Bioinformatics 2 - Lecture 4

Guido Sanguinetti

School of Informatics
University of Edinburgh

February 14, 2011
Many data types are ordered, i.e. you can naturally say what is before and what is after

Chief example, data with a time series structure

Other key biological example, sequences (order given by polarity of the molecules)

Any other examples right in front of your eyes?
Latent variables in sequential data

- Sometimes what we observe is not what we are interested in.
- For example, in a medical application, one could think of a person being either healthy (H), diseased (D) or recovering (R).
- What we measure are (related) quantities such as the temperature, blood pressure, $O_2$ concentration in blood, ...
- The job of the doctor is to infer the latent state from the measurements.
Latent variables in sequential data

- In a transcriptomic experiment, we can measure mRNA abundance at different time points after a stimulus.
- What we may be really interested in is the concentration of active transcription factor proteins, which may give a more direct insight in how the cells respond to the stimulus.
- Again, we are interested in reconstructing a latent variable from observations; this time the latent variables are continuous (concentrations).
We represent the latent states as a sequence of random variables; each of them depends *only* on the previous one.

The observations depend only on the corresponding state.
States and parameters

- We are interested in the posterior distribution of the states $x_{1:T}$ given the observations $y_{1:T}$ (subscript $1:T$ denotes the collection of variables from 1 to $T$).
- Notice that we only have one observation per time point.
- In the independent observations case, this would not be enough.
- We also have parameters which we assume known: these are in the known probabilities:
  
  $$\pi = p(x(1)) \quad T_{x(t-1),x(t)} = p(x(t)|x(t-1)) \quad O_{x,y} = p(y(t)|x(t))$$

- We assume parameters to be time-independent.
The single time marginals

- The joint posterior over the states is, by the rules of probability, proportional to the joint probability of observations and states

\[ p(x_{1:T}|y_{1:T}) \propto p(x_{1:T}, y_{1:T}) \]

- An object of central importance is the *single time marginal* for the latent variable at time \( t \)
- This is obtained by marginalising the latent variables at all other time points; by the proportionality above

\[ p(x(t)|y_{1:T}) \propto p(x(t), y_{1:T}) \]
By using the product rule of probability, we can rewrite the joint probability of states and observations as

$$p(x_1:T, y_1:T) =$$

$$= p(y_{t+1:T} | x_1:T, y_1:t) p(x_1:T, y_1:t)$$

Recall that networks encode conditional independence relations; in particular, areas of the network which are not directly connected are independent of each other given the nodes in between.
By inspection of the network representation of the model (slide 4), we see that

\[
p(y_{t+1:T} \mid x_{1:T}, y_{1:t}) = p(y_{t+1:T} \mid x_{t+1:T})
\]

Also \(x_{t+1:T}\) are conditionally independent of \(y_{1:t}\) given \(x_t\), so that

\[
p(x_{1:T}, y_{1:t}) = p(x_{t+1:T} \mid x_{1:t}, y_{1:t}) p(x_{1:t}, y_{1:t}) = p(x_{t+1:T} \mid x_t) p(x_{1:t}, y_{1:t})
\]
Putting equations (2,3) into (1), we get

\[ p(x_1:T, y_1:T) = p(y_{t+1:T}, x_{t+1:T}|x_t) p(x_1:t, y_1:t) \]

Marginalising \( x_{1:t-1} \) and \( x_{t+1:T} \) we get the following fundamental factorisation of the single time marginal

\[
p(x(t)|y_1:T) \propto \alpha(x(t))\beta(x(t)) = p(x(t)|y_1:t) p(y_{t+1:T}|x(t)) \quad (4)
\]

The single time marginal at time \( t \) is the product of the posterior estimate given all the data up to that point, times the likelihood of future observations given the state at \( t \).
The factorisation in equation (4) is an example of message passing.

\( \alpha(x(t)) \) is a message propagated forwards from the previous observations (forward message or filtered process).

\( \beta(x(t)) \) is a message propagated backwards from future observations (backward message).

Message passing algorithms allow exact inference in tree structured graphical models (why?) and approximate inference in more complicated models.
Filtering: computing the forward message

- **Initialisation:**

\[
\alpha(1) \propto p(y(1), x(1)) = \pi O_{x(1), y(1)}
\]

- **Recursion:**

\[
\alpha(t) \propto p(x(t), y_{1:t}) = \sum_{x(t-1)} p(x(t), x(t-1), y_{1:t}) = \\
= \sum_{x(t-1)} p(y(t)|x(t)) p(x(t)|x(t-1)) p(x(t-1)|y_{1:t-1}) = \\
= \sum_{x(t-1)} O_{x(t), y(t)} T_{x(t-1), x(t)} \alpha(x(t-1))
\]

where I used the conditional independences of the network to go from line 1 to 2

- If \(x(t)\) is a continuous, replace the sum with an integral
Computing the backward message

- **Initialisation:** $\beta(x(T)) = 1$ (why?)
- **Backward recursion:**

$$
\beta(x(t-1)) = p(y_{t:T} | x(t-1)) = \sum_{x(t)} p(y_{t:T}, x(t) | x(t-1)) = \\
= \sum_{x(t)} p(y_{t+1:T} | y(t), x(t), x(t-1)) p(y(t) x(t) | x(t-1)) = \\
= \sum_{x(t)} \beta(x(t)) p(y(t) | x(t)) p(x(t) | x(t-1))
$$

- Once again, if $x$ is continuous replace sum with integral
In some organisms, some of the wiring of the network is known.

