6.864: Lecture 5 (September 22nd, 2005)

The EM Algorithm

Overview

- The EM algorithm in general form
- The EM algorithm for hidden markov models (brute force)
- The EM algorithm for hidden markov models (dynamic programming)

An Experiment/Some Intuition

- I have three coins in my pocket,

  Coin 0 has probability λ of heads;
  Coin 1 has probability p_1 of heads;
  Coin 2 has probability p_2 of heads

- For each trial I do the following:

  First I toss Coin 0
  If Coin 0 turns up **heads**, I toss **coin 1** three times
  If Coin 0 turns up **tails**, I toss **coin 2** three times

  I don't tell you whether Coin 0 came up heads or tails,
  or whether Coin 1 or 2 was tossed three times,
  but I do tell you how many heads/tails are seen at each trial

- you see the following sequence:

$$\langle HHH \rangle, \langle TTT \rangle, \langle HHH \rangle, \langle TTT \rangle, \langle HHH \rangle$$

What would you estimate as the values for λ, p_1 and p_2?

Maximum Likelihood Estimation

- We have data points $x_1, x_2, \ldots x_n$ drawn from some (finite or countable) set $\mathcal{X}$

- We have a parameter vector Θ

- We have a parameter space Ω

- We have a distribution $P(x \mid \Theta)$ for any $\Theta \in \Omega$, such that

$$\sum_{x \in \mathcal{X}} P(x \mid \Theta) = 1 \text{ and } P(x \mid \Theta) \geq 0 \text{ for all x}$$

- We assume that our data points $x_1, x_2, \ldots x_n$ are drawn at random (independently, identically distributed) from a distribution $P(x \mid \Theta^*)$ for some $\Theta^* \in \Omega$

Log-Likelihood

- We have data points $x_1, x_2, \ldots x_n$ drawn from some (finite or countable) set $\mathcal{X}$
- We have a parameter vector Θ, and a parameter space Ω
- We have a distribution $P(x \mid \Theta)$ for any $\Theta \in \Omega$
- The likelihood is

$$Likelihood(\Theta) = P(x_1, x_2, \ldots x_n \mid \Theta) = \prod_{i=1}^{n} P(x_i \mid \Theta)$$

- The log-likelihood is

$$L(\Theta) = \log Likelihood(\Theta) = \sum_{i=1}^{n} \log P(x_i \mid \Theta)$$

A First Example: Coin Tossing

- $\mathcal{X} = \{\texttt{H},\texttt{T}\}$. Our data points $x_1, x_2, \ldots x_n$ are a sequence of heads and tails, e.g.

$$\texttt{HHTTHHHTHH}$$

- Parameter vector Θ is a single parameter, i.e., the probability of coin coming up heads

- Parameter space $\Omega = [0, 1]$

- Distribution $P(x \mid \Theta)$ is defined as

$$P(x \mid \Theta) = \begin{cases} \Theta & \text{If } x = \texttt{H} \\ 1 - \Theta & \text{If } x = \texttt{T} \end{cases}$$

Maximum Likelihood Estimation

- Given a sample $x_1, x_2, \ldots x_n$, choose

$$\Theta_{ML} = \text{argmax}_{\Theta \in \Omega} L(\Theta) = \text{argmax}_{\Theta \in \Omega} \sum_i \log P(x_i \mid \Theta)$$

- For example, take the coin example:
  say $x_1 \ldots x_n$ has $Count(H)$ heads, and $(n - Count(H))$ tails

  $\Rightarrow$

$$\begin{aligned} L(\Theta) &= \log \left(\Theta^{Count(H)} \times (1 - \Theta)^{n - Count(H)} \right) \\ &= Count(H) \log \Theta + (n - Count(H)) \log (1 - \Theta) \end{aligned}$$

- We now have

$$\Theta_{ML} = \frac{Count(H)}{n}$$

A Second Example: Probabilistic Context-Free Grammars

- $\mathcal{X}$ is the set of all parse trees generated by the underlying context-free grammar. Our sample is n trees $T_1 \ldots T_n$ such that each $T_i \in \mathcal{X}$.

- R is the set of rules in the context free grammar
  N is the set of non-terminals in the grammar

- Θ_r for $r \in R$ is the parameter for rule r

- Let $R(\alpha) \subset R$ be the rules of the form $\alpha \rightarrow \beta$ for some β

- The parameter space Ω is the set of $\Theta \in [0, 1]^{|R|}$ such that

$$\text{for all } \alpha \in N \sum_{r \in R(\alpha)} \Theta_r = 1$$

- We have

$$P(T \mid \Theta) = \prod_{r \in R} \Theta_r^{Count(T,r)}$$

where $Count(T, r)$ is the number of times rule r is seen in the tree T

$$\Rightarrow \quad \log P(T \mid \Theta) = \sum_{r \in R} Count(T, r) \log \Theta_r$$

Maximum Likelihood Estimation for PCFGs

- We have

$$\log P(T \mid \Theta) = \sum_{r \in R} Count(T, r) \log \Theta_r$$

where $Count(T, r)$ is the number of times rule r is seen in the tree T

- And,

$$L(\Theta) = \sum_i \log P(T_i \mid \Theta) = \sum_i \sum_{r \in R} Count(T_i, r) \log \Theta_r$$

- Solving $\Theta_{ML} = \text{argmax}_{\Theta \in \Omega} L(\Theta)$ gives

$$\Theta_r = \frac{\sum_i Count(T_i, r)}{\sum_i \sum_{s \in R(\alpha)} Count(T_i, s)}$$

where r is of the form $\alpha \rightarrow \beta$ for some β

Multinomial Distributions

- $\mathcal{X}$ is a finite set, e.g., $\mathcal{X} = \{\texttt{dog}, \texttt{cat}, \texttt{the}, \texttt{saw}\}$

- Our sample $x_1, x_2, \ldots x_n$ is drawn from $\mathcal{X}$
  e.g., $x_1, x_2, x_3 = \texttt{dog}, \texttt{the}, \texttt{saw}$

- The parameter Θ is a vector in $\mathbb{R}^m$ where $m = |\mathcal{X}|$
  e.g., $\Theta_1 = P(dog)$, $\Theta_2 = P(cat)$, $\Theta_3 = P(the)$, $\Theta_4 = P(saw)$

