Chapter 17

Monte Carlo Methods

Monte Carlo Sampling

- To approximate sums or integrals (which are costly to evaluate or intractable) by drawing samples

$$s = \sum_{\boldsymbol{x}} p(\boldsymbol{x}) f(\boldsymbol{x}) \text{ or } s = \int p(\boldsymbol{x}) f(\boldsymbol{x}) d\boldsymbol{x}$$

- **Idea:** To view the sum/integral as an expectation under some distribution and to approximate it by an *average*

$$s = E_p[f(\boldsymbol{x})] \approx \hat{s}_n = \frac{1}{n} \sum_{i=1}^{n} f(\boldsymbol{x}^{(i)})$$

where

$$\boldsymbol{x}^{(i)} \sim p(\boldsymbol{x})$$

- It is easy to verify that the estimator $\hat{s}_n$ is unbiased

$$E[\hat{s}_n] = E_p[f(\boldsymbol{x})] = s$$

- If the samples $\boldsymbol{x}^{(i)}$ are independently and identically distributed (i.i.d.),

$$\begin{aligned} \mathsf{Var}[\hat{s}_n] &= \frac{\mathsf{Var}[f(\boldsymbol{x})]}{n} \\ \hat{s}_n &\sim \mathcal{N}(s, \mathsf{Var}[\hat{s}_n]) \quad \text{(C.L.T.)} \end{aligned}$$

Importance Sampling

- To approximate the expectation based on a proposal distribution $q(\boldsymbol{x})$ that is easier to draw samples from than $p(\boldsymbol{x})$

$$s = \sum_{\boldsymbol{x}} p(\boldsymbol{x}) f(\boldsymbol{x}) = \sum_{\boldsymbol{x}} q(\boldsymbol{x}) \frac{p(\boldsymbol{x}) f(\boldsymbol{x})}{q(\boldsymbol{x})}$$

- Importance sampling estimator $\hat{s}_q$

$$\hat{s}_q = \frac{1}{n} \sum_{i=1}^{n} \frac{p(\boldsymbol{x}^{(i)}) f(\boldsymbol{x}^{(i)})}{q(\boldsymbol{x}^{(i)})} = \frac{1}{n} \sum_{i=1}^{n} \frac{p(\boldsymbol{x}^{(i)})}{q(\boldsymbol{x}^{(i)})} f(\boldsymbol{x}^{(i)})$$

where

$$\boldsymbol{x}^{(i)} \sim q(\boldsymbol{x})$$

- $p(\boldsymbol{x}^{(i)})/q(\boldsymbol{x}^{(i)})$ are known as *importance weights*

- It is readily seen that $\hat{s}_q$ is unbiased irrespective of the choice of $q(\boldsymbol{x})$

$$E_q[\hat{s}_q] = E_q[\frac{p(\boldsymbol{x})f(\boldsymbol{x})}{q(\boldsymbol{x})}] = E_p[f(\boldsymbol{x})] = s$$

- The variance of $\hat{s}_q$ is however highly sensitive to the choice of $q(\boldsymbol{x})$

$$\mathsf{Var}[\hat{s}_q] = \mathsf{Var}[\frac{p(\boldsymbol{x})f(\boldsymbol{x})}{q(\boldsymbol{x})}]/n$$

Biased Importance Sampling

- Oftentimes $p(\boldsymbol{x})$ can only be evaluated up to a normalization constant

$$p(\boldsymbol{x}) = \frac{\tilde{p}(\boldsymbol{x})}{Z_p}$$

That is, $\tilde{p}(\boldsymbol{x})$ is easy to evaluate and Z_p is unknown (or intractable)

- We may also wish to use a $q(\boldsymbol{x})$ with the same property

$$q(\boldsymbol{x}) = \frac{\tilde{q}(\boldsymbol{x})}{Z_q}$$

- The importance sampling estimator is then given by

$$\begin{aligned} \hat{s}_q &= \frac{1}{n} \sum_{i=1}^{n} \frac{p(\boldsymbol{x}^{(i)})}{q(\boldsymbol{x}^{(i)})} f(\boldsymbol{x}^{(i)}) \\ &= \frac{Z_q}{Z_p} \frac{1}{n} \sum_{i=1}^{n} \frac{\tilde{p}(\boldsymbol{x}^{(i)})}{\tilde{q}(\boldsymbol{x}^{(i)})} f(\boldsymbol{x}^{(i)}) \end{aligned}$$

$$= \frac{Z_q}{Z_p} \frac{1}{n} \sum_{i=1}^{n} \tilde{r}_i f(\boldsymbol{x}^{(i)})$$

where

$$\tilde{r}_i = \frac{\tilde{p}(\boldsymbol{x}^{(i)})}{\tilde{q}(\boldsymbol{x}^{(i)})} \text{ and } \boldsymbol{x}^{(i)} \sim q(\boldsymbol{x})$$

