Deep Sequencing Data Analysis

Ross Whetten
Professor
Forestry & Environmental Resources

Background

- Who am I, and why am I teaching this topic?
  - I am not an expert in bioinformatics
  - I started as a biologist with data that needed analysis
  - You can learn as I have, by teaching yourself
  - The key is to be willing to make an effort to learn

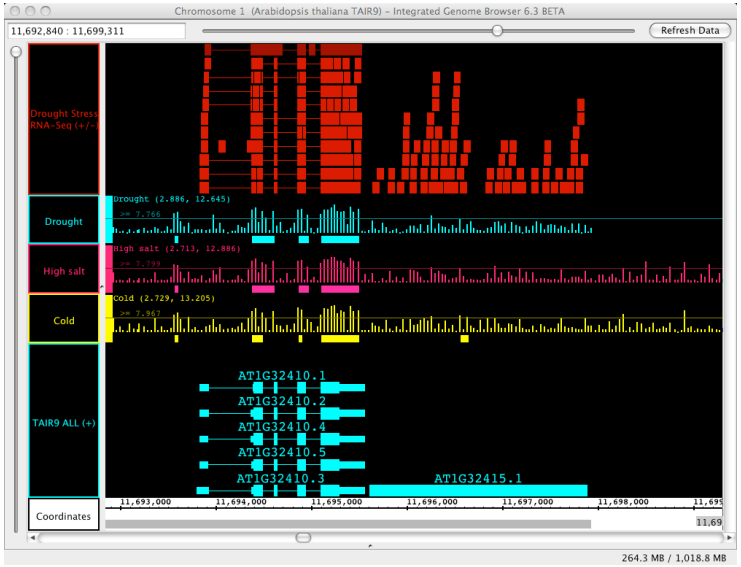
- Who are you, and why are you taking this?
  - How many molecular biologists?
  - How many applied biologists (plant breeders, wildlife biologists, medical microbiologists, etc)?

BIOTECHNOLOGY
NC STATE UNIVERSITY

BIT 815: Analysis of Deep Sequencing Data Fall 2011

Overview:

- This course covers methods for analysis of data from Illumina and Ion Torrent high-throughput sequencing, with or without a reference genome sequence, using free and open-source software tools with an emphasis on the command-line Linux computing environment



Lecture Topics:

- Types of samples and analyses
- Experimental design and analysis
- Data formats and conversion tools
- Alignment, de-novo assembly, and other analyses
- Computing needs and available resources
- Annotation
- Summarizing and visualizing results

Labs:

Lab sessions meet in a computing lab, and will provide students with hands-on experience in managing and analyzing datasets from Illumina and Ion Torrent instruments, covering the same set of topics as the lectures. Example datasets will be available from both platforms, for both DNA and RNA samples.

Objective for lectures and labs: To empower you to teach yourself

- Give an overview of key steps in data analysis process
- Introduce some of the alternative approaches
- Discuss some advantages and disadvantages
- Provide access to the literature on approaches and tools

It is NOT an objective of this class to

- *Provide an optimized workflow for a particular experiment*
- *Troubleshoot individual steps in individual analyses*
- *Provide a complete and final answer regarding the best tool or approach for a particular analysis*

Lecture outline

- Computational resources and requirements
- Workspace environments
- Experimental design
- Data management and manipulation tools, quality assurance
- Mapping sequence reads to a reference genome
- De-novo assembly of reads without a reference genome
- Variant discovery – SNPs, small indels, and copy number variants
- Transcriptome analysis for gene discovery and gene expression measurement
- Annotation resources

Lecture outline

- Computational resources and requirements
- Workspace environments
- Experimental design
- Data management and manipulation tools, quality assurance
- Mapping sequence reads to a reference genome
- De-novo assembly of reads without a reference genome
- Variant discovery – SNPs, small indels, and copy number variants
- Transcriptome analysis for gene discovery and gene expression measurement
- Annotation resources

Computational Resources

- Typical desktop computers often lack enough RAM to analyze sequence datasets
  - 32-bit operating systems cannot address more than 4 Gb
  - 64-bit operating systems can address up to 2000 Gb
  - 64-bit Linux is the platform of choice, because most open-source packages are designed for it
- Alternatives to desktop computers
  - Virtual Computing Lab (http://vcl.ncsu.edu)
  - HPC (http://hpc.ncsu.edu)
  - Amazon Web Services Elastic Compute Cloud
  - iPlant

Computational Resources

- AWS EC2 (http:// aws.amazon.com/ec2/)
  - Online provider of computing resources by the hour
  - Many choices of operating system and machine resources
  - The “r3.2xlarge” instance type has 60 Gb RAM, 160 Gb local SSD space, 8 processor cores, for $0.70/hr
- AWS Elastic Block Storage
  - Stable disk storage in the cloud, automatically backed-up
  - 10 cents/Gb/month
  - Can be attached to any EC2 image in the same zone
  - Public datasets are hosted by AWS and freely available (http://aws.amazon.com/datasets/)

Computational Resources

- A compute cluster is more powerful than a single unit
  - Not all computational problems are suitable for clusters
  - Not all programs are written for cluster computing
  - In cases where the problem is suitable, and the software is available, the advantage can be huge
  - Graphics processor units are a powerful way to parallelize
- More and more software is available for clusters
- Software development for cluster computing is an active area of research
  - Taylor, BMC Bioinformatics 11(Supp12):S1, 2010

Computational Resources

- Examples of cluster-based software
  - Cloudburst – short read mapping to large genomes
  - Crossbow – SNP discovery in short reads mapped to reference
  - Contrail – de-novo assembly
  - Myrna – transcriptome analysis
  - ABySS – de-novo assembly of short reads
  - Ray – de-novo assembly of mixed read-length sequences

Computational Resources

- AWS offers cluster computing resources
  - cc2x.8xlarge: 32 CPU cores, 60 Gb RAM, 3360 Gb local storage
  - $2 per instance per hour
  - g2.2xlarge GPU: 1536 cores; 4 Gb RAM, 60 Gb SSD
  - $.65 per instance per hour
- Galaxy Cloudman package helps manage clusters
  - Afgan et al, *BMC Bioinformatics* **11**(Suppl 12)**:**S4, 2010

"We present a cloud resource management system that makes it possible for individual researchers to compose and control an arbitrarily sized compute cluster on Amazon's EC2 cloud infrastructure without any informatics requirements. …The provided solution makes it possible, using only a web browser, to create a completely configured compute cluster ready to perform analysis in less than five minutes."

Lecture outline

- Computational resources and requirements
- Workspace environments
- Experimental design
- Data management and manipulation tools, quality assurance
- Mapping sequence reads to a reference genome
- De-novo assembly of reads without a reference genome
- Variant discovery – SNPs, small indels, and copy number variants
- Transcriptome analysis for gene discovery and gene expression measurement
- Annotation resources

Workspace environments

- R statistical environment
  - A general-purpose statistical analysis environment
  - Powerful graphics and data-management capabilities
  - A specialized set of tools for genomics: Bioconductor
- Genome Analysis Tool Kit
  - Designed to make programming genomic tools easier
  - A parallel role as workspace for analysis of human data
- Galaxy
  - An environment that connects together various programs
  - Web-server, locally-installed, and cloud versions available

Workspace environments

- None of these are a complete solution
  - Galaxy is perhaps the most complete and most extensible
  - R has the strongest credentials for statistical analysis
- All of them have something useful to offer
  - GATK has an integrated solution for recalibration of base call error probabilities and variant detection
- New versions, and completely new tools, will emerge

Lecture outline

- Computational resources and requirements
- Workspace environments
- Experimental design
- Data management and manipulation tools, quality assurance
- Mapping sequence reads to a reference genome
- De-novo assembly of reads without a reference genome
- Variant discovery – SNPs, small indels, and copy number variants
- Transcriptome analysis for gene discovery and gene expression measurement
- Annotation resources

Experimental design

- Why is this important?
  - It determines the value of the data for analysis
- Doesn't the statistical software take care of that?
  - "No amount of statistical sophistication can separate confounded factors *after* data have been collected" – Auer and Doerge (Genetics 185:405-416, 2010)
- But I don't understand all that statistical jargon
  - Get help from someone who does
  - Each college should have access to a statistics consultant

Experimental design

- What are the sources of variation in the experiment?
  - Among individuals
  - Among treatments
  - Among sequencing runs
  - Among lanes/sectors within run
  - Among library preparations
- Which sources of variation are of interest?
  - Avoid confounding effects of interest with nuisance effects
- Allocate effort to estimate effects of greatest interest
  - Block to replicate measurements across nuisance effects
  - Exploit barcoding/multiplexing tools for blocking
  - Balanced or partially-balanced designs possible, using complete or incomplete blocking

