) < \infty$.)

Note that all the above notions and results are stated for the problem (1.14) only. For a more general problem and more details about Lagrange duality, please refer to [9] (chapter 5).

Back to our problem

We need to minimize

$$f(\boldsymbol{\pi}) = -\ell(\boldsymbol{\pi}) = -\sum_{k=1}^{K} n_k \log \pi_k \tag{1.21}$$

subject to the constraint $\mathbf{1}^T\boldsymbol{\pi} = 1$.
The Lagrangian of this problem is

$$L(\boldsymbol{\pi}, \lambda) = -\sum_{k=1}^{K} n_k \log \pi_k + \lambda \left(\sum_{k=1}^{K} \pi_k - 1 \right) \tag{1.22}$$

Clearly, as $n_k \geq 0$ $(k = 1, 2, \ldots, K)$, f is convex and this problem is a convex optimization problem. Moreover, it is trivial that there exist $\pi_1, \pi_2, \ldots, \pi_K$ such that $\pi_k > 0$ $(k = 1, 2, \ldots, K)$ and $\sum_{k=1}^{K} \pi_k = 1$, so by *Slater's constraint qualification*, the problem has strong duality property. Therefore, we have

$$\min_{\boldsymbol{\pi}} f(\boldsymbol{\pi}) = \max_{\lambda} \min_{\boldsymbol{\pi}} L(\boldsymbol{\pi}, \lambda) \tag{1.23}$$

As $L(\boldsymbol{\pi}, \lambda)$ is convex with respect to $\boldsymbol{\pi}$, to find $\min_{\boldsymbol{\pi}} L(\boldsymbol{\pi}, \lambda)$, it suffices to take derivatives with respect to π_k. This yields

$$\frac{\partial L}{\partial \pi_k} = -\frac{n_k}{\pi_k} + \lambda = 0, \ k = 1, 2, \ldots, K.$$

or

$$\pi_k = \frac{n_k}{\lambda}, \ k = 1, 2, \ldots, K. \tag{1.24}$$

Substituting these into the constraint $\sum_{k=1}^{K} \pi_k = 1$ we get $\sum_{k=1}^{K} n_k = \lambda$, yielding $\lambda = N$. From this and (1.24) we get finally

$$\hat{\pi}_k = \frac{n_k}{N}, \ k = 1, 2, \ldots, K. \tag{1.25}$$

Remark: $\hat{\pi}_k$ is the fraction of the N observations for which $x_k = 1$.

1.4.4 MLE for the univariate Gaussian model

Consider n observations $x_1, x_2, \ldots, x_n$ of a random variable X following a Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$. From (1.12) and (1.6) we have

$$\begin{aligned}
\ell(\mu, \sigma^2) &= \sum_{i=1}^{n} \log p(x_i;\, \mu, \sigma^2) \\
&= \sum_{i=1}^{n} \log \left[\frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left(-\frac{(x_i - \mu)^2}{2\sigma^2} \right) \right] \\
&= -\frac{n}{2} \log(2\pi) - \frac{n}{2} \log(\sigma^2) - \frac{1}{2} \sum_{i=1}^{n} \frac{(x_i - \mu)^2}{\sigma^2}.
\end{aligned}$$

We need to maximize this quantity with respect to μ and σ^2. By taking derivative with respect to μ and then σ^2, it is easy to obtain that the pair $(\hat{\mu}, \hat{\sigma}^2)$, defined by

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{1.26}$$

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n} (x_i - \hat{\mu})^2 \tag{1.27}$$

is the only stationary point of the likelihood. One can actually check (for example computing the Hessian w.r.t. (μ, σ^2) that this actually a maximum. We will have a confirmation of this in the lecture on exponential families.

1.4.5 MLE for the multivariate Gaussian model

Let $X \in \mathbb{R}^d$ be a Gaussian random vector, with mean vector $\mu \in \mathbb{R}^d$ and a covariance matrix $\Sigma \in \mathbb{R}^{d\times d}$ (positive definite):

$$p(x \mid \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{k}{2}}} \frac{1}{\sqrt{\det \Sigma}} \exp\left(\frac{-(x-\mu)^\top \Sigma^{-1} (x-\mu)}{2} \right)$$

Let $x_1, \ldots, x_n$ be a i.i.d. sample. The log-likelihood is given by:

$$\begin{aligned} \ell(\mu, \Sigma) &= \log p(x_1, \ldots, x_n; \mu, \Sigma) \\ &= \log \prod_{i=1}^{n} p(x_i \mid \mu, \Sigma) \\ &= -\left(\frac{nd}{2} \log(2\pi) + \frac{n}{2} \log(\det \Sigma) + \frac{1}{2} \sum_{i=1}^{n} (x_i - \mu)^\top \Sigma^{-1} (x_i - \mu) \right) \end{aligned}$$

In this case, one should be careful that these log-likelihoods are not concave with respect to the pair of parameters (μ, Σ). They are concave w.r.t. μ when Σ is fixed but they are not even concave with respect to Σ when μ is fixed.

1.4.6 Digression: review on differentials

Differentiable function. A function f is differentiable at $x \in \mathbb{R}^d$ if there exists a linear form df such that:

$$f(x+h) - f(x) = df(h) + o(||h||)$$

Since $\mathbb{R}^d$ is a Hilbert space, we know that there exists $g \in \mathbb{R}^d$ such that $df_x(h) = \langle g, h \rangle$. We call g the gradient of f : $g = \nabla f(x)$.

Example 1 : if $f \mapsto a^\top x + b$ then we have :

$$f(x+h) - f(x) = a^\top h$$

and thus

$$\nabla f(x) = a.$$

Example 2 : if $f \mapsto \frac{1}{2} x^\top A x$ then we have :

$$\begin{aligned} f(x+h) - f(x) &= \frac{1}{2}(x+h)^T A(x+h) - \frac{1}{2} x^\top A x \\ &= \frac{1}{2}\left(x^\top A h + h^\top A x\right) + o\left(||h||\right) \end{aligned}$$

The gradient is then :

$$\nabla f(x) = \frac{1}{2}\left(Ax + A^\top x\right)$$

Let us first differentiate $\ell(\mu, \Sigma)$ w.r.t. μ.

We need to differentiate :

$$\mu \mapsto (x_i - \mu)^\top \Sigma^{-1} (x_i - \mu)$$

Which is equal to $f \circ g$ where :

$$\begin{array}{rccc} f & : & \mathbb{R}^d & \to & \mathbb{R} \\ & & y & \mapsto & y^\top \Sigma^{-1} y \end{array}$$

and

$$\begin{array}{rccc} g & : & \mathbb{R}^d & \to & \mathbb{R}^d \\ & & \mu & \mapsto & \mu - x_i \end{array}$$

Reminder : Composition of differentials

$$\begin{aligned} d(f \circ g) &= df_{g(x)}\left(dg_x(h)\right) \\ &= df_{g(x)} \circ dg_x(h) \end{aligned}$$

The differential of f is : $df_y(h) = \langle \nabla f(y), h \rangle = \langle \frac{1}{2} \left(\Sigma^{-1} y + (\Sigma^{-1})^\top y \right), h \rangle = \langle \Sigma^{-1} y, h \rangle$ as Σ^{-1} is symmetric.

The function $\ell : \mu \mapsto (x_i - \mu)\Sigma^{-1}(x_i - \mu)$. We have :

$$d\ell_\mu(h) = \langle \Sigma^{-1}(\mu - x_i), h \rangle$$

We deduce the gradient of ℓ :

$$\nabla \ell(\mu) = \Sigma^{-1}(\mu - x_i)$$

1.4.7 Back to the MLE for the multivariate Gaussian

Remember that the function we want to differentiate is :

$$\ell(\mu, \Sigma) = -\left(\frac{nd}{2} \log(2\pi) + \frac{n}{2} \log(\det \Sigma) + \frac{1}{2} \sum_{i=1}^{n} (x_i - \mu)^\top \Sigma^{-1} (x_i - \mu) \right)$$

According to the results above, we have :

$$\begin{aligned} \nabla_\mu \ell\left(\mu, \Sigma^{-1}\right) &= \sum_{i=1}^{n} \Sigma^{-1}(\mu - x_i) \\ &= \Sigma^{-1} \left(n\mu - \sum_{i=1}^{n} x_i \right) \\ &= \Sigma^{-1} (n\mu - n\overline{x}) \end{aligned}$$

where $\overline{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$

The gradient is equal to 0 iff :

$$\boxed{\widehat{\mu} = \frac{1}{n} \sum_{i=1}^{n} x_i}$$

Let us now differentiate ℓ w.r.t. Σ^{-1}. Let $A = \Sigma^{-1}$. We have :

$$\ell(\mu, \Sigma) = -\left(\frac{nd}{2} \log(2\pi) - \frac{n}{2} \log(\det A) + \frac{1}{2} \sum_{i=1}^{n} (x_i - \mu)^\top A (x_i - \mu) \right)$$

The last term is a real number, so it equal to its trace. Thus :

$$\begin{aligned}\ell(\mu,\Sigma) &= -\left(\frac{nd}{2}\log(2\pi) - \frac{n}{2}\log(\det A) + \frac{1}{2}\sum_{i=1}^{n}\text{Trace}\left((x_i-\mu)^\top A(x_i-\mu)\right)\right)\\ &= -\left(\frac{nd}{2}\log(2\pi) - \frac{n}{2}\log(\det A) + \frac{n}{2}\text{Trace}(A\widetilde{\Sigma})\right)\end{aligned}$$

where

$$\widetilde{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}(x_i-\mu)(x_i-\mu)^\top$$

is the empirical covariance matrix.

Let $f : A \mapsto \frac{n}{2}\text{Trace}\left(A\widetilde{\Sigma}\right)$.

We have :

$$f(A+H) - f(A) = \frac{n}{2}\text{Trace}\left(H\widetilde{\Sigma}\right)$$

The gradient of the last term of ℓ is then :

$$\nabla f(A) = \frac{n}{2}\widetilde{\Sigma}$$

and :

$$\begin{aligned}df_A(H) &= \langle \nabla f(A), H\rangle\\ &= \text{Trace}\left(\nabla f(A)^\top H\right)\end{aligned}$$

Let us focus on the second term. We have :

$$\begin{aligned}\log\left(\det(A+H)\right) &= \log\left(|A^{\frac{1}{2}}\left(I + A^{-\frac{1}{2}}HA^{-\frac{1}{2}}\right)A^{-\frac{1}{2}}|\right)\\ &= \log\left(|A|\right) + \log\left(\det(I+\widetilde{H})\right)\end{aligned}$$

where $\widetilde{H} = \left(A^{-\frac{1}{2}}\right)HA^{-\frac{1}{2}}$.

Let $g : A \mapsto \log\left(\det(A)\right)$ where $A = I + \widetilde{H}$.

We have :

$$\begin{aligned}\log\left(\det(I+\widetilde{H})\right)-\log\left(\det(I)\right) &= \sum_{i=1}^{d}\log(1+\lambda_j)\\ &\simeq \sum_{i=1}^{d}\lambda_j+o\left(||\widetilde{H}||\right)\\ &= \text{Trace}\left(\widetilde{H}\right)+o\left(||\widetilde{H}||\right)\end{aligned}$$

$\widetilde{H}$ is symmetric, so it can be written as :

$$\widetilde{H}=U\Lambda U^{\top}$$

where U is an orthogonal matrix and $\Lambda=\text{diag}(\lambda_1,...,\lambda_d)$.

$$\text{d}\left(\log\left(\det{}_A(H)\right)\right)=\text{Trace}\left(A^{-\frac{1}{2}}HA^{-\frac{1}{2}}\right)=\text{Trace}(HA^{-1})$$

We deduce the gradient of $\log\left(\det(A)\right)$:

$$\nabla\log\left(\det(A)\right)=A^{-1}$$

And the gradient of ℓ w.r.t. A is :

$$\nabla_A(\ell)=-\frac{n}{2}A^{-1}+\frac{n}{2}\widetilde{\Sigma}$$

It is equal to zero iff :

$$\widehat{\Sigma}=\widetilde{\Sigma}$$

when $\widetilde{\Sigma}$ is invertible.
Finally we have shown that the pair

$$\hat{\mu}=\bar{x}=\frac{1}{n}\sum_{i=1}^{n}x_i \quad \text{and}\hat{\Sigma}=\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})(x_i-\bar{x})^{\top}$$

is the only stationary point of the likelihood. One can actually check (for example computing the Hessian w.r.t. (μ,Σ) that this actually a maximum. We will have a confirmation of this in the lecture on exponential families.

Bibliography

[1] J.M Amigo and M.B. Kennel. Variance estimators for the Lempel-Ziv entropy rate estimator. *Chaos*, 16:043102, 2006.

[2] Aim Fuchs Dominique Foata. *Calcul des probabilités*. 2ème édition. Dunod, 2003.

[3] Gilbert Saporta. *Probabilités, analyses des données et statistiques*. Technip, 1990.

[4] Frédéric Bonnans. *Optimisation continue, Cours et problèmes corrigés*. Dunod, 2003.

[5] Michael Jordan. *An introduction to graphical models*. In preparation.

[6] http://fr.wikipedia.org/wiki/multiplicateur_de_lagrange.

[7] S.J.D. Prince. *Computer Vision: Models Learning and Inference*. Cambridge University Press, 2012.

[8] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[9] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

Lecture 2 — October 11

Lecturer: Guillaume Obozinski *Scribes: Aymeric Reshef, Claire Vernade*

- Course webpage: `http://www.di.ens.fr/~fbach/courses/fall2013/`

2.1 Single node models (last part)

The previous course introduced the notion of *Maximum Likelihood Estimator* (MLE). Basic examples on Bernoulli model, multinomial model and Gaussian model were explicited, and side notes detailed the use of Lagrangian operators and of differentials. The last example was using the multivariate Gaussian model. We recall it briefly in the next subsection.

2.1.1 The Multivariate Gaussian model

If X is a random variable taking values in $\mathbb{R}^d$. Let $\mu \in \mathbb{R}^d$ and $\Sigma \in \mathbb{R}^{d\times d}$ be a positive definite matrix. X follows a *multivariate Gaussian model* (denoted by $X \sim \mathcal{N}(\mu, \Sigma)$) if

$$p_{\mu,\Sigma}(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^d \det \Sigma}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^\top \Sigma^{-1}(\mathbf{x}-\mu)\right).$$

Let $X_1, \cdots, X_n \sim \mathcal{N}(\mu, \Sigma)$, *iid*. Then, the negative *log-likelihood* of the joint distribution is

$$\begin{aligned} -l(\mu,\Sigma) &= -\sum_{i=1}^n \log p_{\mu,\Sigma}(\mathbf{x}_i) \\ &= \frac{nd}{2}\log(2\pi) + \frac{n}{2}\log(\det\Sigma) + \frac{1}{2}\sum_{i=1}^n (\mathbf{x}_i-\mu)^\top \Sigma^{-1}(\mathbf{x}_i-\mu). \end{aligned}$$

Its gradient with respect to μ is given by

$$\begin{aligned} -\nabla_\mu l(\mu,\Sigma) &= \sum_{i=1}^n \Sigma^{-1}(\mathbf{x}_i-\mu) \\ &= \Sigma^{-1}\left(\sum_{i=1}^n x_i - n\mu\right) = \Sigma^{-1}(n\bar{x}-n\mu), \end{aligned}$$

which leads to $\boxed{\hat{\mu} = \frac{1}{n}\bar{x}}$, the empirical mean.

In order to compute the gradient with respect to Σ, we first write $A = \Sigma^{-1}$, so that

$$\begin{aligned} -l(\mu,\Sigma) &= \frac{nd}{2}\log(2\pi) - \frac{n}{2}\log(\det A) + \frac{1}{2}\sum_{i=1}^n (\mathbf{x}_i-\mu)^\top A(\mathbf{x}_i-\mu) \\ &= \frac{nd}{2}\log(2\pi) - \frac{n}{2}\log(\det A) + \frac{n}{2}\mathrm{Tr}(A\tilde{\Sigma}), \end{aligned}$$

where we introduced the empirical covariance matrix $\tilde{\Sigma}$ defined as

$$\tilde{\Sigma} = \frac{1}{n}\sum_{i=1}^{n} (\mathbf{x}_i - \mu)^\top (\mathbf{x}_i - \mu).$$

The matrix A appears in the expression of the log-likelihood in two terms: $\frac{n}{2}\log\det A$ and $\frac{n}{2}\operatorname{Tr}(A\tilde{\Sigma})$.
Denote by $f(A) = \operatorname{Tr}(A\tilde{\Sigma})$. Then $f(A+H) - f(A) = \operatorname{Tr}(H\tilde{\Sigma})$, which leads to $\nabla f(A) = \tilde{\Sigma}$. Now, write $\log\det A$ as

$$\log\det(A+H) = \log\det\left(A^{\frac{1}{2}}\left(I + A^{-\frac{1}{2}}HA^{-\frac{1}{2}}\right)A^{\frac{1}{2}}\right) = \log\det A + \log\det(I+\tilde{H})$$

where $A^{\frac{1}{2}}$ stands for the square root matrix of A (it exists, since A is positive definite) and $\tilde{H} = A^{-\frac{1}{2}}HA^{-\frac{1}{2}}$. Let's see how $\log\det(I+\tilde{H})$ looks like. Noting that $\log\det I = 0$, and denoting by $(\lambda_1, \cdots, \lambda_d)$ the eigenvalues of $\tilde{H}$, we have that

$$\log\det(I+\tilde{H}) = \log\det(I+\tilde{H}) - \log\det I = \sum_{j=1}^{d}\log(1+\lambda_j) \approx \sum_{j=1}^{d}\lambda_j + o(\|\tilde{H}\|).$$

But then,

$$\sum_{j=1}^{d}\lambda_j = \operatorname{Tr}(\tilde{H}) = \operatorname{Tr}(A^{-\frac{1}{2}}HA^{-\frac{1}{2}}) = \operatorname{Tr}(HA^{-1}).$$

We conclude that $\nabla_A \log\det A = A^{-1}$.

Plugging these results into the gradient of the log-likelihood with respect to A, we have

$$\nabla_A l(A) = -\frac{n}{2}A^{-1} + \frac{n}{2}\tilde{\Sigma}.$$

The optimality condition $\nabla_A l(A)$ leads to $A^{-1} = \tilde{\Sigma}$, which means that

$$\boxed{\hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \mu)^\top(\mathbf{x}_i - \mu)}$$

is the empirical covariance matrix.

Note that we assumed that A was invertible, which is an implicit condition when writing $\log\det A$. This implies that in a rigorous sense the maximum likelihood estimator is undefined when $\tilde{\Sigma}$ is not invertible. In practice, the MLE is extended by continuity to the rank deficient case.

2.2 Models with two nodes

In this section, we work with two nodes: one node corresponds to an input X, and one node corresponds to an output Y.

Recall that when dealing with two random variables X and Y, one can use a *generative* model, i.e. which models the joint distribution $p(X,Y)$, or one can use instead a *conditional* model (often considered equivalent to the slightly different concept of *discriminative model*), which models the conditional probability of the output, given the input $p(Y|X)$. The two following models, *linear regression* or a *logistic regression*, are *conditional models*.

2.2.1 Linear regression

Let's assume that $Y \in \mathbb{R}$ depends linearly on $X \in \mathbb{R}^p$. Let $w \in \mathbb{R}^p$ be a weighting vector and $\sigma^2 > 0$. We make the following assumption:

$$Y \mid X \sim \mathcal{N}(\mathbf{w}^\top X, \sigma^2),$$

which can be rewritten as

$$Y = \mathbf{w}^\top X + \epsilon,$$

with $\epsilon \sim \mathcal{N}(0, \sigma^2)$. Note that if there is an offset $w_0 \in \mathbb{R}^p$, that is, if $Y = \mathbf{w}^\top X + w_0 + \epsilon$, one can always redefine a weighting vector $\tilde{\mathbf{w}} \in \mathbb{R}^{p+1}$ such that

$$Y = \tilde{\mathbf{w}}^\top \begin{pmatrix} x \\ 1 \end{pmatrix} + \epsilon.$$

Let $\mathcal{D} = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$ be a training set of i.i.d. random variables. Each y_i is a *label* (a decision) on observation $\mathbf{x}_i$. We consider the *conditional* distribution of all outputs given all inputs, which is a product of terms because of the independence of the pairs forming the training set:

$$p(y_1, \cdots, y_n | x_1, \cdots, x_n; \mathbf{w}, \sigma^2) = \prod_{i=1}^{n} p(y_i | \mathbf{x}_i; \mathbf{w}, \sigma^2).$$

The associated log-likelihood has the following expression:

$$-l(\mathbf{w}, \sigma^2) = -\sum_{i=1}^{n} \log p(y_i | \mathbf{x}_i) = \frac{n}{2} \log(2\pi\sigma^2) + \frac{1}{2} \sum_{i=1}^{n} \frac{(y_i - \mathbf{w}^\top \mathbf{x}_i)^2}{\sigma^2}.$$

The minimization problem with respect to w can now be reformulated as:

$$\text{find} \quad \hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \tfrac{1}{2n} \textstyle\sum_{i=1}^{n} (y_i - \mathbf{w}^\top \mathbf{x}_i)^2.$$

Define the so-called *design matrix* $\mathbf{X}$ as

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^\top \\ \vdots \\ \mathbf{x}_n^\top \end{pmatrix} \in \mathbb{R}^{n \times p}$$

and denote by $\mathbf{y}$ the vector of coordinates $(y_1, \cdots, y_n)$. The minimization problem over w can be rewritten in a more compact way as:

$$\text{find} \quad \hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \tfrac{1}{2n} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2.$$

Let $f : w \mapsto \frac{1}{2n}\|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2 = \frac{1}{2n}(\mathbf{y}^\top\mathbf{y} - 2\mathbf{w}^\top\mathbf{X}^\top\mathbf{y} + \mathbf{w}^\top\mathbf{X}^\top\mathbf{X}\mathbf{w}$. f is strictly convex if and only if its Hessian matrix is invertible. This is never the case when $n < p$ (in this case, we deal with underdetermined problems). Most of the time, the Hessian matrix is invertible when $n \geq p$. When this is not the case, we often use the Tikhonov regularization, which adds

a penalization of the ℓ_2-norm of w by minimizing $f(\mathbf{w}) + \lambda\|\mathbf{w}\|^2$ with some hyperparameter $\lambda > 0$.

The gradient of f is

$$\nabla f(\mathbf{w}) = \frac{1}{n}\mathbf{X}^\top(\mathbf{X}\mathbf{w} - \mathbf{y}) = 0 \iff \mathbf{X}^\top\mathbf{X}\mathbf{w} = \mathbf{X}^\top\mathbf{y}.$$

The equation $\boxed{\mathbf{X}^\top\mathbf{X}\mathbf{w} = \mathbf{X}^\top\mathbf{y}}$ is known as the *normal equation*. If $\mathbf{X}^\top\mathbf{X}$ is invertible, then the optimal weighting vector is

$$\boxed{\hat{\mathbf{w}} = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y} = \mathbf{X}^\dagger\mathbf{y}}$$

where $\mathbf{X}^\dagger = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top$ is the *Moore-Penrose pseudo-inverse* of X. If $\mathbf{X}^\top\mathbf{X}$ is not invertible, the solution is not unique anymore, and for any $\mathbf{h} \in \ker(\mathbf{X})$, $\hat{\mathbf{w}} = (\mathbf{X}^\top\mathbf{X})^\dagger\mathbf{X}^\top\mathbf{y} + \mathbf{h}$ is an admissible solution. In that case however it would be necessary to use regularization.

The computational cost to evaluate the optimal weighting vector from $\mathbf{X}$ and $\mathbf{y}$ is $O(p^3)$ (use a Cholesky decomposition of matrix $\mathbf{X}^\top\mathbf{X}$ and solve two triangular systems).

Now, let's differentiate $l(\mathbf{w}, \sigma^2)$ with respect to σ^2: we have

$$\nabla_{\sigma^2} l(\mathbf{w}, \sigma^2) = \frac{n}{2\sigma^2} - \frac{n}{2\sigma^4}\frac{1}{n}\sum_{i=1}^{n}(y_i - \mathbf{w}^\top\mathbf{x}_i)^2.$$

Setting $\nabla_{\sigma^2} l(\mathbf{w}, \sigma^2)$ to zero gives

$$\boxed{\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - \mathbf{w}^\top\mathbf{x}_i)^2.}$$

In practice, whenever we use a data matrix $\mathbf{X}$ in machine learning, we first preprocess it to try and avoid that it would be too badly conditioned, so to avoid numerical issues. Two main operations are applied columnwise: first, a centering (remove the mean of the coefficients) and a normalization (divide coefficients from a column by the standard deviation of the column vector). Note that this preprocessing *does not guarantee* that the matrix we obtain is well-conditioned: in particular, it can be low rank...

2.2.2 Logistic regression

Let $X \in \mathbb{R}^p$, $Y \in \{0, 1\}$. We assume that Y follows a Bernoulli distribution with parameter θ. The problem is to find θ. Let's define the *sigmoid* function σ defined on the real axis and taking values in $[0, 1]$, such that

$$\forall z \in \mathbb{R}, \sigma(z) = \frac{1}{1 + e^{-z}}.$$

The sigmoid function is plot on Figure 2.1.

One can easily prove that

$$\begin{aligned} \forall z \in \mathbb{R}, \quad \sigma(-z) &= 1 - \sigma(z), \\ \forall z \in \mathbb{R}, \quad \sigma'(z) &= \sigma(z)(1 - \sigma(z)) = \sigma(z)\sigma(-z). \end{aligned}$$

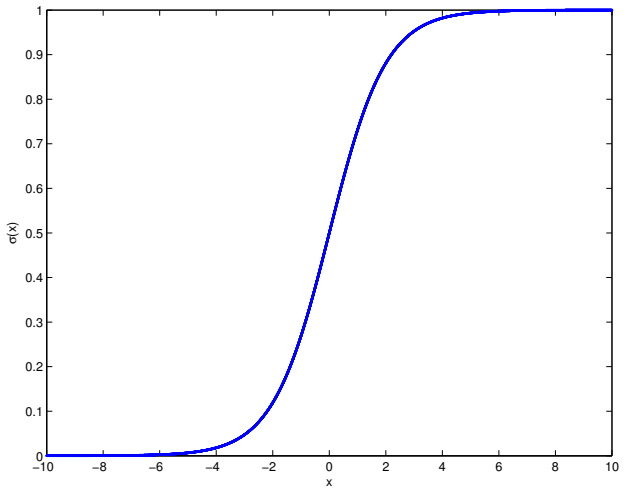


Figure 2.1. Sigmoid function.

We now assume that, for a given observation $X = \mathbf{x}$, the output $Y|X = \mathbf{x}$ follows a Bernoulli law with parameter $\theta = \sigma(\mathbf{w}^\top \mathbf{x})$, where w is again a weighting vector. In practice, we still can add an offset $\mathbf{w}^\top \mathbf{x} + w_0$. Then, the conditional distribution is given by

$$p(Y = y|X = \mathbf{x}) = \theta^y (1-\theta)^{1-y} = \sigma(\mathbf{w}^\top \mathbf{x})^y \sigma(-\mathbf{w}^\top \mathbf{x})^{1-y}.$$

Given a training set $\mathcal{D} = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$ of iid random variables , we can compute the log-likelihood

$$l(\mathbf{w}) = \sum_{i=1}^{n} y_i \log \sigma(\mathbf{w}^\top \mathbf{x}_i) + (1 - y_i) \log \sigma(-\mathbf{w}^\top \mathbf{x}_i).$$

In order to minimize the log-likelihood, since $z \mapsto \log(1 + e^{-z})$ is a convex function and $\mathbf{w} \mapsto \mathbf{w}^\top \mathbf{x}_i$ is linear, we calculate its gradient. We write $\eta_i = \sigma(\theta^\top \mathbf{x}_i)$:

$$\nabla_{\mathbf{w}} l(\mathbf{w}) = \sum_{i=1}^{n} y_i \mathbf{x}_i \frac{\sigma(\mathbf{w}^\top \mathbf{x}_i)\sigma(-\mathbf{w}^\top \mathbf{x}_i)}{\sigma(\mathbf{w}^\top \mathbf{x}_i)} - (1 - y_i)\mathbf{x}_i \frac{\sigma(\mathbf{w}^\top \mathbf{x}_i)\sigma(-\mathbf{w}^\top \mathbf{x}_i)}{\sigma(-\mathbf{w}^\top \mathbf{x}_i)} = \sum_{i=1}^{n} \mathbf{x}_i (y_i - \eta_i)$$

Thus, $\nabla_{\mathbf{w}} l(\mathbf{w}) = 0 \iff \sum_{i=1}^{n} \mathbf{x}_i (y_i - \sigma(\theta^\top \mathbf{x}_i)) = 0$. This equation is nonlinear and we need an iterative optimization method to solve it. For this purpose, we derive the Hessian matrix of l:

$$Hl(\mathbf{w}) = \sum_{i=1}^{n} \mathbf{x}_i (0 - \sigma'(\mathbf{w}^\top \mathbf{x}_i)\sigma'(-\mathbf{w}^\top \mathbf{x}_i)\mathbf{x}_i^\top)$$

$$= \sum_{i=1}^{n} (-\eta_i(1 - \eta_i))\mathbf{x}_i \mathbf{x}_i^\top = -\mathbf{X}^\top \mathrm{Diag}(\eta_i(1 - \eta_i))\mathbf{X}$$

where $\mathbf{X}$ is the design matrix defined previously.

In the following we discuss first- and second-order optimization methods and apply them to logistic regression.

First-order methods

Let $f : \mathbb{R}^p \to \mathbb{R}$ be the convex C^1 function that we want to minimize. A *descent direction* at point $\mathbf{x}$ is a vector d such that $\langle \mathbf{d}, \nabla f(\mathbf{x}) \rangle < 0$. The minimization of f can be done by applying a *descent algorithm*, which iteratively takes a step in a descent direction, leading to an iterative scheme of the form

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \varepsilon^{(k)} \mathbf{d}^{(k)},$$

where $\varepsilon^{(k)}$ is the *stepsize*. The direction $\mathbf{d}^{(k)}$ is often chosen as the opposite of the gradient of f at point $\mathbf{x}^{(k)}$: $\mathbf{d}^{(k)} = -\nabla f(\mathbf{x}^{(k)})$.

There are several choices for $\varepsilon^{(k)}$:

1. Constant step: $\varepsilon^{(k)} = \varepsilon$. But the scheme does not necessarily converge.
2. Decreasing step size: $\varepsilon^{(k)} \propto \frac{1}{k}$ with $\sum_k \varepsilon^{(k)} = \infty$ and $\sum_k (\varepsilon^{(k)})^2 < \infty$. The scheme is guaranteed to converge.
3. One can determine $\varepsilon^{(k)}$ by doing a *Line Search* which tries to find $\min_\varepsilon f(\mathbf{x}^{(k)} + \varepsilon \mathbf{d}^{(k)})$:
   - either exactly but this is costly and rather useless in many situations;
   - or approximately (see the Armijo linesearch). This is a better method.

Second-order methods

This time, let $f : \mathbb{R}^p \to \mathbb{R}$ be the C^2 function that we want to minimize. We write the second-order Taylor-expansion of f:

$$f(\mathbf{x}) = f(\mathbf{x}^t) + (\mathbf{x} - \mathbf{x}^t)^\top \nabla f(\mathbf{x}^t) + \frac{1}{2}(\mathbf{x} - \mathbf{x}^t)^\top Hf(\mathbf{x}^t)(\mathbf{x} - \mathbf{x}^t) + o(\left\|\mathbf{x} - \mathbf{x}^t\right\|^2) \stackrel{def}{=} g_t(\mathbf{x}) + (\left\|\mathbf{x} - \mathbf{x}^t\right\|^2)$$

A local optimum $\mathbf{x}^*$ is then reached when

$$\begin{cases} \nabla f(\mathbf{x}^*) = 0 \\ H(f(\mathbf{x}^*) \succeq 0 \end{cases}$$

In order to solve such a problem, we are going to use *Newton's method*. If f is a convex function, then $\nabla g_t(\mathbf{x}) = \nabla f(\mathbf{x}^t) + Hf(\mathbf{x}^t)(\mathbf{x} - \mathbf{x}^t)$ and we only need to find $\mathbf{x}^*$ so that $\nabla g_t(\mathbf{x}) = 0$, *ie.* we set $\mathbf{x}^{t+1} = \mathbf{x}^t - [Hf(\mathbf{x}^t]^{-1} \nabla f(\mathbf{x}^t)$. If the Hessian matrix is not invertible, we can regularize the problem and minimize $g_t(\mathbf{x}) + \lambda \|\mathbf{x} - \mathbf{x}^t\|^2$ instead.

In general the previous update, called the *Pure Newton step* does not lead to a convergent algorithm even if the function is convex!

In general it is necessary to use the so-called *Damped Newton method*, to obtain a convergent algorithm which consists in doing the following iterations:

$$\mathbf{x}^{t+1} = \mathbf{x}^t - \varepsilon^t (Hf(\mathbf{x}^t))^{-1} \nabla f(\mathbf{x}^t),$$

where ε^t is set with the Armijo *Line Search*

This method may be computationally costly in high dimension because of the inverse of the hessian matrix that needs to be computed at each iteration. For some functions, however, the pure Newton's method does converge. This is the case for logistic regression.

In the context of non-convex optimization, the situation is more complicated because the Hessian can have negative eigenvalues. In that case, so-called trust region methods are typically used.

Application to logistic regression

We will write the form that Newton's algorithm takes for logistic regression. We had :

$$\begin{aligned} l(\mathbf{w}) &= \sum_{i=1}^{n} y_i \log \sigma(\mathbf{w}^\top \mathbf{x}_i) + (1-y_i)\log \sigma(-\mathbf{w}^\top \mathbf{x}_i) \\ \nabla_{\mathbf{w}} l(\mathbf{w}) &= \sum_{i=1}^{n} \mathbf{x}_i (y_i - \eta_i) = \mathbf{X}^\top (\mathbf{y} - \boldsymbol{\eta}) \\ Hl(\mathbf{w}) &= -\mathbf{X}^\top \mathrm{Diag}(\eta_i(1-\eta_i))\mathbf{X} \end{aligned}$$

The second-order Taylor expansion of the loss function leads to

$$l(\mathbf{w}) = l(\mathbf{w}^t) + (\mathbf{w}-\mathbf{w}^t)^\top \nabla l(\mathbf{w}^t) + \frac{1}{2}(\mathbf{w}-\mathbf{w}^t)^\top Hl(\mathbf{w}^t)(\mathbf{w}-\mathbf{w}^t).$$

Let us set $\mathbf{h} = \mathbf{w} - \mathbf{w}^t$. The minimization problem becomes:

$$\min_{\mathbf{h}} \left\{ \mathbf{h}^\top \mathbf{X}^\top (\mathbf{y} - \boldsymbol{\eta}) - \frac{1}{2}\mathbf{h}^\top \mathbf{X}^\top \mathrm{Diag}(\eta(1-\eta))\mathbf{X}\mathbf{h} \right\} \iff \min_{\mathbf{h}} \mathbf{h}^\top \nabla_{\mathbf{w}} l(\mathbf{w}) + \frac{1}{2}\mathbf{h}^\top Hl(\mathbf{w})\mathbf{h}.$$

This leads, according to the previous part, to set $\mathbf{w}^{t+1} = \mathbf{w}^t + Hl(\mathbf{w}^t)^{-1}\nabla_{\mathbf{w}} l(\mathbf{w})$. The minimization problem above can be seen as some *weighted* linear regression over h of some function of the form $\sum_i \frac{(\tilde{y}_i - \tilde{\mathbf{x}_i}^\top \mathbf{h})^2}{\sigma_i^2}$, where $\tilde{y}_i = y_i - \eta_i$ and $\sigma_i^2 = [\eta_i(1-\eta_i)]^{-1}$. Thus, this method is often refered as the *iterative reweighted least squares* algorithm (IRLS).

We may run into a classification problem with more than two classes : $Y \in \{1, \cdots, K\}$ with $Y \sim \mathcal{M}(1, \pi_1(\mathbf{x}), \cdots, \pi_K(\mathbf{x}))$ where We will need to define a rule over the classifiers (softmax function, one-versus-all, etc.) in order to make a decision.

