Data-driven Modeling - Machine Learning

TECHNISCHE UNIVERSITÄT DARMSTADT

Instructor: Prof Heinz Koeppl

Sampling methods

SAMPLING METHODS

Bayesian inference / probabilistic ML

TECHNISCHE UNIVERSITÄT DARMSTADT

The *posterior probability* of the parameters given the data

$$P(\theta \mid x_1, \ldots, x_n) = \frac{f(x_1, \ldots, x_n \mid \theta) P(\theta)}{P(x_1, \ldots, x_n)}$$

in general has a complex functional form and will not be any of the known probability distributions (e.g. Gaussian).

- The posterior mean $\mathsf{E}[\theta \mid x_1, \ldots, x_n]$ as a good point estimate for θ requires the solution of a high-dimensional (intractable) integral.
- If we need to quantify our uncertainty over θ we need to capture the broadness of the distribution (compute variance).
- Often in ML and statistics: Functional form is known, computing integrals is infeasible and sampling directly is infeasible (because non-canonical form).

MOTIVATION

Sampling for Monte Carlo integration

TECHNISCHE UNIVERSITÄT DARMSTADT

For a random variable $X \sim F_X$ with density $f_X(x)$ and an observable $h(X)$, we want to evaluate

$$\mathrm{E}[h(X)] = \int h(x) f(x)\, dx\,. \qquad \text{(Can also be a large sum)}$$

Problem: Often the integral cannot be evaluated analytically.

Idea: Monte Carlo integration

- draw a set of n samples $X_1, \ldots, X_n$ taken i.i.d. from F
- define estimator

$$I_n = \frac{1}{n} \sum_{i=1}^{n} h(X_i) \quad \overset{\text{WLLN}}{\Rightarrow} \quad I_n \to \mathrm{E}[h(X)]$$

- error of the estimate independent of the dimension of X

$$\sqrt{\mathrm{E}[(I_n - \mathrm{E}[h(X)])^2]} = O\left(\frac{1}{\sqrt{n}}\right)$$

MOTIVATION

Estimating probabilities

TECHNISCHE UNIVERSITÄT DARMSTADT

Monte Carlo integration allows estimating probabilities of the form $P(X \in A)$:

- define indicator function for set A as

$$\mathbf{1}_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}$$

- then $P(X \in A)$ can be estimated by MC integration since

$$P(X \in A) = \mathrm{E}[h(X)] \quad \text{with} \quad h(x) = \mathbf{1}_A(x)$$

- powerful numerical method for problems in statistics and machine learning

Problem: How to draw samples from distributions F on a computer?

SAMPLING

Uniform samples

TECHNISCHE UNIVERSITÄT DARMSTADT

Generating *pseudo-random integers*

- computers can only produce deterministic sequences
- construct a sequence such that it *looks random*
- example: linear congruential generator

$$x_{i+1} = (a x_i + c) \bmod m \quad \text{with} \quad a, m, c \in \mathbb{N}$$

- E.g. Lehmer's method: Set $c = 0$ and m a Mersenne prime number, i.e., all integers of the form $2^k - 1$ with k prime *(see also Mersenne Twister)*

Sampling from $\mathcal{U}(0,1)$

- integer generators have a period m
- rescale output

$$y_i = \frac{x_i}{m} \quad \rightarrow \quad y_i \in [0,1)$$

- for a good integer generator with large period, the y_i approximate $\mathcal{U}(0,1)$

SAMPLING

Transformation of continuous random variables

TECHNISCHE UNIVERSITÄT DARMSTADT

Theorem: Let X be a continuous RV with density f_X. For a one-on-one function g with inverse g^{-1}, define $Y = g(X)$. Then

$$f_Y(y) = f_X(g^{-1}(y)) \left| \frac{d}{dy} g^{-1}(y) \right| .$$

Proof: For g increasing

$$F_Y(y) = P(Y \leq y) = P(g(X) \leq y) = P(X \leq g^{-1}(y)) = F_X(g^{-1}(y))$$

$$\Rightarrow f_Y(y) = \frac{d}{dy} F_Y(y) = f_X(g^{-1}(y)) \frac{d}{dy} g^{-1}(y)$$

For g decreasing

$$F_Y(y) = P(Y \leq y) = P(g(X) \leq y) = P(X \geq g^{-1}(y)) = 1 - F_X(g^{-1}(y))$$

$$\Rightarrow f_Y(y) = \frac{d}{dy} F_Y(y) = -f_X(g^{-1}(y)) \frac{d}{dy} g^{-1}(y)$$

SAMPLING

Inverse transform method

TECHNISCHE UNIVERSITÄT DARMSTADT

Goal: We want to sample a continuous RV $X \sim F_X$ with density f_X.

