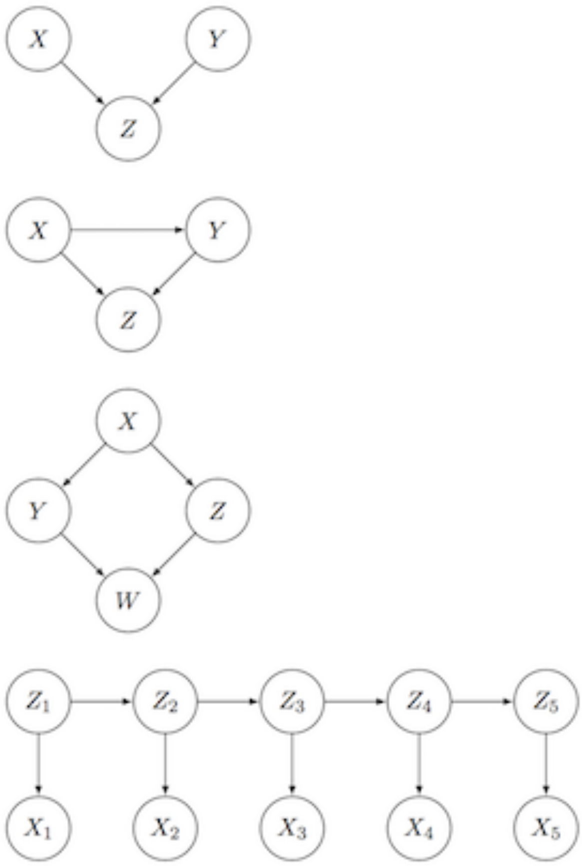
Solutions - ML E2020 - Week 10 - Theoretical Exercises

Graphical Models

As explained in the note *Coditional probabilites and graphical*, a graphical model is a graphical notation to describe the dependency relationships when specifying a joint probility.

From graph to joint probability

Exercise 1: For the following four graphs, write down the joint probabilty of the random variables.



Solution:

$p(X)p(Y)p(Z \mid X, Y)$

$p(X)p(Y \mid X)p(Z \mid X, Y)$

$p(X)p(Y \mid X)p(Z \mid X)p(W \mid Y, Z)$

$p(Z_1) \prod_{i=1}^{5} p(X_i \mid Z_i) \prod_{i=2}^{5} p(Z_i \mid Z_{i-1})$

From joint probability to graph

Exercise 2: Draw the following four joint probabilities as dependency graphs:

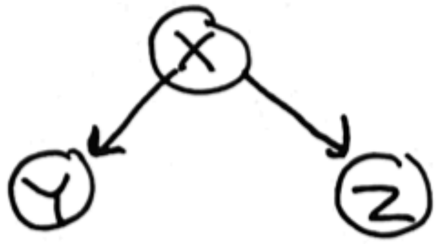
$p(X)p(Y)p(Z)$

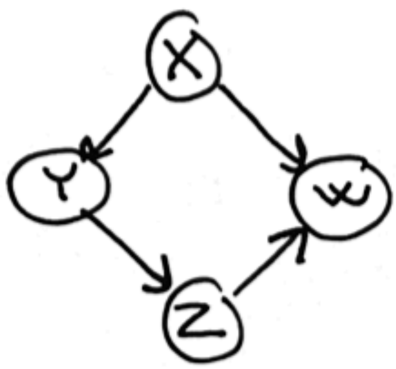
$p(X)p(Y \mid X)p(Z \mid X)$

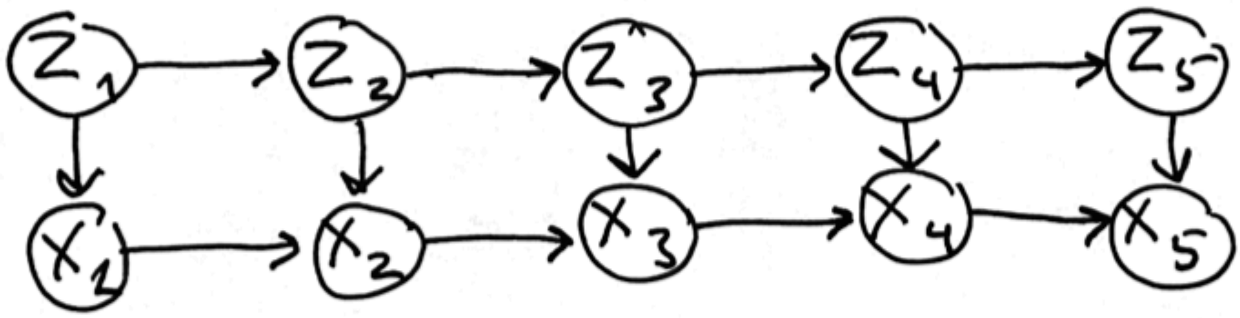
$p(X)p(Y \mid X)p(Z \mid Y)p(W \mid X, Z)$

$p(Z_1)p(X_1 \mid Z_1) \prod_{i=2}^{5} p(X_i \mid Z_i, X_{i-1}) \prod_{i=2}^{5} p(Z_i \mid Z_{i-1})$

Solutions:







Hidden Markov Models

Exercise 3: Questions to slides *Hidden Markov Models - Terminology, Representation and Basic Problems*:

1. How much time does it take to compute the joint probability $P(\mathbf{X}, \mathbf{Z}|\Theta)$ in terms of N and K, where $\mathbf{X} = \mathbf{x}_1, \ldots, \mathbf{x}_N$, $\mathbf{Z} = \mathbf{z}_1, \ldots, \mathbf{z}_N$, and K is the number of hidden states in the hidden Markov model Θ?