2.2.3 Generative models

This section briefly presents the *Fisher linear discriminant* also known as the linear discriminant analysis. Suppose that we have $X \in \mathbb{R}^p$ and $Y \in \{0, 1\}$.

$$P(Y=1 \mid X=\mathbf{x}) = \frac{P(X=\mathbf{x} \mid Y=1)P(Y=1)}{P(X=\mathbf{x} \mid Y=1)P(Y=1) + P(X=\mathbf{x} \mid Y=0)P(Y=0)}$$

The assumption then consists in considering $P(X=\mathbf{x} \mid Y=1) \sim \mathcal{N}(\mathbf{x}, \mu_1, \Sigma_1)$ and $P(X=\mathbf{x} \mid Y=0) \sim \mathcal{N}(\mathbf{x}, \mu_0, \Sigma_0)$. Fisher's assumption is the assumption that $\Sigma_1 = \Sigma_0 = \Sigma$.

2.3 Unsupervised classification

Unsupervised learning consists in finding a label prediction function based on unlabeled training data only. In the case where the learning problem is a classification problem, and under the assumption that the classes form clusters in input space, the problem reduces to a clustering problem, which consists in finding groups of points that form denser clusters. When the clusters are assumed to be isotropic the formulation of the K-means algorithm is appropriate.

The K-means algorithm

We start from a set of data points $(\mathbf{x}_1, \cdots, \mathbf{x}_n)$ (where $\mathbf{x}_i \in \mathbb{R}^p$), that are unlabelled. We wish to divide this set into K clusters defined by their centroids $(\mu_1, \cdots, \mu_K)$. The problem can be formulated as:

$$\min_{\mu_1,\cdots,\mu_K} \frac{1}{n} \sum_{i=1}^{n} \min_k \|\mathbf{x}_i - \mu_k\|^2.$$

The minimization step inside the summation leads to a nonconvex problem. The K-means algorithm is a greedy algorithm which consists in iteratively apply two steps:

$$\begin{aligned} C_k &\leftarrow \left\{ i \mid \|\mathbf{x}_i - \mu_k\|^2 = \min_j \|\mathbf{x}_i - \mu_j\|^2 \right\} \\ \mu_k &\leftarrow \frac{1}{\mid C_k \mid} \sum_{i \in C_k} \mathbf{x}_i. \end{aligned}$$

The first step defines the clusters C_k by assigning each data point to its closest centroid. The second step then updates the centroids given the new cluster.

Two remarks:

- It can be shown that K-means converges in a finite number of steps.
- The algorithm however typically get stuck in local minima and it practice it is necessary to try several restarts of the algorithm with a random initialization to have chances to obtain a better solution.

K-means, EM, Gaussian Mixture, Graph Theory **2013/2014**

Lecture 3 — October 16th

Lecturer: Francis Bach *Scribe: Marie d'Autume, Jean-Baptiste Alayrac*

To talk about estimation of "hidden" parameters, French speaking people and English speaking people use different terms which can lead to some confusions. Within a supervised framework, English people would prefer to use the term *classification* whereas the French use the term *discrimination*. Within an unsupervised context, English people would rather use the term *clustering*, whereas French people would use *classification* or *classification non-supervisée*. In the following we will only use the English terms.

3.1 K-means

K- means clustering is a method of vector quantization. K-means clustering is an algorithm of alternate minimization that aims at partitioning n observations into K clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype to the cluster (see Figure 3.1).

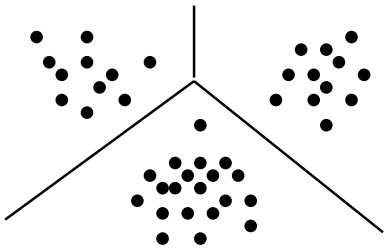


Figure 3.1. Clustering on a 2D point data set with 3 clusters.

3.1.1 Notations and notion of Distortion

We will use the following notations:

- $x_i \in \mathbb{R}^p$, $i \in \{1, ..., n\}$ are the observations we want to partition.
- $\mu_k \in \mathbb{R}^p$, $k \in \{1, ..., K\}$ are the means where μ_k is the center of the cluster k. We will denote μ the associated matrix.
- z_i^k are indicator variables associated to x_i such that $z_i^k = 1$ if x_i belongs to the cluster k, $z_i^k = 0$ otherwise. z is the matrix which components are equal to z_i^k.

Finally, we define the *distortion* $J(\mu, z)$ by:

$$J(\mu, z) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_i^k \|x_i - \mu_k\|^2.$$

3.1.2 Algorithm

The aim of the algorithm is to minimize $J(\mu, z)$. To do so we proceed with an alternating minimization :

- Step 0 : We choose a vector μ
- Step 1 : we minimize J with respect to z : $z_i^k = 1$ if $\| x_i - \mu_k \|^2 = \min_s \| x_i - \mu_s \|^2$, in other words we associate to x_i the nearest center μ_k.
- Step 2 : we minimize J with respect to μ : $\mu_k = \frac{\sum_i z_i^k x_i}{\sum_i z_i^k}$.
- Step 3 : we come back to step 1 until convergence.

Remark 3.1.1 *The step of minimization with respect to z is equivalent to allocating the x_i in the Voronoi cells which centers are the μ_k.*

Remark 3.1.2 *During the step of minimization with respect to μ, μ_k is obtained by setting to zero the k-th coordinate of the gradient of J with respect to μ. Indeed we can easily see that :*

$$\nabla_{\mu_k} J = -2 \sum_i z_i^k (x_i - \mu_k)$$

3.1.3 Convergence and Initialization

We can show that this algorithm converges in a finite number of iterations. Therefore the convergence could be local, thus it introduces the problem of initialization.

A classic method is use of random restarts. It consists in choosing several random vectors μ, computing the algorithm for each case and finally keeping the partition which minimizes the distortion. Thus we hope that at least one of the local minimum is close enough to a global minimum.

One other well known method is the K-means++ algorithm, which aims at correcting a major theoretic shortcomings of the K-means algorithm : the approximation found can be arbitrarily bad with respect to the objective function compared to the optimal clustering.

The K-means++ algorithm addresses this obstacles by specifying a procedure to initialize the cluster centers before proceeding with the standard K-means optimization iterations. With the K-means ++ initialization, the algorithm is guaranteed to find a solution that is $O(\log K)$ competitive to the optimal K-means solution.

The intuition behind this approach is that it is a clever thing to well spread out the K initial cluster centers. At each iteration of the algorithm we will build a new center. We will repeat the algorithm until we have K centers. Here are the steps of the algorithm :

- Step 0 : First initiate the algorithm by choosing the first center uniformly at random among the data points.
- Step 1: For each data point x_i of your data set, compute the distance between x_i and the nearest center that has already been chosen. We denote this distance $D_{\mu_t}(x_i)$ where μ_t is specified to recall that we are minimizing over the current chosen centers.
- Step 2: Choose one new data point at random as a new center, but now using a weighted probability distribution where a point x_i is chosen with probability proportional to $D_{\mu_t}(x_i)^2$.
- Step 3 : Repeat Step 1 and Step 2 until K centers have been chosen.

We see that we have now built K vectors with respect to our first intuition which was to well spread out the centers (because we used a well chosen weighted probability). We can now use those vectors as the initialization of our standard K-means algorithm.

More details can be found on the K-means++ algorithm in [A].

[A] Arthur, D. and Vassilvitskii, S. (2007). k-means++: the advantages of careful seeding. Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms.

3.1.4 Choice of K

It is important to point out that the choice of K is not universal. Indeed, we see that if we increase K, the distortion J decreases, until it reaches 0 when $K = n$, that is to say when each data point is the center of its own center. To address this issue one solution could be to add to J a penalty over K. Usually it takes the following form :

$$J(\mu, z, K) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_i^k \|x_i - \mu_k\|^2 + \lambda K$$

But again the choice of the penalty is arbitrary.

3.1.5 Other problems

We can also point out that K-means will work pretty well when the width of the different clusters are similar, for example if we deal with spheres. But clustering by K-means could also be disappointing in some cases such as the example given in Figure 3.2.

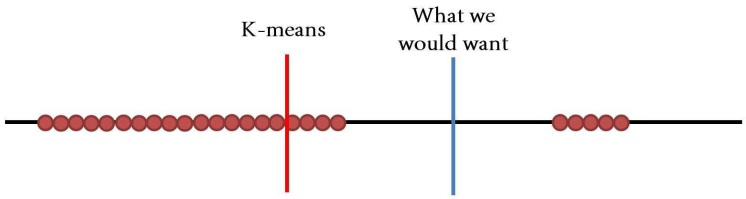


Figure 3.2. Example where K- means does not provide a satisfactory clustering result

Using Gaussian mixtures provides a way to avoid this problem (see next section).

3.2 EM : Expectation Maximization

The Expectation-maximization (EM) algorithm is an iterative method for finding maximum likelihood estimates of parameters in statistical models, where the models depend on unobserved latent or hidden variables z. Latent variables are variables that are not directly observed but are rather inferred from other variables that are observed.

Previous algorithms aimed at estimating the parameter θ that maximized the likelihood of $p_\theta(x)$, where x is the vector of observed variables.

Here it is a little bit different. Indeed we have now :

Assumption : (x, z) are random variables where x is observed (our data) and z is non observed (unknown cluster center for example).

$$p_\theta(x, z) : \text{joint density depending on a parameter } \theta \text{ (the model)}$$

The goal : to maximize the following probability :

$$\max_\theta p_\theta(x) = \sum_z p_\theta(x, z).$$

We can already infer that, because of the sum, the problem should be slightly more difficult than before. Indeed, taking the log of our probability would not lead to a simple convex problem. In the following we will see that EM is a method to solve those kind s of problems.

3.2.1 Example

Let's present a simple example to illustrate what we just said. The probability density represented on Figure 3.2.1 is akin to an average of two Gaussians. Thus, it is natural to use a mixture model and to introduce an hidden variable z, following a Bernoulli distribution defining which Gaussian the point is sampled from.

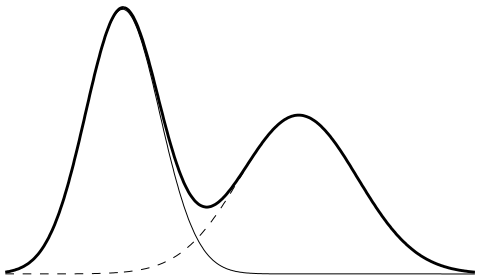


Figure 3.3. Average of two probability distributions of two Gaussian for which it is natural to introduce a mixture model

Thus we have : $z \in \{1, 2\}$ and $x|z = i \sim \mathcal{N}(\mu_i, \Sigma_i)$. The density $p(x)$ is a convex combination of normal density:

$$p(x) = p(x, z = 1) + p(x, z = 2) = p(x|z = 1)p(z = 1) + p(x|z = 2)p(z = 2)$$

It is a mixture model. It represents a simple way to model complicated phenomena.

3.2.2 Objective: maximum likelihood

Let z be the hidden variables and x be the observed data. We make the assumption that the x_i, $i \in \{1, ..., n\}$ are i.i.d..

As we mentioned it in the introduction the aim is to maximize the likelihood

$$p_\theta(x) = \sum_z p_\theta(x, z)$$

$$\log p_\theta(x) = \log \sum_z p_\theta(x, z)$$

.

Note that in practice, we often have $(x, z) = (x_1, z_1, \ldots, x_n, z_n)$ where each pair (x_i, y_i) is i.i.d. In this situation we have $\log p_\theta(x) = \sum_{i=1}^n \log \sum_{z_i} p_\theta(x_i, z_i)$.

There is at least two ways to solve this problem:

1. By a direct way, if we can, by a gradient ascent for example.
2. By using the EM algorithm.

3.2.3 Jensen's Inequality

We will use the following properties :

1. if $f : \mathbb{R} \to \mathbb{R}$ is convex and if X is an integrable random variable :
$$\mathbb{E}_X(f(X)) \geq f(\mathbb{E}_X(X))$$
2. if $f : \mathbb{R} \to \mathbb{R}$ is strictly convex, we have equality in the previous inequality if and only if $X =$ constant a.s.

3.2.4 EM algorithm

We introduce the function $q(z)$ such that $q(z) \geq 0$ and $\sum_z q(z) = 1$ in the expression of the likelihood. Thus we have :

$$\begin{aligned}
\log p_\theta(x) &= \log \sum_z p_\theta(x,z) \\
&= \log \sum_z \left(\frac{p_\theta(x,z)}{q(z)} \right) q(z) \\
&\geq \sum_z q(z) \log \frac{p_\theta(x,z)}{q(z)}, \text{ by the Jensen's inequality because log is concave} \\
&= \sum_z q(z) \log p_\theta(x,z) - \sum_z q(z) \log q(z) \\
&= \mathcal{L}(q,\theta)
\end{aligned}$$

with equality iff $q(z) = \frac{p_\theta(x,z)}{\sum_{z'} p_\theta(x,z')} = p_\theta(z|x)$ (by strict concavity of the logarithm).

Proposition 3.1 *$\forall\theta$, $\forall q$ $\log p_\theta(x) \geq \mathcal{L}(q,\theta)$ with equality if and only if $q(z) = p_\theta(z|x)$.*

Remark 3.2.1 *We have introduced an auxiliary function $\mathcal{L}(q,\theta)$ that is always below the function $\log(p_\theta(x))$*

EM algorithm is an algorithm of alternate maximization with respect to q and θ.

We initialize θ_0, then we iterate for $t > 0$, by alternating the following steps until convergence:

- $q_{t+1} \in \arg\max_q(\mathcal{L}(q,\theta_t))$
- $\theta_{t+1} \in \arg\max_\theta(\mathcal{L}(q_{t+1},\theta))$

Algorithm properties

- It is an ascent algorithm, indeed it goes up in term of likelihood (compare to before where we were descending along the distortion) :

$$\forall t \log(p_{\theta_t}) \geq \log(p_{\theta_{t-1}})$$

- The sequence of log-likelihoods converges.
- It does not converge to a global maximum but rather to a local maximum because we are dealing here with a non-convex problem. An illustration is given in Figure 3.4.

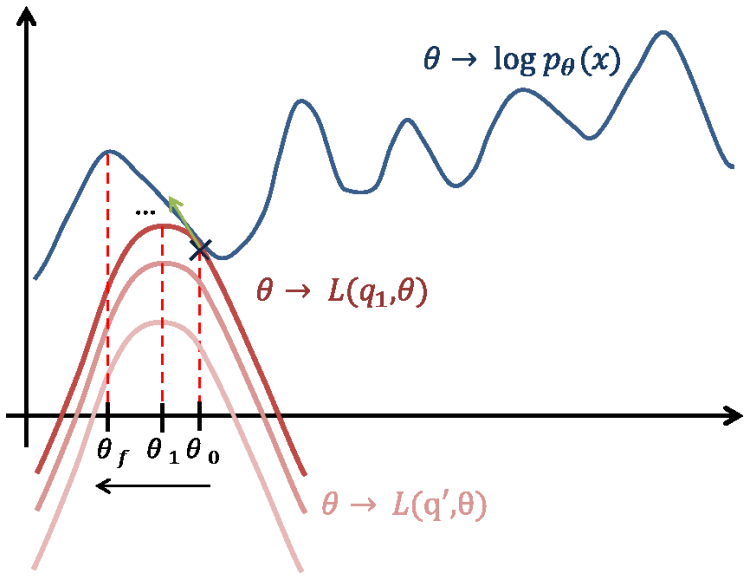


Figure 3.4. An illustration of the EM algorithm that converges to a local minimum.

- As it was already the case for K-means, we reiterate the result in order to be more confident. Then we keep the one with the highest likelihood.

Initialization Because EM gives a local maximum, it is clever to choose a θ_0 relatively close to the final solution. For Gaussian mixtures, it is quite usual to initiate EM by a K-means. The solution of K-means gives the θ_0 and a large variance is used.

The EM recipe Let's recall the initial goal of the algorithm. The goal is to maximize the *incomplete* likelihood $\log(p_\theta(x))$. To do so we want to maximize the following function which is always inferior to $\log(p_\theta(x))$:

$$\mathcal{L}(q,\theta) = \sum_z q(z)\log p_\theta(x,z) - \sum_z q(z)\log q(z).$$

1. Compute the probability of Z given X : $p_{\theta_t}(z|x)$ (Corresponding to $q_{t+1} = \arg\max_q \mathcal{L}(q,\theta_t)$)
2. Write the *complete* likelihood $l_c = \log(p_{\theta_t}(x,z))$.
3. **E-Step** : calculate the expected value of the complete log likelihood function, with respect to the conditional distribution of Z given X under the current estimate of the parameter θ_t : $\mathbb{E}_{Z|X}(l_c)$.
4. **M-Step** : find θ_{t+1} by maximizing $\mathcal{L}(q_{t+1},\theta)$ with respect to θ.

3.2.5 Gaussian Mixture

Let (x_i, z_i) be a couple, for $i \in \{1,...,n\}$ with $x_i \in \mathbb{R}^p$, $z_i \sim \mathcal{M}(1,\pi_1,...,\pi_k)$ and $(x_i|z_i = j) \sim \mathcal{N}(\mu_j,\Sigma_j)$. Here we have $\theta = (\pi,\mu,\Sigma)$.

Calculation of $p_\theta(z|x)$ We write $p_\theta(x_i)$:

$$\begin{aligned} p_\theta(x_i) &= \sum_{z_i} p_\theta(x_i,z_i) = \sum_{z_i} p_\theta(x_i|z_i)p_\theta(z_i) \\ &= \sum_{j=1}^{k} p_\theta(x_i|z_i=j)p_\theta(z_i=j) \end{aligned}$$

Then we use the Bayes formula to estimate $p_\theta(z|x)$:

$$\begin{aligned} p_\theta(z_i=j|x_i) &= \frac{p_\theta(x_i|z_i=j)p_\theta(z_i=j)}{p_\theta(x_i)} \\ &(\propto \quad p_\theta(x_i|z_i=q)p_\theta(z_i=q)) \\ &= \frac{\pi_j \mathcal{N}(x_i|\mu_j,\Sigma_j)}{\sum_{j'}\pi_{j'}\mathcal{N}(x_i|\mu'_j,\Sigma'_j)} \\ &= \tau_i^j(\theta). \end{aligned}$$

We recall that $\mathcal{N}(x_i|\mu,\Sigma) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}}\exp(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu))$.

Suppose that we are at the t-th iteration of the algorithm.

Complete likelihood Let's write the complete likelihood of the problem.

$$\begin{aligned} l_{c,t} = \log p_{\theta_t}(x,z) &= \sum_{i=1}^{n} \log p_{\theta_t}(x_i, z_i) \\ &= \sum_{i=1}^{n} \log(p_{\theta_t}(z_i) p_{\theta_t}(x_i|z_i)) \\ &= \sum_{i=1}^{n} \log(p_{\theta_t}(z_i)) + \log(p_{\theta_t}(x_i|z_i)) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{k} z_i^j \log(\pi_{j,t}) + \sum_{i=1}^{n}\sum_{j=1}^{k} z_i^j \log(\mathcal{N}(x_i|\mu_{j,t}, \Sigma_{j,t})) \end{aligned}$$

where $z_i^j \in \{0,1\}$ with $z_i^j = 1$ if $z_i = j$ and 0 otherwise.

E-Step We can now write the expectation of the previous quantity with respect to the conditional distribution of Z given X. In fact it is equivalent to replace z_i^j by $\mathbb{E}_{Z|X}(z_i^j) = p_{\theta_t}(z = j|x_i) = \tau_i^j(\theta_t)$. Indeed, the other terms of the sum are constant from the point of view of the conditional probability of Z given X, and we finally obtain $\mathbb{E}_{Z|X}(l_{c,t})$. Since the value of θ_t will be fixed during the M-step, we drop the dependence on θ_t and write τ_i^j.

M-Step For the M-step, we this need to maximize:

$$\sum_{i=1}^{n}\sum_{j=1}^{k} \tau_i^j \log(\pi_{j,t}) + \sum_{i=1}^{n}\sum_{j=1}^{k} \tau_i^j \left[\log(\frac{1}{(2\pi)^{\frac{k}{2}}}) + \log(\frac{1}{|\Sigma_{j,t}|^{\frac{1}{2}}}) - \frac{1}{2}(x_i - \mu_{j,t})^T \Sigma_{j,t}^{-1}(x_i - \mu_{j,t}))\right]$$

We want to maximize the previous equation with respect to $\theta_t = (\Pi_t, \mu_t, \Sigma_t)$

As the sum is separated into two terms independent along the variables we can first maximize with respect to π_t :

$$\max_{\Pi} \sum_{j=1}^{k}\sum_{i=1}^{n} \tau_i^j \log \pi_j \quad \Rightarrow \quad \pi_{j,t+1} = \frac{\sum_{i=1}^{n} \tau_i^j}{\sum_{i=1}^{n}\sum_{j'=1}^{k} \tau_i^{j'}} = \frac{1}{n}\sum_{i=1}^{n} \tau_i^j$$

We can now maximize with respect to μ_t and Σ_t. By computing the gradient along the $\mu_{j,t}$ and along the $\Sigma_{j,t}$, we obtain :

$$\mu_{j,t+1} = \frac{\sum_i \tau_i^j x_i}{\sum_i \tau_i^j}$$

$$\Sigma_{j,t+1} = \frac{\sum_i \tau_i^j (x_i - \mu_{j,t+1})(x_i - \mu_{j,t+1})^T}{\sum_i \tau_i^j}$$

The M-step in the EM algorithm corresponds to the estimation of means step in K-means. Note that the value of τ_i^j in the expressions above are taken for the parameter values of the previous iterate, i.e., $\tau_i^j = \tau_i^j(\theta_t)$.

Possible forms for Σ_j

- isotropic: $\Sigma_j = \sigma_j^2 \mathrm{Id}$, 1 parameter, the cluster is a sphere.
- diagonal: Σ_j is a diagonal matrix, d parameters, the cluster is an ellipse oriented along the axis.
- general: Σ_j, $\frac{d(d+1)}{2}$ parameters, the cluster is an ellipse.

3.3 Graph theory

3.3.1 Graph

Definition 3.2 (graph) *A* graph *is a pair $G = (V, E)$ comprising a set V of vertices or nodes together with a set $E \subset V \times V$ of edges or arcs, which are 2-element subsets of V.*

Remark 3.3.1 *In this course we only consider graphs without self-loop.*

3.3.2 Undirected graphs

Definition 3.3 (undirected graph) *$G = (V, E)$ is an if $\forall (u, v) \in V \times V$ with $u \neq v$ we have:*

$$(u, v) \in E \iff (v, u) \in E$$

(Figure 3.5).

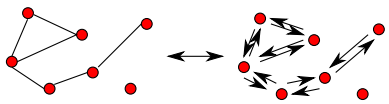


Figure 3.5. two different ways to represent an undirected graph

Definition 3.4 (neighbour) *We define $\mathcal{N}(u)$, the set of the* neighbours *of u, as*

$$\mathcal{N}(u) = \{v \in V, (v, u) \in E\}$$

(Figure 3.6).

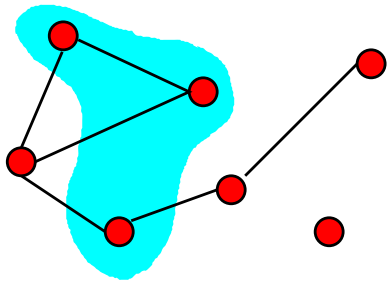


Figure 3.6. A vertex and its neighbours

Definition 3.5 (clique) *A totally connected subset of vertices or a singleton is called a* clique *(Figure 3.7).*

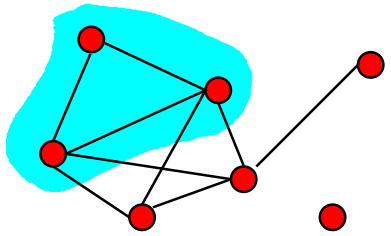


Figure 3.7. A clique.

Definition 3.6 (maximal clique) *A* maximal clique*, C, is a clique which is maximal for the inclusion order:*

$$\nexists v \in V : v \notin C \text{ and } v \cup C \text{ is a clique.}$$

(Figure 3.8).

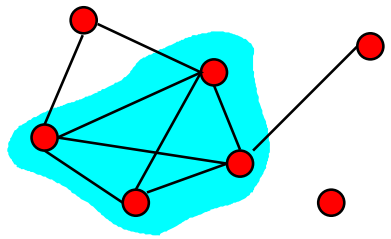


Figure 3.8. A maximal clique

Definition 3.7 (path) *A* path *is a sequence of connected vertices that are globally distinct (Figure 3.9).*

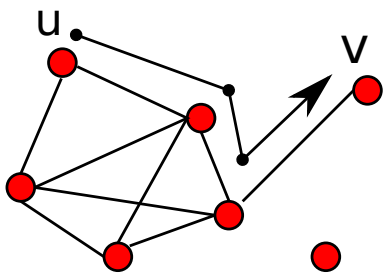


Figure 3.9. A path from u to v.

Definition 3.8 (cycle) *A* cycle *is a sequence of vertices* $(v_0, \ldots, v_k)$ *such that:*

- $v_0 = v_k$
- $\forall j, (v_j, v_{j+1}) \in E$
- $\forall i, j, v_i \neq v_j$ if $\{i, j\} \neq \{1, k\}$

Definition 3.9 *Let* A*,* B*,* C *be distinct subsets of* V*.* C separates A *and* B *if all paths from* A *to* B *go through* C *(Figure 3.10).*

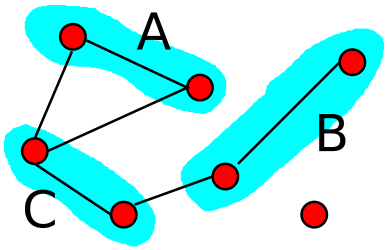


Figure 3.10. C separates A and B.

Definition 3.10 (connected component) *A* connected component *is a subgraph induced by the equivalence class of the relation* $uRv \Leftrightarrow \exists$ *path from* u *to* v *(Figure 3.11).*

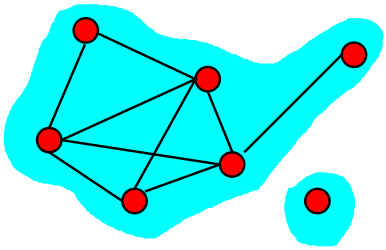


Figure 3.11. A graph with 2 connected components

In this course we will consider there is only one connected component. Otherwise we deal with them independently.

3.3.3 Oriented graphs

Definition 3.11 (parent) *v is a* parent *of u if $(v, u) \in E$*

Definition 3.12 (children) *v is a* children *of u if $(u, v) \in E$*

Definition 3.13 (ancestor) *v is an* ancestor *of u if there exists a path from u to v.*

Definition 3.14 (descendant) *v is a* descendant *of u if there exists a path from u to v*

Definition 3.15 (cycle) *A* cycle *is a sequence of vertices $(v_0, \ldots, v_k)$ (Figure 3.12) such that:*

- $v_0 = v_k$
- $\forall j, (v_j, v_{j+1}) \in E$
- $\forall i, j, v_i \neq v_j$ if $\{i, j\} \neq \{1, k\}$

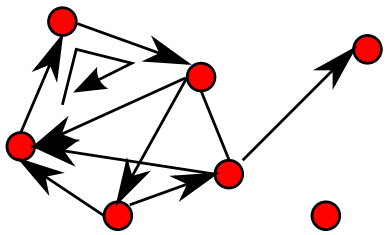


Figure 3.12. Un graphe orienté avec un cycle.

Definition 3.16 (DAG) *A* directed acyclic graph *(DAG) is a directed graph without any cycle.*

Definition 3.17 (topological order) *Let $G = (V, E)$ a graph. I is a* topological order *if*

- *I is a bijection from $\{1, \dots, \}$ to V*
- If *u is a parent of v,* then $I(u) < I(v)$

Proposition 3.18 *$G = (V, E)$ has a topological order $\Leftrightarrow$ G is a DAG.*

Proof $\Rightarrow$ easy, $\Leftarrow$ use a depth-first search ■

3.3.4 Directed graphical models

Notations n discrete random variables $X_1, \dots, X_n$.

- joint distribution:
$$p(x_1, \dots, x_n) = P(X_1 = x_1, \dots, X_n = x_n)$$
- marginal distribution: for $A \subset V$,
$$p(x_A) = p_A(x_A) = P(X_k = x_k, k \in A) = \sum_{x_{A^c}} p(x_A, x_{A^c})$$
- conditional distribution:
$$p(x_A | x_{A^c}) = p_{A|A^c}(x_A | x_{A^c}) = P(X_A = x_A | X_{A^c} = x_{A^c})$$

Review

$$\begin{array}{rcll} X \perp\!\!\!\perp Y & \Leftrightarrow & p(x,y) = p(x)p(y) & \forall x, y \\ & \Leftrightarrow & p_{XY}(x,y) = p_X(x)p_Y(y) & \\ X \perp\!\!\!\perp Y | Z & \Leftrightarrow & p(x,y|z) = p(x|z)p(y|z) & \forall x, y, z \\ & \Leftrightarrow & p(x|y,z) = p(x|z) & \end{array}$$

Definitions and first properties

Let $G = (V, E)$ a DAG with $V = \{1, \dots n\}$ and $(X_1, \dots, X_n)$ n discrete random variables. $\mathcal{L}(G)$ set of $p(x) = p(x_1, \dots, x_n)$ of the form

$$p(x) = \prod_{i=1}^{n} f_i(x_i, x_{\pi_i})$$

with

- π_i set of parents of i

- $\forall i, f_i \geqslant 0$
- $\forall i, \sum_{x_i} f_i(x_i, x_{\pi_i}) = 1$

Proposition 3.19 *If $p(x)$ factorizes in G, i.e. $(p \in \mathcal{L}(G))$, then p is a distribution and*

$$\forall i, f_i(x_i, x_{\pi_i}) = p(x_i | x_{\pi_i})$$

Proof By induction on $n = |V|$. See next class.

■

Directed and undirected graphical models **Fall 2017**

Lecture 4 — October 18th

Lecturer: Guillaume Obozinski *Scribe:*

In this lecture, we will assume that all random variables are discrete, to keep notations as simple as possible. All the theory presented generalizes immediately to continuous random variables that have a density by replacing

- the discrete probability distributions considered in this lecture by densities
- summations by integration with respect to a measure of reference (most of the time the Lebesgue measure).

4.1 Notation and probability review

4.1.1 Notations

We review some notations before establishing some properties of directed graphical models. Let $X_1, X_2, \dots, X_n$ be random variables with distribution:

$$\mathbb{P}(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) = p_X(x_1, \dots, x_n) = p(x)$$

where x stands for $(x_1, \dots, x_n)$. Given $A \subset \{1, \dots, n\}$, we denote the marginal distribution of x_A by:

$$p(x_A) = \sum_{x \in A^c} p(x_A, x_{A^c}).$$

With this notation, we can write the conditional distribution as:

$$p(x_A | x_{A^c}) = \frac{p(x_A, x_{A^c})}{p(x_{A^c})}$$

We also recall the so-called "chain rule" stating:

$$p(x_1, \dots, x_n) = p(x_1)\, p(x_2|x_1)\, p(x_3|x_2, x_1)\, \dots\, p(x_n|x_1, \dots, x_{n-1})$$

4.1.2 Independence and conditional independence

Let A, B, and C be disjoint.
We will say that X_A is (marginally) independent of X_B and write $X_A \perp\!\!\!\perp X_B$ if and only if

$$p(x_A, x_B) = p(x_A)\, p(x_B) \qquad \forall (x_A, x_B), \tag{4.1}$$

or equivalently if and only if

$$p(x_A|x_B) = p(x_A) \qquad \forall x_A, x_B \quad \text{s.t.} \quad p(x_B) > 0. \tag{4.2}$$

Similarly we will say that X_A is independent from X_B conditionally on X_C (or given X_C) and we will write $X_A \perp\!\!\!\perp X_B \mid X_C$ if and only if

$$p(x_A, x_B|x_C) = p(x_A|x_C)\, p(x_B|x_C) \qquad \forall x_A, x_B, x_C \quad \text{s.t.} \quad p(x_C) > 0, \tag{4.3}$$

or equivalently if and only if

$$p(x_A|x_B, x_C) = p(x_A|x_C) \qquad \forall x_A, x_B, x_C \quad \text{s.t.} \quad p(x_B, x_C) > 0. \tag{4.4}$$

More generally we will say that the $(X_{A_i})_{1\leq i\leq k}$ are *mutually independent* if and only if

$$p(x_{A_1}, \ldots, x_{A_k}) = \prod_{i=1}^{k} p(x_{A_i}) \quad \forall x_{A_1}, \ldots, x_{A_k},$$

and that they are *mutually independent conditionally on* X_C (or given X_C) if and only if

$$p(x_{A_1}, \ldots, x_{A_k}|x_C) = \prod_{i=1}^{k} p(x_{A_i}|x_C) \qquad \forall x_{A_1}, \ldots, x_{A_k}, x_C \quad \text{s.t.} \quad p(x_C) > 0.$$

Remark 4.1.1 *Note that the conditional probability $p(x_A, x_B|x_C)$ is the probability distribution over (X_A, X_B) if X_C is known to be equal to x_C. In practice, it means that if the value of X_C is* observed *(e.g. via a measurement) then the distribution over (X_A, X_B) is $p(x_A, x_B|x_C)$. The conditional independence statement $X_A \perp\!\!\!\perp X_B \mid X_C$ should therefore be interpreted as "when the value of X_C is observed (or given) X_A and X_B are independent".*

Remark 4.1.2 *(Pairwise independence vs mutual independence) Consider a collection of r.v. $(X_1, \ldots, X_n)$. We say that these variables are pairwise independent if for all $1 \leq i < j \leq n$, $X_i \perp\!\!\!\perp X_j$. Note that this is different than assuming that $X_1, \ldots, X_n$ are mutually (or jointly or globally) independent. A standard counter-example is as follows: given two variables X, Y that are independent coin flips define Z via the XOR function $\oplus$ with $Z = X \oplus Y$. Then, the three random variables X, Y, Z are* pairwise independent*, but not* mutually independent*. (Prove this as an exercise.) The notations presented for* pairwise independence *could be generalized to collections of variables that are* mutually independent*.*

Three Facts About Conditional Independence

1. **Can repeat variables:** $X \perp\!\!\!\perp (Y, Z) \mid Z, W$ is the same as $(X, Z) \perp\!\!\!\perp Y \mid Z, W$. The repetition is redundant but may be convenient notation.
2. **Decomposition:** $X \perp\!\!\!\perp (Y, Z) \mid W$ implies that $X \perp\!\!\!\perp Y \mid W$ and $X \perp\!\!\!\perp Z \mid W$.
3. The chain rule applies to conditional distributions:

$$p(x, y|z) = p(x|y, z)\, p(y|z). \tag{4.5}$$

Proving each of these three facts are good simple exercises.

4.2 Directed Graphical Model

Graphical models combine probability and graph theory into an efficient data structure. We want to be able to handle probabilistic models of hundreds of variables. For example, assume we are trying to model the probability of diseases given the symptoms, as shown below.

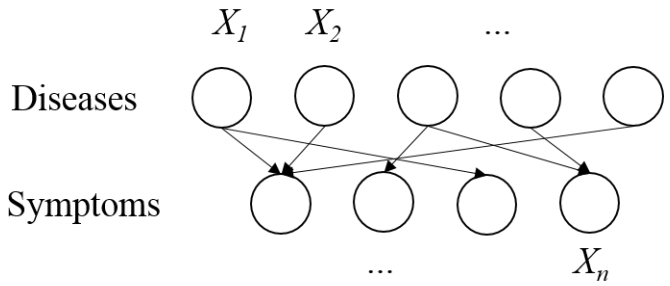


Figure 4.1. Nodes representing binary variables indicating the presence or not of a disease or a symptom.

