WestlakeNLP

Natural Language Processing

Yue Zhang
Westlake University

西湖大學
WESTLAKE UNIVERSITY

Chapter 6

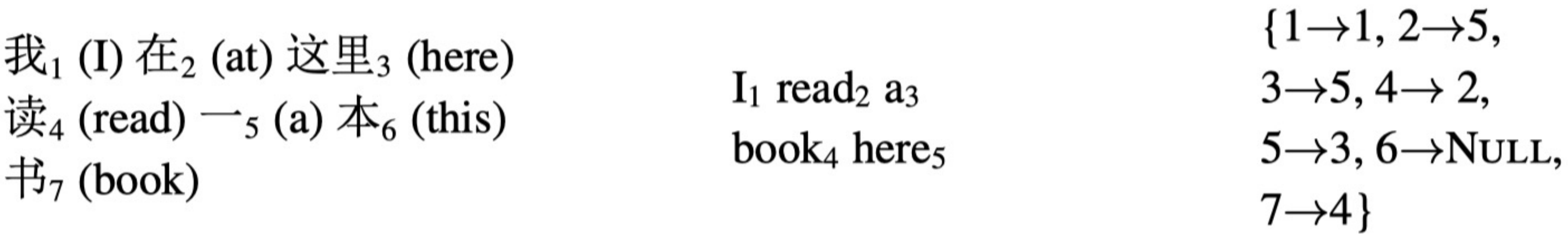
Hidden Variables

Contents

- 6.1 Expectation Maximization
    - 6.1.1 Introduction
    - 6.1.2 EM
    - 6.1.3 Relationships between MLE and EM
- 6.2 Using EM
    - 6.2.1 Unsupervised Naïve Bayes model
    - 6.2.2 IBM Model 1
    - 6.2.3 Probabilistic latent sentiment analysis
    - 6.2.4 The relative advantages of generative models
- 6.3 EM theory
    - 6.3.1 EM and KL-Divergence
    - 6.3.2 EM derivation using numerical optimization

Hidden Variables

- Examples
  - Large unlabeled text corpora for low-resource languages and domains
  - Labelled alignment between word pairs in a sentence pair for Machine Translation



我$_1$ (I) 在$_2$ (at) 这里$_3$ (here) 读$_4$ (read) 一$_5$ (a) 本$_6$ (this) 书$_7$ (book)

I$_1$ read$_2$ a$_3$ book$_4$ here$_5$

$\{1\rightarrow 1, 2\rightarrow 5, 3\rightarrow 5, 4\rightarrow 2, 5\rightarrow 3, 6\rightarrow \text{NULL}, 7\rightarrow 4\}$

- MLE by counting relative frequency is infeasible

Contents

WestlakeNLP

- **6.1 Expectation Maximization**
    - 6.1.1 Introduction
    - 6.1.2 EM
    - 6.1.3 Relationships between MLE and EM
- 6.2 Using EM
    - 6.2.1 Unsupervised Naïve Bayes model
    - 6.2.2 IBM Model 1
    - 6.2.3 Probabilistic latent sentiment analysis
    - 6.2.4 The relative advantages of generative models
- 6.3 EM theory
    - 6.3.1 EM and KL-Divergence
    - 6.3.2 EM derivation using numerical optimization

Dealing with Hidden Variables

- Revisit Naïve Bayes classification
  - Maximizing likelihood $P(D)$
  - $P(D) = \prod_{i=1}^{N} P(d_i, c_i) = \prod_{i=1}^{N}(P(c_i)\prod_{j=1}^{|d_i|} P(w_j^i|c_i))$

Dealing with Hidden Variables

- Revisit Naïve Bayes classification
  - Maximizing likelihood $P(D)$
  - $P(D) = \prod_{i=1}^{N} P(d_i, c_i) = \prod_{i=1}^{N}(P(c_i) \prod_{j=1}^{|d_i|} P(w_j^i|c_i))$

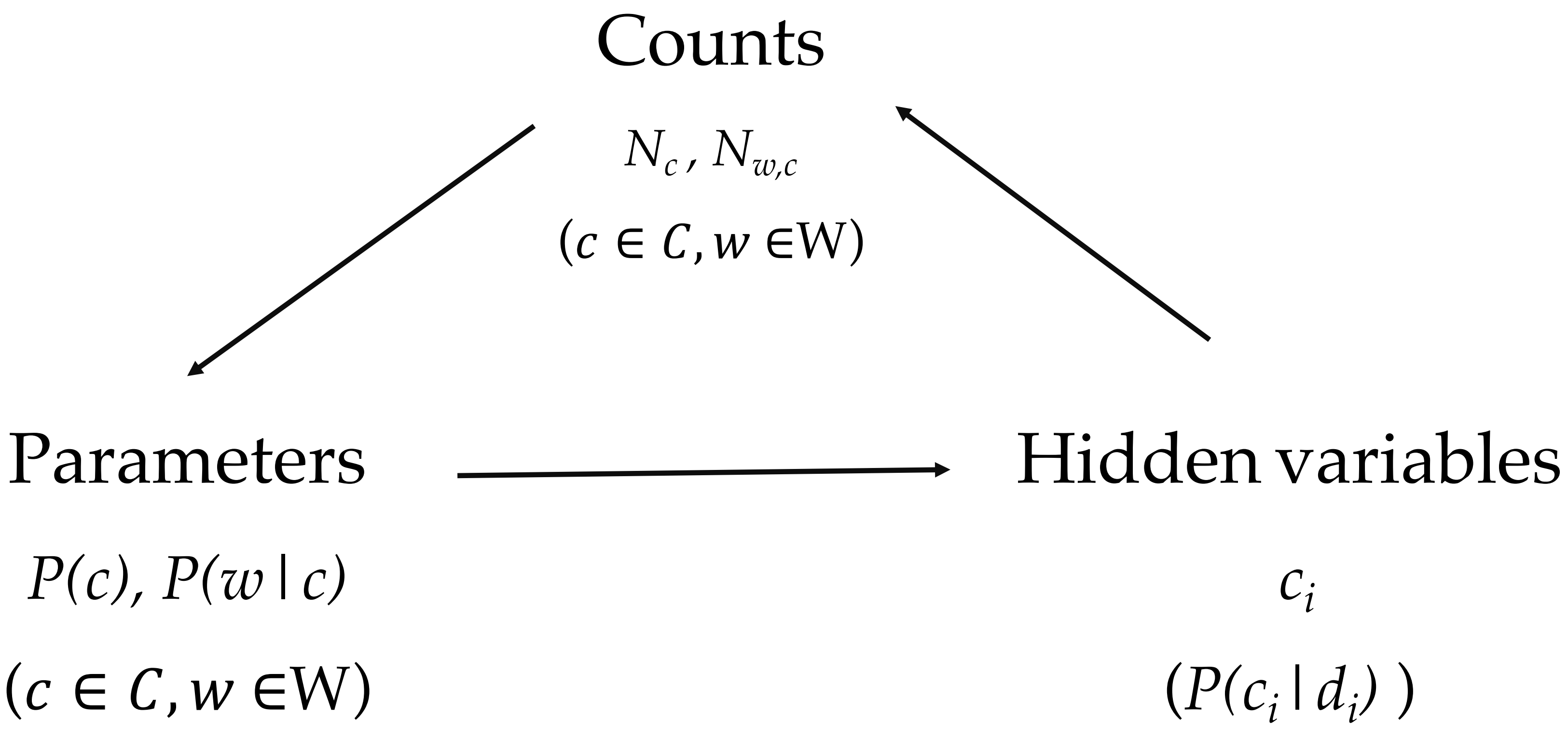
    $= (\prod_{c \in C} P(c)^{N_c}) \cdot (\prod_{w \in V} \prod_{c \in C} P(w|c)^{N_{w,c}})$
- $(\prod_{c \in C} P(c)^{N_c})$ and $(\prod_{w \in V} \prod_{c \in C} P(w|c)^{N_{w,c}})$ can be seen as two independent distribution, each representing the probability of a set of *iid* samples.
- By maximizing $P(D)$ we can derive the value of $P(c)$ and $P(w|c)$ by counting relative frequencies. $P(c) = \frac{N_c}{N}, P(w|c) = \frac{N_{w,c}}{N_c}$

Dealing with Hidden Variables

- Revisit Naïve Bayes classification
  - Maximizing likelihood $P(D)$
  - $P(D) = \prod_{i=1}^{N} P(d_i, c_i) = \prod_{i=1}^{N} (P(c_i) \prod_{j=1}^{|d_i|} P(w_j^i | c_i))$

    $= (\prod_{c \in C} P(c)^{N_c}) \cdot (\prod_{w \in V} \prod_{c \in C} P(w|c)^{N_{w,c}})$
- $(\prod_{c \in C} P(c)^{N_c})$ and $(\prod_{w \in V} \prod_{c \in C} P(w|c)^{N_{w,c}})$ can be seen as two independent distribution, each representing the probability of a set of *iid* samples.
- What if N_c and $N_{w,c}$ are unknown?