   Solution:

   The computation consists of $O(N)$ multiplations of factors that we can look up in constant time, i.e. the running time would be $O(N)$.

1. How many terms are there in the sum on slide 34 for computing $P(\mathbf{X}|\Theta)$? Why?

   Solution:

   We sum over all possible sequences of hidden states $\mathbf{Z} = \mathbf{z}_1, \ldots, \mathbf{z}_N$, where each z_i can have K values, so there are K^N terms in the sum.

1. How many terms are there in the maximization on slide 38 for computing the Viterbi decoding $\mathbf{Z}^*$? Why?

   Solution:

   We maximize over all possible sequences of hidden states $\mathbf{Z} = \mathbf{z}_1, \ldots, \mathbf{z}_N$, where each z_i can have K values, so there are K^N terms in maximization.

1. How many terms are there in the maximixation on slide 39 for computing $\mathbf{z}_n^*$, i.e. the *n*th state in a posterior decoding? Why?

   Solution:

   We maximize over the possible values of $\mathbf{z}_n$, so we maximize over K.

Exercise 4: Questions to slides *Hidden Markov Models - Algorithms for decoding*:

1. Where in the derivation of $\omega(\mathbf{z}_n)$ on slide 7 do we use that the fact that we are working with hidden Markov models? And how do we use it?

   Solution:

   We use it to rewrite the joint probability $p(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{z}_1, \ldots, \mathbf{z}_n)$ as $p(\mathbf{z}_1) \prod_{i=2}^{n} p(\mathbf{z}_i|\mathbf{z}_{i-1}) \prod_{i=1}^{n} p(\mathbf{x}_i|\mathbf{z}_i)$.

1. Where in the derivation of $p(\mathbf{z}_n|\mathbf{x}_1, \ldots, \mathbf{x}_N)$ on slide 16 do we use the fact that we are working with hidden Markov models? And how do we use it?

   Solution:

We use it to rewrite/simplify the probability $p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{z}_n, \mathbf{x}_1, \dots, \mathbf{x}_n)$ to the probability $p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{z}_n)$, i.e. to remove $\mathbf{x}_1, \dots, \mathbf{x}_n)$ from what we condition on. We can do this because $\mathbf{x}_1, \dots, \mathbf{x}_n$ and $\mathbf{x}_{n+1}, \dots, \mathbf{x}_N$ become independent when X and Z depend on the each other as they do in an HMM and we condition on z_n

1. Where in the derivation of $\alpha(\mathbf{z}_n)$ and $\beta(\mathbf{z}_n)$ on slide 20 and 26 do we use that the fact that we are working with hidden Markov models? And how do we use it?

   Solution:

   On slide 20, we use it to rewrite the joint probability $p(\mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{z}_1, \dots, \mathbf{z}_n)$ as $p(\mathbf{z}_1) \prod_{i=2}^{n} p(\mathbf{z}_i | \mathbf{z}_{i-1}) \prod_{i=1}^{n} p(\mathbf{x}_i | \mathbf{z}_i)$.

   On slide 26, we use it to rewrite the joint probability $\sum_{\mathbf{z}_{n+1}, \dots, \mathbf{z}_N} p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N, \mathbf{z}_n, \mathbf{z}_{n+1}, \dots, \mathbf{z}_N)$ as $\sum_{\mathbf{z}_{n+1}, \dots, \mathbf{z}_N} p(\mathbf{z}_n) \prod_{i=n+1}^{N} p(\mathbf{z}_i | \mathbf{z}_{i-1}) \prod_{i=n+1}^{N} p(\mathbf{x}_i | \mathbf{z}_i)$.

1. Why is $P(\mathbf{X}) = \sum_{\mathbf{z}_n} \alpha(\mathbf{z}_n)\beta(\mathbf{z}_n) = \sum_{\mathbf{z}_N} \alpha(\mathbf{z}_N)$ as stated on slide 31?

   Solution:

   $\alpha(\mathbf{z}_n)\beta(\mathbf{z}_n) = p(\mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{z}_n)p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{z}_n) = p(\mathbf{x}_1, \dots, \mathbf{x}_n, , \mathbf{x}_{n+1}, \dots, \mathbf{x}_N, \mathbf{z}_n) = p(\mathbf{X}, \mathbf{z}_n)$. Summing this probability over all K possible values of $\mathbf{z}_n$ yields $p(\mathbf{X})$. Similarly, $\alpha(\mathbf{z}_N) = p(\mathbf{x}_1, \dots, \mathbf{x}_N, \mathbf{z}_N) = p(\mathbf{X}, \mathbf{z}_N)$, and summing over all K possible values of $\mathbf{z}_N$ yields $p(\mathbf{X})$.

1. Algorithmic question: Slide 35 shows how to compute $P(\mathbf{X})$ from $\alpha(\mathbf{z}_N)$ in time $O(K^2 N)$, i.e. the time it takes to compute the last (rightmost) colummn in the α-table. How much space do you need to compute this column? Do you need to store the entire α-table?

   Solution:

   In the forward algorithm, we compute column n in the α-table from column $n-1$. If we in the end only need access to column N, then we only need to keep two columns in memory when we compute the α-table column by column from left to right, namely the current column n, and the previous column $n-1$.

In [ ]: