CSE182-L12

LW statistics/Assembly
Quiz

• Who are these people, and what is the occasion?
Genome Assembly
Questions

• Algorithmic: How do you put the genome back together from the pieces? Will be discussed in the next lecture.

• Statistical? How many pieces do you need to sequence, etc.?
  - The answer to the statistical questions had already been given in the context of mapping, by Lander and Waterman.
Lander Waterman Statistics

\[ G = \text{Genome Length} \]
\[ L = \text{Clone Length} \]
\[ N = \text{Number of Clones} \]
\[ T = \text{Required Overlap} \]
\[ c = \text{Coverage} = \frac{LN}{G} \]
\[ a = \frac{N}{G} \]
\[ q = \frac{T}{L} \]
\[ s = 1 - q \]
LW statistics: questions

- As the coverage $c$ increases, more and more areas of the genome are likely to be covered. Ideally, you want to see 1 island.
- Q1: What is the expected number of islands?
- Ans: $N \exp(-cS)$
- The number increases at first, and gradually decreases.
Analysis: Expected Number Islands

- Computing Expected # islands.
- Let \( X_i = 1 \) if an island ends at position \( i \), \( X_i = 0 \) otherwise.
- Number of islands = \( \sum_i X_i \)
- Expected # islands = \( E(\sum_i X_i) = \sum_i E(X_i) \)
Prob. of an island ending at \( i \)

- \( E(X_i) = \text{Prob (Island ends at pos. } i) \)
- \( = \text{Prob(clone began at position } i-L+1 \text{ AND no clone began in the next } L-T \text{ positions)} \)

\[
E(X_i) = (1 - c)^{L-T} = e^{-cs}
\]

Expected # islands = \( \sum_i E(X_i) = Ge^{-cs} = Ne^{-cs} \)
LW statistics

• Pr[Island contains exactly j clones]?
• Consider an island that has already begun. With probability $e^{-cs}$, it will never be continued. Therefore
• Pr[Island contains exactly j clones] =

\[
(1 - e^{-cs})^j e^{-cs}
\]

• Expected # j-clone islands

\[
= Ne^{-cs} (1 - e^{-cs})^j e^{-cs}
\]
Expected # of clones in an island

• Expected # of clones in an island = $e^{c\theta}$

Q: How? Why do we care?

Often, at the beginning of a genome project, we do not know the length of the genome. This equation helps us determine the length.
Expected length of an island

\[ L = e^c \cdot \frac{1}{c} + (1 - \frac{1}{c}) \]

![Graph showing average length of islands in genome equivalents fingerprinted.](image)
Whole Genome Sequencing & Assembly
Whole Genome Shotgun

- Break up the entire genome into pieces
- Sequence ends, and assemble using a computer
- LW statistics & Repeats argue against the success of such an approach
Problems with Assembly

- Islands might simply be too small in length
- \#Islands = 220K
- Size of an island = 30K
- Not enough to make it an acceptable assembly!
- PLUS, there is the problem of Repeats, Chimerism etc.
Assembling with Repeats

• 40-50% of the human genome is made up of repetitive elements.
• Repeats can cause great problems in the assembly!
• Chimerism causes a clone to be from two different parts of the genome. Can again give a completely wrong assembly
Clones can have mate-pairs

- Recall that we sequence about 1000bp of the end of a clone
- If we sequenced both ends, we get extra information, particularly if we know the length of the original clone.
Mate Pairs

- Mate-pairs allow you to merge islands (contigs) into super-contigs
Super-contigs are quite large

- Make clones of truly predictable length. At Celera 3 sets were used: 2Kb, 10Kb and 50Kb. The variance in these lengths should be small.
- Use the mate-pairs to order and orient the contigs. Note that the gaps are of predictable length.
Whole genome shotgun

- **Input:**
  - Shotgun sequence fragments (reads)
  - Mate pairs

- **Output:**
  - A single sequence created by consensus of overlapping reads

- First generation of assemblers did not include mate-pairs (Phrap, CAP..)

- Second generation: CA, Arachne, Euler

- We will discuss Arachne, a freely available sequence assembler (2nd generation)
Repeats

- Lander Waterman strikes again!
- The expected number of clones in a Repeat containing island is MUCH larger than in a non-repeat containing island (contig).
- Thus, every contig can be marked as Unique, or non-unique. In the first step, throw away the non-unique islands.
Detecting Repeat Contigs 1: Read Density

- Compute the log-odds ratio of two hypotheses:
  - H1: The contig is from a unique region of the genome.
  - The contig is from a region that is repeated at least twice
Arachne: Details

• Initial processing

• Alignment module
  - Input: Collection of DNA sequences of arbitrary length
  - Output: Pairwise alignments between them.
Overlap detection

- **Option 1:** Compute an alignment between every pair.
  - \( G = 3000\text{Mb}, L=500 \)
  - Coverage \( \frac{LN}{G} = 10 \)
  - \( N = \frac{10 \times 3 \times 10^9}{500} = 6 \times 10^7 \)
  - Not good! (Only a small fraction are true overlaps)
K-mer based overlap

• A 25-bp sequence appears at most once in the genome!
• Two overlapping sequences should share a 25-mer
• Two non-overlapping sequences should not!
Sorting k-mers