Experimental design

 Genetics 185(2):405-16, 2010

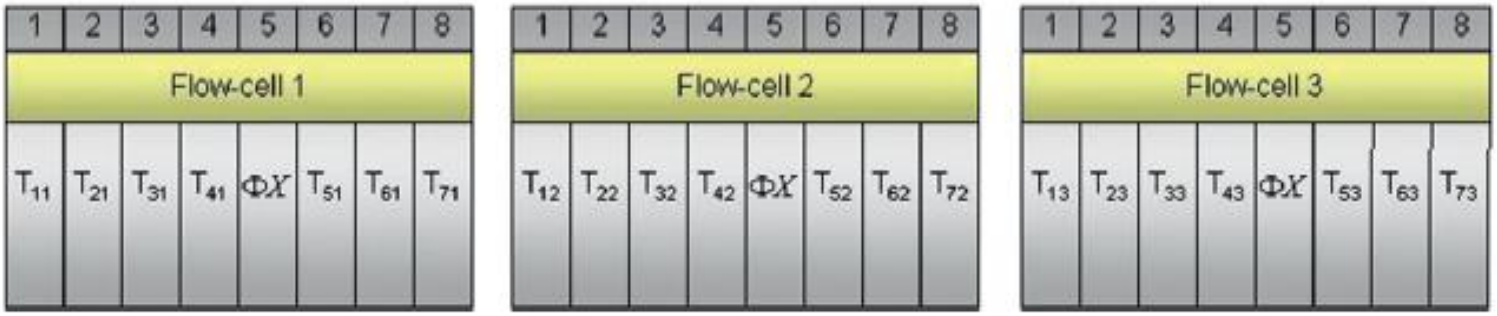


FIGURE 3.—A multiple flow-cell design based on three biological replicates within seven treatment groups. There are three flow cells with eight lanes per flow cell. The control ΦX sample is in lane 5 of each flow cell. T_{ij} refers to the jth replicate in the ith treatment group ($i = 1, \ldots, 7; j = 1, \ldots, 3$).

Here seven treatments are applied to each of three biological replicates , and each sample is sequenced in a separate lane

Are nuisance effects confounded with treatment effects?

Experimental design

 Genetics 185(2):405-16, 2010

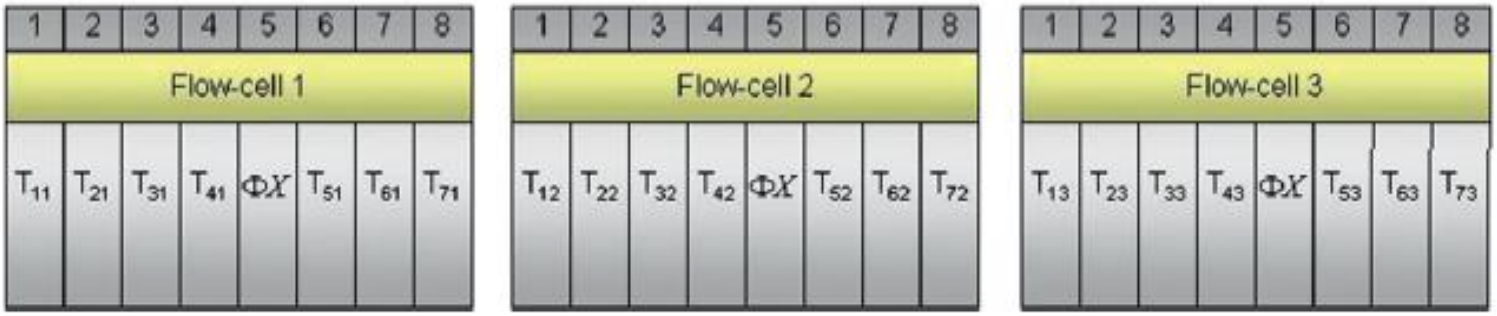


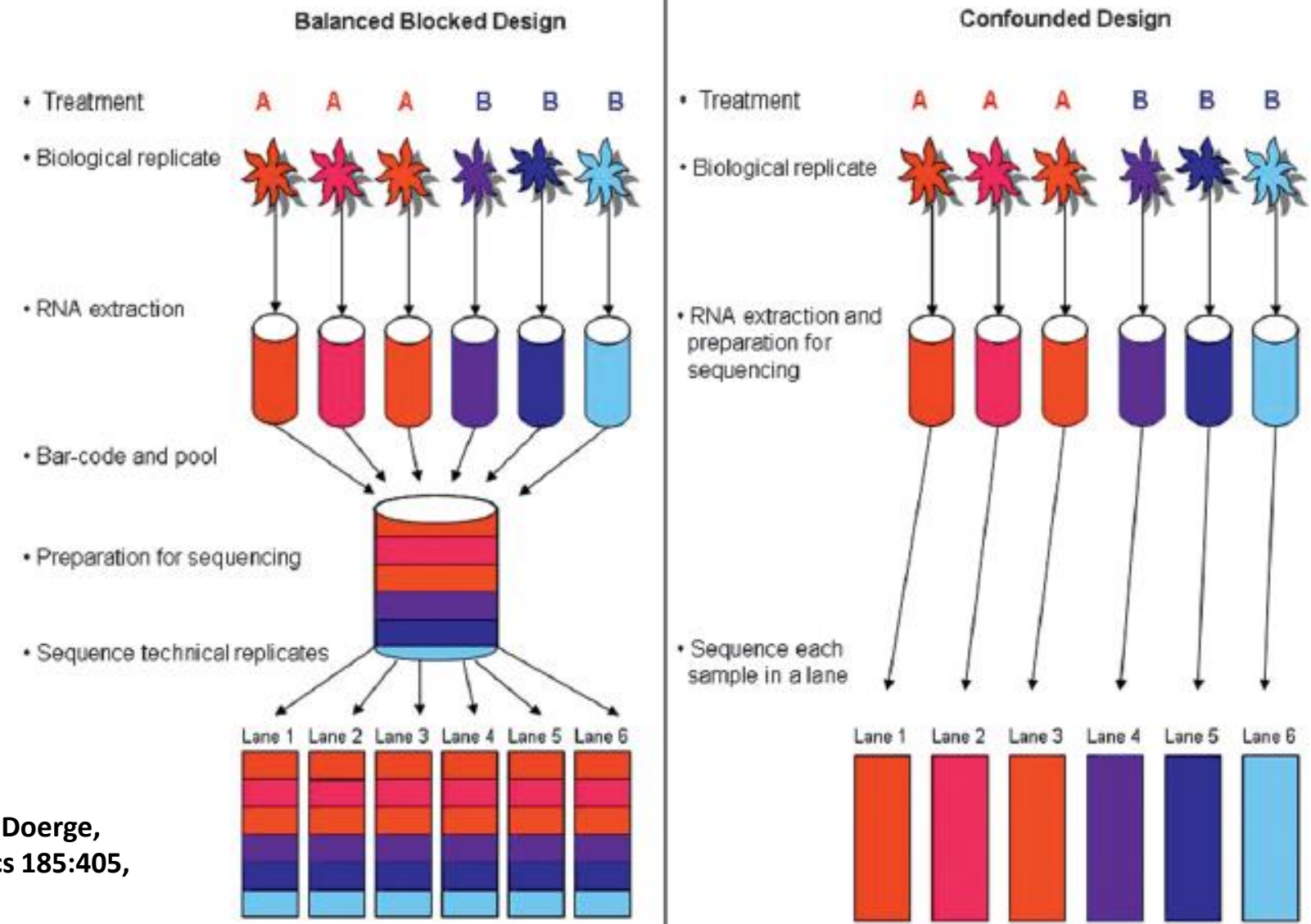
FIGURE 3.—A multiple flow-cell design based on three biological replicates within seven treatment groups. There are three flow cells with eight lanes per flow cell. The control ΦX sample is in lane 5 of each flow cell. T_{ij} refers to the jth replicate in the ith treatment group ($i = 1, \ldots, 7; j = 1, \ldots, 3$).

Here seven treatments are applied to each of three biological replicates , and each sample is sequenced in a separate lane

Are nuisance effects confounded with treatment effects?

