Advanced Machine Learning
Lecture 7

Approximate Inference II

14.11.2016

Bastian Leibe
RWTH Aachen
http://www.vision.rwth-aachen.de/

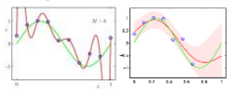
leibe@vision.rwth-aachen.de

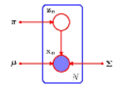
Advanced Machine Learning Winter'16

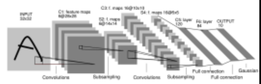
This Lecture: *Advanced Machine Learning*

$f : \mathcal{X} \to \mathbb{R}$

- Regression Approaches
  - Linear Regression
  - Regularization (Ridge, Lasso)
  - Gaussian Processes
- Learning with Latent Variables
  - Probability Distributions
  - Approximate Inference
- Deep Learning
  - Neural Networks
  - CNNs, RNNs, ResNets, etc.







Advanced Machine Learning Winter'16

Topics of This Lecture

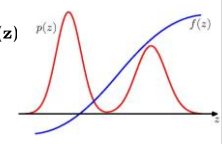
- Recap: Sampling approaches
  - Sampling from a distribution
  - Rejection Sampling
  - Importance Sampling
  - Sampling-Importance-Resampling
- Markov Chain Monte Carlo
  - Markov Chains
  - Metropolis Algorithm
  - Metropolis-Hastings Algorithm
  - Gibbs Sampling

Advanced Machine Learning Winter'16

Recap: Sampling Idea

- Objective:
  - Evaluate expectation of a function $f(\mathbf{z})$ w.r.t. a probability distribution $p(\mathbf{z})$.

$$\mathbb{E}[f] = \int f(\mathbf{z})p(\mathbf{z})d\mathbf{z}$$



- Sampling idea
  - Draw L independent samples $\mathbf{z}^{(l)}$ with l = 1,…,L from $p(\mathbf{z})$.
  - This allows the expectation to be approximated by a finite sum

$$\hat{f} = \frac{1}{L}\sum_{l=1}^{L} f(\mathbf{z}^l)$$

  - As long as the samples $\mathbf{z}^{(l)}$ are drawn independently from $p(\mathbf{z})$, then $\mathbb{E}[\hat{f}] = \mathbb{E}[f]$

⇒ Unbiased estimate, independent of the dimension of $\mathbf{z}$!

Slide adapted from Bernt Schiele

Image source: C.M. Bishop, 2006

Advanced Machine Learning Winter'16

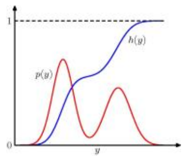
Recap: Sampling from a pdf

- In general, assume we are given the pdf $p(\mathbf{x})$ and the corresponding cumulative distribution:

$$F(x) = \int_{-\infty}^{x} p(z)dz$$

- To draw samples from this pdf, we can invert the cumulative distribution function:

$$u \sim \mathit{Uniform}(0,1) \Rightarrow F^{-1}(u) \sim p(x)$$



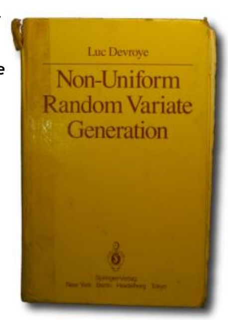
Slide credit: Bernt Schiele

Image source: C.M. Bishop, 2006

Advanced Machine Learning Winter'16

General Advice

- Use library functions whenever possible
  - Many efficient algorithms available for known univariate distributions (and some other special cases)
  - This book (free online) explains how some of them work
  - http://www.nrbook.com/devroye/



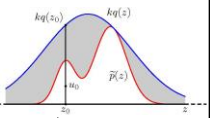
Slide credit: Iain Murray

Recap: Rejection Sampling

- Assumptions
  - Sampling directly from $p(\mathbf{z})$ is difficult.
  - But we can easily evaluate $p(\mathbf{z})$ (up to some norm. factor Z_p):

$$p(\mathbf{z}) = \frac{1}{Z_p}\tilde{p}(\mathbf{z})$$

- Idea
  - We need some simpler distribution $q(\mathbf{z})$ (called proposal distribution) from which we can draw samples.
  - Choose a constant k such that: $\forall z: kq(z) \geq \tilde{p}(z)$
- Sampling procedure
  - Generate a number z_0 from $q(z)$.
  - Generate a number u_0 from the uniform distribution over $[0, kq(z_0)]$.
  - If $u_0 > \tilde{p}(z_0)$ reject sample, otherwise accept.



