Recall: Computing Likelihood

| Problem 1 (Likelihood): | Given an HMM $\lambda = (A, B)$ and an observation sequence $O$, determine the likelihood $P(O|\lambda)$. |
|------------------------|--------------------------------------------------------------------------------------------------|
| Problem 2 (Decoding):  | Given an observation sequence $O$ and an HMM $\lambda = (A, B)$, discover the best hidden state sequence $Q$. |
| Problem 3 (Learning):  | Given an observation sequence $O$ and the set of states in the HMM, learn the HMM parameters $A$ and $B$. |

**Computing Likelihood:** Given an HMM $\lambda = (A, B)$ and an observation sequence $O$, determine the likelihood $P(O|\lambda)$.

**Use the Forward Algorithm**
Recall: Decoding best state sequence

| Problem 1 (Likelihood): | Given an HMM $\lambda = (A, B)$ and an observation sequence $O$, determine the likelihood $P(O|\lambda)$. |
|------------------------|-------------------------------------------------------------------------------------------------|
| Problem 2 (Decoding):  | Given an observation sequence $O$ and an HMM $\lambda = (A, B)$, discover the best hidden state sequence $Q$. |
| Problem 3 (Learning):  | Given an observation sequence $O$ and the set of states in the HMM, learn the HMM parameters $A$ and $B$. |

**Decoding:** Given as input an HMM $\lambda = (A, B)$ and a sequence of observations $O = o_1, o_2, \ldots, o_T$, find the most probable sequence of states $Q = q_1q_2q_3 \ldots q_T$.

Use the **Viterbi Algorithm**
Learning HMM Parameters

**Problem 1 (Likelihood):** Given an HMM \( \lambda = (A, B) \) and an observation sequence \( O \), determine the likelihood \( P(O|\lambda) \).

**Problem 2 (Decoding):** Given an observation sequence \( O \) and an HMM \( \lambda = (A, B) \), discover the best hidden state sequence \( Q \).

**Problem 3 (Learning):** Given an observation sequence \( O \) and the set of states in the HMM, learn the HMM parameters \( A \) and \( B \).

**Learning:** Given an observation sequence \( O \) and the set of possible states in the HMM, learn the HMM parameters \( A \) and \( B \).

Standard algorithm for HMM training: **Forward-backward** or **Baum-Welch** algorithm

Before moving on to **Baum-Welch**, what is the **Expectation Maximization** algorithm?
**EM Algorithm: Fitting Parameters to Data**

Parameter $\theta$ determines $\Pr(x, z; \theta)$ where $x$ is observed and $z$ is hidden.

Observed data: i.i.d samples $x_i, i=1, \ldots, N$

Goal: Find $\arg \max \mathcal{L}(\theta)$ where $\mathcal{L}(\theta) = \sum_{i=1}^{N} \log \Pr(x_i; \theta)$

Initial parameters: $\theta^0$

Iteratively compute $\theta^\ell$ as follows:

$$Q(\theta, \theta^{\ell-1}) = \sum_{i=1}^{N} \sum_{z} \Pr(z|x_i; \theta^{\ell-1}) \log \Pr(x_i, z; \theta)$$

$$\theta^\ell = \arg \max_\theta Q(\theta, \theta^{\ell-1})$$

Estimate $\theta^\ell$ cannot get worse over iterations because for all $\theta$:

$$\mathcal{L}(\theta) - \mathcal{L}(\theta^{\ell-1}) \geq Q(\theta, \theta^{\ell-1}) - Q(\theta^{\ell-1}, \theta^{\ell-1})$$

EM is guaranteed to converge to a local optimum [Wu83]
The following sequence is observed: “HH, TT, HH, TT, HH”

How do you estimate $\rho_1$, $\rho_2$ and $\rho_3$?
Coin example to illustrate EM

Recall, for partially observed data, the likelihood is given by:

\[ \mathcal{L}(\theta) = \sum_{i=1}^{N} \log \Pr(x_i; \theta) = \sum_{i=1}^{N} \log \sum_{z} \Pr(x_i, z; \theta) \]

where, for the coin example:

- each observation \( x_i \in \mathcal{X} = \{\text{HH,HT,TH,TT}\} \)
- the hidden variable \( z \in \mathcal{Z} = \{\text{H,T}\} \)
Coin example to illustrate EM

Recall, for partially observed data, the likelihood is given by:

\[ \mathcal{L}(\theta) = \sum_{i=1}^{N} \log \Pr(x_i; \theta) = \sum_{i=1}^{N} \log \sum_{z} \Pr(x_i, z; \theta) \]

\[ \Pr(x, z; \theta) = \Pr(x|z; \theta) \Pr(z; \theta) \]

where \( \Pr(z; \theta) = \begin{cases} \rho_1 & \text{if } z = H \\ 1 - \rho_1 & \text{if } z = T \end{cases} \)

\[ \Pr(x|z; \theta) = \begin{cases} \rho_2 (1 - \rho_2)^t & \text{if } z = H \\ \rho_3 (1 - \rho_3)^t & \text{if } z = T \end{cases} \]

