Nadia Kulshina
CSE 527 notes, Lecture 15
November 14, 2007

HMM and Gene Finding

HMM summary

Viterbi – best single path (max of products)
Forward – sum of all paths (sum of products)
Backward – similar

Baum-Welch – using forward/backward algorithm
Viterbi training – can be done using EM algorithm, but is based on Viterbi

Pfam – protein family database. For example, the family of globin proteins. Grouping proteins into families helps determine their function, evolutionary path, etc. So it is useful to classify proteins into families.
The alignment of 7 globins, for example, can be done with weight-matrix models, but they don't accommodate gaps very well. Another idea is the profile HMM structure, basically a linear chain. Special states are introduced to accommodate insertions or deletions. One sequence can have several insertions or deletions. In the model shown on the slide, the insertion states loop on themselves. This is somewhat arbitrary and can be architected differently depending on the application. Match states – 20 emission probabilities, insert states – background emission probabilities (use data that is already known, e.g. certain properties of membrane proteins), and delete states – silent states.
There are variations to this model. For example, if some states are skipped, it is a more flexible model, but requires many more parameters. With a chain of "silent" states, there are fewer parameters (think: works faster, can be trained with less data, less danger of overfitting), but we also get less detailed control.

So how do we use these models? For searching using forward or Viterbi algorithm; scoring (see below); alignment (where in the proteins are those helices?).

Likelihood vs. Odds scores

The likelihood is determined as the ratio of the probability of emission by the model to the length of the sequence. The log-odds scoring gives a high spread, e.g. a globin with somewhat different sequence gets a low score, and *vice-versa* a non-related protein with a similar sequence gets a high score.
The Z-score. This method uses a sliding window and therefore there is no length dependence, gives good cut-offs.

There are ~8000 protein families and this classification covers roughly 75% of known proteins. So the rest 25% of proteins are not classified. They are probably too different to be grouped into a certain family, may have 1-2 similar proteins.

Model-building refinements

Pseudocounts (count=0 is common when training with 20 amino acids). Pseudocount "mixtures", e.g. sequence pseudocount vectors for different contexts, gives a significant boost.
More refinements. How to decide when to treat as an insertion or a deletion?

Numerical issues

The product of many probabilities approaches 0, and numerical "underflow" may eventually result, i.e. the value in the computer gets rounded to 0. For the Veterbi algorithm we are just adding log's. The forward/backward algorithm is "log-of-sum-of-product-of-exp-of-logs", therefore, it is slower. Typically, the values are calculated from table/interpolation, but it is still slow, and there's still a danger of underflow. This is helped somewhat by directly building log odds into the model: if emission scores are log ratios of model to background, they tend to be are near 0, so underflow is less likely; transition log probabilities are still negative, however.

Model Structure

So many parameters that you may overfit. If you insert loops, the probability grows exponentially: $P=p^n(1-p)$. You get a geometrical distribution. It is not natural, but seems to work well. Additional states can also be introduced, but it is more time consuming.

HMM in gene prediction

The DNA contains protein genes, codes RNA, has non-coding regions. We will focus on protein coding nuclear DNA. The goal of gene prediction is the automated annotation of new sequence data. It is ~60% reliable.
The central dogma of biology is: DNA->RNA->protein. Three consecutive bases of DNA encode for one amino acid in a protein. There is one start codon and 3 stop codons (they terminate translation). The mRNA also contains the 3' and 5' untranslated regions (UTRs). There is a genetic code table that says which triplet codons code for which amino acids. It is interesting that the third nucleotide is often interchangeable in many amino acids. Maybe it's meant not as much to code for a particular amino acid, but rather for protein binding or to contribute to a certain DNA structure. In most higher organisms genes start with a methionine. The genetic code is nearly universal, except for some organisms and organelles (such as mitochondria and chloroplasts).

Translation

The ribosome starts scanning from the 5' end of the RNA until it gets to the translated region.