Dealing with Hidden Variables



Counts

N_c , $N_{w,c}$

$(c \in C, w \in W)$

Parameters

$P(c)$, $P(w \mid c)$

$(c \in C, w \in W)$

Hidden variables

c_i

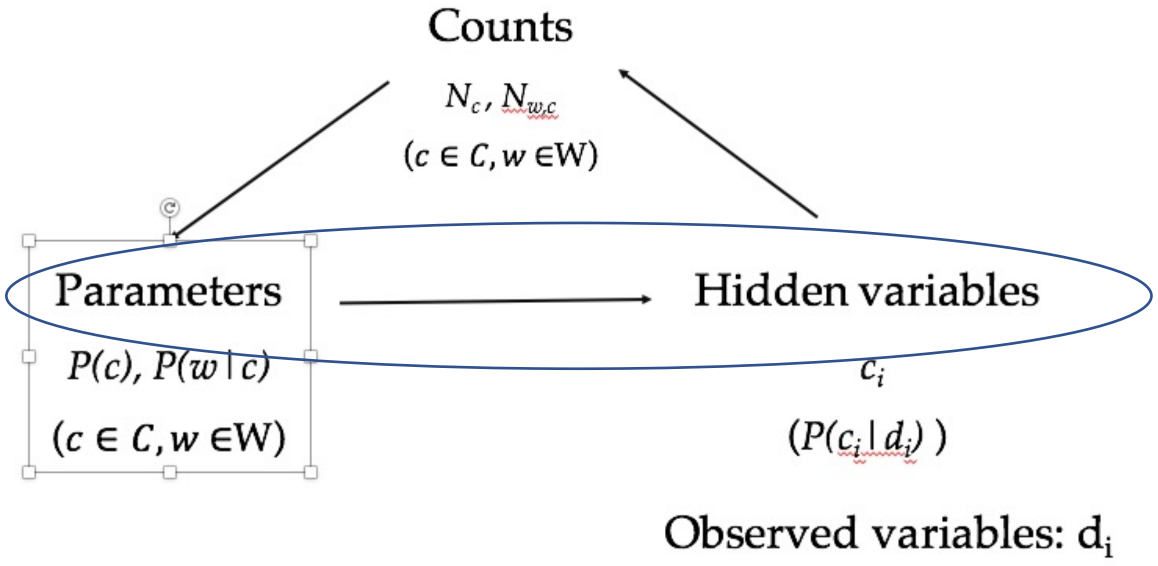
$(P(c_i \mid d_i))$

Observed variables: d_i

Dealing with Hidden Variables

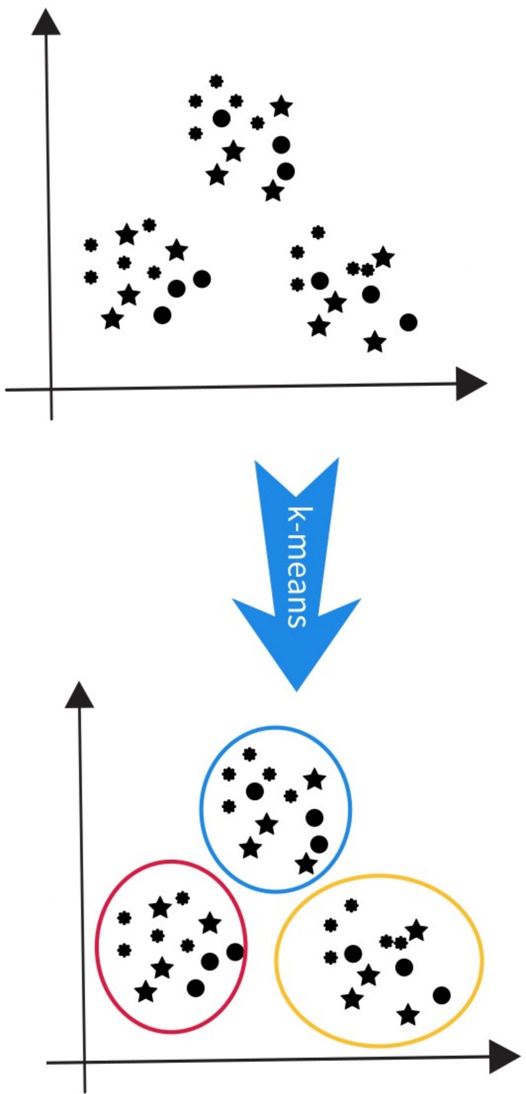
- Iteratively estimate model parameters and hidden variable counts.

- Three ways to estimate hidden variable counts
  - Hard max
  - Soft probabilities
  - Sampling



K-means clustering

- Observation on K-means clustering
  - Observed variables: vectors
  - Hidden variables: cluster assignment
  - Model parameters: centroids
- Initialization: random centroids
- Iterative processes:
  - Hidden variable estimation step:
    - random point-cluster assignment
  - Model parameter estimation step:
    - Centroids are the means



K-means clustering

- Observed points: $O = \{o_i\}|_{i=1}^{N}$

- Hidden variables: $h_{ik} = \begin{cases} I & if\ o_i \in c_k \\ 0 & otherwise \end{cases}$

- Model parameters: c_k is the centroid of cluster k

- If $h_{i,k}$are known(no hidden variable) , learning objective is to find the means, which is to minimize:

  $$L(O) = \sum_{i=1}^{N} \sum_{k=1}^{K} h_{ik} ||o_i - c_k||^2$$

- The optimal value: $c_k^{t+1} = \frac{\sum_{i=1}^{N} \mathrm{b} h_{ik}^{t} o_i}{\sum_{i=1}^{N} \mathrm{b} h_{ik}^{t}}$,

  which is the average of all points in cluster k.

K-means clustering

- Notations:
  - Observed vectors $\theta = \{o_i\}|_{i=1}^{N}$
  - Hidden variable: $H = \{h_{ik}\}|_{i=1,k=1}^{N,K}$
  - Parameter: $\Theta = \{c_k\}|_{k=1}^{K}$
- Training process:
  - Iteration over H: $H^t \leftarrow \arg m\, in \sum_{i=1}^{N} \sum_{k=1}^{K} h_{ik} \, ||o_i - c_k^t \, ||^2$
  - Iteration over θ: $\Theta^{t+1} \leftarrow \arg m\, in \sum_{i=1}^{N} \sum_{k=1}^{K} h_{ik}^t ||o_i - c_k||^2$

Contents

- 6.1 Expectation Maximization
  - **6.1.1 Introduction**
  - 6.1.2 EM
  - 6.1.3 Relationships between MLE and EM
- 6.2 Using EM
  - 6.2.1 Unsupervised Naïve Bayes model
  - 6.2.2 IBM Model 1
  - 6.2.3 Probabilistic latent sentiment analysis
  - 6.2.4 The relative advantages of generative models
- 6.3 EM theory
  - 6.3.1 EM and KL-Divergence
  - 6.3.2 EM derivation using numerical optimization

K-means as "Hard" EM

- General form of EM algorithm

- Notations:
  - Observed data O
  - Hidden data H
  - Model parameter Θ
  - Model $P(O, H | \Theta)$

- Training objective with hidden variables:
  - Maximizing $L(\Theta) = logP(O|\Theta) = log \sum_H P(O, H|\Theta)$

Two steps for EM

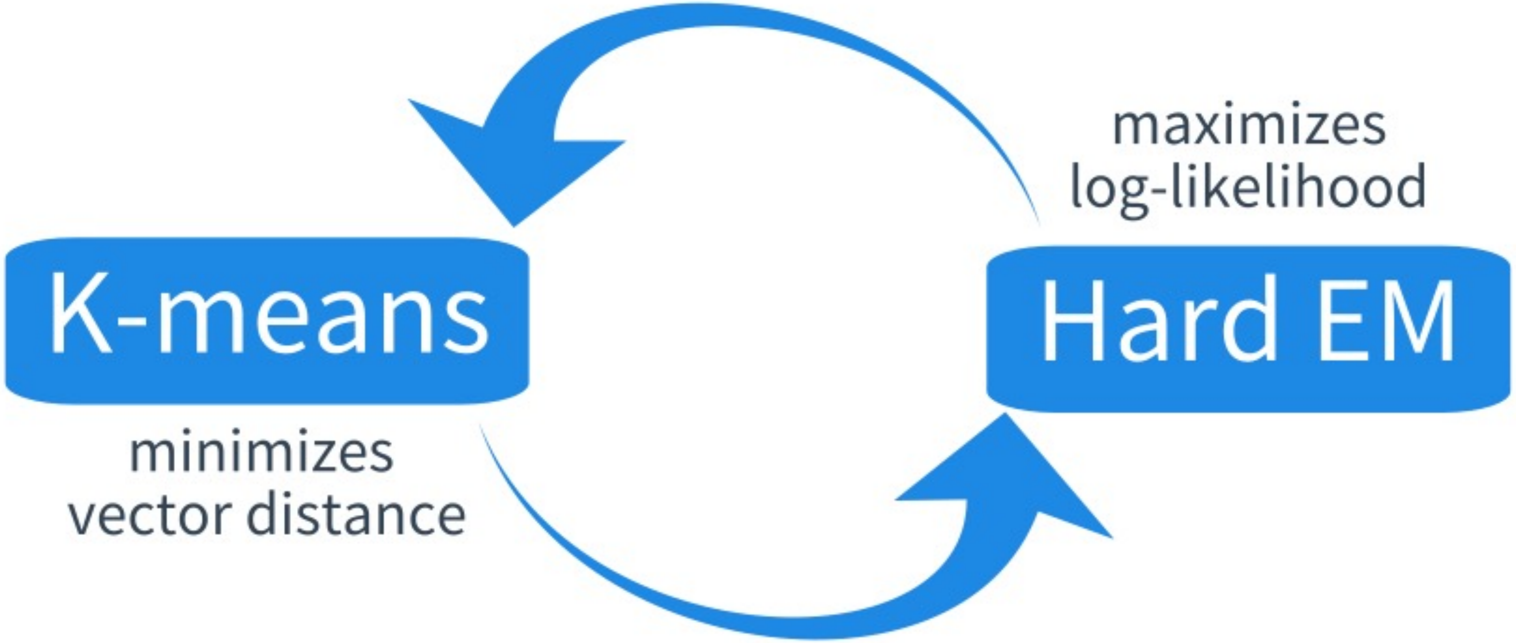
- Expectation (E-step)

