R / *Bioconductor* for Everyone

Martin Morgan[1]
Fred Hutchinson Cancer Research Center
Seattle, WA

30 July 2014

[1] mtmorgan@fhcrc.org

Abstract

This lab provides an introduction to *R* / *Bioconductor* for high-throughput sequence analysis. It is designed for those who have some but not a lot of familiarity with R and Bioconductor. The first part of the lab focuses on *R* data types, functions, classes, methods, the package and help systems, and the Bioconductor web site. The second part of the lab takes a quick tour of essential packages, classes, and methods for sequence analysis. We will make brief stops at *GenomicRanges*, *Biostrings*, *GenomicFeatures*, *ShortRead*, *Rsamtools*, *rtracklayer*, *AnnotationDbi*, and other packages of interest to participants.

Outline

R and *Bioconductor*

R

- http://r-project.org
- Open-source, statistical programming language; widely used in academia, finance, pharma, ...
- Core language, 'base' and $>$ 4000 contributed packages
- Interactive sessions, scripts, packages

Bioconductor

- http://bioconductor.org
- Analysis and comprehension of high-throughput genomic data
- Themes: rigorous statistical analysis; reproducible work flows; integrative analysis
- $>$ 11 years old, $>$ 670 packages

Basic data types

- Vectors of *logical*, *integer*, *numeric*, *complex*, *character*, or *raw* types
- Statistical concepts such as *factor*, `NA`
- More complicated data structures: *data.frame*, *matrix*, *list*
- Object-oriented classes – 'S3' and 'S4' systems

```
> df <- data.frame(
+           age = c(27, 32, 19),
+           sex = factor(c("Male", "Female", NA)))
> df

  age    sex
1  27   Male
2  32 Female
3  19   <NA>
```

Functions

- Typically, act on *vectors*
- Required and / or optional arguments
- Matching by name, then position

```
> y <- 5:1     # vector: 5, 4, 3, 2, 1
> log(y)       # log of each element, 'vectorized'

[1] 1.6094379 1.3862944 1.0986123 0.6931472 0.0000000

> args(log)    # discovery; argument 'base' has default

function (x, base = exp(1))
NULL

> log(base=2, y) # match by name, then position

[1] 2.321928 2.000000 1.584963 1.000000 0.000000
```

Classes and methods

- Coordinate complicated data
- *methods* specialize functions; *accessors*

```
> x <- rnorm(1000, sd=1); y <- x + rnorm(1000, sd=.5)
> fit <- lm(y ~ x); class(fit)
[1] "lm"
> head(methods(class=class(fit)), 3)
[1] "add1.lm"  "alias.lm" "anova.lm"
> anova(fit)
Analysis of Variance Table

Response: y
           Df Sum Sq Mean Sq F value    Pr(>F)
x           1 981.15  981.15  4017.4 < 2.2e-16 ***
Residuals 998 243.73    0.24
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
```

S4 classes and methods

- S4 a more formal class system, used extensively in *Bioconductor*

```
> library(Biostrings)
> dna <- DNAStringSet(c("AACA", "ATTA"))
> ## showMethods(class=class(dna),
> ##     where=search())
> alphabetFrequency(dna, baseOnly=TRUE)
```

```
     A C G T other
[1,] 3 1 0 0     0
[2,] 2 0 0 2     0
```

Packages

- Core and contributed; many
- Technical standards imposed, e.g., *man* page for each exposed function, *Bioconductor* vignettes, examples
- Considerable room for author personality, quality variation
- `biocLite` to install a new package, once only
- `library` to attach an installed package

Installation – once only

```
> source("http://bioconductor.org/biocLite.R")
> biocLite("ShortRead")  # install 'ShortRead' package
> biocLite()             # update all installed packages
> library(ShortRead)     # attach to current session
```

Help

```
> help.start()
> ?data.frame
> ?anova
> ?anova.lm     # anova generic, method for class lm
> class ? DNAStringSet
> method ? "alphabetFrequency,DNAStringSet"
> vignette("GenomicRangesIntroduction", "GenomicRanges")
> help(package="Biostrings")
> RShowDoc("R-intro")
```

Useful functions

`dir`, `read.table`, `scan` List files; input data.

`c`, `factor`, `data.frame`, `matrix` Create vectors, etc.

`summary`, `table`, `xtabs` Summarize or cross-tabulate data.

`t.test`, `lm`, `anova` Compare two or several groups.

`dist`, `hclust` Cluster data.

`plot` Plot data.

`ls`, `library` List objects; attach packages.

`lapply`, `sapply`, `mapply` Apply function to elements of lists.

`match`, `%in%` find elements of one vector in another.

`split`, `cut` Split or cut vectors.

