# Hidden Markov Model I

September 12, 2022

# Motivating Example

- Assume there are two types of weather “Sunny” and “Rainy”. We assume, *a prior*, that their probabilities are 0.7 and 0.3, e.g., $Pr(Sunny) = 0.7$, $Pr(Rainy) = 0.3$.
- Every morning, you do two things: walking dogs (“W”) or reading (“R”). Assume the following conditional probabilities:

$$Pr(W|Sunny) = 0.8, Pr(R|Sunny) = 0.2.$$
$$Pr(W|Rainy) = 0.2, Pr(R|Rainy) = 0.8.$$

- Assume we know your morning activity for a number of days: {W, W, R, R, W, W, R, W, W, W}, but don’t know the weather. How can we estimate the weather condition for each day?

# Motivating Example (cont.)

- Using Bayes' rule, we can compute the following quantity for each day:

$$Pr(Sunny|W) = \frac{Pr(W|Sunny)Pr(Sunny)}{Pr(W|Sunny)Pr(Sunny) + Pr(W|Rainy)Pr(Rainy)}$$
$$= \frac{0.8 * 0.7}{0.8 * 0.7 + 0.2 * 0.3} = 0.9$$
$$Pr(Sunny|R) = \ldots\ldots$$

- However, this assumes **independence** of observations and completely ignores the connections between weather changes, e.g., probability of today is Sunny given yesterday is Sunny, etc.
- With the consideration the connections between weather changes, today's weather $Pr(Sunny|W)$ should also depend on yesterday's weather, in addition to the W/R status.
- Such an approach can be formalized by a "hidden Markov model" (HMM).

- Assume we observe sequential data $\boldsymbol{u} = \{u_1, u_2, \ldots, u_T\}$ (your morning activities).
- $\boldsymbol{u}$ is generated by a chain of **hidden**, unobserved states: $\boldsymbol{s} = \{s_1, s_2, \ldots, s_T\}$.
- Each $s_t$ can take $M$ states, with "**initial probability**" $\pi_k, k = 1, \ldots, M$: $Pr(s_1 = k) = \pi_k$, $\sum_k \pi_k = 1$.
- The distribution of $\boldsymbol{u}$ conditional on $\boldsymbol{s}$ is represented as $b_k(u)$: $u_t | s_t = k \sim b_k(u_t)$. This is called "**emission probability**".
- The changes of states between consecutive hidden state is specified by "**transition probability**": $a_{k,l} = Pr(s_{t+1} = l | s_t = k)$. Or you can write this as $a_{k \to l}$.
- Assume the underlying states follow a **Markov chain**, that is, given present, the future is independent of the past:

$$Pr(s_{t+1} | s_t, s_{t-1}, \ldots, s_1) = Pr(s_{t+1} | s_t).$$

To summarize: a HMM has observed data $\boldsymbol{u}$, missing data $\boldsymbol{s}$, and parameters $\boldsymbol{\lambda} = \{\pi_k, b_k(u), a_{k,l}\}$.

- The possible states are included in a finite discrete set: $\{E_1, E_2, \ldots, E_M\}$.
- From time $t$ to $t+1$, make stochastic movement from one state to another.
- Markov Property: the state of $s_{t+1}$ only depends on the state of $s_t$, not the states before time $t$:

$$Pr(s_{t+1}|s_t, s_{t-1}, \ldots, s_1) = Pr(s_{t+1}|s_t).$$

- Time-homogeneous transition probabilities property: $P(s_{t+1}|s_t)$ independent of $t$.
- Denote the transition probability matrix by $\mathbf{A}$. Define N step transition as: $a_{k,l}(N) = Pr(s_{t+N} = l|s_t = k)$. It can be shown that $\mathbf{A}(N) = \mathbf{A}^N$.