Idea: Find $g : [0,1] \to \mathbb{R}$ such that $X = g(U)$ with $U \sim \mathcal{U}(0,1)$.

How to choose g? Observe the transformation law for this special case read

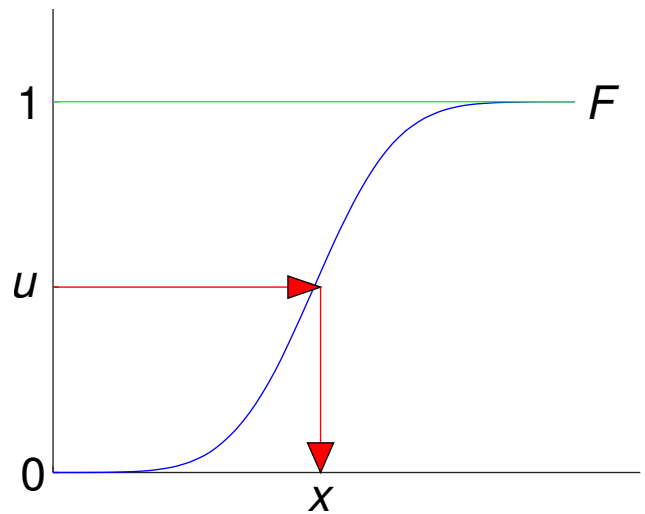
$$f_X(x) = f_U(g^{-1}(x)) \frac{d}{dx} g^{-1}(x) \ .$$

Since $f_U(u) = 1$ for any $u \in [0,1]$

$$f_X(x) = \frac{d}{dx} g^{-1}(x)$$

Integrating on both sides yields

$$g(x) = F_X^{-1}(x)$$



SAMPLING

Sampling from an exponential distribution

TECHNISCHE UNIVERSITÄT DARMSTADT

Example: Let $X \sim \mathrm{Exp}(\lambda)$ for $\lambda > 0$.

