R / Bioconductor for Integrative Genomic Analysis

Martin Morgan (mtmorgan@fredhutch.org)
Fred Hutchinson Cancer Research Center

15 January 2015
Abstract – *Bioconductor* is a collection of almost 1000 packages for the analysis & comprehension of high-throughput genomic data. This general talk starts with a description of *Bioconductor* principles and their translation to software. We then discuss particular challenges and solutions for applying *R* to large-scale data, and illustrate approaches using the GenomicRanges infrastructure. The presentation concludes with interesting challenges of data integration and analysis facing *R*’s use in emerging areas of genomics and medicine.
Outline: $R$ / *Bioconductor* for Integrative Analysis

1. The *Bioconductor* project
2. High-throughput sequencing
3. Genomic Ranges
4. Large data
5. Data integration
Bioconductor

Goal  Analysis and comprehension of high-throughput genomic data

Focus  
▶ Sequencing; RNA-Seq, ChIP-Seq, Variants, . . .
▶ Expression and other microarrays; flow cytometry; proteomics, imaging

Themes  
▶ Contributions from ‘core’ members and (primarily academic) user community
▶ Based on the R programming language – statistics, visualization, interoperability
▶ Reproducible – data structures, scripts, vignettes, packages
▶ Interoperable – formal classes, dependencies on ‘core’ packages
▶ Open source / open development
Why *Bioconductor*?

A *community* of users and developers.

- Extensive & interoperable
- Statistical (volume, technology, experimental design, population samples)
- Reproducible: long-term, multi-participant science
- Leading edge: embrace novel technologies and analysis
- Accessible: affordable, transparent, usable (e.g., vignettes & man pages)

Huber et al., Orchestrating high-throughput genomic analysis with *Bioconductor*. *Nature Methods*: soon!
Why Bioconductor?

More than a software archive.

- Build on relevant software, e.g.,
  - GenomicRanges for efficient interoperability; ExpressionSet / SummarizedExperiment for genetic / phenotypic integration...
  - I/O via rtracklayer, Rsamtools, illuminaio, ...
  - Resource access via biomaRt, GEOquery, ...

- Commit to long-term support
  - e.g., affy in use 10 years after introduction.
  - Comprehensive documentation coupled with traditional scientific publications
  - Engage users via support forum, foster productive collaborations

- Enable transitions
  - User to developer
  - Student to professional

Driving principle: analysis & comprehension of high-throughput genomic data
Project status (December, 2014)

- 320,000 unique IP address package downloads / year
- 1,300 support site contributors / year, 8,200 visitors / month
- 10,500 PubMed Central mentions of ‘Bioconductor’; ≈ 22,000 citations to Bioconductor packages
- At least 12 of 15 initial TCGA publications
- Funding from US NIH & NSF, and (soon!) EC
High-Throughput Sequencing (HTS)

Questions

▶ Which genes are differentially expressed in cancer versus normal tissue?
▶ Which transcription factors are regulating gene expression?
▶ What single nucleotide polymorphisms are present in a population / associated with a disease?

Sample sizes

▶ Designed experiments – e.g., 10’s or 100’s of samples
▶ Cohorts – e.g., 100’s or 1000’s of patients
▶ Populations – 1000’s - 10000’s of individuals

Attributes

▶ 10,000’s of genes
▶ Millions of variants
HTS: Differential Expression Analysis

E.g., Gene differential expression

- Human genome: 22 autosomes, 2 sex chromosomes; 3 billion nucleotides of DNA
- DNA transcribed to mRNA, mRNA translated to proteins
- A ‘gene’: known ranges on the genome that encode proteins
- Roughly, highly expressed genes produce more mRNA

Protocol

- Isolate mRNA from tissue, reverse-transcribe to cDNA
- Fragment and then sequence cDNA – 10M - 100M fragments
- Align sequenced fragments to reference genome
- Summarize (count) aligned fragments in each gene
HTS: Differential Expression Analysis

Summarized data

\[
\begin{bmatrix}
  x_{11} & x_{12} & \ldots & x_{1n} \\
  x_{21} & x_{22} & \ldots & x_{2n} \\
  \vdots & \vdots & \ddots & \vdots \\
  x_{p1} & x_{p2} & \ldots & x_{pn}
\end{bmatrix}
\]

- Array of counts of reads aligned to \( p \) genes in \( n \) samples.