# What can HMM do

A HMM can answer following questions:

- Parameter estimation: estimate the initial/emission/transition probabilities. $\hat{\boldsymbol{\lambda}} = \text{argmax}_{\lambda} Pr(\boldsymbol{u}|\boldsymbol{\lambda})$.
- Estimate the probabilities of the underlying states given the observations: $Pr(\boldsymbol{s}|\boldsymbol{u})$.
- The most likely path: given the observed data, what are the most likely underlying states for all observations: $\hat{\boldsymbol{s}} = \text{argmax}_{s} Pr(\boldsymbol{s}|\boldsymbol{\lambda}, \boldsymbol{u})$.
- Predict future, e.g., $Pr(u_{t+1}|\boldsymbol{u}, \hat{\boldsymbol{\lambda}})$.

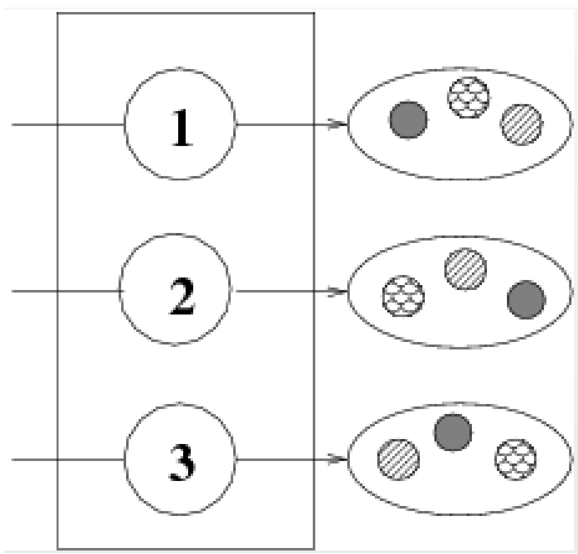
Examples of HMM applications:

- Speech recognition.
- DNA sequence analysis, e.g., gene finding, sequence alignment.
- Financial time series data.

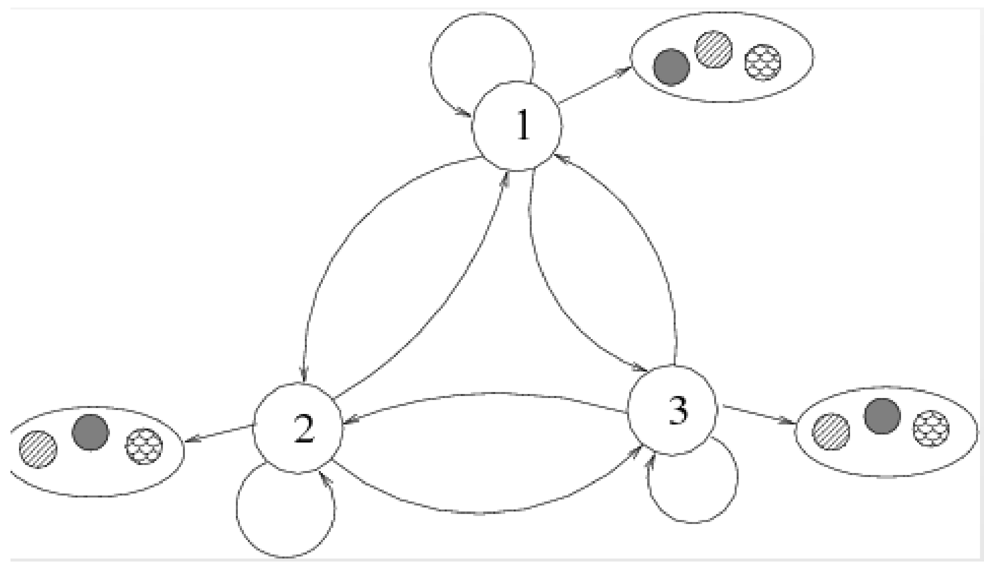
# HMM vs. mixture model

- There's close connection between a HMM and a mixture model: both have hidden states/group assignment, initial and emission probabilities.
- Difference is that mixture model assumes **independent** observations, HMM assumes sequential observation with transition probability.

