Long “Synthetic Reads” aka Moleculo

Genomic DNA

Fragment

Size Select (10kb)

Polish, ligate amplification adaptors
Dilute to 500 molecules per well

Amplify, fragment, add sequencing adaptors
Pool

Sequence
Separate, based on well barcode

Remove barcodes, assemble 10kb fragments

Assemble genome from 10kb fragments
Synthetic Read Characteristics
10x Genomics

- Similar in concept to CPT-Seq from last week’s paper
- Idea is to uniquely barcode reads that derive from a long molecule - ~50-100kb
- 10x Chromium system automates much of the process for you
Benefits of 10x

• Correct placement in difficult to align regions:

Paralog A

Paralog B
Benefits of 10x

- Correct placement in difficult to align regions:

Paralog A

Paralog B
Benefits of 10x

• Correct placement in difficult to align regions:

Paralog A

Paralog B
Benefits of 10x

- Correct placement in difficult to align regions:

Paralog A

Paralog B
Benefits of 10x

• Correct placement in difficult to align regions:
Benefits of 10x

• Haplotype phasing:
Benefits of 10x

• Haplotype phasing:
Benefits of 10x

- Haplotype phasing:
Benefits of 10x

- Haplotype phasing:
Using Hi-C data to aid assemblies

- Hi-C is a proximity ligation method, aimed at reconstructing the 3 dimensional structure of a genome
- Originally developed with the idea of looking at how the genome of an organism for which a good reference exists is physically organized
- But, probability of intrachromosomal contacts is much higher than that of interchromosomal contacts.
- Although the probability of interaction decays rapidly with linear distance, even loci separated by >200 Mb on the same chromosome are more likely to interact than loci on different chromosomes
Hi-C Protocol
Hi-C Data

intrachromosomal

interchromosomal

probability
Applying Hi-C data to assemblies

LACHESIS (ligating adjacent chromatin enables scaffolding *in situ*)
Results

- Applied to human contigs, using Hi-C data from an embryonic cell line
- Most scaffolds ($n = 13,528$, comprising 98.2\% of the length of the shotgun assembly) were clustered into one of the 23 groups
- Nearly all of these groups corresponded to individual chromosomes, with the exceptions of the X chromosome, whose arms were in separate groups
- Within each chromosome group, the vast majority of the length of the clustered scaffolds was successfully ordered and oriented by LACHESIS (94.4\%)
Principle Can also be applied to Metagenomic Samples
Suggests multipronged approach to de novo Assembly

- Provide high coverage data over various different physical distances
  - shotgun fragment
    * Typical paired end, short read sequencing – can be very high coverage
  - Up to 12-kbp mate-pair
    * Allows you to bridge many repeats
  - CPT-seq/10X Genomics
    * Longer range contiguity (limited to ~10Mb genomes and above)
  - Synthetic reads
    * Generate some complete reads than span repeats, but significant PCR bias (limited to ~100Mb genomes and above)
  - Long reads (PacBio, Oxford Nanopore)
    * Low coverage, and high error rates, (though in a hybrid strategy that may become unimportant, as they can be used for scaffolding)
  - Hi-C
    * Incorporates longer range genome information
Applications of Next-Gen Sequencing

- Genome variability
- Metagenomics
- Genome modifications
- Detection of mutations
- Association studies
  - Phylogeny
  - Evolution

- Nucleosome positioning
- 3D genome architecture
  - Active promoters
  - Interactions between nucleic acids and proteins
  - Chromatin modifications

- Interactions between nucleic acids and proteins
  - Transcript identity
  - Transcript abundance
  - RNA editing
  - SNPs
  - Allele specific expression
  - Regulation

**Genome**
- De novo sequencing
  - Assembly
  - Annotation

**Chromatin**
- Resequencing
  - Mapping
  - Detection of variants
  - 3D reconstruction
  - Identification of open chromatin
  - Detection of binding sites

**Transcriptome**
- ATAC-Seq
  - Mapping
  - Identify open chromatin
- ChIP-Seq
  - Mapping
  - Detection of binding sites
- RNA-Seq
  - Mapping
  - Transcript detection and quantification
Applications of Next-Gen Sequencing

- De novo sequencing
  - Assembly
  - Annotation

- Resequencing
  - Mapping
  - Detection of variants

- Hi-C
  - Mapping
  - 3D reconstruction

- ATAC-Seq
  - Mapping
  - Identify open chromatin

- ChIP-Seq
  - Mapping
  - Detection of binding sites

- RNA-Seq
  - Mapping
  - Transcript detection and quantification

- Genome
  - Genome variability
  - Metagenomics
  - Genome modifications
  - Detection of mutations
  - Association studies
    - Phylogeny
    - Evolution

- Chromatin
  - Nucleosome positioning
  - 3D genome architecture
    - Active promoters
    - Interactions between nucleic acids and proteins
    - Chromatin modifications

