Machine Learning
10-701/15-781, Fall 2011

Infinite Mixture Models

Eric Xin

Lecture 10, October 12, 2011

Clustering

- How to label them?
- How many clusters???
Image Segmentation

- How to segment images?
  - Manual segmentation (very expensive)
  - Algorithm segmentation
    - K-means
    - Statistical mixture models
    - Spectral clustering

- Problems with most existing algorithms
  - Ignore the spatial information
  - Perform the segmentation one image at a time
  - Need to specify the number of segments a priori

Object Recognition and Tracking

<table>
<thead>
<tr>
<th>t=1</th>
<th>t=2</th>
<th>t=3</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1.8, 7.4, 2.3)</td>
<td>(1.9, 9.0, 2.1)</td>
<td>(1.9, 6.1, 2.2)</td>
</tr>
<tr>
<td>(0.9, 5.8, 3.1)</td>
<td>(0.7, 5.1, 3.2)</td>
<td>(0.6, 5.9, 3.2)</td>
</tr>
</tbody>
</table>
The Evolution of Science

Research circles
Research topics
PNAS papers

Clustering as (nonparametric) Bayesian Inference

- Cluster = Mixture
- Each data point is generated from one Mixture

How Many clusters
- Cross Validation
- Data hungry!
- Information theoretic
- AIC, MDL, etc.
- Non-Parametric
  - Manage model uncertainty
  - Integrate over different clustering configurations
  - Base for many interesting extensions
Outline

- Dirichlet Process: From finite mixture to Infinite mixture model
  - The Chinese Restaurant Process
  - Example: heliotype inference

- Intro to Markov Chain Monte Carlo
  - Gibbs sampling

- Dynamic Dirichlet Process
  - The recurrent Chinese Restaurant Process
  - Example: Application: evolutionary clustering of documents

Clustering

- How to label them?
- How many clusters???
Mixture model: The Generative Process

- For data point $x_i$
  - Sample $c_i \sim \text{Multi}(\pi)$
  - Sample $x_i \sim f(\phi_{c_i})$

Number of clusters DOES NOT grow with the data

The Chinese Restaurant Process

- Customers correspond to data points
- Tables correspond to clusters/mixture components
- Dishes correspond to parameter of the mixtures
The Chinese Restaurant Process

\[ P(c_i = k | c_{-i}) = \begin{cases} \frac{1}{1+\alpha} & 0 \\ \frac{\alpha}{1+\alpha} & 1 \\ \frac{1}{2+\alpha} & \frac{1}{2+\alpha} \\ \frac{1}{3+\alpha} & \frac{2}{3+\alpha} \\ \frac{m_i}{i+\alpha-1} & \frac{m_i}{i+\alpha-1} \\ \frac{a}{i+\alpha-1} & \frac{a}{i+\alpha-1} \end{cases} \]

The rich gets richer effect

The Chinese Restaurant Process

CRP defines an exchangeable distribution on partitions over an (infinite) sequence of samples, such a distribution is formally known as the Dirichlet Process (DP)

Generative Process

- For data point \( x_i \)
  - Choose table \( j \propto N_j \) and Sample \( x_i \sim f(\phi_j) \)
  - Choose a new table \( K+1 \propto \alpha \)
    - Sample \( \phi_{K+1} \sim G_0 \) and Sample \( x_i \sim f(\phi_{K+1}) \)
More formally …

- This is a Dirichlet Process Mixture Model
- Three equivalent constructions:

$$G \sim \text{Dirichlet}(\alpha G_0, \ldots, \alpha G_m)$$

Where $G$ is the base measure and $\alpha$ is the scale parameter.

