Genome Assembly
Using de Bruijn Graphs

Biostatistics 666
Previously:
Reference Based Analyses

- Individual short reads are aligned to reference
- Genotypes generated by examining reads overlapping each position
- Works very well for SNPs and relatively well for other types of variant
Shotgun Sequence Reads

- Typical short read might be <25-100 bp long and not very informative on its own
- Reads must be arranged (aligned) relative to each other to reconstruct longer sequences
Read Alignment

GCTAGCTGATAGCTAGCTAGCTAGCTGATGAGCCCGA

Short Read (30-100 bp)

5’-ACTGGTGCAGCTAGCTGATAGCTAGCTAGCTGATGAGCCCGACTCGCTGCTAGCTCGACG-3’

Reference Genome (3,000,000,000 bp)

- The first step in analysis of human short read data is to align each read to genome, typically using a hash table based indexing procedure.
- This process now takes no more than a few hours per million reads ...
- Analyzing these data without a reference human genome would require much longer reads or result in very fragmented assemblies.
Mapping Quality

• Measures the confidence in an alignment, which depends on:
  – Size and repeat structure of the genome
  – Sequence content and quality of the read
  – Number of alternate alignments with few mismatches

• The mapping quality is usually also measured on a “Phred” scale

• Idea introduced by Li, Ruan and Durbin (2008) Genome Research 18:1851-1858
Shotgun Sequence Data

5′-ACTGGTCGATGCTAGCTGATAGCTAGCTAGCTGATGAGCCCGATCGCTGCTAGCTCAG-3′

Reference Genome

AGCTGATAGCTAG
CTAGCTGATGAGCC
AGCTGATAGCTAG
CTAGCTGATGAGCCCGATCGCTG
GCTAGCTGATAGCTAG
CTAGCTGATGAGCCCGA

Sequence Reads

5′-ACTGGTCGATGCTAGCTGATAGCTAGCTAGCTAGCTGATGAGCCCGATCGCTGCTAGCTCAG-3′

Reference Genome

Reads overlapping a position of interest are used to calculate genotype likelihoods and interpreted using population information.
Limitations of Reference Based Analyses

• For some species, no suitable reference genome available

• The reference genome may be incomplete, particularly near centromeres and telomeres

• Alignment is difficult in highly variable regions

• Alignment and analysis methods need to be customized for each type of variant
Assembly Based Analyses

• Assembly based approaches to study genetic variation
  – Implementation, challenges and examples

• Approaches that naturally extend to multiple variant types
De Bruijn Graphs

• A representation of available sequence data

• Each \( k \)-mer (or short word) is a node in the graph

• Words linked together when they occur consecutively

Short Sequence
AATCGACAGCCGG

De Bruijn Graph Representation
AATC → ATCG → TCGA → CGAC → GACA → ACAG → CAGC → AGCC → GCCG → CCGG

Iqbal (2012)
Effective Read Depth

• Overlaps must exceed $k$-mer length to register in a de Bruijn graph

• This requirement effectively reduces coverage

• Give read read length $L$, word length $k$, and expected depth $D$ ...

$$D_{\text{effective}} = D \frac{L - k + 1}{L}$$
Cleaning

• De Bruijn graphs are typically “cleaned” before analysis

• Cleaning involves removing portions of the graph that have very low coverage

• For example, most paths with depth = 1 and even with depth <= 2 are likely to be errors
Variation in a de Bruijn Graph

• Variation in sequence produces a bubble in a de Bruijn graph

• Do all bubbles represent true variation? What are other alternative explanations?

AATCGACAGCCGG
AATCGATAGCCGG

CGAT → GATA → ATAG → TAGC

AATC → ATCG → TCGA → CGAC → GACA → ACAG → CAGC → AGCC → GCCG → CCGG

Iqbal (2012)
Effective Read Depth - Consequences

- Consider a simple example where $L = 100$

- With $k = 21$ ...
  - Each read includes 80 words
  - Each SNP generates a bubble of length 22
  - A single read may enable SNP discovery

- With $k = 75$ ...
  - Each read includes 26 words
  - Each SNP generates a bubble of length 76
  - Multiple overlapping reads required to discover SNP
Properties of de Bruijn Graphs

• Many useful properties of genome assemblies (including de Bruijn graphs) can be studied using results of Lander and Waterman (1988)

• Described number of assembled contigs and their lengths as a function of genome size, length of fragments, and required overlap
Lander and Waterman (1988) Notation

• The genome size \( G \)

• The number of fragments in assembly \( N \)

• The length of sequenced fragments \( L \)
  – The fractional overlap required for assembly \( \theta \)

• The depth of coverage \( c = NL/G \)

• Probability a clone starts at a position \( \alpha = N/G \)
Number of Contigs

\( Ne^{-c(1-\theta)} \)