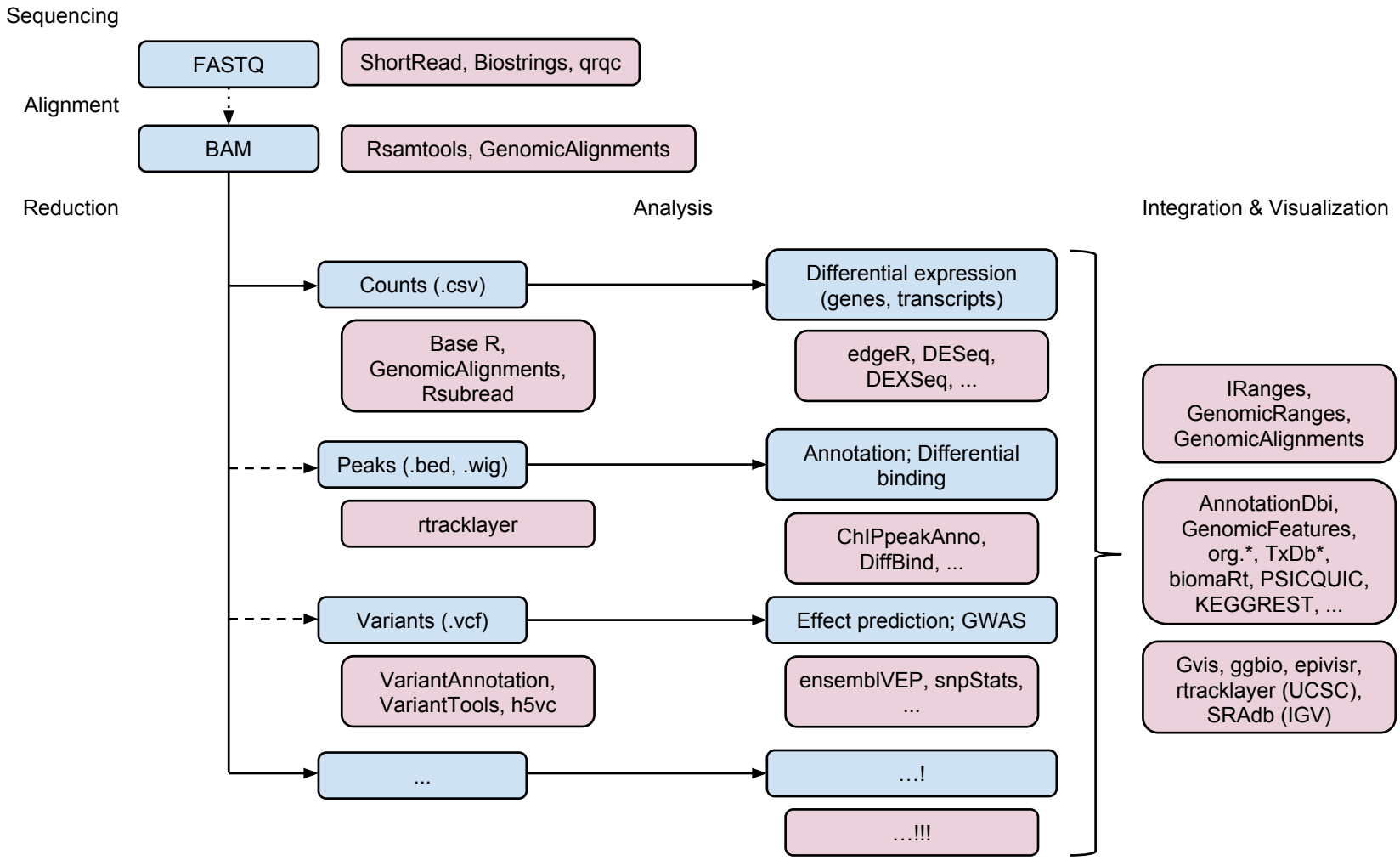
`strsplit`, `grep`, `sub` Operate on character vectors.

`biocLite` Install a package from an on-line repository.

`traceback`, `debug`, `browser` Help debug errors.

Outline

Sequencing Work Flow / Packages



Reads

Data	Short reads and their qualities
Tasks	Input, quality assessment, summary, trimming, ...
Packages	*ShortRead*, *Biostrings*
Functions	▶ `readFastq`, `FastqSampler`, `FasqtStreamer`. ▶ `qa`, `report`, `alphabetFrequency`, `alphabetByCycle`, `consensusMatrix`. ▶ `trimLRPatterns`, `matchPDict`, ...

Sequences

Data Whole-genome sequences

Tasks View sequences, match position weight matricies, match patterns

Packages *Biostrings*, *BSgenome*

Functions

- `available.genomes`
- `Hsapiens[["chr3"]]`, `getSeq`, `mask`
- `matchPWM`, `vcountPattern`, `...`
- `forgeBSgenomeDataPkg`

Alignments

Data	BAM files of aligned reads
Tasks	Input, BAM file manipulation, pileups
Packages	*Rsamtools* (also: *GenomicRanges*)
Functions	▶ `BamFile`, `BamFileList` ▶ `scanBam`, `ScanBamParam` (select a subset of the BAM file) ▶ `asBam`, `sortBam`, `indexBam`, `mergeBam`, `filterBam` ▶ `BamSampler`, `applyPileups`