  Estimate hidden variable probabilities.

- Maximization (M-step)

  Estimate model parameter values.

K-means as “Hard” EM



- K-means is a type of “hard” EM algorithm.

K-means as “Hard” EM

- The correlation between minimizing distance and maximizing likelihood:

$$
\begin{aligned}
P(\vec{v}_i, h_i|\Theta) &= P(\vec{v}_i, h_{i1}, h_{i2}, \ldots, h_{iK}|\Theta) \\
&= \frac{e^{-\sum_{k=1}^{K} h_{ik}||\vec{v}_i - \vec{c}_k||^2}}{Z} \\
&= \mathcal{N}(\sum_{k=1}^{K} h_{ik}\vec{v}_i, I)
\end{aligned}
$$

K-means as “Hard” EM

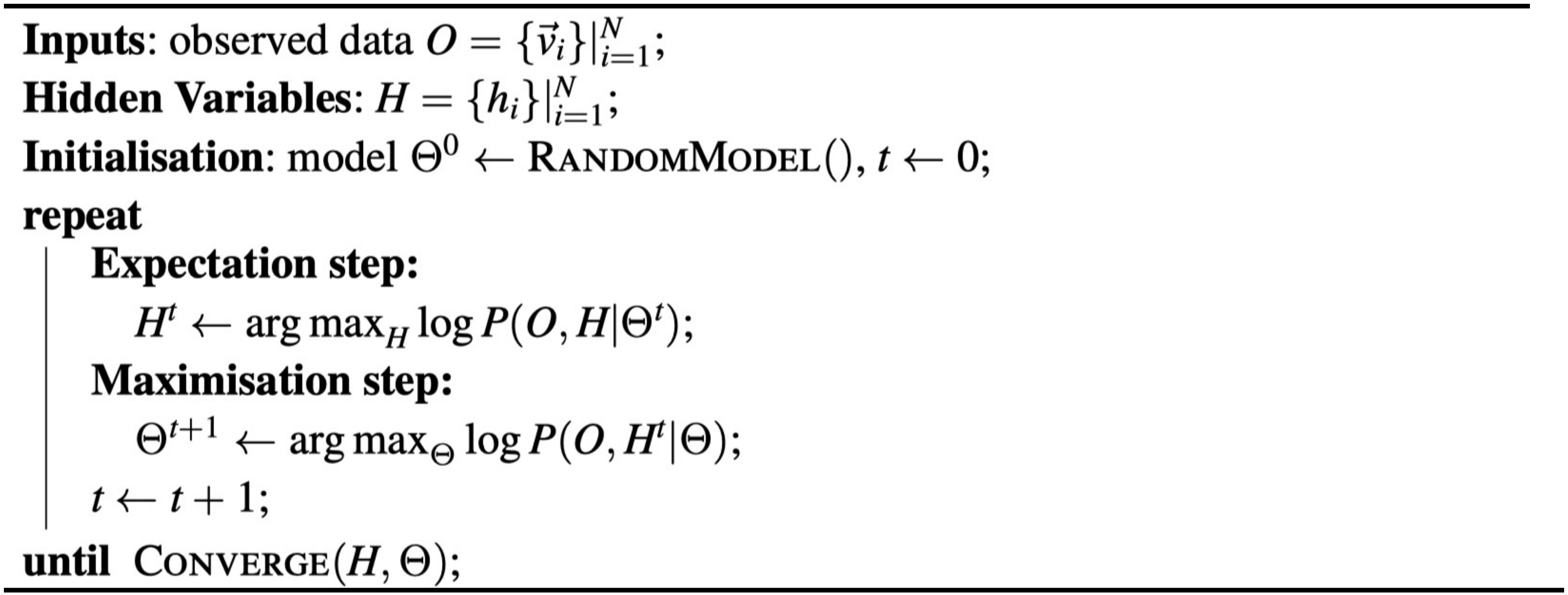
- The correlation between k-means and EM:
  - Expectation step – refer cluster assignment

$$\mathcal{H} = argmax_{\mathcal{H}'} P(O, H = \mathcal{H}'|\Theta) = argmax_{\mathcal{H}'} \sum_{i=1}^{N} P\left(\vec{v_i}, h_i = \hbar_i{}' \middle| \Theta\right)$$

  - Maximization step – cluster centroid estimation

$$\Theta = argmax_{\Theta} P(O, H = \mathcal{H}|\Theta) = argmax_{\Theta} \sum_{i=1}^{N} P\left(\vec{v_i}, h_i = \hbar_i \quad \middle| \Theta\right)$$

K-means as “Hard” EM



```
Inputs: observed data $O = \{\vec{v}_i\}|_{i=1}^{N}$;
Hidden Variables: $H = \{h_i\}|_{i=1}^{N}$;
Initialisation: model $\Theta^0 \leftarrow$ RandomModel(), $t \leftarrow 0$;
repeat
    Expectation step:
        $H^t \leftarrow \arg\max_H \log P(O, H|\Theta^t)$;
    Maximisation step:
        $\Theta^{t+1} \leftarrow \arg\max_\Theta \log P(O, H^t|\Theta)$;
    $t \leftarrow t + 1$;
until Converge($H, \Theta$);
```

- Using a single optimal configuration of H, we optimize

$$L(\Theta) = \log \max_{H} P(O, H|\Theta)$$

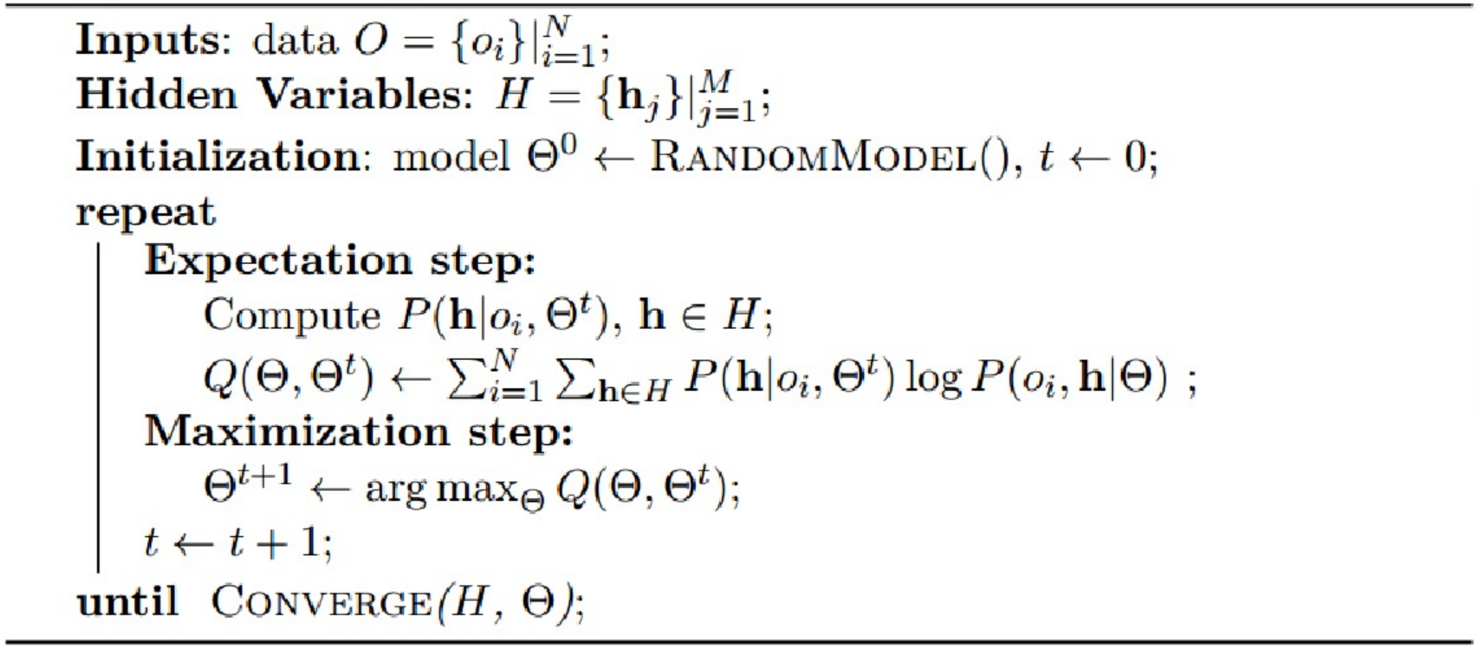
- The optimum is $\Theta^* \leftarrow argmax_{\Theta} max_{H} logP(O, H|\Theta)$