Evaluating Expectations

- Motivation
  - Often, our goal is not sampling from $p(\mathbf{z})$ by itself, but to evaluate expectations of the form

$$\mathbb{E}[f] = \int f(\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

  - Assumption again: can evaluate $p(\mathbf{z})$ up to normalization factor.
- Simplistic strategy: Grid sampling
  - Discretize $\mathbf{z}$-space into a uniform grid.
  - Evaluate the integrand as a sum of the form

$$\mathbb{E}[f] \simeq \sum_{l=1}^{L} f(\mathbf{z}^{(l)})p(\mathbf{z}^{(l)})d\mathbf{z}$$

  - Problem: number of terms grows exponentially with number of dimensions!

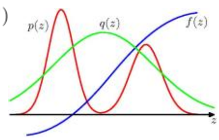
Importance Sampling

- Idea
  - Method approximates expectations directly (but does not enable to draw samples from $p(\mathbf{z})$ directly).
  - Use a proposal distribution $q(\mathbf{z})$ from we can easily draw samples
  - Express expectations in the form of a finite sum over samples $\{\mathbf{z}^{(l)}\}$ drawn from $q(\mathbf{z})$.

$$\begin{aligned}\mathbb{E}[f] &= \int f(\mathbf{z})p(\mathbf{z})d\mathbf{z} = \int f(\mathbf{z})\frac{p(\mathbf{z})}{q(\mathbf{z})}q(\mathbf{z})d\mathbf{z} \\ &\simeq \frac{1}{L}\sum_{l=1}^{L}\boxed{\frac{p(\mathbf{z}^{(l)})}{q(\mathbf{z}^{(l)})}}f(\mathbf{z}^{(l)})\end{aligned}$$

  - with importance weights

$$\boxed{r_l = \frac{p(\mathbf{z}^{(l)})}{q(\mathbf{z}^{(l)})}}$$



Importance Sampling

- Typical setting:
  - $p(\mathbf{z})$ can only be evaluated up to an unknown normalization constant $p(\mathbf{z}) = \tilde{p}(\mathbf{z})/Z_p$
  - $q(\mathbf{z})$ can also be treated in a similar fashion.

$$q(\mathbf{z}) = \tilde{q}(\mathbf{z})/Z_q$$

  - Then

$$\begin{aligned}\mathbb{E}[f] &= \int f(\mathbf{z})p(\mathbf{z})d\mathbf{z} = \frac{Z_q}{Z_p}\int f(\mathbf{z})\frac{\tilde{p}(\mathbf{z})}{\tilde{q}(\mathbf{z})}q(\mathbf{z})d\mathbf{z} \\ &\simeq \frac{Z_q}{Z_p}\frac{1}{L}\sum_{l=1}^{L}\tilde{r}_l f(\mathbf{z}^{(l)})\end{aligned}$$

  - with: $\tilde{r}_l = \dfrac{\tilde{p}(\mathbf{z}^{(l)})}{\tilde{q}(\mathbf{z}^{(l)})}$

Importance Sampling

- Removing the unknown normalization constants
  - We can use the sample set to evaluate the ratio of normalization constants

$$\frac{Z_p}{Z_q} = \frac{1}{Z_q}\int \tilde{p}(\mathbf{z})d\mathbf{z} = \int \frac{\tilde{p}(\mathbf{z}^{(l)})}{\tilde{q}(\mathbf{z}^{(l)})}q(\mathbf{z})d\mathbf{z} \simeq \frac{1}{L}\sum_{l=1}^{L}\tilde{r}_l$$