• Build a list of k-mers that appear in the sequences and their reverse complements
• Create a record with 4 entries:
  - K-mer
  - Sequence number
  - Position in the sequence
  - Reverse complementation flag
• Sort a vector of these according to k-mer
• If number of records exceeds threshold, discard (why?)
Phase 2-4 of Alignment module

- Coalesce k-mer hits into longer, gap-free partial alignments.
- These extended k-mer hits are saved.
- For each pair of sequences, form a directed graph.
- For each maximal path in the graph, construct an alignment.
- Refine alignment via banded DP

![Diagram of partial alignments in the alignment module. Three partial alignments of length k = 6 between a pair of reads coalesce to yield a single full alignment of length k = 19. Vertical bars denote matching bases, whereas x’s denote mismatches. This illustrates the commonly occurring situation where an extended k-mer hit is a full alignment between two reads (k = 6 is used in the figure for simplicity).]
Detecting Chimeric reads

• **Chimeric reads**: Reads that contain sequence from two genomic locations.

• **Good overlaps**: $G(a,b)$ if $a,b$ overlap with a high score.

• **Transitive overlap**: $T(a,c)$ if $G(a,b)$, and $G(b,c)$.

• Find a point $x$ across which only transitive overlaps occur. $X$ is a point of chimerism.

Figure 9  Detection of chimeric reads. Reads $l_1$, $l_2$, $l_3$, $r_1$, $r_2$, and $r_3$, and the absence of a read $r$ (having long overlaps on both sides of a point $x$) suggest that read $c$ may be chimeric, consisting of the juxtaposition of two disparate genomic segments: one corresponding to the part of $c$ before $x$, and one corresponding to the part of $c$ after $x$. We call $x$ the point of chimerism of $c$. Note that reads $l_3$ and $r_3$ extend slightly beyond $x$, as often happens for real chimeric reads.
Repeats

A

B

C

D
Contig assembly

- Reads are merged into contigs up to repeat boundaries.
- \((a,b)\) & \((a,c)\) overlap, \((b,c)\) should overlap as well. Also,
  - \(\text{shift}(a,c) = \text{shift}(a,b) + \text{shift}(b,c)\)
- Most of the contigs are unique pieces of the genome, and end at some Repeat boundary.
- Some contigs might be entirely within repeats. These must be detected.
Detecting Repeat Contigs 1: Read Density

- Compute the log-odds ratio of two hypotheses:
- H1: The contig is from a unique region of the genome.
- The contig is from a region that is repeated at least twice
Creating Super Contigs

Figure 5  Supercontig creation and gap filling. (A) A supercontig is constructed by successively linking pairs of contigs that share at least two forward-reverse links. Here, three contigs are joined into one supercontig. (B) ARACHNE attempts to fill gaps by using paths of contigs. The first gap in the supercontig shown here is filled with one contig, and the second gap is filled by a path consisting of two contigs.
Supercontig assembly

- Supercontigs are built incrementally
- Initially, each contig is a supercontig.
- In each round, a pair of super-contigs is merged until no more can be performed.
- Create a Priority Queue with a score for every pair of ‘mergeable supercontigs’.
  - Score has two terms:
    - A reward for multiple mate-pair links
    - A penalty for distance between the links.
Supercontig merging

- Remove the top scoring pair \((S_1, S_2)\) from the priority queue.
- Merge \((S_1, S_2)\) to form contig \(T\).
- Remove all pairs in \(Q\) containing \(S_1\) or \(S_2\).
- Find all supercontigs \(W\) that share mate-pair links with \(T\) and insert \((T, W)\) into the priority queue.
- Detect Repeated Supercontigs and remove
Repeat Supercontigs

- If the distance between two supercontigs is not correct, they are marked as Repeated.
- If transitivity is not maintained, then there is a Repeat.

**Figure 11** Consistency of forward-reverse links. (A) The distance $d(A, B)$ (length of gap or negated length of overlap) between two linked contigs $A$ and $B$ can be estimated using the forward-reverse linked reads between them. (B) The distance $d(B, C)$ between two contigs $B, C$ that are linked to the same contig $A$, can be estimated from their respective distances to the linked contig.
Figure 12  Filling gaps in supercontigs. (A) Contigs $A$ and $B$ are connected by a path $p$ of contigs $X_1, \ldots, X_k$. The distance $d_p(A,B)$ between $A$ and $B$ (along the path $p$) is the length of the sequence in the path that does not overlap $A$ or $B$. (B) Contigs $Y_1$ and $Y_2$ share forward-reverse links with the supercontig $S$. These links position them in the vicinity of the gap between $A$ and $B$. Therefore, $Y_1$ and $Y_2$ will be used as possible stepping points in the path closing the gap from $A$ to $B$. 
Consensus Derivation

• Consensus sequence is created by converting pairwise read alignments into multiple-read alignments
Summary

• Whole genome shotgun is now routine:
  - Human, Mouse, Rat, Dog, Chimpanzee..
  - Many Prokaryotes (One can be sequenced in a day)
  - Plant genomes: Arabidopsis, Rice
  - Model organisms: Worm, Fly, Yeast

• A lot is not known about genome structure, organization and function.
  - Comparative genomics offers low hanging fruit