- The PDF $f_X(x)$ is for $x \geq 0$ is

$$f_X(x) = \lambda \exp(-\lambda x)\,.$$

- The CDF $F_X(x)$ for $x \geq 0$ is

$$F_X(x) = \int_0^x f_X(x')\,dx' = 1 - \exp(-\lambda x)\,.$$

- With $X = g(U) = F_X^{-1}(U)$ and $U \sim \mathcal{U}(0,1)$ find X s.t

$$1 - \exp(-\lambda X) \stackrel{!}{=} U\,.$$

- Analytic inversion and using $1 - U \sim \mathcal{U}(0,1)$ yields

$$X = -\frac{\log U}{\lambda}\,.$$

SAMPLING

Inverse transform method for discrete RVs

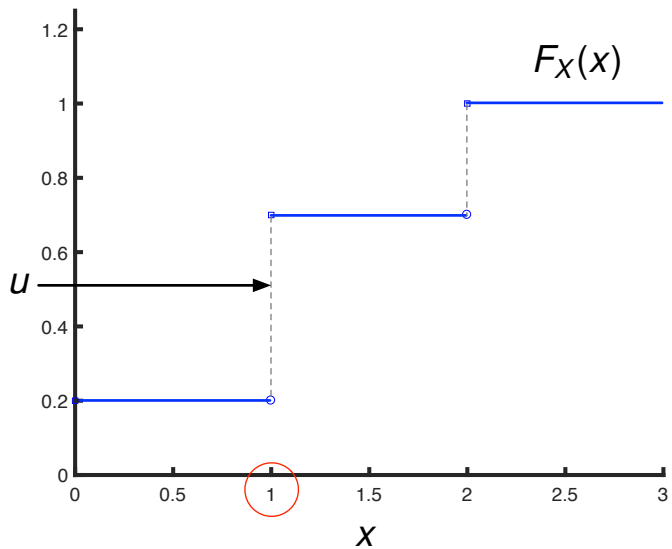
TECHNISCHE UNIVERSITÄT DARMSTADT

Consider the RV $X \in \{0, 1, 2\}$ with PMF $f_X(x)$ and CDF $F_X(x)$ given by

$$f_X(x) = \begin{cases} 0.2 & \text{for } x = 0 \\ 0.5 & \text{for } x = 1 \\ 0.3 & \text{for } x = 2 \end{cases} \qquad F_X(x) = \begin{cases} 0 & \text{for } x < 0 \\ 0.2 & \text{for } 0 < x \leq 1 \\ 0.7 & \text{for } 1 < x \leq 2 \\ 1.0 & \text{for } x \geq 2 \end{cases}$$

Idea:

- throw random "darts" with $Y \sim \mathcal{U}(0, 1)$
- draw a horizontal line
- choose X corresponding to intersection



SAMPLING

Limits of inversion sampling

TECHNISCHE UNIVERSITÄT DARMSTADT

For a continuous function X with density f_X the inversion method requires

- that the the density f_X can be integrated analytically,
- the resulting CDF F_X can be inverted efficiently.

Example: Let $X \sim \mathcal{N}(0,1)$. The PDF and CDF are

$$f_X(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}x^2\right), \quad F_X(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp\left(-\frac{1}{2}x'^2\right) dx'$$

- integral can only be evaluated numerically
- inversion only possible by iterative algorithm
- this would be terribly slow

TRANSFORMATION OF VARIABLES

Multivariate extension - Normalizing flows

TECHNISCHE UNIVERSITÄT DARMSTADT

With the rise of deep neural networks the multivariate extension of the simple *transformation of variable method*, gained importance.
Let $X = (X_1, \ldots, X_n)$ and $Y = (Y_1, \ldots, Y_n)$ be two $\mathbb{R}^n$-valued RVs with densities f_X and f_Y. Let $g : \mathbb{R}^n \to \mathbb{R}^n$ be a one-to-one (invertible) function with $Y = g(X)$ and $X = g^{-1}(Y)$, then

$$f_Y(y) = f_X(g^{-1}(y)) \left| \det \left(\frac{\partial g^{-1}(y)}{\partial y} \right) \right|$$

where $\partial g^{-1}(y)/\partial y$ is the Jacobian matrix of the function g^{-1}.
Normalizing flows in a nutshell: Define

$$Y = g_\theta^m \circ \cdots \circ g_\theta^1(X) = g_\theta^m(g_\theta^{m-1}(\cdots g_\theta^1(X)))$$

where $g^1, \ldots, g^m$ represent layers of a neural network with weight parameters θ. Start with a simple distribution for X (e.g. Gaussian) and learn the transformation to much target distribution f_Y of data (see later).

SAMPLING

Rejection sampling

TECHNISCHE UNIVERSITÄT DARMSTADT

Let $X \sim F_X$ be an RV with density f_X. Often, it is hard to compute the inverse $F_X^{-1}(y)$.