  - and therefore

$$\mathbb{E}[f] \simeq \sum_{l=1}^{L} w_l f(\mathbf{z}^{(l)})$$

with $$w_l = \frac{\tilde{r}_l}{\sum_m \tilde{r}_m} = \frac{\frac{\tilde{p}(\mathbf{z}^{(l)})}{\tilde{q}(\mathbf{z}^{(l)})}}{\sum_m \frac{\tilde{p}(\mathbf{z}^{(m)})}{\tilde{q}(\mathbf{z}^{(m)})}}$$

⇒ In contrast to Rejection Sampling, all generated samples are retained (but they may get a small weight).

Importance Sampling - Discussion

- Observations
  - Success of importance sampling depends crucially on how well the sampling distribution $q(\mathbf{z})$ matches the desired distribution $p(\mathbf{z})$.
  - Often, $p(\mathbf{z})f(\mathbf{z})$ is strongly varying and has a significant proportion of its mass concentrated over small regions of $\mathbf{z}$-space.

  ⇒ Weights r_l may be dominated by a few weights having large values.
  - Practical issue: if none of the samples falls in the regions where $p(\mathbf{z})f(\mathbf{z})$ is large…
    - The results may be arbitrary in error.
    - And there will be no diagnostic indication (no large variance in r_l)!
  - Key requirement for sampling distribution $q(\mathbf{z})$:
    - Should not be small or zero in regions where $p(\mathbf{z})$ is significant!

Sampling-Importance-Resampling (SIR)

- Observation
  - Success of rejection sampling depends on finding a good value for the constant k.
  - For many pairs of distributions $p(\mathbf{z})$ and $q(\mathbf{z})$, it will be impractical to determine a suitable value for k.
    - Any value that is sufficiently large to guarantee $q(\mathbf{z}) \geq p(\mathbf{z})$ will lead to impractically small acceptance rates.
- Sampling-Importance-Resampling Approach
  - Also makes use of a sampling distribution $q(\mathbf{z})$, but avoids having to determine k.

Sampling-Importance-Resampling

- Two stages
  - Draw L samples $\mathbf{z}^{(1)},\ldots,\mathbf{z}^{(L)}$ from $q(\mathbf{z})$.
  - Construct weights using importance weighting

$$w_l = \frac{\tilde{r}_l}{\sum_m \tilde{r}_m} = \frac{\frac{\tilde{p}(\mathbf{z}^{(l)})}{\tilde{q}(\mathbf{z}^{(l)})}}{\sum_m \frac{\tilde{p}(\mathbf{z}^{(m)})}{\tilde{q}(\mathbf{z}^{(m)})}}$$

  and draw a second set of samples $\mathbf{z}^{(1)},\ldots,\mathbf{z}^{(L)}$ with probabilities given by the weights $w^{(1)},\ldots,w^{(L)}$.
- Result
  - The resulting L samples are only approximately distributed according to $p(\mathbf{z})$, but the distribution becomes correct in the limit $L \rightarrow \infty$.

Curse of Dimensionality

- Problem
  - Rejection & Importance Sampling both scale badly with high dimensionality.
  - Example:

$$p(\mathbf{z}) \sim \mathcal{N}(0, I), \qquad q(\mathbf{z}) \sim \mathcal{N}(0, \sigma^2 I)$$

- Rejection Sampling
  - Requires $\sigma \geq 1$. Fraction of proposals accepted: σ^{-D}.
- Importance Sampling
  - Variance of importance weights: $\left(\frac{\sigma^2}{2 - 1/\sigma^2}\right)^{D/2} - 1$
  - Infinite / undefined variance if $\sigma \leq 1/\sqrt{2}$

Slide credit: Iain Murray

Topics of This Lecture

- Recap: Sampling approaches
  - Sampling from a distribution
  - Rejection Sampling
  - Importance Sampling
  - Sampling-Importance-Resampling
- Markov Chain Monte Carlo
  - Markov Chains
  - Metropolis Algorithm
  - Metropolis-Hastings Algorithm
  - Gibbs Sampling