- The parameter space is

$$\Omega = \{\Theta : \sum_{i=1}^{m} \Theta_i = 1 \text{ and } \forall i, \Theta_i \geq 0\}$$

- If our sample is $x_1, x_2, x_3 = \texttt{dog}, \texttt{the}, \texttt{saw}$, then

$$L(\Theta) = \log P(x_1, x_2, x_3 = \texttt{dog}, \texttt{the}, \texttt{saw}) = \log \Theta_1 + \log \Theta_3 + \log \Theta_4$$

Models with Hidden Variables

- Now say we have two sets $\mathcal{X}$ and $\mathcal{Y}$, and a joint distribution $P(x, y \mid \Theta)$

- If we had **fully observed data**, (x_i, y_i) pairs, then

$$L(\Theta) = \sum_i \log P(x_i, y_i \mid \Theta)$$

- If we have **partially observed data**, x_i examples, then

$$\begin{aligned} L(\Theta) &= \sum_i \log P(x_i \mid \Theta) \\ &= \sum_i \log \sum_{y \in \mathcal{Y}} P(x_i, y \mid \Theta) \end{aligned}$$

- The **EM (Expectation Maximization) algorithm** is a method for finding

$$\Theta_{ML} = \text{argmax}_{\Theta} \sum_{i} \log \sum_{y \in \mathcal{Y}} P(x_i, y \mid \Theta)$$

The Three Coins Example

- e.g., in the three coins example:
  $\mathcal{Y} = \{\mathtt{H}, \mathtt{T}\}$
  $\mathcal{X} = \{\mathtt{HHH}, \mathtt{TTT}, \mathtt{HTT}, \mathtt{THH}, \mathtt{HHT}, \mathtt{TTH}, \mathtt{HTH}, \mathtt{THT}\}$
  $\Theta = \{\lambda, p_1, p_2\}$

- and

$$P(x, y \mid \Theta) = P(y \mid \Theta)P(x \mid y, \Theta)$$

where

$$P(y \mid \Theta) = \begin{cases} \lambda & \text{If } y = \mathsf{H} \\ 1 - \lambda & \text{If } y = \mathsf{T} \end{cases}$$

and

$$P(x \mid y, \Theta) = \begin{cases} p_1^h (1 - p_1)^t & \text{If } y = \mathsf{H} \\ p_2^h (1 - p_2)^t & \text{If } y = \mathsf{T} \end{cases}$$

where h = number of heads in x, t = number of tails in x

The Three Coins Example

- Various probabilities can be calculated, for example:

$$P(x = \mathsf{THT}, y = \mathsf{H} \mid \Theta) = \lambda p_1 (1 - p_1)^2$$

The Three Coins Example

- Various probabilities can be calculated, for example:

$$P(x = \mathsf{THT}, y = \mathsf{H} \mid \Theta) \quad = \quad \lambda p_1 (1 - p_1)^2$$

$$P(x = \mathsf{THT}, y = \mathsf{T} \mid \Theta) \quad = \quad (1 - \lambda) p_2 (1 - p_2)^2$$

The Three Coins Example

- Various probabilities can be calculated, for example:

$$P(x = \mathsf{THT}, y = \mathsf{H} \mid \Theta) = \lambda p_1 (1 - p_1)^2$$

$$P(x = \mathsf{THT}, y = \mathsf{T} \mid \Theta) = (1 - \lambda) p_2 (1 - p_2)^2$$

$$\begin{aligned} P(x = \mathsf{THT} \mid \Theta) &= P(x = \mathsf{THT}, y = \mathsf{H} \mid \Theta) \\ &\quad + P(x = \mathsf{THT}, y = \mathsf{T} \mid \Theta) \\ &= \lambda p_1 (1 - p_1)^2 + (1 - \lambda) p_2 (1 - p_2)^2 \end{aligned}$$

The Three Coins Example

- Various probabilities can be calculated, for example:

$$P(x = \mathsf{THT}, y = \mathsf{H} \mid \Theta) = \lambda p_1 (1 - p_1)^2$$

$$P(x = \mathsf{THT}, y = \mathsf{T} \mid \Theta) = (1 - \lambda) p_2 (1 - p_2)^2$$

$$\begin{aligned} P(x = \mathsf{THT} \mid \Theta) &= P(x = \mathsf{THT}, y = \mathsf{H} \mid \Theta) \\ &\quad + P(x = \mathsf{THT}, y = \mathsf{T} \mid \Theta) \\ &= \lambda p_1 (1 - p_1)^2 + (1 - \lambda) p_2 (1 - p_2)^2 \end{aligned}$$

$$\begin{aligned} P(y = \mathsf{H} \mid x = \mathsf{THT}, \Theta) &= \frac{P(x = \mathsf{THT}, y = \mathsf{H} \mid \Theta)}{P(x = \mathsf{THT} \mid \Theta)} \\ &= \frac{\lambda p_1 (1 - p_1)^2}{\lambda p_1 (1 - p_1)^2 + (1 - \lambda) p_2 (1 - p_2)^2} \end{aligned}$$

The Three Coins Example

- Fully observed data might look like:

$$(\langle HHH \rangle, H), (\langle TTT \rangle, T), (\langle HHH \rangle, H), (\langle TTT \rangle, T), (\langle HHH \rangle, H)$$

- In this case maximum likelihood estimates are:

$$\lambda = \frac{3}{5}$$

$$p_1 = \frac{9}{9}$$

$$p_2 = \frac{0}{6}$$

The Three Coins Example

- Partially observed data might look like:

$$\langle HHH \rangle, \langle TTT \rangle, \langle HHH \rangle, \langle TTT \rangle, \langle HHH \rangle$$

- How do we find the maximum likelihood parameters?

