

Lecture 2: Container designs and methods

©2008 VJ Carey, Ph.D.
Channing Lab

1. concepts
 - (a) metadata binding
 - (b) closure under subsetting
 - (c) formal class assignment supports multiple dispatch
2. `expr+SNP`: `racExSet`
3. Gene sets
4. networks/pathways
5. hg18 anno tracks
6. genomic strings
7. machine learning output containers

1

The metadata twist

1. experimental output should include provenance documentation
 - (a) `experimentData` component
2. assay reporter nomenclature resolution must be supported
 - (a) `annotation/featureData` component
3. variable names may need clarification
 - (a) `varMetadata` component

3

Container concepts

1. Recall schematic: N samples, G features are assayed, R sample-level variables (`phenoData`) are collected
2. basic matrix accessor idioms in R:
 - (a) `X[G,]` – all columns, rows identified by G
 - (b) `X[, S]` – all rows, columns identified by S
 - (c) `X[G, S]` – both rows and columns restricted
3. Bioconductor preserves that general concept with two twists
 - (a) twist 1: introspection-oriented metadata must be bound in to X
 - (b) twist 2: the class of X must be closed under subsetting operations

2

```
> afxsubRMAES
ExpressionSet (storageMode: lockedEnvironment)
assayData: 54675 features, 24 samples
element names: exprs
phenoData
 sampleNames: AFX_1_A1.CEL, AFX_1_A2.CEL, ..., AFX_3_D2.CEL (24 total)
 varLabels and varMetadata description:
  site: from cel
  samp: rna src/mixture code
  repl: replicate
  pctBrain: pct of mixture from Ambion brain
featureData
 featureNames: 1007_s_at, 1053_at, ..., AFFX-TrpmX-M_at (54675 total)
 fvarLabels and fvarMetadata description: none
experimentData: use 'experimentData(object)'
pubMedIds: 16964226
Annotation: hgu133plus2
```

4

```
> experimentData(afxsubRMAES)
Experiment data
Experiment name: Shippy R
Laboratory: GE Healthcare, 7700 S. River Pkwy., Suite #2603, Tempe, Arizona
Contact information:
Title: Using RNA sample titrations to assess microarray platform performance
URL:
PMIDs: 16964226
```

```
Abstract: A 158 word abstract is available. Use 'abstract' method.
```

5

```
> varMetadata(afxsubRMAES)
site          labelDescription
samp          from cel
repl          rna src/mixture code
pctBrain     replicate
              from Ambion brain
```

```
> featureData(afxsubRMAES)
```

```
An object of class "AnnotatedDataFrame"
featureNames: 1007_s_at, 1053_at, ..., AFFX-TrpmX-M_at (54675 total)
varLabels and varMetadata description: none
```

6

The closure twist

1. subsetting operations are common
 - (a) focus on a gene set
 - (b) focus on samples sharing a phenotype
 - (c) filter genes with little variation in expression
2. complex operations need inputs with predictable structure
3. therefore preserve structure across subsetting operations

7

8

```

> afxsubRMAES[1, 1:2]
ExpressionSet (storageMode: lockedEnvironment)
assayData: 1 features, 2 samples
element names: exprs
phenodata
sampleNames: AFX_1_A1.CEL, AFX_1_A2.CEL
varLabels and varMetadata description:
  site: from cel
  samp: rna src/mixture code
  repl: replicate
  pcttBrain: pct of mixture from Ambion brain
featureData
featureNames: 1007_s_at
fvarLabels and fvarMetadata description: none
experimentData: use 'experimentData(object)'
pubMedIds: 16964226
Annotation: hgu133plus2

```

9

```

> experimentData(afxsubRMAES[1, 1:2])
Experiment data
  Experiment name: Shippy R
  Laboratory: GE Healthcare, 7700 S. River Pkwy., Suite #2603, Tempe, Ari
  Contact information:
  Title: Using RNA sample titrations to assess microarray platform perfor
  URL:
  PMIDs: 16964226

```

Abstract: A 158 word abstract is available. Use 'abstract' method.