Independent Sampling vs. Markov Chains

- So far
  - We've considered three methods, Rejection Sampling, Importance Sampling, and SIR, which were all based on independent samples from $q(\mathbf{z})$.
  - However, for many problems of practical interest, it is often difficult or impossible to find $q(\mathbf{z})$ with the necessary properties.
  - In addition, those methods suffer from severe limitations in high-dimensional spaces.
- Different approach
  - We abandon the idea of independent sampling.
  - Instead, rely on a Markov Chain to generate dependent samples from the target distribution.
  - Independence would be a nice thing, but it is not necessary for the Monte Carlo estimate to be valid.

Slide credit: Zoubin Ghahramani

MCMC – Markov Chain Monte Carlo

- Overview
  - Allows to sample from a large class of distributions.
  - Scales well with the dimensionality of the sample space.
- Idea
  - We maintain a record of the current state $\mathbf{z}^{(\tau)}$
  - The proposal distribution depends on the current state: $q(\mathbf{z}|\mathbf{z}^{(\tau)})$
  - The sequence of samples forms a Markov chain $\mathbf{z}^{(1)}, \mathbf{z}^{(2)},\ldots$
- Setting
  - We can evaluate $p(\mathbf{z})$ (up to some normalizing factor Z_p):

$$p(\mathbf{z}) = \frac{\tilde{p}(\mathbf{z})}{Z_p}$$

  - At each time step, we generate a candidate sample from the proposal distribution and accept the sample according to a criterion.

Slide credit: Bernt Schiele

MCMC – Metropolis Algorithm

- Metropolis algorithm [Metropolis et al., 1953]
  - Proposal distribution is symmetric: $q(\mathbf{z}_A|\mathbf{z}_B) = q(\mathbf{z}_B|\mathbf{z}_A)$
  - The new candidate sample $\mathbf{z}^*$ is accepted with probability

$$A(\mathbf{z}^\star, \mathbf{z}^{(\tau)}) = \min\left(1, \frac{\tilde{p}(\mathbf{z}^\star)}{\tilde{p}(\mathbf{z}^{(\tau)})}\right)$$

- Implementation
  - Choose random number u uniformly from unit interval (0,1).
  - Accept sample if $A(\mathbf{z}^\star, \mathbf{z}^{(\tau)}) > u$.
- Note
  - New candidate samples always accepted if $\tilde{p}(\mathbf{z}^\star) \geq \tilde{p}(\mathbf{z}^{(\tau)})$.
    - I.e. when new sample has higher probability than the previous one.
  - The algorithm sometimes accepts a state with lower probability.

MCMC – Metropolis Algorithm

- Two cases
  - If new sample is accepted: $\mathbf{z}^{(\tau+1)} = \mathbf{z}^\star$
  - Otherwise: $\mathbf{z}^{(\tau+1)} = \mathbf{z}^{(\tau)}$
  - This is in contrast to rejection sampling, where rejected samples are simply discarded.

  $\Rightarrow$ Leads to multiple copies of the same sample!

MCMC – Metropolis Algorithm

- Property
  - When $q(\mathbf{z}_A|\mathbf{z}_B) > 0$ for all $\mathbf{z}$, the distribution of $\mathbf{z}^\tau$ tends to $p(\mathbf{z})$ as $\tau \rightarrow \infty$.
- Note
  - Sequence $\mathbf{z}^{(1)}$, $\mathbf{z}^{(2)}$,… is not a set of independent samples from $p(\mathbf{z})$, as successive samples are highly correlated.
  - We can obtain (largely) independent samples by just retaining every M[th] sample.
- Example: Sampling from a Gaussian
  - Proposal: Gaussian with σ = 0.2.
  - Green: accepted samples
  - Red: rejected samples

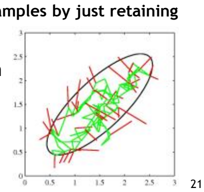
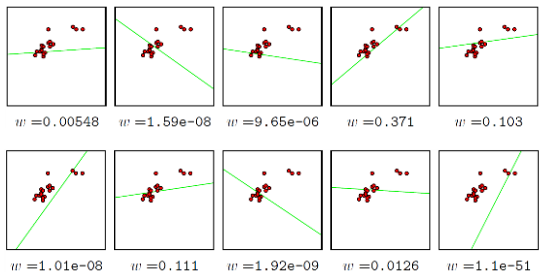


Image source: C.M. Bishop, 2006

Line Fitting Example

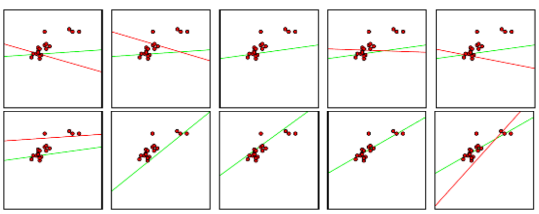
- Importance Sampling weights



$\Rightarrow$ Many samples with very low weights…

Line Fitting Example (cont'd)

- Metropolis algorithm



  - Perturb parameters: $Q(\mathbf{z}'; \mathbf{z})$, e.g. $\mathcal{N}(\mathbf{z}, \sigma^2)$
  - Accept with probability $\min\left(1, \frac{p(\mathbf{z}'|\mathcal{D})}{p(\mathbf{z}|\mathcal{D})}\right)$
  - Otherwise, keep old parameters.

Markov Chains

- Question
  - How can we show that $\mathbf{z}^\tau$ tends to $p(\mathbf{z})$ as $\tau \rightarrow \infty$?
- Markov chains
  - First-order Markov chain:

$$p\left(\mathbf{z}^{(m+1)}|\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(m)}\right) = p\left(\mathbf{z}^{(m+1)}|\mathbf{z}^{(m)}\right)$$

  - Marginal probability

$$p\left(\mathbf{z}^{(m+1)}\right) = \sum_{\mathbf{z}^{(m)}} p\left(\mathbf{z}^{(m+1)}|\mathbf{z}^{(m)}\right) p\left(\mathbf{z}^{(m)}\right)$$

  - A Markov chain is called homogeneous if the transition probabilities $p(\mathbf{z}^{(m+1)} \mid \mathbf{z}^{(m)})$ are the same for all m.

Markov Chains - Properties

- **Invariant distribution**
  - A distribution is said to be invariant (or stationary) w.r.t. a Markov chain if each step in the chain leaves that distribution invariant.
  - Transition probabilities:
$$T\left(\mathbf{z}^{(m)}, \mathbf{z}^{(m+1)}\right) = p\left(\mathbf{z}^{(m+1)} | \mathbf{z}^{(m)}\right)$$
  - For homogeneous Markov chain, distribution $p^*(\mathbf{z})$ is invariant if:
$$p^\star(\mathbf{z}) = \sum_{\mathbf{z}'} T(\mathbf{z}', \mathbf{z})\, p^\star(\mathbf{z}')$$
- **Detailed balance**
  - Sufficient (but not necessary) condition to ensure that a distribution is invariant:
$$p^\star(\mathbf{z}) T(\mathbf{z}, \mathbf{z}') = p^\star(\mathbf{z}') T(\mathbf{z}', \mathbf{z})$$
  - A Markov chain which respects *detailed balance* is reversible.