Idea: Acceptance-rejection method

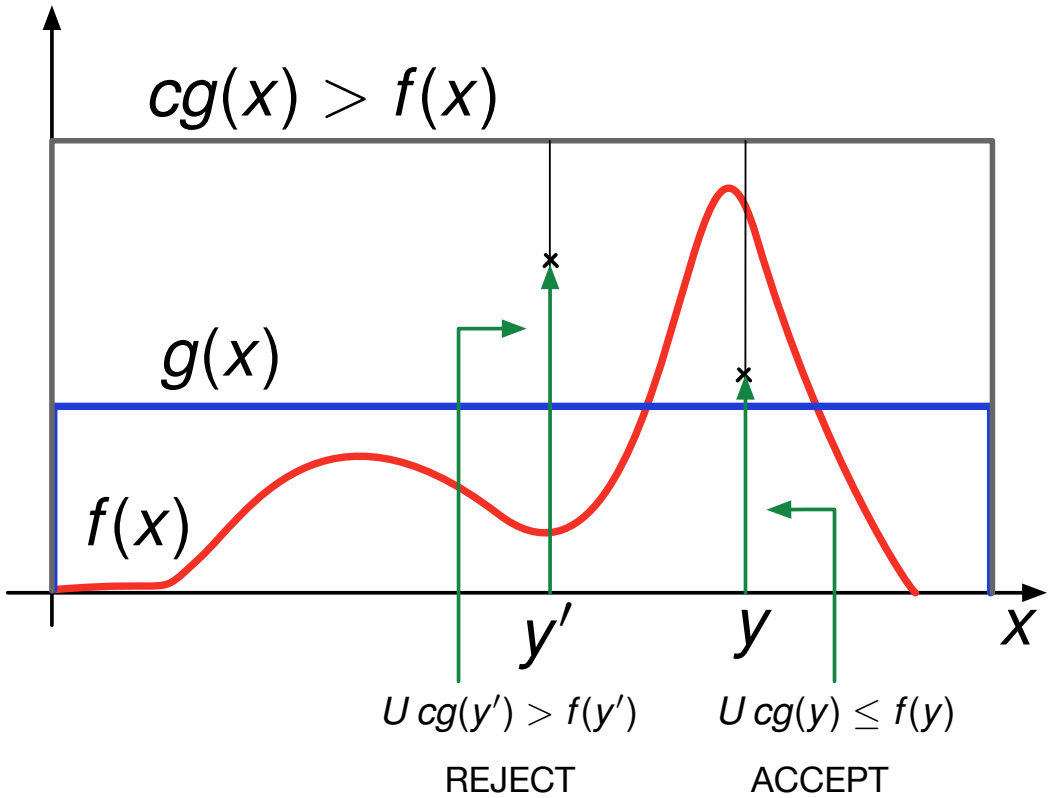
- sample from a simpler proposal distribution with density g_X.
- accept or reject samples, such that the remaining samples have distribution F_X

The exact algorithm proceeds as follows: Let g_X be a density and $c > 0$ constant such that $f_X(x) < c g_X(x)$ for all x

1. Generate $Y \sim G_X$.
2. Generate $U \sim \mathcal{U}(0,1)$.
3. If
$$U \leq \frac{f_X(Y)}{c g_X(Y)} \quad \text{or} \quad U c\, g_X(Y) \leq f_X(Y)$$
set $X = Y$ (accept proposal), otherwise go back to step 1 (reject proposal)

SAMPLING

Rejection sampling: Illustration with a uniform density



- Does work for any g_X as long as there exists a constant $c > 0$ such that $cg_X(x) > f_X(x)$ for all x ($cg_X(x)$ dominates $f_X(x)$ everywhere).

SAMPLING

Importance sampling

TECHNISCHE UNIVERSITÄT DARMSTADT

Recall the Monte Carlo integration setting: For $X \sim F_X$ with density f_X we want to evaluate

$$\mathrm{E}[h(X)] = \int h(x) f_X(x)\, dx\,.$$

What if we cannot sample from F?

Idea: importance sampling

- take a second distribution G_X with density g_X and note that

$$\mathrm{E}[h(X)] = \int h(x) \frac{f_X(x)}{g_X(x)} g_X(x)\, dx$$

- draw samples $X_1, \ldots X_n$ from G_X
- define the new estimator

$$I_n = \frac{1}{n} \sum_{i=1}^{n} w_i h(X_i) \quad \text{with importance weights } w_i = \frac{f_X(X_i)}{g_X(X_i)}$$

MARKOV CHAIN MONTE CARLO

Markov chains

TECHNISCHE UNIVERSITÄT DARMSTADT

Definition: A (discrete-time) *Markov chain* is a indexed family of RVs $(X_0, X_1, \ldots)$ taking values in a countable set $\mathcal{X}$ with joint probability

$$P(X_0, \ldots, X_N) = P(X_0) \prod_{n=1}^{N} P(X_n \mid X_{n-1}, \ldots, X_0)$$

that satisfies

$$P(X_n = v \mid X_{n-1} = u, X_{n-2} = w, \ldots, X_0 = z) = P(X_n = v \mid X_{n-1} = u)$$

for all $u, w, \ldots, z \in \mathcal{X}$ and $n \in \mathbb{N}_0$.

Definition: If the transition probability does not change with time, i.e.

$$P(X_{n+1} = j \mid X_n = i) = p_{ij} \quad \text{for all } n\,,$$

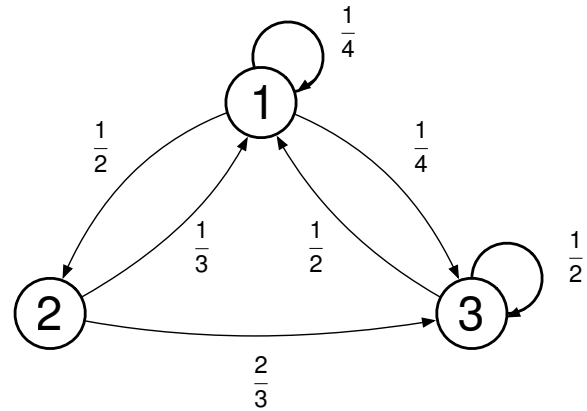
then the Markov chain is called *homogeneous*. The matrix $\mathbf{P}$ whose element (i, j) is p_{ij} is called the *transition matrix*.

MARKOV CHAIN MONTE CARLO

Markov chains continued

TECHNISCHE UNIVERSITÄT DARMSTADT

Example: Let $\mathcal{X} = \{1, 2, 3\}$ with the transition probabilities as in the figure.



$$\Rightarrow \mathbf{P} = \begin{pmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{3} & 0 & \frac{2}{3} \\ \frac{1}{2} & 0 & \frac{1}{2} \end{pmatrix}$$

Denoting by $p_j(n) = P(X_n = j)$ the j-th component of vector $\mathbf{p}(n)$, the probability over states at time n can then be written in recursive form as

$$\mathbf{p}(n) = \mathbf{p}(n-1)\mathbf{P} \quad \text{with} \quad \mathbf{p}(0) = \mathbf{q}$$

where $\mathbf{q}$ is the initial probability vector with components $q_j = P(X_0 = j)$. Hence,

$$\mathbf{p}(n) = \mathbf{q}\mathbf{P}^n.$$

MARKOV CHAIN MONTE CARLO

Stationarity and detailed balance

TECHNISCHE UNIVERSITÄT DARMSTADT

Definition: We say that π is a *stationary distribution* if $\pi = \pi\mathbf{P}$.

Defintion: A distribution π satisfies detailed balance, if $\pi_i p_{ij} = p_{ji}\pi_j$ for all i, j.

- Detailed balance means that transition $i \to j$ is as likely as the inverse $j \to i$.

Theorem: If π satisfies detailed balance, then π is a stationary distribution (proof at home!).

In uncountable spaces $\mathcal{X}$ (e.g. $\mathbb{R}^n$):

- distribution π $\quad\to\quad$ density $f(x)$
- transition element p_{ij} $\quad\to\quad$ transition kernel density $p(x' \mid x)$
- detailed balance: $\quad p(x' \mid x)f(x) = p(x \mid x')f(x')$

MARKOV CHAIN MONTE CARLO

Metropolis-Hasting (MH): Derivation

TECHNISCHE UNIVERSITÄT DARMSTADT

We want to generate samples from a distribution F with density f. You need to be able to evaluate f for any x up to the normalizer.

Idea: construct a Markov chain with F as a stationary distribution

- key step 1: split transition kernel of the Markov chain into a *proposal* $q(x' \mid x)$ and an *acceptance probability* $\alpha(x' \mid x)$

$$p(x' \mid x) = q(x' \mid x)\alpha(x' \mid x)$$

- key step 2: design α such that detailed balance holds

$$\underbrace{q(x' \mid x)\alpha(x' \mid x)}_{p(x'|x)} f(x) = \underbrace{q(x \mid x')\alpha(x \mid x')}_{p(x|x')} f(x') \rightarrow \frac{\alpha(x' \mid x)}{\alpha(x \mid x')} = \frac{q(x \mid x')f(x')}{q(x' \mid x)f(x)}$$

- simplest choice satisfying this condition: $\alpha(x' \mid x) = \min\left\{1, \frac{q(x|x')f(x')}{q(x'|x)f(x)}\right\}$

MARKOV CHAIN MONTE CARLO

Intuition

TECHNISCHE UNIVERSITÄT DARMSTADT

If f does not satisfy detailed balance wrt q, for instance,

$$q(x' \mid x)f(x) > q(x \mid x')f(x')$$

then we can choose $\alpha(x' \mid x) < 1$ and $\alpha(x \mid x') = 1$ in the modified chain

$$q(x' \mid x)\alpha(x' \mid x)f(x) = q(x \mid x')\alpha(x \mid x')f(x')$$

Hence

$$\alpha(x' \mid x) = \frac{q(x \mid x')f(x')}{q(x' \mid x)f(x)}$$

However, for that choice, $\alpha(x' \mid x) < 1$ implies $\alpha(x \mid x') > 1$. Hence,