**Mixture model**




**HMM**



# Some results for a HMM

According to Markov property, we have:

- Joint probability of hidden states:

$$\begin{aligned} P(s_1, s_2, \ldots, s_T) &= P(s_1)P(s_2|s_1)\ldots P(s_T|s_{T-1}) \\ &= \pi_{s_1} a_{s_1,s_2} \ldots a_{s_{T-1},s_T} \end{aligned}$$

- Conditional on the states, the observations are independent of each other:

$$P(u_i, u_j|\boldsymbol{s}) = P(u_i|\boldsymbol{s})P(u_j|\boldsymbol{s})$$

So the joint probability of observations, given hidden states is:

$$P(\boldsymbol{u}|\boldsymbol{s}) = \prod_{i=1}^{T} P(u_i|s_i) = \prod_{i=1}^{T} b_{s_i}(u_i)$$

**Note:** marginally the observations are NOT independent.

- Joint probability of hidden states and observed data

$$
\begin{aligned}
P(\boldsymbol{u}, \boldsymbol{s}) &= P(\boldsymbol{s})P(\boldsymbol{u}|\boldsymbol{s}) \\
&= [P(s_1)p(u_1|s_1)][P(s_2|s_1)P(u_2|s_2)]\dots[P(s_T|s_{T-1})P(u_T|s_T)] \\
&= \pi_{s_1} b_{s_1}(u_1) a_{s_1,s_2} b_{s_2}(u_2) a_{s_2,s_3} b_{s_3}(u_3)\dots a_{s_{T-1},s_T} b_{s_T}(u_T)
\end{aligned}
$$

- Marginal probability of observed data:

$$
\begin{aligned}
P(\boldsymbol{u}) &= \sum_{\boldsymbol{s}} P(\boldsymbol{s})P(\boldsymbol{u}|\boldsymbol{s}) \\
&= \sum_{\boldsymbol{s}} \pi_{s_1} b_{s_1}(u_1) a_{s_1,s_2} b_{s_2}(u_2) a_{s_2,s_3}\dots a_{s_{T-1},s_T} b_{s_T}(u_T)
\end{aligned}
$$

# Model Estimation

- First need to make parametric assumption of the emission probabilities $b_k(u)$.
- In this lecture, we assume $b_k(u)$ is normal, e.g., $b_k(u) = N(u : \mu_k, \sigma_k^2)$, then the model parameters to be estimated are:

$$\boldsymbol{\lambda} = \{\pi_k, \mu_k, \sigma_k, a_{k,l} : k, l = 1, \ldots, M\}$$

- One can obtain the MLEs for $\boldsymbol{\lambda}$ from the marginal probability of observed data. However it's very difficult because the marginal probability involves summing over all possible underlying states ($\sum_{\boldsymbol{s}}$).
- Clever algorithm was invented to solve the problem.

- Define $L_k(t)$ to be the conditional probability of being in state $k$ at position $t$ given the observed data $\boldsymbol{u}$:

$$L_k(t) = P(s_t = k|\boldsymbol{u})$$

- Define $H_{k,l}(t)$ be the conditional probability of being in state $k$ at position $t$ and being in state $l$ at position $t+1$ (i.e., seeing a transition from $k$ to $l$ at $t$), given the observed data $\boldsymbol{u}$:

$$H_{k,l}(t) = P(s_t = k, s_{t+1} = l|\boldsymbol{u})$$

- Note that $L_k(t) = \sum_{l=1}^{M} H_{k,l}(t)$, $\sum_{k=1}^{M} L_k(t) = 1$.

- Then the parameters can be estimated by EM:
  - E-step: Compute $L_k(t)$ and $H_{k,l}(t)$ given current parameters.
  - M-step: update parameters:

$$
\begin{aligned}
\mu_k &= \frac{\sum_{t=1}^{T} L_k(t) u_t}{\sum_{t=1}^{T} L_k(t)} \\
\sigma_k^2 &= \frac{\sum_{t=1}^{T} L_k(t)(u_t - \mu_k)^2}{\sum_{t=1}^{T} L_k(t)} \\
a_{k,l} &= \frac{\sum_{t=1}^{T-1} H_{k,l}(t)}{\sum_{t=1}^{T-1} L_k(t)} \\
\pi_k &= L_k(1)
\end{aligned}
$$

- Derivation steps are similar to that in M-component normal mixture model (try it yourself). The new items are the transition probabilities.