- Transcriptome
  - Interactions between nucleic acids and proteins
    - Transcript identity
    - Transcript abundance
      - RNA editing
      - SNPs
      - Allele specific expression
    - Regulation
Mapping Short Reads

- Many options; often a trade off between speed, resources and sensitivity.
- Several open source projects to solve this problem, continually improving in speed and memory requirements.
- New features being added all the time.
- When dealing with short read data, make sure you have the very latest versions of the software you’re using, as some are updated frequently.
Alignment

@HWI-EAS412_4:1:1:1376:380
AATGATACGGCGACCACCGAGATCTA
Approaches to Short Read Alignment

Two main approaches

• Hash-Based mapping:
  – Hashing of reads (E.g. Maq, Eland, SHRiMP)
  – Hashing of genome (E.g. novoalign, SHRiMP2)

• Indexing of tree-like structures:
  – Bowtie (ungapped)
  – Bowtie2 (gapped)
  – Bwa (gapped)
  – these all use Suffix Arrays/Burrows-Wheeler Transform (BWT), coupled with FM index
Hashing

- A hash function simply converts a string ("key") to an integer ("value").
- The integer is then used as an index in an array, for fast look up.
MAQ

- First widely used open source short read aligner
- Very fast, but at the cost of accuracy (ungapped)
- Uses hashing to index sequence reads, then scans with reference sequence
- Guaranteed to find alignments with up to 2 mismatches
- Can take advantage of paired end reads
- **Uses quality scores** to determine best alignments
- Generally no longer used, as has been superceded by newer aligners

http://sourceforge.net/projects/maq/
Tries

• The term trie comes from retrieval
• Sometimes pronounced tree, sometimes try.
• Is a tree representing a collection of strings, with one common node per prefix
• Smallest tree such that:
  – Each edge is labeled with a character $c \in \Sigma$
  – A node has at most one outgoing edge labeled $c$, for $c \in \Sigma$
  – Each key is “spelled out” along some path starting at the root
Trie Example

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>instant</td>
<td>1</td>
</tr>
<tr>
<td>internal</td>
<td>2</td>
</tr>
<tr>
<td>internet</td>
<td>3</td>
</tr>
</tbody>
</table>

- Smallest tree such that:
  - Each edge is labeled with a character $c \in \Sigma$
  - A node has at most one outgoing edge labeled $c$, for $c \in \Sigma$
  - Each key is “spelled out” along some path starting at the root
We can index $T$ with a trie. The trie maps substrings to offsets where they occur
Bowtie

• Also uses quality scores to find best alignments.
• Uses “Burrows-Wheeler index” to keep its memory footprint small.
• Can find alignments with up to 3 mismatches in the first L bases of the read.
• Only ungapped alignments
• Also supports paired end reads.
  
  http://bowtie-bio.sourceforge.net/
• Bowtie2 supports gapped alignments too.
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACGGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380

AATGATACCGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACGGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACGGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACGGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380

AATGATACGGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACGGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACGGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACCGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACGGCGACCA[CCGAGATCTA]
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACGGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACGGCGACCACCGAGATCTA
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380
AATGATACGGCGACCACCGAGATCTA
AATAATACGGCGACCACCGAGATCTA

@HWI-EAS412_4:1:1:1376:380
AATG[RED]ATACGGCGACCACCGAGATCTA
AATAATACGGCGACCACCGAGATCTA

↑
Bowtie Algorithm

@HWI-EAS412_4:1:1:1376:380

AATAATACGGCGACCACCGAGATCTA
BWA

- From the author of Maq, but now also uses Burrows-Wheeler transform to significantly speed it up, and use less memory.
- Can also find small indels, in contrast to both Maq and Bowtie.
- Is slightly slower than bowtie, but ability to find indels make it more useful if SNVs are important to you.
Current Practice

- Most people use bwa for mapping their short read data if they want to discover variants.
- Novoalign is actually a more accurate aligner, but slower, and often impractical for large genomes (though we often use it for yeast).
- If not interested in variants, people use bowtie for speed.
- There is no standard benchmark dataset, though see:
- It doesn’t hurt to experiment.
- Always do some online research to see if there’s something new out there.
SAM files