Contents

- 6.1 Expectation Maximization
  - 6.1.1 Introduction
  - **6.1.2 EM**
  - 6.1.3 Relationships between MLE and EM
- 6.2 Using EM
  - 6.2.1 Unsupervised Naïve Bayes model
  - 6.2.2 IBM Model 1
  - 6.2.3 Probabilistic latent sentiment analysis
  - 6.2.4 The relative advantages of generative models
- 6.3 EM theory
  - 6.3.1 EM and KL-Divergence
  - 6.3.2 EM derivation using numerical optimization

EM

EM considers all possible values of hidden variables.



```
Inputs: data $O = \{o_i\}|_{i=1}^{N}$;
Hidden Variables: $H = \{\mathbf{h}_j\}|_{j=1}^{M}$;
Initialization: model $\Theta^0 \leftarrow$ RANDOMMODEL(), $t \leftarrow 0$;
repeat
    Expectation step:
        Compute $P(\mathbf{h}|o_i, \Theta^t)$, $\mathbf{h} \in H$;
        $Q(\Theta, \Theta^t) \leftarrow \sum_{i=1}^{N} \sum_{\mathbf{h} \in H} P(\mathbf{h}|o_i, \Theta^t) \log P(o_i, \mathbf{h}|\Theta)$ ;
    Maximization step:
        $\Theta^{t+1} \leftarrow \arg\max_{\Theta} Q(\Theta, \Theta^t)$;
    $t \leftarrow t + 1$;
until CONVERGE($H$, $\Theta$);
```

Hard EM

$h = argmax_h P(h'|o, \Theta)$

$Q(\Theta, \Theta^t) = \sum_{i=1}^{N} logP(o_i, h_i|\Theta)$

- $P(h|o_i, \Theta^t), h \in H$ is the assignment distribution of H.
- $Q(\Theta, \Theta^t)$ is called the Q-function.

Contents

- 6.1 Expectation Maximization
  - 6.1.1 Introduction
  - 6.1.2 EM
  - **6.1.3 Relationships between MLE and EM**
- 6.2 Using EM
  - 6.2.1 Unsupervised Naïve Bayes model
  - 6.2.2 IBM Model 1
  - 6.2.3 Probabilistic latent sentiment analysis
  - 6.2.4 The relative advantages of generative models
- 6.3 EM theory
  - 6.3.1 EM and KL-Divergence
  - 6.3.2 EM derivation using numerical optimization

Relationships between MLE and Q-function

- When the outputs are hidden variables, and if h is known, we can turn EM algorithm to MLE in supervised settings.
  - supposed that each o_i has a supervised label y_i
  - defining

$$P(h|o_i,\Theta^t) = \begin{cases} 1 \text{ if } h = y_i \\ 0 \text{ otherwise} \end{cases}$$

$$\begin{aligned} Q\left(\Theta,\Theta^t\right) &= \sum_{i=1}^{N} \sum_{\mathbf{h}\in H} P(\mathbf{h}|o_i,\Theta^t) \log P\left(o_i,\mathbf{h}|\Theta\right) \\ &= \sum_{i=1}^{N} \log P\left(o_i,y_i|\Theta\right) \end{aligned}$$

which is exactly the maximum log-likelihood training objective.

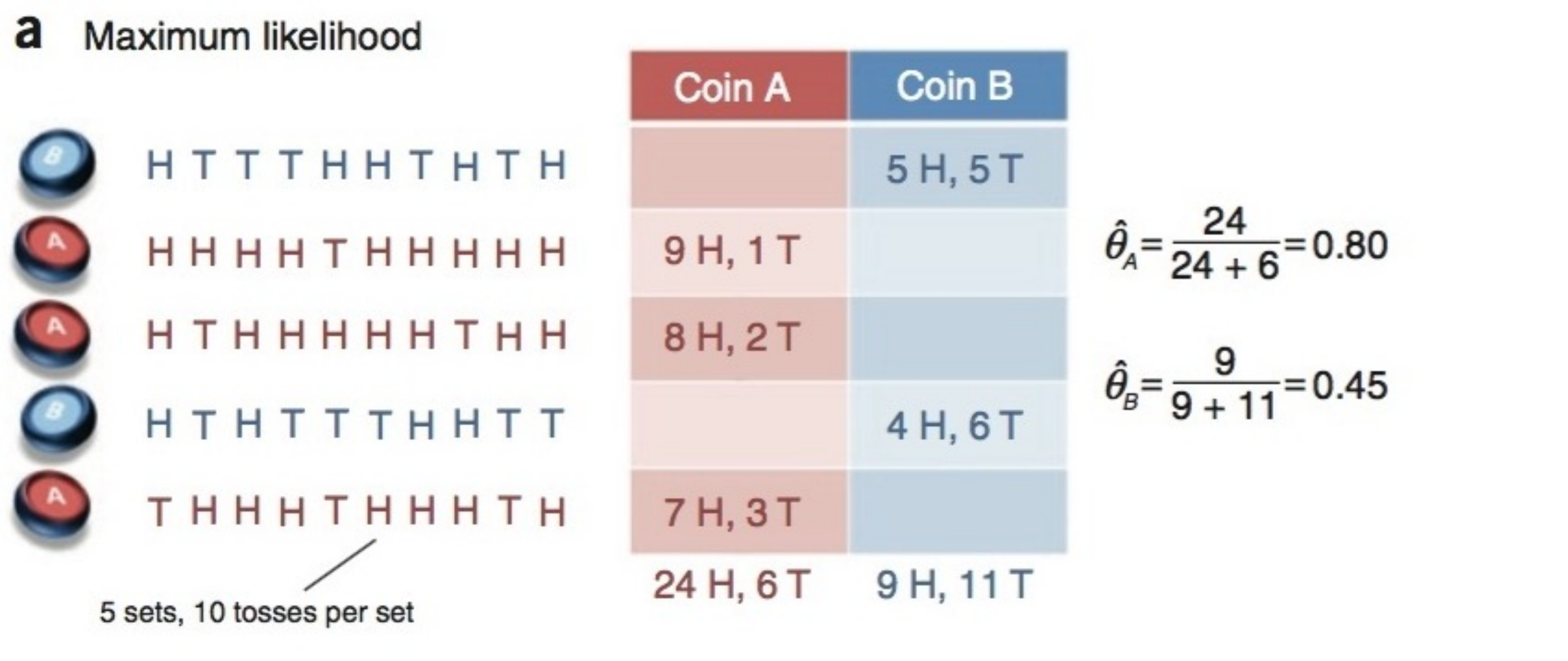
Case study

A coin-flipping experiment

For a pair of coins A and B of unknown biases, θ_A and θ_B. Our goal is to estimate $\theta = (\theta_A, \theta_B)$ by repeating the following procedure five times: randomly choose one of the two coins (with equal probability), and perform ten independent coin tosses with the selected coin. Thus, the entire procedure involves a total of 50 coin tosses.