## Comparison with Normal mixture model



- In the M-step, $L_k(t)$ plays the role of the expected value for the missing data (group assignment).
  - In the mixture model (assuming independent observations), the state given the observation is $p_{t,k} = P(s_t = k|u_t)$.
  - In a HMM, $L_k(t) = P(s_t = k|u_1, u_2, \ldots, u_T)$.
- If one ignores the connections among observations, e.g., $s_t$'s are independent and thus $u_t$'s are iid, then $L_k(t) = p_{t,k}$, and HMM reduce to a M-component Normal mixture model.
- In a mixture model, $s_t$ only depends on $u_t$ because observations are independent.
- In a HMM, $s_t$ depends on the entire sequence of observations because of the underlying Markov process.

The forward-backward algorithm is designed to efficiently compute:

$$L_k(t) = P(s_t = k|\boldsymbol{u})$$
$$H_{k,l}(t) = P(s_t = k, s_{t+1} = l|\boldsymbol{u})$$

- Define the **forward probability** $\alpha_k(t)$ as the **joint probability** of observing the first $t$ data $u_i, i = 1, \ldots, t$ and being in state $k$ at time $t$:

$$\alpha_k(t) = P(u_1, u_2, \ldots, u_t, s_t = k)$$

- The forward probability can be computed recursively:

$$\alpha_k(1) = \pi_k b_k(u_1) \quad 1 \le k \le M$$
$$\alpha_k(t) = b_k(u_t) \sum_{l=1}^{M} \alpha_l(t-1) a_{l,k} \quad 1 < t \le T, 1 \le k \le M.$$

# Derivation of forward probability calculation 

$$
\begin{aligned}
\alpha_k(t) &= P(u_1, u_2, \ldots, u_t, s_t = k) \\
&= \sum_{l=1}^{M} P(u_1, u_2, \ldots, u_t, s_t = k, s_{t-1} = l) \\
&= \sum_{l=1}^{M} P(u_1, u_2, \ldots, u_{t-1}, s_{t-1} = l) P(u_t, s_t = k \mid u_1, u_2, \ldots, u_{t-1}, s_{t-1} = l) \\
&= \sum_{l=1}^{M} \alpha_l(t-1) P(u_t, s_t = k \mid s_{t-1} = l) \\
&= \sum_{l=1}^{M} \alpha_l(t-1) P(u_t \mid s_t = k, s_{t-1} = l) P(s_t = k \mid s_{t-1} = l) \\
&= \sum_{l=1}^{M} \alpha_l(t-1) P(u_t \mid s_t = k) P(s_t = k \mid s_{t-1} = l) \\
&= b_k(u_t) \sum_{l=1}^{M} \alpha_l(t-1) a_{l,k}
\end{aligned}
$$

# Backward probability

- Define the **backward probability** $\beta_k(t)$ as the **conditional probability** of observing the data after time $t$, $u_i, i = t+1, \ldots, T$, given the state at time $t$ is $k$.

$$\beta_k(t) = P(u_{t+1}, \ldots, u_T \mid s_t = k) \quad 1 \le t \le T-1$$

- Again, the backward probability can be computed by following recursive formula:

$$\beta_k(T) = 1$$

$$\beta_k(t) = \sum_{l=1}^{M} a_{k,l}\, b_l(u_{t+1})\, \beta_l(t+1) \quad 1 \le t < T$$

# Derivation of backward probability calculation

$$
\begin{aligned}
\beta_k(t) &= P(u_{t+1}, \ldots, u_T \mid s_t = k) \\
&= \sum_{l=1}^{M} P(u_{t+1}, \ldots, u_T, s_{t+1} = l \mid s_t = k) \\
&= \sum_{l=1}^{M} P(u_{t+1}, \ldots, u_T \mid s_{t+1} = l, s_t = k)\; P(s_{t+1} = l \mid s_t = k) \\
&= \sum_{l=1}^{M} P(u_{t+1}, \ldots, u_T \mid s_{t+1} = l)\; a_{k,l} \\
&= \sum_{l=1}^{M} P(u_{t+2}, \ldots, u_T \mid s_{t+1} = l, u_{t+1})\; P(u_{t+1} | s_{t+1} = l)\; a_{k,l} \\
&= \sum_{l=1}^{M} P(u_{t+2}, \ldots, u_T \mid s_{t+1} = l)\; b_l(u_{t+1})\; a_{k,l} \\
&= \sum_{l=1}^{M} a_{k,l}\; b_l(u_{t+1})\; \beta_l(t+1)
\end{aligned}
$$