Thus a Dirichlet Process $G$ defines a distribution of distribution.
Stick-breaking Process

\[ G \sim \text{DP}(\alpha, G_0) \]

\[ G = \sum_{k=1}^{\infty} \pi_k \delta(\theta_k) \]

\[ \theta_k \sim G_0 \]

\[ \sum_{k=1}^{\infty} \pi_k = 1 \]

\[ \pi_k = \beta_k \prod_{j=1}^{k-1} (1 - \beta_j) \]

\[ \beta_k \sim \text{Beta}(1, \alpha) \]

Chinese Restaurant Process

\[ P(c_i = k | c_{i-1}) = \begin{cases} 
\frac{1}{1+\alpha} & \text{if } c_{i-1} = \emptyset \\
\frac{\alpha}{1+\alpha} & \text{if } c_{i-1} = \{k\} \\
\frac{1}{2+\alpha} & \text{if } c_{i-1} = \{k, \emptyset\} \\
\frac{2}{3+\alpha} & \text{if } c_{i-1} = \{k, \emptyset, \emptyset\} \\
\frac{m_k}{i+\alpha-1} & \text{if } c_{i-1} = \{k, \emptyset, \ldots, \emptyset\} \\
\alpha \frac{1}{i+\alpha-1} & \text{if } c_{i-1} = \{k, \emptyset, \ldots, \emptyset, \emptyset\} \\
\end{cases} \]

CRP defines an exchangeable distribution on partitions over an (infinite) sequence of samples, such a distribution is formally known as the Dirichlet Process (DP).
Case study: ancestral Inference

Xing et al. 2004

Essentially a clustering problem, but ...

- Better recovery of the ancestors leads to better haplotyping results (because of more accurate grouping of common haplotypes)
- True haplotypes are obtainable with high cost, but they can validate model more subjectively (as opposed to examining saliency of clustering)
- Many other biological/scientific utilities

Background: haplotype ambiguity

Heterozygous diploid individual

Genotype g
pairs of alleles, whose associations to chromosomes are unknown

haplotype h=(h_1, h_2)
possible associations of alleles to chromosomes
A Finite (Mixture of ) Allele Model

- The probability of a genotype $g$:
  \[ p(g) = \sum_{h_1, h_2 \in H} p(h_1, h_2) p(g \mid h_1, h_2) \]

- Standard settings:
  - $|H| = K \ll 2^J$ fixed-sized population haplotype pool
  - $p(h_1, h_2) = p(h_1)p(h_2) = f_1f_2$ Hardy-Weinberg equilibrium

- Problem: $K$? $H$?

---

Example: DP-haplotyper [Xing et al., 2004]

- Clustering human populations

- Inference: Markov Chain Monte Carlo (MCMC)
  - Gibbs sampling
  - Metropolis Hasting
The DP Mixture of Ancestral Haplotypes

- The customers around a table in CRP form a cluster
  - associate a mixture component (i.e., a population haplotype) with a table
  - sample \( \{a, \theta\} \) at each table from a base measure \( G_0 \) to obtain the population haplotype and nucleotide substitution frequency for that component

\[
\{A, \theta\} \quad \{A, \theta\} \quad \{A, \theta\} \quad \{A, \theta\} \quad \{A, \theta\} \quad \ldots
\]

- With \( p(h|\{A, \theta\}) \) and \( p(g|h_1,h_2) \), the CRP yields a posterior distribution on the number of population haplotypes (and on the haplotype configurations and the nucleotide substitution frequencies)

MCMC for Haplotype Inference

- Gibbs sampling for exploring the posterior distribution under the proposed model
  - Integrate out the parameters such as \( \theta \) or \( \lambda \), and sample \( c_{i_x}, a_k \) and \( h_i \)

\[
p(c_{i_x} = k | c_{[-i_x]}, h, a) \propto p(c_{i_x} = k | c_{[-i_x]}) p(h_{i_x} | a_k, h_{[-i_x]}, c)
\]

- Gibbs sampling algorithm: draw samples of each random variable to be sampled given values of all the remaining variables
MCMC for Haplotype Inference

1. Sample $c^{(j)}_i$ from
\[ p(c^{(j)}_i = k | c^{[-i,j]}_i, \mathbf{h}, \mathbf{a}) \]
\[ \propto p(c^{(j)}_i = k | c^{[-i,j]}_i, \mathbf{m}, \mathbf{n}) p(h^{(j)}_i | a_k, c_i, h^{[-i,j]}_i) \]
\[ \propto (m_k^{[-i,j]} + \tau \beta_k) p(h^{(j)}_i | a_k, h^{[-i,j]}_i), \text{ for } k = 1, \ldots, K+1 \]