Case 1

Parameter estimation in this setting is known as the complete data case in that the values of all relevant random variables in our model (that is, the result of each coin flip and the type of coin used for each flip) are known.



a Maximum likelihood

		Coin A	Coin B
B	H T T T H H T H T H		5 H, 5 T
A	H H H H T H H H H H	9 H, 1 T	
A	H T H H H H H T H H	8 H, 2 T	
B	H T H T T T H H T T		4 H, 6 T
A	T H H H T H H H T H	7 H, 3 T	
		24 H, 6 T	9 H, 11 T

5 sets, 10 tosses per set

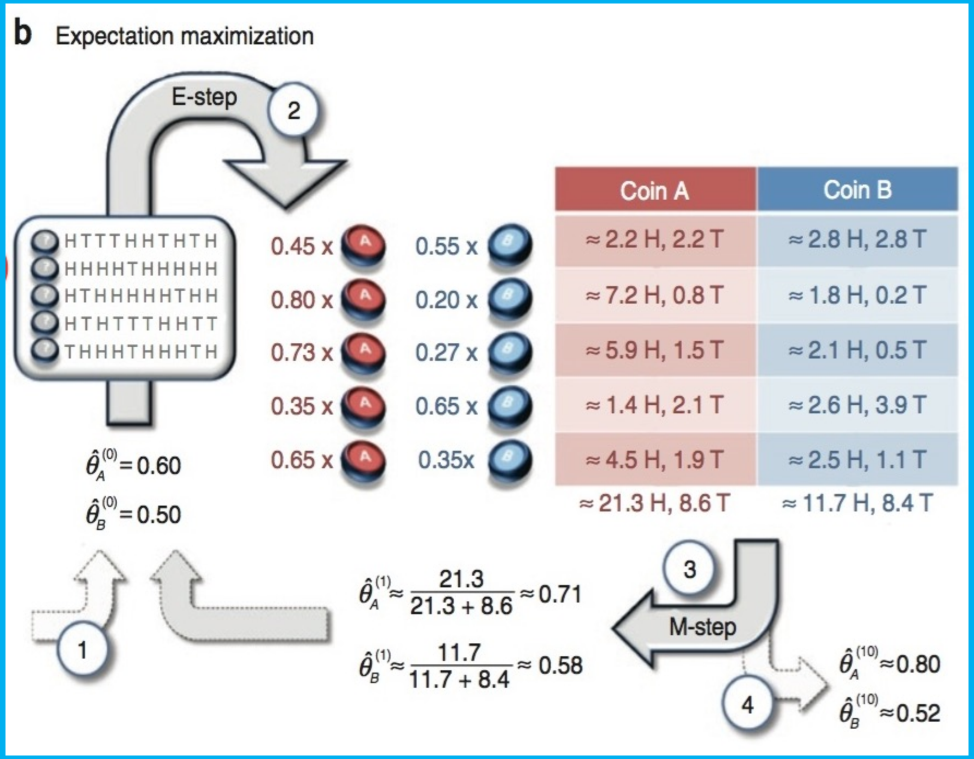
$$\hat{\theta}_A = \frac{24}{24+6} = 0.80$$

$$\hat{\theta}_B = \frac{9}{9+11} = 0.45$$

Problem set originated from Chuong B Do & Serafim Batzoglou

Case 2

Now consider a more challenging variant of the parameter estimation problem in which we are given the recorded head counts but not the identities of the coins used for each set of tosses.


b Expectation maximization
E-step
2
HTTTHHTHTH
HHHHTHHHHH
HTHHHHHTHH
HTHTTTHHTT
THHHTHHHTH
0.45 x
0.55 x
0.80 x
0.20 x
0.73 x
0.27 x
0.35 x
0.65 x
0.65 x
0.35x
A
B
$\hat{\theta}_A^{(0)} = 0.60$
$\hat{\theta}_B^{(0)} = 0.50$
1
3
M-step
4
$\hat{\theta}_A^{(1)} \approx \frac{21.3}{21.3 + 8.6} \approx 0.71$
$\hat{\theta}_B^{(1)} \approx \frac{11.7}{11.7 + 8.4} \approx 0.58$
$\hat{\theta}_A^{(10)} \approx 0.80$
$\hat{\theta}_B^{(10)} \approx 0.52$

Coin A	Coin B
≈ 2.2 H, 2.2 T	≈ 2.8 H, 2.8 T
≈ 7.2 H, 0.8 T	≈ 1.8 H, 0.2 T
≈ 5.9 H, 1.5 T	≈ 2.1 H, 0.5 T
≈ 1.4 H, 2.1 T	≈ 2.6 H, 3.9 T
≈ 4.5 H, 1.9 T	≈ 2.5 H, 1.1 T
≈ 21.3 H, 8.6 T	≈ 11.7 H, 8.4 T

O=HTTTHHTHTH

$$P(A|O,\tilde{\theta}_A^{(o)},\tilde{\theta}_B^{(o)})$$

$$=\frac{P\left(A,O\middle|\tilde{\theta}_A^{(o)},\tilde{\theta}_B^{(o)}\right)}{P(O,\tilde{\theta}_A^{(o)},\tilde{\theta}_B^{(o)})}$$

$$=\frac{P\left(A\middle|\tilde{\theta}_A^{(o)},\tilde{\theta}_B^{(o)}\right)\cdot P\left(O\middle|A,\tilde{\theta}_A^{(o)},\tilde{\theta}_B^{(o)}\right)}{P(O,\tilde{\theta}_A^{(o)},\tilde{\theta}_B^{(o)})}$$

$$P(A|O,\theta)=\frac{0.6^5\times 0.4^5}{0.6^5\times 0.4^5+0.5^5\times 0.5^5}$$

$$P(B|O,\theta)=\frac{0.5^5\times 0.5^5}{0.6^5\times 0.4^5+0.5^5\times 0.5^5}$$

Contents

WestlakeNLP

- 6.1 Expectation Maximization
  - 6.1.1 Introduction
  - 6.1.2 EM
  - 6.1.3 Relationships between MLE and EM
- **6.2 Using EM**
  - 6.2.1 Unsupervised Naïve Bayes model
  - 6.2.2 IBM Model 1
  - 6.2.3 Probabilistic latent sentiment analysis
  - 6.2.4 The relative advantages of generative models
- 6.3 EM theory
  - 6.3.1 EM and KL-Divergence
  - 6.3.2 EM derivation using numerical optimization

EM algorithm summary

To apply EM to a certain task, we need three particular steps:

1. define the complete data likelihood $P(O, H|\Theta)$, namely parameterizing the model.
2. compute $P(H|O, \Theta)$

$$P(H|O, \Theta) = \frac{P(O, H|\Theta)}{P(O|\Theta)}$$

3. maximize $Q(\Theta, \Theta^t)$

$$Q(\Theta, \Theta^t) = \sum_h P(h|O, \Theta^t) log P(O, h|\Theta)$$

Contents

- 6.1 Expectation Maximization
  - 6.1.1 Introduction
  - 6.1.2 EM
  - 6.1.3 Relationships between MLE and EM
- 6.2 Using EM
  - **6.2.1 Unsupervised Naïve Bayes model**
  - 6.2.2 IBM Model 1
  - 6.2.3 Probabilistic latent sentiment analysis
  - 6.2.4 The relative advantages of generative models
- 6.3 EM theory
  - 6.3.1 EM and KL-Divergence
  - 6.3.2 EM derivation using numerical optimization

Unsupervised Naïve Bayes model

- Revisit Naïve Bayes

$$P(c|d) = P(c)\prod_{i=1}^{|d|} P(w_i|c)$$

- If class label is hidden

$$P(h|d) = P(h) \cdot \prod_{i=1}^{|d|} P(w_i|h)$$

- Model parameters

$$\Theta = \begin{cases} P(h) \;\; for\, all\, h, \\ P(w|h) \;\; for \;\; all \;\; w, h \end{cases}$$

Unsupervised Naïve Bayes model

- The model target

$$P(d_i, h|\Theta^t) = P(h|\Theta^t)P(d_i|h, \Theta^t) = P(h|\Theta^t)\prod_{i=1}^{|d_i|} P(w_i|h, \Theta^t)$$

- The hidden variable posterior

$$P(h|d_i, \Theta^t) = \frac{P(d_i, h|\Theta^t)}{\sum_{h'} P(d_i, h'|\Theta^t)} = \frac{P(h|\Theta^t)\prod_{i=1}^{|d_i|} P(w_i|h, \Theta^t)}{\sum_{h'} P(h'|\Theta^t)\prod_{i=1}^{|d_i|} P(w_i|h', \Theta^t)}$$

- The objective to minimize

$$Q(\Theta, \Theta^t) = \sum_{i=1}^{N}\sum_{h} P(h|d_i, \Theta^t) log P(d_i, h|\Theta)$$

Unsupervised Naïve Bayes model

Unsupervised Naïve Bayes model

- Finding $\arg\max_{\Theta} Q(\Theta, \Theta^t)$

  s.t. $\sum_h P(h|\Theta) = \sum_{w \in V} P(w|h,\Theta) = I$

- Using Lagrangian multiplier,

  $$P(h|\Theta) = \frac{\sum_{i=1}^{N} P\left(h\middle|d_i, \Theta^t\right)}{N}$$

  $$P(w|h,\Theta) = \frac{\sum_{i=1}^{N} P\left(h\middle|d_i, \Theta^t\right) \sum_{j=1}^{|d_i|} \delta(w_j, w)}{\sum_{i=1}^{N} P\left(h\middle|d_i, \Theta^t\right)|d_i|}$$

  soft EM can be executed now.

Comparison between related models

- Unsupervised Naïve Bayes vs Naïve Bayes

$$P(c) \qquad\qquad P(\hbar|\Theta)$$

$$P(w|c) \qquad\qquad P(w|\hbar,\Theta)$$

$$\sum_{i=1}^{N}\left(\delta(c_i,c)\cdot\sum_{j=1}^{|d_i|}\delta(w_j^i,w)\right) \qquad \sum_{i=1}^{N} P(\hbar|d_i,\Theta^t)\sum_{j=1}^{|d_i|}\delta(w_j,w)$$

- Unsupervised Naïve Bayes vs k-means
  - K-means is based on vector space geometry, while Naïve Bayes is direct probability model.
  - Naïve Bayes is optimized with EM, while k-means is a hard variant of EM.