Slide credit: Bernt Schiele

Detailed Balance

- **Detailed balance** means
  - If we pick a state from the target distribution $p(\mathbf{z})$ and make a transition under T to another state, it is just as likely that we will pick $\mathbf{z}_A$ and go from $\mathbf{z}_A$ to $\mathbf{z}_B$ than that we will pick $\mathbf{z}_B$ and go from $\mathbf{z}_B$ to $\mathbf{z}_A$.
  - It can easily be seen that a transition probability that satisfies detailed balance w.r.t. a particular distribution will leave that distribution invariant, because
$$\begin{aligned}\sum_{\mathbf{z}'} p^\star(\mathbf{z}') T(\mathbf{z}', \mathbf{z}) &= \sum_{\mathbf{z}'} p^\star(\mathbf{z}) T(\mathbf{z}, \mathbf{z}') \\ &= p^\star(\mathbf{z}) \sum_{\mathbf{z}'} p(\mathbf{z}' | \mathbf{z}) = p^\star(\mathbf{z})\end{aligned}$$

Ergodicity in Markov Chains

- **Remark**
  - Our goal is to use Markov chains to sample from a given distribution.
  - We can achieve this if we set up a Markov chain such that the desired distribution is invariant.
  - However, must also require that for $m \to \infty$, the distribution $p(\mathbf{z}^{(m)})$ converges to the required invariant distribution $p^*(\mathbf{z})$ irrespective of the choice of initial distribution $p(\mathbf{z}^{(0)})$.
  - This property is called ergodicity and the invariant distribution is called the equilibrium distribution.
  - It can be shown that this is the case for a homogeneous Markov chain, subject only to weak restrictions on the invariant distribution and the transition probabilities.

Mixture Transition Distributions

- **Mixture distributions**
  - In practice, we often construct the transition probabilities from a set of 'base' transitions $B_1, \ldots, B_K$.
  - This can be achieved through a mixture distribution
$$T(\mathbf{z}', \mathbf{z}) = \sum_{k=1}^{K} \alpha_k B_k(\mathbf{z}', \mathbf{z})$$
  with mixing coefficients $\alpha_k \geq 0$ and $\sum_k \alpha_k = 1$.
- **Properties**
  - If the distribution is invariant w.r.t. each of the base transitions, then it will also be invariant w.r.t. T(z',z).
  - If each of the base transitions satisfies detailed balance, then the mixture transition T will also satisfy detailed balance.
  - Common example: each base transition changes only a subset of variables.

MCMC - Metropolis-Hastings Algorithm

- **Metropolis-Hastings Algorithm**
  - Generalization: Proposal distribution not required to be symmetric.
  - The new candidate sample $\mathbf{z}^*$ is accepted with probability
$$A(\mathbf{z}^\star, \mathbf{z}^{(\tau)}) = \min\left(1, \frac{\tilde{p}(\mathbf{z}^\star) q_k(\mathbf{z}^{(\tau)} | \mathbf{z}^\star)}{\tilde{p}(\mathbf{z}^{(\tau)}) q_k(\mathbf{z}^\star | \mathbf{z}^{(\tau)})}\right)$$
  - where k labels the members of the set of possible transitions considered.
- **Note**
  - Evaluation of acceptance criterion does not require normalizing constant Z_p.
  - When the proposal distributions are symmetric, Metropolis-Hastings reduces to the standard Metropolis algorithm.

Slide credit: Bernt Schiele

MCMC - Metropolis-Hastings Algorithm

- **Properties**
  - We can show that $p(\mathbf{z})$ is an invariant distribution of the Markov chain defined by the Metropolis-Hastings algorithm.
  - We show detailed balance:
$$A(\mathbf{z}', \mathbf{z}) = \min\left\{1, \frac{\tilde{p}(\mathbf{z}') q_k(\mathbf{z} | \mathbf{z}')}{\tilde{p}(\mathbf{z}) q_k(\mathbf{z}' | \mathbf{z})}\right\}$$
$$\begin{aligned}\tilde{p}(\mathbf{z}) q_k(\mathbf{z}' | \mathbf{z}) A_k(\mathbf{z}', \mathbf{z}) &= \min\{\tilde{p}(\mathbf{z}) q_k(\mathbf{z}' | \mathbf{z}), \tilde{p}(\mathbf{z}') q_k(\mathbf{z} | \mathbf{z}')\} \\ &= \min\{\tilde{p}(\mathbf{z}') q_k(\mathbf{z} | \mathbf{z}'), \tilde{p}(\mathbf{z}) q_k(\mathbf{z}' | \mathbf{z})\} \\ \tilde{p}(\mathbf{z}) q_k(\mathbf{z}' | \mathbf{z}) A_k(\mathbf{z}', \mathbf{z}) &= \tilde{p}(\mathbf{z}') q_k(\mathbf{z} | \mathbf{z}') A_k(\mathbf{z}, \mathbf{z}') \\ \tilde{p}(\mathbf{z}) T(\mathbf{z}', \mathbf{z}) &= \tilde{p}(\mathbf{z}') T(\mathbf{z}, \mathbf{z}')\end{aligned}$$