2. Sample $a_k$ from
\[ p(a_k | c, h) \propto \prod_{i \in \mathcal{I}_k} p(h^{(j)}_i | a_k, c_i, h^{[-i,j]}_i) \]
\[ = \frac{\Gamma(\alpha_h + h_k) \Gamma(\beta_h + f'_k)}{\Gamma(\alpha_h + \beta_h + m_k) (|D| - 1) f'_k} R(\alpha_h, \beta_h) \]

3. Sample $h^{(j)}_i$ from
\[ p(h^{(j)}_i | h^{[-i,j]}_i, c_i, a_i, g) \]

- For DP scale parameter $\alpha$: a vague inverse Gamma prior

Convergence of Ancestral Inference

![Convergence of Ancestral Inference Graph](image-url)
DP vs. Finite Mixture via EM

Outline
- Dirichlet Process: From finite mixture to Infinite mixture model
  - The Chinese Restaurant Process
  - Example: heliotype inference
- Intro to Markov Chain Monte Carlo
  - Gibbs sampling
- Dynamic Dirichlet Process
  - The recurrent Chinese Restaurant Process
  - Example: Application: evolutionary clustering of documents
Monte Carlo methods

- Draw random samples from the desired distribution
- Yield a stochastic representation of a complex distribution
  - marginals and other expectations can be approximated using sample-based averages
    \[
    E[f(x)] = \frac{1}{N} \sum_{i=1}^{N} f(x^{(i)}) \quad X \sim p(x)
    \]
- **Asymptotically** exact and easy to apply to arbitrary models
- **Challenges:**
  - how to draw samples from a given dist. (not all distributions can be trivially sampled)?
  - how to make better use of the samples (not all samples are useful, or equally useful, see an example later)?
  - how to know we’ve sampled enough?

Example: naive sampling

- **Sampling:** Construct samples according to probabilities given in a BN.

Alarm example: (Choose the right sampling sequence)
1) Sampling: \( P(B) = 0.001, 0.999 \) suppose it is false, \( B_0 \). Same for \( E_0 \). \( P(A | B_0, E_0) = 0.001, 0.999 \) suppose it is false...
Example: naive sampling

- Sampling: Construct samples according to probabilities given in a BN.

Alarm example: (Choose the right sampling sequence)
1) Sampling: P(B)=<0.001, 0.999> suppose it is false, B0. Same for E0. P(A|B0, E0)=<0.001, 0.999> suppose it is false...
2) Frequency counting: In the samples right, P(J|A0)=P(J,A0)/P(A0)=<1/9, 8/9>.

3) what if we want to compute P(J|A1)? we have only one sample ...

4) what if we want to compute P(J|B1)? No such sample available!
P(J|A1)=P(J,B1)/P(B1) can not be defined.

For a model with hundreds or more variables, rare events will be very hard to garner enough samples even after a long time or sampling ...
Monte Carlo methods (cond.)

- Direct Sampling
  - We have seen it.
  - Very difficult to populate a high-dimensional state space.

- Rejection Sampling
  - Create samples like direct sampling, only count samples which is consistent with given evidences.

- ....

- Markov chain Monte Carlo (MCMC)

Markov chain Monte Carlo

- Samples are obtained from a Markov chain (of sequentially evolving distributions) whose stationary distribution is the desired $p(x)$.

- Construct a Markov chain whose stationary distribution is the target density $= P(X|e)$.

- Run for $T$ samples (burn-in time) until the chain converges/mixes/reaches stationary distribution.

- Then collect $M$ (correlated) samples $x_m$.

- Key issues:
  - Designing proposals so that the chain mixes rapidly.
  - Diagnosing convergence.
Gibbs sampling

- Gibbs sampling is an MCMC algorithm that is especially appropriate for inference in graphical models.