- The same set of data $\boldsymbol{x}^{(i)}$ can be used to approximate the ratio Z_p/Z_q

$$\begin{aligned}\frac{Z_p}{Z_q} &= \frac{\sum_{\boldsymbol{x}} \tilde{p}(\boldsymbol{x})}{Z_q} \\ &= \sum_{\boldsymbol{x}} \tilde{p}(\boldsymbol{x}) \frac{1}{Z_q} \\ &= \sum_{\boldsymbol{x}} \tilde{p}(\boldsymbol{x}) \frac{q(\boldsymbol{x})}{\tilde{q}(\boldsymbol{x})} \\ &= \sum_{\boldsymbol{x}} \frac{\tilde{p}(\boldsymbol{x})}{\tilde{q}(\boldsymbol{x})} q(\boldsymbol{x})\end{aligned}$$

$$\backsimeq \frac{1}{n}\sum_i \frac{\tilde{p}(\boldsymbol{x}^{(i)})}{\tilde{q}(\boldsymbol{x}^{(i)})}$$

$$= \frac{1}{n}\sum_i \tilde{r}_i$$

- We then arrive at a *biased importance estimator*

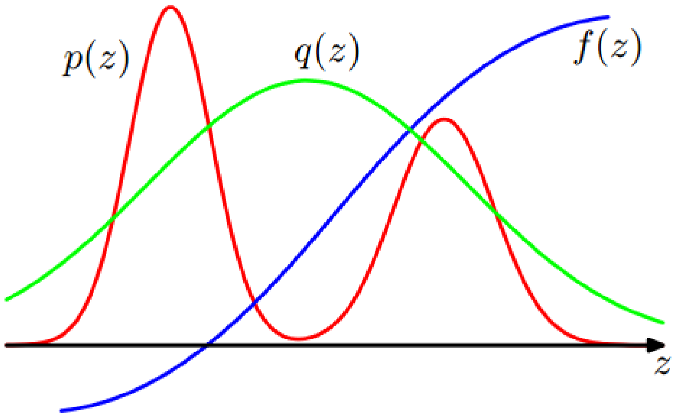
$$\hat{s}_{BIS} = \frac{\sum_{i=1}^{n} \tilde{r}_i f(\boldsymbol{x}^{(i)})}{\sum_{i=1}^{n} \tilde{r}_i} = \sum_{i=1}^{n} \tilde{w}_i f(\boldsymbol{x}^{(i)})$$

  where

$$\tilde{w}_i = \frac{\tilde{r}_i}{\sum_{i=1}^{n} \tilde{r}_i}$$

- $\hat{s}_{BIS}$ is asymptotically unbiased; that is, as $n \to \infty$, $E[\hat{s}_{BIS}] = s$

- The success of importance sampling depends crucially on how well $q(\boldsymbol{x})$ matches the desired distribution $p(\boldsymbol{x})$
- When $p(\boldsymbol{x})f(\boldsymbol{x})$ is strongly varying and has its mass concentrated over small regions of $\boldsymbol{x}$ space, most samples collected may be useless since they contribute little to the final estimate due to the fact $q(\boldsymbol{x}^{(i)}) \gg p(\boldsymbol{x}^{(i)})|f(\boldsymbol{x}^{(i)})|$
- As such, underestimation of $E_p[f(\boldsymbol{x})]$ is typical, especially when $\boldsymbol{x}$ is high dimensional



Markov Chain Monte Carlo Methods

- Methods that involve drawing samples from Markov chains to perform Monte Carlo estimation

- Drawing samples from a Markov Chain

  1. Start with an initial state $\boldsymbol{x}^{(1)}$
  2. Sample repeatedly from transition distributions $p(\boldsymbol{x}^{(\tau+1)}|\boldsymbol{x}^{(\tau)})$

$$\text{Sample } \ \boldsymbol{x}^{(\tau+1)} \sim p(\boldsymbol{x}^{(\tau+1)}|\boldsymbol{x}^{(\tau)}), \ \tau = 1, \ldots, t-1$$

- Given a desired distribution $p^\star(\boldsymbol{x})$, we choose transition distributions such that $\boldsymbol{x}^{(t)}$ eventually becomes a fair sample of $p^\star(\boldsymbol{x})$