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

MARKOV CHAIN MONTE CARLO

Metropolis-Hasting: Algorithm

TECHNISCHE UNIVERSITÄT DARMSTADT

Data : $X_0 = x_0$
for i from 0 to $n-1$ **do**
  generate
    $x' \sim q(x' \mid X_i = x_i)$
  evaluate accept. prob.
  $\alpha(x' \mid x) =$
    $\min\left\{1, \frac{q(x|x')f(x')}{q(x'|x)f(x)}\right\}$
  accept with prob. α:
  generate $U \sim \mathcal{U}(0,1)$
  if $U < \alpha(x' \mid x)$ **then**
    $x_{i+1} = x'$
  else
    $x_{i+1} = x_i$
  end
end
return $(x_0, \ldots, x_n)$

Remarks:

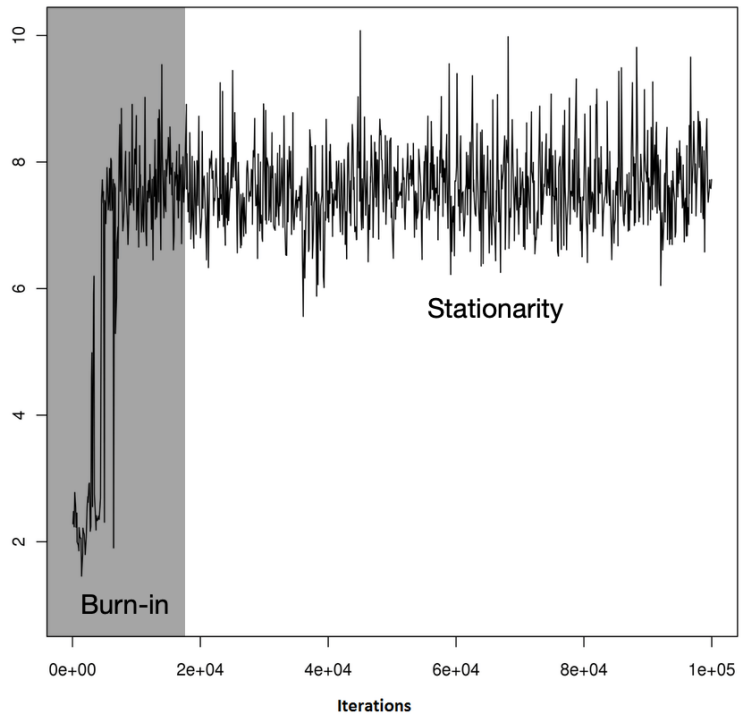
- Any constant c s.t. $f = cf^*$ cancels out in the acceptance ratio.
- Hence, MH-sampling can be done when the normalization constant is unknown.
- If the proposal is symmetric in x, x' the acceptance ratio reduces to $\alpha(x' \mid x) = \min\{1, \frac{f(x')}{f(x)}\}$
- This is known as random walk MH or classical Metropolis algorithm.
- A popular choice for the proposal distribution is a (multivariate) Gaussian $x' \sim \mathcal{N}(x, \Sigma)$.

MARKOV CHAIN MONTE CARLO

Practical Remarks

TECHNISCHE UNIVERSITÄT DARMSTADT

- Only samples at stationarity are samples from the target distribution F. The transient / non-stationary part is called “burn-in” and needs to be discarded.



- Low acceptance probability ($< 10\%$) indicates bad mixing / exploration and high autocorrelation.

MARKOV CHAIN MONTE CARLO

Monte-Carlo estimators

TECHNISCHE UNIVERSITÄT DARMSTADT

- But samples of a Markov chain are not i.i.d. samples. The weak law of large numbers (WLLN) does not apply.
- Let $X_1, \ldots, X_n$ be dependent samples from an *ergodic* random process with stationary distribution F, then the Monte Carlo estimator

$$I_n = \frac{1}{n} \sum_{i=1}^{n} h(X_i) \to \mathrm{E}[h(X)] = \int h(x) f(x) dx$$

  for $n \to \infty$ (Birkhoff's ergodic theorem).
- The Markov chain used for MCMC is a ergodic process (irreducible and aperiodic).
- However, generally, estimators from dependent samples have a higher uncertainty than estimators from i.i.d. samples. Effective sample size (ESS)

$$\hat{n} = \frac{n}{1 + 2\sum_{k=1}^{\infty} \rho(k)} \quad \text{with autocorrelation} \quad \rho(k) = \frac{\mathrm{Cov}(X_i, X_{i+k})}{\mathrm{Var}(X_i)}$$

MARKOV CHAIN MONTE CARLO

Gibbs sampling

TECHNISCHE UNIVERSITÄT DARMSTADT

Suppose we have continuous RVs $X_1, \ldots, X_n$ with joint density $f(x_1, \ldots, x_n)$. The *full conditional densities* of each variable given the other variables is

$$f(x_i \mid x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n), \quad \text{for } i = 1, \ldots, n.$$

If one can sample directly from all conditionals: Gibbs sampling

- Iterate over the coordinates deterministically from 1 to n.
  For step $i+1$ use the updated values from steps $1, \ldots, i$.
- At each step, pick a coordinate i from $1, \ldots, n$ and update coordinate i.
- Forms a Markov chain with stationary distribution $f(x_1, \ldots, x_n)$.

Observation: Use full conditionals as a proposal in an MH-sampler

- It can be shown that the acceptance probability $\alpha(x' \mid x) = 1$.

MARKOV CHAIN MONTE CARLO

Acceptance rate of the Gibbs update

TECHNISCHE UNIVERSITÄT DARMSTADT

Suppose we have continuous RVs $X_1, \ldots, X_n$ with joint density $f(x_1, \ldots, x_n)$. The *full conditional densities* of each variable given the other variables is

$$f(x_i \mid x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n), \quad \text{for } i = 1, \ldots, n.$$

Let $q(x' \mid x) = f(x'_i \mid x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$. The acceptance ratio becomes

$$\frac{q(x \mid x')}{q(x' \mid x)} \frac{f(x')}{f(x)} = \frac{f(x_i \mid x'_1, \ldots, x'_{i-1}, x'_{i+1}, \ldots, x'_n)}{f(x'_i \mid x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)} \frac{f(x_1, \ldots, x_{i-1}, x'_i, x_{i+1}, \ldots, x_n)}{f(x'_1, \ldots, x'_{i-1}, x_i, x'_{i+1}, \ldots, x'_n)}$$

The numerator and denominator of the right-hand term can be written as

$$f(x_1, \ldots, x'_i, \ldots, x_n) = f(x'_i \mid x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) f(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$$
$$f(x'_1, \ldots, x_i, \ldots, x'_n) = f(x_i \mid x'_1, \ldots, x'_{i-1}, x'_{i+1}, \ldots, x'_n) f(x'_1, \ldots, x'_{i-1}, x'_{i+1}, \ldots, x'_n)$$

Inserting into the acceptance ratio shows that $\alpha(x' \mid x) = 1$.

MARKOV CHAIN MONTE CARLO

Gibbs sampling illustration

TECHNISCHE UNIVERSITÄT DARMSTADT

- Two dimensional distribution $f(x, y)$ (here a zero mean correlated Gaussian).
- Iteratively sample from $x^{i+1} \sim f(x \mid Y = y^i)$ and $y^{i+1} \sim f(y \mid X = x^{i+1})$
- One component update
- Still a Markov chain and hence requires “burn-in”.