Note: This is wrong in the Bishop book!

Random Walks

- Example: Random Walk behavior
  - Consider a state space consisting of the integers $z \in \mathbb{Z}$ with initial state $z(1) = 0$ and transition probabilities

$$\begin{aligned} p(z^{(\tau+1)} = z^{(\tau)}) &= 0.5 \\ p(z^{(\tau+1)} = z^{(\tau)} + 1) &= 0.25 \\ p(z^{(\tau+1)} = z^{(\tau)} - 1) &= 0.25 \end{aligned}$$

- Analysis
  - Expected state at time τ : $\mathbb{E}[z^{(\tau)}] = 0$
  - Variance: $\mathbb{E}[(z^{(\tau)})^2] = \tau/2$
  - After τ steps, the random walk has only traversed a distance that is on average proportional to $\sqrt{\tau}$.

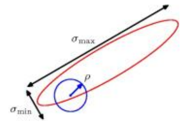
  ⇒ Central goal in MCMC is to avoid random walk behavior!

MCMC - Metropolis-Hastings Algorithm

- Schematic illustration
  - For continuous state spaces, a common choice of proposal distribution is a Gaussian centered on the current state.

  ⇒ What should be the variance of the proposal distribution?
    - Large variance: rejection rate will be high for complex problems.
    - The scale ρ of the proposal distribution should be as large as possible without incurring high rejection rates.

    ⇒ ρ should be of the same order as the smallest length scale $\sigma_{\min}$.



  - This causes the system to explore the distribution by means of a random walk.
    - Undesired behavior: number of steps to arrive at state that is independent of original state is of order $(\sigma_{\max}/\sigma_{\min})^2$.
    - Strong correlations can slow down the Metropolis(-Hastings) algorithm!

Image source: C.M. Bishop, 2006

Gibbs Sampling

- Approach
  - MCMC-algorithm that is simple and widely applicable.
  - May be seen as a special case of Metropolis-Hastings.
- Idea
  - Sample variable-wise: replace $\mathbf{z}_i$ by a value drawn from the distribution $p(z_i | \mathbf{z}_{\setminus i})$.
    - This means we update one coordinate at a time.
  - Repeat procedure either by cycling through all variables or by choosing the next variable.

Slide adapted from Bernt Schiele

Gibbs Sampling

- Example
  - Assume distribution $p(z_1, z_2, z_3)$.
  - Replace $z_1^{(\tau)}$ with new value drawn from $z_1^{(\tau+1)} \sim p(z_1|z_2^{(\tau)}, z_3^{(\tau)})$
  - Replace $z_2^{(\tau)}$ with new value drawn from $z_2^{(\tau+1)} \sim p(z_2|z_1^{(\tau+1)}, z_3^{(\tau)})$
  - Replace $z_3^{(\tau)}$ with new value drawn from $z_3^{(\tau+1)} \sim p(z_3|z_1^{(\tau+1)}, z_2^{(\tau+1)})$
  - And so on...

