20.1 HMM notation recap

- State space $S = \{s_1, \ldots, s_N\}$.
- Set of possible observations $V = \{v_1, \ldots, v_M\}$.
- State transition probabilities $A = [a_{ij}]$, where $a_{ij} = P(q_{t+1} = s_j \mid q_t = s_i)$.
- Observation probabilities $B = [b_j(k)]$, where $b_j(k) = P(v_k \text{ at } t \mid q_t = s_j)$.
- Initial state distribution $\pi = (\pi_1, \ldots, \pi_N)$, where $\pi_i = P(q_1 = s_i)$.

20.2 HMMs as generative models

Once we have specified a hidden Markov model, we can use it to generate a sequence of observations $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_T)$, where $\sigma_t$ is the observation at time $t$. We do so via the following procedure.

1. Chose an initial state $q_1$ according to the probability distribution given by $\pi = (\pi_1, \ldots, \pi_N)$.
2. Set $t = 1$.
3. Choose the observation symbol $\sigma_t$ according to the distribution of observations given the current state. In other words, if the current state is $q_t = s_i$, choose $\sigma_t$ according to the distribution $b_i = [b_i(1), \ldots, b_i(M)]$.
4. Transition to a new state $q_{t+1}$ according to the transition probability matrix $A$. If the current state is $q_t = s_i$, choose the next state according to the distribution $a_i = [a_{i1}, \ldots, a_{iN}]$.
5. Increment $t$ by one. If $t < T$, return to step 3; stop otherwise.

20.3 The evaluation/model scoring problem

**Given:** A hidden Markov model $\lambda = (A, B, \pi)$ and an observation sequence $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_T)$.

**Compute:** The probability of observing the sequence $\sigma$ given the model, $p(\sigma \mid \lambda)$.

We need to calculate $p(\sigma \mid \lambda)$. Consider a sequence of hidden states $Q = (q_1, q_2, \ldots, q_T)$. Since each $q_t$
can be any one of the $N$ states in $S$ and there are $T$ such $q_t$’s, the number of possible hidden state sequences is $N^T$. So it will not be feasible to compute a probability for every possible sequence of hidden states.

Suppose we did have a sequence of hidden states $Q$. The probability of the observed sequence $\sigma$ given this state sequence $Q$ is given by the entries in the matrix $B$:

$$p(\sigma|Q) = \prod_{t=1}^{T} p(\sigma_t|q_t) = b_{q_1}(\sigma_1) \cdot b_{q_2}(\sigma_2) \cdot \ldots \cdot b_{q_T}(\sigma_T).$$

The probability of the sequence of states $Q$ is given by the transition probabilities in the matrix $A$ and the initial state distribution $\pi$:

$$p(Q) = \pi_{q_1} \cdot a_{q_1q_2} \cdot a_{q_2q_3} \cdot \ldots \cdot a_{q_{T-1}q_T}.$$

Combining these results, we obtain the joint probability of the observation sequence and the state sequence

$$p(\sigma, Q) = p(\sigma|Q)p(Q).$$

From this joint distribution, we can compute the marginal distribution of $\sigma$ by summing over all possible state sequences $Q$:

$$p(\sigma) = \sum_{Q} p(\sigma|Q)p(Q) = \sum_{Q} \pi_{q_1} \cdot b_{q_1}(\sigma_1) \cdot b_{q_2}(\sigma_2) \cdot \ldots \cdot b_{q_T}(\sigma_T) \cdot a_{q_1q_2} \cdot a_{q_2q_3} \cdot \ldots \cdot a_{q_{T-1}q_T}$$

$$= \sum_{Q} \pi_{q_1} \left( \prod_{t=1}^{T} b_{q_t}(\sigma_t) \right) \left( \prod_{t=1}^{T-1} a_{q_tq_{t+1}} \right).$$

If we were to create an algorithm to compute this marginal distribution, it would run in $O(N^T)$ time, as discussed above. We need a better approach. The forward algorithm allows us to efficiently compute $p(\sigma)$.

### 20.4 The forward algorithm

Define the forward variable

$$\alpha_t(i) = P[\sigma = (\sigma_1, \ldots, \sigma_t), q_t = s_i].$$

In words, this is the probability of seeing the observed partial sequence $(\sigma_1, \ldots, \sigma_t)$ and ending up in state $s_i$ at time $t$.

#### 20.4.1 The algorithm

The algorithm finds $\alpha_t(i)$ and $p(\sigma|\lambda)$.

1. **Initialization**

   $$\alpha_1(i) = \pi_i b_i(\sigma_1)$$

   for $1 \leq i \leq N$. 
2. **Recurrence**

\[ \alpha_{t+1}(j) = \left[ \sum_{i=1}^{N} \alpha_t(i)a_{ij} \right] b_j(\sigma_{t+1}) \]

for \( 1 \leq i, j \leq N \) and \( 1 \leq t \leq T - 1 \).

3. **Termination**

\[ p(\sigma|\lambda) = \sum_{i=1}^{N} \alpha_T(i). \]