We have n nodes, each a binary variable ($X_i \in \{0, 1\}$), indicating the presence or absence of a disease or a symptom. The number of joint probability terms would grow exponentially. For 100 diseases and symptoms, we would need a table of size 2^{100} to store all the possible states. This is clearly intractable. Instead, we will use graphical models to represent the relationships between nodes.

General issues in this class

1. Representation → DGM, UGM / parameterization → exponential family
2. Inference (computing $p(x_A|x_B)$) → sum-product algorithm
3. Statistical estimation → maximum likelihood, maximum entropy

A directed graphical model, also historically called "Bayesian network" when the variables are discrete, represents a *family of distributions*, denoted $\mathcal{L}(G)$, where $\mathcal{L}(G) \triangleq \{p : \exists$ legal factors, f_i, s.t. $p(x_V) = \prod_{i=1}^{n} f_i(x_i, x_{\pi_i})\}$, where the legal factors satisfy $f_i \geq 0$ and $\sum_{x_i} f_i(x_i, x_{\pi_i}) = 1 \; \forall i, x_{\pi_i}$.

4.2.1 First definitions and properties

Let $X_1, \ldots, X_n$ be n random variables with joint distribution $p(x) = p_X(x_1, \ldots, x_n)$.

Definition 4.1 *Let $G = (V, E)$ be a DAG with $V = \{1, \dots, n\}$. We say that $p(x)$ factorizes in G, denoted $p(x) \in \mathcal{L}(G)$, if there exists some functions f_i, called factors, such that:*

$$\begin{aligned} \forall x,\ p(x) &= \prod_{i=1}^{n} f_i(x_i, x_{\pi_i}) \\ f_i \geq 0, \quad \forall i, \forall x_{\pi_i},\ \sum_{x_i} f_i(x_i, x_{\pi_i}) &= 1 \end{aligned} \tag{4.6}$$

where we recall that π_i stands for the set of parents of the vertex i in G.

We prove the following useful and fundamental property of directed graphical models: if a probability distribution factorizes according to a directed graph $G = (V, E)$, the distribution obtained by marginalizing a leaf[1] i factorizes according to the graph induced on $V \backslash \{i\}$.

Proposition 4.2 (Leaf marginalization) *Suppose that p factorizes in G, i.e. $p(x_V) = \prod_{j=1}^{n} f_j(x_j, x_{\pi_j})$. Then for any leaf i, we have that $p(x_{V\backslash\{i\}}) = \prod_{j \neq i} f_j(x_j, x_{\pi_j})$, hence $p(x_{V\backslash\{i\}})$ factorizes in $G' = (V\backslash\{i\}, E')$, the induced graph on $V\backslash\{i\}$.*

Proof Without loss of generality, we can assume that the leaf is indexed by n. Since it is a leaf, we clearly have that $n \notin \pi_i, \forall i \leq n-1$. We have the following computation:

$$\begin{aligned} p(x_1, ...x_{n-1}) &= \sum_{x_n} p(x_1, \dots, x_n) \\ &= \sum_{\substack{x_n \\ n-1}} \left(\prod_{i=1}^{n-1} f_i(x_i|x_{\pi_i}) f_n(x_n|x_{\pi_n}) \right) \\ &= \prod_{\substack{i=1 \\ n-1}} f_i(x_i|x_{\pi_i}) \sum_{x_n} f_n(x_n|x_{\pi_n}) \\ &= \prod_{i=1}^{n-1} f_i(x_i|x_{\pi_i}). \end{aligned}$$

■

Remark 4.2.1 *Note that the new graph obtained by removing a leaf is still a DAG. Indeed, since we only removed edges and nodes, if there was a cycle in the induced graph, the same cycle would be present in the original graph, which is not possible since it is DAG.*

Remark 4.2.2 *Also, by induction, this result shows that in the definition of factorization we do not need to assume that p is a probability distribution. Indeed, if any function p satisfies (4.6) then it is a probability distribution, because its non-negative as a product of non-negative factors and it sums to 1 by using formula proved by induction.*

[1] We call here a *leaf* or *terminal node* of a DAG a node that has no descendant.

Lemme 4.3 *Let A, B, C be three sets of nodes such that $C \subset B$ and $A \cap B = \varnothing$. If $p(x_A \mid x_B)$ is a function of only (x_A, x_C) then $p(x_A|x_B) = p(x_A|x_C)$.*

Proof We denote by $f(x_A, x_C) := p(x_A|x_B)$ the corresponding function. Then $p(x_A, x_B) = p(x_A \mid x_B)\, p(x_B) = f(x_A, x_C)\, p(x_B)$. By summing over $x_{B\setminus C}$, we have

$$p(x_A, x_C) = \sum_{x_{B\setminus C}} p(x_A, x_B) = f(x_A, x_C) \sum_{x_{B\setminus C}} p(x_B) = f(x_A, x_C)\, p(x_C),$$

which proves that $p(x_A|x_C) = f(x_A, x_C) = p(x_A|x_B)$. ■

Now we try to characterize the factor functions. The following result will imply that if p factorizes in G, then we have a uniqueness of the factors.

Proposition 4.4 *If $p(x) \in \mathcal{L}(G)$ then, for all $i \in \{1, \dots, n\}$, $f_i(x_i, x_{\pi_i}) = p(x_i|x_{\pi_i})$.*

Proof Assume, without loss of generality, that the nodes are sorted in a topological order. Consider a node i. Since the node are in topological order, for any $1 \leq j \leq n$, we have $\pi_j \subset \{1, \dots, j-1\}$; as a consequence we can apply Proposition 4.2 $n-i$ times to obtain that $p(x_1, \dots, x_i) = \prod_{j \leq i} f(x_j, x_{\pi_j})$. Since we also have $p(x_1, \dots, x_{i-1}) = \prod_{j<i} f(x_j, x_{\pi_j})$, taking the ratio, we have

$$p(x_i \mid x_1, \dots, x_{i-1}) = f(x_i, x_{\pi_i}).$$

Since $\pi_i \subset \{1, \dots, i-1\}$, this entails by the previous lemma that $p(x_i \mid x_1, \dots, x_{i-1}) = p(x_i \mid x_{\pi_i}) = f(x_i, x_{\pi_i})$. ■

Hence we can give an equivalent definition for a DAG to the notion of factorization:

Definition 4.5 *(Equivalent definition) The probability distribution $p(x)$ factorizes in G, denoted $p(x) \in \mathcal{L}(G)$, iff*

$$\forall x, \quad p(x) = \prod_{i=1}^{n} p(x_i|x_{\pi_i}) \tag{4.7}$$

Example 4.2.1

- *(Trivial Graphs) Assume $E = \varnothing$,* i.e. *there are no edges. We then have $p(x) = \prod_{i=1}^{n} p(x_i)$, implying the random variables $X_1, \dots, X_n$ are independent, that is variables are mutually independent if they factorize in the empty graph.*
- *(Complete Graphs) Assume now we have a complete graph (thus with $n(n-1)/2$ edges as we need acyclicity for it to be a DAG), we have: $p(x) = \prod_{i=1}^{n} p(x_i|x_1, \dots, x_{i-1})$, the so-called "chain rule" which is always true. Every probability distribution factorizes in complete graphs. Note that there are $n!$ complete graph possible that are all equivalent...*
- *(Graphs with several connected components) If G has several connected components $C_1, \dots, C_k$, then $p \in \mathcal{L}(G) \Rightarrow p(x) = \prod_{j=1}^{k} p(x_{C_j})$* (Exercise). *As a consequence, each connected component can be treated separately. In the rest of the lecture, we will therefore focus on connected graphs.*

4.2.2 Graphs with three nodes

We consider all connected graphs with three nodes, except for the complete graph, which we have already discussed.

- (Markov chain) The Markov chain on three nodes is illustrated on Fig.(4.2). For this graph we have
$$p(x, y, z) \in \mathcal{L}(G) \Rightarrow X \perp\!\!\!\perp Y \mid Z \tag{4.8}$$
Indeed we have:
$$p(y|z, x) = \frac{p(x, y, z)}{p(x, z)} = \frac{p(x, y, z)}{\sum_{y'} p(y', x, z)} = \frac{p(x)p(z|x)p(y|z)}{\sum_{y'} p(x)p(z|x)p(y'|z)} = p(y|z)$$

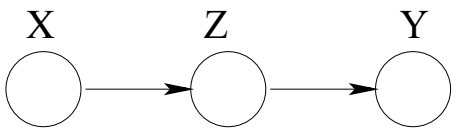


Figure 4.2. Markov Chain

- (Latent cause) It is the type of DAG given in Fig.(4.3). We show that:
$$p(x, y, z) \in \mathcal{L}(G) \Rightarrow X \perp\!\!\!\perp Y \mid Z \tag{4.9}$$
Indeed:
$$p(x, y|z) = \frac{p(x, y, z)}{p(z)} = \frac{p(z)p(y|z)p(x|z)}{p(z)} = p(x|z)p(y|z)$$

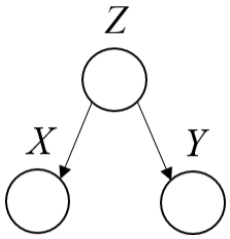


Figure 4.3. Common latent "cause"

- (Explaining away) Represented in Fig.(4.4), we can show for this type of graph:
$$p(x) \in \mathcal{L}(G) \Rightarrow X \perp\!\!\!\perp Y \tag{4.10}$$
It basically stems from:
$$p(x, y) = \sum_z p(x, y, z) = p(x)p(y) \sum_z p(z|x, y) = p(x)p(y)$$

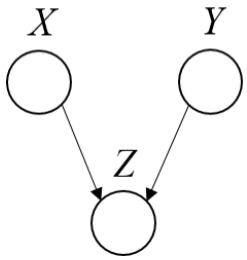


Figure 4.4. Explaining away, or v-structure

Remark 4.2.3 *The word "cause" should here be between quotes and used very carefully, because the same way that* Correlation is not causation*, conditional dependance is not causation either. This is however the historical name for this model. The reason why cause is a bad name, and that* latent factor *might be better, is that the factorisation properties that are encoded by graphical models do not in general correspond to the existence of a causal mechanisms, but only to conditional independence relations.*

Remark 4.2.4 *If p factorizes in the "latent cause" graph, then $p(x, y, z) = p(z)p(x|z)p(y|z)$. But using Bayes' rule $p(z)\,p(x|z) = p(x)\,p(z|x)$, adn so we also have that $p(x, y, z) = p(x)p(z|x)p(y|z)$ which shows that p is a Markov chain (i.e. factorizes in the Markov chain graph). This is an example of* basic edge reversal *that we will discuss in the next section. Note that we proceeded by equivalence, which shows that the Markov chain graph and the "latent cause" graph and the reversed Markov chain graph are in fact equivalent in the sense that distribution that factorize according to one factorize according to the others. This is what we will call Markov equivalence.*

Remark 4.2.5 *In the "explaining away" graph, in general $X \perp\!\!\!\perp Y|Z$ is not true in the sense that there exist elements in $\mathcal{L}(G)$ such that this statement is violated.*

Remark 4.2.6 *For a graph, $(p \in \mathcal{L}(G))$ implies that p satisfies some list of (positive) conditional independence statements (CIS). The fact that p is in $\mathcal{L}(G)$ cannot guarantee that a given CIS does not hold. This should be obvious because the independent distribution belongs to all graphical models and satisfies all CIS...*

Remark 4.2.7 *It is important to note that not all lists of CIS correspond to a graph, in the sense that there are lists of CIS for which there exists is no graph such that $\mathcal{L}(G)$ is formed exactly of the distributions which satisfy only the conditional independences that are listed or that are consequences of the ones listed. In particular there is no graph G on three variables such that $\mathcal{L}(G)$ contains all distributions on (X,Y,Z) that satisfy $X \perp\!\!\!\perp Y$, $Y \perp\!\!\!\perp Z$, $X \perp\!\!\!\perp Z$ and does not contain distributions for which any of these statements is violated. (Remember that pairwise independence does not imply mutual independence: see Remark 4.1.2).*

4.2.3 Inclusion, reversal and marginalization properties

Inclusion property. Here is a quite intuitive proposition about included graphs and their factorization.

Proposition 4.6 *If $G = (V, E)$ and $G' = (V, E')$ then:*

$$E \subset E' \quad \Rightarrow \quad \mathcal{L}(G) \subset \mathcal{L}(G'). \tag{4.11}$$

Proof If $p \in \mathcal{L}(G)$, then $p(x) = \prod_{i=1}^{n} p(x_i, x_{\pi_i(G)})$. Since $E \subset E'$, it is obvious that $\pi_i(G) \subset \pi_i(G')$, and we can define $f_i(x_i, x_{\pi_i(G')}) := p(x_i|x_{\pi_i(G)})$. Since $p(x) = \prod_{i=1}^{n} f_i(x_i, x_{\pi_i(G')})$ and f_i meets the requirements of Definition 4.1, this proves that $p \in \mathcal{L}(G')$. ■

The converse of the previous proposition is not true. In particular, different graphs can define the same set of distribution. We introduce first some new definitions:

Definition 4.7 *(Markov equivalence) We say that two graphs G and G' are Markov equivalent if $\mathcal{L}(G) = \mathcal{L}(G')$.*

Proposition 4.8 *(Basic edge reversal) If $G = (V, E)$ is a DAG and if for $(i, j) \in E$, i has no parents and the only parent of j is i, then the graph obtained by reversing the edge (i, j) is Markov equivalent to G.*

Proof First, note that by reversing such an edge no cycle can be created because the cycle would necessarily contain (j, i) and j has no parent other than i. Using Bayes' rule: $p(x_i)\, p(x_j|x_i) = p(x_j)\, p(x_i|x_j)$ we convert the factorization w.r.t. to G to factorization w.r.t. to the graph obtained by edge reversal. ■

Informally, the previous result can be reformulated as: an edge reversal that does not remove or creates any v-structure leads to a graph which is Markov equivalent.

When applied to the 3-nodes graphs considered earlier, this property proves that the Markov chain and the "latent cause" graph are equivalent. On the other hand, the fact that the "explain away" graph has a v-structure is the reason why it is not equivalent to the others.

Definition 4.9 *(Covered edge) An edge (i, j) is said to be covered if $\pi_j = \{i\} \cup \pi_i$.*

Proposition 4.10 *(Covered edge reversal) Let $G = (V, E)$ be a DAG and $(i, j) \in E$ a covered edge. Let $G' = (V, E')$ with $E' = (E \backslash \{(i, j)\}) \cup \{(j, i)\}$, then G' is necessarily also a DAG and $\mathcal{L}(G) = \mathcal{L}(G')$.*

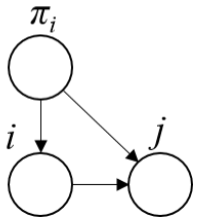


Figure 4.5. Edge (i, j) is covered

Proof Exercise in Homework 2. ■

Marginalization. We have proved in Proposition 4.2 that if $p(x_1, \ldots, x_n)$ factorizes in G, the distribution obtained by marginalizing a leaf i factorizes in the graph G' induced on $V \backslash \{i\}$ by G. A nice property of the obtained graph is that all the conditional independences between variables $X_1, \ldots, X_{n-1}$ that were implied by G are still implied by G': marginalization has lost CI information about X_n but not about the rest of the distribution. It would be natural to try to generalize this and a legitimate question is: if we marginalise a node i in a distribution of $\mathcal{L}(G)$ is there a simple construction of a graph G' such that the marginalized distribution factorizes in G' and such that all the CIS that hold in G and do not involve X_i are still implied by G'. Unfortunately this is not true. Another less ambitious natural question is then: is there a unique smallest graph G' such that if $p \in \mathcal{L}(G)$ then the distribution obtained by marginalizing i is in $\mathcal{L}(G')$. Unfortunately this is not the case either, as illustrated by the following exemple.

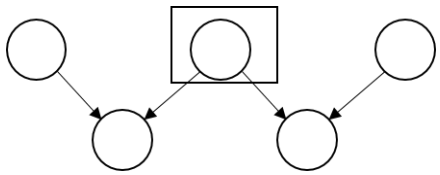


Figure 4.6. Marginalizing the boxed node would not result in family of distributions that cannot be exactly represented by a directed graphical model and one can check that there is no unique smallest graph in which the obtained distribution factorize.

Conditional independence with the non-descendents. In a Markov chain, a well known property is that the X_t is independent of the past given X_{t-1}. This result generalizes as follows in a directed graphical model: if $p(x)$ factorizes in G then every single random variable is independent from the set of its non-descendants given its parents.

Definition 4.11 *The set of non-descendants of i denoted $nd(i)$ is the set of nodes that are not descendants of i.*

Lemme 4.12 *For a graph $G = (V, E)$ and a node i, there exists a topological order such that all elements of $nd(i)$ appear before i.*

Proof This is easily proved constructively: we construct the topological order in reverse order. At each iteration we remove a node among leaves (of the remaining graph) which we add in the reverse order, and specifically, if some leaves are descendants of i then we remove one of those. If at any iteration there is no leaf that is a descendant of i, it means that all descendants of i have been removed from the graph. Indeed, if there were some descendants of i left in the graph, since all their descendants are descendants of i as well there would exist a leaf node which is a descendant of i. This procedure thus removes all strict descendants of i first, then i and then only all elements of $nd(i)$. ■

With this lemma, we can show our main result.

Proposition 4.13 *If G is a DAG, then:*

$$p(x) \in \mathcal{L}(G) \Leftrightarrow X_i \perp\!\!\!\perp X_{nd(i)} | X_{\pi_i} \tag{4.12}$$

Proof First, we consider the $\Rightarrow$ direction. Based on the previous lemma we can find an order such that $nd(i) = \{1, \dots, i-1\}$. But we have proven in Proposition 4.4 that $p(x_i|x_{\pi_i}) = p(x_i|x_{1:(i-1)})$, which given the order chosen is also $p(x_i|x_{1:(i-1)}) = p(x_i|x_{\pi_i}, x_{nd(i)\setminus\pi_i})$; this proves what we wanted to show: $X_i \perp\!\!\!\perp X_{nd(i)\setminus\pi_i} | X_{\pi_i}$.

We now prove the $\Leftarrow$ direction. Let $1:n$ be a topological order, Then $\{1, \cdots i-1\} \subseteq nd(i)$. (By contradiction, suppose $j \in \{1 \cdots i-1\}$ and $j \notin nd(i)$, then $\exists$ path from i to j, which contradicts the topological order property as there would be an edge from i to an element of $\{1, \dots, i-1\}$.)

By the chain rule, we always have $p(x_V) = \prod_{i=1}^{n} p(x_i|x_{1:i-1})$ but by the conditional independence assumptions $p(x_i|x_{1:i-1}) = p(x_i|x_{\pi_i})$, hence the result by substitution. ■

4.2.4 d-separation

Given a graph G and A, B and C, three subsets it would be useful to be able to answer the question: is $X_A \perp\!\!\!\perp X_B \mid X_C$ true for all $p \in \mathcal{L}(G)$? An answer is provided by the concept of d-separation, or directed separation.

We call a chain a path in the symmetrized graph, *i.e.* in the graph the undirected graph obtained by ignoring the directionality of the edges.

Definition 4.14 *(Chain) Let $a, b \in V$, a chain from a to b is a sequence of nodes, say $(v_1, \dots, v_n)$ such that $v_1 = a$ and $v_n = b$ and $\forall j, (v_j, v_{j+1}) \in E$ or $(v_{j+1}, v_j) \in E$.*

Assume C is a set that is observed. We want to define a notion of being 'blocked' by this set C in order to answer the underlying question above.

Definition 4.15 *(Blocking node in a chain, blocked chain and d-separation)*

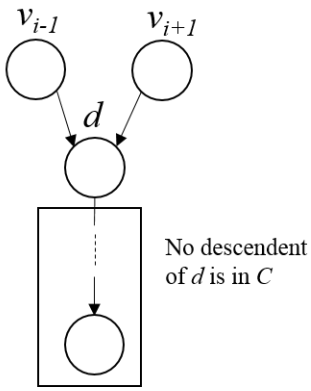


Figure 4.7. D-separation

1. *A chain from a et b is blocked at d if:*
   - *either $d \in C$ and (v_{i-1}, d, v_{i+1}) is not a v-structure;*
   - *or $d \notin C$ and (v_{i-1}, d, v_{i+1}) is a v-structure and no descendants of d is in C.*
2. *A chain from a to b is blocked if and only if it is blocked at any node.*
3. *A and B are said to be d-separated by C if and only if all chains that go from $a \in A$ to $b \in B$ are blocked.*

Example 4.2.2

- ***Markov chain:*** *Applying d-separation to the Markov chain retrieves the well know results that the future is independent to the past given the present.*

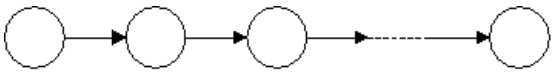


Figure 4.8. Markov chain

- ***Hidden Markov Model:*** *We can apply it as well to the hidden Markov chain graph of Figure 4.9.*

4.2.5 Bayes ball algorithm

Checking whether two nodes are d-separated is not always easy. The Bayes ball algorithm is an intuitive "reachability" algorithm to answer this question. Suppose we want to determine if X is conditionally independent from Z given Y. The principle of the algorithm is to place initially a ball on each of the nodes in X, to then let them bounce around according to some

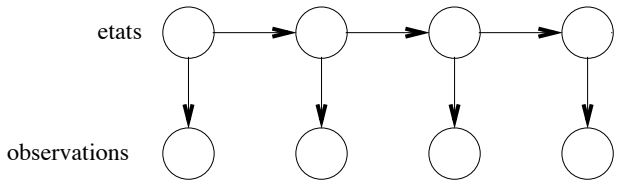


Figure 4.9. Hidden Markov Model

rules described below and to see if any reaches Z. $X \perp\!\!\!\perp Z \mid Y$ is true if none reached Z, but not otherwise.

The rules are as follows for the three canonical graph structures. Note that the balls are allowed to travel in either direction along the edges of the graph.

1. **Markov chain:** Balls pass through when we do not observe Y, but are blocked otherwise.

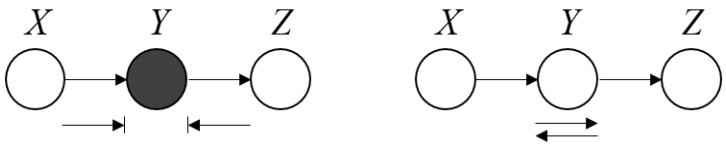


Figure 4.10. Markov chain rule: When Y is observed, balls are blocked (left). When Y is not observed, balls pass through (right)

2. **Two children:** Balls pass through when we do not observe Y, but are blocked otherwise.

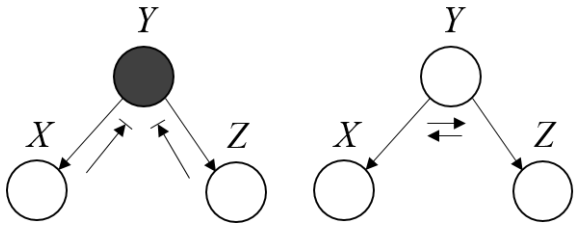


Figure 4.11. Rule when X and Z are Y's children: When Y is observed, balls are blocked (left). When Y is not observed, balls pass through (right)

3. **v-structure:** Balls pass through when we observe Y, but are blocked otherwise.

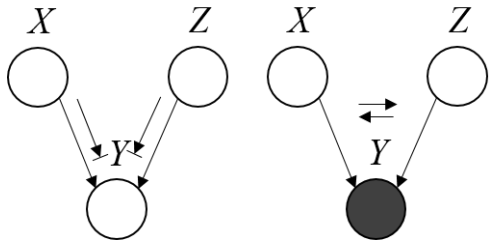


Figure 4.12. v-structure rule: When Y is not observed, balls are blocked (left). When Y is observed, balls pass through (right)

4.3 Undirected graphical models

4.3.1 Definition

Definition 4.16 *Let $G = (V, E)$ be a* **undirected graph**. *We denote by $\mathcal{C}$ the set of all cliques of G . We say that a probability distribution p factorizes in G and write $p \in \mathcal{L}(G)$ if $p(x)$ is of the form:*

$$p(x) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C) \quad \text{with} \quad \psi_C \geq 0,\ C \in \mathcal{C} \quad \text{and} \quad Z = \sum_x \prod_{C \in \mathcal{C}} \psi_C(x_C).$$

The functions ψ_C are not probability distributions like in the directed graphical models. They are called *potentials*.

Remark 4.3.1 *With the normalization by Z of this expression, we see that the function ψ_C are defined up to a multiplicative constant.*

Remark 4.3.2 *We may restrict $\mathcal{C}$ to $\mathcal{C}_{max}$, the set of maximal cliques.*

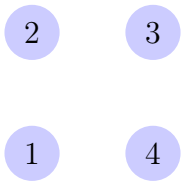
Remark 4.3.3 *This definition can be extended to any function: f is said to factorize in $G \iff f(x) = \prod_{C \in \mathcal{C}} \psi_C(x_C)$.*

4.3.2 Trivial graphs

Empty graphs We consider $G = (V, E)$ with $E = \emptyset$. For $p \in \mathcal{L}(G)$, we get:

$$p(x) = \prod_{i=1}^{n} \psi_i(x_i) \quad \text{given that } \mathcal{C} = \{\{i\} \in V\}.$$

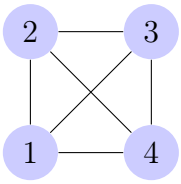
So $X_1, ..., X_n$ must be mutually independent.



Complete graphs We consider $G = (V, E)$ with $\forall i, j \in V, (i, j) \in E$. For $p \in \mathcal{L}(G)$, we get:

$$p(x) = \frac{1}{Z}\psi_V(x_V) \text{ given that } \mathcal{C} \text{ is reduced to a single set } V.$$

This places no constraints on the distribution of $(X_1, ..., X_n)$.



4.3.3 Separation and conditional dependence

Proposition 4.17 *Let $G = (V, E)$ and $G' = (V, E')$ be two undirected graphs.*

$$E \subseteq E' \Rightarrow \mathcal{L}(G) \subseteq \mathcal{L}(G')$$

Proof The cliques of G are included in cliques of G'. ■

Definition 4.18 *We say that p satisfies the* **Global Markov property** *w.r.t. G if and only if for all $A, B, S \subset V$ disjoint subsets: (A and B are separated by S) $\Rightarrow (X_A \perp\!\!\!\perp X_B \mid X_S)$.*

Proposition 4.19 *If $p \in \mathcal{L}(G)$ then, p satisfies the Global Markov property w.r.t. G.*

Proof We suppose without loss of generality that A, B, and S are disjoint sets such that $A \cup B \cup S = V$, as we could otherwise replace A and B by :

$$A' = A \cup \{a \in V / a \text{ and } A \text{ are not separated by } S\}$$

$$B' = V \setminus \{S \cup A'\}$$

A' and B' are separated by S and we have the disjoint union $A' \cup B' \cup S = V$. If we can show that $X_{A'} \perp\!\!\!\perp X_{B'}|X_S$, then by the decomposition property, we also have that $X_A \perp\!\!\!\perp X_B|X_S$ for any subset A of A' and B of B', giving the required general case.

We consider $C \in \mathcal{C}$. It is not possible to have both $C \cap A \neq \emptyset$ and $C \cap B \neq \emptyset$ as A and B are separated by S and C is a clique. Thus $C \subset A \cup S$ or $C \subset B \cup S$ (or both if $C \subset S$). Let $\mathcal{D}$ be the set of cliques C such that $C \subset A \cup S$ and $\mathcal{D}'$ the set of all other cliques. We have:

$$p(x) = \frac{1}{Z} \prod_{\substack{C \in \mathcal{C} \\ C \subset A \cup S}} \psi_C(x_C) \prod_{C \in \mathcal{D}'} \psi_C(x_C) = f(x_{A \cup S}) g(x_{B \cup S}).$$

Thus:

$$p(x_A, x_S) = \frac{1}{Z} f(x_A, x_S) \sum_{x_B} g(x_B, x_S) \implies p(x_A|x_S) = \frac{f(x_A, x_S)}{\sum_{x'_A} f(x'_A, x_S)}.$$

Similarly: $p(x_B|x_S) = \frac{g(x_B, x_S)}{\sum_{x'_B} g(x'_A, x_S)}$. Hence:

$$p(x_A, x_S)p(x_B|x_S) = \frac{\frac{1}{Z} f(x_A, x_S) g(x_B, x_S)}{\frac{1}{Z} \sum_{x'_A} f(x'_A, x_S) \sum_{x'_B} g(x'_A, x_S)} = \frac{p(x_A, x_B, x_S)}{p(x_S)} = p(x_A, x_B|x_S).$$

i.e. $X_A \perp\!\!\!\perp X_B|X_S$. ■

Theorem 4.20 *(Hammersley - Clifford) If* $\forall x,\ p(x) > 0$ *then* $p \in \mathcal{L}(G) \iff$ *p satisfies the global Markov property.*

4.3.4 Marginalization

As for directed graphical models, we also have a marginalization notion in undirected graphs. It is slightly different. If $p(x)$ factorizes in G, then $p(x_1, \ldots, x_{n-1})$ factorizes in the graph where the node n is removed and all neighbors are connected.

Proposition 4.21 *Let $G = (V, E)$ be an undirected graph. Let $G' = (V', E')$ be the graph where n is removed and its neighbors are connected, i.e. $V' = V \setminus \{n\}$, and E' is obtained from the set E by first connecting together all the neighbours of n and then removing n. If $p \in \mathcal{L}(G)$ then $p(x_1, ..., x_{n-1}) \in \mathcal{L}(G')$. Hence undirected graphical models are closed under marginalization as the construction above is true for any vertex.*

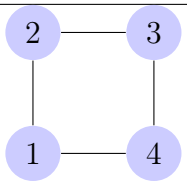
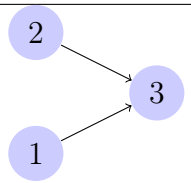
We now introduce the notion of Markov blanket

Definition 4.22 *For $i \in V$, the* **Markov blanket** *of a graph G is the smallest set of nodes that makes X_i independent to the rest of the graph.*

Remark 4.3.4 *The Markov blanket in an undirected graph for $i \in V$ is the set of its neighbors. For a directed graph, it is the union of all parents, all children and parents of children.*

4.3.5 Relation between directed and undirected graphical models

Since now we have seen that many notions developed for directed graph naturally extended to undirected graphs. The raising question is thus to know whether we can find a theory including both directed and undirected graphs, in particular, is there a way—for instance by symmetrizing the directed graph as we have done repeatedly—to find a general equivalence between those two notions. The answer is no, as we will discuss—though it might work in some special cases described above.

	Directed graphical model	Undirected graphical model
Factorization	$p(x) = \prod_{i=1}^{n} p(x_i \mid x_{\pi_i})$	$p(x) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C)$
Set independence	d-separation $[x_i \perp\!\!\!\perp x_{nd(i)} \mid x_{\pi_i}]$ (and many more)	separation $[X_A \perp\!\!\!\perp X_B \mid X_S]$
Marginalization	not closed in general, only when marginalizing leaf nodes	closed
Difference	grid 	v-structure 

Let G be DAG. Can we find G' undirected such that $\mathcal{L}(G) = \mathcal{L}(G')$? $\mathcal{L}(G) \subset \mathcal{L}(G')$?

Definition 4.23 *Let $G = (V, E)$ be a* DAG*. The* **symmetrized graph** *of G is $\tilde{G} = (V, \tilde{E})$, with $\tilde{E} = \{(u, v), (v, u) / (u, v) \in E\}$, ie. an edge going the opposite direction is added for every edge in E.*

Definition 4.24 *Let $G = (V, E)$ be a* DAG*. The* **moralized graph** *$\bar{G}$ of G is the symmetrized graph $\tilde{G}$, where we add edge such that for all $v \in V$, π_v is a clique.*

We admit the following proposition:

Proposition 4.25 *Let G be a* DAG *without any v-structure, then $\bar{G} = \tilde{G}$ and $\mathcal{L}(G) = \mathcal{L}(\tilde{G}) = \mathcal{L}(\bar{G})$.*

In case there is a v-structure in the graph, we can only conclude:

Proposition 4.26 *Let G be a* DAG*, then $\mathcal{L}(G) \subset \mathcal{L}(\bar{G})$.*

$\bar{G}$ is minimal for the number of edges in the set H of undirected graphs such that $\mathcal{L}(G) \subset \mathcal{L}(H)$.

Not all conditional independence structure for random variables can be factorized in a graphical model (directed or undirected).

Information Theory, Exponential families **2013/2014**

Lecture 5 — October 30th

Lecturer: Guillaume Obozinski *Scribe: Thomas Belhalfaoui, Lénaïc Chizat*

5.1 Information Theory

5.1.1 Entropy

We will use the following properties (Jensen Inequality):

1. if $f : \mathbb{R} \to \mathbb{R}$ is convex and if X is an integrable random variable :
$$\mathbb{E}_X(f(X)) \geq f(\mathbb{E}_X(X))$$
2. if $f : \mathbb{R} \to \mathbb{R}$ is strictly convex, we have equality if and only if X is constant a.s.

Definition 5.1 (Entropy) *Let X be a random variable taking values in the finite set $\mathcal{X}$. We denote $p(x) = P(X = x)$.*
In information theory, the quantity
$$I(x) = \log \frac{1}{p(x)}$$
can be interpreted as a quantity of information carried by the occurrence of x. (This is sometimes called self-information). Entropy is defined as the expected amount of information of the random variable.
$$H(X) = E_{p(x)}\left[I(X)\right] = -\sum_{x\in\mathcal{X}} p(x)\, \log\, p(x)$$
The base of the logarithm is the natural base or 2, the latter being more consistent with bit coding interpretations of entropy. In this course we will use the natural logarithm.

5.1.2 Kullback-Leibler divergence

Definition 5.2 (Kullback Leibler Divergence) *Let p and q be two finite distributions on $\mathcal{X}$. The Kullback Leibler Divergence between p and q is defined by*
$$\begin{aligned} D(p \parallel q) &= \sum_{x\in\mathcal{X}} p(x) \log \frac{p(x)}{q(x)} \\ &= \sum_{x\in\mathcal{X}} \frac{p(x)}{q(x)} \left(\log \frac{p(x)}{q(x)}\right) q(x) \\ &= E_{X\sim q}\left[\frac{p(X)}{q(X)} \log \frac{p(X)}{q(X)}\right] \end{aligned}$$

KL Divergence is *not* a distance as it is not symmetric.