• Consider the probability that a fragment starts is not linked to another before ending

\[ \alpha (1 - \alpha)^L (1-\theta) = \alpha (1 - N/G) \frac{Gc}{N} (1-\theta) = \alpha e^{-c(1-\theta)} \]

• Then, the expected number of fragments that are not linked to another is

\[ G\alpha e^{-c(1-\theta)} = Ne^{-c(1-\theta)} \]

• This is also the number of contigs!
Number of Contigs

Number of contigs peaks when depth

\[ c = (1 - \alpha)^{-1} \]
Contig Lengths

• Probability a fragment ends the contig:
  \[ e^{-c(1-\theta)} \]

• Probability of contig with exactly \( j \) fragments:
  \[ (1 - e^{-c(1-\theta)})^{j-1} e^{-c(1-\theta)} \]

• The number of contigs with \( j \) fragments is:
  \[ Ne^{-c(1-\theta)}(1 - e^{-c(1-\theta)})^{j-1} \]

• How many contigs will have 2+ fragments?
Contig Lengths (in bases)

• The expected contig length, in fragments, is
  \[ E(J) = e^{c(1-\theta)} \]

• Each fragment contributes \( X \) bases ...
  \[ P(X = m) = (1 - \alpha)^{m-1}\alpha \text{ for } 0 < m \leq L(1 - \theta) \]
  \[ P(X = L) = (1 - \alpha)^L(1-\theta) \]

• After some algebra:
  \[ E(X) = L\left[\frac{1-e^{-c(1-\theta)}}{c} - \theta e^{-c(1-\theta)}\right] \]

• The expected contig length in bases is \( E(X) E(J) \)
  \[ L\left[\frac{e^{c(1-\theta)} - 1}{c} - \theta\right] \]
Contig Lengths

Lander and Waterman also studied gap lengths.
Enhanced De Bruijn Graphs

• Usefulness of a de Bruijn graph increases if we annotate each node with useful information

• Basic information might include the number of times each word was observed

• More detailed information might include the specific individuals in which the word was present
Variant Analysis Algorithm 1: “Bubble Calling”

- Create a de Bruijn graph of reference genome
  - Bubbles in this graph are paralogous sequences

- Using a different label, assemble sample of interest

- Systematically search for bubbles
  - Nodes where two divergent paths eventually connect

Iqbal (2012)
Word size $k$ and Accessible Genome

Iqbal (2012)
Power of Homozygous Variant Discovery
(100-bp reads, no errors)

Iqbal (2012)
Power of Heterozygous Variant Discovery
(100-bp reads, no errors)

Iqbal (2012)
Power of Homozygous Variant Discovery
(Simulated 30x genomes, 100-bp reads)

Iqbal (2012)
Power of Heterozygous Variant Discovery (Simulated 30x genomes, 100-bp reads)

Dotted lines (…) refer to theoretical expectations. Solid lines (---) refer to simulation results. Iqbal (2012)
Variant Analysis Algorithm 2: Path Divergence

• Bubble calling requires accurately both alleles
  – Power depends on word length $k$, allele length, genome complexity and error model
  – Low power for the largest events

• Path divergence searches for regions where a sample path differs from the reference

• Especially increases power for deletions
  – Deletion often easier to assemble than reference
Path Divergence Example

Black line represents assembly of sample.
We can infer a variant between positions a and b, because the path between them differs from reference.
Variant Analysis Algorithm 3: Multi-Sample Analysis

- Improves upon simple bubble calling by tracking which paths occur on each sample

- Improved ability to distinguish true variation from paralogous sequence and errors

Iqbal (2012)
Classifying Sites

• Evaluate ratio of coverage along the two branches of each bubble and in each individual

• If the ratio is uniform across individuals ...
  – **Error:** Ratio consistently low for one branch
  – **Repeat:** Ratio constant across individuals

• If the ratio varies across individuals ...
  – **Variant:** Ratio clusters around 0, ½ and 1 with probability of these outcomes depending on HWE
Variant Analysis Algorithm 4: Genotyping

- Calculate probability that a certain number of k-mers cover each path
- To improve accuracy, short duplicate regions within a path can be ignored.
- Allows likelihood calculation for use in imputation algorithms

Iqbal (2012)
Example Application to High Coverage Genome

- 26x, 100-bp reads, k = 55

- 2,777,252,792 unique k-mers
  - 2,691,115,653 also in reference
  - 23% more k-mers before cleaning

- 2,686,963 bubbles found by Bubble Caller
  - 5.6% of these also present in reference

- 528,651 divergent paths
  - 39.8% of these also present in reference

- 2,245,279 SNPs, 361,531 short indels, 1,100 large or complex events
  - Reproduces 67% of heterozygotes from mapping (87% of homozygotes)
Comparison to Mapping Based Algorithms
Summary

• Assembly based algorithms currently reach about 80% of the genome.

• These algorithms can handle different variant types more conveniently than mapping based approaches.

• Incorporating population information allows repeats to be distinguished from true variation.
Recommended Reading