Simplest possible model, log-linear model of gene expression

\[ g_i(t) = \sum_j S_{ij} X_{ij} TF_j(t) + \epsilon \]

where \( X \) is a binary matrix encoding the network and \( \epsilon \approx \mathcal{N}(0, \sigma^2) \) is an error term.
The simple model of regulation states that gene expression levels are a weighted linear combination of TF levels.

Usually, we do not know the TF (protein) levels, so we treat this as a latent variable problem.

To incorporate dynamics, we assume the TF levels at time $t$ to depend on the levels at time $t - 1$, and gene expression measurements to be conditionally independent given TF levels.

Both TF and gene expression levels are assumed to be Gaussian; Linear Dynamical System (LDS).
The time evolution of the hidden states is given by a Gaussian random walk

\[ x(t + 1) = Ax(t) + w(t) \rightarrow p(x(t + 1)|x(t)) = \mathcal{N}(x(t), \Sigma_w) \]  

(5)

The term \( w \sim \mathcal{N}(0, \Sigma_w) \) is the system noise term; the matrix \( A \) is sometimes called the gain matrix.

Observations are related to states using another linear Gaussian model

\[ y(t) = Bx(t) + \epsilon(t) \rightarrow p(y(t)|x(t)) = \mathcal{N}(Bx(t), \Sigma_\epsilon) \]

(6)

where \( \epsilon \sim \mathcal{N}(0, \Sigma_\epsilon) \) is the observation noise and \( B \) is the observation matrix.
Inference for LDS

- Since both noises are Gaussian and all equations are linear, all the messages will be Gaussian.
- This simplifies the inference as we do not need to compute normalisation constants.
- For example, the forward message is computed as

\[
\alpha(x(t)) = \mathcal{N}(x(t)|\mu_t, \Sigma_t) = \int dx(t-1) \alpha(x(t-1)) \mathcal{N}(x(t)|Ax(t-1), \Sigma_w) \mathcal{N}(y(t)|Bx(t), \Sigma_\epsilon)
\]

- Exercise: calculate the forward message
Biological motivations

- In many cases, we observe intrinsically discrete variables (e.g. DNA bases)
- Also, we are interested in intrinsically discrete latent states (e.g. is this fragment of DNA a gene or not?)
- These situations often arise when dealing with problems in genomics and functional genomics
- We will give three examples, and show some details on how to deal with one of these
How to find genes

- The outcome of a sequencing experiment is the sequence of a region of the genome
- Which parts of the sequence gets transcribed into mRNA?
- Possible solution: sequence the mRNA (laborious)
- Alternatively, use the *codon effect*: genic DNA is not uniformly distributed since triplets of basis code for specific amino-acids
- Thus, the sequence of a gene will look different from the sequence of a not gene region
CpG islands

- In the genome, a G nucleotide preceded by a C nucleotide is rare (strong tendency to be methylated and mutate into T)
- In some regions related to promoters of genes, methylation is inhibited so many more C followed by G (CpG)
- These functional regions are called CpG islands and they are characterized by a different nucleotide distribution
ChIP-on-chip data

- Technology to measure binding of transcription factors to DNA
- Observe an intensity signal (optical)
- Want to infer whether a certain intensity associated with a certain fragment of DNA implies binding or not
- More in Ian Simpson’s guest lecture
When the latent states can only assume a finite number of discrete values, we have a Hidden Markov Model (HMMs).

HMMs have a long history in speech recognition and signal processing and they have their own terminology.

The conditional probabilities $p(x(t+1)|x(t))$ are called *transition probabilities*. They are collected in a matrix $T_{ij} = p(x(t+1) = i|x(t) = j)$.

The conditional probabilities $p(y(t)|x(t))$ are called *emission probabilities*. If the observed variables are also discrete, we can collect the emission probabilities in another matrix $O_{ij} = p(y(t) = i|x(t) = j)$. 
Inference in HMM

- The forward and backward messages are simply computed as matrix multiplications involving emission and transition matrices.

- The forward message is

\[ \alpha(t) \propto \sum_{x(t-1)} O_{x(t), y(t)} T_{x(t-1), x(t)} \alpha(x(t-1)) \]

- The backward message is

\[ \beta(t-1) = \sum_{x(t)} \beta(x(t)) p(y(t)|x(t)) p(x(t)|x(t-1)) \]
We construct latent variables with eight states representing bases in normal DNA and CpG regions, \((A,C,G,T,\bar{A},\bar{C},\bar{G},\bar{T})\)

The \(8 \times 8\) transition matrix will have very low entry for \(T_{C,G}\) and higher entry for \(T_{\bar{C},\bar{G}}\)

The emission matrix is just \(O_{x,x} = 1 = O_{x,\bar{x}}\) with all other entries zero, indicating that the observation is just the nucleotide without the CpG/normal label

The specific entries in the transition matrix will be determined from annotated databases