Proposition 5.3 *$D(p \parallel q) \geq 0$ and equality holds if and only if $p = q$.*

Proof If there exists $x \in \mathcal{X}$ such that $q(x) = 0$ and $p(x) \neq 0$ then $D(p \parallel q) = +\infty$. Otherwise, we can without loss of generality assume that $q(x) > 0$ everywhere. We make this assumption in the rest of the proof. By convexity of the function $y \mapsto y \log y$, and by Jensen's inequality, we have

$$D(p \parallel q) = E_q\left[\frac{p(X)}{q(X)} \log\left(\frac{p(X)}{q(X)}\right)\right] \geq E_q\left[\frac{p(X)}{q(X)}\right] \log E_q\left[\frac{p(X)}{q(X)}\right] = 0$$

since

$$E_q\left[\frac{p(X)}{q(X)}\right] = \sum_{x \in \mathcal{X}} \frac{p(x)}{q(x)} q(x) = \sum_{x \in \mathcal{X}} p(x) = 1.$$

Furthermore, $D(p \parallel q) = 0$ iff there is an equality in Jensen's inequality above which implies that $p(x) = cq(x)$ q-a.s., but summing this last equality over x implies that $c = 1$, which in turn implies that $p = q$. ■

Proposition 5.4 *We have the following inequalities:*

1. *$H(X) \geq 0$ with equality if X is constant a.s*
2. *$H(X) \leq \log(Card(\mathcal{X}))$*

Proof Since $p(x) = \mathbb{P}_p(X = x) \leq 1$ then $-p(x) \log p(x) \geq 0$ which implies that $H(X) \geq 0$ with equality iff $-p(x) \log p(x) = 0$ for all $x \in \mathcal{X}$, which proves the first point. Then

$$\begin{aligned} D(p \parallel q) &= -\sum_{x \in \mathcal{X}} p(x) \log q(x) - (-\sum_{x \in \mathcal{X}} p(x) \log p(x)) \\ &= -\sum_{x \in \mathcal{X}} p(x) \log q(x) - H(X) \end{aligned}$$

We choose $q_0(x) = \frac{1}{Card(\mathcal{X})}$. Then $H(X) = \log(Card(\mathcal{X})) - D$. Hence $H(X) \leq log(Card(\mathcal{X}))$. ■

Definition 5.5 (Mutual information) *Let X, Y be two random variables of joint distribution $p_{X,Y}(x, y) = P(X = x, Y = y)$ and with marginal distributions $p_X(x) = \sum_y p_{X,Y}(x, y)$ and $p_Y(y) = \sum_x p_{X,Y}(x, y)$. The mutual information of X and Y is defined by*

$$\begin{aligned} I(X, Y) &= \sum_{x,y} p_{X,Y}(x, y) \log \frac{p_{X,Y}(x, y)}{p_X(x)\, p_Y(y)} \\ &= D(p_{X,Y} \parallel p_X p_Y) \end{aligned}$$

Proposition 5.6 $I(X,Y) = 0 \Leftrightarrow X \perp\!\!\!\perp Y$

Proof It directly follows from the fact that $D(p_{X,Y} \parallel p_X\, p_Y) = 0$ implies that $p_{X,Y}(x,y) = p_X(x)p_Y(y)$ which is the definition of the independence of X and Y. ■

Independent $\Rightarrow$ not correlated **but** not correlated $\not\Rightarrow$ independence
The first implication comes from the fact that if $X \perp\!\!\!\perp Y$ then $E(X,Y) = E(X)E(Y)$ and then $Cov(X,Y) = 0$.
Counter-example for the reverse implication: if Θ is a r.v. following the uniform distribution on $[0,1]$ and we define the random variables X and Y by $X = \sin(2\pi\Theta)$ and $Y = \cos(2\pi\Theta)$ then X and Y are not correlated but dependent.

Remark 5.1.1 *The reverse is only true for Gaussian random variables.*

5.1.3 Relation between minimum Kullback-Leibler divergence and maximum likelihood principle

Definition 5.7 (Empirical distribution) *Let $x_1, ..., x_N \in \mathcal{X}$ be N i.i.d. observations of a random variable X.*
The empirical distribution of X derived from this sample is

$$\hat{p}(x) = \frac{1}{N}\sum_{n=1}^{N} \delta(x - x_n)$$

Where δ is the Dirac function, null everywhere except in 0 where it takes the value 1.

Proposition 5.8 *Let p_θ be a parameterized distribution on $\mathcal{X}$.*
Maximizing the likelihood $p_\theta(x)$ is equivalent to minimizing the KL Divergence $D(\hat{p}||p_\theta)$

Proof

$$\begin{aligned} D(\hat{p}||p_\theta) &= \sum_{x\in\mathcal{X}} \hat{p}(x) \log \frac{\hat{p}(x)}{p_\theta(x)} \\ &= -H(\hat{p}) - \sum_{x\in\mathcal{X}} \hat{p}(x)\, \log\, p_\theta(x) \\ &= -H(\hat{p}) - \frac{1}{N}\sum_{x\in\mathcal{X}}\sum_{n=1}^{N} \delta(x - x_n)\, \log\, p_\theta(x) \\ &= -H(\hat{p}) - \frac{1}{N}\sum_{n=1}^{N} \log\, p_\theta(x_n) \end{aligned}$$

The second term is equal to the opposite of the log-likelihood $p_\theta(x)$. Hence the conclusion. ■

Remark 5.1.2 *$p_\theta(x) = 0 \Rightarrow \hat{p}(x) = 0$, but $\hat{p}(x) = 0 \not\Rightarrow p_\theta(x) = 0$. So we should not try to compute $D(p_\theta||\hat{p})$, because this would rule out all the values of x that we have not encountered yet (i.e. such that $\hat{p}(x) = 0$).*

5.1.4 Maximum entropy principle

The maximum entropy principle is a different principle than the maximum likelihood principle and solves a different kind of problem. It assumes that we use the data to specify a constraint on the possible distribution we choose. The idea is to maximize the entropy $H(p)$ under the constraint that $p \in \mathcal{P}(\mathcal{X})$ where $\mathcal{P}(\mathcal{X})$ is a set of possible distribution typically specified from the data.

Let 's consider the following examples

1. A study on kangaroos estimated that $p = 3/4$ of the kangaroos are left-handed and $q = 2/3$ drink Foster beer. What is a reasonable estimate of the fraction of kangaroos that are both left-handed and drink Foster beer? The maximum entropy principle can be invoked to choose among all distributions of pairs of binary random variables. In particular, one way to formalize that we want to choose the least specific distribution that satisfies these constraints is to find the distribution with maximal entropy that satisfies the constraints on the marginals. If X is the variable "is left-handed" and Y "drinks Foster beer", then the problem is formalized as

$$\max_{p_{X,Y}} H(p_{X,Y}) \quad \text{s.t.} \quad p_{X,Y}(1,0) + p_{X,Y}(1,1) = p, \quad p_{X,Y}(0,1) + p_{X,Y}(1,1) = q.$$

   What is the solution to this problem? (Exercise)

2. Among all distributions on $\{1, \ldots, 10\}$ what is the distribution with expected value equal to 2 which has the largest entropy? (Exercise)

3. It is possible to show that the distribution on $\mathbb{R}$ with fixed mean μ and fixed variance σ^2 that has maximal differential entropy is the Gaussian distribution.

4. The principle of maximum entropy is also the principle invoked to construct distribution on angles with fixed mean and variance. It leads to the so-called *wrapped normal distribution*. A related distribution on angle which is also a maximum entropy distribution is the von Mises distribution.

The maximum entropy principle is used often when working with *contingency tables*.

5.1.5 Entropy and KL divergence for continuous random variables

Let X be a continuous random variable taking its values in the continuous space $\mathcal{X}$ and let p be its probability density function. We have the following adapted expressions of entropy and KL Divergence:

- Differential entropy:

$$H_{\text{diff}}(p) = -\int_{\mathcal{X}} p(x)\log(p(x))d\mu(x)$$

- Differential Kullback Leibler Divergence:

$$\begin{aligned} D_{\text{diff}}(p \parallel q) &= \int_{\mathcal{X}} p(x)\log\frac{p(x)}{q(x)}d\mu(x) \\ &= E_{X\sim p}\left[\log\frac{p(X)}{q(X)}\right] \end{aligned}$$

In the continuous case, the entropy is not necessarily non-negative.

Remark 5.1.3 *The definition of $H_{diff}(p)$ depends on the reference measure μ. This means that $H_{diff}(p)$ does not capture any intrinsic properties of p any more, and loses its "physical interpretation" in terms of quantity of information, at least in an absolute sense. By contrast $D_{diff}(p \parallel q)$ does not depend on the choice of the reference measure and has therefore a stronger interpretation.*

5.2 Exponential families

Let $x_1, ..., x_N \in \mathcal{X}$ be N i.i.d. observations of a random variable X.

Definition 5.9 *A statistic Φ is just a function of the data: $x \mapsto \Phi(x) = \Phi(x_1, ..., x_N)$*

Definition 5.10 (Sufficient statistic (statistique exhaustive in French)) *A function $T : x \mapsto T(x)$ is a sufficient statistic for a model $\mathcal{P}_\Theta$ if and only if*

$$\forall\theta \in \Theta, \quad p_\theta(x) = h(x)\, g(T(x);\theta)$$

Note that in order to estimate θ from data x using the maximum likelihood principle the information of the statistics $T(x)$ carries all the information that is relevant.

Another way of interpreting what a sufficient statistic is is to take the Bayesian point of view. In Bayesian statistics, the parameter θ is modelled as a random variable and we then have:

$$p(x,\theta) = p(x|\theta)\, p(\theta) = h(x)\, g(T(x);\theta)\, p(\theta),$$

which means that $\theta \perp\!\!\!\perp X \mid T(X)$.

Definition 5.11 (Exponential family) *Let X be a random variable on $\mathcal{X}$. An exponential family is a family of distribution of the form*

$$p(x;\theta)\, d\mu(x) = h(x)\exp\left\{b(\theta)^T\phi(x) - \tilde{A}(\theta)\right\} d\mu(x),$$

where

- $h(x)$ *the ancillary statistic,*
- $h(x)d\mu(x)$ *the reference measure (or base measure),*
- $\phi(x)$ *the sufficient statistic (also called feature vector),*
- θ *the parameter,*
- $\eta = b(\theta)$ *the canonical parameter,*
- $\tilde{A}(\theta) = A(\eta)$ *the log-partition function.*

Proposition 5.12

$$A(\eta) = \log \int_{\mathcal{X}} h(x) \exp\left\{\eta^T \phi(x)\right\} d\mu(x)$$

Proof

$$1 = \int_{\mathcal{X}} p(x|\eta) d\mu(x) \;=\; e^{-A(\eta)} \int_{\mathcal{X}} h(x) \exp\left\{\eta^T \phi(x)\right\} d\mu(x)$$

■

Definition 5.13 (Canonical exponential family) *A canonical exponential family is an exponential family which such that* $b(\theta) = \theta = \eta$, *i.e.:*

$$p(x;\eta) = h(x) \exp(\eta^T \phi(x) - A(\eta))$$

Definition 5.14 (Domain) *The domain of an exponential family is defined by:*

$$\Omega \;= \{\eta \in \mathbb{R}^p \mid A(\eta) < \infty\}$$

Example 5.2.1 (Multinomial model) *Let* X *be a random variable on* $\mathcal{X} = \{0,1\}^K$. X *follows a multinomial distribution of parameter* $\pi \in [0,1]^K$.

$$\begin{aligned} p(x;\pi) &= \prod_{k=1}^{K} \pi_k^{x_k} \\ &= \exp\Big(\sum_{k=1}^{K} x_k \log \pi_k\Big) \\ &= \exp\Big(\sum_{k=1}^{K} x_k \eta_k\Big) \\ &= \exp(\langle x, \eta \rangle) \end{aligned}$$

In this expression we easily recognize:

- $\eta = (\log \pi_1, \log \pi_2, \cdots, \log \pi_K)^T$;
- $\phi(x) = x$;
- $d\mu(x)$ *the counting measure*
- $h(x) = 1$ *the constant function equal to one;*

But we don't recognize $A(\eta)$*. Let us find it using Proposition 5.12:*

$$\begin{aligned} A(\eta) &= \log\Big(\sum_{x\in\mathcal{X}} \exp(\eta^T x)\Big) \\ &= \log\Big(\sum_{k=1}^{K} \exp(\eta_k)\Big) \end{aligned}$$

$$\begin{aligned} p(x;\eta) &= \exp(\eta^T x - A(\eta)) \\ &= \exp\left(\sum_{k=1}^{K} \eta_k x_k - A(\eta)\right) \\ &= \exp\left(\sum_{k=1}^{K} (\eta_k - A(\eta)) x_k\right) \\ &= \exp\left(\sum_{k=1}^{K} \log\left(\frac{\exp \eta_k}{\sum_{k'=1}^{K} \exp \eta_k}\right) x_k\right) \end{aligned}$$

We see that in the first expression of the likelihood in its exponential form, we did not take into account the fact that $\sum_k \pi_k = 1$*. There was a hidden constraint on* η*. Now we have a new expression for* π_k *and no more constraint over the values that* η *can take:*

$$\tilde{\pi}_k = \frac{\exp(\eta_k)}{\sum_{k'} \exp(\eta_k)}.$$

Example 5.2.2 (Gaussian distribution (μ, σ) **over** $\mathbb{R}$**)**

$$\begin{aligned} p(x;\mu,\sigma^2) &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \\ &= \exp\left\{x^2\left(\frac{-1}{2\sigma^2}\right) + x\frac{\mu}{\sigma^2} - \left[\frac{\mu^2}{2\sigma^2} + \frac{1}{2}\log(2\pi\sigma^2)\right]\right\} \end{aligned}$$

We recognize an exponential family with:

- $\phi(x) = (x,\ x^2)^T$
- $\eta = (\frac{\mu}{\sigma^2},\ -\frac{1}{2\sigma^2})^T = (\eta_1,\ \eta_2)^T$

- $A(\eta) = \frac{1}{2}\log\left(-\frac{2\pi}{2\eta_2}\right) - \frac{\eta_1^2}{4\eta_2}$

$$p(x) = \exp\left\{\phi(x)^T\eta - A(\eta)\right\}$$

on the domain: $\{\eta \in \mathbb{R}^2, \eta_2 < 0\}$.

Example 5.2.3 *Many other common distributions are exponential families: Binomial law, Poisson law ($\mathcal{X} = \mathbb{N}$), Dirichlet law, Gamma law, exponential law.*

5.2.1 Link with the graphical models

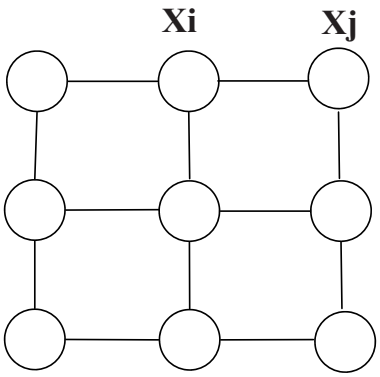


Figure 5.1. Ising model

Example 5.2.4 (Ising model)

$$p_\eta(x) = \frac{1}{Z(\eta)}\exp\sum_{(i,j)\in E}\psi_{ij}(x_i, x_j, \eta)$$

$$\psi_{ij}(x_i, x_j) = V_{ij}^{11}x_ix_j + V_{ij}^{10}x_i(1-x_j) + V_{ij}^{01}(1-x_i)x_j + V_{ij}^{00}(1-x_i)(1-x_j)$$

$$\eta = (V_{ij}^{kk'})_{\substack{(i,j)\in E \\ k,k'\in\{0,1\}}}$$

$$\phi(x) = \begin{pmatrix} x_ix_j \\ (1-x_i)x_j \\ \vdots \end{pmatrix}_{(i,j)\in E}$$

This first expression is overparametrized. We can rewrite the expression with just one parameter per pair (x_i, x_j):

$$p_\eta(x) = \frac{1}{Z}\prod_{(i,j)\in E}\exp\left(\tilde{\eta}_{ij}x_ix_j\right)\prod_{i\in V}\exp\left(\tilde{\eta}_i x_i\right).$$

Example 5.2.5 (General discrete graphical model) *In the general case of a discrete graphical model* ***such that*** $p(x) > 0$ *for all* $x \in \mathcal{X}$*, we have:*

$$\begin{aligned} p(x) &= \frac{1}{Z} \prod_{c \in \mathcal{C}} \Psi_c(x_c) \\ &= \frac{1}{Z} \exp\left\{ \sum_{c \in \mathcal{C}} \log \Psi_c(x_c) \right\} \\ &= \frac{1}{Z} \exp\left\{ \sum_{c \in \mathcal{C}} \sum_{y_c \in \mathcal{X}_c} \delta_{\{y_c = x_c\}} \log(\Psi_c(y_c)) \right\} \end{aligned}$$

Where $\mathcal{X}_c = \{$ *set of all possible values of the r.v. on the clique* $c\}$
We recognize:

$$\Phi(x) = \left(\delta_{(x_c = y_c)}\right)_{\substack{y_c \in \mathcal{X}_c \\ c \in \mathcal{C}}}$$

and

$$\eta = \left(\log(\Psi_c(y_c))\right)_{\substack{y_c \in \mathcal{X}_c \\ c \in \mathcal{C}}}$$

5.2.2 Minimal representation

Remark 5.2.1 *Let* $p_\eta(x) = \exp\left(\eta^\top \phi(x) - A(\eta)\right) h(x) d\mu(x)$.
The set $\mathcal{N}_\eta := \{x : p_\eta(x) = 0\}$ *actually does not depend on* η *but only on* $h(x)$.

Definition 5.15 (Common set of probability zero)

$$\mathcal{N} := \{x : h(x) = 0\}$$

Definition 5.16 (Affinely dependent statistics) *We denote* $\phi(x) = (\phi_1(x), \ldots, \phi_K(x))^\top$.
The sufficient statistics are said to be affinely dependent if:

$$\exists (c_0, \ldots, c_K) \neq 0, \quad \forall x \notin \mathcal{N},\ c_0 + c_1\phi_1(x) + \ldots + c_K\phi_K(x) = 0.$$

Definition 5.17 (Minimal representation of an exponential family) *A vector of sufficient statistics provides a minimal representation of the exponential family these statistics are affinely independent.*

Theorem 5.18 *Every exponential family admits at least one minimal representation (not necessarily unique) of unique minimal dimension K.*

Remark 5.2.2 *We will quite often use redundant (i.e. not minimal) representations.*

5.2.3 Exponential family of an i.i.d. sample

We consider an i.i.d. sample $X_1, \dots, X_n$ distributed according to p_η, which belongs to an exponential family. Then

$$p_\eta(x_1, \dots, x_n) = \prod_{i=1}^n p_\eta(x_i) = \prod_{i=1}^n \left[\exp\left(\eta^\top \phi(x_i) - A(\eta)\right) h(x_i)\right]$$
$$= \exp\left(\eta^\top \Big(\sum_{i=1}^n \phi(x_i)\Big) - nA(\eta)\right) \prod_i h(x_i)$$

1. The sufficient statistics is $n\bar{\phi}$, where $\bar{\phi} = \frac{1}{n}\sum_{i=1}^n \phi(x_i)$,
2. The canonical parameter η and the domain $\Omega = \{\eta \mid A(\eta) < \infty\}$ remain the same as for a single observation,
3. The log-partition function is $nA(\eta)$.

5.2.4 General exponential family

In general, in an exponential family, we can parametrize η with a function b such that $\eta = b(\theta)$ and θ in an open connected subset Θ of $\mathbb{R}^d$.

Definition 5.19 (Curved exponential family) *An exponential family is said to be curved if its Jacobian* $J = \left\{ \frac{\partial b_j(\theta)}{\partial \theta_i} \right\}_{i,j}$ *is not full rank.*

Example 5.2.6 $p_\mu(x) = \mathcal{N}(x; \mu, \mu^2)$

5.2.5 Convexity and differentiability in exponential families

Lemme 5.20 (Hölder's inequality)

$$\forall x, y \in \mathbb{R}^d,\ p, q \geqslant 1 \text{ such that } \frac{1}{p} + \frac{1}{q} = 1$$

$$|x^\top y| \leqslant ||x||_p ||y||_q \quad \text{where } ||x||_p = \left(\sum_{k=1}^n x_k^p \right)^{\frac{1}{p}}.$$

$$\forall f, g : \mathbb{R}^n \to \mathbb{R}, \quad \int |f(x)g(x)|dx \leqslant \left(\int |f(x)|^p dx \right)^{\frac{1}{p}} \left(\int |g(x)|^q dx \right)^{\frac{1}{q}}.$$

Theorem 5.21 (Convexity) *In a canonical exponential family, we have the following properties:*

1. Ω *is a convex subset of* $\mathbb{R}^p$
2. $Z : \eta \mapsto \int \exp\left(\eta^\top \phi(x)\right) h(x)dx$ *is a convex function*
3. $A : \eta \mapsto \log\left(Z(\eta)\right)$ *is a convex function*

Proof If $\Omega = \varnothing$ or Ω is a singleton, the result is trivial.

If not, there exist $\eta_1, \eta_2 \in \Omega$ such that $\eta_1 \neq \eta_2$. Let $\eta = \alpha\,\eta_1 + (1-\alpha)\,\eta_2, \quad \alpha \in]0,1[$.

$$\begin{array}{rcll} \exp(\eta^\top\phi(x)) & \leqslant & \alpha\exp(\eta_1^\top\phi(x)) & + \quad (1-\alpha)\exp(\eta_2^\top\phi(x)) \\ \displaystyle\int \ldots h(x)d\mu(x) & \leqslant & \displaystyle\alpha\int \ldots h(x)d\mu(x) & + \quad (1-\alpha)\int \ldots h(x)d\mu(x) \\ Z(\eta) & \leqslant & \alpha\, Z(\eta_1) & + \quad (1-\alpha)\, Z(\eta_2). \end{array}$$

Thus Z is a convex function. Moreover:

$$\eta_1, \eta_2 \in \Omega \quad \Rightarrow \quad Z(\eta) \leqslant \alpha Z(\eta_1) + (1-\alpha)Z(\eta_2) < \infty \quad \Rightarrow \quad \eta \in \Omega$$

which proves that Ω is a convex set.

$$Z(\eta) = \int \exp\left(\eta^\top\phi(x)\right) h(x)d\mu(x) = \int \underbrace{(\exp\eta_1^\top\phi(x))^\alpha h(x)^\alpha}_{f(x)^\alpha} \underbrace{(\exp\eta_2^\top\phi(x))^{1-\alpha} h(x)^{1-\alpha}}_{g(x)^{(1-\alpha)}} d\mu(x)$$

By taking $p = \frac{1}{\alpha}$, we obtain:

$$\begin{array}{rcll} \displaystyle\int f(x)^\alpha g(x)^{1-\alpha} d\mu(x) \leqslant & \multicolumn{2}{l}{\displaystyle\left(\int f(x)^{\alpha p} d\mu(x)\right)^{\frac{1}{p}} \left(\int g(x)^{(1-\alpha)q} d\mu(x)\right)^{\frac{1}{q}}} \\ Z(\eta) \leqslant & Z(\eta_1)^\alpha & Z(\eta_2)^{1-\alpha} \\ A(\eta) = \log(Z(\eta)) \leqslant & \alpha A(\eta_1) & +(1-\alpha)A(\eta_2). \end{array}$$

Hence A is a convex function. ■

Corollary 5.22 *In a canonical exponential family, the maximum likelihood estimator is the solution of a convex optimization problem.*

Proof The log-likelihood is concave:

$$\ell(\eta) = \log p_\eta(x) = \eta^\top \bar{\phi}(x) - A(\eta) + \log h(x).$$

■

Remark 5.2.3 *The theorem does not hold in any of those two cases:*

1. *The family is curved,*
2. *ϕ is not fully observed and we consider the marginal likelihood of the observations.*

Theorem 5.23 *If $\eta \in \overset{\circ}{\Omega}$, then Z is $\mathcal{C}^\infty$ (and so is A) and:*

$$\frac{\partial Z}{\partial \eta_k} = \mathbb{E}_\eta[\phi_k(x)]Z(\eta)$$

$$\frac{\partial^m}{\partial \eta_1^{m_1} \dots \partial \eta_K^{m_K}} Z(\eta) = \mathbb{E}_\eta[\phi_1(x)^{m_1} \dots \phi_K(x)^{m_K}]Z(\eta)$$

Proof It is a bit technical but standard to show using the dominated convergence theorem that one can exchange differentiation and expectation in the computations of the differentials of Z. One then has

$$\begin{aligned}
\frac{\partial Z}{\partial \eta_k} &= \int \phi_k(x) \exp\left\{\eta^\top \phi(x)\right\} h(x) d\mu(x) \\
&= \int \phi_k(x) \exp\left\{\eta^\top \phi(x) - A(\eta)\right\} h(x) d\mu(x) \underbrace{exp(A(\eta))}_{Z(\eta)} \\
&= \mathbb{E}_\eta[\phi_k(x)]Z(\eta),
\end{aligned}$$

which proves the first formula (the general one can be deduced by induction). ■

Directed and Undirected Graphical Models **2013/2014**

Lecture 6 — November 6th

Lecturer: Guillaume Obozinski *Scribe: Lucas Plaetevoet, Ismael Belghiti*

6.1 Moment vector

Definition 6.1 (Moment vector) *We define the* moment vector *(or* moment parameter *as:*

$$\mu(\eta) = \nabla A(\eta) = E_\eta[\phi(X)].$$

6.1.1 Examples of moment vectors

Bernoulli

For a Bernoulli distribution, we can write:

$$p(x) = \pi^x(1-\pi)^{1-x} = e^{x\log\pi - x\log(1-\pi) + \log(1-\pi)} = e^{x\eta - A(\eta)}$$

with $\eta = \log\frac{\pi}{1-\pi}$ and $A(\eta) = -\log(1-\pi)$.

From this we get that $\pi = (1-\pi)e^\eta$ and thus $\pi = \frac{e^\eta}{1+e^\eta} = \frac{1}{1+e^{-\eta}} = \sigma(\eta)$. Remark that in logistic regression we have $\eta = w^\top x$.

Moreover, we can write $A(\eta) = -\log(1-\pi) = \log(1+e^\eta)$ and the moment vector is:

$$\mu(\eta) = E_\eta[\phi(X)] = E_\eta[X] = \pi.$$

Multinomial

In the multinomial case we consider $Z \to \{0,1\}^k$. We have $\phi(Z) = \begin{pmatrix} Z_1 \\ \vdots \\ Z_k \end{pmatrix}$ and the moment vector is:

$$\mu(\eta) = E_\eta[\phi(Z)] = \begin{pmatrix} \pi_1 \\ \vdots \\ \pi_k \end{pmatrix}.$$

Gaussian

In the gaussian model, we have $\phi(x) = \begin{pmatrix} x \\ x^2 \end{pmatrix}$ and we obtain:

$$\mu(\eta) = E_\eta \begin{bmatrix} x \\ x^2 \end{bmatrix} = \begin{pmatrix} \mu \\ \sigma^2 + \mu^2 \end{pmatrix}$$

6.2 Hessian of A

Proposition 6.2 *The hessian of A is the covariance matrix of the sufficient statistic:*

$$\nabla^2 A(\eta) = E[(\phi(X) - \mu(\eta))(\phi(X) - \mu(\eta))^\top] = Cov(\phi(X))$$

Proof We can write:

$$\begin{aligned}\nabla^2 A(\eta) = \nabla\nabla A(\eta) = \nabla\left(\frac{\nabla Z(\eta)}{Z(\eta)}\right) &= \frac{\nabla^2 Z(\eta)}{Z(\eta)} + \nabla Z(\eta)\left(\frac{-\nabla Z(\eta)}{Z(\eta)^2}\right)^\top \\ &= \frac{\nabla^2 Z(\eta)}{Z(\eta)} - \left(\frac{\nabla Z(\eta)}{Z(\eta)}\right)\left(\frac{\nabla Z(\eta)}{Z(\eta)}\right)^\top\end{aligned}$$

Moreover we have $[\nabla^2 Z(\eta)]_{k,k'} = E\left[\phi_k(X)\phi_{k'}(X)\right] Z(\eta)$ ie:

$$\nabla^2 Z(\eta) = E[\phi(X)\phi(X)^\top] Z(\eta).$$

Consequently:

$$\begin{aligned}\nabla^2 A(\eta) &= E[\phi(X)\phi(X)^\top] - \mu(\eta)\mu(\eta)^\top \\ &= E[(\phi(X) - \mu(\eta))(\phi(X) - \mu(\eta)^\top] \\ &= Cov(\phi(X))\end{aligned}$$

■

Remark: Z can be seen as a moment generating function $t \to Z(\eta + t)$ and A as the cumulative generating function $t \to A(\eta + t)$.

Corollary 6.3 *We have the three following properties:*

1. $\nabla^2 A(\eta) \succeq 0$ *(semi-positive definite).*
2. *A is convex.*
3. *A is strictly convex on $\mathring{\Omega}$ if, and only if, $\phi(X)$ is a minimal representation of the exponential family.*

Proof

1. $\forall c, c^\top \nabla^2 A(\eta) c = E[c^\top(\phi - \mu)(\phi - \mu)^\top c] = E[((\phi - \mu)^\top c)^2] \geq 0$
2. Since $\nabla^2 A \succeq 0$, A is convex.
3. If A is not strictly convex, then there exists η and c such that $c^\top \nabla^2 A(\eta) c = 0$ therefore, for all x, $\text{Var}(c^\top \phi(x)) = 0$ thus $c^\top \phi(x) = -c_o$. We can thus write: $\forall x, c_0 + c_1\phi_1(x) + \ldots + c_k\phi_k(x) = 0$. Since we can go backward, we have the equivalence.

■

6.3 Log-Likelihood of an exponential function

Denoting $\bar{\phi} = \frac{1}{n}\sum_i \phi(x_i)$, we have:

$$-l(\eta) = -\eta^\top \bar{\phi} n + nA(\eta)$$

and

$$-\nabla l(\eta) = -\bar{\phi} n + n\mu(\eta).$$

Consequently, we have the following equivalence:

$$\nabla l(\eta) = 0 \Leftrightarrow \mu(\eta) = \bar{\phi}$$

Theorem 6.4 *The maximum likelihood estimator η is such that $\bar{\phi} = \mu(\eta)$. This result is called "Moment Matching".*

$$\boxed{\bar{\phi} = E_\eta[\phi(x)] = \mu(\eta)}$$

$$\eta \overset{\text{inference}}{\underset{\text{learning}}{\rightleftarrows}} \mu(\eta) = \bar{\phi}$$

6.4 Link between Maximum Likelihood and Maximum Entropy

The Maximum Entropy principle can be applied: we want to find the disribution p such that $E[\phi(X)] = \bar{\phi}$ and has maximal entropy.

We can write this as a convex optimization problem:

$$\begin{array}{ll} \underset{p}{\text{Minimize}} & -H(p) \\ \text{subject to} & \begin{cases} E_p[\phi(X)] = \bar{\phi} \\ p(x) \geq 0 \\ \sum\limits_x p(x) = 1 \end{cases} \end{array}$$

Let us introduce the corresponding Lagrangian:

$$\mathcal{L}(p, \lambda, c) = \sum_x p(x) \log p(x) - \lambda^\top \left(\sum_x p(x)\phi(x) - \bar{\phi} \right) + c \left(\sum_x p(x) - 1 \right)$$

Since the problem is convex, we have strong duality:

$$\min_p \max_{\lambda, c} \mathcal{L}(p, \lambda, c) = \max_{\lambda, c} \min_p \mathcal{L}(p, \lambda, c)$$

Slater's condition corresponds to the existence of p in the relative interior of the domain of the function that is in $\mathbb{R}_{+*}^{|X|}$ and such that $\sum_{x \in \mathcal{X}} p(x) = 1$. If we do not find such a p then we can reduce our set taken $\mathcal{X}' = \mathcal{X} \setminus \{x | p(x) = 0\}$.

Without loss of generality, we can hence assume that $p > 0$ and that the moment condition holds. The gradient of the Lagrangian with respect to p is given by:

$$\nabla_p \mathcal{L}(p, \lambda, c) = \log p(x) + 1 - \lambda^\top \phi(x) + c$$

and we have:

$$\begin{aligned} \nabla_p \mathcal{L} = 0 &\Leftrightarrow \log p(x) = \lambda^\top \phi(x) - (c+1) \\ &\Leftrightarrow p(x) = C e^{\lambda^\top \phi(x)} \text{ with } C = e^{-(c+1)} \end{aligned}$$

We recognize here an exponential family. Reinjecting this value of p and maximizing with respect to λ and c, we obtain the maximum likelihood estimator.

Theorem 6.5 *If $X_1, \ldots, X_n$ is an iid sample and $\phi(X)$ a statistic, then the maximum entropy estimator satisfying the equality $E_p[\phi(X)] = \bar{\phi}$ is the maximum likelihood distribution in the exponential family with sufficient statistic ϕ.*

6.5 Gaussian graphical models

6.5.1 Canonical parameterization

We consider a Gaussian random variable $X \in \mathbb{R}^p : X \sim \mathcal{N}(\mu, \Sigma)$ with $\mu \in \mathbb{R}^p$, $\Sigma \in \mathbb{R}^{p \times p}$, $\Sigma \succ 0$. We recall the expression of its density:

$$p(x, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^p |\Sigma|}} \exp\left[-\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right]$$

Denoting $\eta = \Sigma^{-1}\mu$ et $\Lambda = \Sigma^{-1}$ we get:

$$\begin{aligned}
(x-\mu)^T \Sigma^{-1} (x-\mu) &= x^T\Sigma^{-1}x - x\mu^T\Sigma^{-1}x + \mu^T\Sigma^{-1}\mu \\
&= x^T\Lambda x - 2\eta^T x + \eta^T\Lambda^{-1}\eta \\
p(x,\mu,\Lambda) &= \exp\left[\eta^T x - \frac{1}{2}x^T\Lambda x - A(\eta,\Lambda)\right] \\
A(\eta,\Lambda) &= \frac{1}{2}\eta^T\Lambda^{-1}\eta + \frac{p}{2}\log 2\pi - \frac{1}{2}\log|\Lambda|
\end{aligned}$$

$\theta = \{\Lambda, \eta\}$ are the canonical parameters. Λ is called the *precision matrix*, and η is the *loading vector*. We have the following sufficient statistic, which is not a minimal representation:

$$\Phi(x) = \begin{pmatrix} x \\ -\frac{1}{2}\mathrm{Vec}\left(xx^T\right) \end{pmatrix}$$

Mean and covariance

The mean and covariance of X are given by :

$$\begin{aligned}
\nabla_\theta A(\eta,\Lambda) &= \mathbb{E}_\theta[\Phi(X)] \\
&= \begin{pmatrix} \mathbb{E}_\theta[X] \\ -\frac{1}{2}\mathbb{E}_\theta\left[XX^T\right] \end{pmatrix} \\
\mathbb{E}_\theta[X] &= \nabla_\eta A(\eta,\Lambda) \\
&= \Lambda^{-1}\eta \\
&= \mu \\
-\frac{1}{2}\mathbb{E}_\theta\left[XX^T\right] &= \nabla_\Lambda A(\eta,\Lambda) \\
&= -\frac{1}{2}\Lambda^{-1}\eta\eta^T\Lambda^{-1} - \frac{1}{2}\Lambda^{-1} \\
&= -\frac{1}{2}\left[\mu\mu^T + \Lambda^{-1}\right]
\end{aligned}$$

Hence

$$\begin{aligned}
\mathrm{Cov}[X] &= \mathbb{E}_\theta\left[XX^T\right] - \mathbb{E}_\theta[X]\,\mathbb{E}_\theta[X]^T \\
&= \Lambda^{-1} \\
&= \Sigma
\end{aligned}$$

Please note that we could have also computed the covariance with:

$$\nabla^2_\theta A(\eta,\Lambda) = \begin{pmatrix} \mathrm{Cov}(X) & \cdots \\ \cdots & \mathrm{Cov}(\mathrm{Vec}(XX^T)) \end{pmatrix}$$

and $\nabla^2_\eta A(\eta,\Lambda) = \Lambda^{-1}$

6.5.2 Conditioning and marginalization in Gaussian GM

We partition the random variable $X \in \mathbb{R}^p$ into two components $X_1 \in \mathbb{R}^{p_1}$ and $X_2 \in \mathbb{R}^{p_2}$ such that $X = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$ and $p = p_1 + p_2$. We now seek to determine the law of X_1 and $X_2|X_1$.

$$X_1 \sim ?, \quad X_2|X_1 \sim ?$$

Before doing so, we need to partition the moment parameters μ, Σ and the canonical parameters Λ, η in the same way:

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \quad \Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}, \quad \eta = \begin{pmatrix} \eta_1 \\ \eta_2 \end{pmatrix}, \quad \Lambda = \Sigma^{-1} = \begin{pmatrix} \Lambda_{11} & \Lambda_{12} \\ \Lambda_{21} & \Lambda_{22} \end{pmatrix}.$$

from which we get a partitioned form for the joint distribution:

$$p(x_1, x_2) = \frac{1}{(2\pi)^{p/2} |\Sigma|^{1/2}} \exp\left[-\frac{1}{2} \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix}^T \Lambda \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix} \right] \tag{6.1}$$

In what follows, we will introduce a tool to block diagonalize partitioned matrices. We will then be able to develop general formulas for marginalization and conditioning in the multivariate Gaussian setting.

6.5.3 Digression on Schur complement

Let us consider the block matrix $M = \begin{pmatrix} A & L \\ R & U \end{pmatrix}$. Our goal is to explicit the blocks of its inverse in terms of the initial blocks A, L (L stands for left), U (U stands for upper) and R (R stands for right).