# Compute $L_k(t)$

Compute $L_k(t)$ using forward and backward probabilities:

$$L_k(t) \equiv P(s_t = k \mid \boldsymbol{u}) = \frac{P(\boldsymbol{u}, s_t = k)}{P(\boldsymbol{u})} = \frac{\alpha_k(t)\, \beta_k(t)}{P(\boldsymbol{u})}$$

Proof:

$$\begin{aligned} P(\boldsymbol{u}, s_t = k) &= P(u_1, \ldots, u_T, s_t = k) \\ &= P(u_1, \ldots, u_t, s_t = k)\, P(u_{t+1}, \ldots, u_T \mid u_1, \ldots, u_t, s_t = k) \\ &= P(u_1, \ldots, u_t, s_t = k)\, P(u_{t+1}, \ldots, u_T \mid s_t = k) \\ &= \alpha_k(t)\, \beta_k(t) \end{aligned}$$

# Compute $H_{k,l}(t)$

Compute $H_{k,l}(t)$ using forward and backward probabilities:

$$H_{k,l}(t) = P(s_t = k, s_{t+1} = l | \boldsymbol{u}) = \frac{P(s_t = k, s_{t+1} = l, \boldsymbol{u})}{P(\boldsymbol{u})}$$

$$= \frac{1}{P(\boldsymbol{u})} \alpha_k(t)\, a_{k,l}\, b_l(u_{t+1})\, \beta_l(t+1)$$

Proof:

$$\begin{aligned}
P(s_t = k, s_{t+1} = l, \boldsymbol{u}) =\ & P(u_1, \ldots, u_t, \ldots, u_T, s_t = k, s_{t+1} = l) \\
=\ & P(u_1, \ldots, u_t, s_t = k) P(u_{t+1}, s_{t+1} = l \mid s_t = k, u_1, \ldots, u_t) \\
& P(u_{t+2}, \ldots, u_T \mid s_{t+1} = l, s_t = k, u_1, \ldots, u_{t+1}) \\
=\ & \alpha_k(t) P(u_{t+1}, s_{t+1} = l \mid s_t = k) P(u_{t+2}, \ldots, u_T \mid s_{t+1} = l) \\
=\ & \alpha_k(t) P(s_{t+1} = l \mid s_t = k) P(u_{t+1} \mid s_{t+1} = l, s_t = k)\, \beta_l(t+1) \\
=\ & \alpha_k(t)\, a_{k,l}\, P(u_{t+1} \mid s_{t+1} = l)\, \beta_l(t+1) \\
=\ & \alpha_k(t)\, a_{k,l}\, b_l(u_{t+1})\, \beta_l(t+1)
\end{aligned}$$

## Compute $P(\boldsymbol{u})$ 

The marginal observed data likelihood is:

$$P(\boldsymbol{u}) = \sum_{k=1}^{M} \alpha_k(t)\beta_k(t), \forall t$$

Proof:

$$\begin{aligned}
P(\boldsymbol{u}) &= \sum_{k=1}^{M} P(u_1, \ldots, u_t, \ldots, u_T, s_t = k) \\
&= \sum_{k=1}^{M} P(u_1, \ldots, u_t, s_t = k) P(u_{t+1}, \ldots, u_T \mid s_t = k, u_1, \ldots, u_t) \\
&= \sum_{k=1}^{M} P(u_1, \ldots, u_t, s_t = k) P(u_{t+1}, \ldots, u_T \mid s_t = k) \\
&= \sum_{k=1}^{M} \alpha_k(t)\beta_k(t)
\end{aligned}$$

To summarize, estimation of model parameters requires iterating following steps, under the current estimates of parameters:

1. Compute the forward and backward probabilities (two matrices of dimension $M \times T$):

$$\alpha_k(1) = \pi_k b_k(u_1) \quad 1 \leq k \leq M$$

$$\alpha_k(t) = b_k(u_t) \sum_{l=1}^{M} \alpha_l(t-1) a_{l,k} \quad 1 < t \leq T, 1 \leq k \leq M.$$

$$\beta_k(T) = 1$$

$$\beta_k(t) = \sum_{l=1}^{M} a_{k,l}\, b_l(u_{t+1})\, \beta_l(t+1) \quad 1 \leq t < T$$

# The estimation algorithm (cont.)



2. Compute whole data likelihood: $P(\boldsymbol{u}) = \sum_{k=1}^{M} \alpha_k(t)\beta_k(t)$. This is independent of $t$. Can use $t = 1$ or $t = T$.

3. Compute $L_k(t)$ and $H_{k,l}(t)$ from forward/backward probabilities:

$$L_k(t) = \frac{\alpha_k(t)\,\beta_k(t)}{P(\boldsymbol{u})}$$

$$H_{k,l}(t) = \frac{1}{P(\boldsymbol{u})}\,\alpha_k(t)\,a_{k,l}\,b_l(u_{t+1})\,\beta_l(t+1)$$

4. Update parameters using $L_k(t)$ and $H_{k,l}(t)$ (assuming Normal emission probabilities):

$$\mu_k = \frac{\sum_{t=1}^{T} L_k(t) u_t}{\sum_{t=1}^{T} L_k(t)}, \quad \sigma_k^2 = \frac{\sum_{t=1}^{T} L_k(t)(u_t - \mu_k)^2}{\sum_{t=1}^{T} L_k(t)},$$

$$a_{k,l} = \frac{\sum_{t=1}^{T-1} H_{k,l}(t)}{\sum_{t=1}^{T-1} L_k(t)}, \quad \pi_k = L_k(1)$$

# Numerical consideration



Long HMM chain causes numerical problem.

- The computation of forward/backward matrices requires multiplying probabilities.
- Probabilities are quantities less than 1. Multiplying too many probabilities gives very small number, and will exceed the computer precision quickly and become 0 numerically.

Solution: the computation of forward/backward matrices are done in logarithm scale, i.e., instead of storing $P$, we store $\log P$.

- Running `exp(-1000)*exp(-1000)` gives 0 in R, but we know it's `exp(-2000)`.

However we also have sums of probabilities.

- We can't exp the numbers back, sum up, and then take log.
- $\log(e^a + e^b)$ will become negative infinity when $a$ or $b$ are negative number with large absolute values: try to run `log(exp(-1000) + exp(-1000))` in R.

Use the following trick to deal with the scenario:
$log(e^a + e^b) = log(e^a(1 + e^{b-a})) = a + log(1 + e^{b-a})$.

- It equals $b$ when $b >> a$, equals $a$ when $b << a$.
- When the values of $b$ and $a$ are close, the computation is numerically stable.

Following is an R implementation of the algorithm, which works for two vectors:

```
Raddlog <- function (a, b)
{
    result <- rep(0, length(a))
    idx1 <- a > b + 200
    result[idx1] <- a[idx1]
    idx2 <- b > a + 200
    result[idx2] <- b[idx2]
    idx0 <- !(idx1 | idx2)
    result[idx0] <- a[idx0] + log1p(exp(b[idx0] - a[idx0]))
    result
}
```

# Numerical consideration (cont.)



Some simple tests:

```
> log(exp(-100)+exp(-100))
[1] -99.30685
> Raddlog(-100, -100)
[1] -99.30685
```

```
> log(exp(-1000)+exp(-1000))
[1] -Inf
> Raddlog(-1000, -1000)
[1] -999.3069
```

```
> log(exp(-100)+exp(-1000))
[1] -100
> Raddlog(-100, -1000)
[1] -100
```

# Review

- HMM is used to model sequential data.
- Difference between HMM and mixture model: mixture model assumes iid observations, HMM assumes underlying sequential correlation among hidden states.
- Important components in a HMM: initial, emission and transition probabilities.
- Goals of HMM: estimate hidden states and model parameters, find best path, future prediction.
- Parameter estimation via EM and forward-backward algorithm.
- Next lecture: dynamic programming and Viterbi algorithm to find the best path.