School of Computing and Information Systems
The University of Melbourne
COMP30027 MACHINE LEARNING (Semester 1, 2018)

Tutorial sample solutions: Week 12

1. Hidden Markov Models (HMMs) are best used when the observables are a **univariate time series**: we are just observing a single variable, which changes over time due to some factor that can be estimated from previous observations.

   (a) Recall the two main assumptions (Markov, output independence) that are built into an HMM.

   - **Markov assumption**: the likelihood of transitioning into a given state depends only on the current state, and not the previous state(s) (or output(s)), i.e. $P(q_i|q_1 \cdots q_{i-1}) \approx P(q_i|q_{i-1})$
   - **Output independence assumption**: the likelihood of a state producing a certain observation (as output) does not depend on the preceding (or following) state(s) (or output(s)), i.e $P(o_i|q_1 \cdots q_i, o_1 \cdots o_{i-1}) \approx P(o_i|q_i)$

   (b) Could we construct the HMM in such a way to relax these assumptions? What would the model look like, and what is the major downside?

   - Well, we could have pairs of states in the conditions for our transition probability matrix A, and pairs of states in the conditions for our output proabability matrix B, but this will vastly increase the number of parameters in the model.

   (c) Could we build an HMM for a **multivariate time series**, where we have a number of observed variables for a given (hidden) state?

   - We could represent the outputs as a tuple, again at the cost of vastly increasing the number of parameters. However, sometimes coupling the outputs like this is unnecessary; it might be possible to just generate independent HMMs for each output.

2. **Natural language processing** is one common application for HMMs: we have a single observation (a "word") that varies over time (a "sentence" or "document"), where each observation is associated with some property (like "part of speech").
   Consider the following HMM: $\Pi[\mathrm{J}, \mathrm{N}, \mathrm{V}] = [0.3, 0.4, 0.3]$

A	J	N	V	B	`brown`	`leaves`	`turn`
J	0.4	0.5	0.1	J	0.8	0.1	0.1
N	0.1	0.4	0.5	N	0.3	0.4	0.3
V	0.4	0.5	0.1	V	0.1	0.3	0.6

   (a) How might we go about obtaining the values in the matrices Π, A, and B given above, in a **supervised** context?

   - Each element a_{ij} of A is the count of how many times the state sequence i, j was observed in the labelled data, out of the number of times the state i was observed. In this case, this is a pair of part–of–speech tags.
   - Each element b_{ik} of A is the count of how many times the observation k was observed labelled with state i in the training data, out of the number of times the state i was observed. In this case, this is a word being labelled as the equivalent part–of–speech.
   - Each element π_i is the count of how many times state i was the start of an observation sequence, out of the number of observation sequences. In this case, this is where some part–of–speech starts a sentence.

   (b) Use the **forward** algorithm to find the probability of the "sentence" `brown leaves turn`.

   - See overleaf.
   - The overall probability can be obtained by summing the values in the final column: $0.002256 + 0.014499 + 0.024246 = 0.041001$

α	1:*brown*		2:*leaves*	3:*turn*
J:	J	$\pi[\text{J}]B[\text{J}, \text{brown}]$		
		$0.3 \times 0.8 = 0.24$		
N:	N	$\pi[\text{N}]B[\text{N}, \text{brown}]$		
		$0.4 \times 0.3 = 0.12$		
V:	V	$\pi[\text{V}]B[\text{V}, \text{brown}]$		
		$0.3 \times 0.1 = 0.03$		

α	1:*brown*	2:*leaves*		3:*turn*
J:	0.24	J $\rightarrow$ J	A[J,J]	
		0.24	$\times 0.4 = 0.096$	
		N $\rightarrow$ J	A[N,J]	
		0.12	$\times 0.1 = 0.012$	
		V $\rightarrow$ J	A[V,J]	
		0.03	$\times 0.4 = 0.012$	
		$\sum_i(\alpha_1(i)a_{iJ})b_{Jl}$	$(0.096 + 0.012 + 0.012) \times 0.1 = 0.012$	
N:	0.12	J $\rightarrow$ N	A[J,N]	
		0.24	$\times 0.5 = 0.12$	
		N $\rightarrow$ N	A[N,N]	
		0.12	$\times 0.4 = 0.048$	
		V $\rightarrow$ N	A[V,N]	
		0.03	$\times 0.5 = 0.015$	
		$\sum_i(\alpha_1(i)a_{iN})b_{Nl}$	$(0.12 + 0.048 + 0.015) \times 0.4 = 0.0732$	
V:	0.03	J $\rightarrow$ V	A[J,V]	
		0.24	$\times 0.1 = 0.024$	
		N $\rightarrow$ V	A[N,V]	
		0.12	$\times 0.5 = 0.06$	
		V $\rightarrow$ V	A[V,V]	
		0.03	$\times 0.1 = 0.003$	
		$\sum_i(\alpha_1(i)a_{iV})b_{Vl}$	$(0.024 + 0.06 + 0.003) \times 0.3 = 0.0261$	

