Lecture 9 - SAMPLING METHODS

So far, we covered:

→ exact inference on trees/chains (Belief Propagation)

→ variational inference

$$\min_{q \in Q} KL(q \,\|\, p)$$

$q \in Q$ ← variational family

p ← target: ↗ $p_\theta(x)$ ↘ $p(z|x)$ Bayesian posterior

Variational Inference:

☺: - fast, scalable optimization algorithms
- easy to assess convergence

☹: - tractable variational families may be restrictive
→ biased approximations
- often problem-specific algorithms, e.g. using conjugate priors & corresponding updates (but: recently we have "black-box" V.I. algos and Neural Nets may fix some of this)

Sampling:

- Flexible algorithms to approximate target p_θ by samples $\{x^{(1)}, \dots, x^{(m)}\}$

- Allows computing many quantities of interest by approximating expectations using Monte Carlo:

$$\mathbb{E}_{p_\theta}[f(x)] \approx \frac{1}{n}\sum_{i=1}^{n} f(x^{(i)})$$

Ex: marginals $f(x) = x_s \Rightarrow \mathbb{E}_\theta[x_s] \approx \frac{1}{n}\sum_{i=1}^{n} x_s^{(i)}$

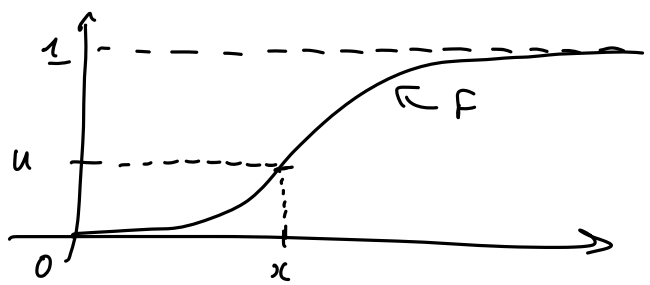
- Usually, approximation is exact / unbiased when $n \to \infty$
- But, algorithms can be more costly (than V.I.) converge slowly (i.e., need very large n)

Flexible inference: Stan (mc-stan.org) (both MCMC and V.I.)

1. Basic sampling methods

- Using the CDF (when we know it...)

Fact. Let F be the CDF of a r.v. X. If $U \sim \text{Unif}([0,1])$, then $F^{-1}(U) \sim F$



proof: $P(F^{-1}(U) \le x) = P(U \le F(x)) = F(x)$

(F^{-1} is non-decreasing) □

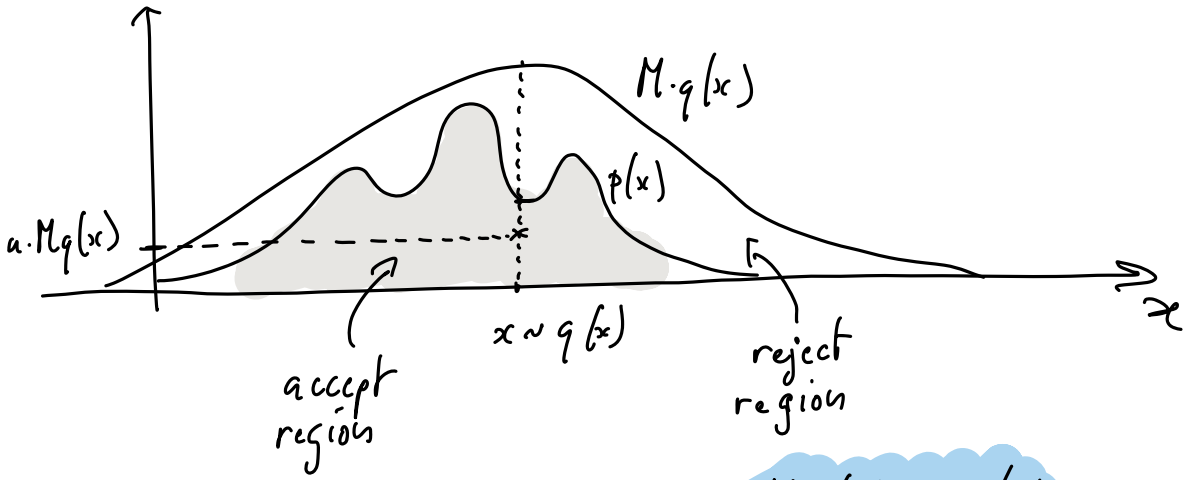
Rejection sampling

Goal: Sample from $p(x)$, known up to normalization:

$$p(x) = \frac{\tilde{p}(x)}{Z}$$

Idea: Use a proposal distribution $q(x)$ that is easy to sample from (e.g. a Gaussian)

Method:



→ Assume there is $M > 0$ s.t. $Mq(x) \geq \tilde{p}(x)$

→ . sample $x \sim q(x)$
. sample $u \sim \text{Unif}([0,1])$
. "accept" the sample x if $u \cdot Mq(x) \leq \tilde{p}(x)$

Fact: Accepted samples are distributed according to $p(x)$

proof: $\Pr(x \leq x_0 \mid x \text{ accepted}) = \dfrac{\Pr(x \leq x_0 \text{ and } x \text{ accepted})}{\Pr(x \text{ accepted})}$

$$= \frac{\mathbb{E}_{x,u}\left[\mathbb{1}\{x \le x_0\}\,\mathbb{1}\{u \le \frac{\tilde{p}(x)}{Mq(x)}\}\right]}{\mathbb{E}_{x,u}\left[\mathbb{1}\{u \le \frac{\tilde{p}(x)}{Mq(x)}\}\right]}$$

$$= \frac{\mathbb{E}_{x}\left[\mathbb{1}\{x \le x_0\}\,\frac{\tilde{p}(x)}{Mq(x)}\right]}{\mathbb{E}_{x}\left[\frac{\tilde{p}(x)}{Mq(x)}\right]}$$

$$= \frac{\int_{-\infty}^{x_0} \tilde{p}(x)\,\frac{q(x)}{q(x)}\,dx}{\int_{-\infty}^{\infty} \tilde{p}(x)\,dx} = p(x \le x_0)$$

□

Remark: We have $\Pr(\text{accept}) = \frac{C}{M}$

$\Rightarrow$ M should be as small as possible while ensuring $Mq(x) \ge \tilde{p}(x)$

■ Importance Sampling

Goal: Approximate integrals of the form

$$I = \mathbb{E}_{x\sim p}[f(x)] = \int f(x)\,p(x)\,dx$$

for a given $f(x)$.

Idea: Sample from a proposal $q(x)$ that approximates $p(x)$ well especially where $|f(x)|$ is large.

→ exploit the specific form of $f(x)$.

$$I = \int f(x) \frac{p(x)}{q(x)} q(x)\, dx \approx \frac{1}{n} \sum_{i=1}^{n} w_i f(x_i)$$

$w_i = \frac{p(x_i)}{q(x_i)}$: importance weights

$x_i \sim q$

Q: What is a good proposal ?

→ small variance

$$\mathrm{Var}_q\left(f(x) w(x)\right) = \mathbb{E}_q\left[f(x)^2 w(x)^2\right] - I^2$$

$$\geq \left(\int |f(x)| p(x)\right)^2 - I^2 \quad \text{(Jensen's)}$$

→ lower bound achieved for $|f(x)| w(x) = \text{constant}$, i.e.

$$q^*(x) \propto |f(x)|\, p(x)$$

Q: What about unnormalized $\tilde{p}(x)$?

→ The normalization constant can be estimated as

$$Z = \int \tilde{p}(x)\, dx = \int \frac{\tilde{p}(x)}{q(x)} q(x) dx \approx \frac{1}{n} \sum_{i=1}^{n} w_i$$

$$w_i = \frac{\tilde{p}(x_i)}{q(x_i)}, \quad x_i \sim q$$

$$\Rightarrow I \approx \frac{\sum_{i=1}^{n} w_i f(x_i)}{\sum_i w_i}$$

"self-normalized"

Note: $q(x)$ may also be unnormalized since Z_q cancels out!

