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

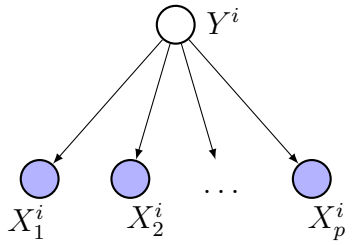
The naive Bayes classifier is relevant when modeling the joint distribution of $p(x|y)$ is too complicated. We will present it the special case where the input data is a vector of binary random variable. $X^i : \Omega \mapsto \{0,1\}^p$

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

$$p(x_1, \ldots, x_d) = \prod_{i=1}^{d} p(x_i \mid x_{\pi_i}).$$

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 $\mathbb{P}_p(X_i = .|X_{-i} = z_{\pi_i})$
end for
  return $(z_1, \ldots, z_d)$

Proposition 8.3 *The random variable* $(Z_1, \ldots, Z_d)$ *returned by the* ancestral sampling algorithm *follows exactly the distribution* $p(x_1, \ldots, x_d)$.

This result is intuitively obvious by induction, but proving it formally is somewhat tedious.

We first prove the results for a graph with two nodes.

Lemme 8.4 *If* $\tilde{X}$ *is drawn from* p_X *and given the value* $\tilde{X}$ *obtained then* $\tilde{Y}$ *is drawn from the conditional distribution* $p_{Y|X}(\cdot|\tilde{X})$ *then the pair* $(\tilde{X}, \tilde{Y})$ *follows the joint distribution* $p_{X,Y}$.

Proof This result is fairly obvious by construction. To prove it formally, we consider the case where the quantities p_X, $p_{Y|X}$ and $p_{X,Y}$ are densities with respect to the Lebesgue measure. We will prove that for any function f of the pair (x, y), we have, that for $(\tilde{X}, \tilde{Y})$ drawn sequentially as above,

$$\mathbb{E}[f(\tilde{X}, \tilde{Y})] = \int f(x, y)\, p_{X,Y}(x, y)\, dxdy,$$

as, this is true if and only if the joint distribution of $(\tilde{X}, \tilde{Y})$ is $p_{X,Y}$.

[1] Cumulative Distribution Function

Since $\tilde{Y}$ is drawn from $p_{Y|X}(\cdot|\tilde{X})$ we have that

$$\mathbb{E}[f(\tilde{X},\tilde{Y}) \mid \tilde{X}] = \int f(\tilde{X},y)\, p_{Y|X}(y|\tilde{X})dy.$$

But then

$$\begin{aligned}\mathbb{E}[f(\tilde{X},\tilde{Y})] = \mathbb{E}\big[\mathbb{E}[f(\tilde{X},\tilde{Y}) \mid \tilde{X}]\big] &= \int \Big(\int f(x,y)\, p_{Y|X}(y|x)dy\Big)p_X(x)dx, \\ &= \iint f(x,y)\, p_{Y|X}(y|x)\, p_X(x)\, dydx,\end{aligned}$$

which shows the result simply because $p_{X,Y}(x,y) = p_{Y|X}(y|x)p_X(x)$. ■

Proof of proposition 8.3. The proof we did in class is the generalization of the proof of the lemma for n variables, involving a backward recursion on $\mathbb{E}[f(X_1,\dots,X_n) \mid X_1,\dots,X_k]$. However, we can use the result of the lemma to construct another proof based on a forward recursion: we will prove by induction on k that $(Z_1,\dots,Z_k)$ has the same distribution as $(X_1,\dots,X_k)$. The induction hypothesis is true for $k=1$, since Z_1 is drawn from $p(x_1)$ and has thus the same distribution as X_1. Then, assuming the induction hypothesis is true for $k-1$ we prove it for k. First, by the induction hypothesis, we know that $(Z_1,\dots,Z_{k-1})$ follows the same distribution as $(X_1,\dots,X_{k-1})$ and Z_k is drawn from $p_{X_k|X_{\pi_k}}(\cdot,Z_{\pi_k})$. Then, we know that $p_{X_k|X_{\pi_k}}(\cdot,Z_{\pi_k}) = p_{X_k|X_{1..k-1}}(\cdot,Z_{1..k-1})$ by construction of the graphical model and the fact that in a directed graphical model a node is independent of its ancestors given its parents. If we identify Z_k with $\tilde{Y}$ and $(Z_1,\dots,Z_{k-1})$ with $\tilde{X}$ in the previous lemma, then by that lemma we know that $(Z_1,\dots,Z_k)$ follows the joint distribution $p_{Z_{1..k-1}}p_{Z_k|Z_{1..k-1}}$ with $p_{Z_{1..k-1}} = p_{X_{1..k-1}}$ and $p_{Z_k|Z_{1..k-1}} = p_{X_k|X_{1..k-1}}$, which proves that $p_{Z_{1..k}} = p_{X_{1..k}}$, so the induction hypothesis is true for k. By induction result is also true for n. ■

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 8.5 *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 to compute 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 8.6 *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 8.7 (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 8.8 (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 8.9 (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 8.10 (Perron-Frobenius) *Every stochastic matrix S has* at least *one stationary distribution.*

Definition 8.11 (Regular Markov Chain) *A Markov chain is regular (or equivalently aperiodic irreductible) if $\forall x, y \in \mathcal{X}, S(x,y) > 0$*

Proposition 8.12 *If a Markov chain is regular, 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*

$$q_n \to \pi$$

Goal We want to construct a regular transition S whose stationary distribution is

$$\pi(x) = \frac{1}{Z}\prod_{c\in\mathcal{C}} \psi_c(x_c)$$

.

Definition 8.13 (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 8.14 *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 8.15 *With that choice of $\alpha(x, z)$, if $T(\cdot, \cdot)$ is regular 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

$$\begin{cases} \alpha(x, z) \in [0, 1] \\ (*) \text{ is satisfied} \end{cases} \implies \text{detailed balance}$$

■