α	1:*brown*	2:*leaves*	3:*turn*	
J:	0.24	0.012	J $\rightarrow$ J	A[J,J]
			0.012	$\times 0.4 = 0.0048$
			N $\rightarrow$ J	A[N,J]
			0.0732	$\times 0.1 = 0.00732$
			V $\rightarrow$ J	A[V,J]
			0.0261	$\times 0.4 = 0.01044$
			$\sum_i(\alpha_2(i)a_{iJ})b_{Jt}$	$(0.0048 + 0.00732 + 0.01044) \times 0.1 = 0.002256$
N:	0.12	0.0732	J $\rightarrow$ N	A[J,N]
			0.012	$\times 0.5 = 0.006$
			N $\rightarrow$ N	A[N,N]
			0.0732	$\times 0.4 = 0.02928$
			V $\rightarrow$ N	A[V,N]
			0.0261	$\times 0.5 = 0.01305$
			$\sum_i(\alpha_2(i)a_{iN})b_{Nt}$	$(0.006 + 0.02928 + 0.01305) \times 0.3 = 0.014499$
V:	0.03	0.0261	J $\rightarrow$ V	A[J,V]
			0.012	$\times 0.1 = 0.0012$
			N $\rightarrow$ V	A[N,V]
			0.0732	$\times 0.5 = 0.0366$
			V $\rightarrow$ V	A[V,V]
			0.0261	$\times 0.1 = 0.00261$
			$\sum_i(\alpha_2(i)a_{iV})b_{Vt}$	$(0.0012 + 0.0366 + 0.00261) \times 0.6 = 0.024246$

(c) Use the **Viterbi** algorithm to find the most likely state sequence for the sentence `brown leaves turn`.

- See below, and overleaf.
- The most likely tag sequence can be read right-to-left, based upon the maximum probability we've observed: in this case, 0.0144 when *turn* is a V; this value is derived from the N $\rightarrow$ V transition, so we can infer that *leaves* is an N; that in turn comes from the J $\rightarrow$ N transition, so *brown* is a J.

α		1:*brown*	2:*leaves*	3:*turn*
J:	J	$\pi[\text{J}]B[\text{J}, \text{brown}]$		
		$0.3 \times 0.8 = 0.24$		
N:	N	$\pi[\text{N}]B[\text{N}, \text{brown}]$		
		$0.4 \times 0.3 = 0.12$		
V:	V	$\pi[\text{V}]B[\text{V}, \text{brown}]$		
		$0.3 \times 0.1 = 0.03$		

α	1:*brown*	2:*leaves*		3:*turn*
J:	0.24	J $\rightarrow$ J	$A[\text{J,J}]B[\text{J}, \text{leaves}]$	
		0.24	$\times 0.4 \times 0.1 = \mathbf{0.0096}$	
		N $\rightarrow$ J	$A[\text{N,J}]B[\text{J}, \text{leaves}]$	
		0.12	$\times 0.1 \times 0.1 = 0.0012$	
		V $\rightarrow$ J	$A[\text{V,J}]B[\text{J}, \text{leaves}]$	
		0.03	$\times 0.4 \times 0.1 = 0.0012$	
N:	0.12	J $\rightarrow$ N	$A[\text{J,N}]B[\text{N}, \text{leaves}]$	
		0.24	$\times 0.5 \times 0.4 = \mathbf{0.048}$	
		N $\rightarrow$ N	$A[\text{N,N}]B[\text{N}, \text{leaves}]$	
		0.12	$\times 0.4 \times 0.4 = 0.0192$	
		V $\rightarrow$ N	$A[\text{V,N}]B[\text{N}, \text{leaves}]$	
		0.03	$\times 0.5 \times 0.4 = 0.006$	
V:	0.03	J $\rightarrow$ V	$A[\text{J,V}]B[\text{V}, \text{leaves}]$	
		0.24	$\times 0.1 \times 0.3 = 0.0072$	
		N $\rightarrow$ V	$A[\text{N,V}]B[\text{V}, \text{leaves}]$	
		0.12	$\times 0.5 \times 0.3 = \mathbf{0.018}$	
		V $\rightarrow$ V	$A[\text{V,V}]B[\text{V}, \text{leaves}]$	
		0.03	$\times 0.1 \times 0.3 = 0.0009$	

α	*1:brown*	*2:leaves*	*3:turn*	
J:	0.24	0.0096	J $\rightarrow$ J	A[J,J]B[J, turn]
		J $\rightarrow$ J	0.0096	$\times 0.4 \times 0.1 = 0.000384$
			N $\rightarrow$ J	A[N,J]B[J, turn]
			0.048	$\times 0.1 \times 0.1 = 0.00048$
			V $\rightarrow$ J	A[V,J]B[J, turn]
			0.018	$\times 0.4 \times 0.1 =$ **0.00072**
N:	0.12	0.048	J $\rightarrow$ N	A[J,N]B[N, turn]
		J $\rightarrow$ N	0.0096	$\times 0.5 \times 0.3 = 0.00144$
			N $\rightarrow$ N	A[N,N]B[N, turn]
			0.048	$\times 0.4 \times 0.3 =$ **0.00576**
			V $\rightarrow$ N	A[V,N]B[N, turn]
			0.018	$\times 0.5 \times 0.3 = 0.0027$
V:	0.03	0.018	J $\rightarrow$ V	A[J,V]B[V, turn]
		N $\rightarrow$ V	0.0096	$\times 0.1 \times 0.6 = 0.000576$
			N $\rightarrow$ V	A[N,V]B[V, turn]
			0.048	$\times 0.5 \times 0.6 =$ **0.0144**
			V $\rightarrow$ V	A[V,V]B[V, turn]
			0.018	$\times 0.1 \times 0.6 = 0.00108$