Bioinformatics - Lecture 2

Louis Wehenkel

Department of Electrical Engineering and Computer Science
University of Liège

Montefiore - Liège - October 9, 2007

Find slides: *http://montefiore.ulg.ac.be/∼lwh/IBIOINFO/*

Chapter 2. All the sequence's men - Gene finding

Gene finding - an introduction

Objective:

- Address the computational challenge of finding genes in a genome
- For simplicity, we focus on prokaryote genomes for the time being

Introduction to genes and proteins

- Proteins are the main building block for many tasks in living organisms. They are themselves build up as a chain of amino-acids (AA) (200-300, typically).
- The chain of amino-acids of a protein is produced by translation of an RNA sequence (via the ribosome; while translation takes place, the protein folds progressively to take its three-dimensional structure).
- The RNA sequence needed to produce a given protein is normally obtained by transcribing a part of the DNA contained in the genome (it is then called mRNA) and the corresponding subsequence of DNA is called a gene coding for that protein.
- A given genome can contain as few as 500 genes or as many as 30,000 genes.
- Central dogma: DNA→RNA→Protein

Genetic code

- Correspondence between tri-mers (codons) of nucleotides and amino-acids
- 20 amino-acids, but 64 codons (see book, or Internet, for explanations)
- Some amino-acids correspond to several codons: A (Alanine) corresponds to GCA, GCG, GCT
- Some codons do not correspond to an amino-acid: TAA, TAG, TGA (these are stop codons, see below).
- One codon is special: ATG, it is the sole codon corresponding to Methionine, and is also called start codon (see below).

Genetic code

- Correspondence between tri-mers (codons) of nucleotides and amino-acids
- 20 amino-acids, but 64 codons (see book, or Internet, for explanations)
- Some amino-acids correspond to several codons: A (Alanine) corresponds to GCA, GCG, GCT
- Some codons do not correspond to an amino-acid: TAA, TAG, TGA (these are stop codons, see below).
- One codon is special: ATG, it is the sole codon corresponding to Methionine, and is also called start codon (see below).

NB. Although it is RNA that is translated into amino-acids, we use the DNA alphabet (T instead of U) to describe the genetic code, because we will directly search DNA sequences for protein coding sequences.

Open reading frames

An open reading frame, is a sequence of DNA nucleotides that could be translated into a protein. We know that:

- Translation goes from 5' to 3' end of a strain (sense, or anti-sense)
- Translation always starts with a methionine codon (ATG)
- Translation always stops, as soon as a stop codon is found (and the AA-sequence ends with the AA corresponding to the last non-stop codon).

Open reading frames

An open reading frame, is a sequence of DNA nucleotides that could be translated into a protein. We know that:

- Translation goes from 5' to 3' end of a strain (sense, or anti-sense)
- Translation always starts with a methionine codon (ATG)
- Translation always stops, as soon as a stop codon is found (and the AA-sequence ends with the AA corresponding to the last non-stop codon).

Def. Given a sequence $\mathbf{s}$ over the alphabet $\mathcal{N} = \{\texttt{A}, \texttt{C}, \texttt{G}, \texttt{T}\}$, we define an open reading frame (ORF) as any subsequence whose length L is a multiple of 3, starting with the codon ATG, ending with any one of the stop codons $\{\texttt{TAA}, \texttt{TAG}, \texttt{TGA}\}$, and with no stop codon in the middle.

(NB: Internal start codons are allowed!).

Algorithm for finding ORFs in a genomic sequence

Algorithm 'Search candidate open reading frames of length $\geq k$':

- Given the sequence $\mathbf{s}(0:n)$ and a positive integer k.
- For $i = 0, 1, 2$ do

Loop. Process $\mathbf{s}(i:n)$ codon by codon, from left-to-right, until finding a start codon. If none is found, exit Loop.

- Let $3j + i$ be the position found and search for the first stop codon in $\mathbf{s}(3(j+1)+i:n)$. If none is found, exit Loop.
- Otherwise, let $3(j+l)+i$ be the position of the first stop codon found. If $l + 1 \geq k$, output position j and length l.
- Set $i = 3(j+1)+i$. Continue Loop.

Algorithm for finding ORFs in a genomic sequence

Algorithm 'Search candidate open reading frames of length $\geq k$':

- Given the sequence $\mathbf{s}(0:n)$ and a positive integer k.
- For $i = 0, 1, 2$ do

Loop. Process $\mathbf{s}(i:n)$ codon by codon, from left-to-right, until finding a start codon. If none is found, exit Loop.