We can zero out the L and R by *premultiplying* M by D and *postmultiplying* by D. We denote Δ this block diagonal matrix.

$$\begin{aligned} \begin{pmatrix} I & 0 \\ -RA^{-1} & I \end{pmatrix} \times \begin{pmatrix} A & L \\ R & U \end{pmatrix} \times \begin{pmatrix} I & -A^{-1}L \\ 0 & I \end{pmatrix} &= D \times M \times G \\ &= \begin{pmatrix} I & 0 \\ -RA^{-1} & I \end{pmatrix} \times \begin{pmatrix} A & 0 \\ R & U - RA^{-1}L \end{pmatrix} \\ \Delta &= \begin{pmatrix} A & 0 \\ 0 & U - RA^{-1}L \end{pmatrix} \end{aligned}$$

Definition 6.6 *The* Schur complement *of the matrix* $M = \begin{pmatrix} A & L \\ R & U \end{pmatrix}$ *with respect to* A *is* $\left[M_{/A}\right] = U - RA^{-1}L$.

By symmetry we obtain the Schur complement of M with respect to U: $\left[M_{/U}\right] = A - LU^{-1}R$

Lemme 6.7 (Determinant lemma)

$$|M| = |A| \times \left|\left[M_{/A}\right]\right| = |U| \times \left|\left[M_{/U}\right]\right|$$

Proof

$$|\Delta| = \underbrace{|D|}_{=1} |M| \underbrace{|G|}_{=1} = |M|$$

and we have also

$$|\Delta| = |A| \left|\left[M_{/A}\right]\right|$$

and

$$|\Delta| = |U| \left|\left[M_{/U}\right]\right|$$

■

Lemme 6.8 (Positivity lemma) *If M is symmetric then $M \succcurlyeq 0$ if and only if $A \succcurlyeq 0$ and $\left[M_{/A}\right] \succcurlyeq 0$.*

Please note that we have the same lemma for strict inequalities.

Proof $G = D^T$. $A \succcurlyeq 0$ and $\left[M_{/A}\right] \succcurlyeq 0 \Leftrightarrow \forall x,\, x^T \Delta x \geqslant 0 \Leftrightarrow \forall x, \left(D^T x\right)^T M \left(D^T x\right) \geqslant 0$, hence $\forall y, y^T M y \geqslant 0$ because $G = D^T$ is invertible.

■

Woodbury-Sherman-Morrison inversion formula for partitioned matrices

We have that M is invertible if and only if $A \succcurlyeq 0$ and $\left[M_{/A}\right] \succcurlyeq 0$. Then $\Delta^{-1} = G^{-1} M^{-1} D^{-1}$, and $M^{-1} = G \Delta^{-1} D$. The explicit computation of this matrix product gives the so-called Woodbury-Sherman-Morrison formula:

$$\begin{aligned} M^{-1} &= \begin{pmatrix} I & -A^{-1}L \\ 0 & I \end{pmatrix} \times \begin{pmatrix} A^{-1} & 0 \\ 0 & \left[M_{/A}\right]^{-1} \end{pmatrix} \times \begin{pmatrix} I & 0 \\ -RA^{-1} & I \end{pmatrix} \\ &= \begin{pmatrix} A^{-1} + A^{-1}L \left[M_{/A}\right]^{-1} RA^{-1} & -A^{-1}L \left[M_{/A}\right]^{-1} \\ -\left[M_{/A}\right]^{-1} RA^{-1} & \left[M_{/A}\right]^{-1} \end{pmatrix} \end{aligned} \tag{6.2}$$

Similarly we obtain:

$$M^{-1} = \begin{pmatrix} \left[M_{/U}\right]^{-1} & -U^{-1}R \left[M_{/U}\right]^{-1} \\ -\left[M_{/U}\right]^{-1} LU^{-1} & U^{-1} + U^{-1}R \left[M_{/U}\right]^{-1} LU^{-1} \end{pmatrix}$$

6.5.4 Back to the problem

We now use the Woodbury formula (6.2) to compute an interesting expression for the quadratic form of the multivariate Gaussian distribution.

$$\begin{aligned}(x-\mu)^T\Sigma^{-1}(x-\mu) &= \begin{pmatrix} x_1-\mu_1 \\ x_2-\mu_2 \end{pmatrix}^T \begin{pmatrix} I & -\Sigma_{11}^{-1}\Sigma_{12} \\ 0 & I \end{pmatrix} ... \\ &\times \begin{pmatrix} \Sigma_{11}^{-1} & 0 \\ 0 & \left[\Sigma_{/\Sigma_{11}}\right]^{-1} \end{pmatrix} \times \begin{pmatrix} I & 0 \\ -\Sigma_{21}\Sigma_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} x_1-\mu_1 \\ x_2-\mu_2 \end{pmatrix} \\ &= (x_1-\mu_1)^T(x_1-\mu_1) + (x_2-\mu_2-b)^T \left[\Sigma_{/\Sigma_{11}}\right]^{-1}(x_2-\mu_2-b)\end{aligned} \tag{6.3}$$

where we denoted $b = \Sigma_{21}\Sigma_{11}^{-1}(x_1-\mu_1)$.
Now recall that we have $|\Sigma| = |\Sigma_{11}|\left|\left[\Sigma_{/\Sigma_{11}}\right]\right|$. The joint distribution can be expressed as:

$$\begin{aligned}p(x_1,x_2) = &\underbrace{\frac{1}{\sqrt{(2\pi)^{p_1}|\Sigma_{11}|}}\exp\left[-\frac{1}{2}\left((x_1-\mu_1)^T\Sigma_{11}^{-1}(x_1-\mu_1)\right)\right]}_{p(x_1)} \times ... \\ &\underbrace{\frac{1}{\sqrt{(2\pi)^{p_1}\left|\left[\Sigma_{/\Sigma_{11}}\right]\right|}}\exp\left[-\frac{1}{2}\left((x_2-\mu_2-b)\left[\Sigma_{/\Sigma_{11}}\right]^{-1}(x_2-\mu_2-b)\right)\right]}_{p(x_2|x_1)}\end{aligned} \tag{6.4}$$

From (6.4) we deduce that $X_1 \sim \mathcal{N}(\mu_1,\Sigma_{11})$, et $X_2|X_1 \sim \mathcal{N}\left(\mu_2+b,\left[\Sigma_{/\Sigma_{11}}\right]\right)$.

We denote by (μ_1,Σ_1), respectively $(\mu_{2|1},\Sigma_{2|1})$, the moment parameters of the marginal distribution of x_1, respectively the moment parameters of the conditional distribution of x_2 given x_1. We have a similar notation for the canonical parameters η and Λ. We summarize our results in the following:

Moment parameterization summary

$$\begin{cases} \mu_1 = \mu_1 \\ \Sigma_1 = \Sigma_{11} \\ \mu_{2|1} = \mu_2 + b = \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(x_1-\mu_1) \\ \Sigma_{2|1} = \left[\Sigma_{/\Sigma_{11}}\right] = \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12} \end{cases}$$

Canonical parameterization summary

$$\begin{cases} \eta_1 = \left[\Lambda_{/\Lambda_{22}}\right]\mu_1 = \eta_2 - \Lambda_{12}\Lambda_{22}^{-1}\eta_2 \\ \Lambda_1 = \Sigma_{11}^{-1} = \left[\Lambda_{/\Lambda_{22}}\right] = \Lambda_{11} - \Lambda_{12}\Lambda_{22}^{-1}\Lambda_{21} \\ \eta_{2|1} = \Lambda_{22|1} \times \mu_{2|1} = \Lambda_{22}\mu_2 - \Lambda_{21}(x_1-\mu_1) = \eta_2 - \Lambda_{21}x_1 \\ \Lambda_{22|1} = \Lambda_{22} \end{cases}$$

We can notice that in the moment parameterization, the marginalization operation is simple and the conditioning is complicated and the opposite holds in the canonical parameterization.

6.5.5 Zeros of the precision matrix and Markov properties

Let $p(x_1, ..., x_p)$ a joint Gaussian distribution. We denote $I = \{i, j\}$ and we consider $p(x_i, x_j|x_B)$, with $B = \{1, \dots, p\} \setminus \{i, j\}$. Using the canonical parameterization:

$$\eta_I|B = \begin{pmatrix} \eta_i - \Lambda_{iB}x_B \\ \eta_j - \Lambda_{jB}x_B \end{pmatrix} \qquad \text{and} \qquad \Lambda_{II|B} = \Lambda_{II} = \begin{pmatrix} \lambda_{ii} & \lambda_{ij} \\ \lambda_{ji} & \lambda_{jj} \end{pmatrix}$$

we have the following expression for the covariance matrix of $X_I|X_B$:

$$\text{Cov}\left(X_I|X_B\right) = \Sigma_{II|B} = \Lambda^{-1}_{II|B} = \frac{1}{|\Lambda_{II}|}\begin{pmatrix} \lambda_{jj} & -\lambda_{ji} \\ -\lambda_{ij} & \lambda_{ii} \end{pmatrix}$$

Hence $\text{Cov}\left(x_i, x_j|X_B\right) = \frac{-\lambda_{ij}}{\sqrt{\lambda_{ii}\times\lambda_{jj}}}$ and $\lambda_{ij} = 0 \Rightarrow X_i \perp X_j \mid X_B$.

Proposition 6.9 *The non zero coefficients in Λ correspond to edges in the underlying graphical model.*

Indeed, the distribution is proportional to $\exp(\eta^T - \frac{1}{2}x\Lambda x^T) = \prod_i \exp(\eta_i x_i) \prod_{ij} \exp(-\frac{1}{2}x_i\lambda_{ij}x_j)$

6.5.6 Matrix inversion lemma

A useful consequence of the Schur component is to prove rigorously the following inversion lemma:

Lemme 6.10 *(Matrix inversion) Let $X \in \mathbb{R}^{p\times n}$*

$$\left(\text{Id} + \lambda\, X^T X\right)^{-1} = \text{Id} - \lambda X \left(\text{Id} + \lambda\, X X^T\right)^{-1} X^T$$

In practice, we often want to invert matrix such as $\left(\text{Id} + \lambda X^T X\right)$ where $X \in \mathbb{R}^{p\times n}$ is a *design* matrix. n represents an i.i.d sample while p represents the features, and we usually have $n \gg p$. In that case, the inversion lemma 6.10 replaces the problem of inverting a $n \times n$ matrix (complexity in $O(n^3)$) by a less costly one: inverting a $p \times p$ matrix.

Proof We consider $M = \begin{pmatrix} \text{Id} & X \\ X^T & -\frac{1}{\lambda}\text{Id} \end{pmatrix} = \begin{pmatrix} A & L \\ R & U \end{pmatrix}$, then $\left[M_{/U}\right]^{-1} = \left(\text{Id} + \lambda X^T X\right)$. Recall the Woodbury formula (6.2), we have:

$$\left[M_{/U}\right]^{-1} = A^{-1} + A^{-1}L\left[M_{/A}\right]^{-1} RA^{-1}$$

which gives us the inversion lemma since here $\left[M_{/U}\right]^{-1} = \text{Id} + X\left(-\frac{1}{\lambda}\text{Id} - XX^T\right)^{-1} X^T$.

■

Sum Product Algorithm and Hidden Markov Model **2014/2015**

Cours 7 — 12th November 2014

Enseignant: Francis Bach *Scribe: Pauline Luc, Mathieu Andreux*

7.1 Sum Product Algorithm

7.1.1 Motivations

Inference, along with *estimation* and *decoding*, are the three key operations one must be able to perform efficiently in graphical models.

Given a discrete Gibbs model of the form: $p(x) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C)$, where $\mathcal{C}$ is the set of cliques of the graph, inference enables:

- Computation of the marginal $p(x_i)$ or more generally, $p(x_C)$.
- Computation of the partition function Z
- Computation of the conditional marginal $p(x_i | X_j = x_j, X_k = x_k)$

And as a consequence

- Computation of the gradient in a exponential family
- Computation of the expected value of the loglikelihood of an exponential family at step E of the EM algorithm (for example for HMM)

Example 1: Ising model

Let $X = (X_i)_{i \in V}$ be a vector of random variables, taking value in $\{0,1\}^{|V|}$, of which the exponential form of the distribution is:

$$p(x) = e^{-A(\eta)} \prod_{i \in V} e^{\eta_i x_i} \prod_{(i,j) \in E} e^{\eta_{i,j} x_i x_j} \tag{7.1}$$

We then have the log-likelihood:

$$l(\eta) = \sum_{i \in V} \eta_i x_i + \sum_{(i,j) \in E} \eta_{i,j} x_i x_j - A(\eta) \tag{7.2}$$

We can therefore write the sufficient statistic:

$$\phi(x) = \begin{pmatrix} (x_i)_{i \in V} \\ (x_i x_j)_{(i,j) \in E} \end{pmatrix} \tag{7.3}$$

But we have seen that for exponential families:

$$l(\eta) = \phi(x)^T \eta - A(\eta) \tag{7.4}$$
$$\nabla_\eta \, l(\eta) = \phi(x) - \underbrace{\nabla_\eta \, A(\eta)}_{\mathbb{E}_\eta[\phi(X)]} \tag{7.5}$$

We therefore need to compute $\mathbb{E}_\eta[\phi(X)]$. In the case of the Ising model, we get:

$$\mathbb{E}_\eta[X_i] = \mathbb{P}_\eta[X_i = 1] \tag{7.6}$$
$$\mathbb{E}_\eta[X_i X_j] = \mathbb{P}_\eta[X_i = 1, X_j = 1] \tag{7.7}$$

The equation 7.6 is one of the motivations for solving the problem of inference. In order to be able to compute the gradient of the log-likelihood, we need to know the marginal laws.

Example 2: Potts model

X_i are random variables, taking value in $\{1, \dots, K_i\}$. We note Δ_{ik} the random variable such that $\Delta_{ik} = 1$ if and only if $X_i = k$. Then,

$$p(\delta) = \exp\left[\sum_{i \in V} \sum_{k=1}^{K_i} \eta_{i,k} \delta_{ik} + \sum_{(i,j) \in E} \sum_{k=1}^{K_i} \sum_{k'=1}^{K_j} \eta_{i,j,k,k'} \delta_{ik} \delta_{jk'} - A(\eta)\right] \tag{7.8}$$

and

$$\phi(\delta) = \begin{pmatrix} (\delta_{ik})_{i,k} \\ (\delta_{ik}\delta_{jk'})_{i,j,k,k'} \end{pmatrix} \tag{7.9}$$
$$\mathbb{E}_\eta[\Delta_{ik}] = \mathbb{P}_\eta[X_i = k] \tag{7.10}$$
$$\mathbb{E}_\eta[\Delta_{ik}\Delta_{jk'}] = \mathbb{P}_\eta[X_i = k, X_j = k'] \tag{7.11}$$

These examples illustrate the need to perform inference.

Problem: In general, the inference problem is NP-hard.

For trees the inference problem is efficient as it is linear in n.

For "tree-like" graphs we use the *Junction Tree Algorithm* which enables us to bring the situation back to that of a tree.

In the general case we are forced to carry out approximative inference.

7.1.2 Inference on a chain

We define X_i a random variable, taking value in $\{1, \dots, K\}$, $i \in V = \{1, \dots, n\}$ with joint distribution $p(x)$ defined as:.

$$p(x) = \frac{1}{Z} \prod_{i=1}^{n} \psi_i(x_i) \prod_{i=2}^{n} \psi_{i-1,i}(x_{i-1}, x_i) \tag{7.12}$$

We wish to compute $p(x_j)$ for a certain j. The naive solution would be to compute the marginal

$$p(x_j) = \sum_{x_{V \setminus \{j\}}} p(x_1, \dots, x_n) \tag{7.13}$$

Unfortunately, this type of calculation is of complexity $O(K^n)$. We therefore develop the expression

$$p(x_j) = \frac{1}{Z} \sum_{x_{V \setminus \{j\}}} \prod_{i=1}^{n} \psi_i(x_i) \prod_{i=2}^{n} \psi_{i-1,i}(x_{i-1}, x_i) \tag{7.14}$$

$$= \frac{1}{Z} \sum_{x_{V \setminus \{j\}}} \prod_{i=1}^{n-1} \psi_i(x_i) \prod_{i=2}^{n-1} \psi_{i-1,i}(x_{i-1}, x_i) \psi_n(x_n) \psi_{n-1,n}(x_{n-1}, x_n) \tag{7.15}$$

$$= \frac{1}{Z} \sum_{x_{V \setminus \{j,n\}}} \sum_{x_n} \prod_{i=1}^{n-1} \psi_i(x_i) \prod_{i=2}^{n-1} \psi_{i-1,i}(x_{i-1}, x_i) \psi_n(x_n) \psi_{n-1,n}(x_{n-1}, x_n) \tag{7.16}$$

Which allows us to bring out the messaged passed by (n) to $(n-1)$: $\mu_{n \to n-1}(x_{n-1})$. When continuing, we obtain:

$$p(x_j) = \frac{1}{Z} \sum_{x_{V \setminus \{j,n\}}} \prod_{i=1}^{n-1} \psi_i(x_i) \prod_{i=2}^{n-1} \psi_{i-1,i}(x_{i-1}, x_i) \underbrace{\sum_{x_n} \psi_n(x_n) \psi_{n-1,n}(x_{n-1}, x_n)}_{\mu_{n \to n-1}(x_{n-1})} \tag{7.17}$$

$$= \frac{1}{Z} \sum_{x_{V \setminus \{j,n,n-1\}}} \prod_{i=1}^{n-2} \psi_i(x_i) \prod_{i=2}^{n-2} \psi_{i-1,i}(x_{i-1}, x_i) \times$$
$$\times \underbrace{\sum_{x_{n-1}} \psi_{n-1}(x_{n-1}) \psi_{n-2,n-1}(x_{n-2}, x_{n-1}) \mu_{n \to n-1}(x_{n-1})}_{\mu_{n-1 \to n-2}(x_{n-2})} \tag{7.18}$$

$$= \frac{1}{Z} \sum_{x_{V \setminus \{1,j,n,n-1\}}} \mu_{1 \to 2}(x_2) \ \dots \ \mu_{n-1 \to n-2}(x_{n-2}) \tag{7.19}$$

In the above equation, we have implicitely used the following definitions for descending and ascending messages:

$$\mu_{j\to j-1}(x_{j-1}) = \sum_{x_j} \psi_j(x_j)\psi_{j-1,j}(x_{j-1}, x_j)\mu_{j+1\to j}(x_j) \tag{7.20}$$

$$\mu_{j\to j+1}(x_{j+1}) = \sum_{x_j} \psi_j(x_j)\psi_{j,j+1}(x_j, x_{j+1})\mu_{j-1\to j}(x_j) \tag{7.21}$$

Each of these messages is computed with complexity $O((n-1)K^2)$. And finally, we get

$$\boxed{p(x_j) = \frac{1}{Z}\ \mu_{j-1\to j}(x_j)\ \psi_j(x_j)\ \mu_{j+1\to j}(x_j)} \tag{7.22}$$

With only $2(n-1)$ messages, we have calculated $p(x_j)\ \forall j \in V$. Z is otained by summing

$$Z = \sum_{x_i}\ \mu_{i-1\to i}(x_i)\ \psi_i(x_i)\ \mu_{i+1\to i}(x_i) \tag{7.23}$$

7.1.3 Inference in undirected trees

We note i the vertice of which we want to compute the marginal law $p(x_i)$. We set i to be the root of our tree. $\forall\ j \in V$, we note $\mathcal{C}(j)$ the set of childen of j and $\mathcal{D}(j)$ the set of descendants of j. The joint probability is:

$$p(x) = \frac{1}{Z}\prod_{i\in V}\psi_i(x_i)\prod_{(i,j)\in E}\psi_{i,j}(x_i, x_j) \tag{7.24}$$

For a tree with at least two vertices, we define by reccurence,

$$F(x_i, x_j, x_{\mathcal{D}(j)}) \triangleq \psi_{i,j}(x_i, x_j)\psi_j(x_j)\prod_{k\in\mathcal{C}(j)} F(x_j, x_k, x_{\mathcal{D}(k)}) \tag{7.25}$$

Then by reformulating the marginal:

$$p(x_i) = \frac{1}{Z} \sum_{x_{V\setminus\{i\}}} \psi_i(x_i) \prod_{j\in\mathcal{C}(i)} F(x_i, x_j, x_{\mathcal{D}(j)}) \tag{7.26}$$

$$= \frac{1}{Z}\, \psi_i(x_i) \prod_{j\in\mathcal{C}(i)} \sum_{x_j, x_{\mathcal{D}(j)}} F(x_i, x_j, x_{\mathcal{D}(j)}) \tag{7.27}$$

$$= \frac{1}{Z}\, \psi_i(x_i) \prod_{j\in\mathcal{C}(i)} \sum_{x_j, x_{\mathcal{D}(j)}} \psi_{i,j}(x_i, x_j)\psi_j(x_j) \prod_{k\in\mathcal{C}(j)} F(x_j, x_k, x_{\mathcal{D}(k)}) \tag{7.28}$$

$$= \frac{1}{Z}\, \psi_i(x_i) \prod_{j\in\mathcal{C}(i)} \sum_{x_j} \psi_{i,j}(x_i, x_j)\psi_j(x_j) \prod_{k\in\mathcal{C}(j)} \underbrace{\sum_{x_k, x_{\mathcal{D}(k)}} F(x_j, x_k, x_{\mathcal{D}(k)})}_{\mu_{k\to j}(x_j)} \tag{7.29}$$

$$= \frac{1}{Z}\, \psi_i(x_i) \prod_{j\in\mathcal{C}(i)} \underbrace{\sum_{x_j} \psi_{i,j}(x_i, x_j)\psi_j(x_j) \prod_{k\in\mathcal{C}(j)} \mu_{k\to j}(x_j)}_{\mu_{j\to i}(x_i)} \tag{7.30}$$

Which leads us to the recurrence relation for the Sum Product Algorithm (SPA):

$$\boxed{\mu_{j\to i}(x_i) = \sum_{x_j} \psi_{i,j}(x_i, x_j)\psi_j(x_j) \prod_{k\in\mathcal{C}(j)} \mu_{k\to j}(x_j)} \tag{7.31}$$

7.1.4 Sum Product Algorithm (SPA)

Sequential SPA for a rooted tree

For a rooted tree, of root (i), the Sum Product Algorithm is written as follows:

1. All the leaves send $\mu_{n\to\pi_n}(x_{\pi_n})$

$$\mu_{n\to\pi_n}(x_{\pi_n}) = \sum_{x_n} \psi_n(x_n)\psi_{n,\pi_n}(x_n, x_{\pi_n}) \tag{7.32}$$

2. Iteratively, at each step, all the nodes (k) which have received messages from all their children send $\mu_{k\to\pi_k}(x_{\pi_k})$ to their parents.

3. At the root we have

$$p(x_i) = \frac{1}{Z}\, \psi_i(x_i) \prod_{j\in\mathcal{C}(i)} \mu_{j\to i}(x_i) \tag{7.33}$$

This algorithm only enables us to compute $p(x_i)$ at the root. To be able to compute all the marginals (as well as the conditional marginals), one must not only *collect* all the messages from the leafs to the root, but then also *distribute* them back to the leafs. In fact, the algorithm can then be written independently from the choice of a root.

SPA for an undirected tree

The case of undirected trees is slightly different:

1. All the leaves send $\mu_{n\to\pi_n}(x_{\pi_n})$
2. At each step, if a node (j) hasn't send a message to one of his neighbours, say (i) (Note: (i) here is not the root) and if it has received messages from all his other neighbours $\mathcal{N}(j)\backslash i$, it send to (i) the following message

$$\mu_{j\to i}(x_i) = \sum_{x_j} \psi_j(x_j)\psi_{j,i}(x_i,x_i) \prod_{k\in\mathcal{N}(j)\backslash\{i\}} \mu_{k\to j}(x_j) \tag{7.34}$$

Parallel SPA (flooding)

1. Initialise the messages randomly
2. At each step, each node sends a new message to each of its neighbours, using the messages received at the previous step.

Marginal laws

Once all messages have been passed, we can easily calculate all the marginal laws

$$\forall\ i\in V,\ p(x_i) = \frac{1}{Z}\ \psi_i(x_i) \prod_{k\in\mathcal{N}(i)} \mu_{k\to i}(x_i) \tag{7.35}$$

$$\forall\ (i,j)\in E,\ p(x_i,x_j) = \frac{1}{Z}\ \psi_i(x_i)\ \psi_j(x_j)\ \psi_{j,i}(x_i,x_i) \prod_{k\in\mathcal{N}(i)\backslash j} \mu_{k\to i}(x_i) \prod_{k\in\mathcal{N}(j)\backslash i} \mu_{k\to j}(x_j) \tag{7.36}$$

Conditional probabilities

We can use a clever notation to calculate the conditional probabilities. Suppose that we want to compute

$$p(x_i|x_5=3,x_{10}=2) \propto p(x_i,x_5=3,x_{10}=2)$$

We can set

$$\tilde{\psi_5}(x_5) = \psi_5(x_5)\ \delta(x_5,3)$$

Generally speaking, if we observe $X_j = x_{j0}$ for $j \in J_{\text{obs}}$, we can define the modified potentials:

$$\tilde{\psi_j}(x_j) = \psi_j(x_j)\ \delta(x_j, x_{j0})$$

such that

$$p(x|X_{J_{\text{obs}}} = x_{J_{\text{obs}}0}) = \frac{1}{\tilde{Z}} \prod_{i \in V} \tilde{\psi}_i(x_i) \prod_{(i,j)\in E} \psi_{i,j}(x_i, x_j) \tag{7.37}$$

Indeed we have

$$p(x|X_{J_{\text{obs}}} = x_{J_{\text{obs}}0})\ p(X_{J_{\text{obs}}} = x_{J_{\text{obs}}0}) = p(x) \prod_{j \in J_{\text{obs}}} \delta(x_j, x_{j0}) \tag{7.38}$$

so that by dividing the equality by $p(X_{J_{\text{obs}}} = x_{J_{\text{obs}}0})$ we obtain the previous equation with $\tilde{Z} = Zp(X_{J_{\text{obs}}} = x_{J_{\text{obs}}0})$.

We then simply apply the SPA to these new potentials to compute the marginal laws $p(x_i|X_{J_{\text{obs}}} = x_{J_{\text{obs}}0})$

7.1.5 Remarks

- The SPA is also called *belief propagation* or *message passing*. On trees, it is an exact inference algorithm.
- If G is not a tree, the algorithm doesn't converge in general to the right marginal laws, but sometimes gives reasonable approximations. We then refer to "Loopy belief propagation", which is still often used in real life.
- The only property that we have used to construct the algorithm is the fact that $(\mathbb{R}, +, \times)$ is a semi-ring. It is interesting to notice that the same can therefore also be done with $(\mathbb{R}_+, \max, \times)$ and $(\mathbb{R}, \max, +)$.

  Example For $(\mathbb{R}_+, \max, \times)$ we define the Max-Product algorithm, also called "Viterbi algorithm" which enables us to solve the *decoding* problem, namely to compute the most probable configuartion of the variables, given fixed parameters, thanks to the messages

$$\mu_{j\to i}(x_i) = \max_{x_j} \left[\psi_{i,j}(x_i, x_j)\psi_j(x_j) \prod_{x_k} \mu_{k\to j}(x_j) \right] \tag{7.39}$$

  If we run the Max-Product algorithm with respect to a chosen root, the *collection* phase of the messages to the root enables us to compute the maximal probability over all configurations, and if at each calculation of a message we have also kept the argmax, we can perform a *distribution* phase, which instead of propagating the messages, will consist of recursively calculating one of the configurations which will reach the maximum.

- In practice, we may be working on such small values that the computer will return errors. For instance, for k binary variables, the joint law $p(x_1, x_2...x_n) = \frac{1}{2^n}$ can take infinitesimal values for a large k. The solution is to work with logarithms: if $p = \sum_i p_i$, by setting $a_i = \log(p_i)$ we have:

$$\log(p) = \log\left[\sum_i e^{a_i}\right]$$
$$\log(p) = a_i^* + \log\left[\sum_i e^{(a_i - a_i^*)}\right] \tag{7.40}$$

  With $a_i^* = \max_i a_i$. Using logarithms ensures a numerical stability.

7.1.6 Proof of the algorithm

We are going to prove that the SPA is correct by recurrence. In the case of two nodes, we have:

$$p(x_1, x_2) = \frac{1}{Z}\psi_1(x_i)\psi_2(x_2)\psi_{1,2}(x_1, x_2)$$

We marginalize, and we obtain

$$p(x_1) = \frac{1}{Z}\psi_1(x_1)\underbrace{\sum_{x_2}\psi_{1,2}(x_1, x_2)\psi_2(x_2)}_{\mu_{2\to1}(x_1)}$$

We can hence deduct

$$p(x_1) = \frac{1}{Z}\psi_1(x_1)\mu_{2\to1}(x_1)$$

And

$$p(x_2) = \frac{1}{Z}\psi_2(x_2)\mu_{1\to2}(x_2)$$

We assume that the result is true for trees of size $n-1$, and we consider a tree of size n. Without loss of generality, we can assume that the nodes are numbered, so that the $n-$th be a leaf, and we will call π_n its parent (which is unique, the graph being a tree). The first message to be passed is:

$$\mu_{n\to\pi_n}(x_{\pi_n}) = \sum_{x_n}\psi_n(x_n)\psi_{n,\pi_n}(x_n, x_{\pi_n}) \tag{7.41}$$

And the last message to be passed is:

$$\mu_{\pi_n\to n}(x_n) = \sum_{x_{\pi_n}}\psi_{\pi_n}(x_{\pi_n})\psi_{n,\pi_n}(x_n, x_{\pi_n})\prod_{k\in\mathcal{N}(\pi_n)\setminus\{n\}}\mu_{k\to\pi_n}(x_{\pi_n}) \tag{7.42}$$

We are going to construct a tree $\tilde{T}$ of size $n-1$, as well as a family of potentials, such that the $2(n-2)$ messages passed in T (i.e. all the messages except for the first and the last) be equal to the $2(n-2)$ messages passed in $\tilde{T}$. We define the tree and the potentials as follows:

- $\tilde{T} = (\tilde{V}, \tilde{E})$ with $\tilde{V} = \{1, \dots, n-1\}$ and $\tilde{E} = E \backslash \{n, \pi_n\}$ (i.e., it is the subtree corresponding to the $n-1$ first vertices).
- The potentials are all the same as those of T, except for the potential

$$\tilde{\psi}_{\pi_n}(x_{\pi_n}) = \psi_{\pi_n}(x_{\pi_n}) \mu_{n \to \pi_n}(x_{\pi_n}) \tag{7.43}$$

- The root is unchanged, and the topological order is also kept.

We then obtain two important properties:

1) The product of the potentials of the tree of size $n-1$ is equal to:

$$\begin{aligned}
\tilde{p}(x_1, \dots, x_{n-1}) &= \frac{1}{Z} \prod_{i \neq n, \pi_n} \psi_i(x_i) \prod_{(i,j) \in E \backslash \{n, \pi_n\}} \psi_{i,j}(x_i, x_j) \tilde{\psi}_{\pi_n}(x_{\pi_n}) \\
&= \frac{1}{Z} \prod_{i \neq n, \pi_n} \psi_i(x_i) \prod_{(i,j) \in E \backslash \{n, \pi_n\}} \psi_{i,j}(x_i, x_j) \sum_{x_n} \psi_n(x_n) \psi_{\pi_n}(x_{\pi_n}) \psi_{n, \pi_n}(x_n, x_{\pi_n}) \\
&= \sum_{x_n} \frac{1}{Z} \prod_{i=1}^{n} \psi_i(x_i) \prod_{(i,j) \in E} \psi_{i,j}(x_i, x_j) \\
&= \sum_{x_n} p(x_1, \dots, x_{n-1}, x_n)
\end{aligned}$$

which shows that these new potentials define on $(X_1, \dots, X_{n-1})$ exactly the distribution induced by p when marginalizing X_n.

2) All of the messages passed in $\tilde{T}$ correspond to the messages passed in T (except for the first and the last).

Now, with the recurrence hypothesis that the SPA is true for trees of size $n-1$, we are going to show that it is true for trees of size n. For nodes $i \neq n, \pi_n$, the result is obvious, as all messages passed are the same:

$$\forall\ i \in V \backslash \{n, \pi_n\},\ p(x_i) = \frac{1}{Z}\ \psi_i(x_i) \prod_{k \in \mathcal{N}(i)} \mu_{k \to i}(x_i) \tag{7.44}$$

For the case $i = \pi_n$, we deduct:

$$\begin{aligned} p(x_{\pi_n}) &= \frac{1}{Z}\tilde{\psi}_{\pi_n}(x_{\pi_n}) \prod_{k\in\tilde{\mathcal{N}}(\pi_n)} \mu_{k\to\pi_n}(x_{\pi_n}) \quad (\text{ product over the neighbours of } \pi_n \text{ in } \tilde{T}) \\ &= \frac{1}{Z}\tilde{\psi}_{\pi_n}(x_{\pi_n}) \prod_{k\in\mathcal{N}(\pi_n)\setminus\{n\}} \mu_{k\to\pi_n}(x_{\pi_n}) \\ &= \frac{1}{Z}\psi_{\pi_n}(x_{\pi_n})\mu_{n\to\pi_n}(x_{\pi_n}) \prod_{k\in\mathcal{N}(\pi_n)\setminus\{n\}} \mu_{k\to\pi_n}(x_{\pi_n}) \\ &= \frac{1}{Z}\psi_{\pi_n}(x_{\pi_n}) \prod_{k\in\mathcal{N}(\pi_n)} \mu_{k\to\pi_n}(x_{\pi_n}) \end{aligned}$$

For the case $i = n$, we have:

$$p(x_n, x_{\pi_n}) = \sum_{x_{V\setminus\{n,\pi_n\}}} p(x) = \psi_n(x_n)\psi_{\pi_n}(x_{\pi_n})\psi_{n,\pi_n}(x_n, x_{\pi_n}) \underbrace{\sum_{x_{V\setminus\{n,\pi_n\}}} \frac{p(x)}{\psi_n(x_n)\psi_{\pi_n}(x_{\pi_n})\psi_{n,\pi_n}(x_n, x_{\pi_n})}}_{\alpha(x_{\pi_n})}$$

Therefore:

$$p(x_n, x_{\pi_n}) = \psi_{\pi_n}(x_{\pi_n})\alpha(x_{\pi_n})\psi_n(x_n)\psi_{n,\pi_n}(x_n, x_{\pi_n}) \tag{7.45}$$

Consequently:

$$p(x_{\pi_n}) = \psi_{\pi_n}(x_{\pi_n})\alpha(x_{\pi_n}) \underbrace{\sum_{x_n} \psi_n(x_n)\psi_{n,\pi_n}(x_n, x_{\pi_n})}_{\mu_{n\to\pi_n}(x_{\pi_n})}$$

Hence:

$$\alpha(x_{\pi_n}) = \frac{p(x_{\pi_n})}{\psi_{\pi_n}(x_{\pi_n})\mu_{n\to\pi_n}(x_{\pi_n})} \tag{7.46}$$

By using (7.31), (7.32) and the previous result, we deduct that:

$$\begin{aligned} p(x_n, x_{\pi_n}) &= \psi_{\pi_n}(x_{\pi_n})\psi_n(x_n)\psi_{n,\pi_n}(x_n, x_{\pi_n}) \frac{p(x_{\pi_n})}{\psi_{\pi_n}(x_{\pi_n})\mu_{n\to\pi_n}(x_{\pi_n})} \\ &= \psi_{\pi_n}(x_{\pi_n})\psi_n(x_n)\psi_{n,\pi_n}(x_n, x_{\pi_n}) \frac{\frac{1}{Z}\psi_{\pi_n}(x_{\pi_n}) \prod_{k\in\mathcal{N}(\pi_n)} \mu_{k\to\pi_n}(x_{\pi_n})}{\psi_{\pi_n}(x_{\pi_n})\mu_{n\to\pi_n}(x_{\pi_n})} \\ &= \frac{1}{Z}\psi_{\pi_n}(x_{\pi_n})\psi_n(x_n)\psi_{n,\pi_n}(x_n, x_{\pi_n}) \prod_{k\in\mathcal{N}(\pi_n)\setminus\{n\}} \mu_{k\to\pi_n}(x_{\pi_n}) \end{aligned}$$

By summing with respect to x_{π_n}, we get the result for $p(x_n)$:

$$p(x_n) = \sum_{x_{\pi_n}} p(x_n, x_{\pi_n}) = \frac{1}{Z}\psi_n(x_n)\mu_{\pi_n \to n}(x_n)$$

Proposition:

Let $p \in \mathcal{L}(G)$, for $G = (V, E)$ a tree, then we have:

$$p(x_1, \dots, x_n) = \frac{1}{Z}\prod_{i \in V}\psi(x_i)\prod_{(i,j)\in E}\frac{p(x_i, x_j)}{p(x_i)p(x_j)} \tag{7.47}$$

Proof: we prove it by reccurence. The case $n = 1$ is trivial. Then, assuming that n is a leaf, and we can write $p(x_1, \dots, x_n) = p(x_1, \dots, x_{n-1})\, p(x_n|x_{\pi_n})$. But multiplying by $p(x_n|x_{\pi_n}) = \frac{p(x_n, x_{\pi_n})}{p(x_n)p(x_{\pi_n})}\, p(x_n)$ boils down to adding the edge potential for (n, π_n) and the node potential for the leaf n. The formula is hence verified by reccurence.

7.1.7 Junction tree

Junction tree is an algorithm designed to tackle the problem of *inference on general graphs*. The idea is to look at a general graph from far away, where it can be seen as a tree. By merging nodes, one will hopefully be able to build a tree. When this is not the case, one can also think of adding some edges to the graph (i.e., cast the present distribution into a larger set) to be able to build such a graph.

The trap is that if one collapses too many nodes, the number of possibles values will explode, and as such the complexity of the whole algorithm. The *tree width* is the smallest possible clique size. For instance, for a 2D regular grid with n points, the tree width is equal to $\sqrt{n}$.

7.2 Hidden Markov Model (HMM)

The Hidden Markov Model ("Modèle de Markov caché" in French) is one of the most used Graphical Models. We note $z_0, z_1, \dots, z_T$ the states corresponding to the latent variables, and $y_0, y_1, \dots, y_T$ the states corresponding to the observed variables. We further assume that:

1. $\{z_0, \dots, z_T\}$ is a Markov chain (hence the name of the model);
2. $z_0, \dots, z_T$ take K values;
3. z_0 follows a multinomial distribution: $p(z_0 = i) = (\pi_0)_i$ with $\sum_i (\pi_0)_i = 1$;

4. The *transition* probabilities are homogeneous: $p(z_t = i|z_{t-1} = j) = A_{ij}$, where A satisfies $\sum_i A_{ij} = 1$;

5. The emission probabilities $p(y_t|z_t)$ are homogeneous, *i.e.* $p(y_t|z_t) = f(y_t, z_t)$.

6. The joint probability distribution function can be written as:

$$p(z_0, ...z_T, y_0, ..., y_T) = p(z_0) \prod_{t=0}^{T-1} p(z_{t+1}|z_t) \prod_{t=0}^{T} p(y_t|z_t).$$

There are different tasks that we want to perform on this model:

- Filtering: $p(z_t|y_1, \ldots, y_{t-1})$
- Smoothing: $p(z_t|y_1, \ldots, y_T)$
- Decoding: $\max_{z_0,\ldots,z_T} p(z_0, \ldots, z_T|y_0, \ldots, y_T)$

All these tasks can be performed with a sum-product or max-product algorithm.

Sum-product

From now on, we note the observations $\overline{y} = \{\overline{y_1}, \ldots, \overline{y_T}\}$. The distribution on y_t simply becomes the delta function $\delta(y_t = \overline{y_t})$. To use the sum-product algorithm, we define z_T as the root, *i.e.* we send all forward messages to z_T and go back afterwards.

Forward:

$$\begin{aligned}
\mu_{y_0 \to z_0}(z_0) &= \sum_{y_0} \delta(y_0 = \overline{y_0}) p(y_0|z_0) = p(\overline{y_0}|z_0) \\
\mu_{z_0 \to z_1}(z_1) &= \sum_{z_0} p(z_1|z_0) \mu_{y_0 \to z_0}(z_0) \\
&\vdots \\
\mu_{y_{t-1} \to z_{t-1}}(z_{t-1}) &= \sum_{y_{t-1}} \delta(y_{t-1} = \overline{y_{t-1}}) p(y_{t-1}|z_{t-1}) = p(\overline{y_{t-1}}|z_{t-1}) \\
\mu_{z_{t-1} \to z_t}(z_t) &= \sum_{z_{t-1}} p(z_t|z_{t-1}) \mu_{z_{t-2} \to z_{t-1}}(z_{t-1})\ p(z_{t-1}|y_{t-1})
\end{aligned}$$

Let us define the "alpha-message" $\alpha_t(z_t)$ as:

$$\alpha_t(z_t) = \mu_{y_t \to z_t}(z_t) \mu_{z_{t-1} \to z_t}(z_t)$$

$\alpha_0(z_0)$ is initialized with the virtual message $\mu_{z_{-1} \to z_0}(z_0) = p(z_0)$

Property 7.1

$$\alpha_t(z_t) = \mu_{y_t \to z_t}(z_t)\mu_{z_{t-1} \to z_t}(z_t) = p(z_t, \overline{y_0}, \ldots, \overline{z_t})$$

This is due to the definition of the messages: the product $\mu_{y_t \to z_t}(z_t)\mu_{z_{t-1} \to z_t}(z_t)$ represents a marginal of the distribution corresponding to the sub-HMM $\{z_0, \ldots, z_t\}$.
Moreover, the following recursion formula for $\alpha_t(z_t)$ (called "alpha-recursion") holds:

$$\alpha_{t+1}(z_{t+1}) = p(\overline{y_{t+1}}|z_{t+1}) \sum_{z_t} p(z_{t+1}|z_t)\alpha(z_t)$$

Backward:

$$\mu_{z_{t+1} \to z_t}(z_t) = \sum_{z_{t+1}} p(z_{t+1}|z_t)\mu_{z_{t+2} \to z_{t+1}}(z_{t+1})p(\overline{y_{t+1}}|z_{t+1})$$

We define the "beta-message" $\beta_t(z_t)$ as:

$$\beta_t(z_t) = \mu_{z_{t+1} \to z_t}(z_t)$$

As an initialization, we take $\beta_T(z_T) = 1$. The following recursion formula for $\beta_t(z_t)$ (called "beta-recursion") holds:

$$\beta_t(z_t) = \sum_{z_{t+1}} p(z_{t+1}|z_t)p(\overline{y_{t+1}}|z_{t+1})\beta_{t+1}(z_{t+1})$$

Property 7.2

1. $p(z_t, \overline{y_0}, \ldots, \overline{y_T}) = \alpha_t(z_t)\beta_t(z_t)$
2. $p(\overline{y_0}, \ldots, \overline{y_T}) = \sum_{z_t} \alpha_t(z_t)\beta_t(z_t)$
3. $p(z_t|\overline{y_0}, \ldots, \overline{y_T}) = \frac{p(z_t, \overline{y_0}, \ldots, \overline{y_T})}{p(\overline{y_0}, \ldots, \overline{y_T})} = \frac{\alpha_t(z_t)\beta_t(z_t)}{\sum_{z_t} \alpha_t(z_t)\beta_t(z_t)}$
4. *For all* $t < T$, $p(z_t, z_{t+1}|\overline{y_0}, \ldots, \overline{y_T}) = \frac{1}{p(\overline{y_0}, \ldots, \overline{y_T})}\, \alpha_t(z_t)\, \beta_{t+1}(z_{t+1})\, p(z_{t+1}|z_t)\, p(\overline{y_{t+1}}|z_{t+1})$

Remark 7.3

1. *The alpha-recursion and beta-recursion are easy to implement, but one needs to avoid errors in the indices!*
2. *The sums need to be coded using logs in order to prevent numerical errors.*

EM algorithm

With the previous notations and assumptions, we write the complete log-likelihood $l_c(\theta)$ with θ the parameters of the model (containing (π_0, A), but also parameters for f):

$$\begin{aligned} l_c(\theta) &= \log\Big(p(z_0)\prod_{t=0}^{T-1} p(z_{t+1}|z_t)\prod_{t=0}^{T} p(\overline{y_t}|z_t)\Big) \\ &= \log(p(z_0)) + \sum_{t=0}^{T}\log p(\overline{y_t}|z_t) + \sum_{t=0}^{T}\log p(z_{t+1}|z_t) \\ &= \sum_{i=1}^{K}\delta(z_0=i)\log((\pi_0)_i) + \sum_{t=0}^{T-1}\sum_{i,j=1}^{K}\delta(z_{t+1}=i, z_t=j)\log(A_{i,j}) + \sum_{t=0}^{T}\sum_{i=1}^{K}\delta(z_t=i)\log f(\overline{y_t}, z_t) \end{aligned}$$

When applying E-M to estimate the parameters of this HMM, we use Jensen's inequality to obtain a lower bound on the log-likelihood:

$$\log p(\overline{y_0}, \ldots, \overline{y_T}) \geq \mathbb{E}_q[\log p(z_0, \ldots, z_T, \overline{y_0}, \ldots, \overline{y_T})] = \mathbb{E}_q[l_c(\theta)]$$

At the k-th expectation step, we use $q(z_0, \ldots, z_T) = \mathbb{P}(z_0, \ldots, z_T|\overline{y_0}, \ldots, \overline{y_T}; \theta_{k-1})$, and this boils down to applying the following rules:

- $\mathbb{E}[\delta(z_0 = i)|\overline{y}] = p(z_0 = i|\overline{y}; \theta^{k-1})$
- $\mathbb{E}[\delta(z_t = i)|\overline{y}] = p(z_t = i|\overline{y}; \theta^{k-1})$
- $\mathbb{E}[\delta(z_{t+1} = i, z_t = j|\overline{y}; \theta^{k-1}] = p(z_{t+1} = i, z_t = j|\overline{y}; \theta^{k-1})$

Thus, in the former expression of the complete log-likelihood, we just have to replace $\delta(z_0 = i)$ by $p(z_0 = i|\overline{y}; \theta^{k-1})$, and similarly for the other terms.
At the k-th maximization step, we maximize the new obtained expression with respect to the parameters θ in the usual manner to obtain a new estimator θ^k. The key is that everything will decouple, thus maximizing is simple and can be done in closed form.

7.3 Principal Component Analysis (PCA)

Framework: $x_1, \ldots, x_N \in \mathbb{R}^d$
Goal: put points on a closest affine subspace

Analysis view

Find $w \in \mathbb{R}^d$ such that $\mathrm{Var}(x^T w)$ is maximal, with $||w|| = 1$

With centered data, *i.e.* $\frac{1}{N}\sum_{n=1}^{N} x_n = 0$, the empirical variance is:

$$\hat{\mathrm{Var}}(x^T w) = \frac{1}{N}\sum_{n=1}^{N}(x_n^T w)^2 = \frac{1}{N} w^T (X^T X) w$$

where $X \in \mathbb{R}^{N\times d}$ is the design matrix. In this case: w is the eigenvector of $X^T X$ with largest eigenvalue. It is not obvious *a priori* that this is the direction we care about.
If more than one direction is required, one can use *deflation*:

1. Find w
2. Project x_n onto the orthogonal of $\mathrm{Vect}(w)$
3. Start again

Synthesis view

$$\min_w \sum_{n=1}^{N} d(x_n, \{w^T x = 0\})^2 \text{ with } w \in \mathbb{R}^D,\ ||w|| = 1.$$

Advantage: if one wants more than 1 dimension, replace $\{w^T x = 0\}$ by any subspace.

Probabilistic approach: Factor Analysis

Model:

- $\Lambda = (\lambda_1, \ldots, \lambda_K) \in \mathbb{R}^{d\times k}$
- $X \in \mathbb{R}^k \sim \mathcal{N}(0, I)$
- $\epsilon \sim \mathcal{N}(0, \Psi)$, $\epsilon \in \mathbb{R}^d$ independent from X with Ψ diagonal.
- $Y \in \mathbb{R}^d$: $Y = \Lambda X + \mu + \epsilon$

We have $Y|X \sim \mathcal{N}(\Lambda X + \mu, \Psi)$.
Problem: get $X|Y$.

(X, Y) is a Gaussian vector on $\mathbb{R}^{d+k}$ which satisfies:

- $\mathbb{E}[X] = 0 = \mu_X$
- $\mathbb{E}[Y] = \mathbb{E}[\Lambda X + \mu + \epsilon]\mu = \mu_Y$

- $\Sigma_{XX} = I$
- $\Sigma_{XY} = \text{Cov}(X, \Lambda X + \mu + \epsilon) = \text{Cov}(X, \Lambda X) = \Lambda^T$
- $\Sigma_{YY} = \text{Var}(\Lambda X + \epsilon, \Lambda X + \epsilon) = \text{Var}(\Lambda X, \Lambda X) + \text{Var}(\epsilon, \epsilon) = \Lambda\Lambda^T + \Psi$

Thanks to the results we know on exponential families, we know how to compute $X|Y$:

$$\begin{aligned} \mathbb{E}[X|Y = y] &= \mu_X + \Sigma_{XY}\Sigma_{YY}^{-1}(y - \mu_Y) \\ \text{Cov}[X|Y = y] &= \Sigma_{XX} - \Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{YX} \end{aligned}$$

In our case, we therefore have:

$$\begin{aligned} \mathbb{E}[X|Y = y] &= \Lambda^T(\Lambda\Lambda^T + \Psi)^{-1}(y - \mu) \\ \text{Cov}[X|Y = y] &= I - \Lambda^T(\Lambda\Lambda^T + \Psi)^{-1}\Lambda^T \end{aligned}$$

To apply EM, one needs to write down the complete log-likelihood.

$$\log p(X, Y)\alpha - \frac{1}{2}X^T X - \frac{1}{2}(Y - \Lambda X - \mu)^T\Psi^{-1}(Y - \Lambda X - \mu) - \frac{1}{2}\log\det\Psi$$

Trap: $\mathbb{E}[XX^T|Y] \neq \text{Cov}(X|Y)$
Rather, $\mathbb{E}[XX^T|Y] = \text{Cov}(X|Y) + \mathbb{E}[X|Y]\mathbb{E}[X|Y]^T$

Remark 7.4

- $\text{Cov}(X) = \Lambda\Lambda^T + \Psi$*: our parameters are not identifiable,* $\Lambda \leftarrow \Lambda R$ *with* R *a rotation gives the same results (in other words, a subspace has different orthonormal bases).*
- *Why do we care ?*
  1. *A probabilistic interpretation allows to model in a finer way the problem.*
  2. *It is very flexible and therefore allows to combine multiple models.*

Bibliography

Learning in graphical models & MC methods **Fall 2015**

Cours 8 — November 18

Enseignant: Guillaume Obozinski *Scribe: Khalife Sammy, Maryan Morel*

8.1 HMM (end)

As a reminder, the message propagation algorithm for Hidden Markov Models requires 2 recursions to compute $\alpha_t(z_t) := p(z_t, y_1, \dots, y_t)$ and $\beta_t(z_t) := p(y_{t+1}, \dots, y_T | z_t)$:

$$\alpha_{t+1}(z_{t+1}) = p\left(y_{t+1} \mid z_{t+1}\right) \sum_{z_t} p\left(z_{t+1} \mid z_t\right) \alpha_t\left(z_t\right) \tag{8.1}$$

$$\beta_t(z_t) = \sum_{z_{t+1}} p\left(y_{t+1} \mid z_{t+1}\right) p\left(z_{t+1} \mid z_t\right) \beta_{t+1}\left(z_{t+1}\right) \tag{8.2}$$

Addressing practical implementation issues

Since α_t and β_t are respectively joint probabilities of $t+1$ and $T-t$ variables they tend to become exponentially small respectively for t large and t small. A naive implementation of the forward-backward algorithm therefore typically leads to rounding errors. It is therefore necessary to work on a logarithmic scale.

So when considering operations on quantities say $a_1, \dots, a_n$ whose logarithms are $\ell_i = \log(a_i)$, the log of the product is easily computed as $\ell_\Pi = \log \prod_i a_i = \sum_i \ell_i$ and the log of the sum can be computed with the smallest amount of numerical errors by factoring the largest element. Precisely if $i_* = \arg\max_i a_i$ and $\ell_* = \log a_{i_*}$ then

$$\ell_\Sigma = \log \sum_i a_i = \log \sum_i \exp(\ell_i) = \log \Big[\exp(\ell_*) \sum_i \exp(\ell_i - \ell_*) \Big] = \ell_* + \log \Big(1 + \sum_{i \neq i_*} \exp(\ell_i - \ell_*) \Big)$$

provides a stable way of computing the logarithm of the sum.

For hidden Markov models, remember that the max-product (aka Viterbi) algorithm allows to compute the most probable sequence for hidden states.

8.2 Multiclass classification

We return briefly to classification to mention two simple yet classical and useful models for multi-class classification: the naive Bayes model and the multiclass logistic regression. We consider classification problems where the input data is in $\mathcal{X} = \mathbb{R}^p$ and the output variable is a binary indicator in $\mathcal{Y} = \{y \in \{0,1\}^K \mid y_1 + \ldots + y_K = 1\}$.

8.2.1 Naive Bayes classifier

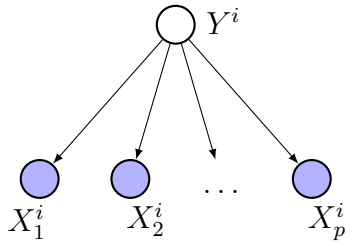
The naive Bayes classifier is relevant when modeling the joint distribution of $p(x|y)$ is too complicated. We will present it the special case where the input data is a vector of binary random variable. $X^i : \Omega \mapsto \{0,1\}^p$

A practical example of classification problem in this setting is the problem of classification of documents based on a bag of word representation. In the bag-of-word approach, a document is represented as a long binary vector which indicates for each word of a reference dictionary whether that word is present in the document considered or not. So the document i would be represented by a vector $x^i \in \{0,1\}^p$, with $x^i_j = 1$ iff word j of the dictionary is present in the ith document.

As we saw in the second lecture, it is possible to approach the problem using directly a *conditional model* of $p(y \mid x)$ or using a *generative model* of the joint distribution modeling separately $p(y)$ and $p(x|y)$ and computing $p(y|x)$ using Bayes rule. The naive Bayes model is an instance of a generative model. By contrast the multi class logistic regression of the following section is an example of a conditional model.

Y^i is naturally modeled as a multinomial distribution with $p(y^i) = \prod_{k=1}^K \pi_k^{y^i_k}$. However $p(x^i|y^i) = p(x^i_1, \ldots, x^i_p|y^i)$ has a priori $2^p - 1$ parameters. The key assumption made in the naive Bayes model is that $X^i_1, \ldots, X^i_p$ are all independent conditionally on Y^i. This assumption is not realistic and simplistic, hence the term "naive". This assumption is clearly not satisfied in practice for documents where one would expect that there would be correlations between words that are not just explained by a document category. The corresponding modeling strategy is nonetheless working well in practice.

These conditional independence assumptions correspond to the following graphical model:



The distribution of Y^i is a multinomial distribution which we parameterize with $(\pi_1, ..., \pi_K)$, and we write $\mu_{jk} = P(X^{(i)}_j = 1|Y^{(i)}_k = 1)$ We then have

$$p(X^i = x^i, Y^i = y^i) = p(x_i, y_i) = p(x^i|y^i)p(y^i) = \prod_{j=1}^{p} p(x^i_j|y^i)p(y^i)$$

which leads to

$$p(x^i, y^i) = \Big[\prod_{j=1}^{p}\prod_{k=1}^{K} {\mu_{jk}}^{x^i_j y^i_k} (1-\mu_{jk})^{(1-x^i_j)y^i_k}\Big] \prod_{k=1}^{K} \pi_k^{y^i_k}$$

and

$$\log p(x^i, y^i) = \sum_{k=1}^{K} \Big(\sum_{j=1}^{p} \big(x_j^i y_k^i \log \mu_{jk} + (1 - x_j^i) y_k^i \log(1 - \mu_{jk})\big) + y_k^i \log(\pi_k) \Big)$$

We can then use Bayes' rule (hence the "Bayes" in "Naive Bayes"), which leads to

$$\log p(y^i | x^i) = \eta(x^i)^\top y^i - A(\eta(x^i))$$

with $\eta(x) = (\eta_1(x), \ldots, \eta_K(x)) \in \mathbb{R}^K$ and

$$\eta_k(x) = w_k^\top x + b_k, \quad w_k \in \mathbb{R}^p, \quad [w_k]_j = \log \frac{\mu_{jk}}{1 - \mu_{jk}}, \quad b_k = \log \pi_k.$$

Note that, in spite of the name the naive Bayes classifier is not a Bayesian approach to classification.

Multiclass logistic regression

In the light of the course on exponential families, the logistic regression model can be seen as resulting from a linear parameterization as a function of x of the natural parameter $\eta(x)$ of the Bernoulli distribution corresponding to the conditional distribution of Y given $X = x$. Indeed for binary classification, we have that $Y|X = x \sim \text{Ber}(\mu(x))$ and in the logistic regression model we set $\mu(x) = \exp(\eta(x) - A(\eta(x))) = (1 + \exp(-\eta(x))^{-1}$ and $\eta(x) = w^\top x + b$.

It is then natural to consider the generalization to a multiclass classification setting. In that case, $Y|X = x$ is multinomial distribution with natural parameters $(\eta_1(x), \ldots, \eta_K(x))$. To again parameterize them linearly as a function of x, we need to introduce parameters $w_k \in \mathbb{R}^p$ and $b_k \in \mathbb{R}$, for all $1 \leq k \leq K$ and set $\eta_k(x) = w_k^\top x + b_k$. We then have

$$\mathbb{P}(Y_k = 1 | X = x) = \exp(\eta_k(x) - A(\eta(x))) = \frac{e^{\eta_k(x)}}{\sum_{k'=1}^{K} e^{\eta_{k'}(x)}} = \frac{e^{w_k^\top x + b_k}}{\sum_{k'=1}^{K} e^{w_{k'}^\top x + b_{k'}}},$$

and thus

$$\log \mathbb{P}(Y_k = y | X = x) = \sum_{k=1}^{K} y_k (w_k^\top x + b_k) - \log \Big[\sum_{k'=1}^{K} e^{w_{k'}^\top x + b_{k'}} \Big].$$

Like for binary logistic regression, the maximum likelihood principle can used to learn $(w_k, b_k)_{1 \leq k \leq K}$ using numerical optimization methods such as the IRLS algorithm.

Note that the form of the parameterization obtained is the same as for the Naive Bayes model; however, the Naive Bayes model is learnt as a generative model, while the logistic regression is learnt as conditional model.

We have not talked about the multi class generalization of Fisher's linear discriminant. It exists as well as the multi class counterpart of the model seen for binary regression. It relies like in the binary case on the assumption that $p(x|y)$ is Gaussian. This is good exercise to derive it.

8.3 Learning on graphical models

8.3.1 ML principle for general Graphical Models

Directed graphical model

Proposition : Let G be a directed graph with p nodes. Assume that $(X^1, ...X^n)$ are i.i.d., with p features : i.e $\forall i \in \{1, .., n\}, X_i \in \mathbb{R}^p$, and that are fully observed, i.e., there is no latent or hidden variable among them. Then the ML principle decouples in p optimisation problems.

Proof : Let us assume we have a decoupled model $\mathcal{P}_\Theta$, i.e. :

$$\mathcal{P}_\Theta := \left\{ p_\theta(x) = \prod_j p(x_j | x_{\pi_j}, \theta_j) \mid \theta = (\theta_1, ..., \theta_p) \in \Theta = \Theta_1 \times ... \times \Theta_p \right\}$$

$$L(\theta) = \prod_{i=1}^{n} p(x^i|\theta) = \prod_{i=1}^{n} \prod_{j=1}^{p} p(x_j^i | x_{\pi_j}^i, \theta_j)$$

$$\ell(\theta) = \sum_{j=1}^{p} \sum_{i=1}^{n} \log p(x_j^i | x_{\pi j}^i, \theta_j).$$

Then the ML principle reduces to solving p optimization problems of the form

$$\max_{\theta_j} \ell_j(\theta_j) \quad \text{s.t} \quad \theta_j \in \Theta_j, \qquad \text{with} \qquad \ell_j(\theta_j) := \sum_{i=1}^{n} \log p(x_j^i | x_{\pi_j}^i, \theta_j).$$

Undirected graphical model

→ The ML problem is convex with respect to canonical parameters if: the data is fully observed (no latent or hidden variable), and the parameters are decoupled.
→ In general, if the data is not fully observed, the EM scheme or similar scheme is used.
If the parameters are coupled, the problem remains convex in some cases (e.g linear coupling), but not in general.
→ If the model is a tree, one can reformulate the model as a directed tree to get back to the directed case.
→ In general, to compute the gradient of the log partition function and thus to compute the gradient of the log-likelihood, it is necessary to perform *probabilistic inference* on the model (i.e. to compute $\nabla A(\theta) = \mu(\theta) = \mathbb{E}_\theta[\phi(X)]$). If the model is a tree, this can be done with the sum-product algorithm and if the model is a close to a tree, the junction tree theory can be leveraged to perform *probabilistic inference*; however in general *probabilistic inference* is NP-hard and so one needs to use approximate probabilistic inference techniques.

8.4 Approximate inference with Monte Carlo methods

8.4.1 Sampling methods

We often need to compute the expectation of a function f under some distribution p that cannot be computed. Let X be a random variable following the distribution p, we want to compute $\mu = \mathbb{E}[f(X)]$.

Example 8.4.1 *For $X = (X_1, ..., X_p)$ the vector of variables corresponding to a graphical model,*

$$f(X) = \delta(X_A = x_A)$$
$$\mathbb{E}[f(X)] = \mathbb{P}(X_A = x_A)$$

If we know how to sample from p, we can use the following method :

Algorithm 1 Monte Carlo Estimation

1: Draw $X^{(1)}, ..., X^{(n)} \overset{i.i.d.}{\sim} p$
2: $\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} f(X^{(i)})$

This method relies on the two following propositions :

Proposition 8.1 (Law of Large Numbers (LLN))

$$\hat{\mu} \xrightarrow{a.s.} \mu \quad \text{if} \quad ||\mu|| < \infty$$

Proposition 8.2 (Central Limit Theorem (CLT)) *For X a scalar random variable, if $\mathbb{V}ar(f(X)) = \sigma^2 < \infty$, then*

$$\sqrt{n}(\hat{\mu} - \mu) \xrightarrow{\mathcal{D}} \mathcal{N}(0, \sigma^2)$$

thus $\mathbb{E}(||\hat{\mu} - \mu||_2^2) = \frac{\sigma^2}{n}$

How to sample from a specific distribution ?

1. Uniform distribution on [0, 1] : use `rand`
2. Bernoulli distribution of parameter p : $X = \mathbf{1}_{\{U<p\}}$ with $U \sim \mathcal{U}([0, 1])$
3. Using inverse transform sampling :

$$\forall x \in \mathbb{R} \qquad F(x) = \int_{-\infty}^{x} p(t)dt = \mathbb{P}(X \in [-\infty, x])$$

$$X = F^{-1}(U) \text{ avec } U \sim \mathcal{U}([0, 1])$$

Proof $\mathbb{P}(X \leq y) = \mathbb{P}(F^{-1}(U) \leq y) = \mathbb{P}(U \leq F(y)) = F(y)$ ■

Example 8.4.2 *Exponential distribution (one of the rare cases admitting an explicit inverse CDF[1])*

$$p(x) = \lambda e^{-\lambda x} \mathbf{1}_{\mathbb{R}_+}(x)$$

$$X = -\frac{1}{\lambda} \ln(U)$$

8.4.2 Ancestral sampling

Consider the problem of sampling from a directed graphical model, whose distribution takes the form

$$p(x_1, \dots, x_d) = \prod_{i=1}^{d} p(x_i \mid x_{\pi_i}).$$

We assume, without loss of generality, that the variables are indexed in a topological order.

Consider the following algorithm

Algorithm 2 Ancestral sampling

for $i = 1$ to d **do**
  Draw z_i from $\mathbb{P}(X_i = .|X_{\pi_i} = z_{\pi_i})$
end for
  return $(z_1, \dots, z_d)$

Proposition 8.3 *The random variable* $(Z_1, \dots, Z_d)$ *returned by the* ancestral sampling algorithm *follows exactly the distribution* $p(x_1, \dots, x_d) = \mathbb{P}(X_1 = x_1, \dots, X_d = x_d)$.

Proof We prove the result by induction. It is clearly obvious for a graph with a single node. For two nodes corresponding the pair of variable (X_1, X_2), then either X_1 and X_2 are independent and we are back to the single node case. Or $\pi_2 = \{1\}$ and then if z_1 is drawn from p_{X_1} and, given the value z_1 obtained, z_2 is drawn from the conditional distribution $p_{X_2|X_1}(\,\cdot\,|z_1)$, then the pair (z_1, z_2) follows the joint distribution p_{X_1,X_2}. Now, assuming the result is true for $n-1$ nodes we prove that it is also true for n nodes. First note that after sampling $z_1, \dots, z_{n-1}$ according to the algorithm $z_1, \dots, z_{n-1}$ follows the distribution given by $\prod_{i=1}^{n-1} p(x_i \mid x_{\pi_i})$ which is exactly the marginal distribution of $(X_1, \dots, X_{n-1})$. But then z_n is drawn according to the distribution $\mathbb{P}(X_n = \cdot \mid X_{\pi_n} = z_{\pi_n})$ which by the Markov property is equal to $\mathbb{P}\big(X_n = \cdot \mid (X_1, \dots, X_{n-1}) = (z_1, \dots, z_{n-1})\big)$. Setting $\tilde{X}_2 = X_n$ and $\tilde{X}_1 = (X_1, \dots, X_{n-1})$, we see that we are essentially back to the two nodes case, and that clearly $(z_1, \dots, z_{n-1})$ is indeed drawn from the distribution of the random variable $(X_1, \dots, X_n)$. By induction the result is proven. ■

[1] Cumulative Distribution Function

8.4.3 Rejection sampling

Assume that $p(x)$ is the density of x with respect to some measure μ (typically the Lebesgue measure for a continuous random variable and the counting measure for a discrete variable) is known up to a constant

$$p(x) = \frac{\tilde{p}(x)}{Z_p}$$

Assume that we can construct and compute q_k such that

$$\tilde{p}(x) < kq_k(x)$$

with q_k a probability distribution. Assume we can sample from q. We define the rejection sampling algorithm as :

Algorithm 3 Rejection Sampling Algorithm

1: Draw X from q
2: Accept X with probability $\frac{\tilde{p}(x)}{kq_k(x)} \in [0,1]$, otherwise, reject the sample

Proposition 8.4 *Accepted draws from rejection sampling follow exactly the distribution p.*

Proof We write the proof for the case of a discrete random variable X.

$$\begin{aligned}
\mathbb{P}(X = x, X \text{ is accepted}) &= \mathbb{P}(X = x, X \text{ is accepted}) \\
&= \mathbb{P}(X \text{ is accepted} | X = x)\mathbb{P}(X = x) \\
&= \frac{\tilde{p}(x)}{kq(x)} q(x) \\
&= \frac{\tilde{p}(x)}{k}
\end{aligned}$$

and

$$\mathbb{P}(X \text{ is accepted}) = \sum_x \frac{\tilde{p}(x)}{k} = \frac{Z_p}{k}$$

so that

$$\mathbb{P}(X = x | X \text{ is accepted}) = \frac{\tilde{p}(x)}{k} \frac{k}{Z_p} = p(x).$$

To write the general version of this proof formally for any random variable (continuous or not) that has a density with respect to a measure μ, we would need to define Y to be the Bernoulli random variable such that $\{Y = 1\} = \{X \text{ is accepted}\}$, and to consider $p_{X,Y}$ the joint density of (X, Y) with respect to the product measure $\mu \times \nu$, where ν is the counting measure on $\{0, 1\}$. The computations of $p_{X,Y}(x, y)$ and $p_{X|Y}(x, 1)$ then lead to the exact same calculations as above, but with less transparent notations. ■

Remark 8.4.1 *In practice, finding q and k such that acceptance has a reasonably large probability is hard, because it requires to find a fairly tight bound on $p(x)$ over the entire space.*

8.4.4 Importance Sampling

Assume $X \sim p$. We aim at compuing the expectation of a function f :

$$\begin{aligned}
\mathbb{E}_p(f(X)) &= \int f(x)p(x)\mathrm{d}x \\
&= \int \frac{f(x)p(x)}{q(x)}q(x)\mathrm{d}x \\
&= \mathbb{E}_q\left(f(Y)\frac{p(Y)}{q(Y)}\right) \qquad \text{with } Y \sim q \\
&= \mathbb{E}_q(g(Y)) \\
&\approx \frac{1}{n}\sum_{j=1}^{n} g(Y_j) \qquad \text{with } Y_j \overset{iid}{\sim} q \\
&= \frac{1}{n}\sum_{j=1}^{n} f(Y_j)\frac{p(Y_j)}{q(Y_j)}
\end{aligned}$$

$w(Y_i) = \frac{p(Y_j)}{q(Y_j)}$ are called *importance weights*. Remember that

$$\mu = \mathbb{E}_p(f(X)) \approx \hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} f(X_i)$$

Thus we get

$$\begin{aligned}
\mathbb{E}(\hat{\mu}) &= \frac{1}{n}\sum \int f(x)\frac{p(x)}{q(x)}q(x)dx = \int f(x)p(x)dx \\
Var(\hat{\mu}) &= \frac{1}{n}Var_{q(x)}\left(\frac{f(x)p(x)}{q(x)}\right).
\end{aligned}$$

Lemme 8.5 *If $\forall x,\ |f(x)| \leq M$,*

$$Var(\hat{\mu}) \leq \frac{M^2}{n}\int \frac{p(x)^2}{q(x)}dx.$$

Proof

$$\begin{aligned}
Var(\hat{\mu}) = &\ \tfrac{1}{n}Var_{q(x)}\left(\tfrac{f(x)p(x)}{q(x)}\right) \\
\leq &\ \tfrac{1}{n}\int \tfrac{f(x)^2p(x)^2}{q(x)^2}q(x)dx \\
\leq &\ \tfrac{M^2}{n}\int \tfrac{p(x)^2}{q(x)}dx.
\end{aligned}$$

■

Remark 8.4.2

$$\begin{aligned}\int \frac{p(x)^2}{q(x)}dx &= \int \frac{p^2(x) - 2p(x)q(x) + q^2(x)}{q(x)}dx + \int \frac{2p(x)q(x) - q^2(x)}{q(x)}dx \\ &= \underbrace{\int \frac{(p(x) - q(x))^2}{q(x)}dx}_{\chi^2 \text{ divergence between } p \text{ and } q.} + 1\end{aligned}$$

Hence, importance sampling will give good results if q has mass where p has. Indeed, if for some y, $q(y) \ll p(y)$, importance weights $Var(\hat{\mu})$ may be very large.

Extension of Importance Sampling Assume we only know p and q up to a constant : $p(x) = \frac{\tilde{p}(x)}{Z_p}$ and $q(x) = \frac{\tilde{q}(x)}{Z_p}$, and only $\tilde{p}(x)$ and $\tilde{q}(x)$ are known.

$$\begin{aligned}\mathbb{E}\left(f(Y)\frac{\tilde{p}(Y)}{\tilde{q}(Y)}\right) &= \mathbb{E}\left(f(Y)\frac{p(Y)}{q(Y)}\frac{Z_p}{Z_q}\right) = \mu\frac{Z_p}{Z_q} \\ \hat{\tilde{\mu}} &= \frac{1}{n}\sum_{i=1}^{n} f(Y_i)\frac{\tilde{p}(Y_i)}{\tilde{q}(Y_i)} \xrightarrow{a.s.} \mu\frac{Z_p}{Z_q}\end{aligned}$$

Take f to be a constant, we get

$$\begin{aligned}\hat{Z}_{p/q} &= \frac{1}{n}\sum_{i=1}^{n}\frac{p(Y_i)}{q(Y_i)} \xrightarrow{a.s.} \frac{Z_p}{Z_q} \\ \hat{\mu} &= \frac{\hat{\tilde{\mu}}}{\hat{Z}_{p/q}} \xrightarrow{a.s.} \mu\end{aligned}$$

Remark 8.4.3 *Even if $Z_p = Z_q = 1$, renormalizing by $\hat{Z}_{p/q}$ often improves the estimation.*

8.5 Markov Chain Monte Carlo (MCMC)

Unfortunately, the previous techniques are often insufficient, especially for complex multivariate distributions, so that it is not possible to draw exactly from the distribution of interest or to obtain a reasonably good estimates based on importance sampling. The idea

of MCMC is that in many cases, even though it is not possible to sample directly from a distribution of interest, it is possible to construct a Markov Chain of samples $X_0, X_1, \ldots$ whose distribution $q_t(x) = p(X_t = x)$ converges to a target distribution $p(x)$.