First-Order Markov Chains

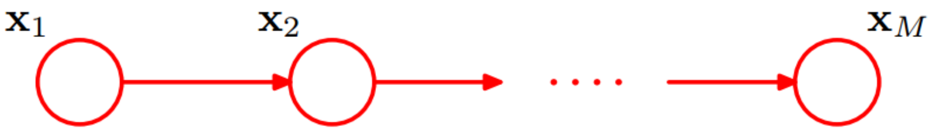
- A sequence of discrete-valued random variables $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(M)}$ with the conditional independence property

$$p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(m)}, \ldots, \boldsymbol{x}^{(1)}) = p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(m)}),$$

  for $m \in \{1, \ldots, M-1\}$

- The joint distribution of $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(M)}$ is characterized by $p(\boldsymbol{x}^{(1)})$ together with the transition probabilities $p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(m)})$

$$p(\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(M)}) = p(\boldsymbol{x}^{(1)}) \prod_{i=1}^{M-1} p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(m)})$$



- The marginal distribution $p(\boldsymbol{x}^{(m+1)})$ can be expressed as

$$p(\boldsymbol{x}^{(m+1)}) = \sum_{\boldsymbol{x}^{(m)}} p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(m)})p(\boldsymbol{x}^{(m)})$$

- In matrix form, we have

$$\boldsymbol{v}^{(m+1)} = \boldsymbol{A}^{(m)}\boldsymbol{v}^{(m)}$$

where

$$\begin{aligned} v_i^{(m+1)} &= p(\boldsymbol{x}^{(m+1)} = \boldsymbol{s}_i), && \text{Prob. of } \boldsymbol{x}^{(m+1)} \text{ in state } \boldsymbol{s}_i \\ v_j^{(m)} &= p(\boldsymbol{x}^{(m)} = \boldsymbol{s}_j), && \text{Prob. of } \boldsymbol{x}^{(m)} \text{ in state } \boldsymbol{s}_j \\ A_{i,j}^{(m)} &= p(\boldsymbol{x}^{(m+1)} = \boldsymbol{s}_i|\boldsymbol{x}^{(m)} = \boldsymbol{s}_j), && \text{Transition probabilities} \end{aligned}$$

- A Markov chain is said to be homogeneous if the transition probability $p(\boldsymbol{x}^{(m+1)}|\boldsymbol{x}^{(m)})$ does not depend on m

- In this case, we see that $\boldsymbol{A}^{(m)} = \boldsymbol{A}$ is a constant matrix and that over time, all the eigenvalues are exponentiated

$$\boldsymbol{v}^{(t)} = \boldsymbol{A}^{t-1}\boldsymbol{v}^{(1)} = \boldsymbol{U}\Lambda^{t-1}\boldsymbol{U}^{-1}\boldsymbol{v}^{(1)}$$

- Under some conditions (e.g. non-zero transition probabilities), $\boldsymbol{A}$ has only one eigenvector $\boldsymbol{v}$ with the largest eigenvalue 1
- $\boldsymbol{v}^{(t)}$ eventually converges to that eigenvector $\boldsymbol{v}$, which denotes the equilibrium distribution, regardless of the choice of initial state $\boldsymbol{v}^{(1)}$

$$\boldsymbol{A}\boldsymbol{v} = \boldsymbol{v}$$

- We hope that by choosing transition probabilities correctly, $\boldsymbol{v}$ will be equal to the distribution we wish to sample from

- Running the Markov chain until it reaches its equilibrium is called burning in and the time required is called the mixing time
- Unfortunately, we only know that the chain will converge under some mild conditions, but not how much time it will take
- Most properties of discrete-valued Markov chains as presented here can carry over to the continuous-valued case

Gibbs Sampling

- To build a Markov chain that samples from a distribution $p_{\text{model}}(\boldsymbol{x})$

$$p_{\text{model}}(\boldsymbol{x}) = p_{\text{model}}(x_1, x_2, \ldots, x_M)$$

- Procedure
  1. Start with an initial state $x_i^{(1)}, i = 1, 2, \ldots, M$
  2. For $\tau = 1, \ldots, t-1$
     - Sample $x_1^{(\tau+1)} \sim p_{\text{model}}(x_1 | x_2^{(\tau)}, x_3^{(\tau)}, \ldots, x_M^{(\tau)})$
     - Sample $x_2^{(\tau+1)} \sim p_{\text{model}}(x_2 | x_1^{(\tau+1)}, x_3^{(\tau)}, \ldots, x_M^{(\tau)})$

       $\vdots$
     - Sample $x_j^{(\tau+1)} \sim p_{\text{model}}(x_j | x_1^{(\tau+1)}, \ldots, x_{j-1}^{(\tau+1)}, x_{j+1}^{(\tau)} \ldots, x_M^{(\tau)})$

       $\vdots$
     - Sample $x_M^{(\tau+1)} \sim p_{\text{model}}(x_M | x_1^{(\tau+1)}, x_2^{(\tau+1)}, \ldots, x_{M-1}^{(\tau+1)})$

- In words, each step replaces one variable x_i by drawing a sample from the distribution $p_{\text{model}}(x_i|\boldsymbol{x}_{-i})$ of x_i conditioned on the values of the remaining variables $\boldsymbol{x}_{-i}$