Contents

- 6.1 Expectation Maximization
  - 6.1.1 Introduction
  - 6.1.2 EM
  - 6.1.3 Relationships between MLE and EM
- 6.2 Using EM
  - 6.2.1 Unsupervised Naïve Bayes model
  - **6.2.2 IBM Model 1**
  - 6.2.3 Probabilistic latent sentiment analysis
  - 6.2.4 The relative advantages of generative models
- 6.3 EM theory
  - 6.3.1 EM and KL-Divergence
  - 6.3.2 EM derivation using numerical optimization

IBM Model 1

- A probabilistic model for Machine Translation (MT)
  - source sentence X
  - target language translation Y
  - source word $X = x_1 x_2 \ldots x_{|X|}$
  - Target word $Y = y_1 y_2 \ldots y_{|Y|}$

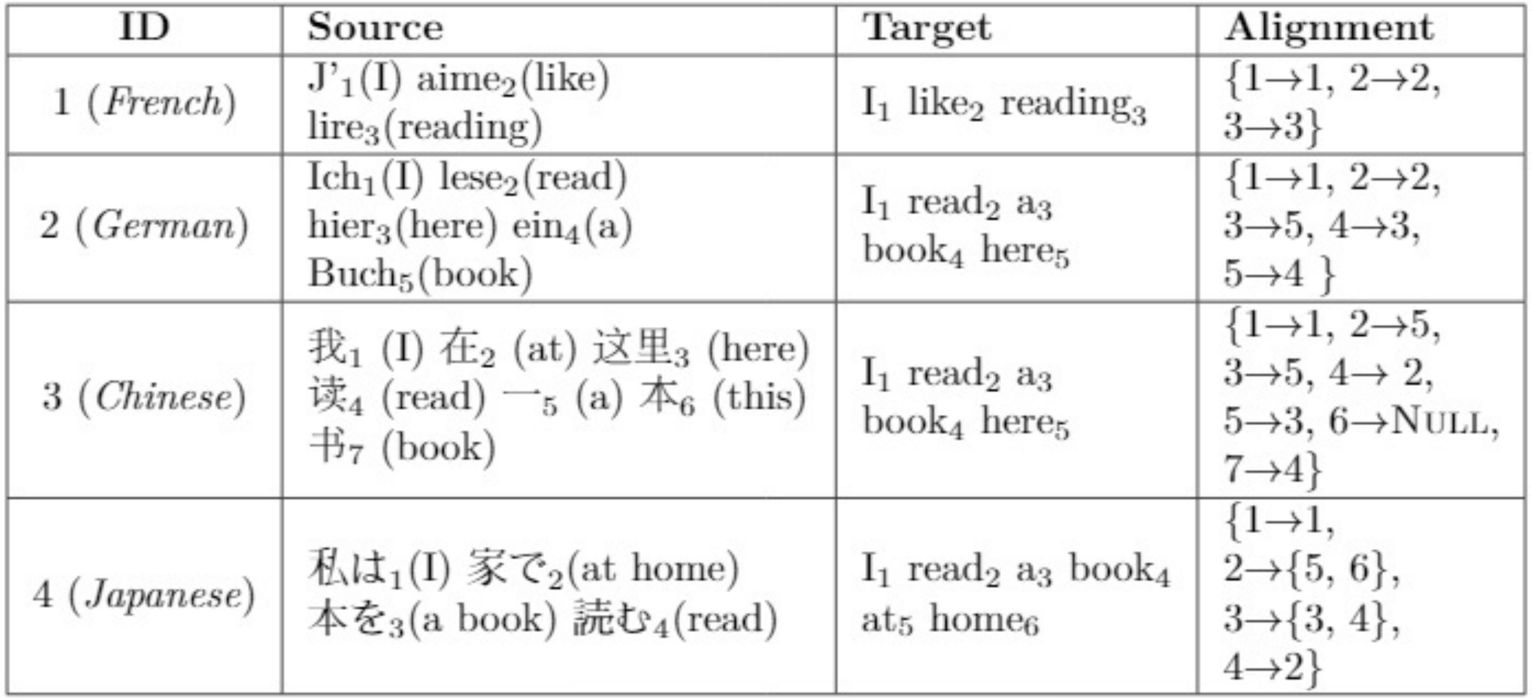
- Bayes rule: $P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)} \propto P(X|Y)P(Y)$
  - Language model : $P(Y)$ ensure fluency
  - Translation model : $P(X|Y)$ ensure adequacy

- Using probability chain rule and assuming each source word x_i is conditionally dependent to only one target word y_{a_i}, we have

$$P(X|Y) = P\left(x_1 \middle| y_{a_1}\right) P\left(x_2 \middle| y_{a_2}\right) \ldots P\left(x_{|X|} \middle| y_{a_{|X|}}\right)$$

Word alignment

- Word alignment $A = \{a_i\}|_{i=1}^{|X|}$, a_i denotes the index of the target word that the i-th source word translates to.



ID	Source	Target	Alignment
1 (*French*)	J’$_1$(I) aime$_2$(like) lire$_3$(reading)	I$_1$ like$_2$ reading$_3$	$\{1\to1,\ 2\to2,\ 3\to3\}$
2 (*German*)	Ich$_1$(I) lese$_2$(read) hier$_3$(here) ein$_4$(a) Buch$_5$(book)	I$_1$ read$_2$ a$_3$ book$_4$ here$_5$	$\{1\to1,\ 2\to2,\ 3\to5,\ 4\to3,\ 5\to4\ \}$
3 (*Chinese*)	我$_1$ (I) 在$_2$ (at) 这里$_3$ (here) 读$_4$ (read) 一$_5$ (a) 本$_6$ (this) 书$_7$ (book)	I$_1$ read$_2$ a$_3$ book$_4$ here$_5$	$\{1\to1,\ 2\to5,\ 3\to5,\ 4\to\ 2,\ 5\to3,\ 6\to\textsc{Null},\ 7\to4\}$
4 (*Japanese*)	私は$_1$(I) 家で$_2$(at home) 本を$_3$(a book) 読む$_4$(read)	I$_1$ read$_2$ a$_3$ book$_4$ at$_5$ home$_6$	$\{1\to1,\ 2\to\{5,\ 6\},\ 3\to\{3,\ 4\},\ 4\to2\}$

- Types of word alignment between sentence translation pairs: Monotonic , Non-monotonic , Many-to-one , Null

EM training

- Observation variable: $O = (X, Y)$, i.e. sentence translation pairs $D = \{(X_i, Y_i)\}|_{i=1}^{N}$
- Hidden variable: $H = A$,i.e. word alignment A_i

$$P(A, X|Y) = P(A|Y)P(X|A, Y)$$

$$= \frac{\prod_{i=1}^{|X|} P\left(x_i|y_{a_i}\right)}{\#(A \text{ between } X \text{ and } Y)}$$

$$= \frac{\prod_{i=1}^{|X|} P\left(x_i|y_{a_i}\right)}{(|Y|+1)^{|X|}}$$

$$P(A|X, Y) = \frac{P(A,X|Y)}{P(X|Y)}$$

$$= \frac{\prod_{i=1}^{|X|} P\left(x_i|y_{a_i}\right)}{\prod_{i=1}^{|X|} \sum_{j=0}^{|Y|} P(x_i|y_j)}$$

$$= \prod_{i=1}^{|X|} \frac{P\left(x_i|y_{a_i}\right)}{\sum_{j=0}^{|Y|} P(x_i|y_j)}$$

EM training

After knowing $P(A|X,Y)$ and $P(A,X|Y)$, we can define the Q-function for sentence translation pair (X,Y):

$$\begin{aligned} Q\left(\Theta,\Theta^{t}\right) &= \sum_{A} P(A|X,Y,\Theta^{t}) \log P(A,X|Y,\Theta) \\ &= \sum_{A} P(A|X,Y,\Theta^{t}) \log \frac{\prod_{i=1}^{|X|} P\left(x_i|y_{a_i},\Theta\right)}{(|Y|+1)^{|X|}} \end{aligned}$$

EM training

- Maximizing $Q(\theta, \theta^{+})$ over data

$$\Lambda = \sum_A P(A|X,Y) \log \frac{\Pi_{i=1}^{|X|} P\left(x_i \middle| y_{a_i}\right)}{(|Y|+1)^{|X|}} + \sum_y \lambda^y (\sum_X P(X|y) - 1)$$

$$\frac{\partial \Lambda}{\partial P(X|Y)} = \sum_A P(A|X,Y) \cdot \frac{\partial \sum_{i=1}^{|X|} \log P\left(x_i \middle| y_{a_i}\right)}{\partial P(x|y)} + \lambda^y$$

$$= \sum_A P(A|X,Y) \sum_{i=1}^{|X|} \frac{\delta(x, x_i)\delta\left(y, y_{a_i}\right)}{P(x|y)} + \lambda = 0$$

$$\Longrightarrow P(x|y) \propto \sum_A P(A|X,Y) \sum_{i=1}^{|X|} \delta(x, x_i)\delta\left(y, y_{a_i}\right)$$