Slide credit: Bernt Schiele

Gibbs Sampling

- Properties
  - Since the components are unchanged by sampling: $\mathbf{z}^{*}_{\setminus k} = \mathbf{z}_{\setminus k}$.
  - The factor that determines the acceptance probability in the Metropolis-Hastings is thus determined by

$$A(\mathbf{z}^\star, \mathbf{z}) = \frac{p(\mathbf{z}^\star) q_k(\mathbf{z}|\mathbf{z}^\star)}{p(\mathbf{z}) q_k(\mathbf{z}^\star|\mathbf{z})} = \frac{p(z_k^\star|\mathbf{z}^\star_{\setminus k}) p(\mathbf{z}^\star_{\setminus k}) p(z_k|\mathbf{z}^\star_{\setminus k})}{p(z_k|\mathbf{z}_{\setminus k}) p(\mathbf{z}_{\setminus k}) p(z_k^\star|\mathbf{z}_{\setminus k})} = 1$$

  - (we have used $q_k(\mathbf{z}^*|\mathbf{z}) = p(z^*_k|\mathbf{z}_{\setminus k})$ and $p(\mathbf{z}) = p(z_k|\mathbf{z}_{\setminus k})\, p(\mathbf{z}_{\setminus k})$).
  - I.e. we get an algorithm which always accepts!

⇒ If you can compute (and sample from) the conditionals, you can apply Gibbs sampling.

⇒ The algorithm is completely parameter free.

⇒ Can also be applied to subsets of variables.

Slide adapted from Zoubin Ghahramani

Discussion

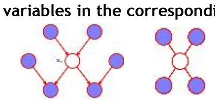
- Gibbs sampling benefits from few free choices and convenient features of conditional distributions:
  - Conditionals with a few discrete settings can be explicitly normalized:

$$p(x_i | \mathbf{x}_{j \neq i}) = \frac{p(x_i, \mathbf{x}_{j\neq i})}{\sum_{x_i'} p(x_i', \mathbf{x}_{j \neq i})}$$

This sum is small and easy.

  - Continuous conditionals are often only univariate.

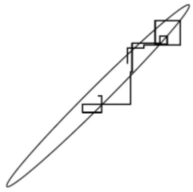
  ⇒ amenable to standard sampling methods.
  - In case of graphical models, the conditional distributions depend only on the variables in the corresponding Markov blankets.



Slide adapted from Iain Murray

Gibbs Sampling

- Example
  - 20 iterations of Gibbs sampling on a bivariate Gaussian.



  - Note: strong correlations can slow down Gibbs sampling.

How Should We Run MCMC?

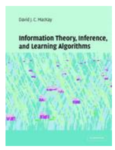
- Arbitrary initialization means starting iterations are bad
  - Discard a "burn-in" period.
- How do we know if we have run for long enough?
  - You don't. That's the problem.
- The samples are not independent
  - Solution 1: Keep only every Mth sample ("thinning").
  - Solution 2: Keep all samples and use the simple Monte Carlo estimator on MCMC samples
    - It is consistent and unbiased if the chain has "burned in".
  - ⇒ Use thinning only if computing $f(\mathbf{x}^{(s)})$ is expensive.
- For opinion on thinning, multiple runs, burn in, etc.
  - Charles J. Geyer, Practical Markov chain Monte Carlo, Statistical Science. 7(4):473{483, 1992. (http://www.jstor.org/stable/2246094)

Summary: Approximate Inference

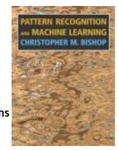
- Exact Bayesian Inference often intractable.
- Rejection and Importance Sampling
  - Generate independent samples.
  - Impractical in high-dimensional state spaces.
- Markov Chain Monte Carlo (MCMC)
  - Simple & effective (even though typically computationally expensive).
  - Scales well with the dimensionality of the state space.
  - Issues of convergence have to be considered carefully.
- Gibbs Sampling
  - Used extensively in practice.
  - Parameter free
  - Requires sampling conditional distributions.

References and Further Reading

- Sampling methods for approximate inference are described in detail in Chapter 11 of Bishop's book.



Christopher M. Bishop
Pattern Recognition and Machine Learning
Springer, 2006



David MacKay
Information Theory, Inference, and Learning Algorithms
Cambridge University Press, 2003

- Another good introduction to Monte Carlo methods can be found in Chapter 29 of MacKay's book (also available online: http://www.inference.phy.cam.ac.uk/mackay/itprnn/book.html)