- This procedure eventually yields samples of $p_{\text{model}}(\boldsymbol{x})$ because
  - The resulting Markov chain will converge to an equilibrium distribution, if none of the transition probabilities is zero anywhere
  - $p_{\text{model}}(\boldsymbol{x})$ is invariant w.r.t. this Markov chain

- A distribution $p^\star(\boldsymbol{x})$ is said to be invariant w.r.t. a Markov chain if each step in the chain leaves that distribution invariant, i.e.

$$p(\boldsymbol{x}') = \sum_{\boldsymbol{x}} p(\boldsymbol{x}'|\boldsymbol{x}) p^\star(\boldsymbol{x}) = p^\star(\boldsymbol{x}')$$

- In the present case, we have

$$\boldsymbol{x} = (x_i^{old}, \boldsymbol{x}_{-i}^{old}) \sim p_{\textsf{model}}(\boldsymbol{x})$$

$$\boldsymbol{x}' = (x_i^{new}, \boldsymbol{x}_{-i}^{old}) \text{ with } x_i^{new} \sim p_{\textsf{model}}(x_i|\boldsymbol{x}_{-i}^{old})$$

- It can be shown that $p(\boldsymbol{x}') = p_{\textsf{model}}(\boldsymbol{x}')$; that is, $p_{\textsf{model}}(\boldsymbol{x})$ is invariant

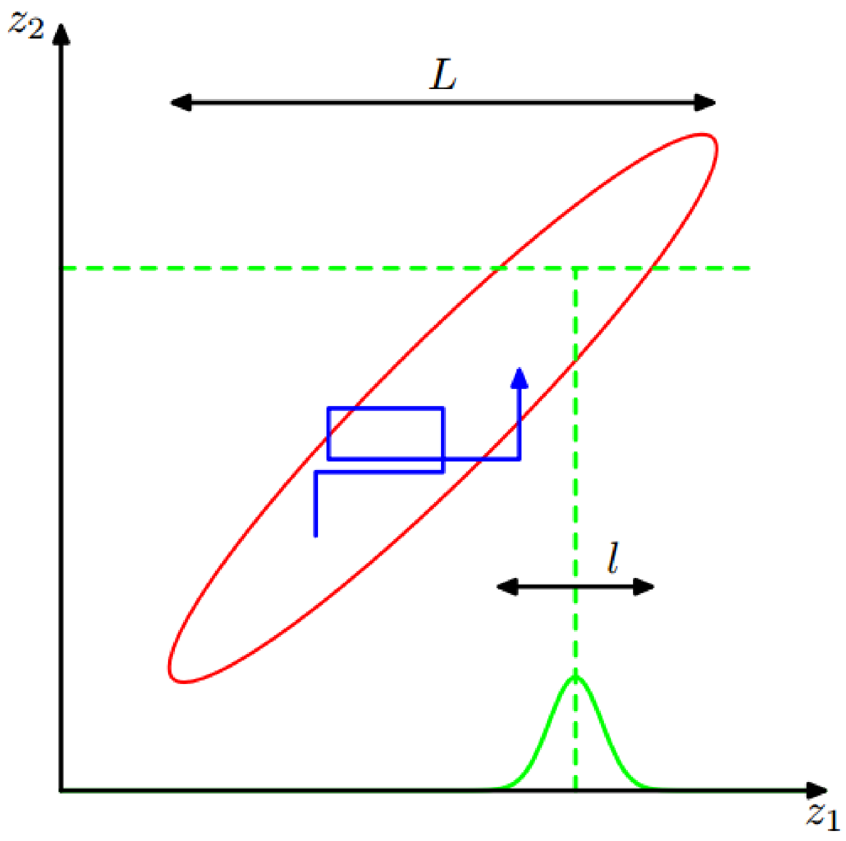
$$\begin{aligned} p(\boldsymbol{x}') &= p(x_i^{new}, \boldsymbol{x}_{-i}^{old}) \\ &= p(\boldsymbol{x}_{-i}^{old})p(x_i^{new}|\boldsymbol{x}_{-i}^{old}) \\ &= p_{\textsf{model}}(\boldsymbol{x}_{-i}^{old})p_{\textsf{model}}(x_i^{new}|\boldsymbol{x}_{-i}^{old}) \\ &= p_{\textsf{model}}(x_i^{new}, \boldsymbol{x}_{-i}^{old}) \\ &= p_{\textsf{model}}(\boldsymbol{x}') \end{aligned}$$

- **Block Gibbs sampling:** In some cases, it is possible to sample many variables simultaneously; for example, in RBM, $p(\boldsymbol{h}|\boldsymbol{v})$ and $p(\boldsymbol{v}|\boldsymbol{h})$ are factorial, suggesting that the elements of $\boldsymbol{h}$ and of $\boldsymbol{v}$ can be sampled simultaneously