Ranges

Data	Genomic coordinates to represent data (e.g., aligned reads) or annotation (e.g., gene models).
Tasks	Input, counting, coverage, manipulation, ...
Packages	*GenomicAlignments*, *GenomicRanges*, *IRanges*
Functions	▶ `readGAlignments`, `readGAlignmentsList` ▶ Many intra-, inter-, and between-range manipulating, e.g., `narrow`, `flank`, `shift`, `intersect`, `findOverlaps`, `countOverlaps`

Features

Data	Genomic coordinates
Tasks	Group exons by transcript or gene; discover transcript / gene identifier mappings
Packages	*GenomicFeatures* and *TxDb.** packages (also: *rtracklayer*)
Functions	▶ `exonsBy`, `cdsBy`, `transcriptsBy` ▶ `select` (see Annotations, below) ▶ `makeTranscriptDb*`

Annotations

Data	Gene symbols or other identifiers
Tasks	Discover annotations associated with genes or symbols
Packages	*AnnotationDbi* (*org.**, *GO.db*, ...), *biomaRt*
Functions	▶ Discovery: `cols`, `keytype`, `keys` ▶ `select`, `merge` ▶ *biomaRt*: `listMarts`, `listDatasets`, `listAttributes`, `listFilters`, `getBM`

Genome annotations

Data	FASTA, GTF, VCF, . . . from internet resources
Tasks	Define regions of interests; incorporate known features (e.g., ENCODE marks, dbSNP variants) in work flows
Packages	*AnnotationHub*
Functions	▶ `AnnotationHub`, `filters` ▶ `metadata`, `hub$<tab>`

Import / export

Data	Common text-based formats, `gff`, `wig`, `bed`; UCSC tracks
Tasks	Import and export
Packages	*rtracklayer*
Functions	▶ `import`, `export` ▶ `browserSession`, `genome`

RNA-seq differential representation

Data	Counts of reads per gene across samples in designed experiments
Tasks	Identify differentially expressed genes or exons
Packages	*edgeR*, *DESeq2*, *DEXSeq*, *goseq*
Functions	▶ ...

Variants

Data	VCF (Variant Call Format) file
Tasks	Calling, input, summary, coding consequences
Packages	*VariantTools* (linux only), *VariantAnnotation*, *ensemblVEP*
Functions	▶ `tallyVariants` ▶ `readVcf`, `locateVariants`, `predictCoding` ▶ Also: SIFT, PolyPhen data bases

And...

Data representation: *IRanges*, *GenomicRanges*, *GenomicFeatures*, *Biostrings*, *BSgenome*, *girafe*. Input / output: *ShortRead* (fastq), *Rsamtools* (bam), *rtracklayer* (gff, wig, bed), *VariantAnnotation* (vcf), *R453Plus1Toolbox* (454). Annotation: *GenomicFeatures*, *ChIPpeakAnno*, *VariantAnnotation*. Alignment: *Rsubread*, *Biostrings*. Visualization: *ggbio*, *Gviz*. Quality assessment: *qrqc*, *seqbias*, *ReQON*, *htSeqTools*, *TEQC*, *Rolexa*, *ShortRead*. RNA-seq: *BitSeq*, *cqn*, *cummeRbund*, *DESeq*, *DEXSeq*, *EDASeq*, *edgeR*, *gage*, *goseq*, *iASeq*, *tweeDEseq*. ChIP-seq, etc.: *BayesPeak*, *baySeq*, *ChIPpeakAnno*, *chipseq*, *ChIPseqR*, *ChIPsim*, *CSAR*, *DiffBind*, *MEDIPS*, *mosaics*, *NarrowPeaks*, *nucleR*, *PICS*, *PING*, *REDseq*, *Repitools*, *TSSi*. Motifs: *BCRANK*, *cosmo*, *cosmoGUI*, *MotIV*, *seqLogo*, *rGADEM*. 3C, etc.: *HiTC*, *r3Cseq*. Copy number: *cn.mops*, *CNAnorm*, *exomeCopy*, *seqmentSeq*. Microbiome: *phyloseq*, *DirichletMultinomial*, *clstutils*, *manta*, *mcaGUI*. Work flows: *ArrayExpressHTS*, *Genominator*, *easyRNASeq*, *oneChannelGUI*, *rnaSeqMap*. Database: *SRAdb*. ...

Outline

Resources

- Packages and their vignettes:
  http://bioconductor.org/packages/release
- Course and conference material:
  http://bioconductor.org/help/course-materials
- Introduction to *R* – `RShowDoc('R-intro')`
- Mailing list
  http://bioconductor.org/help/mailing-list for
  support

Acknowledgements

- *Bioconductor* team: Sonali Arora, Marc Carlson, Nate Hayden, Valerie Obenchain, Hervé Pagès, Paul Shannon, Dan Tenenbaum
- Technial advisory council: Vincent Carey, Wolfgang Huber, Robert Gentleman, Rafael Irizzary, Sean Davis, Kasper Hansen
- Scientific advisory board: Simon Tavaré, Vivian Bonazzi, Vincent Carey, Wolfgang Huber, Robert Gentleman, Rafael Irizzary, Paul Flicek, Simon Urbanek.
- NIH / NHGRI U41HG0004059
- . . . and the *Bioconductor* community!