The idea is then that if T_0 is sufficiently large, we can consider that for all $t \geq T_0$, X_t follows approximately the distribution p and that

$$\frac{1}{T - T_0} \sum_{t=T_0+1}^{T} f(X_t) \approx \frac{1}{T - T_0} \sum_{t=T_0+1}^{T} f(X_t) \approx \mathbb{E}_p[f(X)]$$

Note that there is a double approximation: one due to the use of the law of large numbers and the second due to the approximation $q_t \approx p$ for t sufficiently large. Note also that the draws of X_t, X_{t+1} etc are not independent (but this is not necessary here to have a law of large numbers). The times before T_0 is often called the *burn in* period. The most classical procedure to obtain such a Markov Chain in the context of graphical models is called *Gibbs sampling*. We will see it in more details later.

N.B.: In this whole section we write X_t instead of $X^{(t)}$ which would match better with other sections. Indeed, here X_t should be thought typically as the whole vector of variables corresponding to a graphical model $X_t = (X_{t,j})_{1 \leq j \leq d}$. We write t as an index just to simplify notations.

In the rest of this section we will assume that we work with random variables taking values in a set $\mathcal{X}$ with $|\mathcal{X}| = K < \infty$. However K is typically very large since it corresponds to all the configurations that the set of variables of a graphical model can take.

8.5.1 Review of Markov chains

Consider an order 1 homogenous Markov chain, i.e. such that for all t,

$$\mathbb{P}(X_t = y | X_{t-1} = x) = \mathbb{P}(X_{t-1} = y | X_{t-2} = x)$$

Definition 8.6 (Time Homogenous Markov chain)

$$\begin{aligned} \forall t \geq 0,\ \forall (x, y) \in \mathcal{X}, \qquad & p(X_{t+1} = y \mid X_t = x, X_{t-1}, \ldots, X_0) \\ & = p(X_{t+1} = y \mid X_t = x) \\ & = p(X_1 = y \mid X_0 = x) \\ & = S(x, y) \end{aligned}$$

Definition 8.7 (Transition matrix) *Let $k = card(\mathcal{X}) < \infty$. We define the matrix $S \in \mathbb{R}^{k \times k}$ such that $\forall x, y \in \mathcal{X}, S(x, y) = \mathbb{P}(X_t = y | X_{t-1} = x)$. S is called* transition matrix *of the Markov chain $(X_k)_k$.*

Properties 8.5.1 *If $k = card(\mathcal{X}) < \infty$, then:*

- $\forall x, y \in \mathcal{X}, \quad S(x, y) \geq 0$

- $S\mathbf{1} = \mathbf{1}$ *(i.e. row sums are equal to* 1*)*

S is a stochastic matrix

Definition 8.8 (Stationary Distribution) *The distribution π on $\mathcal{X}$ is stationary if*

$$S^T\pi = \pi \quad \textit{with} \quad \pi = \big(\pi(x)\big)_{x\in\mathcal{X}} \qquad \textit{or equivalently} \quad \forall y \in \mathcal{X},\ \pi(y) = \sum_{x\in\mathcal{X}} \pi(x)S(x,y).$$

If $\mathbb{P}(X_n = x) = \pi(x)$ with π a stationary distribution of S, then we have

$$\mathbb{P}(X_{n+1} = y) = \sum_x \mathbb{P}(X_{n+1} = y|X_n = x)\mathbb{P}(X_n = x) = \sum_x S(x,y)\pi(x) = \pi(y)$$

Theorem 8.9 (Perron-Frobenius) *Every stochastic matrix S has* at least *one stationary distribution.*

Let $S^m(x,y) := \mathbb{P}(X_{t+m} = y|X_t = x)$.

Definition 8.10 (Irreducible Markov Chain) *A Markov chain is* irreducible *if*

$$\forall x, y \in \mathcal{X}, \exists m \in \mathbb{N},\ S^m(x,y) > 0.$$

Definition 8.11 (Period of a state) *The greatest common divider of the elements in the set $\{m > 0 \mid S^m(x,x) > 0\}$ is called the period of a state. When the period is equal to* 1 *the state is said to be aperiodic.*

Definition 8.12 (Aperiodic Markov Chain) *If all the states of a Markov chain are aperiodic the chain is said to be aperiodic.*

Definition 8.13 (Regular Markov Chain) *A Markov chain is regular if*

$$\forall x, y \in \mathcal{X}, S(x,y) > 0.$$

Remark 8.5.1 *A regular Markov chain is clearly irreducible aperiodic. The converse is not true.*

Proposition 8.14 *If a Markov chain on a finite state space is irreducible and aperiodic, then its transition matrix has a unique stationary distribution π and for any initial distribution q_0 on X_0, if $q_t(\cdot) = \mathbb{P}(X_t = \cdot)$, then $q_t \underset{t\to+\infty}{\longrightarrow} \pi$ Let q_n be the distribution of X_n, then for all distribution q_0 we get*

$$q_n \to \pi$$

Remark 8.5.2 *If the state space is not finite, an additional assumption is needed on the Markov chain: that it is recurrent positive. We do not define this notion in this course.*

Goal We want to construct a irreducible aperiodic transition S whose stationary distribution is

$$\pi(x) = \frac{1}{Z} \prod_{c \in \mathcal{C}} \psi_c(x_c)$$

.

Definition 8.15 (Detailed Balance) *A Markov chain is* reversible *if for the transition matrix S,*

$$\exists \pi, \forall x, y \in \mathcal{X}, \pi(x)S(x,y) = \pi(y)S(y,x)$$

This equation is called the detailed balance equation*. It can be reformulated*

$$\mathbb{P}(X_{t+1} = y, X_t = x) = \mathbb{P}(X_{t+1} = x, X_t = y)$$

Proposition 8.16 *If π satisfies detailed balance, then π is a stationary distribution.*

Proof $\sum_x S(x,y)p(x) = \sum_x p(y)S(y,x) = p(y)\sum_x S(y,x) = p(y).$ ■

8.5.2 Metropolis-Hastings Algorithm

Proposal transition $T(x,z) = \mathbb{P}(Z = z | X = x)$

Acceptance probability $\alpha(x,t) = \mathbb{P}(\text{Accept z } | X = x, Z = z)$

α is not a transition matrix.

Algorithm 4 Metropolis Hastings

1: Initialize x_0 from $X_0 \sim q$
2: **for** $t = 1, \ldots, T$ **do**
3: Draw z_t from $\mathbb{P}(Z = \cdot | X_{t-1} = x_{t-1}) = T(x_{t-1}, \cdot)$
4: With probability $\alpha(z_t, x_{t-1})$, set $x_t = z_t$, otherwise, set $x_t = x_{t-1}$
5: **end for**

Proposition 8.17 *With that choice of $\alpha(x,z)$, if $\mathcal{X}$ is finite, if $T(\cdot,\cdot)$ is the transition matrix of an irreducible Markov chain such that, for any (x,z), $\big(T(x,z) > 0 \Rightarrow T(z,x) > 0\big)$, and if $\pi(x) > 0$ for all $x \in \mathcal{X}$, then the Metropolis-Hastings algorithm defines a Markov chain that converges to π.*

Proof $\mathbb{P}(X_t = x_t | X_{t-1} = x_{t-1}) = S(x_{t-1}, x_t)$

$$\begin{aligned} \forall z \neq x, S(x,z) &= T(x,z)\alpha(x,z) \\ S(x,x) &= T(x,x) + \sum_{z \neq x} T(x,z)(1-\alpha(x,z)) \end{aligned}$$

Let π be given; we want to choose S such that we have *detailed balance*: The fact that we have detailed balance for a transition from x to x is obvious because When $z \neq x$, we have

$$\begin{aligned} \pi(x)S(x,z) &= \pi(z)S(z,x) \\ \pi(x)T(x,z)\alpha(x,z) &= \pi(z)T(z,x)\alpha(z,x) \end{aligned}$$

Then

$$\frac{\alpha(x,z)}{\alpha(z,x)} = \frac{\pi(z)T(z,x)}{\pi(x)T(x,z)} \ (*)$$

If

$$\alpha(x,z) = \min\left(1, \frac{\pi(z)T(z,x)}{\pi(x)T(x,z)}\right)$$

then

$$\begin{cases} \alpha(x,z) \in [0,1] \\ (*) \text{ is satisfied} \end{cases} \implies \text{detailed balance}$$

■

Approximate inference, Sampling & Variational inference **Fall 2015**

Cours 9 — November 25

Enseignant: Guillaume Obozinski *Scribe: Basile Clément, Nathan de Lara*

9.1 Approximate inference with MCMC

9.1.1 Gibbs sampling

Let us consider an undirected graph and its associated distribution p from which we want to sample (in order to do inference for example). It is assumed that:

- It is difficult to sample directly from p.
- It is easy to sample from $\mathbb{P}_p(X_i = .|X_{-i} = x_{-i})$

The idea consists in using the Markov property so that:

$$\mathbb{P}_p(X_i = .|X_{-i} = x_{-i}) = \mathbb{P}_p(X_i = .|X_{N_i} = x_{N_i}) \tag{9.1}$$

Where N_i is the Markov blanket of the node i. Based on this, Gibbs sampling is a process that converges in distribution to p.

The most classical version of the Gibbs sampling algorithm is *cyclic scan Gibbs sampling*.

Algorithm 1 Cyclic scan Gibbs sampling

initialize $t = 0$ and $\mathbf{x}^0$
while $t < T$ **do**
  for $i = 1..d$ **do**
    $x_i^t \sim \mathbb{P}_p(X_i = .|X_{-i} = x_{-i}^{t-1})$
    $x_j^t = x_j^{t-1} \ \forall j \neq i$
    $t = t + 1$
  end for
end while
return $\mathbf{x}^T$

Another version of the algorithm called *random scan Gibbs sampling* consists in picking the index i at random at each step t.

Algorithm 2 Random scan Gibbs sampling

initialize $t = 0$ and $\mathbf{x}^0$
while $t < T$ **do**
  Draw i uniformly at random in $\{1, \dots, d\}$
  $x_i^t \sim \mathbb{P}_p(X_i = .|X_{-i} = x_{-i}^{t-1})$
  $x_j^t = x_j^{t-1} \ \forall j \neq i$
  $t = t + 1$
end while
return $\mathbf{x}^T$

9.1.2 Application to the Ising Model

Let us now consider the Ising model on a graph $G = (V, E)$. X is a random variable which takes values in $\{0, 1\}^d$ with a probability distribution that depends on some parameter η:

$$p_\eta(x) = \exp\left(\sum_i \eta_i x_i + \sum_{\{i,j\}\in E} \eta_{ij} x_i x_j - A(\eta)\right) \tag{9.2}$$

To apply the Gibbs sampling algorithm, we need to compute $\mathbb{P}(X_i = x_i|X_{-i} = x_{-i})$

We have

$$\mathbb{P}(X_i = x_i, X_{-i} = x_{-i}) = \frac{1}{Z(\eta)} \exp\left(\eta_i x_i + \sum_{j\in N_i} \eta_{ij} x_i x_j + \sum_{j\neq i} \eta_j x_j + \sum_{\{j,j'\}\in E,\, j,j'\neq i} \eta_{jj'} x_j x_{j'}\right)$$

and thus

$$\mathbb{P}(X_{-i} = x_{-i}) = \frac{1}{Z(\eta)} \sum_{z\in\{0,1\}} \exp\left(\eta_i z + \sum_{j\in N_i} \eta_{ij} z x_j + \sum_{j\neq i} \eta_j x_j + \sum_{\{j,j'\}\in E,\, j,j'\neq i} \eta_{jj'} x_j x_{j'}\right)$$

Taking the ratio of the two previous quantities, the two last terms cancel out and we get

$$\mathbb{P}(X_i = x_i|X_{-i} = x_{-i}) = \frac{\exp\left(x_i\eta_i + \sum_{j\in N_j} x_i x_j \eta_{ij}\right)}{1 + \exp\left(\eta_i + \sum_{j\in N_j} x_j \eta_{ij}\right)}$$

In particular:

$$\begin{aligned}
\mathbb{P}(X_i = x_i|X_{-i} = x_{-i}) &= \frac{\exp\left(\eta_i + \sum_{j\in N_j} x_j \eta_{ij}\right)}{1 + \exp\left(\eta_i + \sum_{j\in N_j} x_j \eta_{ij}\right)} \\
&= \left(1 + \exp\left(-(\eta_i + \sum_{j\in N_i} \eta_{ij} x_j)\right)\right)^{-1} \\
&= \sigma\Big(\eta_i + \sum_{j\in N_i} \eta_{ij} x_j\Big),
\end{aligned}$$

where σ is the logistic function $\sigma : z \mapsto (1 + e^{-z})^{-1}$.

Without surprise, the conditional distribution $\mathbb{P}(X_i = x_i | X_{-i} = x_{-i})$ only depends on the variables that are neighbors of i in the graph and that form its Markov blanket, since we must have

$$\mathbb{P}(X_i = x_i | X_{-i} = x_{-i}) = \mathbb{P}(X_i = x_i \mid X_{N_i} = x_{N_i}).$$

Since the conditional distribution of X_i given all other variable is Bernoulli, it is easy to sample it, using a uniform random variable.

Proposition 1 *Random scan Gibbs sampling satisfies detailed balance for π the Gibbs distribution of interest (i.e. the distribution of the graphical model).*

Proof Let us consider one step of the random scan Gibbs sampling algorithm starting from π, the distribution of the graphical model. The idea is to prove the reversibility. We first prove the result for an index i fixed, that is we prove that the transition $q_{i,Gibbs}(x^{t+1} \mid x^t)$ that only resamples the ith coordinate of x^t is reversible for π. We write $p_\pi(x_i|x_{-i})$ the conditional distribution $p_\pi(x_i|x_{-i}) = \pi(x_i, x_{-i})/(\sum_{x'_{-i}} \pi(x_i, x'_{-i}))$ of the Gibbs distribution π. Using the Kronecker symbol δ defined by $\delta(x, y) = 1$ if $x = y$ and $\delta(x, y) = 0$ else we have:

$$\begin{aligned}
\pi(x^t)\, q_{i,Gibbs}(x^{t+1} \mid x^t) &= \pi(x^t)\, \delta(x^{t+1}_{-i}, x^t_{-i})\, p_\pi(x^{t+1}_i \mid x^t_{-i}) \\
&= \pi(x^t_{-i})\, p_\pi(x^t_i | x^t_{-i})\, \delta(x^{t+1}_{-i}, x^t_{-i})\, p_\pi(x^{t+1}_i \mid x^t_{-i}) \\
&= \pi(x^{t+1}_{-i})\, p_\pi(x^t_i \mid x^{t+1}_{-i})\, \delta(x^t_{-i}, x^{t+1}_{-i})\, p_\pi(x^{t+1}_i \mid x^{t+1}_{-i}) \\
&= \pi(x^{t+1})\, q_{i,Gibbs}(x^t \mid x^{t+1}).
\end{aligned}$$

Detailed balance for $q_{i,Gibbs}$ is valid for any i. In the random scan case, the index i being chosen at random uniformly with probability $\frac{1}{d}$, the Gibbs transition is in fact:

$$\frac{1}{d} \sum_{i=1}^{d} q_{i,Gibbs}(x^{t+1} \mid x^t)$$

The result is then obtained by taking the average over i in the previous derivation. Thus π is a stationary distribution of the random scan Gibbs transition. ■

Proposition 2 *If, the Gibbs distribution π satisfies $\pi(x) > 0$ for all $x \in \mathcal{X}$, the MC defined by the Gibbs sampling algorithms (cyclic and random variants) converge in distribution to π.*

Exercise 1 *Extend Gibbs method to Potts model.*

Exercise 2 *Prove that the Gibbs transition is a special case of Metropolis-Hastings proposal that is always accepted.*

9.2 Variational inference

9.2.1 Overview

The goal is to do approximate inference without using sampling. Indeed, algorithms such as Metropolis-Hastings or Gibbs sampling can be very slow to converge; besides, in practice, it is very difficult to find a good stopping criterion. People working on MCMC methods try to find clever tricks to speed up the process, hence the motivation for variational methods.

Let us consider a distribution on $\mathcal{X}$ finite (but usually very large) and Q an exponential family with $q_\eta(x) = \exp(\eta^T \phi(x) - A(\eta))$. Let us assume that the distribution of interest p, that is for example the distribution of our graphical model that we are working with, is in Q. The goal is to compute $\mathbb{E}_p[\phi(x)]$.

Computing this expectation corresponds to probabilistic inference in general. For example, for Potts model, using the notation $[K] := \{1, \dots, K\}$, we have

$$\phi(x) = \begin{pmatrix} (x_{ik})_{i \in V, k \in [K]} \\ (X_{ik} X_{jl})_{ij \in E;\, k,l \in [K]} \end{pmatrix}$$

We recall that: $p = \text{argmin}_q D(q||p)$ where:

$$D(q||p) = \sum_{x \in \mathcal{X}} q(x) \log \frac{q(x)}{p(x)} = \mathbb{E}_q[-\log p(X)] - H(q)$$

Since p is in Q, it is associated with a parameter η:

$$\begin{aligned} \mathbb{E}_q[-\log p(X)] &= \mathbb{E}_q\left[-\eta^T \phi(X) + A(\eta)\right] \\ &= -\eta^T \underbrace{\mathbb{E}_q[\phi(X)]}_{\mu(q)} + A(\eta) \end{aligned}$$

where $\mu(q)$ is the moment parameter (see course on exponential families). Thus we have:

$$-D(p||q) = \eta^T \mu(q) + H(q) - A(\eta)$$

This quantity is always negative (≤ 0) thus, for all q, $A(\eta) \geq \eta^T \mu(q) + H(q)$. Maximizing with respect to q in the exponential family leads to:

$$\boxed{A(\eta) = \max_{q \in Q} \eta^T \mu(q) + H(q)} \tag{9.3}$$

and the unique value of q that attains the maximum is p.

Remark 9.2.1 *It is possible here to get rid of q and express things only in terms of the moment. It is indeed a way to parameterize the distribution q : for a μ realizable in the exponential family there is a single distribution q_μ. The maximization problem becomes:*

$$\max_{\mu \in \mathcal{M}} \eta^T \mu + \tilde{H}(\mu),$$

where $\tilde{H}(\mu) = H(q_\mu)$ and where $\mathcal{M}$ is called the marginal polytope and is the set of all possible moments[1]*. The maximum is only attained for $\mu^* = \mu(p) = \mathbb{E}_p[\phi(X)]$, which is exactly the expectation that needs to be computed.*

It turns out that it is possible to show that $\tilde{H}$ is always a concave function, so that the optimization problem above is a convex optimization problem.

It is interesting to note that we have thus turned the probabilistic inference problem, which, a priori, required to compute expectations, that is integrals, into an optimization problem, which is furthermore convex. Unfortunately this convex optimization problem is NP-hard to solve in general because it solves the NP-hard probabilistic inference problem, and it is not possible to escape the fact that the latter is NP-hard. This optimization problem is thus in general intractable and this is because of two reasons:

- *For a general graph the marginal polytope $\mathcal{M}$ has number of faces which is exponential in the tree width of the graph.*
- *The function $\tilde{H}(\mu)$ can be extremely complicated to write explicitly.*

9.2.2 Mean field

In order to approximate the optimization problem it is possible either to change the set of distribution Q, the moments M or to change the definition of the entropy $\tilde{H}$. The mean field technique consists in choosing q in a set that makes all variables independent:

For a graphical models on variables $x_1 \dots x_d$, let us consider:

$$Q_{\perp\!\!\!\perp} = \{q \mid q(x) = q_1(x_1) \dots q_d(x_d)\},$$

the collection of distributions that make the variables $X_1, \dots, X_d$ independents.

We consider the optimization problem (9.3), but in which we replace Q by Q_π

$$\max_{q \in Q_{\perp\!\!\!\perp}} \eta^T \mu(q) + H(q). \tag{9.4}$$

Note that in general $p \notin Q_\pi$ so that the solution cannot be exactly $\mu(p)$.

In order to write this optimization problem for a Potts model, we need to write explicitly $\eta^T \mu(q)$ and $H(q)$

[1] We have seen in the course on exponential families that the distribution of maximum entropy q under the moment constraint $\mathbb{E}_q[\phi(X)] = \mu$ is also, when it exists, the distribution of maximum likelihood in the exponential family associated with the sufficient statistic ϕ. This essentially – but not exactly – shows that for any moment μ there exists a member of the exponential family, say q, such that $\mu = \mu(q)$. In fact, to be rigorous one has to be careful about what happens at points of the boundary of the set $\mathcal{M}$: the correct statement is that for every μ in the interior of $\mathcal{M}$ there exists a distribution q in the exponential family such that $\mu(q) = \mu$. The points on the boundary of $\mathcal{M}$ are only corresponding to limits of distributions of the exponential family that can be degenerate, like the Bernoulli distribution with probability 1 (or 0) for example in the Bernoulli family case, which are themselves not in the family.

Moments in the mean field formulation

$$\begin{aligned}\eta^T \mu(q) &= \eta^T \mathbb{E}_q\left[\phi(X)\right] \\ &= \sum_{i\in V, k\in[K]} \eta_{ik}\, \mathbb{E}_q\left[X_{ik}\right] + \sum_{(i,j)\in E} \eta_{ijkl}\, \mathbb{E}_q\left[X_{ik}X_{ji}\right]\end{aligned}$$

We have

$$\mathbb{E}_q\left[X_{ik}\right] = \mathbb{E}_{q_i}\left[X_{ik}\right] = \mu_{ik}(q)$$

On the other hand, the independence of the variables lead to:

$$\mathbb{E}_q\left[X_{ik}X_{jl}\right] = \mathbb{E}_{q_i}\left[X_{ik}\right]\mathbb{E}_{q_j}\left[X_{jl}\right] = \mu_{ik}\,\mu_{jl}$$

Note that if we had not constrained q to make these variables independent, we would in general have a moment here of the form $\mathbb{E}_q\left[X_{ik}X_{jl}\right] = \mu_{ijkl}$. This is the main place where the mean field approximation departs from the exact variational formulation (9.3).

Entropy $H(q)$ in the mean field formulation

By independence of the variables: $H(q) = H(q_1) + \cdots + H(q_d)$. Recall that q_i is the distribution on a single node, and that X_i is a multinomial random variable:

$$H(q_i) = -\sum_{k=1}^{K} \mathbb{P}_{q_i}(X_{ik}=1)\log \mathbb{P}_{q_i}(X_{ik}=1) = -\sum_{k=1}^{K} \mu_{ik}\log\mu_{ik}$$

Mean field formulation for the Potts model

In the end, putting everything together the optimization problem (9.4) can be written as

$$\begin{aligned}\max_{\mu}\ & \sum_{i,k}\eta_{ik}\,\mu_{ik} + \sum_{i,j,k,l}\eta_{ijkl}\mu_{ik}\mu_{jl} - \sum_{i,k}\mu_{ik}\log\mu_{ik} \\ \text{s.t. } & \forall i,k,\ \mu_{ik}\geq 0 \\ & \forall i,\ \sum_{k=1}^{K}\mu_{ik} = 1.\end{aligned}$$

The problem is simple to express, however we cannot longer expect that it will solve our original problem (9.3), because by restricting to the set $Q_{\perp\!\!\!\perp}$, we have restrained the forms that the moment parameters $\mu_{ijkl} := \mathbb{E}[X_{ik}X_{jl}]$ can take. In particular since p is not in $Q_{\perp\!\!\!\perp}$ in general, the optimal solution of the mean field formulation does not retrieve the correct moment parameter $\mu(p)$. The approximation will be reasonable if $\mu(p)$ is not too far from the sets of moments that are achievable by moments of distributions in $Q_{\perp\!\!\!\perp}$, since the moments of p are approximated by the moments of the closest independent distribution. Note

however that the mean field approximation is much more subtle than ignoring the binary potentials in the model, which would be a too naive way of finding an "approximation" with an independent distribution.

One difficulty though is that the objective function is no longer concave, because of the products $\mu_{ik}\mu_{jl}$ which arise because of the independence assumption from the mean field approximation. Coordinate descent on each of the μ_i (not the μ_{ik}) is an algorithm of choice to solve this kind of problem. To present the algorithm we consider the case of the Ising model, which is a special case of the Potts model with 2 states for each variable.

Mean field formulation for the Ising model

When working with the Ising model is simple to reduce the number of variables by using the fact that if $\mu_{i2} = 1 - \mu_{i1}$, we therefore write μ_i for μ_{i1} and the mean field optimization problem becomes

$$\begin{aligned} &\max_{\mu} \sum_i \eta_i \, \mu_i + \sum_{i,j} \eta_{ij} \, \mu_i \mu_j - \sum_i \big(\mu_i \log \mu_i + (1-\mu_i)\log(1-\mu_i)\big) \\ &\text{s.t.} \quad \mu_i \in [0,1]. \end{aligned}$$

The stationary points for each coordinate correspond to the zeros of the partial derivatives:

$$\frac{df}{d\mu_i} = \eta_i + \sum_{j \in N_i} \eta_{ij}\mu_j - \log \frac{\mu_i}{1-\mu_i}$$

So that

$$\begin{aligned} \frac{df}{d\mu_i} = 0 \;&\Leftrightarrow\; \log \mu_i/(1-\mu_i) = \eta_i + \sum_{j \in N_i} \eta_{ij}\mu_j \\ &\Leftrightarrow\; \mu_i^* = \sigma(\eta_i + \sum_{j \in N_i} \eta_{ij}\mu_j), \end{aligned}$$

where σ is the logistic function $\sigma : z \mapsto (1+e^{-z})^{-1}$.

Note that in Gibbs sampling $x_i^{t+1} = 1$ with probability $\sigma(\eta_i + \sum_{j \in N_i} \eta_{ij} x_j)$. This is called mean field because the sampling is replaced by an approximation where it is assumed that the sample value is equal to its expectation, which for the physicist correspond to the mean field in the ferromagnetic Ising model.

Finally, lets insist that the mean field formulation is only one of the formulations for variational inference, there are several other ones, among which structured mean field, expectation propagation, loopy belief propagation (which can be reinterpreted as as solving a variational formulation as well), tree-reweighted variational inference, etc.

Introduction to probabilistic graphical models **2015/2016**

Lecture 10 — December 3, 2015

Lecturer: Simon Lacoste-Julien *Scribes: Gauthier Gidel and Lilian Besson*

Note: These scribed notes have only been lightly proofread.

10.1 Bayesian Method

10.1.1 Introduction

Vocabulary:

- *a priori* or prior: $p(\theta)$
- *likelihood*: $p(x|\theta)$
- *marginal* likelihood: $\int p(x|\theta)\, p(\theta)\, d\theta$
- *a posteriori* or posterior: $p(\theta|x)$

Caricature Bayesian vs Frequentist:

1. the *Bayesian* is "optimistic": he thinks that he can come up with good models and obtain a method by "pulling the Bayesian crank" (basically a high dimensional integral),
2. the *frequentist* is more "pessimistic" and uses analysis tools.

The Bayesian formulation enables us to introduce the a priori information in the process of estimation. For instance , let's imagine that we play heads or tails. The Bayesian model is:

$$X_i \in \{0,1\}, \qquad X_i|\theta \sim Ber(\theta), \quad p(x_i|\theta) = \theta^{x_i}\,(1-\theta)^{1-x_i}$$

the graphical model associated is represented on Figure 10.1.

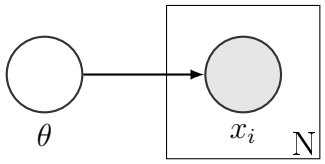


Figure 10.1. Graphical model of the biased coin game

Now we can compute the posterior:

$$p(\theta|x_{1:n}) \propto p(x_{1:n}|\theta)p(\theta)$$

then

$$p(\theta|x_{1:n}) = \theta^{n_1}(1-\theta)^{n-n_1}\mathbf{1}_{[0,1]}(\theta) = Beta(\alpha,\beta)$$

where $n_1 = \sum_{i=1}^{n} x_i$ is the number of 1, $\beta = n - n_1 + 1$ and $\alpha = n_1 + 1$.

Question: what is the probability of head on the next flip?

- Frequensist: $\hat{\theta}_{ML} = n_1/n$ by a maximum likelihood approach.
- Bayesian: $p(x_{n+1}|x_{1:n}) = \int p(x_{n+1}|\theta)p(\theta|x_{1:n})d\theta$, where $p(\theta|x_{1:n})d\theta$ is the posterior distribution. Then,

  $$\hat{\theta}_B = \frac{\alpha}{\alpha+\beta} = \frac{n_1+1}{n+2}$$

  hence,

  $$\hat{\theta}_B = \frac{n_1}{n}\left[\frac{n}{n+2}\right] + \frac{1}{2}\left[\frac{2}{n+2}\right] = \rho_n\hat{\theta}_{ML} + (1-\rho_n)\,\hat{\theta}_{prior}$$

  is a convex combination of $\hat{\theta}_{ML}$ and $\hat{\theta}_{prior}$. Then we can notice that for $n = 0$, the quantity $\hat{\theta}_B = \frac{1}{2}$ whereas $\hat{\theta}_{ML}$ is not defined. It underlines the importance of the prior distibution:

  - with an "unknown" coin, we've got the information a priori : we'll use the uniform law for $p(\theta)$.
  - with a "normal" coin , we'll use a distribution with an important concentration of mass around 0,5 for $p(\theta)$.

  For a Bayesian, offering a "limited" estimator, as the maximum likelihood estimator, which gives a unique value for θ, is not enough because the estimator itself do not translate the inherent uncertainty of the learning process. Thus, its estimator will be the density a posteriori, obtained from the Bayes rule, which is written in continuous notations as:

  $$p(\theta|x) = \frac{p(x|\theta)\,p(\theta)}{\int p(x|\theta)\,p(\theta)\,d\theta}$$

  The Bayesian specifies the uncertainty with distributions that form its estimator, rather than combining an estimator with confidence intervals.

  If the Bayesian is forced to produce a limited estimator, he uses the expectation of the underlying quantity under the a posteriori distribution; for instance for θ:

$$\mu_{post} = \mathbb{E}[\theta|D] = \mathbb{E}[\theta|x_1,x_2,\ldots,x_n] = \int \theta p(\theta|x_1,x_2,\ldots,x_n)\,d\theta$$

For more details about Bayesians see subsection B.1 and B.1.1 in annex.

We then need to show that $\hat{\theta}_{ML} \to \theta^*$. Its variance is the variance of a Beta law

$$\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} = \left(\frac{n_1}{n}\right)\left(1-\frac{n_1}{n}\right)\cdot O\left(\frac{1}{n}\right) = \hat{\theta}_{ML}\left(1-\hat{\theta}_{ML}\right)O\left(\frac{1}{n}\right)$$

then the posterior covariance vanishes and

$$\hat{\theta}_B \overset{a.s.}{\to} \hat{\theta}_{ML} \overset{a.s.}{\to} \theta^*$$

where θ^* is the "true" parameter of the model.

10.1.2 Bernstein von Mises Theorem

It says that if prior puts non-zero mass around the true model θ^*, then posterior asymptotically concentrate around θ^* as a Gaussian.

Revisiting example Consider repeating several times the experiment above: T coins picked randomly each flipped n times. (Figure 10.2)

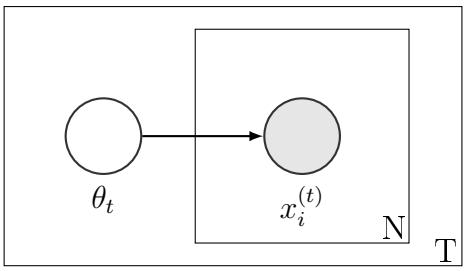


Figure 10.2. Graphical model of the biased coin game repeated T times

As a frequentist, empirical distribution on $x_{1:n}$ will converge (as $T \to \infty$) to

$$p(x_1,\ldots,x_n) = \int_\theta \left(\prod_{i=1}^n p(x_i|\theta)\right) p(\theta)d\theta$$

where $p(\theta)$ is the distribution of coins of parameter θ in the jar and $\prod_{i=1}^n p(x_i|\theta)$ is the mixture distribution. Note that $X_1,\ldots,X_n$ are **NOT** independent.

On the other hand, for all $\pi \in \mathcal{S}_n$

$$p(x_1,\ldots,x_n) = p\left(x_{\pi(1)},\ldots,x_{\pi(n)}\right)$$

10.1.3 Exchangeable situations

Exchangeablility

The random variables X_1, X_2, ..., X_n are exchangeable if they have the same distribution as $X_{\pi(1)}$,$X_{\pi(2)}$, ..., $X_{\pi(n)}$ for any permutation of indices $\pi \in \mathcal{S}_n$.

Infinite Exchangeablility

The definition naturally generalizes to infinite families (indexed by $\mathbb{N}$). The random variables $X_1, X_2, \ldots$ are exchangeable if every finite subfamily $X_{i_1}, \ldots, X_{i_n}$ is exchangeable.

de Finetti's theorem

X_1, X_2, ... are infinitely exchangeable, if and only if $\exists!\ p(\theta)$ (on some space Θ) such that

$$\forall n \in \mathbb{N},\ p\left(x_1, x_2, \ldots, x_n\right) = \int \left(\prod_{i=1}^{n} p\left(x_i|\theta\right) \right) p(\theta) d\theta$$

Why do we care about exchangeable situations?

The i.i.d. variables are a particular case of the situation of exchangeable variables, that we see in practice. However when the i.i.d. data are combined with non scalar observations, the different components are no longer independent. In some cases, those components are nonetheless exchangeable. For instance in a text, words are shown as sequences that are not exchangeable because of the syntax. But if we forget the order of the words as in the "bag of word" model, then the components are exchangeable. It's the basic principle used in the LDA model.

Multinomial example

Let $X|\theta \sim Mult(\theta, 1)$ where $\theta \in \Delta_k$ i.e.

$$p(X = l|\theta) = \theta_l \quad \text{and} \quad \sum_{l=1}^{k} \theta_l = 1,\ 0 \leq \theta_l \leq 1.$$

for that distribution we have,

$$\hat{\theta}_l^{ML} = \frac{n_l}{n}$$

hence if $k \geq n$ there exists a l such that $\hat{\theta}_l^{ML} = 0$.

In that case this frequentist model overfits. In the Bayesian model one puts a prior on $\Delta_k = \Theta$, but which one? A convenient property of prior families is "conjugacy", introduced below:

Conjugacy Consider a family of distribution

$$F = \{p(\theta|\alpha) \ : \ \alpha \in \mathcal{A}\}.$$

One says that F is a "conjugate family" for the observation model $p(x|\theta)$ if the posterior

$$p(\theta|x,\alpha) = \frac{p(x|\theta)p(\theta|\alpha)}{p(x|\alpha)}$$

belongs to the same family F than the prior, i.e.

$$\exists\, \alpha' \in \mathcal{A} \quad s.t \quad p(\theta|x,\alpha) = p(\theta|\alpha')$$

For the multinomial distribution it gives us

$$p(x_{1:n}|\theta) = \prod_{l=1}^{n} p(x_l|\theta) = \prod_{l=1}^{n} \theta_l^{n_l}$$

so if $p(\theta) \propto \prod_{l=1}^{n} \theta_l^{\alpha_l}$, then $p(x_{1:n}|\theta) \propto \prod_{l=1}^{n} \theta_l^{\beta_l}$.