Challenges

- Successive samples are preferably independent and different regions in $\boldsymbol{x}$ space should be visited proportional to their probability
- In reality, successive samples are highly correlated even though they have identical distributions
- Independent samples may be obtained by retaining every M samples for sufficiently large M, or by running multiple chains in parallel

- Moreover, Gibbs sampling may mix slowly when the variables of $p_{\text{model}}(\boldsymbol{x})$ are highly correlated



Sampling a correlated Gaussian of two variables

- Mixing between modes may be difficult if they are widely separated by regions of low probability
  - Toy problem: Consider the following energy model

    $$\tilde{p}(a,b) = \exp(-E(a,b)),\ a,b \in \{-1,1\}$$

    where

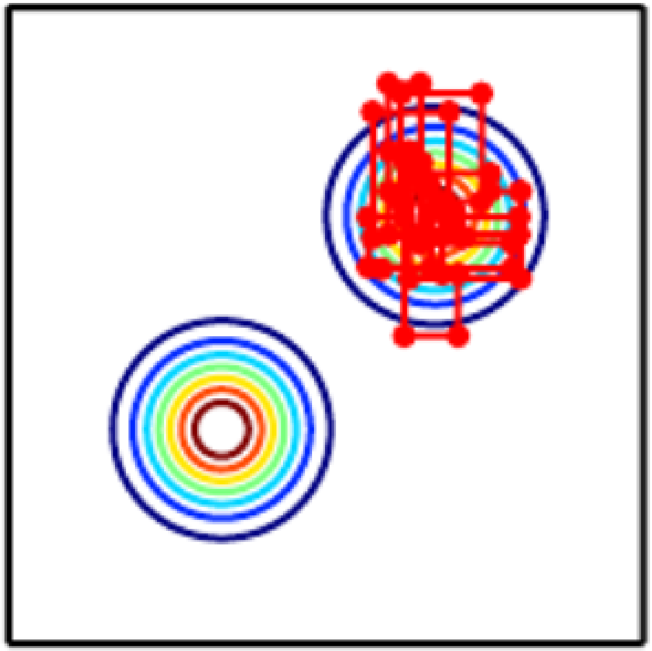
    $$E(a,b) = -wab$$

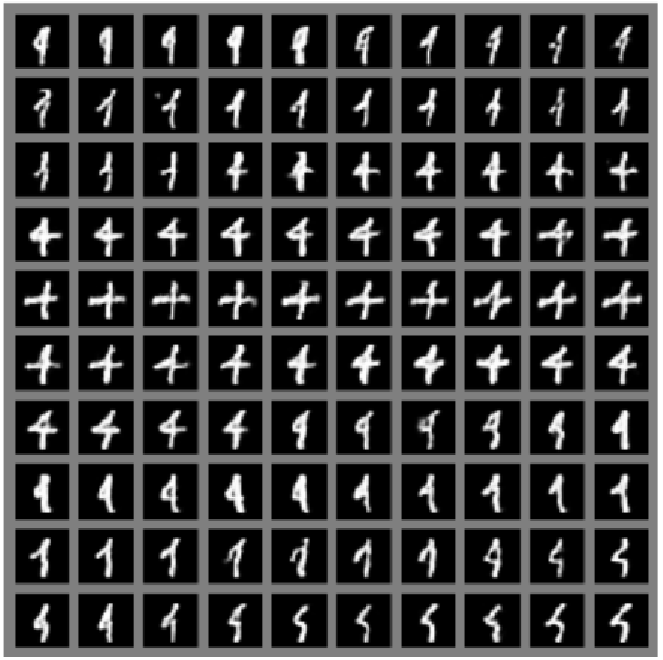
  - It is seen that

    $$p(b=1|a=1) = \sigma(w)$$

  - When w is extremely large, Gibbs sampling will only rarely flip the signs of a, b even if $p(b=1,a=1) = p(b=-1,a=-1)$

– More examples:





Confronting The Partition Function

- Many undirected graphical models are defined by an unnormalized distribution $\tilde{p}_{\text{model}}(\boldsymbol{x};\boldsymbol{\theta})$ with an intractable partition function $Z(\boldsymbol{\theta})$

$$p_{\text{model}}(\boldsymbol{x};\boldsymbol{\theta}) = \frac{\tilde{p}_{\text{model}}(\boldsymbol{x};\boldsymbol{\theta})}{Z(\boldsymbol{\theta})}$$

where

$$Z(\boldsymbol{\theta}) = \sum_{\boldsymbol{x}} \tilde{p}_{\text{model}}(\boldsymbol{x};\boldsymbol{\theta}) \; or \; Z(\boldsymbol{\theta}) = \int_{\boldsymbol{x}} \tilde{p}_{\text{model}}(\boldsymbol{x};\boldsymbol{\theta}) d\boldsymbol{x}$$