Yes – all samples from each rep are on the same flowcell, so flowcell effects are confounded with replicate effects, and lanes with treatments (because T1 is always in lane 1, T2 in lane 2, …)

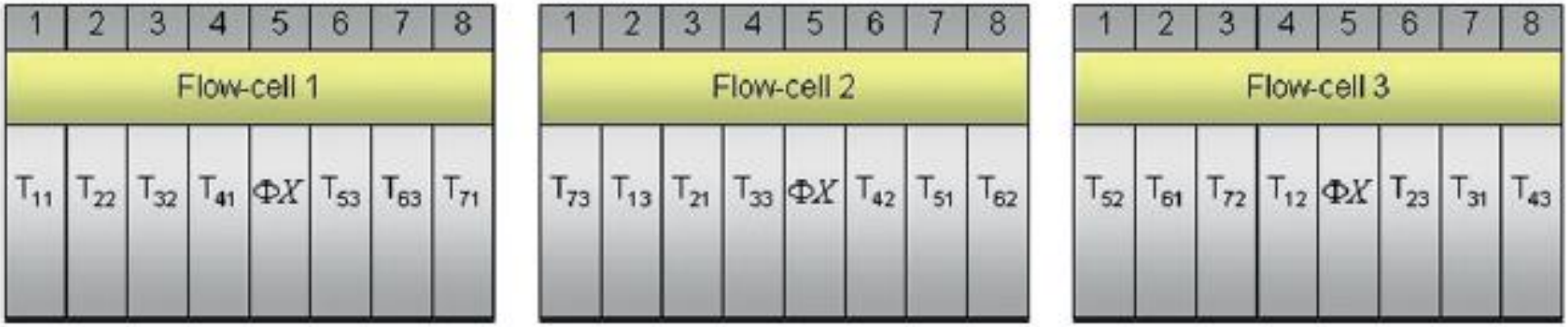
Experimental design



Auer & Doerge, Genetics 185:405, 2010

Experimental design

Auer & Doerge, Genetics 185:405, 2010

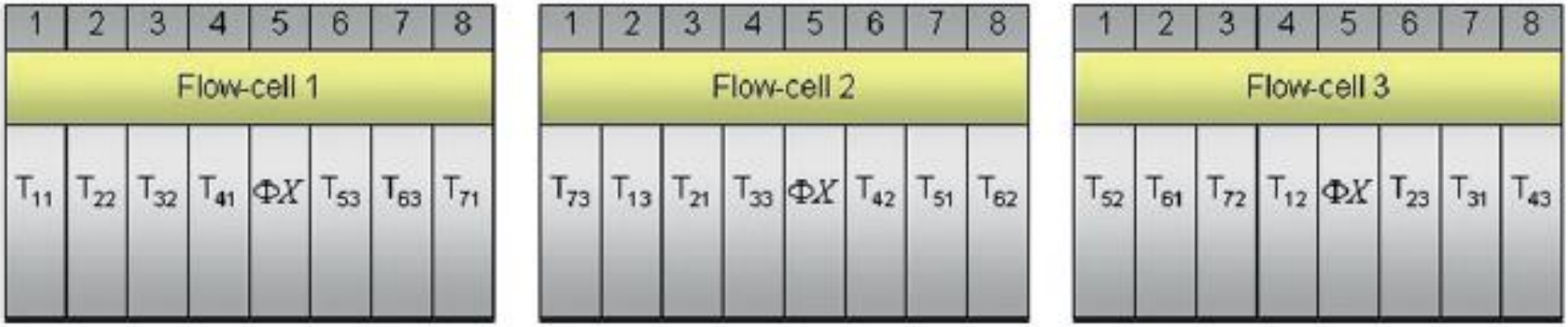


Same experiment: seven treatments applied to each of three biological replicates , but samples are allocated differently

Are nuisance effects confounded with treatment effects now?

Experimental design

Auer & Doerge, Genetics 185:405, 2010



Same experiment: seven treatments applied to each of three biological replicates , but samples are allocated differently

Are nuisance effects confounded with treatment effects now?
No – biological replicates are randomized across flowcells, and treatments are randomized across lanes

Lecture outline

- Computational resources and requirements
- Workspace environments
- Experimental design
- Data management and manipulation tools, quality assurance
- Mapping sequence reads to a reference genome
- De-novo assembly of reads without a reference genome
- Variant discovery – SNPs, small indels, and copy number variants
- Transcriptome analysis for gene discovery and gene expression measurement
- Annotation resources

Data management and manipulation tools

- FASTX-toolkit contains a variety of processing tools

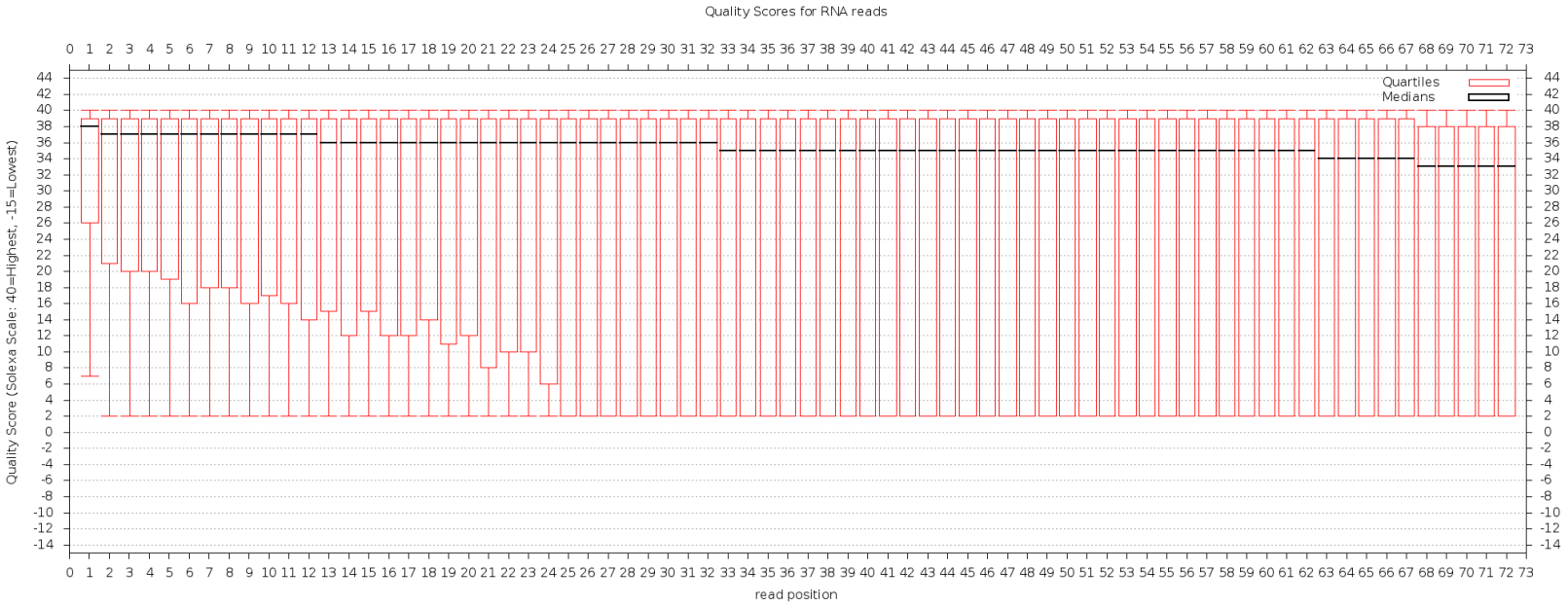
- FASTQ to FASTA converter
- FASTQ Information
- FASTQ/A Collapser
- FASTQ/A Trimmer
- FASTQ/A Renamer
- FASTQ/A Clipper
- FASTQ/A Reverse-Complement
- FASTQ/A Barcode splitter
- FASTA Formatter
- FASTA Nucleotide Changer
- FASTQ Quality Filter
- FASTQ Quality Trimmer
- FASTQ Masker

- Runs on command line or through Galaxy

Quality assurance & filtering

- "Don't waste clean thinking on dirty data" – E Racker
- FastQC – a Java program to summarize quality data from FASTQ files
  - Summarizes quality by nucleotide position in read and searches for over-represented sequences and k-mers
- flexbar – removes adapter sequences, trims low-quality regions, splits reads based on barcodes in sequences
- sickle – another quality filtering and trimming option

Fastx-toolkit – Plots of quality distribution by cycle



Error correction

- Quorum – part of the MaSuRCA assembler
  - Analyzes k-mer composition, infers accuracy of reads
  - Requires > 15x coverage, because it relies on the frequencies of k-mers to distinguish correct and incorrect k-mers
  - Produces files of corrected reads that are then used for assembly of whole-genome shotgun projects
  - Not suitable for transcriptome reads because k-mer abundance differs with transcript abundance and is not a good guide to error frequency

Assuming that k-mer sequences occur at random, the Poisson distribution can be used to model the frequency of detection ("k-mer coverage") in a set of sequencing reads. Errors are relatively infrequent, so k-mers containing errors are found at much lower levels of coverage than true k-mers. Note that 4^{15} is over 1 billion, so not all possible 15-mers can occur in the E. coli genome

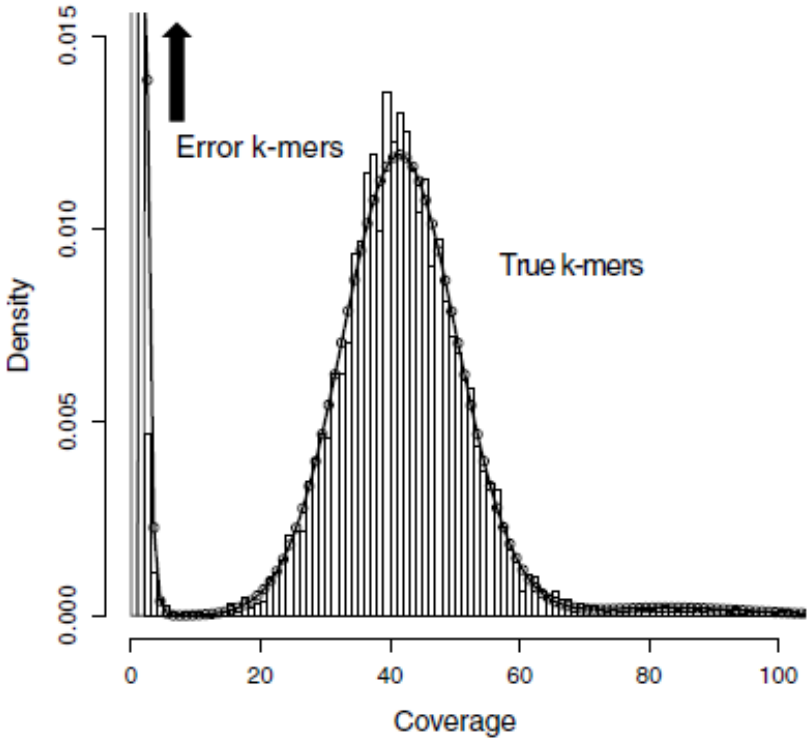


Figure 3 *k*-mer coverage. 15-mer coverage model fit to 76× coverage of 36 bp reads from *E. coli*. Note that the expected coverage of a *k*-mer in the genome using reads of length *L* will be $\frac{L-k+1}{L}$ times the expected coverage of a single nucleotide because the full *k*-mer must be covered by the read. Above, *q*-mer counts are binned at integers in the histogram. The error *k*-mer distribution rises outside the displayed region to 0.032 at coverage two and 0.691 at coverage one. The mixture parameter for the prior probability that a *k*-mer's coverage is from the error distribution is 0.73. The mean and variance for true *k*-mers are 41 and 77 suggesting that a coverage bias exists as the variance is almost twice the theoretical 41 suggested by the Poisson distribution. The likelihood ratio of error to true *k*-mer is one at a coverage of seven, but we may choose a smaller cutoff for some applications.

Figure from Kelley et al,
Genome Biol 11(11):R116, 2010

Lecture outline

- Computational resources and requirements
- Workspace environments
- Experimental design
- Data management and manipulation tools, quality assurance
- Mapping sequence reads to a reference genome
- De-novo assembly of reads without a reference genome
- Variant discovery – SNPs, small indels, and copy number variants
- Transcriptome analysis for gene discovery and gene expression measurement
- Annotation resources

Mapping sequence reads

- Two general classes
  - hash-table methods (hash reads, or hash genome)
  - suffix-array methods (Burrows-Wheeler Transform)
- Tools differ in capabilities and in performance
  - amount of memory required
  - speed of mapping
  - number of mismatches or size of indels tolerated
  - how reads mapped to multiple sites are handled

See Li et al, BMC Genomics: http://www.biomedcentral.com/1471-2164/14/S1/S13/ for an introduction to read alignment methods and software

Mapping sequence reads

- Hash-table aligners
  - Hash genome: SHRiMP, BFAST, Mosaik, GNUMap
  - Hash reads: MAQ, SOAP
- Suffix-array (Burrows-Wheeler transform) aligners
  - BWA, Bowtie, SOAP2
- Wikipedia has list of alignment software, short-read alignment programs are at bottom of page

http://en.wikipedia.org/wiki/List_of_sequence_alignment_software

Handling alignment files

- SAM (sequence alignment/mapping) and BAM (binary)
  - Widely accepted as standard file formats
  - Encode map position, quality, mismatches, indels
- Multiple software packages for manipulation
  - SAMtools – command-line
  - SAMMATE – graphical interface
- Visualization tools – Java-based
  - Integrated Genome Viewer (web or local)
  - GenomeView (web or local)
  - Integrated Genome Browser (web only)

Lecture outline

- Computational resources and requirements
- Workspace environments
- Experimental design
- Data management and manipulation tools, quality assurance
- Mapping sequence reads to a reference genome
- De-novo assembly of reads without a reference genome
- Variant discovery – SNPs, small indels, and copy number variants
- Transcriptome analysis for gene discovery and gene expression measurement
- Annotation resources

De-novo assembly

- Multiple approaches possible
  - Overlap/layout/consensus
  - De Bruijn graph
  - Hybrid approaches that combine methods
- Requires large amounts of RAM and CPU time
- Parallel-processing on a compute cluster is useful
  - ABySS and Ray programs utilize clusters
- Microbial genomes can be assembled on desktop computers

