PAI. Approximate Inference
Anastasia Makarova

ETH Zürich

2.11.2018
Inference

Tree-structured:
- Variable elimination
- Belief propagation

Loopy networks:
- Loopy belief propagation
- Variational inference
- Gibbs sampling (Monte Carlo Sampling)
Stochastic Approximate Inference

- Algorithms that “randomize” to compute marginals as expectations
- In contrast to the deterministic methods, guaranteed to converge to right answer (if wait looong enough..)
- More exact, but slower than deterministic variants
- Also work for continuous distributions
Monte Carlo methods aim to find the expectation of some function $f(x)$ with respect to a probability distribution $p(x)$:

- Draw samples $x_1, \ldots, x_N$
- Compute $\hat{f} = \frac{1}{N} \sum_{i=1}^{N} f(x_i)$

For i.i.d from $p(x)$: $\hat{f}$ is unbiased with variance $\frac{1}{N} \mathbb{E}[(f - \mathbb{E}(f))^2]$

Basic samplings:

- Uniform Sampling
- Rejection Sampling
- Importance Sampling

**Problem:** can be very ineffective, particularly in high dimensions
Problem with Rejection sampling

If proposal distribution $q(x)$ poorly matches our target distribution $p(x)$ – almost always rejects

![Diagram showing the problem with Rejection sampling](image)

**Example**: $d$-dimensional target $p(x) = N(x; \mu, \sigma_p^{2/d})$ and the proposal $q(x) = N(x; \mu, \sigma_q^{2/d})$. Optimal acceptance rate can be accomplished with $k = \frac{\sigma_q}{\sigma_p}$. With $d = 1000$ and $\sigma_q = 1.01\sigma_p$, $k = 1/20000$ resulting in a large waste in samples.
**Markov chains:** random variables $\{x_1, \ldots, x_N\} \ n \in \{1, \ldots, N - 1\}$:

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

**Transitional kernel:** $T(x^n, x^{n+1}) = p(x^{n+1}|x^n)$

**Stationary distribution** $\pi^\infty$: $\pi^\infty T = \pi^\infty$

A given Markov chain may have many stationary distributions.

**Example:** $T(x', x) = \mathbb{I}(x' = x)$: any distribution is invariant.

**Detailed balance:** sufficient condition for ensuring $\pi^\infty$ is stationary: choose $T$ such that

$$\pi^\infty(x) T(x, x') = \pi^\infty(x') T(x', x)$$
MCMC: Metropolis-Hastings

- Aim to sample from $p(x)$ (possibly unnormalized)
- Use easier distribution $q(x^*|x)$ (opposed to $q(x)$ and given as a stochastic matrix) and acceptance test to sample
  
  1. Initialize $x^0$
  2. Burn-in: for $t \in \{1, \ldots, t_0\}$:
     - $x = x^t$
     - $t = t + 1$
     - sample $u \sim \text{Unif}(0, 1)$
     - sample $x^* \sim q(x^*|x)$:
       - if $u \leq A(x^*|x) = \min\{1, \frac{p(x^*)q(x^*|x)}{p(x)q(x|x^*)}\}$: $x^t = x^*$ (transition)
       - else: $x^t = x$ (stay in current state)
  3. Draw samples

- This induces a transition matrix $T(x^*|x) = q(x^*|x)A(x^*|x)$ that satisfies detailed balance $\rightarrow$ after $t_0$ sampling will lead to sampling from stationary $p(x)$
Gibbs sampling: acceptance probability is 1

1. Initializing starting values for $x_1, \ldots, x_n$
2. Do until convergence:
   - randomly pick $x_j$
   - $x \sim P(x_j|x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n)$
   - $x_j = x$

Note: given Markov Blanket of $x_j$:

$$bl(x_j) = pa(j) \cup ch(j) \cup \bigcup_{v \in ch(i)} pa(v)$$

$$P(x_j|x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_n) = P(x_j|bl(x_j))$$
Computing Expectations via GS

One of the MCMC goals - compute the mean of $f(x)$ with respect to $p(x)$:

1. Use Gibbs Sampling to obtain $T$ samples: $\{X^t\}_{t=1}^{T}$
2. Note: $t_0$ samples for burn-in
3. 

$$
\mathbb{E}[f(x)|x_B] \approx \frac{1}{T - t_0} \sum_{t=t_0+1}^{T} f(X^t)
$$
(i) \[ S_1 \rightarrow O \rightarrow \cdots \rightarrow O \rightarrow \]
\[ O \quad O \quad B_t \]

(iii) estimate \( e_i = T \)
we have \( N \) samples \( j = 1 \ldots N \):
\[ u(\cdot) = \frac{1}{N} \sum_{i=1}^{N} u(i) \]
estimation: \[ u = \frac{1}{N} \sum_{j=1}^{N} u(j) \]

(ii) single sample:
\[ (T^5, F^6)^{30}, (T^3, F^5)^{30} \]
pseudo code:
\[ s_1 \sim p(S_1) \]
\[ b_t \sim p(B_i | S_i = s_i) \]
for \( t = 2 \ldots 30 \):
\[ s_t \sim p(S_t | S_{t-1} = s_{t-1}) \]
\[ b_t \sim p(B_t | S_t = s_t) \]
return \((s_1 \ldots s_{30});(b_1 \ldots b_{30})\)
Exam 2016. Sampling

(i) Transition matrix:
\[ T(x, y) = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix} \]

(ii) Stationary distribution:
\[ \pi(S_0) = r \quad \pi(S_1) = 1 - r \]
\[ \begin{pmatrix} r & 1-r \end{pmatrix} \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix} = \begin{pmatrix} r & 1-r \end{pmatrix} \text{ or } rp = (1-r)q \]

(iii) Solve linear alg. eq.:
\[ \begin{pmatrix} 1/3 & 1-p & p \\ 2/3 & q & 1-q \end{pmatrix} \begin{pmatrix} 1/3 \\ 2/3 \end{pmatrix} = \begin{pmatrix} 1/3 \\ 2/3 \end{pmatrix} \]
\[ p = 2q \]
\[ (p, q) = (2k, k) \quad 0 \leq k \leq 0.5 \]
Questions
When we draw a sample $x'$ given $Q(x'|x)$, the transition kernel is $T(x'|x) = Q(x'|x)A(x'|x)$. In Metropolis-Hastings Algorithm, we compute the ratio of importance weight where $A(x'|x) = \min(1, \frac{P(x')Q(x|x')}{P(x)Q(x'|x)})$. Suppose $A(x'|x) < 1$ and $A(x|x') = 1$, we have:

\[
A(x'|x) = \frac{P(x')Q(x|x')}{P(x)Q(x'|x)}
\]

\[
P(x)Q(x'|x)A(x'|x) = P(x')Q(x|x')
\]

\[
P(x)Q(x'|x)A(x'|x) = P(x')Q(x|x')A(x|x')
\]

\[
P(x)T(x'|x) = P(x')T(x|x')
\]