Task

- Fit a linear model to each row, \( \text{Count} \sim \text{Treatment} \)
HTS: Differential Expression Analysis

Summarized data

\[
\begin{bmatrix}
  x_{11} & x_{12} & \ldots & x_{1n} \\
  x_{21} & x_{22} & \ldots & x_{2n} \\
  \vdots & \vdots & \ddots & \vdots \\
  x_{p1} & x_{p2} & \ldots & x_{pn}
\end{bmatrix}
\]

- Array of counts of reads aligned to \( p \) genes in \( n \) samples.

Task

- Fit a linear model to each row, Count \( \sim \) Treatment

Challenges

- \( p \gg n \)
- Filtering (?)
- Sample normalization – technical variation between columns
- Negative binomial error model
- Shared experimental design – moderated test statistics
- Batch effects
HTS: Package Ecosystem

Sequencing
- FASTQ
  - ShortRead, Biostrings

Alignment
- BAM
  - Rsamtools, GenomicAlignments

Reduction
- Counts (.csv)
  - Base R, GenomicAlignments, Rsубread
  - Peaks (.bed, .wig)
    - rtracklayer
  - Variants (.vcf)
    - VariantAnnotation, VariantTools, h5vc
    - ...

Analysis
- Differential expression (genes, transcripts)
  - edgeR, DESeq, DEXSeq, ...
  - Annotation; Differential binding
    - ChIPpeakAnno, DiffBind, ...
- Effect prediction; GWAS
  - ensemblVEP, snpStats, ...
  - ...

Integration & Visualization
- IRanges, GenomicRanges, GenomicAlignments
- AnnotationDbi, GenomicFeatures, org.*, TxDB*, biomaRt, PSICQUIC, KEGGREST, ...
- Gvis, ggbioma, epivisr, rtracklayer (UCSC), SRAdb (IGV)
Genomic Ranges

A central concept
- Chromosome, start, end, strand provide coordinates specifying where in the genome a range occurs
- Describes data, e.g., aligned reads, and annotation, e.g., locations of genes

Many useful operations are based on genomic ranges
- E.g., reduction of aligned reads to a matrix of counts represents a simple tally of the number of overlaps between genomic ranges describing aligned reads and genomic ranges described gene locations.

Software
- GenomicRanges, GenomicAlignments, GenomicFeatures
- GRanges, GRangesList
Genomic Ranges: **GRanges**

```r
> gr = exons(TxDb.Hsapiens.UCSC.hg19.knownGene); gr
GRanges with 289969 ranges and 1 metadata column:

<table>
<thead>
<tr>
<th>seqnames</th>
<th>ranges</th>
<th>strand</th>
<th>exon_id</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Rle&gt;</td>
<td>&lt;IRanges&gt;</td>
<td>&lt;Rle&gt;</td>
<td>&lt;integer&gt;</td>
</tr>
<tr>
<td>[1]</td>
<td>chr1 [11874, 12227]</td>
<td>+</td>
<td>1</td>
</tr>
<tr>
<td>[2]</td>
<td>chr1 [12595, 12721]</td>
<td>+</td>
<td>2</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>[289967]</td>
<td>chrY [59358329, 59359508]</td>
<td>-</td>
<td>277748</td>
</tr>
<tr>
<td>[289968]</td>
<td>chrY [59360007, 59360115]</td>
<td>-</td>
<td>277749</td>
</tr>
<tr>
<td>[289969]</td>
<td>chrY [59360501, 59360854]</td>
<td>-</td>
<td>277750</td>
</tr>
</tbody>
</table>
```