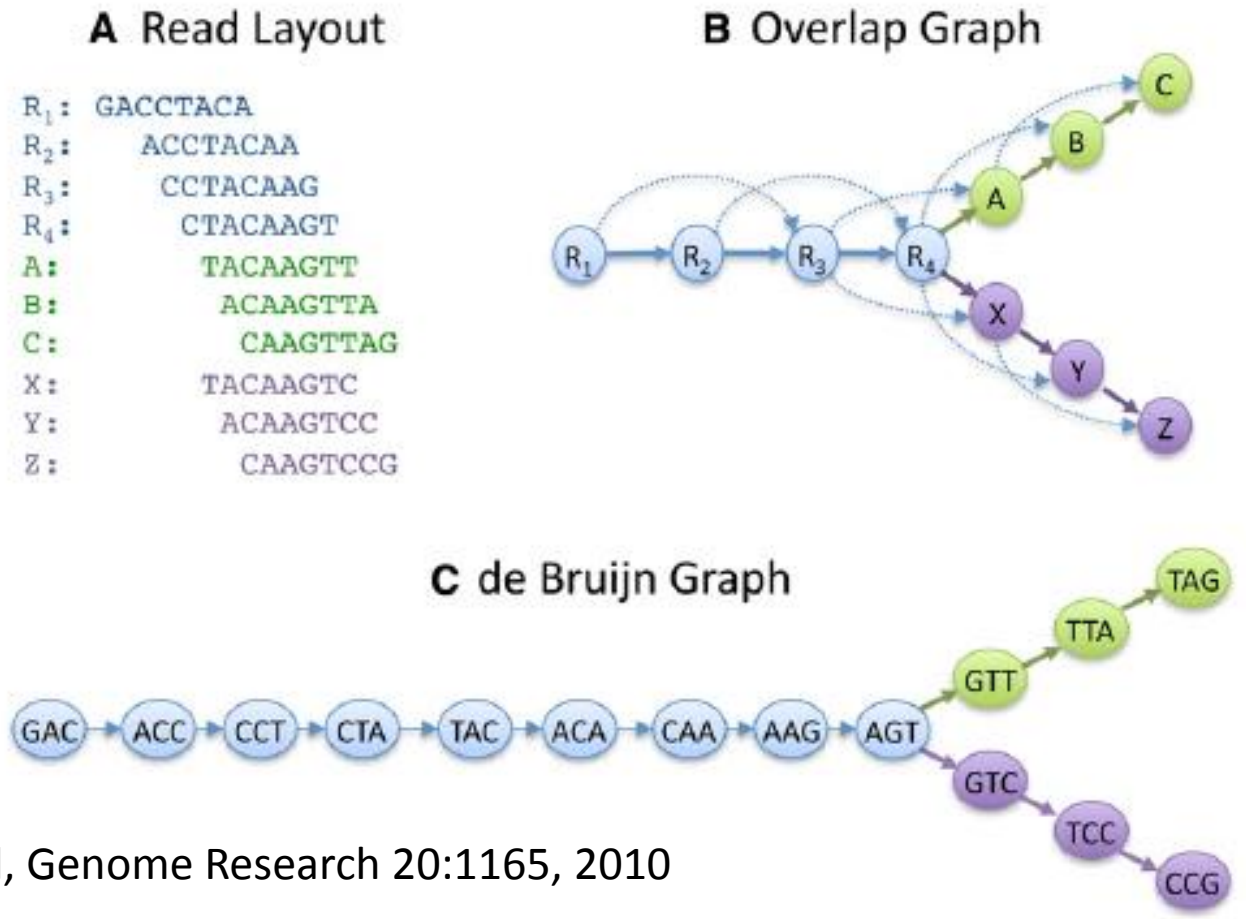
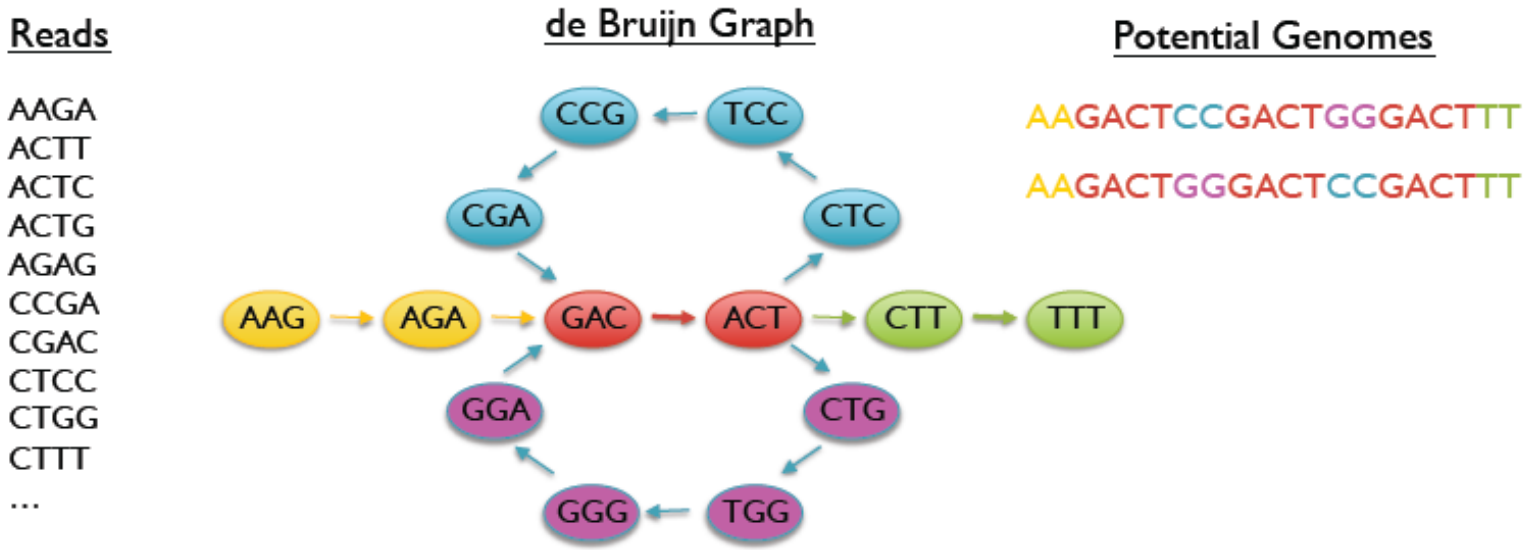


Figure from Schatz et al, Genome Research 20:1165, 2010

Overlap/Layout/Consensus uses full-length reads and finds overlaps, while de Bruijn graph uses k-mers extracted from reads, finds k-1 overlaps. Comparing frequencies of k-mers allows error correction.

Short Read Assembly



- Genome assembly as finding an Eulerian tour of the de Bruijn graph
  - Human genome: >3B nodes, >10B edges
- The new short read assemblers require tremendous computation
  - Velvet (Zerbino & Birney, 2008) serial: > 2TB of RAM x weeks
  - ABySS (Simpson *et al.*, 2009) MPI: 168 cores x ~96 hours
  - SOAPdenovo (Li *et al.*, 2010) pthreads: 40 cores x 40 hours, >140 GB RAM

ALLPATHS-LG (Gnerre *et al.*, 2011) pthreads: 48 cores, 512 Gb RAM, ~ 3.5 weeks

From a presentation by Michael Schatz, 2010

De-novo assembly recommendations

Table 1. Provisional sequencing model for de novo assembly

Libraries, insert types*	Fragment size, bp	Read length, bases	Sequence coverage, ×	Required
Fragment	180†	≥100	45	Yes
Short jump	3,000	≥100 preferable	45	Yes
Long jump	6,000	≥100 preferable	5	No‡
Fosmid jump	40,000	≥26	1	No‡

*Inserts are sequenced from both ends, to provide the specified coverage.
†More generally, the inserts for the fragment libraries should be equal to ~1.8 times the sequencing read length. In this way, the reads from the two ends overlap by ~20% and can be merged to create a single longer read. The current sequencing read length is ~100 bases.
‡Long and Fosmid jumps are a recommended option to create greater continuity.

- Paired-end sequencing provides more information than single-end sequencing
- "Paired-end" is different from "mate-pair" – the first line in this table refers to "paired-end" libraries, and the other three lines to "mate-pair" libraries.
- Multiple jump sizes of mate-pair libraries add significant value when trying to assemble genomes with substantial repetitive sequence content.

High-quality draft assemblies of mammalian genomes from massively parallel sequence data. Gnerre et al., Proc Natl Acad Sci USA 108(4): 1513 – 1518, 2011

Lecture outline

- Computational resources and requirements
- Workspace environments
- Experimental design
- Data management and manipulation tools, quality assurance
- Mapping sequence reads to a reference genome
- De-novo assembly of reads without a reference genome
- Variant discovery – SNPs, small indels, and copy number variants
- Transcriptome analysis for gene discovery and gene expression measurement
- Annotation resources

Variant discovery

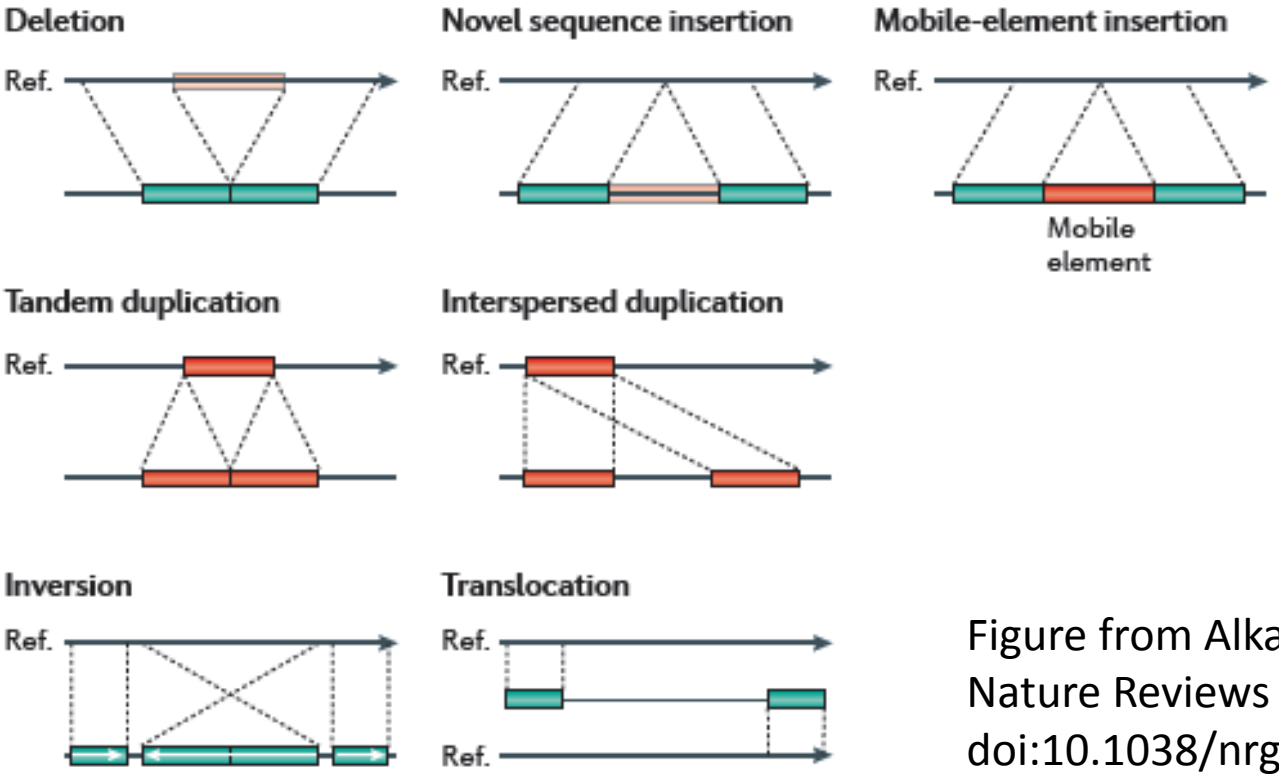


Figure from Alkan et al, Nature Reviews Genetics 2011 doi:10.1038/nrg2958

Types of variation

- Seven types of structural variation, plus SNPs (sequence variants)
- A reference genome is essential for structural variation discovery , although this may change with release of Cortex, a new assembler