\( h \) : number of heads, \( t \) : number of tails
Our observed data is: \{HH, TT, HH, TT, HH\}

Let’s use EM to estimate \( \theta = (\rho_1, \rho_2, \rho_3) \)

**[EM Iteration, E-step]**

Compute quantities involved in

\[
Q(\theta, \theta^{\ell-1}) = \sum_{i=1}^{N} \sum_{z} \gamma(z, x_i) \log \Pr(x_i, z; \theta)
\]

where \( \gamma(z, x) = \Pr(z \mid x; \theta^{\ell-1}) \)

i.e., compute \( \gamma(z, x_i) \) for all \( z \) and all \( i \)

Suppose \( \theta^{\ell-1} \) is \( \rho_1 = 0.3, \rho_2 = 0.4, \rho_3 = 0.6 \):

What is \( \gamma(H, HH) \)? \( = 0.16 \)

What is \( \gamma(H, TT) \)? \( = 0.49 \)
Our observed data is: \{HH, TT, HH, TT, HH\}

Let’s use EM to estimate \( \theta = (\rho_1, \rho_2, \rho_3) \)

**[EM Iteration, M-step]**

Find \( \theta \) which maximises

\[
Q(\theta, \theta^{\ell-1}) = \sum_{i=1}^{N} \sum_{z} \gamma(z, x_i) \log \Pr(x_i, z; \theta)
\]

- \( \rho_1 = \frac{\sum_{i=1}^{N} \gamma(H, x_i)}{N} \)
- \( \rho_2 = \frac{\sum_{i=1}^{N} \gamma(H, x_i)h_i}{\sum_{i=1}^{N} \gamma(H, x_i)(h_i + t_i)} \)
- \( \rho_3 = \frac{\sum_{i=1}^{N} \gamma(T, x_i)h_i}{\sum_{i=1}^{N} \gamma(T, x_i)(h_i + t_i)} \)

Coin example to illustrate EM
Coin example to illustrate EM

This was a very simple HMM (with observations from 2 states)

State remains the same after the first transition

$\gamma$ estimated the distribution of this state

More generally, will need the distribution of the state at each time step

EM for general HMMs: Baum-Welch algorithm (1972) predates the general formulation of EM (1977)
Baum-Welch Algorithm as EM

Observed data: $N$ sequences, $x_i = (x_{i1}, \ldots, x_{iT_i}), i=1 \ldots N$ where $x_{it} \in \mathbb{R}^d$

Parameters $\theta$: transition matrix $A$, observation probabilities $B$

[EM Iteration, E-step]
Compute quantities involved in $Q(\theta, \theta^{\ell-1})$

$\gamma_{i,t}(j) = \Pr(z_t = j \mid x_i; \theta^{\ell-1})$

$\xi_{i,t}(j,k) = \Pr(z_{t-1} = j, z_t = k \mid x_i; \theta^{\ell-1})$
Baum-Welch Algorithm as EM

Observed data: \( N \) sequences, \( x_i = (x_{i1}, \ldots, x_{iT_i}), i = 1 \ldots N \) where \( x_{it} \in \mathbb{R}^d \)

Parameters \( \theta \): transition matrix \( A \), observation probabilities \( B \)

**[EM Iteration, M-step]**
Find \( \theta \) which maximises \( Q(\theta, \theta^{(l-1)}) \)

\[
A_{j,k} = \frac{\sum_{i=1}^{N} \sum_{t=2}^{T_i} \xi_{i,t}(j, k)}{\sum_{i=1}^{N} \sum_{t=2}^{T_i} \sum_{k'} \xi_{i,t}(j, k')} 
\]

\[
B_{j,v} = \frac{\sum_{i=1}^{N} \sum_{t:x_{it}=v} \gamma_{i,t}(j)}{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j)} 
\]
Gaussian Observation Model

- So far we considered HMMs with *discrete* outputs.
- In acoustic models, HMMs output real valued vectors.
- Hence, observation probabilities are defined using probability density functions.
- A widely used model: Gaussian distribution

\[
\mathcal{N}(x | \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}
\]

- HMM emission/observation probabilities \( b_j(x) = \mathcal{N}(x | \mu_j, \sigma_j^2) \)
  where \( \mu_j \) is the mean associated with state \( j \) and \( \sigma_j^2 \) is its variance.
- For multivariate Gaussians, \( b_j(x) = \mathcal{N}(x | \mu_j, \Sigma_j) \) where \( \Sigma \) is the covariance associated with state \( j \).
BW for Gaussian Observation Model

Observed data: $N$ sequences, $x_i = (x_{i1}, \ldots, x_{iT_i}), i=1\ldots N$ where $x_{it} \in \mathbb{R}^d$