The Three Coins Example

- Partially observed data might look like:

$$\langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle$$

- If current parameters are λ, p_1, p_2

$$\begin{aligned} P(y = \mathsf{H} \mid x = \langle \mathsf{HHH}\rangle) &= \frac{P(\langle \mathsf{HHH}\rangle, \mathsf{H})}{P(\langle \mathsf{HHH}\rangle, \mathsf{H}) + P(\langle \mathsf{HHH}\rangle, \mathsf{T})} \\ &= \frac{\lambda p_1^3}{\lambda p_1^3 + (1-\lambda)p_2^3} \end{aligned}$$

$$\begin{aligned} P(y = \mathsf{H} \mid x = \langle \mathsf{TTT}\rangle) &= \frac{P(\langle \mathsf{TTT}\rangle, \mathsf{H})}{P(\langle \mathsf{TTT}\rangle, \mathsf{H}) + P(\langle \mathsf{TTT}\rangle, \mathsf{T})} \\ &= \frac{\lambda(1-p_1)^3}{\lambda(1-p_1)^3 + (1-\lambda)(1-p_2)^3} \end{aligned}$$

The Three Coins Example

- If current parameters are λ, p_1, p_2

$$P(y = \mathsf{H} \mid x = \langle \mathsf{HHH} \rangle) \;\; = \;\; \frac{\lambda p_1^3}{\lambda p_1^3 + (1-\lambda) p_2^3}$$

$$P(y = \mathsf{H} \mid x = \langle \mathsf{TTT} \rangle) \;\; = \;\; \frac{\lambda (1-p_1)^3}{\lambda (1-p_1)^3 + (1-\lambda)(1-p_2)^3}$$

- If $\lambda = 0.3, p_1 = 0.3, p_2 = 0.6$:

$$P(y = \mathsf{H} \mid x = \langle \mathsf{HHH} \rangle) \;\; = \;\; 0.0508$$

$$P(y = \mathsf{H} \mid x = \langle \mathsf{TTT} \rangle) \;\; = \;\; 0.6967$$

The Three Coins Example

- After filling in hidden variables for each example, partially observed data might look like:

$$
\begin{array}{ll}
(\langle \mathsf{HHH} \rangle, H) & P(y = \mathsf{H} \mid \mathsf{HHH}) = 0.0508 \\
(\langle \mathsf{HHH} \rangle, T) & P(y = \mathsf{T} \mid \mathsf{HHH}) = 0.9492 \\
(\langle \mathsf{TTT} \rangle, H) & P(y = \mathsf{H} \mid \mathsf{TTT}) = 0.6967 \\
(\langle \mathsf{TTT} \rangle, T) & P(y = \mathsf{T} \mid \mathsf{TTT}) = 0.3033 \\
(\langle \mathsf{HHH} \rangle, H) & P(y = \mathsf{H} \mid \mathsf{HHH}) = 0.0508 \\
(\langle \mathsf{HHH} \rangle, T) & P(y = \mathsf{T} \mid \mathsf{HHH}) = 0.9492 \\
(\langle \mathsf{TTT} \rangle, H) & P(y = \mathsf{H} \mid \mathsf{TTT}) = 0.6967 \\
(\langle \mathsf{TTT} \rangle, T) & P(y = \mathsf{T} \mid \mathsf{TTT}) = 0.3033 \\
(\langle \mathsf{HHH} \rangle, H) & P(y = \mathsf{H} \mid \mathsf{HHH}) = 0.0508 \\
(\langle \mathsf{HHH} \rangle, T) & P(y = \mathsf{T} \mid \mathsf{HHH}) = 0.9492
\end{array}
$$

The Three Coins Example

- New Estimates:

$$
\begin{array}{ll}
(\langle \mathsf{HHH} \rangle, H) & P(y = \mathsf{H} \mid \mathsf{HHH}) = 0.0508 \\
(\langle \mathsf{HHH} \rangle, T) & P(y = \mathsf{T} \mid \mathsf{HHH}) = 0.9492 \\
(\langle \mathsf{TTT} \rangle, H) & P(y = \mathsf{H} \mid \mathsf{TTT}) = 0.6967 \\
(\langle \mathsf{TTT} \rangle, T) & P(y = \mathsf{T} \mid \mathsf{TTT}) = 0.3033 \\
\ldots &
\end{array}
$$

$$\lambda = \frac{3 \times 0.0508 + 2 \times 0.6967}{5} = 0.3092$$

$$p_1 = \frac{3 \times 3 \times 0.0508 + 0 \times 2 \times 0.6967}{3 \times 3 \times 0.0508 + 3 \times 2 \times 0.6967} = 0.0987$$

$$p_2 = \frac{3 \times 3 \times 0.9492 + 0 \times 2 \times 0.3033}{3 \times 3 \times 0.9492 + 3 \times 2 \times 0.3033} = 0.8244$$

The Three Coins Example: Summary

- Begin with parameters $\lambda = 0.3, p_1 = 0.3, p_2 = 0.6$
- Fill in hidden variables, using

$$P(y = \mathsf{H} \mid x = \langle \mathsf{HHH} \rangle) = 0.0508$$

$$P(y = \mathsf{H} \mid x = \langle \mathsf{TTT} \rangle) = 0.6967$$

- Re-estimate parameters to be $\lambda = 0.3092, p_1 = 0.0987, p_2 = 0.8244$

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.3000	0.6000	0.0508	0.6967	0.0508	0.6967
1	0.3738	0.0680	0.7578	0.0004	0.9714	0.0004	0.9714
2	0.4859	0.0004	0.9722	0.0000	1.0000	0.0000	1.0000
3	0.5000	0.0000	1.0000	0.0000	1.0000	0.0000	1.0000

The coin example for $\mathbf{y} = \{\langle HHH \rangle, \langle TTT \rangle, \langle HHH \rangle, \langle TTT \rangle\}$. The solution that EM reaches is intuitively correct: the coin-tosser has two coins, one which always shows up heads, the other which always shows tails, and is picking between them with equal probability ($\lambda = 0.5$). The posterior probabilities $\tilde{p}_i$ show that we are certain that coin 1 (tail-biased) generated y_2 and y_4, whereas coin 2 generated y_1 and y_3.

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$	$\tilde{p}_5$
0	0.3000	0.3000	0.6000	0.0508	0.6967	0.0508	0.6967	0.0508
1	0.3092	0.0987	0.8244	0.0008	0.9837	0.0008	0.9837	0.0008
2	0.3940	0.0012	0.9893	0.0000	1.0000	0.0000	1.0000	0.0000
3	0.4000	0.0000	1.0000	0.0000	1.0000	0.0000	1.0000	0.0000

The coin example for $\{\langle HHH \rangle, \langle TTT \rangle, \langle HHH \rangle, \langle TTT \rangle, \langle HHH \rangle\}$. λ is now 0.4, indicating that the coin-tosser has probability 0.4 of selecting the tail-biased coin.