Variant discovery

- Different approaches: With or without a reference?
- With a reference
  - Limiting factors are CPU time and memory required
  - Crossbow – a cluster-based cloud computing approach
- Without a reference
  - CPU time and RAM requirements are still limiting
  - Now error rate and distribution become limiting also
  - Statistical methods for estimating probability that a putative SNP is a true SNP are still developing
  - Some analytical methods require experimental designs specifically for the variant discovery objective

Four strategies for discovery of structural variation using parallel sequencing technologies

All are affected by the repetitive sequence content of the genome and by sampling error

Assembly *de novo* of a complete genome sequence is the most expensive but most complete approach

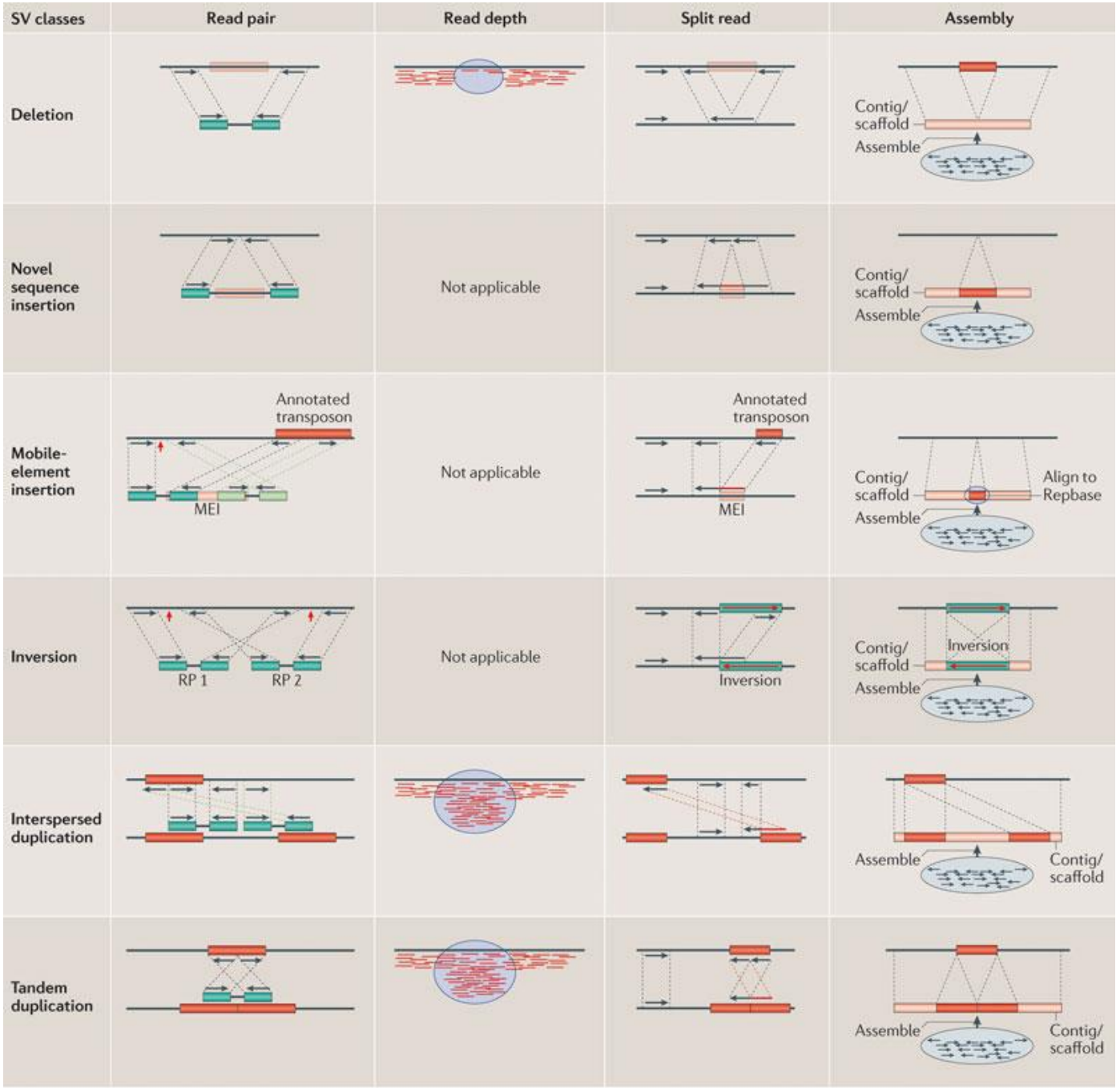


Figure from Alkan et al, Nature Reviews Genetics 2011 doi:10.1038/nrg2958

Comparison of various methods for variant discovery, based on analysis of 185 human genomes

(a) Two microarray-based methods compared with Sanger sequencing of fosmid ends (40 kb inserts) – counts include only variants > 5 kb

(b) Three parallel-sequence-based methods compared.

- The numbers of variants discovered is several-fold higher than in part (a)
- There is relatively little overlap among the variants discovered using different methods

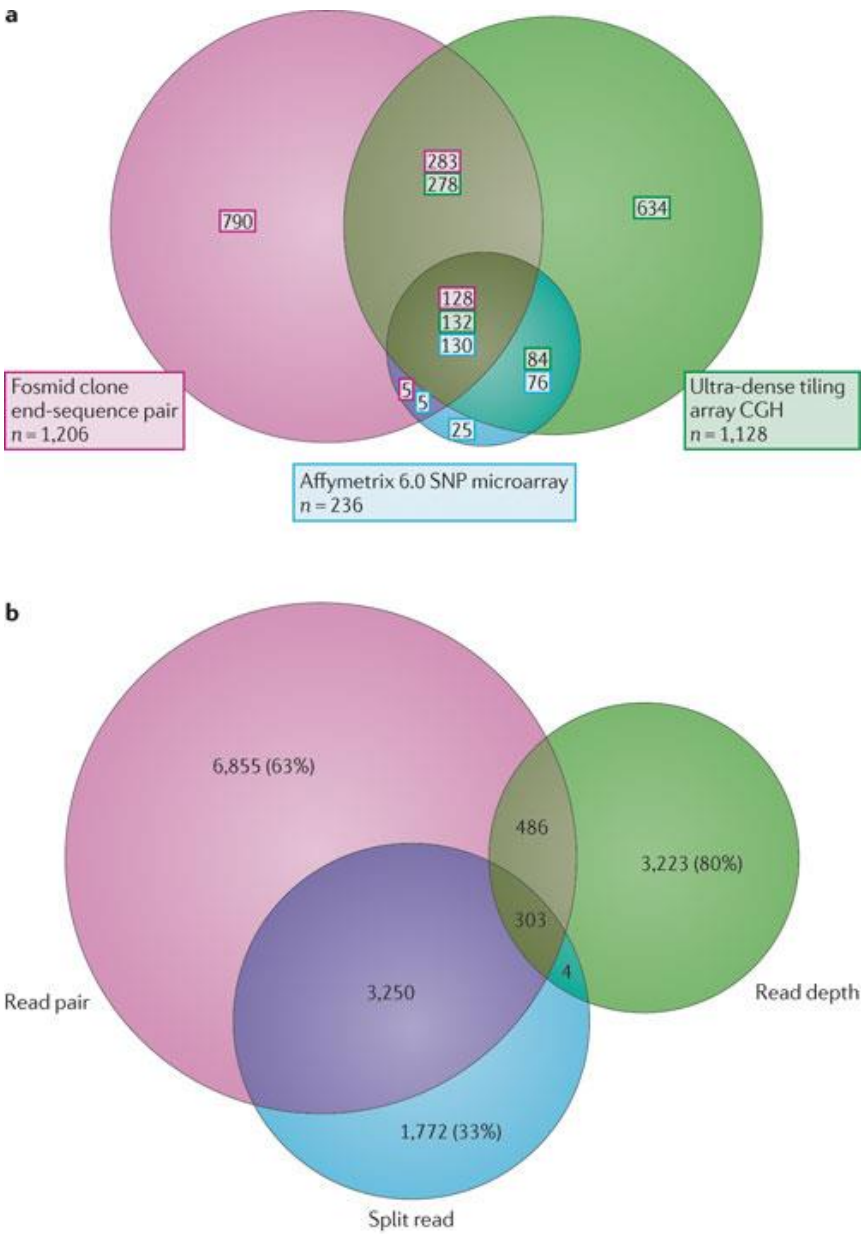


Figure from Alkan et al, Nature Reviews Genetics 2011 doi:10.1038/nrg2958

Nature Reviews | Genetics

Lecture outline

- Computational resources and requirements
- Workspace environments
- Experimental design
- Data management and manipulation tools, quality assurance
- Mapping sequence reads to a reference genome
- De-novo assembly of reads without a reference genome
- Variant discovery – SNPs, small indels, and copy number variants
- Transcriptome analysis for gene discovery and gene expression measurement
- Annotation resources