- Let $3j + i$ be the position found and search for the first stop codon in $\mathbf{s}(3(j+1)+i:n)$. If none is found, exit Loop.
- Otherwise, let $3(j+l)+i$ be the position of the first stop codon found. If $l + 1 \geq k$, output position j and length l.
- Set $i = 3(j+1)+i$. Continue Loop.

NB. The same procedure has to be applied to the anti-sense strain, in order to find all putative genomic subsequences coding for AA-strings of length at least $k-1$.

Detecting spourious signals: hypothesis testing

- Suppose that our sequence has no genes, i.e. no 'true' ORFs.
- Our procedure will nevertheless output 'wrong' ORFs.
- How to improve our procedure so that, for real genomic sequences (which have 'true' and 'false' ORFs), it outputs mostly 'true' ORFs, while missing as few of them as possible ?
- Unfortunately, there is no single good answer to this question!
- But hypothesis testing is a generic (and widely accepted) procedure to handle such situations.
- Mastering the use of hypothesis testing is difficult !
- This will be our first, but not last, discussion about this subject!

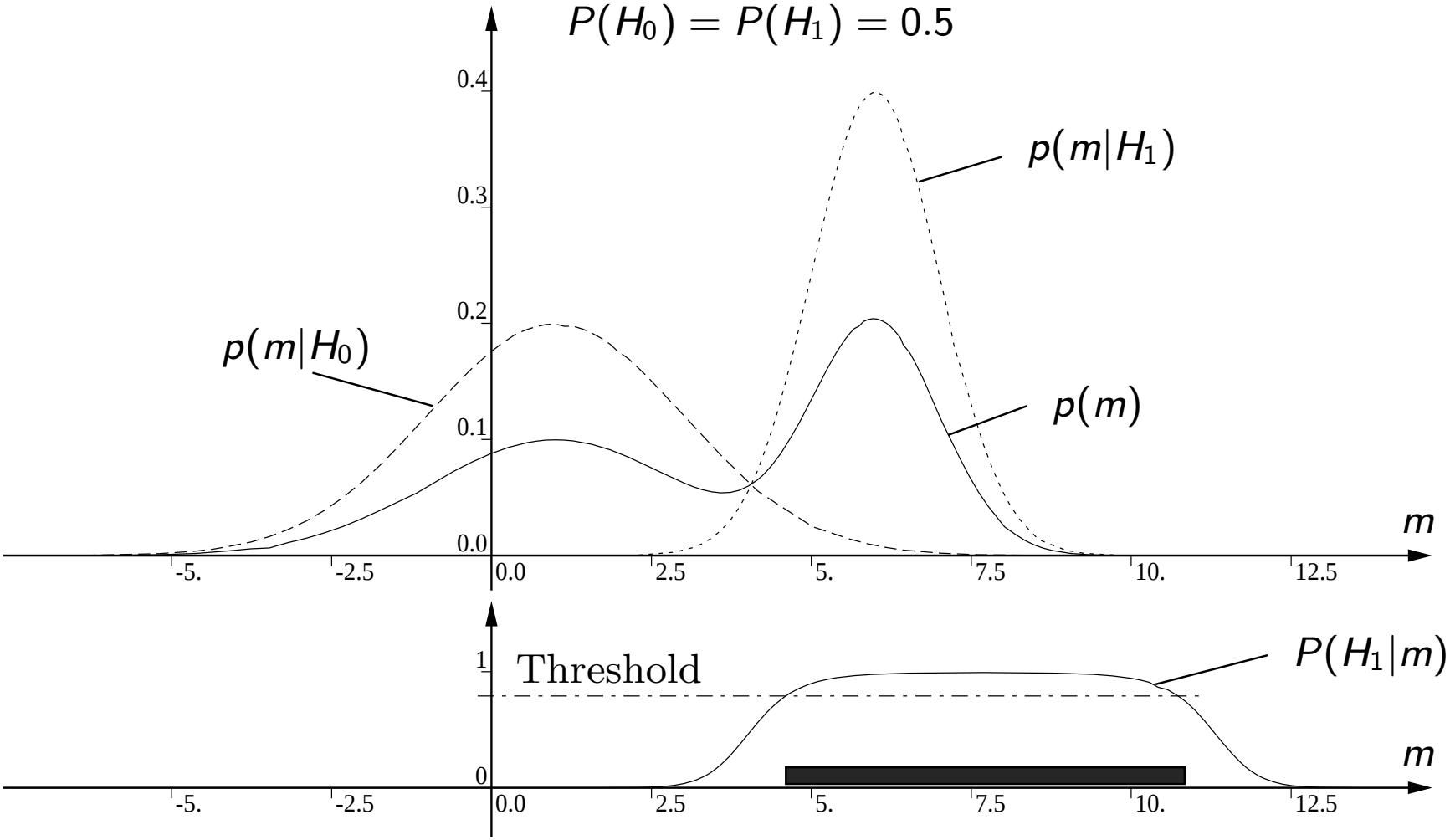
Optimization (under uncertainty)

- Possible situations: $s \in \{s_1, \ldots, s_I\}$
- Possible actions: $a \in \{a_1, \ldots, a_k\}$
- (Net) costs: $c_{ij} = f(s_i, a_j)$, if we take action a_j in situation s_i.
- If we have full information about situation before acting: $a_*(s) = \arg\min_a f(s, a)$
- If we have only 'partial information' in the form of a probability distribution $p(s)$: $a_* = \arg\min_j \sum_{i=1}^{I} p(s_i) c_{ij}$
- What if we have 'side information' in the form of a measurement m, and know the distribution $p(m|s)$ for each possible s, in addition to $p(s)$?

Decision theory

- Let us suppose that there are only two kinds of situations H_0 and H_1 and that we are making a measurement m.
- By using measurement m we want to decide whether we are in situation H_0 or H_1 so as to take an appropriate action.
- Let us suppose that we know that in situation H_i, m has a certain probability distribution $p(m|H_i)$.
- Let us assume that m is obtained by letting 'nature' chose
  - first $i \in \{0, 1\}$ (with probability $p(H_0)$, $p(H_1) = 1 - p(H_0)$)
  - and then m (with probability $p(m|H_i)$).
- We want to design a procedure that on the average (over an infinity of trials) leads to minimum cost! (ie. better than - or at least as good as - any other procedure)

Graphically



Computing the optimal decision threshold

- Cost of action a_0: 0 if H_0 is true, and λ_0 if H_1 is true.
- Cost of action a_1: 0 if H_1 is true, and λ_1 if H_0 is true.
- Expected cost of a_0, given m: $\lambda_0 p(H_1|m)$
- Expected cost of a_1, given m: $\lambda_1 p(H_0|m)$
- Decision rule: choose a_0 if $\lambda_0 p(H_1|m) \leq \lambda_1 p(H_0|m)$.
- Decide a_1 if $P(H_1|m) \geq \frac{\lambda_1}{\lambda_0+\lambda_1}$.
- Type I error: $\alpha = p(a_1|H_0)$
- Type II error: $\beta = p(a_0|H_1)$
- The value(s) of m corresponding to the threshold are called critical values.

Statistical hypothesis testing: example

- We want to decide whether to call the doctor, based on body temperature of our kid.
- Need to define a temperature threshold T_{cr} above which we call the Doctor.
- H_0: kid is not sick (action: don't call the doctor).
- H_1: kid is sick (action: call the doctor).
- Problem:
  - the only thing we know is $p(T|H_0)$ (distribution of normal body temperature measurements) and λ_1
  - I.e.: we ignore, $P(H_0)$ (and $P(H_1)$), $p(T|H_1)$, and λ_0.
  - How to define a rational procedure to decide what to do ?
- Solution: Hypothesis testing
  - Fix Type I error α (e.g. 0.05)
  - Compute T_{cr} suchthat $p(T \geq T_{cr}|H_0) = \alpha$.
  - Reject H_0 if $T \geq T_{cr}$.

Statistical hypothesis testing: ORFs

- We want to define a threshold k on ORF length, so as to reduce the number of 'false' ORFs detected by our procedure.
- Approach
  - Define what we mean by H_0 ('false' ORFs)
  - Compute distribution of lengths of 'false' ORFs.
  - Fix type I risk α, and compute smallest k_{cr} such that $P(k \geq k_{cr}|H_0) \leq \alpha$.
- Example (see book for further examples):
  - H_0: the sequence of non-stop codons following a start codon is generated by a random process choosing all codons with equal probability (1/64).
  - Under this hypothesis, the probability of observing a sequence of non-stop codons of at least length k is equal to $(61/64)^k$.
  - Choose $k = \left\lceil \frac{\log \alpha}{\log 61 - \log 64} \right\rceil$.

Homework 2

Personal Homework for Chapter 2

1. Find all ORFs in human, chimp and mouse mtDNA...
2. Repeat the search on randomized mtDNA sequences...
3. Find ORFs in *H. influenzae*...
4. ($\longrightarrow$ see end of chapter 2 in reference book.)