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.3000	0.6000	0.1579	0.6967	0.0508	0.6967
1	0.4005	0.0974	0.6300	0.0375	0.9065	0.0025	0.9065
2	0.4632	0.0148	0.7635	0.0014	0.9842	0.0000	0.9842
3	0.4924	0.0005	0.8205	0.0000	0.9941	0.0000	0.9941
4	0.4970	0.0000	0.8284	0.0000	0.9949	0.0000	0.9949

The coin example for $\mathbf{y} = \{\langle HHT \rangle, \langle TTT \rangle, \langle HHH \rangle, \langle TTT \rangle\}$. EM selects a tails-only coin, and a coin which is heavily heads-biased ($p_2 = 0.8284$). It's certain that y_1 and y_3 were generated by coin 2, as they contain heads. y_2 and y_4 could have been generated by either coin, but coin 1 is far more likely.

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.7000	0.7000	0.3000	0.3000	0.3000	0.3000
1	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
2	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
3	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
4	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
5	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000
6	0.3000	0.5000	0.5000	0.3000	0.3000	0.3000	0.3000

The coin example for $\mathbf{y} = \{\langle HHH \rangle, \langle TTT \rangle, \langle HHH \rangle, \langle TTT \rangle\}$, with p_1 and p_2 initialised to the same value. EM is stuck at a saddle point

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.7001	0.7000	0.3001	0.2998	0.3001	0.2998
1	0.2999	0.5003	0.4999	0.3004	0.2995	0.3004	0.2995
2	0.2999	0.5008	0.4997	0.3013	0.2986	0.3013	0.2986
3	0.2999	0.5023	0.4990	0.3040	0.2959	0.3040	0.2959
4	0.3000	0.5068	0.4971	0.3122	0.2879	0.3122	0.2879
5	0.3000	0.5202	0.4913	0.3373	0.2645	0.3373	0.2645
6	0.3009	0.5605	0.4740	0.4157	0.2007	0.4157	0.2007
7	0.3082	0.6744	0.4223	0.6447	0.0739	0.6447	0.0739
8	0.3593	0.8972	0.2773	0.9500	0.0016	0.9500	0.0016
9	0.4758	0.9983	0.0477	0.9999	0.0000	0.9999	0.0000
10	0.4999	1.0000	0.0001	1.0000	0.0000	1.0000	0.0000
11	0.5000	1.0000	0.0000	1.0000	0.0000	1.0000	0.0000

The coin example for $\mathbf{y} = \{\langle HHH\rangle, \langle TTT\rangle, \langle HHH\rangle, \langle TTT\rangle\}$. If we initialise p_1 and p_2 to be a small amount away from the saddle point $p_1 = p_2$, the algorithm diverges from the saddle point and eventually reaches the global maximum.

Iteration	λ	p_1	p_2	$\tilde{p}_1$	$\tilde{p}_2$	$\tilde{p}_3$	$\tilde{p}_4$
0	0.3000	0.6999	0.7000	0.2999	0.3002	0.2999	0.3002
1	0.3001	0.4998	0.5001	0.2996	0.3005	0.2996	0.3005
2	0.3001	0.4993	0.5003	0.2987	0.3014	0.2987	0.3014
3	0.3001	0.4978	0.5010	0.2960	0.3041	0.2960	0.3041
4	0.3001	0.4933	0.5029	0.2880	0.3123	0.2880	0.3123
5	0.3002	0.4798	0.5087	0.2646	0.3374	0.2646	0.3374
6	0.3010	0.4396	0.5260	0.2008	0.4158	0.2008	0.4158
7	0.3083	0.3257	0.5777	0.0739	0.6448	0.0739	0.6448
8	0.3594	0.1029	0.7228	0.0016	0.9500	0.0016	0.9500
9	0.4758	0.0017	0.9523	0.0000	0.9999	0.0000	0.9999
10	0.4999	0.0000	0.9999	0.0000	1.0000	0.0000	1.0000
11	0.5000	0.0000	1.0000	0.0000	1.0000	0.0000	1.0000

The coin example for $\mathbf{y} = \{\langle HHH \rangle, \langle TTT \rangle, \langle HHH \rangle, \langle TTT \rangle\}$. If we initialise p_1 and p_2 to be a small amount away from the saddle point $p_1 = p_2$, the algorithm diverges from the saddle point and eventually reaches the global maximum.

The EM Algorithm

- Θ^t is the parameter vector at t'th iteration

- Choose Θ^0 (at random, or using various heuristics)

- Iterative procedure is defined as

$$\Theta^t = \text{argmax}_\Theta Q(\Theta, \Theta^{t-1})$$

where

$$Q(\Theta, \Theta^{t-1}) = \sum_i \sum_{y \in \mathcal{Y}} P(y \mid x_i, \Theta^{t-1}) \log P(x_i, y \mid \Theta)$$

The EM Algorithm

- Iterative procedure is defined as $\Theta^t = \text{argmax}_\Theta Q(\Theta, \Theta^{t-1})$, where

$$Q(\Theta, \Theta^{t-1}) = \sum_i \sum_{y \in \mathcal{Y}} P(y \mid x_i, \Theta^{t-1}) \log P(x_i, y \mid \Theta)$$

- Key points:
  - Intuition: fill in hidden variables y according to $P(y \mid x_i, \Theta)$
  - EM is guaranteed to converge to a local maximum, or saddle-point, of the likelihood function
  - In general, if

    $$\text{argmax}_\Theta \sum_i \log P(x_i, y_i \mid \Theta)$$

    has a simple (analytic) solution, then

    $$\text{argmax}_\Theta \sum_i \sum_y P(y \mid x_i, \Theta) \log P(x_i, y \mid \Theta)$$

    also has a simple (analytic) solution.

Overview

- The EM algorithm in general form
- The EM algorithm for hidden markov models (brute force)
- The EM algorithm for hidden markov models (dynamic programming)

The Structure of Hidden Markov Models

- Have N states, states $1 \dots N$
- Without loss of generality, take N to be the final or stop state
- Have an alphabet K. For example $K = \{a, b\}$
- Parameter π_i for $i = 1 \dots N$ is probability of starting in state i
- Parameter $a_{i,j}$ for $i = 1 \dots (N-1)$, and $j = 1 \dots N$ is probability of state j following state i
- Parameter $b_i(o)$ for $i = 1 \dots (N-1)$, and $o \in K$ is probability of state i emitting symbol o

An Example

- Take $N = 3$ states. States are $\{1, 2, 3\}$. Final state is state 3.