- For training, we maximize the log-likelihood w.r.t. training data

$$E_{\boldsymbol{x}\sim p_{\text{data}}} \log p_{\text{model}}(\boldsymbol{x};\boldsymbol{\theta}) = E_{\boldsymbol{x}\sim p_{\text{data}}} \log \tilde{p}_{\text{model}}(\boldsymbol{x};\boldsymbol{\theta}) - \log Z(\boldsymbol{\theta})$$

through gradient descent

$$\nabla_{\boldsymbol{\theta}} E_{\boldsymbol{x}\sim p_{\text{data}}} \log p_{\text{model}}(\boldsymbol{x};\boldsymbol{\theta}) = \underbrace{E_{\boldsymbol{x}\sim p_{\text{data}}} \nabla_{\boldsymbol{\theta}} \log \tilde{p}_{\text{model}}(\boldsymbol{x};\boldsymbol{\theta})}_{\text{Positive phase}} - \underbrace{\nabla_{\boldsymbol{\theta}} \log Z(\boldsymbol{\theta})}_{\text{Negative phase}}$$

- For discrete-valued $\boldsymbol{x}$, the gradient of $\log Z$ can be evaluated as

$$\nabla_{\boldsymbol{\theta}} \log Z(\boldsymbol{\theta}) = \frac{\nabla_{\boldsymbol{\theta}} Z(\boldsymbol{\theta})}{Z(\boldsymbol{\theta})} = \frac{\nabla_{\boldsymbol{\theta}} \sum_{\boldsymbol{x}} \tilde{p}_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta})}{Z(\boldsymbol{\theta})} = \frac{\sum_{\boldsymbol{x}} \nabla_{\boldsymbol{\theta}} \tilde{p}_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta})}{Z(\boldsymbol{\theta})}$$

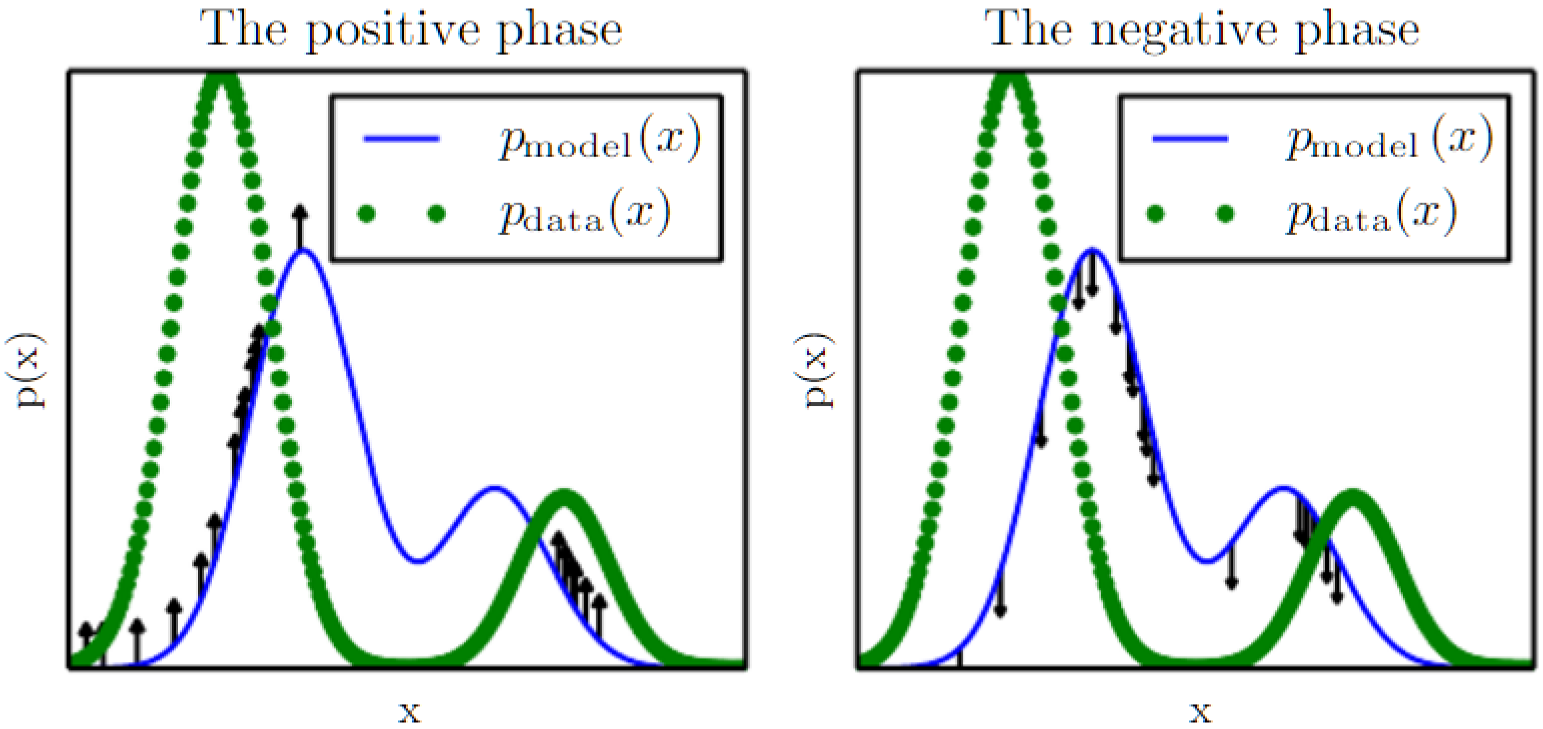
- Additionally, if $\tilde{p}_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta}) > 0$ for all $\boldsymbol{x}$ (e.g. energy-based models),