Dirichlet Distribution

The Dirichlet distribution is the conjugate of the Multinomial law (see on Wikipédia for more details).

$$p(\theta_1, \theta_2, \ldots, \theta_K) = \frac{\Gamma(\alpha_1 + \alpha_2 + \ldots + \alpha_K)}{\Gamma(\alpha_1)\Gamma(\alpha_2)\ldots\Gamma(\alpha_K)} \theta_1^{\alpha_1 - 1}\theta_2^{\alpha_2 - 1} \ldots \theta_K^{\alpha_K - 1} d\mu(\theta)$$

Where μ stands for the uniform measure on $\Delta_K = \left\{s \in \mathbb{R}^K \mid \sum_i s_i = 1\,;\, \forall i,\ s_i \geq 0\right\}$ (K-dim simplex).

- $\mathbb{E}[\theta_l|\alpha_1, \ldots, \alpha_K]$,
- $\mathbb{V}(\theta_l) \equiv O\left(\frac{1}{\sum_{j=1}^{K} \alpha_j}\right)$,
- If $\alpha_l = 1$ for all l then one gets an uniform distribution,
- if $k = 2$ one gets the Beta distribution,
- if there exists l such that $\alpha_l < 1$ one gets a $\cup$ shape distribution,
- if $\alpha_l \geq 1$ for all l, one gets a $\cap$ (unimodal bump).

For the multinomial model, if the we assume that the prior is

$$p(\theta) = Dir(\theta|\alpha)$$

then the posterior is

$$p(\theta|x_{1:n}) \propto \prod_{l=1}^{K} \theta_l^{n_l+\alpha_l-1}$$

and the posterior mean is

$$\mathbb{E}\left[\theta_l|x_{1:n}\right] = \frac{n_l + \alpha_l}{n + \sum_{j=1}^{K} \alpha_j}$$

for instance with $\alpha_l = 1$ for all l it adds 1, "smoothing" the maximum likelihood estimator.

$$\mathbb{E}\left[\theta_l|x_{1:n}\right] = \frac{n_l + 1}{n + K}$$

NB One can consider that posterior can be used for prior of next observation. This is the *sequential approach*.

10.2 Bayesian linear regression

Let us assume that

$$y = \omega^T x + \epsilon \tag{10.1}$$

where $\epsilon \sim \mathcal{N}(0, \sigma^2)$. Then the observation issue

$$p(y|x) = \mathcal{N}\left(y \,|\, \omega^T x \,,\, \sigma^2\right)$$

Then if we also choose a Gaussian prior on ω.

$$p(\omega) = \mathcal{N}\left(\omega \,;\, 0, \frac{I_n}{\lambda}\right)$$

then the posterior is also a Gaussian with the following parameters

- covariance: $\hat{\Sigma}_n = \lambda I_n + \frac{X^T X}{\sigma^2}$
- mean: $\hat{\mu}_n = \hat{\Sigma}_n^{-1}\left(X^T \vec{y}/\sigma^2\right)$

where

$$X = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad \text{and} \quad \vec{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}$$

the covariance and the mean are the same as the ones for the *ridge regression* with $\tilde{\lambda} = \lambda\sigma^2$.

As a Bayesian: compute predictive distribution

$$\begin{aligned} p(y_{new}|x_{new}, x_{1:n}, y_{1:n}) &= \int_\omega p(y_{new}|x_{new}, \omega) p(\omega|data) d\omega \\ &= \mathcal{N}\left(y_{new}|\hat{\mu}_n^T x_{new}, \sigma^2_{predictive}\right) \end{aligned}$$

where

$$\sigma^2_{predictive}(x_{new}) = \sigma^2 + x_{new}^T \hat{\Sigma}_n x_{new},$$

the real number σ comes from the noise model and the second quantity of the right hand side comes from the posterior covariance.

10.3 Model Selection

10.3.1 Introduction

Let's consider two models $M_1 \subset M_2$ with $\Theta_1 \subset \Theta_2$. We define:

$$\widehat{\Theta}_{M_i} = \arg\max_{\theta \in \Theta_i} \log\left(p_\theta\left(x_1, x_2, \ldots, x_n\right)\right)$$

where $i \in \{1, 2\}$.

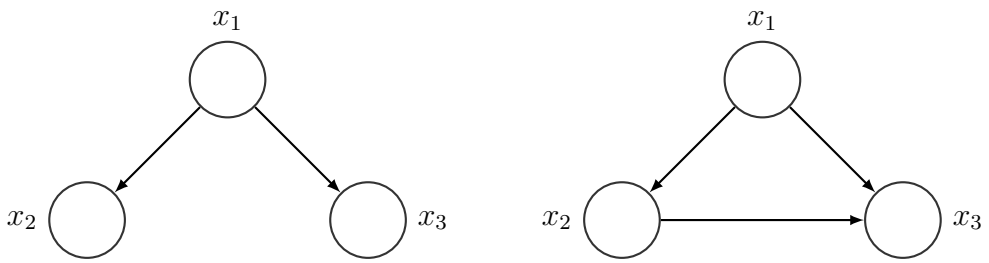


Figure 10.3. Example of Model Section for $n = 2$ (M_1 on the l.h.s and M_2 on the r.h.s)

We can't use the maximum likelihood as a score since we have by definition:

$$\log\left(p_{\widehat{\Theta}_{M_2}}\right) \geq \log\left(p_{\widehat{\Theta}_{M_1}}\right).$$

We are interested in the capacity of the generalisation of the model: we'd like to avoid over-fitting. Commonly, one way of dealing with that task is to select the size of the model by cross-validation. Here, we'll not develop it furthermore.

In this part we present the *Bayes factors*, which gives us the main Bayes principal for selecting models. Also we will show the link with the penalised version BIC, (Bayesian Information Criterion) which is used by the frequentists so as to "correct" the maximum likelihood and which has good proprieties. The issue with the selection model ask is the issue with the selection of the variables which are an active topic of research. There are others ways of penalising the maximum likelihood and of selecting models.

If p_0 is the distribution of the real data, we wish to choose between difference models $(M_i)_{i \in I}$ by maximising $\mathbb{E}_{p_0}\left[\log\left(p_{M_i}\left(X^*|D\right)\right)\right]$, where X^* is a new test sample distributed as p_0 (in fact, it's still the maximum likelihood principle but we take the expectation on new data).

In the Bayesian framework, we can compute the marginal probability of data for a given model

$$\int p\left(x_1, x_2, \ldots, x_n|\theta\right) p\left(\theta|M_i\right) d\theta = p\left(D|M_i\right)$$

and, by applying the Bayes rule, compute the a posteriori probability of the model:

$$p\left(M_i|D\right) = \frac{p\left(D|M_i\right)p\left(M_i\right)}{p\left(D\right)}$$

10.3.2 Bayes Factor

Let's introduce the Bayes factors, which enables us to compare two models:

$$\frac{p\left(M_1|D\right)}{p\left(M_2|D\right)} = \frac{p\left(D|M_1\right)p\left(M_1\right)}{p\left(D|M_2\right)p\left(M_2\right)}$$

The marginal probability of data

$$p\left(D|M_i\right) = p\left(x_1, x_2, \ldots, x_n|M_i\right)$$

can decompose itself in a sequential way by using:

$$p\left(x_n|x_1, x_2, \ldots, x_{n-1}, M\right) = \int p\left(x_n|\theta\right)p\left(\theta|x_1, x_2, \ldots, x_{n-1}, M\right)d\theta.$$

Indeed, we get:

$$p(D|M) = p(x_n|x-1, \ldots, x_{n-1}, M)\, p(x_{n-1}|x-1, \ldots, x_{n-2}, M) \ldots\, p(x_1|M)$$

Such as

$$\frac{1}{n}\log p\left(D|M_i\right) = \frac{1}{n}\sum_{i=1}^{n}\log p(x_i|x_1, \ldots, x_{i-1}, M) \simeq \mathbb{E}_{p_0}\left[\log p_M\left(X|D\right)\right]$$

10.3.3 Bayesian Information Criterion

The Bayesian score is approximated by the BIC:

$$\log p\left(D|M\right) = \log p_{\widehat{\theta}_{MV}}\left(D\right) - \frac{K}{2}\log\left(n\right) + O\left(1\right)$$

With $p_{\widehat{\theta}_{MV}}\left(D\right)$ the data's distribution when the parameter is the maximum likelihood estimator $\widehat{\theta}_{MV}$, K is the number of parameters of the model and n the number of observations.

In the following section, we outline the proof of this result in the case of an exponential family given by $p\left(x|\theta\right) = \exp\left(\langle\theta, \phi\left(X\right)\rangle - A\left(\theta\right)\right)$.

10.3.4 Laplace's Method

$$\begin{aligned} p\left(D|M\right) &= \int \prod_{i=1}^{n} p\left(x_i|\theta\right) p\left(\theta\right) d\theta \\ &= \int \exp\left(\left\langle \theta, n\bar{\phi}\right\rangle - n\, A\left(\theta\right)\right) p\left(\theta\right) d\theta \end{aligned}$$

$$\begin{aligned} \langle \theta, n\bar{\phi}\rangle - n\, A(\theta) = \langle \widehat{\theta}, n\bar{\phi}\rangle - n\, A(\widehat{\theta}) + \langle \theta - \widehat{\theta}, n\bar{\phi}\rangle \\ - n(\theta - \widehat{\theta})^T \nabla_\theta A(\widehat{\theta}) - \frac{1}{2}(\theta - \widehat{\theta})^T n \nabla^2_\theta A(\widehat{\theta})(\theta - \widehat{\theta}) \\ + \mathrm{R}_n \end{aligned}$$

where R_n is a negligible rest.

But the maximum likelihood is the dual of the maximum entropy: $\max H(p_\theta)$ such that $\mu(\theta) = \bar{\phi}$.

$$\mu(\widehat{\theta}) = \bar{\phi}$$

$$p(D|M) \simeq \exp(\langle \widehat{\theta}, n\bar{\phi}\rangle - n\, A(\widehat{\theta})) \times \int \exp\left(-\frac{1}{2}(\theta - \widehat{\theta})^T n \widehat{\Sigma} (\theta - \widehat{\theta})\right) p(\theta) d\theta$$

However:

1. the information of fisher is equal to $\widehat{\Sigma}^{-1}$
2. $\displaystyle\int \exp\left(-\frac{1}{2}\left(\theta - \widehat{\theta}\right)^T n \widehat{\Sigma}\left(\theta - \widehat{\theta}\right)\right) p\left(\theta\right) d\theta \simeq c\sqrt{\left(2\pi\right)^k \left|\frac{\widehat{\Sigma}^{-1}}{n}\right|}$

Thus:

$$\begin{aligned} \log p\left(D|M\right) &= \log p_{\widehat{\theta}}\left(X\right) + \frac{1}{2}\log\left(\left(2\pi\right)^k \left|\frac{\widehat{\Sigma}^{-1}}{n}\right|\right) \\ &= \log p_{\widehat{\theta}}\left(X\right) + \frac{k}{2}\log\left(2\pi\right) + \frac{1}{2}\log\left(\left(\frac{1}{n}\right)^k \left|\widehat{\Sigma}^{-1}\right|\right) \\ &= \log p_{\widehat{\theta}}\left(X\right) + \frac{k}{2}\log\left(2\pi\right) - \frac{k}{2}\log\left(n\right) + \frac{1}{2}\log\left(\left|\widehat{\Sigma}^{-1}\right|\right) \end{aligned}$$

The main reason why presenting the BIC is that a theorem prove the consistency of the BIC. In other words, when the number of observations is sufficient, thanks to this criterion we choose with a probability that converges to 0, a model that satisfies:

$$M_k \in \mathrm{Argmax}_M\, \mathbb{E}_{p_0}\left[\log\left(p_{\widehat{\theta}_{MV}}\left(X\,;\,M\right)\right)\right]$$

To bring a quick clarification about the notations used in this part (model selection), please read below. The notation is a bit confusing (it was used for example in Bishop's book, but is a bit sloppy).

From the Bayesian perspective, we could treat the model choice as a random variable M. In the M_1 vs. M_2 vs. M_3 example, there are only 3 models, and thus M is a discrete variable with 3 possible values ($M = M_1$, $M = M_2$ or $M = M_3$).

Therefore, when we were writing quantities like the Bayes factor $p(M_1|D)/p(M_2|D)$, It really meant $p(M = M_1|D)/p(M = M_2|D)$. It did not mean that M_1 and M_2 were two different random variables which can take complicated values (someone asked what space M_1 was in and it seemed very complicated – what is meant is just that M is an index in possible (few) models).

D was the data random variable as usual. The mixing of random variables (here M) vs. their possible values ($M = 1, 2$ etc) in the same notation (like $p(M_1|D)$) is usual but confusing; better to use the explicit $p(M = M_1|D)$ notation to distinguish a value vs. a generic random variable....

However, in general, M could be as complicated as we want. For example, it could be a vector of hyper-parameters for the prior distributions. Or it could also have binary component indicating the absence or presence of an edge in graphical model, etc. It does not have to just be an index. It could even be a continuous objects !

It is also fine to have infinite dimensional objects[1]. For example, consider the latent variable model: x is observed, θ and α are latent variables; and M decides the prior over α. I.e. suppose $p(x|\theta, \alpha, M) = Multi(\theta, 1)$, $p(\theta|\alpha, M) = Dir(\theta|\alpha)$, and $p(\alpha|M) = M(\alpha)$ i.e. M ranges over possible distributions over the positive vector α. M here is quite a complicated object, but this is fine...

[1]This would be in the "non-parametric setting" – non-parametric = infinite dimensional.

Appendix A

A.1 Example of model

A.1.1 Bernoulli variable

Let's consider random variables $X_i \in \{0, 1\}$. We'll assume that the X_i are i.i.d. conditionally to θ. Then they follow a Bernoulli law:

$$p\left(x|\theta\right) = \theta^x \left(1-\theta\right)^{1-x}$$

A.1.2 Priors

Let's introduce the *distribution* Beta whose density on $[0, 1]$ is

$$p(\theta; \alpha, \beta) = \frac{1}{B(\alpha, \beta)} \theta^{\alpha-1}(1-\theta)^{\beta-1}$$

Where $B(\alpha, \beta)$ is a short-name of the Beta *function*:

$$\forall \alpha > 0, \forall \beta > 0, \; B\left(\alpha, \beta\right) = \int_0^1 \theta^{\alpha-1} \left(1-\theta\right)^{\beta-1} d\theta$$

And the Gamma function:

$$\Gamma\left(x\right) = \int_0^{+\infty} t^{x-1} \exp\left(-t\right) dt$$

We can show that $B\left(\alpha, \beta\right)$ is symmetric and satisfies:

$$B\left(\alpha, \beta\right) = \frac{\Gamma\left(\alpha\right)\Gamma\left(\beta\right)}{\Gamma\left(\alpha+\beta\right)}$$

We choose as the prior distribution on θ the Beta distribution:

$$p(\theta) \propto \theta^{\alpha-1}(1-\theta)^{\beta-1}$$

$$p(\theta) = \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha,\beta)}$$

A.1.3 A posteriori

$$p(\theta|x) = \frac{p(x,\theta)}{p(x)} \propto p(x,\theta)$$

But:

$$p(x,\theta) = \theta^{x}(1-\theta)^{1-x}\frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha,\beta)}$$

Hence:

$$p(\theta|x) \propto \frac{\theta^{x+\alpha-1}(1-\theta)^{1-x+\beta-1}}{B(\alpha,\beta)}$$

$$p(\theta|x) = \frac{\theta^{x+\alpha-1}(1-\theta)^{1-x+\beta-1}}{B(x+\alpha,1-x+\beta)}$$

Thus, if instead of considering a unique variable , we observe an i.i.d. sample of data, the joint distribution can be written as:

$$\theta^{\alpha-1}(1-\theta)^{\beta-1}\prod_{i=1}^{n}\theta^{x_i}(1-\theta)^{1-x_i}.$$

Let's introduce:

$$k = \sum_{i=1}^{n} x_i$$

Then we get:

$$p(\theta|x_1,x_2,\ldots,x_n) = \frac{\theta^{k+\alpha-1}(1-\theta)^{n-k+\beta-1}}{B(k+\alpha,n-k+\beta)}$$

A.2 Special case of the Beta distribution

We remind that:

$$\theta \sim Beta(\alpha,\beta)$$

For $\alpha=\beta=1$, we get a uniform prior.
For $\alpha=\beta>1$, we get a bell curve.
For $\alpha=\beta<1$, we get a U curve.

$\mathbb{E}[\theta] = \frac{\alpha}{\alpha+\beta}$
$\mathbb{V}[\theta] = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} = \frac{\alpha}{(\alpha+\beta)} \times \frac{\beta}{(\alpha+\beta)} \times \frac{1}{(\alpha+\beta+1)}$
For $\alpha > 1$ and $\beta > 1$, we get the mode: $\frac{\alpha-1}{\alpha+\beta-2}$.
In the case, let's write D for the data:

$$\theta_{post} = \mathbb{E}[\theta|D] = \frac{\alpha+k}{\alpha+\beta+n} = \frac{\alpha}{(\alpha+\beta)} \times \frac{(\alpha+\beta)}{(\alpha+\beta+n)} + \frac{n}{(\alpha+\beta+n)} \times \frac{k}{n}$$

We can see that the a posteriori expectation of the parameter is a convex combination of the maximum likelihood estimator and the prior expectation. It converges asymptotically to the maximum likelihood estimator .

If we use a uniform prior distribution, $\mathbb{E}[\theta|D] = \frac{k+1}{n+2}$. Laplace proposed to correct the frequentist estimator, it seemed odd to him that he was not defined in the absence of data. He proposed to add two virtual observation (0 and 1) such that in the absence of data the estimator equals $\frac{1}{2}$. This correction is known as *Laplace's correction*.

The variance of the a posteriori distribution decrease in $\frac{1}{n}$.

$$\mathbb{V}[\theta|D] = \theta_M(1-\theta_M)\frac{1}{(\alpha+\beta+n)}$$

We have chosen a sharper distribution around θ_M, in the same way than in a frequentist approach, the confidence intervals narrow around the estimator when the number of observations increase.

A.2.1 Playful propriety

$$p(x_1, x_2, \ldots, x_n) = \frac{B(k+\alpha, n-k+\beta)}{B(\alpha,\beta)} = \frac{\Gamma(\alpha+k)\Gamma(\beta+n-k)\Gamma(\alpha+\beta)}{\Gamma(\alpha+\beta+n)\Gamma(\alpha)\Gamma(\beta)} \tag{A.1}$$

Let's use this well-known property of the Gamma function:

$$\Gamma(n+1) = n!$$

$$\text{and} \qquad \forall x > -1,\ \Gamma(x+1) = x\Gamma(x)$$

such that

$$\Gamma(\alpha+k) = (\alpha+k-1)(\alpha+k-2)\ldots\alpha\Gamma(\alpha)$$

let's write $\alpha^{[k]} = \alpha(\alpha+1)\ldots(\alpha+k-1)$ and simplify the expression A.1:

$$p(x_1, x_2, \ldots, x_n) = \frac{\alpha^{[k]}\beta^{[n-k]}}{(\alpha+\beta)^{[n]}}$$

We shall note the analogy with the Polya urn model: let us consider $(\alpha + \beta)$ balls of colour: α are black, β are white. When drawing a first black ball, the probability of the event is:

$$\mathbb{P}(X_1 = 1) = \frac{\alpha}{\alpha + \beta}$$

After the drawing, we put back the ball in the urn and we add a ball of the same colour. Let's imagine that we draw again a black ball then the probability of this event is:

$$\mathbb{P}(X_1 = 1,\, X_2 = 1) = \mathbb{P}(X_1 = 1)\,\mathbb{P}(X_2 = 1|X_1 = 1) = \frac{\alpha}{\alpha + \beta} \times \frac{\alpha + 1}{\alpha + \beta + 1}$$

However:

$$\mathbb{P}(X_1 = 1,\, X_2 = 0) = \frac{\alpha}{\alpha + \beta} \times \frac{\beta}{\alpha + \beta + 1}$$

In more general case , we show by recurrence that the marginal probability of obtaining some sequence of colours by drawing from a Polya urn is exactly the marginal probability of obtaining the same result from the marginal model, obtained by integrating on a priori *theta*. First, this show that drawings from a Polya urn are exchangeable; Secondly, the mechanism of this type of urn, and its exchangeability, we'll be useful for the Gibbs sampling and for the same type of Bayesian models.

A.2.2 Conjugate priors

Let $\mathbb{F}$ be a set. We assume that $p(x|\theta)$ known, we deduce from that: $p(\theta) \in \mathbb{F}$ such that $p(\theta|x) \in \mathbb{F}$. We say that $p(\theta)$ is conjugated to the model $p(x|\theta)$.

Exponential model

Let's consider:

$$p(x|\theta) = \exp(\langle\theta, \phi(x)\rangle - A(\theta))$$
$$p(\theta) = \exp(\langle\alpha, \theta\rangle - \tau A(\theta) - B(\alpha, \tau))$$

For $p(x|\theta)$, θ is the canonical parameter. For $p(\theta)$, α is the canonical parameter and θ is the sufficient statistic. Let us note that B do not stand for the Beta distribution.

$$p(\theta|x) \propto p(x|\theta)\,p(\theta) \propto \exp(\langle\theta, \phi(x)\rangle - A(\theta) + \langle\alpha, \theta\rangle - \tau A(\theta) - B(\alpha, \tau))$$

Let us define:

$$\bar{\phi} = \frac{1}{n}\sum_{i=1}^{n} \phi(x_i)$$

Then:

$$p(\theta|x_i) \propto \exp(\langle\theta, \alpha + \phi(x_i)\rangle - (\tau + 1)\,A(\theta) - B(\alpha + \phi(x_i), \tau + 1))$$

$$p\left(\theta|x_1,x_2,\ldots,x_n\right) \propto \exp\left(\left\langle\theta,\alpha+n\bar{\phi}\right\rangle-(\tau+n)\,A\left(\theta\right)-B\left(\alpha+n\bar{\phi},\tau+n\right)\right)$$

$$p\left(x_1,x_2,\ldots,x_n\right) \propto \exp\left(B\left(\alpha,\tau\right)-B\left(\alpha+n\bar{\phi},\tau+n\right)\right)$$

Since the family is an exponential one,

$$\nu_{post}=\mathbb{E}\left[\theta|D\right]=\nabla_\alpha B\left(\alpha+n\bar{\phi},\tau+n\right)$$

θ_{MAP} results from:

$$\nabla_\theta p\left(\theta|x_1,x_2,\ldots,x_n\right)=0$$
$$\alpha+n\bar{\phi}=(\tau+n)\,\nabla_\theta A\left(\theta\right)=(\tau+n)\,\mu\left(\theta\right)$$

Thus we get $\mu_{MAP}=\mu\left(\theta\right)$ in the previous equation. Consequently:

$$\mu_{MAP}=\frac{\alpha+n\bar{\phi}}{\tau+n}=\frac{\alpha}{\tau}\times\frac{\tau}{\tau+n}+\frac{n}{\tau+n}\bar{\phi}$$

Univariate Gaussian

With and a priori on μ but not on σ^2

$$p\left(x|\mu,\sigma^2\right)=\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{1}{2}\frac{\left(x-\mu\right)^2}{\sigma^2}\right)$$

$$p\left(\mu|\mu_0,\tau^2\right)=\frac{1}{\sqrt{2\pi\tau^2}}\exp\left(-\frac{1}{2}\frac{\left(\mu-\mu_0\right)^2}{\tau^2}\right)$$

Thus:

$$\begin{aligned}p\left(D|\mu,\sigma^2\right)&=p\left(x_1,x_2,\ldots,x_n|\mu,\sigma^2\right)\\&=\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n\exp\left(-\frac{1}{2}\sum_{i=1}^{n}\frac{\left(x_i-\mu\right)^2}{\sigma^2}\right)\end{aligned}$$

$$\begin{aligned}p\left(\mu|D\right)&=p\left(\mu|x_1,x_2,\ldots,x_n\right)\\&=\exp\left(-\frac{1}{2}\left(\frac{\left(\mu-\mu_0\right)^2}{\tau^2}+\sum_{i=1}^{n}\frac{\left(x_i-\mu\right)^2}{\sigma^2}\right)\right)\\&=\exp\left(-\frac{1}{2}\left(\frac{\mu^2-2\mu\mu_0+\mu_0^2}{\tau^2}+\sum_{i=1}^{n}\frac{\mu^2-2\mu x_i+x_i^2}{\sigma^2}\right)\right)\\&=\exp\left(-\frac{1}{2}\left(\mu^2\Lambda-2\mu\eta+\left(\frac{\mu_0^2}{\tau^2}+\sum_{i=1}^{n}\frac{x_i^2}{\sigma^2}\right)\right)\right)\end{aligned}$$

Where:

$$\Lambda = \frac{1}{\tau^2} + \frac{n}{\sigma^2}$$
$$\eta = \frac{\mu_0}{\tau^2} + \frac{n\overline{x}}{\sigma^2}$$
$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Thus:

$$\begin{aligned}
\mu_{post} &= \mathbb{E}\left[\mu|D\right] \\
&= \frac{\eta}{\Lambda} \\
&= \frac{\frac{\mu_0}{\tau^2} + \frac{n\overline{x}}{\sigma^2}}{\frac{1}{\tau^2} + \frac{n}{\sigma^2}} \\
&= \frac{\sigma^2\mu_0 + n\tau^2\overline{x}}{\sigma^2 + n\tau^2} \\
&= \frac{\sigma^2}{\sigma^2 + n\tau^2}\mu_0 + \frac{n\tau^2}{\sigma^2 + n\tau^2}\overline{x}
\end{aligned}$$

And:

$$\begin{aligned}
\widehat{\Sigma}^2_{post} &= \mathbb{V}\left[\mu|D\right] \\
&= \frac{1}{\Lambda} \\
&= \frac{\sigma^2\tau^2}{\sigma^2 + n\tau^2}
\end{aligned}$$

Indeed, the variance decreases in $\frac{1}{n}$.

With an a priori on σ^2 but not onμ We get $p\left(\sigma^2\right)$ as an Inverse Gamma form.

With an a priori on μ and σ^2 Gaussian a priori on x and μ, Inverse Gamma a priori on σ^2. Please refer to the chapter 9 of the course handout (Jordan's polycopié).

Appendix B

B.1 A posteriori Maximum (MAP)

$$\begin{aligned}\theta_{MAP} &= \arg\max_{\theta} p\left(\theta|x_1, x_2, \ldots, x_n\right) \\ &= \arg\max_{\theta} p\left(x_1, x_2, \ldots, x_n|\theta\right) p\left(\theta\right)\end{aligned}$$

Because, with the Bayes rule:

$$p\left(\theta|x_1, x_2, \ldots, x_n\right) = \frac{p\left(x_1, x_2, \ldots, x_n|\theta\right) p\left(\theta\right)}{p\left(x\right)}$$

The a posteriori maximum is not really Bayesian, it's rather a slight modification brought to the frequentist estimator.

B.1.1 Predictive probability

In the Bayesian paradigm, the probability of a future observation x^* will be estimated by the *Predictive probability*:

$$\begin{aligned}p\left(x^*|D\right) &= p\left(x^*|x_1, x_2, \ldots, x_n\right) \\ &= \int p\left(x^*|\theta\right) p\left(\theta|x_1, x_2, \ldots, x_n\right) d\theta\end{aligned}$$

$$\begin{aligned}p\left(\theta|x_1, x_2, \ldots, x_n\right) &\propto p\left(x_n|\theta\right) p\left(x_1|\theta\right) p\left(x_2|\theta\right) \ldots p\left(x_{n-1}|\theta\right) p\left(\theta\right) \\ &\propto p\left(x_n|\theta\right) p\left(\theta|x_1, x_2, \ldots, x_{n-1}\right) p\left(x_1, x_2, \ldots, x_{n-1}\right) \\ &\propto p\left(x_n|\theta\right) p\left(\theta|x_1, x_2, \ldots, x_{n-1}\right) \frac{p\left(x_1, x_2, \ldots, x_{n-1}\right)}{p\left(x_1, x_2, \ldots, x_n\right)}\end{aligned}$$

A sequential calculus is possible since:

$$p(\theta|x_1, x_2, \ldots, x_n) = \frac{p(x_n|\theta)\, p(\theta|x_1, x_2, \ldots, x_{n-1})}{p(x_n|x_1, x_2, \ldots, x_{n-1})}$$

Vocabulary:

- a priori information: $p(\theta|x_1, x_2, \ldots, x_{n-1})$
- likelihood: $p(x_n|\theta)$
- a posteriori information: $p(\theta|x_1, x_2, \ldots, x_n)$

$$p(x_1, x_2, \ldots, x_n) = \int \prod_{i=1}^{n} p(x_i|\theta)\, p(\theta)\, d\theta$$

B.2 Naive Bayes

B.2.1 Introduction

Remarque: Contrary to its name, "Naive Bayes" is *not* a Bayesian method**.**

Let's Consider the following problem of classification $x \in \mathbb{X}^p \longmapsto y \in \{1, 2, \ldots, M\}$.

Here, $x = (x_1, x_2, \ldots, x_p)$ is a vector of descriptors (or features): $\forall i \in \{1, 2, \ldots, p\}, x_i \in \mathbb{X}$, with $\mathbb{X} = \{1, 2, \ldots, K\}$ (or $\mathbb{X} = \mathbb{R}$).

Goal: Learn $p(y|x)$.

A very naive method will trigger off a combinatorial explosion: $\theta \in \mathbb{R}^{K^p}$.

Bayes formula gets us:

$$p(y|x) = \frac{p(x|y)\, p(y)}{p(x)}$$

The Naive Bayes method consists in assuming that the features x_i are all conditionally independent from the class, hence:

$$p(x|y) = \prod_{i=1}^{p} p(x_i|y)$$

Then, the Bayes formula gives us:

$$p(y|x) = \frac{p(y) \prod_{i=1}^{p} p(x_i|y)}{p(x)} = \frac{p(y) \prod_{i=1}^{p} p(x_i|y)}{\sum_{y'} p(y') \prod_{i=1}^{p} p(x_i|y')}$$

We consider the case where the features take discrete values. Consequently the new graphical model contains only discrete random variables. Then, we can write a discrete model as an exponential family. Indeed we can write:

$$\log p\left(x_i = k | y = k'\right) = \delta\left(x_i = k,\, y = k'\right) \theta_{ikk'}$$

and

$$\log p\left(y = k'\right) = \delta\left(y = k'\right) \theta_{k'}$$

We can see that the dummy functions $\delta(x_i = k, y = k')$ and $\delta(y = k')$ are the *sufficient statistics* of the joint distribution model for y and the variables x_i, where $\theta_{ikk'}$ and $\theta_{k'}$ are *canonical parameters*. Thus , we can write:

$$\log p(y, x_1, \ldots, x_p) = \sum_{i,k,k'} \delta(x_i = k, y = k')\theta_{ikk'} + \sum_{k'} \delta(y = k')\theta_{k'} - A((\theta_{ikk'})_{i,k,k'}, (\theta_{k'})_{k'})$$

Where $A((\theta_{ikk'})_{i,k,k'}, (\theta_{k'})_{k'})$ is the log-partition function.

We have rewritten the joint distribution model of $(y, x_1, \ldots, x_p)$ as an exponential family. Given that the maximum of likelihood estimator of an exponential family, where the canonical parameters are not combined, is also the maximum entropy estimator; as seen in a previous course and provided that the statistical moments of the sufficient statistics equal their empirical moments.

Thus, if we introduce

$$N_{ikk'} = \#\left\{(x_i, y) = (k, k')\right\}$$
$$N = \sum_{i,k,k'} N_{ikk'},$$

The maximum likelihood estimator must satisfy the moment constraints

$$\widehat{p}\left(y = k'\right) = \frac{\sum\limits_{i,k} N_{ikk'}}{N} \qquad \text{et} \qquad \widehat{p}\left(x_i = k | y = k'\right) = \frac{N_{ikk'}}{\sum\limits_{k''} N_{ik''k'}},$$

which define them completely.

Then, we can write the estimators of the canonical parameters as:

$$\widehat{\theta}_{ikk'} = \log \widehat{p}\left(x_i = k | y = k'\right) \qquad \text{et} \qquad \widehat{\theta}_{k'} = \log \widehat{p}\left(y = k'\right).$$

However, our goal is to obtain a classification model, that is to say, a model of only the conditional probability law. From the approximated generative model and applying the Bayes rule we can get:

$$\log \widehat{p}\,(y = k'|x) = \sum_{i=1}^{p} \log \widehat{p}\,(x_i|y = k') + \log \widehat{p}\,(y = k') - \log \sum_{k'} \left(\widehat{p}\,(y = k') \prod_{i=1}^{p} \widehat{p}\,(x_i|y = k') \right)$$

We can re write the conditional model as an exponential family

$$\log p\,(y|x) = \sum_{i,k,k'} \delta(x_i = k, y = k')\theta_{ikk'} + \sum_{k'} \delta(y = k')\theta_{k'} - \log p(x)$$

Its sufficient statistics and canonical parameters are equal to those of the generative model, but seen as functions of the random variable y, given that x is fixed (we could write $\phi_{x,i,k,k'}(y) = \delta(x_i = k, y = k')$). As for the log-partition function, it is now equal to $\log p(x)$.

Warning: $\widehat{\theta}_{ikk'}$ is the maximum likelihood estimator in the generative model which, usually, is not equal to the maximum likelihood estimator in the conditional model.

B.2.2 Advantages and Drawbacks

Advantages:

- Doable in line.
- Computationally tractable solution.

Drawbacks:

- Generative: generative models produce good estimator whenever the model is "true", or in statistical words *well specified*, which means that the process that generate the real data induce a distribution equal to the one of the generative model. When the model is not *well specified* (which is the most common case) we'd better use a discriminative method.

B.2.3 Discriminative method

The problem that we have considered in the previous section is the generative model for classification in K classes. How to learn, in a discriminatory way , a classifier in K classes? Is it possible to use an exponential family?

We have already seen the logistic regression for 2 classes classification:

$$p\,(y = 1|x) = \frac{\exp\left(\omega^T x\right)}{1 + \exp\left(\omega^T x\right)}$$

Let's study the K-multiclass logistic regression:

$$
\begin{aligned}
p\left(y=k'|x\right) &= \frac{\exp\left(\sum_{i=1}^{p}\sum_{k=1}^{K}\delta\left(x_i=k\right)\theta_{ikk'}\right)}{\sum_{k''=1}^{M}\exp\left(\sum_{i=1}^{p}\sum_{k=1}^{K}\delta\left(x_i=k\right)\theta_{ikk''}\right)} \\
&= \exp\left(\sum_{i=1}^{p}\sum_{k=1}^{K}\delta\left(x_i=k\right)\theta_{ikk'} - \log\left(\sum_{k''=1}^{M}\exp\left(\sum_{i=1}^{p}\sum_{k=1}^{K}\delta\left(x_i=k\right)\theta_{ikk''}\right)\right)\right) \\
&= \exp\left(\theta_{k'}^{T}\phi\left(x\right) - \log\left(\sum_{k''=1}^{M}\exp\left(\theta_{k''}^{T}\phi\left(x\right)\right)\right)\right) \\
&= \frac{\exp\left(\theta_{k'}^{T}\phi\left(x\right)\right)}{\sum_{k''=1}^{M}\exp\left(\theta_{k''}^{T}\phi\left(x\right)\right)}
\end{aligned}
$$

Although we have built the model from different staring consideration, the resulting modelling (that is the set of possible distribution) is of the same exponential family than the Naive Bayes model.

Nonetheless, the fitted model in a discriminatory approach will be different from the one fitted in a generative approach: the fitting of the K-multiclass logistic regression results from the maximisation of the likelihood of the classes $y^{(j)}$ of a set of learning, given that $x^{(j)}$ are fixed. In other words, the fitting is obtained by computing the maximum likelihood estimator in the conditional model. Unlike what happens in the generative model, the estimator can't be obtained in a analytical form and the learning requires solving a numerical optimisation problem.