EM training

- The expected alignment between a word translation pair

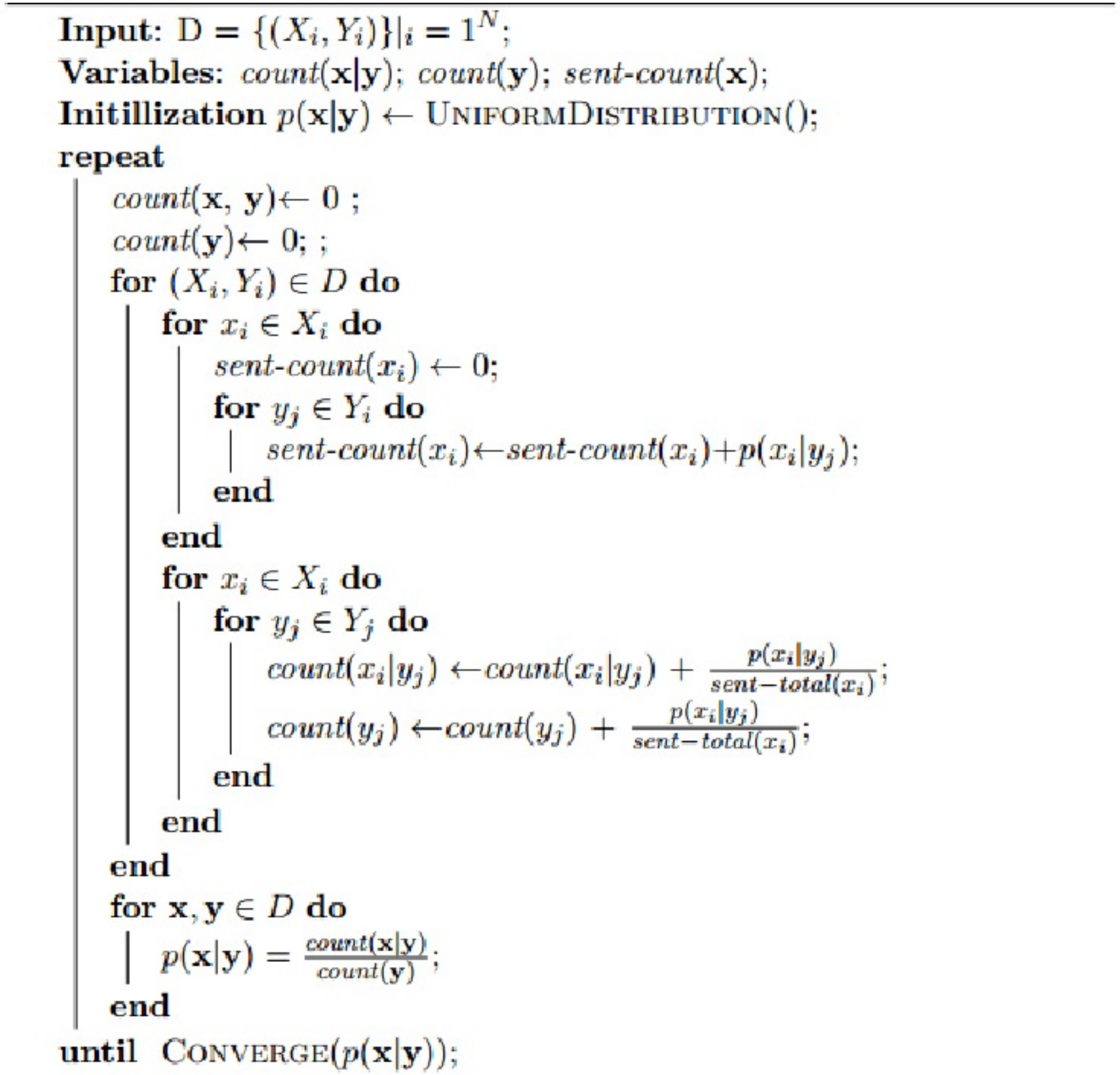
$$\text{EXPECTEDALIGN}\ (x, y, X, Y)$$

$$= \sum_{A} P(A|X, Y) \cdot \sum_{k=1}^{|X|} \delta(x, x_k)\delta(y, y_{a_k})$$

$$= \frac{P(x|y)}{\sum_{j=0}^{|Y|} P(x|y_j)} \sum_{i=1}^{|X|} \delta(x, x_i) \sum_{j=0}^{|Y|} \delta(y, y_j)$$

EXPECTEDALIGN (x, y, X, Y) represents a soft count.

IBM model 1



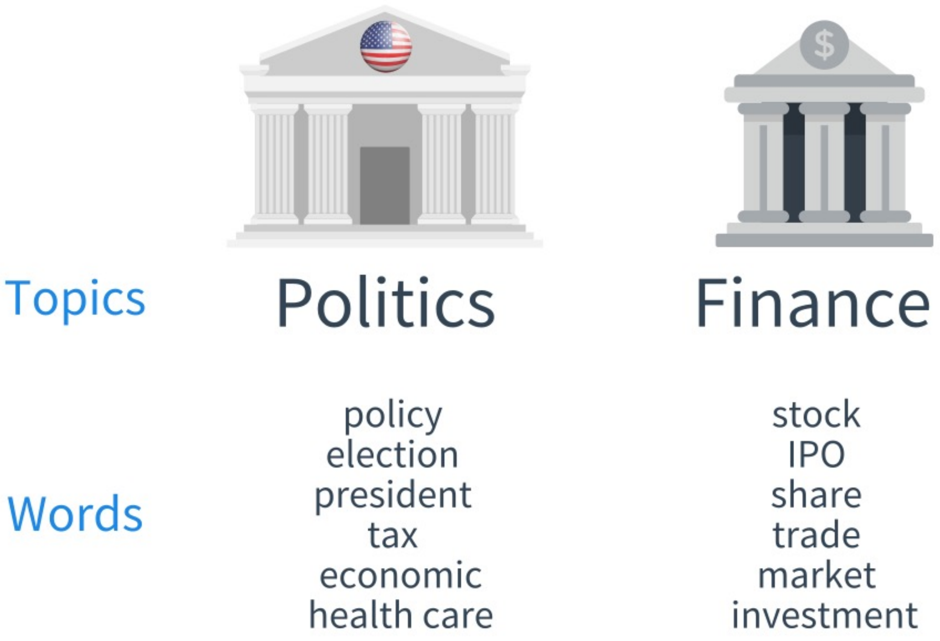
```
Input: $D = \{(X_i, Y_i)\}|_i = 1^N$;
Variables: $count(\mathbf{x}|\mathbf{y})$; $count(\mathbf{y})$; $sent\text{-}count(\mathbf{x})$;
Initillization $p(\mathbf{x}|\mathbf{y}) \leftarrow$ UniformDistribution();
repeat
    $count(\mathbf{x}, \mathbf{y}) \leftarrow 0$ ;
    $count(\mathbf{y}) \leftarrow 0$; ;
    for $(X_i, Y_i) \in D$ do
        for $x_i \in X_i$ do
            $sent\text{-}count(x_i) \leftarrow 0$;
            for $y_j \in Y_i$ do
                $sent\text{-}count(x_i) \leftarrow sent\text{-}count(x_i) + p(x_i|y_j)$;
            end
        end
        for $x_i \in X_i$ do
            for $y_j \in Y_j$ do
                $count(x_i|y_j) \leftarrow count(x_i|y_j) + \frac{p(x_i|y_j)}{sent-total(x_i)}$;
                $count(y_j) \leftarrow count(y_j) + \frac{p(x_i|y_j)}{sent-total(x_i)}$;
            end
        end
    end
    for $\mathbf{x}, \mathbf{y} \in D$ do
        $p(\mathbf{x}|\mathbf{y}) = \frac{count(\mathbf{x}|\mathbf{y})}{count(\mathbf{y})}$;
    end
until Converge($p(\mathbf{x}|\mathbf{y})$);
```

Contents

WestlakeNLP

- 6.1 Expectation Maximization
  - 6.1.1 Introduction
  - 6.1.2 EM
  - 6.1.3 Relationships between MLE and EM
- 6.2 Using EM
  - 6.2.1 Unsupervised Naïve Bayes model
  - 6.2.2 IBM Model 1
  - **6.2.3 Probabilistic latent sentiment analysis**
  - 6.2.4 The relative advantages of generative models
- 6.3 EM theory
  - 6.3.1 EM and KL-Divergence
  - 6.3.2 EM derivation using numerical optimization

Probabilistic Latent Semantic Analysis



Topics	Politics	Finance
Words	policy election president tax economic health care	stock IPO share trade market investment

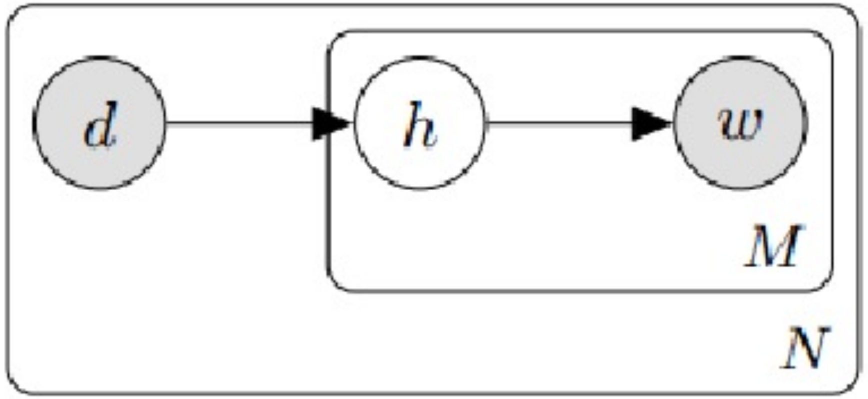
PLSA is a generative model for document semantic analysis.