$$\begin{aligned}
\frac{\sum_{\boldsymbol{x}} \nabla_{\boldsymbol{\theta}} \tilde{p}_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta})}{Z(\boldsymbol{\theta})} &= \frac{\sum_{\boldsymbol{x}} \nabla_{\boldsymbol{\theta}} \exp(\log \tilde{p}_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta}))}{Z(\boldsymbol{\theta})} \\
&= \frac{\sum_{\boldsymbol{x}} \exp(\log \tilde{p}_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta})) \nabla_{\boldsymbol{\theta}} \log \tilde{p}_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta})}{Z(\boldsymbol{\theta})} \\
&= \frac{\sum_{\boldsymbol{x}} \tilde{p}_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} \log \tilde{p}_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta})}{Z(\boldsymbol{\theta})} \\
&= \sum_{\boldsymbol{x}} p_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} \log \tilde{p}_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta}) \\
&= E_{\boldsymbol{x} \sim p_{\mathsf{model}}} \nabla_{\boldsymbol{\theta}} \log \tilde{p}_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta})
\end{aligned}$$

- To summarize, we see that

$$\begin{aligned}&\nabla_{\boldsymbol{\theta}} E_{\boldsymbol{x}\sim p_{\text{data}}} \log p_{\text{model}}(\boldsymbol{x};\boldsymbol{\theta})\\ &= E_{\boldsymbol{x}\sim p_{\text{data}}} \nabla_{\boldsymbol{\theta}} \log \tilde{p}_{\text{model}}(\boldsymbol{x};\boldsymbol{\theta}) - E_{\boldsymbol{x}\sim p_{\text{model}}} \nabla_{\boldsymbol{\theta}} \log \tilde{p}_{\text{model}}(\boldsymbol{x};\boldsymbol{\theta})\end{aligned}$$

- In the positive phase, we increase the log-likelihood by increasing $\log \tilde{p}(\boldsymbol{x};\boldsymbol{\theta})$ with $\boldsymbol{x}$ drawn from training data $p_{\text{data}}(\boldsymbol{x})$
- In the negative phase, we increase the log-likelihood by decreasing the partition function $Z(\boldsymbol{\theta})$, or equivalently, by decreasing $\log \tilde{p}(\boldsymbol{x};\boldsymbol{\theta})$ with $\boldsymbol{x}$ drawn from the model distribution $p_{\text{model}}(\boldsymbol{x})$
- When $p_{\text{model}}(\boldsymbol{x}) = p_{\text{data}}(\boldsymbol{x})$, there is no longer gradient


The positive phase
The negative phase
$p_{\text{model}}(x)$
$p_{\text{data}}(x)$
p(x)
x