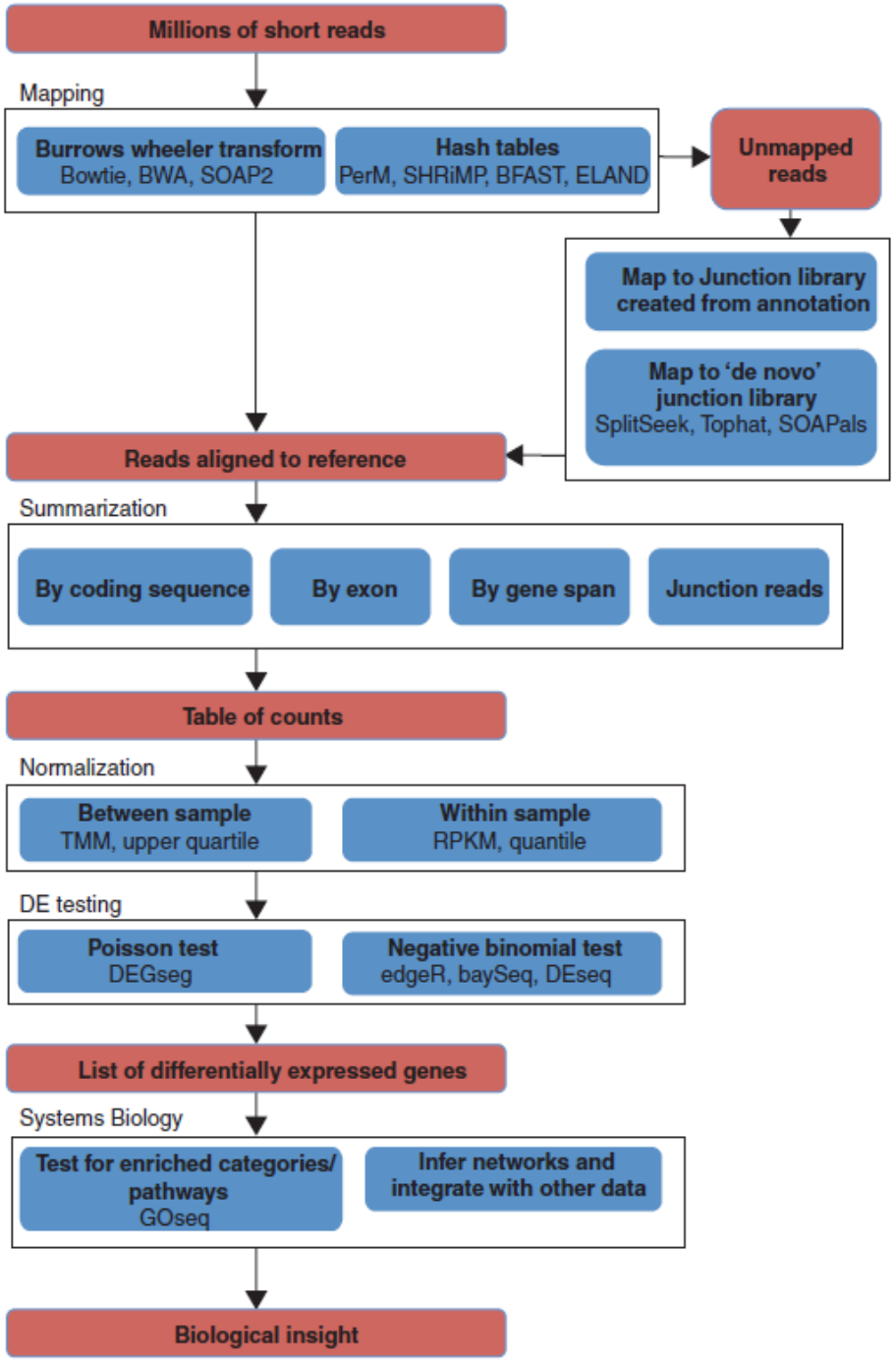
Transcriptome analysis

- With a reference
  - Challenging due to size and complexity of datasets
  - Many tools available, driven by biomedical research
  - GATK and R/Bioconductor offer many options
  - Start by mapping reads to reference genome with a mapping/alignment tool – deal with exon-intron junctions
  - Reconstruct transcripts from mapped reads – deal with alternate splicing products
  - Calculate relative abundance of different transcripts
  - Estimate biological significance based on annotation
  - Example tools: Bowtie/TopHat, Cufflinks, Myrna

BIOTECHNOLOGY
NC STATE UNIVERSITY

Workflow summary from a review "From RNA-seq reads to differential expression results", by Oshlack et al, Genome Biol 11:220, 2010.

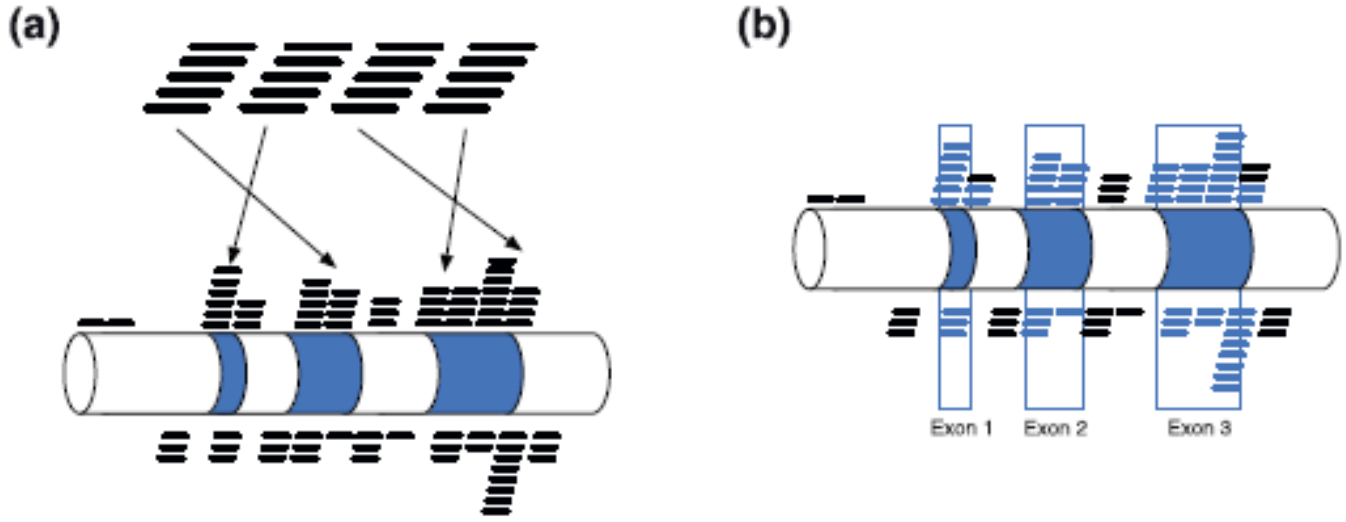
Note emphasis on statistical analysis methods; an equal emphasis should be placed on experimental design.