Parameters $\theta$: transition matrix $A$, observation prob. $B = \{(\mu_j, \Sigma_j)\}$ for all $j$

[EM Iteration, M-step]
Find $\theta$ which maximises $Q(\theta, \theta^{\ell-1})$

$A$ same as with discrete outputs

$$
\mu_j = \frac{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j)x_{it}}{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j)}
$$

$$
\Sigma_j = \frac{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j)(x_{it} - \mu_j)(x_{it} - \mu_j)^T}{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j)}
$$
Gaussian Mixture Model

- A single Gaussian observation model assumes that the observed acoustic feature vectors are unimodal
Unimodal

\[ \phi_{\mu,\sigma^2}(x) \]

- $\mu = 0, \sigma^2 = 0.2$,
- $\mu = 0, \sigma^2 = 1.0$,
- $\mu = 0, \sigma^2 = 5.0$,
- $\mu = -2, \sigma^2 = 0.5$.
Gaussian Mixture Model

• A single Gaussian observation model assumes that the observed acoustic feature vectors are unimodal

• More generally, we use a “mixture of Gaussians” to model multiple modes in the data
Mixture Models
Gaussian Mixture Model

- A single Gaussian observation model assumes that the observed acoustic feature vectors are unimodal.

- More generally, we use a “mixture of Gaussians” to model multiple modes in the data.

- Instead of $b_j(x) = \mathcal{N}(x \mid \mu_j, \Sigma_j)$ in the single Gaussian case, $b_j(x)$ now becomes:

$$b_j(x) = \sum_{m=1}^{M} c_{jm} \mathcal{N}(x \mid \mu_{jm}, \Sigma_{jm})$$

where $c_{jm}$ is the mixing probability for Gaussian component $m$ of state $j$.

$$\sum_{m=1}^{M} c_{jm} = 1, \quad c_{jm} \geq 0$$
BW for Gaussian Mixture Model

Observed data: \( N \) sequences, \( x_i = (x_{i1}, \ldots, x_{iT_i}), i=1\ldots N \) where \( x_{it} \in \mathbb{R}^d \)
Parameters \( \theta \): transition matrix \( A \), observation prob. \( B = \{ (\mu_{jm}, \Sigma_{jm}, c_{jm}) \} \) for all \( j, m \)

[EM Iteration, M-step]
Find \( \theta \) which maximises \( Q(\theta, \theta^{\ell-1}) \)

\[
\mu_{jm} = \frac{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j, m)x_{it}}{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j, m)}
\]

\[
\Sigma_{jm} = \frac{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j, m)(x_{it} - \mu_{jm})(x_{it} - \mu_{jm})^T}{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j, m)}
\]

\( \gamma_{i,t}(j) = Pr(q_i=j|x_i) \)

\[
c_{jm} = \frac{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j, m)}{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j)}
\]

Mixing probabilities
ASR Framework: Acoustic Models

- Acoustic models are estimated using training data: \(\{x_i, y_i\}, i=1\ldots N\) where \(x_i\) corresponds to a sequence of acoustic feature vectors and \(y_i\) corresponds to a sequence of words.

- For each \(x_i, y_i\), a composite HMM is constructed using the HMMs that correspond to the triphone sequence in \(y_i\).

```
“Hello world”
```

```
“sil hh ah l ow w er l d sil”
```

```
“sil sil/hh/ah hh/ah/l ah/l/ow l/ow/w er/w l/e r/d er/l/d l/d/sil sil”
```
Acoustic models are estimated using training data: \(\{x_i, y_i\}, i=1\ldots N\) where \(x_i\) corresponds to a sequence of acoustic feature vectors and \(y_i\) corresponds to a sequence of words.

For each \(x_i, y_i\), a composite HMM is constructed using the HMMs that correspond to the triphone sequence in \(y_i\).

Parameters of these composite HMMs are the parameters of the constituent triphone HMMs.

These parameters are fit to the acoustic data \(\{x_i\}, i=1\ldots N\) using the Baum-Welch algorithm (EM).
Baum Welch: In summary

[Every EM Iteration]
Compute $\theta = \{ A_{jk}, (\mu_{jm}, \Sigma_{jm}, c_{jm}) \}$ for all $j,k,m$

$$A_{j,k} = \frac{\sum_{i=1}^{N} \sum_{t=2}^{T_i} \xi_{i,t}(j, k)}{\sum_{i=1}^{N} \sum_{t=2}^{T_i} \sum_{k'} \xi_{i,t}(j, k')}$$

$$\mu_{jm} = \frac{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j, m) x_{it}}{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j, m)}$$

$$\Sigma_{jm} = \frac{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j, m)(x_{it} - \mu_{jm})(x_{it} - \mu_{jm})^T}{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j, m)}$$

$$c_{jm} = \frac{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j, m)}{\sum_{i=1}^{N} \sum_{t=1}^{T_i} \gamma_{i,t}(j)}$$

How do we efficiently compute these quantities? Next class!