- Sequence Alignment/Map format
- Is a concise file format that contains information about how sequence reads maps to a reference genome
- Can be further compressed in BAM format, which is a binary format of SAM.
- Can also be sorted and indexed to provide fast random access, using SAMtools (more on this in a minute).
- Requires ~1 byte per input base to store sequences, qualities and meta information.
- Supports paired-end reads
- Is produced by bowtie, bwa
- SAM can be converted to pileup format.
<table>
<thead>
<tr>
<th>No.</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>QNAME</td>
<td>Query NAME of the read or read pair</td>
</tr>
<tr>
<td>2</td>
<td>FLAG</td>
<td>Bitwise FLAG</td>
</tr>
<tr>
<td>3</td>
<td>RNAME</td>
<td>Reference Sequence Name</td>
</tr>
<tr>
<td>4</td>
<td>POS</td>
<td>1-Based leftmost POSitionof clipped alignment</td>
</tr>
<tr>
<td>5</td>
<td>MAPQ</td>
<td>MAPping Quality (Phred-scaled)</td>
</tr>
<tr>
<td>6</td>
<td>CIGAR</td>
<td>Extended CIGAR string (operations: MIDNSHP)</td>
</tr>
<tr>
<td>7</td>
<td>MRNM</td>
<td>Mate Reference NaMe (‘=’ if same as RNAME)</td>
</tr>
<tr>
<td>8</td>
<td>MPOS</td>
<td>1-Based leftmost Mate POSition</td>
</tr>
<tr>
<td>9</td>
<td>ISIZE</td>
<td>Inferred Insert SIZE</td>
</tr>
<tr>
<td>10</td>
<td>SEQ</td>
<td>Query SEQuence on the same strand as the reference</td>
</tr>
<tr>
<td>11</td>
<td>QUAL</td>
<td>Query QUALity (ASCII-33=Phred base quality)</td>
</tr>
</tbody>
</table>
SAM Format

coor  12345678901234  5678901234567890123456789012345
ref   AGCATGTTAGATAA**GATAGCTGTGCTAGTAGGCAGTCAGCGCCAT

r001+   TTAGATAAAAGGATA*CTG
r004+   ATAGCT..............TCAGC
r001-   CAGCGCCCAT

@SQ SN:ref LN:45
r001 163 ref 7 30 8M2I4M1D3M = 37 39 TTAGATAAAAGGATACTG *
r004  0 ref 16 30 6M14N5M    *  0  0 ATAGCTTCAGC      *
r001  83 ref 37 30 9M        *  0  0 CAGCGCCCAT      *
Alignment Visualization

- SAMtools has its own terminal-based text visualization tool, called tview
- Very simple, but fast, and useful for looking at SNPs and indels
Alignment Visualization

- For more complex viewing needs, use either GenomeView or IGV
- Can display BAM files, plus annotation tracks
- Java based dynamic visualization software
- Shows snps, indels, spliced reads and a signal track

http://genomeview.sourceforge.net/
http://www.broadinstitute.org/igv/

- Also check out JBrowse for online viewing of BAM files

NB: Browser-driven analyses are necessarily anecdotal and at best semi-quantitative, but nonetheless can be useful as a sanity check.
Genome Resequencing

- SNP/indel discovery
  - Can now multiplex 96 haploid yeast strains on a single lane of the HiSeq 2500 (~30x coverage each)
  - 1 lane of HiSeq = ~10x human genome
- Structural variant discovery
- Copy Number changes
- Novel sequence discovery
SNP and indel calling

• SNPs and indels are ‘called’ from SAM/BAM files
• Typically use GATK (Genome Analysis Toolkit), which will identify nucleotides with support for variation
  – Sounds trivial, but typically lots of false positives
  – Harder in a diploid than a haploid
  – Harder with lower coverage
  – You *must* read the “Best Practice Variant Detection with the GATK” page before using GATK; it changes frequently
  – Make sure the version of GATK you are using (it is updated frequently) corresponds to the latest “best practices”
  – Always document what you did – this is best done in the form of a Perl or Python script that glues GATK calls into a pipeline, and that documents its parameters.
  – There are now other tools, like FreeBayes, and a deep learning solution from Google
VCF files

- VCF stands for variant call format
- Produced by the GATK, and describes all variant positions derived from bam files
- Meta information lines, preceded by `##`, indicate the source of data used, and how the file was generated.
- Variants are not annotated – to do this, using a tool such as SNPeff or ANNOVAR
  - Helps you zero in on which variants might be of interest
Copy Number Variation

- Sequence coverage of a region can allow detection of amplifications or deletions – i.e. duplicated regions will have higher coverage.
- Power to detect such regions depends on their size, copy number and number of reads.
CNV Detection Power

Coverage Shows Amplification

HXT7  HXT6
Structural Variation

• Using paired-end read data, it’s possible to identify structural variants:

Adapted from Korbel at al, 2007.
Structural Variant Identification

• There are several tools, e.g.:
  – BreakDancer (used by WashU Sequencing Center)
  – VariationHunter
  – MoDIL
  – inGAP-SV
    • has a nice visualization tool (http://ingap.sourceforge.net/)
  – PEMer (developed by the Gerstein lab)
  – Lumpy
  – Delly

• No real robust comparison between these exists on a standard dataset

• Ask around, find out what others are using, get latest versions and experiment
Identifying Novel Sequences

- Typically after aligning your data, you have some number of left over reads, which didn’t map to your reference sequence
- These are either reads with a bunch of errors, or correspond to additional sequence in your sample of interest not present in the reference sequence
- Can often assemble these left over reads into contigs of novel sequence, for example using Velvet, after which you can annotate them.
- Often interesting things in your unmapped reads, so don’t just discard them without thinking about it
Recommended Reading

File Formats and Tools


Mapping and Variant Calling