**seqlengths:**

<table>
<thead>
<tr>
<th>seqnames</th>
<th>lengths</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr1</td>
<td>249250621</td>
</tr>
<tr>
<td>chr2</td>
<td>243199373</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>chrUn.g1000249</td>
<td>38502</td>
</tr>
</tbody>
</table>

**Seqinfo**

- seqlevels(gr)
- seqlengths(gr)
- genome(gr)
A useful summary table of genomic ranges operations is in PLOS Computational Biology 10.1371/journal.pcbi.1003118.
Genomic Ranges

Building blocks for range-based data structures

- GAlignments, GAlignmentsList (GenomicAlignments)
- SummarizedExperiment (GenomicRanges)
- VCF (VariantAnnotation)

Example

- What genic regions (coding, intron, 5’ or 3’ UTR, promoter, …) do SNPs occur in?

library(VariantAnnotation)
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
param <- ScanVcfParam(info=NA, geno=NA)
vcf <- readVcf("my.vcf", "hg19", param)
locateVariants(vcf, TxDb.Hsapiens.UCSC.hg19.knownGene)
Genomic Ranges: GRanges Implementation

- Recall: R works well on vectors; object creation is expensive
- GRanges class models columns of data; one class instance for millions of ranges.
- Vector-like API – length, [], [[]] returns number and subset of ranges
- DataFrame metadata associated with ranges
Genomic Ranges: **GRangesList**

```r
> grl = exonsBy(TxDb.Hsapiens.UCSC.hg19.knownGene, "tx", use.names=TRUE);
> grl
GRangesList of length 82960:

$uc001aa.3
GRanges with 3 ranges and 3 metadata columns:

<table>
<thead>
<tr>
<th>seqnames</th>
<th>ranges</th>
<th>strand</th>
<th>exon_id</th>
<th>exon_name</th>
<th>exon_rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr1</td>
<td>[11874, 12227]</td>
<td>+</td>
<td>1</td>
<td>&lt;NA&gt;</td>
<td>1</td>
</tr>
<tr>
<td>chr1</td>
<td>[12613, 12721]</td>
<td>+</td>
<td>3</td>
<td>&lt;NA&gt;</td>
<td>2</td>
</tr>
<tr>
<td>chr1</td>
<td>[13221, 14409]</td>
<td>+</td>
<td>5</td>
<td>&lt;NA&gt;</td>
<td>3</td>
</tr>
</tbody>
</table>

$uc010nx.1
GRanges with 3 ranges and 3 metadata columns:

<table>
<thead>
<tr>
<th>seqnames</th>
<th>ranges</th>
<th>strand</th>
<th>exon_id</th>
<th>exon_name</th>
<th>exon_rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr1</td>
<td>[11874, 12227]</td>
<td>+</td>
<td>1</td>
<td>&lt;NA&gt;</td>
<td>1</td>
</tr>
<tr>
<td>chr1</td>
<td>[12595, 12721]</td>
<td>+</td>
<td>2</td>
<td>&lt;NA&gt;</td>
<td>2</td>
</tr>
<tr>
<td>chr1</td>
<td>[13403, 14409]</td>
<td>+</td>
<td>6</td>
<td>&lt;NA&gt;</td>
<td>3</td>
</tr>
</tbody>
</table>

$uc010nxr.1
GRanges with 3 ranges and 3 metadata columns:

<table>
<thead>
<tr>
<th>seqnames</th>
<th>ranges</th>
<th>strand</th>
<th>exon_id</th>
<th>exon_name</th>
<th>exon_rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr1</td>
<td>[11874, 12227]</td>
<td>+</td>
<td>1</td>
<td>&lt;NA&gt;</td>
<td>1</td>
</tr>
<tr>
<td>chr1</td>
<td>[12646, 12697]</td>
<td>+</td>
<td>4</td>
<td>&lt;NA&gt;</td>
<td>2</td>
</tr>
<tr>
<td>chr1</td>
<td>[13221, 14409]</td>
<td>+</td>
<td>5</td>
<td>&lt;NA&gt;</td>
<td>3</td>
</tr>
</tbody>
</table>

... <82957 more elements>

---

seqlengths:

<table>
<thead>
<tr>
<th>seqnames</th>
<th>length</th>
</tr>
</thead>
<tbody>
<tr>
<td>chr1</td>
<td>249250621</td>
</tr>
<tr>
<td>chr2</td>
<td>243199373</td>
</tr>
<tr>
<td>chrUn.g1000249</td>
<td>38502</td>
</tr>
</tbody>
</table>
```