Topics are hidden variables .

- Document-topic distribution $P(h|d_i)$
- Topic-word distribution $P(w|h)$

Probabilistic Latent Semantic Analysis

WestlakeNLP



For every document d, for every position l:

1. select a document d_i from $P(d_i)$

2. generate a topic h_j from $P(h_j|d)$

3. generate a word w_j from $P(w|h_j)$

Probabilistic Latent Semantic Analysis

- Model target $P(w,h|d) = P(h|d)P(w|h,d)$
- Hidden variable posteria probability
- $P(h|d_i,w,\Theta^t) = \frac{P\left(h,d_i,w\middle|\Theta^t\right)}{P\left(d_i,w\middle|\Theta^t\right)} = \frac{P\left(h\middle|d_i,\Theta^t\right)P\left(w\middle|h,\Theta^t\right)}{\sum_{h'} P\left(h'\middle|d_i,\Theta^t\right)P\left(w\middle|h',\Theta^t\right)}$
- The Q-function:

$$Q(\Theta,\Theta^t) = \sum_{i=1}^{N}\sum_{w\in d_i}\sum_h P(h|d_i,w,\Theta^t)logP(h,d_i,w|\Theta)$$

$$= \sum_{i=1}^{N}\sum_{w\in V} C(w,d_i)\sum_h P(h|d_i,w,\Theta^t)[logP(h|d_i,\Theta) + logP(w|h,\Theta)]$$

- Constraints:

$$\sum_h P(h|d_i,\Theta) = 1\ ,\ \sum_w P(w|h,\Theta) = 1$$

Probabilistic Latent Semantic Analysis

- Define a Lagrangian function

$$\Lambda(\Theta, \lambda)$$
$$= Q(\Theta, \Theta^t) - \sum_i \lambda_{d_i} \left(\sum_h P(h|d_i, \Theta) - 1 \right) - \sum_h \lambda_h \left(\sum_w P(w|h, \Theta) - 1 \right)$$

- Consider $\frac{\partial \Lambda(\Theta, \lambda)}{\partial P(h|d_i, \Theta)} = 0$ and $\sum_h P(h|d_i, \Theta) - 1 = 0$

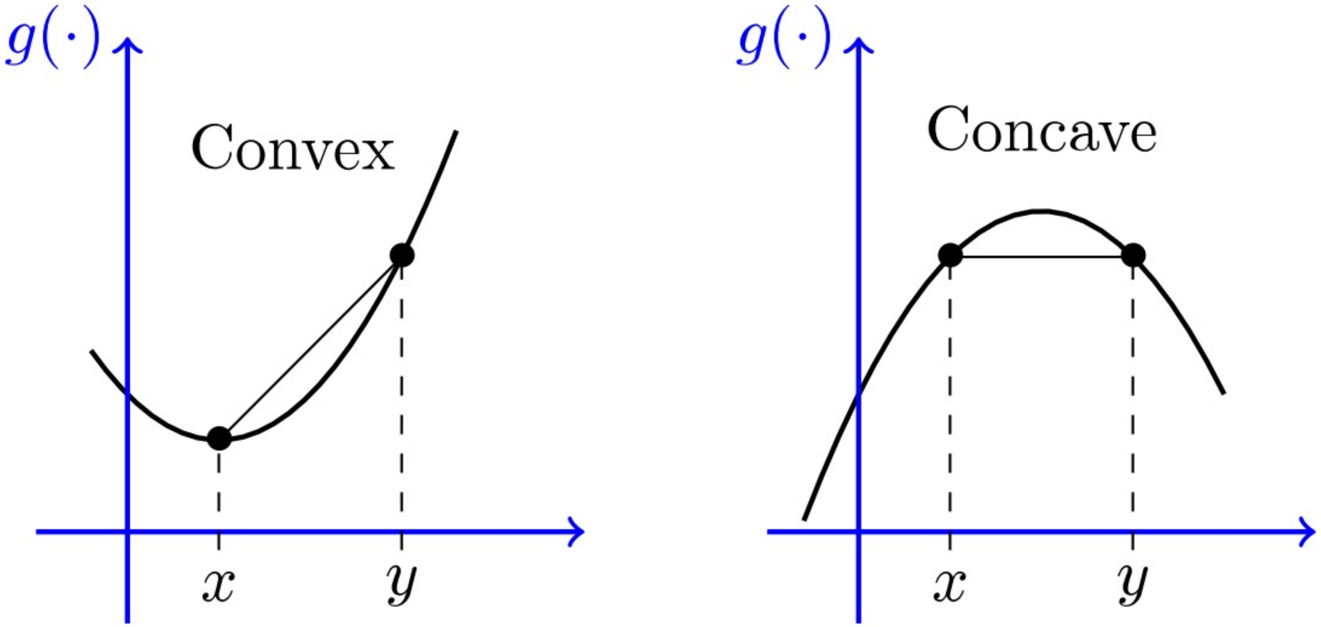
$$P(h|d_i, \Theta) = \frac{\sum_{w \in V} C(w, d_i) P(h|d_i, w, \Theta^t)}{\sum_{w \in V} C(w, d_i)}$$

$$P(w|h, \Theta) = \frac{\sum_{i=1}^{N} C(w, d_i) P(h|d_i, w, \Theta^t)}{\sum_{i=1}^{N} \sum_{w \in V} C(w, d_i) P(h|d_i, w, \Theta^t)}$$

Contents

- 6.1 Expectation Maximization
  - 6.1.1 Introduction
  - 6.1.2 EM
  - 6.1.3 Relationships between MLE and EM
- 6.2 Using EM
  - 6.2.1 Unsupervised Naïve Bayes model
  - 6.2.2 IBM Model 1
  - 6.2.3 Probabilistic latent sentiment analysis
  - 6.2.4 The relative advantages of generative models
- 6.3 EM theory
  - **6.3.1 EM and KL-Divergence**
  - 6.3.2 EM derivation using numerical optimization

Jensen inequality



- for convex functions, $\mathrm{E}\big(g(\mu)\big) \geq g\big(\mathrm{E}(\mu)\big)$,
- for concave functions, $\mathrm{E}\big(g(\mu)\big) \leq g\big(\mathrm{E}(\mu)\big)$.

Using Jensen inequality

- $L(\Theta) = log\sum_h P(O,h|\Theta) \geq \sum_h P_C(h)\log\frac{P(O,h|\Theta)}{P_C(h)} = F(\Theta, Pc),$

  $F(\Theta, Pc)$ is a lower bound of $L(\Theta)$

- Also, $F(\Theta, Pc) = L(\Theta) - KL\left(P_C(h), P(h|O,\Theta)\right)$

  KL-divergence is always non-negative

  $KL(P,Q)$ is zero if and only if $P = Q$

- To make the bound as tight as possible, $Pc(h) = P(h|O,\Theta)$

  In this scenario, $F(\Theta, Pc) = L(\Theta)$

Contents

- 6.1 Expectation Maximization
  - 6.1.1 Introduction
  - 6.1.2 EM
  - 6.1.3 Relationships between MLE and EM
- 6.2 Using EM
  - 6.2.1 Unsupervised Naïve Bayes model
  - 6.2.2 IBM Model 1
  - 6.2.3 Probabilistic latent sentiment analysis
  - 6.2.4 The relative advantages of generative models
- 6.3 EM theory
  - 6.3.1 EM and KL-Divergence
  - **6.3.2 EM derivation using numerical optimization**

EM derivation using numerical optimization

Another way to maximizes $F(\Theta, P_C)$

Coordinate ascent

- Expectation step.

  finds an optimum distribution $Pc(H)$ that maximizes $F(\Theta^t, Pc)$

- Maximization step.

  finds the optimal Θ^{t+1} for $F(\Theta, P_C^{t+1})$ using P_C^{t+1}.

EM derivation using numerical optimization

- **Convergence.**

  after every iteration of E-step and M-step, $L(\Theta^{t+1}) - L(\Theta) \geq 0$,

  $L(\Theta)$ is a monotonically increasing function, EM is guaranteed to

  converge to local optimums.

Summary

- The concept of hidden variables
- Expectation Maximization (EM) algorithm
- The correlation between EM and MLE for training probabilistic models
- EM for unsupervised text classification
- IBM model 1 for statistical machine translation
- Probabilistic latent semantic allocation