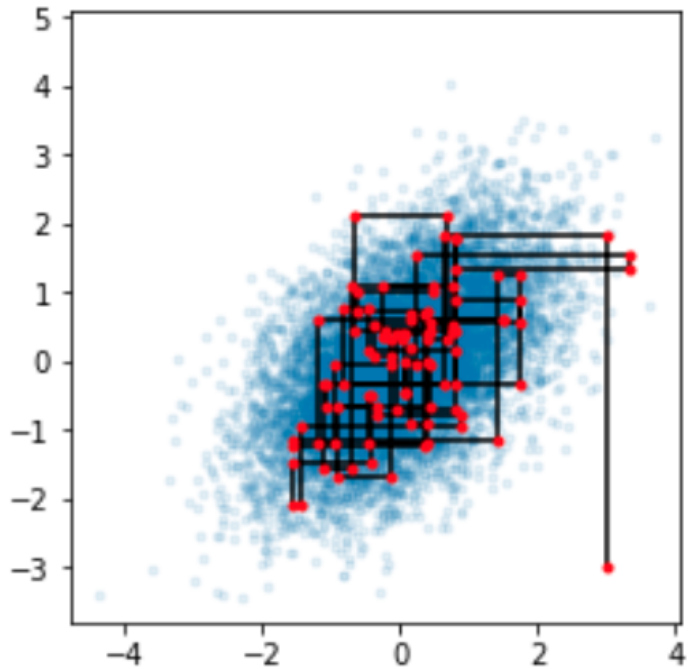


ILLUSTRATION FOR BAYESIAN INFERENCE

Sampling from the parameter posterior

TECHNISCHE UNIVERSITÄT DARMSTADT

We have a machine learning model with parameters $\theta \in \mathbb{R}^n$ and collected training data $X = (x_1, \ldots, x_N)$ with e.g. $x_i \in \mathbb{R}^m$. Often we are able to compute the posterior $P(\theta \mid X)$ or at least an unnormalized version thereof:

$$P(\theta \mid X) = \frac{f(X \mid \theta)P(\theta)}{P(X)} = \frac{f(X \mid \theta)P(\theta)}{\int f(X \mid \theta)P(\theta)\mathrm{d}\theta} \propto f(X \mid \theta)P(\theta)$$

Hence, we can evaluate $P(\theta \mid X)$ (or at least $f(X \mid \theta)P(\theta)$) but cannot sample from $P(\theta \mid X)$ because it is not in a canonical form.

Construct a (easy to sample from) proposal distribution $q(\theta' \mid \theta)$, then accept a new parameter point θ' with probability

$$\alpha(\theta' \mid \theta) = \min\left\{1, \frac{P(\theta' \mid X)q(\theta \mid \theta')}{P(\theta \mid X)q(\theta' \mid \theta)}\right\} = \min\left\{1, \frac{P(X \mid \theta')P(\theta')q(\theta \mid \theta')}{P(X \mid \theta)P(\theta)q(\theta' \mid \theta)}\right\}$$

Proposal $\theta' \sim q(\theta' \mid \theta)$ according to random walk $\theta' = \theta + \eta$ with e.g. $\eta \sim \mathcal{N}(0, \sigma^2 I)$

ILLUSTRATION FOR BAYESIAN INFERENCE

Sampling from the parameter posterior

TECHNISCHE UNIVERSITÄT DARMSTADT

For symmetric random walk proposals we have $q(\theta' \mid \theta) = q(\theta \mid \theta')$.

Interpretation of MCMC update: If new θ' is such that $P(\theta' \mid X) > P(\theta \mid X)$ then we move into higher posterior region always ($\alpha = 1$). In the other case, we move into low posterior region with some probability $\alpha < 1$.

Gibbs sampler: denote by $\theta_{\neg i}$ all components of the parameter vector except the i-the component θ_i. By conditional probability we have

$$P(\theta \mid X) = P(\theta_i, \theta_{\neg i} \mid X) = P(\theta_i \mid X, \theta_{\neg i}) P(\theta_{\neg i} \mid X)$$

Then, with proposal $q(\theta' \mid \theta) = P(\theta_i' \mid X, \theta_{\neg i})$, the new candidate is $\theta' = (\theta_i', \theta_{\neg i})$

$$\begin{aligned}\alpha(\theta' \mid \theta) &= \min\left\{1, \frac{P(\theta' \mid X)\, P(\theta_i \mid X, \theta'_{\neg i})}{P(\theta \mid X)\, P(\theta_i' \mid X, \theta_{\neg i})}\right\} \\ &= \min\left\{1, \frac{P(\theta_i' \mid X, \theta'_{\neg i}) P(\theta'_{\neg i} \mid X)\, P(\theta_i \mid X, \theta'_{\neg i})}{P(\theta_i \mid X, \theta_{\neg i}) P(\theta_{\neg i} \mid X), P(\theta_i' \mid X, \theta_{\neg i})}\right\} = 1,\end{aligned}$$

where we exploited that for a Gibbs iteration $\theta'_{\neg i} = \theta_{\neg i}$ holds.