**Two kinds of fun!**

```r
introns = psetdiff(range(grl), grl)
grr = unlist(grl)
# transform grr, then...
grl = relist(grr, grl)
```

‘flesh’ ‘skeleton’
Genomic Ranges: \textit{GRangesList} Implementation

- List-like where each element of the list is must be a \textit{GRanges}
- Implementation: a single \textit{GRanges} instance, and a \textit{partitioning} describing how ranges are grouped into list elements
- Only two objects
- Some operations can be very fast – unlist, transform, relist.
Strategies for Large Data

Memory management

▶ Restrict input to relevant ‘columns’, e.g., readGAlignments inputs only columns necessary to describe geometry of alignment.
▶ Select relevant rows, e.g., ScanBamParam(which=...)
▶ Iterate: read in and operate on successive chunks – e.g., open(BamFile(..., yieldSize=1e7)); reduceByYield(...)

Speed

▶ Efficient R code – 10-100× speed-up
▶ Parallel evaluation – 2-10× speed-up
▶ Often implies memory management
▶ BiocParallel, GenomicFiles
Integrative Analysis: Annotation

- Gene identifiers (e.g., org.Hs.eg.db) and models (e.g., TxDb.Hsapiens.UCSC.hg19.knownGene)
- Web-based resources (e.g., biomaRt, KEGGREST, UniProt.ws)
- Whole-genome annotations via AnnotationHub, e.g., Ensembl, UCSC, ad hoc
Integrative Analysis: *AnnotationHub*

File-based resources, e.g., UCSC *liftOver* files

```r
## hg19SNPs <- GRanges(...)  
library(AnnotationHub)  
hub <- AnnotationHub()  
chain <- query(hub, 'hg19ToHg38')[[1]]  
hg38SNPs <- liftOver(hg19SNPs, chain)
```

Annotation-style resources, e.g., *grasp2*

```r
library(grasp2db) # Annotation package,  
    # 6 Gb AnnotationHub resource  
d <- GRASP2() # dplyr instance  
hispanic <- tbl(d, "count") %>%  
    filter(Population=="Hispanic")  
semi_join(tbl(d, "variant"), hispanic)
```
Integrative Analysis: *SummarizedExperiment / ExpressionSet*

- Co-ordinate subsetting of ‘data’ and row (e.g., genomic location) or column (e.g., sample treatment) metadata

```r
se <- SummarizedExperiment(
  assays,
  rowData,
  colData,
  exptData
)
se %in% CNVs
```
Integrative Analysis: Diverse Data Types

- Co-ordinated management of diverse data types
- In-memory and on-disk
- e.g., Identifier / genomic ranges conversion, \( x[i, , ] \)
- e.g., \( j = \)
  \[ \text{complete.cases}(x$mRNA, x$miRNA); x[, j, ] \]
- e.g., data type selection, \( x[, , c("mRNA", "miRNA") \]
- Curated collections of public integrated data sets

TCGA Ovarian gene expression / copy number correlation

flagged samples

Frequency

0 0.05 0.10 0.15
0 10 20 30 40 50
Future events

- Computational Statistics for Genome Biology (CSAMA), 15-19 June, Brixen / Bressanone, Italy
- useR!, 1-3 July, Aalborg, Denmark
- BioC 2015, 20 - 22 July, Seattle, WA USA
Acknowledgments

Core (Seattle): Sonali Arora, Marc Carlson, Nate Hayden, Valerie Obenchain, Hervé Pagès, Paul Shannon, Dan Tenenbaum.


Scientific Advisory Board: Paul Flicek, Simon Tavaré, Simon Urbanek.