- The procedure
  - We have variable set $X = \{x_1, x_2, x_3, \ldots, x_N\}$ for a GM.
  - At each step, one of the variables $X_i$ is selected (at random or according to some fixed sequences), and denote the remaining variables as $X_{-i}$, and its current value as $x_{-i}(t-1)$.
  - Using the "alarm network" as an example, say at time $t$ we choose $X_E$, and we denote the current value assignments of the remaining variables, $X_{-E}$, obtained from previous samples, as $x_{-E}(t)$.
  - The conditional distribution $p(X_i | x_{-i}(t-1))$ is computed.
  - A value $x_i(t)$ is sampled from this distribution.
  - The sample $x_i(t)$ replaces the previous sampled value of $X_i$ in $x_{-E}(t)$.

Why Gibbs Sampling is Simple

- Markov-Blanket
  - A variable is independent from others, given its parents, children, and children’s parents.
  - d-separation.
  \[ p(X_i | X_{-i}) = p(X_i | MB(X_i)) \]

- Gibbs sampling
  - Create a random sample.
  - Every step, choose one variable and sample it by $p(X_i | MB(X))$ based on previous sample.

$MB(A) = \{B, E, J, M\}$

$MB(E) = \{A, B\}$
Example: alarm network again

To calculate $P(J|B1,M1)$

Choose $\{B1,E0,A1,M1,J1\}$ as a start

Evidences are $B1$, $M1$, variables are $A$, $E$, $J$.

Choose next variable as $A$

Sample $A$ by $P(A|MB(A))=P(A|B1, E0, M1, J1)$ suppose to be false.

$\{B1, E0, A0, M1, J1\}$

Choose next random variable as $E$, sample $E\sim P(E|B1,A0)$

...
Example:

\[
P(B_1 \mid M_1)
\]

Example

\[
P(E_{11} \mid E_1, M_2)
\]
Example

\[
P(J_1 | B_1, M_1) = 0.90 \\
P(J_1 | E_1, M_0) = 0.14 \\
P(E_1 | J_1) = 0.01 \\
P(E_1 | M_1) = 0.04 \\
P(E_1 | M_1, J_1) = 0.17
\]

Outline

- Dirichlet Process: From finite mixture to Infinite mixture model
  - The Chinese Restaurant Process
  - Example: heliotype inference

- Intro to Markov Chain Monte Carlo
  - Gibbs sampling

- Dynamic Dirichlet Process
  - The recurrent Chinese Restaurant Process
  - Example: Application: evolutionary clustering of documents
Evolutionary Clustering

- Adapts the number of mixture components over time
  - Mixture components can die out
  - New mixture components are born at any time
  - Retained mixture components parameters evolve according to a Markovian dynamics

Temporal DPM [Ahmed and Xing 2008]

- The Recurrent Chinese Restaurant Process
  - The restaurant operates in epochs
  - The restaurant is closed at the end of each epoch
  - The state of the restaurant at time epoch $t$ depends on that at time epoch $t-1$
    - Can be extended to higher-order dependencies.
Dish eaten at table 3 at time epoch 1
OR the parameters of cluster 3 at time epoch 1

Generative Process
- Customers at time $T=1$ are seated as before:
  - Choose table $j \sim N_{j,1}$ and Sample $x_i \sim f(\phi_{j,1})$
  - Choose a new table $K+1 \sim \alpha$
  - Sample $\phi_{K+1,1} \sim G_0$ and Sample $x_i \sim f(\phi_{K+1,1})$
Sample $\phi_{1.2} \sim P(.| \phi_{1.1})$
At the end of epoch 2

Newly born cluster

Died out cluster

And so on......
Temporal DPM

- Can be extended to model higher-order dependencies
- Can decay dependencies over time
  - Pseudo-counts for table $k$ at time $t$ is

$$
\sum_{w=1}^{W} \left( e^{-w} N_{k,t-w} \right)
$$
TDPM Generative Power

DPM
\( W = T \)
\( \lambda = \infty \)

TDPM
\( W = 4 \)
\( \lambda = .4 \)

Independent DPMs
\( W = 0 \)
\( \lambda = \text{? (any)} \)
Results: NIPS 12

- Building a simple dynamic topic model
- Chain dynamics is as before
- Emission model for document $x_{k,t}$ is:
  - Project $\phi_{k,t}$ over the simplex
  - Sample $x_{k,t} | \phi_{k,t} \sim \text{Multinomial}(\phi_{k,t})$
- Unlike LDA here a document belongs to one topic
- Use this model to analyze NIPS12 corpus
  - Proceeding of NIPS conference 1987-1999
The Big Picture