2. Markov Chain Monte Carlo (MCMC)

Idea: To sample from $p(x)$, construct a Markov Chain such that $p(x)$ is the stationary distribution, and sample from the chain until "convergence".

Defs: • Homogeneous Markov Chain:

$$p\left(x^{(t)} \mid x^{(t-1)}, x^{(t-2)}, \cdots\right) = p\left(x^{(t)} \mid x^{(t-1)}\right) = T\left(x^{(t)} \mid x^{(t-1)}\right)$$

↑ Transition Kernel / Matrix

discrete case: $T_{ij} = T(x' = j \mid x = i)$

• Stationary distribution: $\pi(x)$ such that

$$\pi T = \pi$$ (i.e. fixed point of T)

i.e.

$$\int \pi(x)\, T(x'|x)\, dx = \pi(x')$$

$$\sum_i \pi_i T_{ij} = \pi_j$$

Note: • IF $x^{(t)} \sim p_0$, then $x^{(t)} \sim p_0 T^t$

Goal of MCMC: Construct T such that $p_0 T^t \xrightarrow[t \to \infty]{} \pi = p$

Q: When does the chain converge to π? i.e. $p_0 T^t \xrightarrow[t\to\infty]{} \pi$

Fact: If T is irreducible and aperiodic, then for any p_0 we have $p_0 T^t \to \pi$

Note: Convergence speed depends on eigenvalues of T.

Q: How can we ensure that π is stationary?

Fact: (detailed balance condition / reversibility)
A sufficient (but not necessary) condition for π to be stationary for T is: for any x, x'

$$\pi(x)\, T(x'|x) = \pi(x')\, T(x|x')$$

proof: indeed,

$$\begin{aligned}\pi T(x') &= \int \pi(x)\, T(x'|x)\, dx \\ &= \int \pi(x')\, T(x|x')\, dx \\ &= \pi(x') \int T(x|x')\, dx = \pi(x')\end{aligned}$$

□

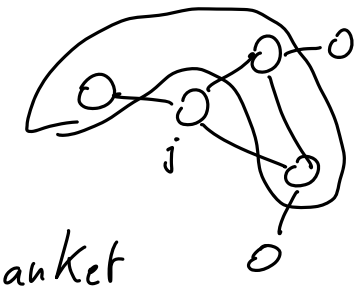
→ This gives a convenient way to construct T from some p that we want to sample from, s.t. p is stationary

Gibbs Sampling

Motivation: • Often, sampling from the target $p(x)$ is difficult, but sampling from conditionals $p(x_j | x_{-j})$ is easy

• E.g. in Markov Random Fields

$$p(x_j | x_{-j}) = p(x_j | x_{\mathrm{MB}(j)})$$



Markov Blanket

Algorithm: From a sample $x^{(t)}$, pick index j (random or sequential)

$$\begin{cases} x_j^{(t+1)} \sim p(x_j^{(t+1)} | x_{-j}^{(t)}) \\ x_i^{(t+1)} = x_i^{(t)} \quad \text{for } i \neq j \end{cases}$$

Fact: Gibbs Sampling satisfies detailed balance

Proof:

$$\begin{aligned} p(x^{(t)}) T(x^{(t+1)} | x^{(t)}) &= p(x^{(t)}) \, p(x_j^{(t+1)} | x_{-j}^{(t)}) \\ &= p(x_{-j}^{(t)}) \, p(x_j^{(t)} | x_{-j}^{(t)}) \, p(x_j^{(t+1)} | x_{-j}^{(t)}) \\ &= p(x_{-j}^{(t+1)}) \, p(x_j^{(t)} | x_{-j}^{(t+1)}) \, p(x_j^{(t+1)} | x_{-j}^{(t+1)}) \\ &= p(x^{(t+1)}) \, p(x_j^{(t)} | x_{-j}^{(t+1)}) = p(x^{(t+1)}) T(x^{(t)} | x^{(t+1)}) \end{aligned}$$