10

Upshots

1. we have dramatically cut down the number of features and samples
2. this was accomplished with $Y = X[G, S]$
3. operations that work on “full” arrays still work on Y
4. metadata on the retained features are still available for Y

11

Principles

1. the container class is derived from Biobase::eSet
2. at least two major numerical/factor data components
 - (a) Assay data ($G \times N$)
 - i. attributes of features in feature data ($G \times q$)
 - (b) ‘pheno’ data (sample-level, $N \times R$)
 - i. attributes of variables varMetadata ($R \times s$)
3. textual prose metadata experimentData (MIAME schema, abstract)
4. platform token (annotation)
5. IF ANY OF THESE ARE ABSENT BE SURE YOU CAN JUSTIFY IT AS THE DEFICIT WILL BE PROPAGATED TO PEOPLE WHO COULD HAVE BENEFITED HAD IT BEEN REMEDIED IT

12

More general containers

1. racExSet: rare allele count + expression

```
> library(GGtools)
> data(chr20GGdem)
> chr20GGdem

racExSet instance (SNP rare allele count + expression)
rare allele count assayData:
Storage mode: lockedEnvironment
featureNames: rs4814683, rs6076506, ..., rs6062370, rs6090120 (117417 total)
Dimensions:
  racs
Features 117417
Samples 58

expression assayData
Storage mode: lockedEnvironment
featureNames: 1007_s_at, 1053_at, ..., AFFX-r2-p1-cre-3_at, AFFX-r2-p1-cre-5_at (8793 total)
Dimensions:
  exprs
Features 8793
Samples 58

phenodata
An object of class "AnnotatedDataFrame"
rownames: NA06985, NA06993, ..., NA12892 (58 total)
varLabels and varMetadata description:
  sample: hapmap id

Experiment data
Experiment name: Cheung VG
```

13

Questions and answers

1. questions:

- You are interested in the hypothesis that a given SNP may be an eQTL for a given gene. How can you test it?
- You wish to check a certain interval around a gene for SNPs for which rare allele copy number is associated with expression. How?
- You wish to search the entire genome for SNPs that may be eQTL for any genes in a given pathway. How?

2. answers:

- genotype the individuals on whom transcript profile arrays are available
- create a racExSet or allied structure
- use GGtools snpScreen, twSnpScreen or similar function

15

Laboratory: Department of Pediatrics, University of Pennsylvania, Philadelphia, Pennsylvania 19104, USA.
Contact information:
Title: Mapping determinants of human gene expression by regional and genome-wide association.
URL:
PMIDs: 16251966

Abstract: A 130 word abstract is available. Use 'abstract' method.

Annotation [1] "hgfocuss"

14

How do these analyses work?

1. the class structure is defined

```
> getClass("racExSet")

Slots:
Name:      racAssays      rarebase      SNPalleles
Class:     AssayData      character     character
Name:      assayData     phenodata     featureData
Class:     AssayData     AnnotatedDataFrame AnnotatedDataFrame
Name:      experimentData annotation     classVersion__
Class:     MIAME         character     character     Versions

Extends:
Class "eSet", directly
Class "VersionedBiobase", by class "eSet", distance 2
Class "Versioned", by class "eSet", distance 3
```

2. methods are defined for specific signatures (combinations of classes)

- validity checking on class instances yields guarantees that input to method has appropriate structure
- method can construct an instance of a result class
- chain of methods has high reliability of succeeding at each step

16

Another species of container: gene sets

```
> library(GSEABase)
> fl <- system.file("extdata", "Broad.xml", package = "GSEABase")
> gs2 <- getBroadSets(fl)[[1]]
> gs2
setName: chr5q23
geneIds: ZNF474, OCDC100, ..., LOC728586 (total: 86)
geneIdType: Symbol
collectionType: Broad
bcCategory: c1 (Positional)
bcSubCategory: NA
details: use 'details(object)'
> getClass(class(gs2))
Slots:
```

```
Name:      geneIdType      geneIds      setName
Class: GeneIdentifierType character      ScalarCharacter
Name:      setIdentifier  shortDescription longDescription
Class:     ScalarCharacter ScalarCharacter  ScalarCharacter
Name:      organism      pubMedIds      urls
Class:     ScalarCharacter character        character
Name:      contributor   version         creationDate
```

17

working with gene sets

```
> geneIds(gs2)[1:5]
[1] "ZNF474" "OCDC100" "ANKRD43" "NRG2" "LOC391828"
> gs3 = gs2
> geneIdType(gs3) = AnnotationIdentifier("hgu133plus2")
> geneIds(gs3)[1:5]
[1] "1556907_at" "1554606_at" "226449_at" "230238_at" "206879_s_at"
> gs4 = gs2
> geneIdType(gs4) = AnnotationIdentifier("illuminaHumanv2")
> geneIds(gs4)[1:5]
[1] "ILMN_26423" "ILMN_29165" "ILMN_14177" "ILMN_566" "ILMN_3332"
> gs3
setName: chr5q23
geneIds: 1556907_at, 1554606_at, ..., 226660_at (total: 143)
geneIdType: Annotation (hgu133plus2)
collectionType: Broad
bcCategory: c1 (Positional)
bcSubCategory: NA
details: use 'details(object)'
> gs4
setName: chr5q23
geneIds: ILMN_26423, ILMN_29165, ..., ILMN_11282 (total: 55)
```