K-means & mixture

Fixed-dimensions Dynamic

DPM

TDPM

Appendix:
Theory of MCMC (optional)

- **Definition:** Markov Chains
  - Given an n-dimensional state space
  - Random vector $X = (x_1, ..., x_n)$
  - $x^{(0)} = x$ at time-step $t$
  - $x^{(0)}$ transitions to $x^{(t)}$ with prob
    $P(x^{(t+1)} | x^{(0)}, ..., x^{(t)}) = T(x^{(t)} | x^{(t-1)}) = T(x^{(t)} \rightarrow x^{(t+1)})$

- **Homogenous:** chain determined by state $x^{(0)}$, fixed *transition kernel* $T$ (rows sum to 1)

- **Equilibrium:** $\pi(x)$ is a **stationary (equilibrium) distribution** if
  $\pi(x') = \sum_x \pi(x) T(x \rightarrow x')$.
  i.e., is a left eigenvector of the transition matrix $\pi T = \pi T$.

Markov Chains

- An MC is **irreducible** if transition graph connected
- An MC is **aperiodic** if it is not trapped in cycles
- An MC is **ergodic** (regular) if you can get from state $x$ to $x'$ in a finite number of steps.
- **Detailed balance:** $\text{prob}(x^{(t)} \rightarrow x^{(t-1)}) = \text{prob}(x^{(t-1)} \rightarrow x^{(t)})$

  $p(x^{(t)}) T(x^{(t-1)} | x^{(t)}) = p(x^{(t-1)}) T(x^{(t)} | x^{(t-1)})$

  summing over $x^{(t-1)}$

  $p(x^{(t)}) = \sum_{x^{(t-1)}} p(x^{(t-1)}) T(x^{(t)} | x^{(t-1)})$

- Detailed bal $\rightarrow$ stationary dist exists
MCMC Via Metropolis-Hastings

- Treat the target distribution as stationary distribution
- Sample from an easier proposal distribution, followed by an acceptance test
- This induces a transition matrix that satisfies detailed balance
  - MH proposes moves according to $Q(x' | x)$ and accepts samples with probability $A(x' | x)$.
  - The induced transition matrix is $T(x \rightarrow x') = Q(x' | x) A(x' | x)$
  - Detailed balance means
    $$\pi(x) Q(x' | x) A(x' | x) = \pi(x') Q(x | x') A(x | x')$$
  - Hence the acceptance ratio is
    $$A(x' | x) = \min \left(1, \frac{\pi(x') Q(x | x')}{\pi(x) Q(x' | x)} \right)$$

Gibbs sampling

- Gibbs sampling is a special case of MH
- The transition matrix updates each node one at a time using the following proposal:
  $$Q((x_i, x_{\neq i}) \rightarrow (x_i', x_{\neq i})) = p(x_i' | x_{\neq i})$$
- This is efficient since for two reasons
  - It leads to samples that is always accepted
    $$A((x_i, x_{\neq i}) \rightarrow (x_i', x_{\neq i})) = \min \left(1, \frac{p(x_i', x_{\neq i})}{p(x_i, x_{\neq i})} \frac{Q((x_i, x_{\neq i}) \rightarrow (x_i', x_{\neq i}))}{Q((x_i, x_{\neq i}) \rightarrow (x_i, x_{\neq i}))} \right)$$
    $$= \min \left(1, \frac{p(x_i', x_{\neq i})}{p(x_i, x_{\neq i})} \frac{Q((x_i, x_{\neq i}) \rightarrow (x_i', x_{\neq i}))}{Q((x_i, x_{\neq i}) \rightarrow (x_i, x_{\neq i}))} \right) = \min(1, 1)$$
    Thus
    $$T((x_i, x_{\neq i}) \rightarrow (x_i', x_{\neq i})) = p(x_i' | x_{\neq i})$$
  - It is efficient since $p(x_i' | x_{\neq i})$ only depends on the values in $x_i$'s Markov blanket

© Eric Xing @ CMU, 2008-2011