□

Example: Ising model $x_i \in \{0,1\}$

$$p_\theta(x) \propto \exp\left\{ \sum_{i \in V} \theta_i x_i + \sum_{i \sim j} \theta_{ij} x_i x_j \right\}$$

→ conditionals

$$p_\theta(x_i \mid x_{-i}) \propto \exp\left\{ \theta_i x_i + \sum_{i \sim j} \theta_{ij} x_i x_j \right\}$$

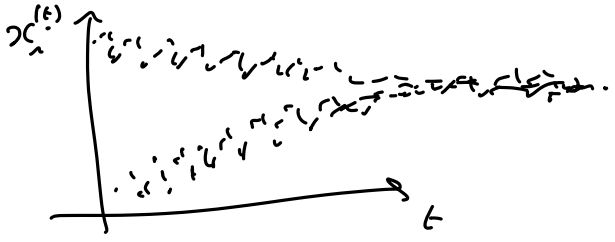
i.e. $$p_\theta(x_i = 1 \mid x_{-i}) = \sigma\left(\theta_i + \sum_{i \sim j} \theta_{ij} x_j \right)$$

[sigmoid $\sigma(z) = (1 + \exp(-z))^{-1}$

Remark: Very similar form to MeanField V.I. !!

Notes: In practice:

- ignore initial samples ("burn-in" phase)
- run multiple chains in parallel, with very ≠ initializations
- convergence diagnostics: compare values of some variables across the different chains



- "Collapsed" Gibbs / "Rao-Blackwellization": marginalize out certain variables if possible for faster convergence (e.g. parameters in Bayesian mixture)

Metropolis-Hastings algorithm

$p(x)$: target distribution

Method: From current sample $x^{(t)}$:

- sample $\tilde{x} \sim q(\cdot | x^{(t)})$ from a proposal $q(x'|x)$
- set $x^{(t+1)} = \tilde{x}$ with acceptance probability

$$\alpha(x^{(t)}, \tilde{x}) := \min\left(1, \frac{p(\tilde{x})\, q(x^{(t)}|\tilde{x})}{p(x^{(t)})\, q(\tilde{x}|x^{(t)})}\right)$$

otherwise, set $x^{(t+1)} = x^{(t)}$

Remarks:

- The proposal can be arbitrary! But it should be carefully designed for fast convergence.
- $p(x)$ can be unnormalized! (Z cancels out)
- If q is symmetric ($q(x'|x) = q(x|x')$) then

$$\alpha(x, x') = \min\left(1, \frac{p(x')}{p(x)}\right)$$

$\Rightarrow$ encourage moves towards higher $p(x')$

- The transition kernel in discrete case is:

$$T_{ij} = \begin{cases} q(j|i)\,\alpha(i,j) & \text{if } j \neq i \\ q(i|i) + \sum_{j' \neq i} q(j'|i)(1 - \alpha(i,j')) & \text{o/w} \end{cases}$$

Fact: M-H satisfies detailed balance

proof: consider states $x = i$ and $x' = j$

Assume w.l.o.g. that $p(i)\,q(j|i) > p(j)\,q(i|j)$

- We have $\alpha(i,j) < 1$ and $\alpha(j,i) = 1$
- $T_{ij} = q(j|i) \cdot \alpha(i,j) = \dfrac{p(j)\,q(i|j)}{p(i)}$

$$p(i)\,T_{ij} = p(j)\,q(i|j)$$

- Note also that $T_{ji} = q(i|j) \cdot \alpha(j,i) = q(i|j)$

$$\Rightarrow \quad p(i)\,T_{ij} = p(j)\,T_{ji}$$

□