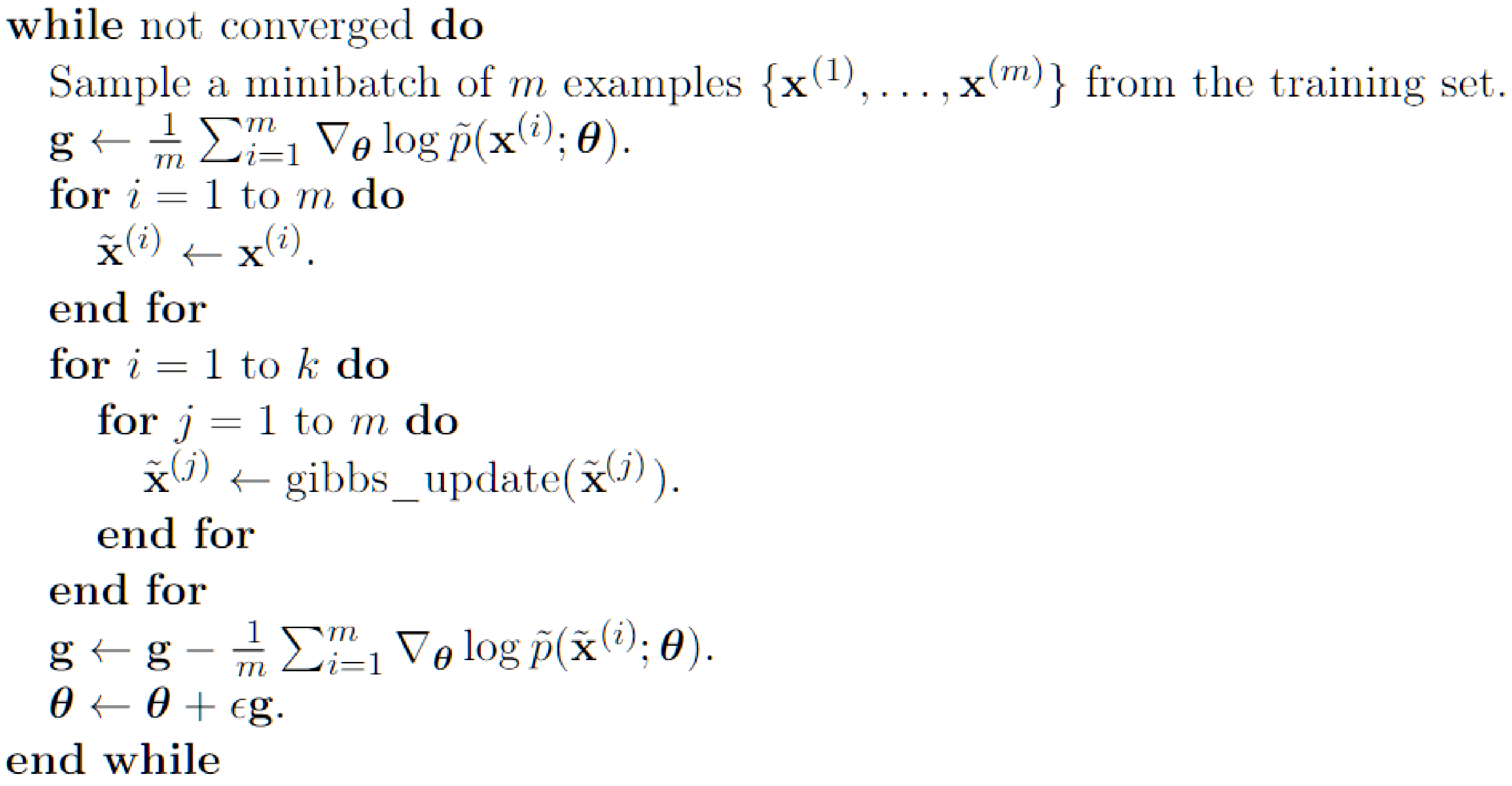
Contrastive Divergence and Its Variants

- To compute the gradient of the negative phase with Gibbs sampling

$$E_{\boldsymbol{x}\sim p_{\mathsf{model}}} \nabla_{\boldsymbol{\theta}} \log \tilde{p}_{\mathsf{model}}(\boldsymbol{x};\boldsymbol{\theta})$$

- There are different strategies for initializing the Markov chains
  - Contrastive divergence (CD) – from training data
  - Persistent contrastive divergence (PCD) – from previous step
  - (Study by yourself)

- Example: Contrastive Divergence (CD)



```
while not converged do
  Sample a minibatch of $m$ examples $\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}\}$ from the training set.
  $\mathbf{g} \leftarrow \frac{1}{m} \sum_{i=1}^{m} \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\mathbf{x}^{(i)}; \boldsymbol{\theta})$.
  for $i = 1$ to $m$ do
    $\tilde{\mathbf{x}}^{(i)} \leftarrow \mathbf{x}^{(i)}$.
  end for
  for $i = 1$ to $k$ do
    for $j = 1$ to $m$ do
      $\tilde{\mathbf{x}}^{(j)} \leftarrow$ gibbs_update($\tilde{\mathbf{x}}^{(j)}$).
    end for
  end for
  $\mathbf{g} \leftarrow \mathbf{g} - \frac{1}{m} \sum_{i=1}^{m} \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\tilde{\mathbf{x}}^{(i)}; \boldsymbol{\theta})$.
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \epsilon \mathbf{g}$.
end while
```

Review

- Why sampling?
- Importance sampling
- Gibbs sampling
- Issues with mixing of MCMC methods
- MCMC approach to learning with intractable partition functions