How is it decoded? Transfer RNAs (tRNA) have anticodons, they form Watson-Crick base pairs with the template RNA to make sure they carry the right amino acid. Then the ribosome moves 3 steps (nucleotides) at a time. Translation has to be very accurate.
Another process that's going on during translation – the tRNA has to get "recharged", i.e. pick up a new amino acid to replace the one it put on the protein that's being synthesized. There are 20 basic amino acids. Speculations exist that some amino acids came into the genetic code earlier than others.

How to find genes?
#1 Find long ORFs.
There are 3 possible sequences that the ribosome will read. They are called frames. An open reading frame (ORF) is one with no stop codons. The length of an average ORF = 64/3=21 codons (triplets), *assuming* thath DNA is random. However, an average protein is on the order of 1000 bp. A 300 bp ORF is encountered once per 36 kbp per strand. Therefore, to find a gene we should look for long ORFs, they are unlikely to be there by chance.
#2 Codon frequency
In random DNA the ratio is:

Leu:Ala:Trp = 6:4:1

But in a real protein:

Leu:Ala:Trp = 6.9:6.5:1

So, coding DNA is not random. Even more, synonym usage is biased. The third base is relatively "loose" (presumably selectively useless). It may be that the 3^{rd} position is not for coding, but for protein binding, or serves as information for other things (such as histones, enhancers, splicing info).

Recognizing codon bias
Assume that $a_1,a_2,\ldots a_{3n+2}$ is coding, but the frame is unknown.
Calculate 3 frames:
$p_1=f(a_1a_2a_3)f(a_4a_5a_6)\ldots$
$p_2=f(a_2,a_3,a_4)f(a_5,a_6,a_7)\ldots$
$p_3=f(a_3,a_4,a_5)f(a_6,a_7,a_8)\ldots$

$P_i=p_i/(p_1+p_2+p_3)$

If the sequence is random, then the P_i's don't differ.
A more general case: *k*-th order Markov model. *k* typically equals to 5 or 6. E.g. it is likely that 2 Ala are in a row.

Codon usage in Φχ174 – Most genes get a high score in one of the three frames, but is low in the same region in the other two. Two overlapping genes going in different directions – E and D regions.

Promoters

In prokaryotes most of the DNA is coding.

Prediction won't find short genes and 5', 3'-UTRs. This can be improved by modeling promoters and other signals, e.g. a weight matrix for the TATA box.

P. Sharp discovered introns. He hybridized mRNA to genomic DNA, it matched in segments, but there were loops on DNA that the RNA wouldn't match. The mRNA was much shorter than the gene. It is observed in certain bacteria and ubiquitous in eukaryotes. The exons code for parts of the protein, the introns don't code and they are excised from the RNA by spliceosomes. Parts of splicing are mediated by small RNAs, e.g. U4, U5, U1. U1 base pairs with the intron on the 5'-end of it, U5 pairs with a piece of that at a different time. A lot of proteins (~50) are also involved in this process, they share similar features.

Hints to origin?

How can something so complex have arisen? The machinery has to be extremely precise at the base pair level.

T. thermophyla is a eukaryote that lives in fresh water ponds. It has RNA molecules that are self-splicing. A long RNA molecule folds up into a structure (see slide), a G nucleotide that sticks out causes the strand to break, a piece falls out, and the remainder of the RNA is then glued back together. The chemistry of this process is similar to that of what happens in the spliceosome. This RNA has a stable tertiary structure, and requires no auxiliary machinery for splicing. This is a ribosomal RNA.

The chemistry can go backwards - the reverse reaction is not as likely. The intron could possibly be inserted back. Maybe it was inserted in the genome and there was no evolutionary pressure to get rid of it. Maybe there's even a positive pressure to maintain it!

Eukaryotes

In eukaryotes there is more variability. New features include introns, exons, splicing. There is some sequence conservation in the vicinity of the splice sites. Branch point signals.

Characterization of human genes – Nature paper (2001), a good paper to read!

Eukaryotes have big genes. Many genes are over 100 kb long. The biggest gene known is dystrophin (DMD) which is 2.4 Mb. It takes 16 hours to transcribe this gene. It is found in the nerve tissue, what makes sense, because it doesn't divide rapidly.