- Alphabet $K = \{the, dog\}$.

- Distribution over initial state is $\pi_1 = 1.0, \pi_2 = 0, \pi_3 = 0$.

- Parameters $a_{i,j}$ are

	j=1	j=2	j=3
i=1	0.5	0.5	0
i=2	0	0.5	0.5

- Parameters $b_i(o)$ are

	o=the	o=dog
i=1	0.9	0.1
i=2	0.1	0.9

A Generative Process

- Pick the start state s_1 to be state i for $i = 1 \ldots N$ with probability π_i.

- Set $t = 1$

- Repeat while current state s_t is not the stop state (N):
  - Emit a symbol $o_t \in K$ with probability $b_{s_t}(o_t)$
  - Pick the next state s_{t+1} as state j with probability $a_{s_t,j}$.
  - $t = t + 1$

Probabilities Over Sequences

- An **output sequence** is a sequence of observations $o_1 \ldots o_T$ where each $o_i \in K$
  e.g. the dog the dog dog the
- A **state sequence** is a sequence of states $s_1 \ldots s_T$ where each $s_i \in \{1 \ldots N\}$
  e.g. 1 2 1 2 2 1
- HMM defines a probability for each state/output sequence pair

e.g. the/1 dog/2 the/1 dog/2 the/2 dog/1 has probability

$$\pi_1\ b_1(\text{the})\ a_{1,2}\ b_2(\text{dog})\ a_{2,1}\ b_1(\text{the})\ a_{1,2}\ b_2(\text{dog})\ a_{2,2}\ b_2(\text{the})\ a_{2,1}\ b_1(\text{dog}) a_{1,3}$$

Formally:

$$P(s_1 \ldots s_T, o_1 \ldots o_T) = \pi_{s_1} \times \left(\prod_{i=2}^{T} P(s_i \mid s_{i-1}) \right) \times \left(\prod_{i=1}^{T} P(o_i \mid s_i) \right) \times P(N \mid s_T)$$

A Hidden Variable Problem

- We have an HMM with $N = 3$, $K = \{e, f, g, h\}$
- We see the following **output sequences** in training data

e	g
e	h
f	h
f	g

- How would you choose the parameter values for π_i, $a_{i,j}$, and $b_i(o)$?

Another Hidden Variable Problem

- We have an HMM with $N = 3$, $K = \{e, f, g, h\}$
- We see the following **output sequences** in training data

  e g h
  e h
  f h g
  f g g
  e h

- How would you choose the parameter values for π_i, $a_{i,j}$, and $b_i(o)$?

A Reminder: Models with Hidden Variables

- Now say we have two sets $\mathcal{X}$ and $\mathcal{Y}$, and a joint distribution $P(x, y \mid \Theta)$

- If we had **fully observed data**, (x_i, y_i) pairs, then

$$L(\Theta) = \sum_i \log P(x_i, y_i \mid \Theta)$$

- If we have **partially observed data**, x_i examples, then

$$\begin{aligned} L(\Theta) &= \sum_i \log P(x_i \mid \Theta) \\ &= \sum_i \log \sum_{y \in \mathcal{Y}} P(x_i, y \mid \Theta) \end{aligned}$$

Hidden Markov Models as a Hidden Variable Problem

- We have two sets $\mathcal{X}$ and $\mathcal{Y}$, and a joint distribution $P(x, y \mid \Theta)$

- In Hidden Markov Models:
  each $x \in \mathcal{X}$ is an output sequence $o_1 \ldots o_T$
  each $y \in \mathcal{Y}$ is a state sequence $s_1 \ldots s_T$

Maximum Likelihood Estimates

- We have an HMM with $N = 3$, $K = \{e, f, g, h\}$
  We see the following **paired sequences** in training data

  e/1 g/2
  e/1 h/2
  f/1 h/2
  f/1 g/2

- Maximum likelihood estimates:

$$\pi_1 = 1.0, \quad \pi_2 = 0.0, \quad \pi_3 = 0.0$$

for parameters $a_{i,j}$:

	j=1	j=2	j=3
i=1	0	1	0
i=2	0	0	1

for parameters $b_i(o)$:

	o=e	o=f	o=g	o=h
i=1	0.5	0.5	0	0
i=2	0	0	0.5	0.5

The Likelihood Function for HMMs: Fully Observed Data

- Say $(x, y) = \{o_1 \ldots o_T, s_1 \ldots s_T\}$, and

$$
\begin{aligned}
f(i, j, x, y) &= \text{Number of times state } j \text{ follows state } i \text{ in (x,y)} \\
f(i, x, y) &= \text{Number of times state } i \text{ is the initial state in (x,y) (1 or 0)} \\
f(i, o, x, y) &= \text{Number of times state } i \text{ is paired with observation } o
\end{aligned}
$$

- Then

$$
P(x, y) = \prod_{i \in \{1 \ldots N-1\}} \pi_i^{f(i,x,y)} \prod_{\substack{i \in \{1 \ldots N-1\}, \\ j \in \{1 \ldots N\}}} a_{i,j}^{f(i,j,x,y)} \prod_{\substack{i \in \{1 \ldots N-1\}, \\ o \in K}} b_i(o)^{f(i,o,x,y)}
$$

The Likelihood Function for HMMs: Fully Observed Data

- If we have training examples (x_l, y_l) for $l = 1 \ldots m$,

$$
\begin{aligned}
L(\Theta) &= \sum_{l=1}^{m} \log P(x_l, y_l) \\
&= \sum_{l=1}^{m} \Bigg(\sum_{i \in \{1 \ldots N-1\}} f(i, x_l, y_l) \log \pi_i + \\
&\qquad \sum_{\substack{i \in \{1 \ldots N-1\}, \\ j \in \{1 \ldots N\}}} f(i, j, x_l, y_l) \log a_{i,j} + \\
&\qquad \sum_{\substack{i \in \{1 \ldots N-1\}, \\ o \in K}} f(i, o, x_l, y_l) \log b_i(o) \Bigg)
\end{aligned}
$$

- Maximizing this function gives maximum-likelihood estimates:

$$\pi_i = \frac{\sum_l f(i, x_l, y_l)}{\sum_l \sum_k f(k, x_l, y_l)}$$

$$a_{i,j} = \frac{\sum_l f(i, j, x_l, y_l)}{\sum_l \sum_k f(i, k, x_l, y_l)}$$

$$b_i(o) = \frac{\sum_l f(i, o, x_l, y_l)}{\sum_l \sum_{o' \in K} f(i, o', x_l, y_l)}$$

The Likelihood Function for HMMs: Partially Observed Data

- If we have training examples (x_l) for $l = 1 \ldots m$,

$$L(\Theta) = \sum_{l=1}^{m} \log \sum_{y} P(x_l, y)$$

$$Q(\Theta, \Theta^{t-1}) = \sum_{l=1}^{m} \sum_{y} P(y \mid x_l, \Theta^{t-1}) \log P(x_l, y \mid \Theta)$$

$$Q(\Theta, \Theta^{t-1}) = \sum_{l=1}^{m} \sum_{y} P(y \mid x_l, \Theta^{t-1}) \left(\sum_{i \in \{1 \ldots N-1\}} f(i, x_l, y) \log \pi_i + \right.$$

$$\left. \sum_{\substack{i \in \{1 \ldots N-1\}, \\ j \in \{1 \ldots N\}}} f(i, j, x_l, y) \log a_{i,j} + \sum_{\substack{i \in \{1 \ldots N-1\}, \\ o \in K}} f(i, o, x_l, y) \log b_i(o) \right)$$

$$= \sum_{l=1}^{m} \left(\sum_{i \in \{1 \ldots N-1\}} g(i, x_l) \log \pi_i + \sum_{\substack{i \in \{1 \ldots N-1\}, \\ j \in \{1 \ldots N\}}} g(i, j, x_l) \log a_{i,j} + \sum_{\substack{i \in \{1 \ldots N-1\}, \\ o \in K}} g(i, o, x_l) \log b_i(o) \right)$$

where each g is an **expected count**:

$$g(i, x_l) = \sum_{y} P(y \mid x_l, \Theta^{t-1}) f(i, x_l, y)$$

$$g(i, j, x_l) = \sum_{y} P(y \mid x_l, \Theta^{t-1}) f(i, j, x_l, y)$$

$$g(i, o, x_l) = \sum_{y} P(y \mid x_l, \Theta^{t-1}) f(i, o, x_l, y)$$

- Maximizing this function gives EM updates:

$$\pi_i = \frac{\sum_l g(i, x_l)}{\sum_l \sum_k g(k, x_l)} \quad a_{i,j} = \frac{\sum_l g(i, j, x_l)}{\sum_l \sum_k g(i, k, x_l)} \quad b_i(o) = \frac{\sum_l g(i, o, x_l)}{\sum_l \sum_{o' \in K} g(i, o', x_l)}$$

- Compare this to maximum likelihood estimates in fully observed case:

$$\pi_i = \frac{\sum_l f(i, x_l, y_l)}{\sum_l \sum_k f(k, x_l, y_l)} \quad a_{i,j} = \frac{\sum_l f(i, j, x_l, y_l)}{\sum_l \sum_k f(i, k, x_l, y_l)} \quad b_i(o) = \frac{\sum_l f(i, o, x_l, y_l)}{\sum_l \sum_{o' \in K} f(i, o', x_l, y_l)}$$

A Hidden Variable Problem

- We have an HMM with $N = 3$, $K = \{e, f, g, h\}$
- We see the following **output sequences** in training data

e	g
e	h
f	h
f	g

- How would you choose the parameter values for π_i, $a_{i,j}$, and $b_i(o)$?

- Four possible state sequences for the first example:

e/1 g/1
e/1 g/2
e/2 g/1
e/2 g/2

- Four possible state sequences for the first example:

e/1 g/1
e/1 g/2
e/2 g/1
e/2 g/2

- Each state sequence has a different probability:

e/1	g/1	$\pi_1 a_{1,1} a_{1,3} b_1(e) b_1(g)$
e/1	g/2	$\pi_1 a_{1,2} a_{2,3} b_1(e) b_2(g)$
e/2	g/1	$\pi_2 a_{2,1} a_{1,3} b_2(e) b_1(g)$
e/2	g/2	$\pi_2 a_{2,2} a_{2,3} b_2(e) b_2(g)$

A Hidden Variable Problem

- Say we have initial parameter values:

$$\pi_1 = 0.35, \quad \pi_2 = 0.3, \quad \pi_3 = 0.35$$

$a_{i,j}$	j=1	j=2	j=3
i=1	0.2	0.3	0.5
i=2	0.3	0.2	0.5

$b_i(o)$	o=e	o=f	o=g	o=h
i=1	0.2	0.25	0.3	0.25
i=2	0.1	0.2	0.3	0.4

- Each state sequence has a different probability:

e/1	g/1	$\pi_1 a_{1,1} a_{1,3} b_1(e) b_1(g) = 0.0021$
e/1	g/2	$\pi_1 a_{1,2} a_{2,3} b_1(e) b_2(g) = 0.00315$
e/2	g/1	$\pi_2 a_{2,1} a_{1,3} b_2(e) b_1(g) = 0.00135$
e/2	g/2	$\pi_2 a_{2,2} a_{2,3} b_2(e) b_2(g) = 0.0009$

A Hidden Variable Problem

- Each state sequence has a different probability:

e/1	g/1	$\pi_1 a_{1,1} a_{1,3} b_1(e) b_1(g) = 0.0021$
e/1	g/2	$\pi_1 a_{1,2} a_{2,3} b_1(e) b_2(g) = 0.00315$
e/2	g/1	$\pi_2 a_{2,1} a_{1,3} b_2(e) b_1(g) = 0.00135$
e/2	g/2	$\pi_2 a_{2,2} a_{2,3} b_2(e) b_2(g) = 0.0009$

- Each state sequence has a different **conditional** probability, e.g.:

$$P(1\ 1 \mid \text{e g}, \Theta) = \frac{0.0021}{0.0021 + 0.00315 + 0.00135 + 0.0009} = 0.28$$

e/1	g/1	$P(1\ 1 \mid \text{e g}, \Theta) = 0.28$
e/1	g/2	$P(1\ 2 \mid \text{e g}, \Theta) = 0.42$
e/2	g/1	$P(2\ 1 \mid \text{e g}, \Theta) = 0.18$
e/2	g/2	$P(2\ 2 \mid \text{e g}, \Theta) = 0.12$

fill in hidden values for (e g), (e h), (f h), (f g)

e/1	g/1	$P(1\ 1 \mid \text{e g}, \Theta) = 0.28$
e/1	g/2	$P(1\ 2 \mid \text{e g}, \Theta) = 0.42$
e/2	g/1	$P(2\ 1 \mid \text{e g}, \Theta) = 0.18$
e/2	g/2	$P(2\ 2 \mid \text{e g}, \Theta) = 0.12$
e/1	h/1	$P(1\ 1 \mid \text{e h}, \Theta) = 0.211$
e/1	h/2	$P(1\ 2 \mid \text{e h}, \Theta) = 0.508$
e/2	h/1	$P(2\ 1 \mid \text{e h}, \Theta) = 0.136$
e/2	h/2	$P(2\ 2 \mid \text{e h}, \Theta) = 0.145$
f/1	h/1	$P(1\ 1 \mid \text{f h}, \Theta) = 0.181$
f/1	h/2	$P(1\ 2 \mid \text{f h}, \Theta) = 0.434$
f/2	h/1	$P(2\ 1 \mid \text{f h}, \Theta) = 0.186$
f/2	h/2	$P(2\ 2 \mid \text{f h}, \Theta) = 0.198$
f/1	g/1	$P(1\ 1 \mid \text{f g}, \Theta) = 0.237$
f/1	g/2	$P(1\ 2 \mid \text{f g}, \Theta) = 0.356$
f/2	g/1	$P(2\ 1 \mid \text{f g}, \Theta) = 0.244$
f/2	g/2	$P(2\ 2 \mid \text{f g}, \Theta) = 0.162$

Calculate the expected counts:

$$\sum_l g(1, x_l) = 0.28 + 0.42 + 0.211 + 0.508 + 0.181 + 0.434 + 0.237 + 0.356 = 2.628$$

$$\sum_l g(2, x_l) = 1.372$$

$$\sum_l g(3, x_l) = 0.0$$

$$\sum_l g(1, 1, x_l) = 0.28 + 0.211 + 0.181 + 0.237 = 0.910$$

$$\sum_l g(1, 2, x_l) = 1.72$$

$$\sum_l g(2, 1, x_l) = 0.746$$

$$\sum_l g(2, 2, x_l) = 0.626$$

$$\sum_l g(1, 3, x_l) = 1.656$$

$$\sum_l g(2, 3, x_l) = 2.344$$

Calculate the expected counts:

$$\begin{aligned}
\sum_l g(1,e,x_l) &= 0.28 + 0.42 + 0.211 + 0.508 = 1.4 \\
\sum_l g(1,f,x_l) &= 1.209 \\
\sum_l g(1,g,x_l) &= 0.941 \\
\sum_l g(1,h,x_l) &= 0.827 \\
\sum_l g(2,e,x_l) &= 0.6 \\
\sum_l g(2,f,x_l) &= 0.385 \\
\sum_l g(2,g,x_l) &= 1.465 \\
\sum_l g(2,h,x_l) &= 1.173
\end{aligned}$$

Calculate the new estimates:

$$\pi_1 = \frac{\sum_l g(1,x_l)}{\sum_l g(1,x_l) + \sum_l g(2,x_l) + \sum_l g(3,x_l)} = \frac{2.628}{2.628 + 1.372 + 0} = 0.657$$

$$\pi_2 = 0.343 \quad \pi_3 = 0$$

$$a_{1,1} = \frac{\sum_l g(1,1,x_l)}{\sum_l g(1,1,x_l) + \sum_l g(1,2,x_l) + \sum_l g(1,3,x_l)} = \frac{0.91}{0.91 + 1.72 + 1.656} = 0.212$$

$a_{i,j}$	j=1	j=2	j=3
i=1	0.212	0.401	0.387
i=2	0.201	0.169	0.631

$b_i(o)$	o=e	o=f	o=g	o=h
i=1	0.320	0.276	0.215	0.189
i=2	0.166	0.106	0.404	0.324

Iterate this 3 times:

$$\pi_1 = 0.9986, \quad \pi_2 = 0.00138 \quad \pi_3 = 0$$

$a_{i,j}$	j=1	j=2	j=3
i=1	0.0054	0.9896	0.00543
i=2	0.0	0.0013627	0.9986

$b_i(o)$	o=e	o=f	o=g	o=h
i=1	0.497	0.497	0.00258	0.00272
i=2	0.001	0.000189	0.4996	0.4992

Overview

- The EM algorithm in general form
- The EM algorithm for hidden markov models (brute force)
- The EM algorithm for hidden markov models (dynamic programming)

The Forward-Backward or Baum-Welch Algorithm

- Aim is to (efficiently!) calculate the expected counts:

$$
\begin{aligned}
g(i, x_l) &= \sum_y P(y \mid x_l, \Theta^{t-1}) f(i, x_l, y) \\
g(i, j, x_l) &= \sum_y P(y \mid x_l, \Theta^{t-1}) f(i, j, x_l, y) \\
g(i, o, x_l) &= \sum_y P(y \mid x_l, \Theta^{t-1}) f(i, o, x_l, y)
\end{aligned}
$$

The Forward-Backward or Baum-Welch Algorithm

- Suppose we could calculate the following quantities, given an input sequence $o_1 \ldots o_T$:

$$\alpha_i(t) = P(o_1 \ldots o_{t-1}, s_t = i \mid \Theta) \quad \text{forward probabilities}$$

$$\beta_i(t) = P(o_t \ldots o_T \mid s_t = i, \Theta) \quad \text{backward probabilities}$$

- The probability of being in state i at time t, is

$$\begin{aligned} p_t(i) &= P(s_t = i \mid o_1 \ldots o_T, \Theta) \\ &= \frac{P(s_t = i, o_1 \ldots o_T \mid \Theta)}{P(o_1 \ldots o_T \mid \Theta)} \\ &= \frac{\alpha_t(i)\beta_t(i)}{P(o_1 \ldots o_T \mid \Theta)} \end{aligned}$$

also,

$$P(o_1 \ldots o_T \mid \Theta) = \sum_i \alpha_t(i)\beta_t(i) \text{ for any } t$$