Transcriptome analysis

- Without a reference
  - First step is assembly
  - Transcriptome assembly pipelines
    - Velvet/Oases – Oases is a post-assembly processor for Velvet
    - Trans-ABySS (BCGSC) – based on ABySS parallel assembler
    - Rnnotator – based on Velvet
    - Trinity (Broad Institute) – a set of three programs
  - Common strategy: Assembly at multiple k-values, then merging of resulting contigs, followed by refinement
  - Once an assembly is available, continue with analysis as before

Transcriptome analysis - Myrna

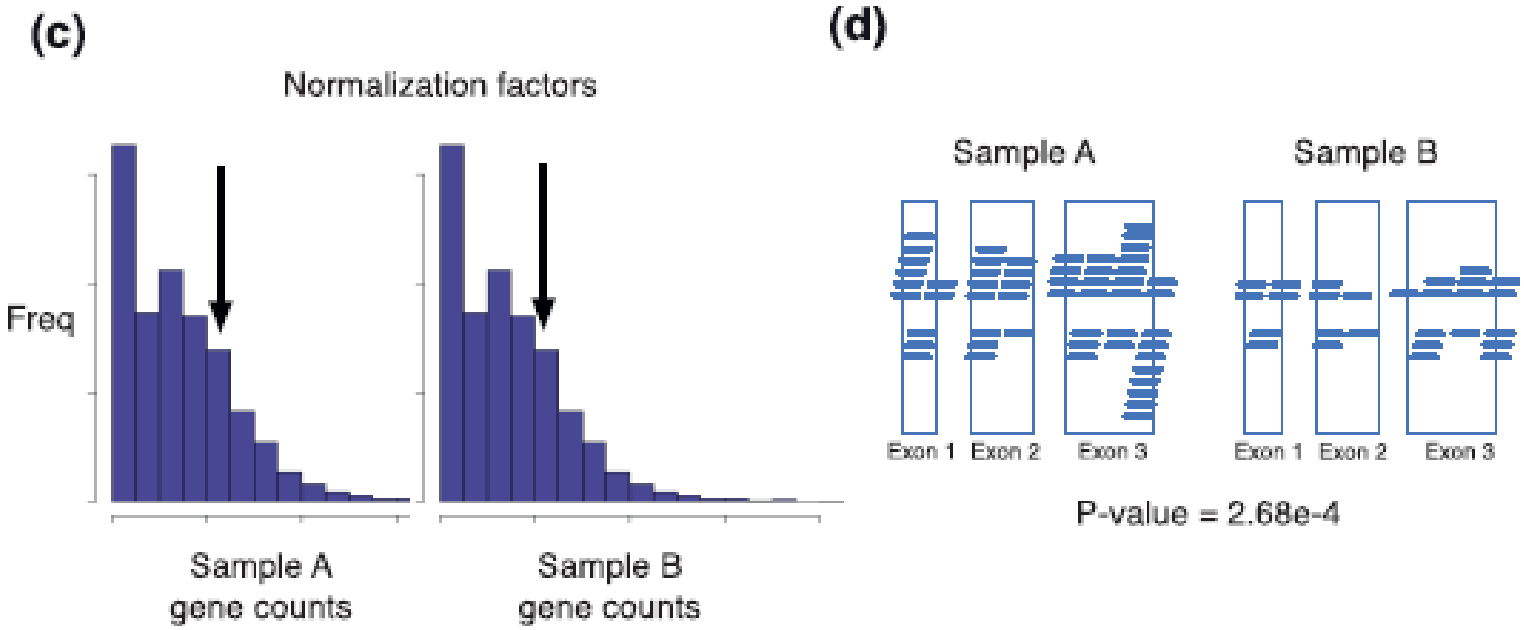


(a). Reads are aligned to genome using a parallel version of Bowtie

(b). Reads are aggregated into counts for each genomic feature –for example, each gene in the annotation files.

Figure from Langmead et al, Cloud-scale RNA-sequencing differential expression analysis with Myrna. Genome Biol.11(8):R83, 2010.

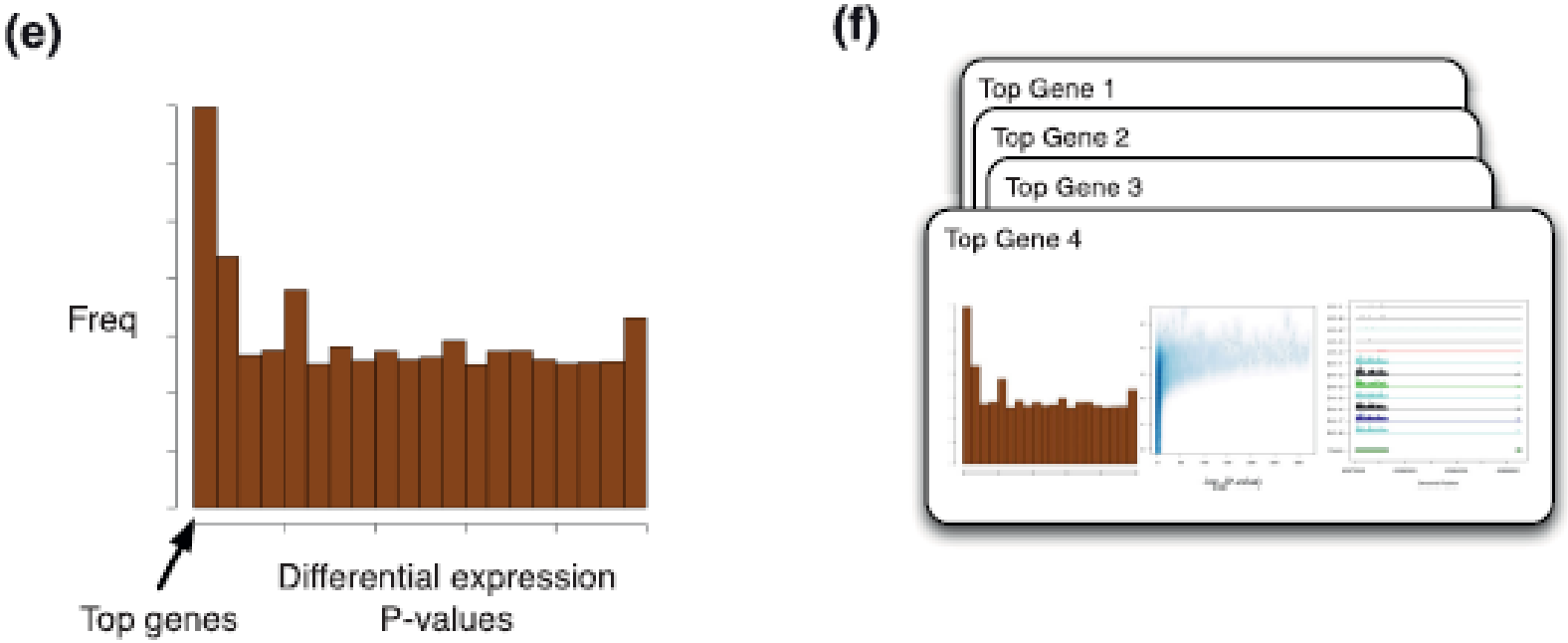
Transcriptome analysis - Myrna



(c). For each sample, a normalization constant is calculated based on a summary of the count distribution.

(d). Statistical models are used to calculate differential expression in the R programming language parallelized across multiple processors.

Transcriptome analysis - Myrna



(e). Significance summaries such as P-values and gene-specific counts are calculated and returned.

(f). Myrna also returns publication-ready coverage plots for differentially-expressed genes.

Lecture outline

- Computational resources and requirements
- Workspace environments
- Experimental design
- Data management and manipulation tools, quality assurance
- Mapping sequence reads to a reference genome
- De-novo assembly of reads without a reference genome
- Variant discovery – SNPs, small indels, and copy number variants
- Transcriptome analysis for gene discovery and gene expression measurement
- Annotation resources

Annotation resources

- BioMart provides a common platform for databases
- Currently lists 41 databases, ranging from humans to model animals to crops to microbes
- Developed by Ontario Institute for Cancer Research and European Bioinformatics Institute
  - Strong links to ENSEMBL databases of bacteria, fungi, metazoan, plant, and protist sequences and annotation
- Can be accessed through Galaxy, R/Bioconductor, and other workspace environments

Summary

Data are not information

Information is not knowledge

Knowledge is not wisdom – Anon.

Sense from sequence reads: methods for alignment and assembly.

Flicek & Birney, Nat Methods 6(11 Suppl):S6-S12, 2009

"...the individual outputs of the sequence machines are essentially worthless by themselves"

"... once analyzed collectively DNA sequencing reads have tremendous versatility. . ."

"Biologists interested in sequencing to answer their experimental questions should **prepare themselves to join a fast-moving field and embrace the tools being developed** specifically for it. As more sequence is generated, effective use of computational resources will be more and more important."