Expected Initial Counts

- As before,

$$g(i, o_1 \ldots o_T) = \text{expected number of times state } i \text{ is state 1}$$

- We can calculate this as

$$g(i, o_1 \ldots o_T) = p_1(i)$$

Expected Emission Counts

- As before,

$$g(i, o, o_1 \ldots o_T) = \text{expected number of times state } i \text{ emits the symbol } o$$

- We can calculate this as

$$g(i, o, o_1 \ldots o_T) = \sum_{t: o_t = o} p_t(i)$$

The Forward-Backward or Baum-Welch Algorithm

- Suppose we could calculate the following quantities, given an input sequence $o_1 \ldots o_T$:

$$\alpha_i(t) = P(o_1 \ldots o_{t-1}, s_t = i \mid \Theta) \quad \text{forward probabilities}$$

$$\beta_i(t) = P(o_t \ldots o_T \mid s_t = i, \Theta) \quad \text{backward probabilities}$$

- The probability of being in state i at time t, and in state j at time $t+1$, is

$$\begin{aligned} p_t(i,j) &= P(s_t = i, s_{t+1} = j \mid o_1 \ldots o_T, \Theta) \\ &= \frac{P(s_t = i, s_{t+1} = j, o_1 \ldots o_T \mid \Theta)}{P(o_1 \ldots o_T \mid \Theta)} \\ &= \frac{\alpha_t(i) a_{i,j} b_i(o_t) \beta_{t+1}(j)}{P(o_1 \ldots o_T \mid \Theta)} \end{aligned}$$

also,

$$P(o_1 \ldots o_T \mid \Theta) = \sum_i \alpha_t(i) \beta_t(i) \text{ for any } t$$

Expected Transition Counts

- As before,

$$g(i, j, o_1 \ldots o_T) = \text{expected number of times state } j \text{ follows state } i$$

- We can calculate this as

$$g(i, j, o_1 \ldots o_T) = \sum_t p_t(i, j)$$

Recursive Definitions for Forward Probabilities

- Given an input sequence $o_1 \ldots o_T$:

$$\alpha_i(t) \quad = \quad P(o_1 \ldots o_{t-1}, s_t = i \mid \Theta)$$ forward probabilities

- **Base case:**

$$\alpha_i(1) = \pi_i \quad \text{for all } i$$

- **Recursive case:**

$$\alpha_j(t{+}1) = \sum_i \alpha_i(t) a_{i,j} b_i(o_t) \quad \text{for all } j = 1 \ldots N \text{ and } t = 2 \ldots T$$

Recursive Definitions for Backward Probabilities

- Given an input sequence $o_1 \ldots o_T$:

$$\beta_i(t) \quad = \quad P(o_t \ldots o_T \mid s_t = i, \Theta)$$ backward probabilities

- **Base case:**

$$\beta_i(T+1) = 1 \quad \text{for } i = N$$

$$\beta_i(T+1) = 0 \quad \text{for } i \neq N$$

- **Recursive case:**

$$\beta_i(t) = \sum_j a_{i,j} b_i(o_t) \beta_j(t+1) \quad \text{for all } j = 1 \ldots N \text{ and } t = 1 \ldots T$$

Overview

- The EM algorithm in general form
  (more about the 3 coin example)

- The EM algorithm for hidden markov models (brute force)

- The EM algorithm for hidden markov models (dynamic programming)

- Briefly: The EM algorithm for PCFGs

EM for Probabilistic Context-Free Grammars

- A PCFG defines a distribution $P(S, T \mid \Theta)$ over tree/sentence pairs (S, T)

- If we had tree/sentence pairs **(fully observed data)** then

$$L(\Theta) = \sum_i \log P(S_i, T_i \mid \Theta)$$

- Say we have sentences only, $S_1 \ldots S_n$
$\Rightarrow$ trees are hidden variables

$$L(\Theta) = \sum_i \log \sum_T P(S_i, T \mid \Theta)$$

EM for Probabilistic Context-Free Grammars

- Say we have sentences only, $S_1 \dots S_n$
  $\Rightarrow$ trees are hidden variables

$$L(\Theta) = \sum_i \log \sum_T P(S_i, T \mid \Theta)$$

- EM algorithm is then $\Theta^t = \mathrm{argmax}_\Theta Q(\Theta, \Theta^{t-1})$, where

$$Q(\Theta, \Theta^{t-1}) = \sum_i \sum_T P(T \mid S_i, \Theta^{t-1}) \log P(S_i, T \mid \Theta)$$

- Remember:

$$\log P(S_i, T \mid \Theta) = \sum_{r \in R} Count(S_i, T, r) \log \Theta_r$$

where $Count(S, T, r)$ is the number of times rule r is seen in the sentence/tree pair (S, T)

$$\begin{aligned} \Rightarrow Q(\Theta, \Theta^{t-1}) &= \sum_i \sum_T P(T \mid S_i, \Theta^{t-1}) \log P(S_i, T \mid \Theta) \\ &= \sum_i \sum_T P(T \mid S_i, \Theta^{t-1}) \sum_{r \in R} Count(S_i, T, r) \log \Theta_r \\ &= \sum_i \sum_{r \in R} Count(S_i, r) \log \Theta_r \end{aligned}$$

where $Count(S_i, r) = \sum_T P(T \mid S_i, \Theta^{t-1}) Count(S_i, T, r)$
the expected counts

- Solving $\Theta_{ML} = \mathrm{argmax}_{\Theta \in \Omega} L(\Theta)$ gives

$$\Theta_{\alpha \to \beta} = \frac{\sum_i Count(S_i, \alpha \to \beta)}{\sum_i \sum_{s \in R(\alpha)} Count(S_i, s)}$$

- There are efficient algorithms for calculating

$$Count(S_i, r) = \sum_T P(T \mid S_i, \Theta^{t-1}) Count(S_i, T, r)$$

for a PCFG. See (Baker 1979), called “The Inside Outside Algorithm”. See also Manning and Schuetze section 11.3.4.