19

```
Class:      character
Name:      collectionType
Class:     CollectionType
Known SubClasses: "GeneColorSet"
```

```
Class:      character
Name:      collectionType
Class:     CollectionType
Known SubClasses: "GeneColorSet"
geneIdType: Annotation (illuminaHumanv2)
collectionType: Broad
bcCategory: c1 (Positional)
bcSubCategory: NA
details: use 'details(object)'
```

18

20

containers for graphs and networks

```
> library(keggorth)
> data(KOgraph)
> KOgraph
A graphNEL graph with directed edges
Number of Nodes = 283
Number of Edges = 282
> nodes(KOgraph)[1:4]
[1] "KO.June07root"
[3] "Carbohydrate Metabolism"
"Metabolism"
"Glycolysis / Gluconeogenesis"
> adj(KOgraph, "Metabolism")
$Metabolism
[1] "Carbohydrate Metabolism"
[2] "Biosynthesis of Secondary Metabolites"
[3] "Xenobiotics Biodegradation and Metabolism"
[4] "Energy Metabolism"
[5] "Lipid Metabolism"
[6] "Nucleotide Metabolism"
[7] "Amino Acid Metabolism"
[8] "Metabolism of Other Amino Acids"
[9] "Glycan Biosynthesis and Metabolism"
[10] "Biosynthesis of Polyketides and Nonribosomal Peptides"
[11] "Metabolism of Cofactors and Vitamins"
```

21

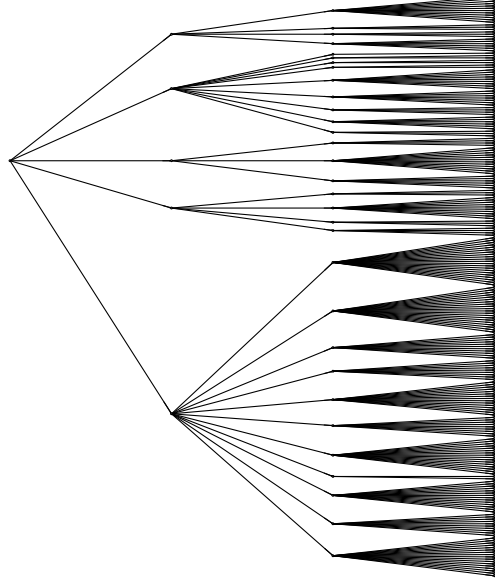
Visualizing

```
> library(Rgraphviz)
> plot(KOgraph)
```

22

Visualizing a pathway

```
> library(pathRender)
> data(pancrCaIni)
> plot(pancrCaIni,
+      nodeAttrs=pwayRenderAttrs(pancrCaIni))
```



23

24

Enhancing a pathway diagram with observational data

```

> library(ALL)
> data(ALL)
> library(hgu95av2.db)
> rmap = revmap(hgu95av2SYMBOL)
> rALL = reduceES( ALL, nodes(pancrCaIni), rmap, collapseFun=mean )
> par(mfrow=c(2,2))
> plotExGraph(pancrCaIni, rALL, 1, main="BCR/ABL 1")
> text(120,780,"BCR/ABL 1", cex=1.3)
> plotExGraph(pancrCaIni, rALL, 3, main="BCR/ABL 2")
> text(120,780,"BCR/ABL 2", cex=1.3)
> plotExGraph(pancrCaIni, rALL, 3, main="NEG 1")
> text(120,780,"NEG 1", cex=1.3)
> plotExGraph(pancrCaIni, rALL, 4, main="ALL1/AF4 1")
> text(120,780,"ALL1/AF4 1", cex=1.3)
> par(mfrow=c(1,1))

```

26

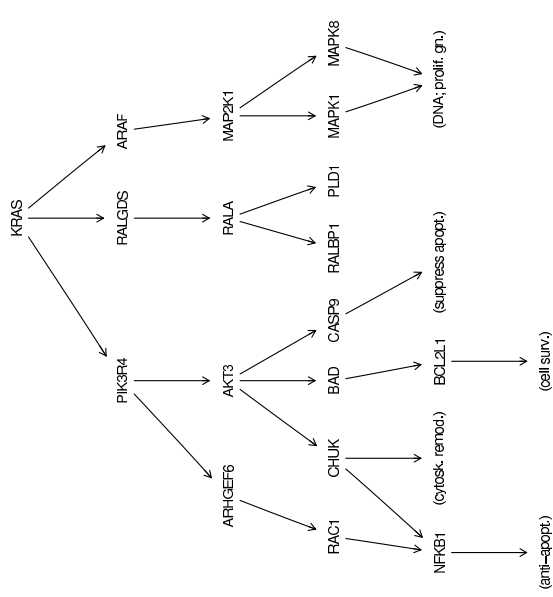
creating a pathway graph

```

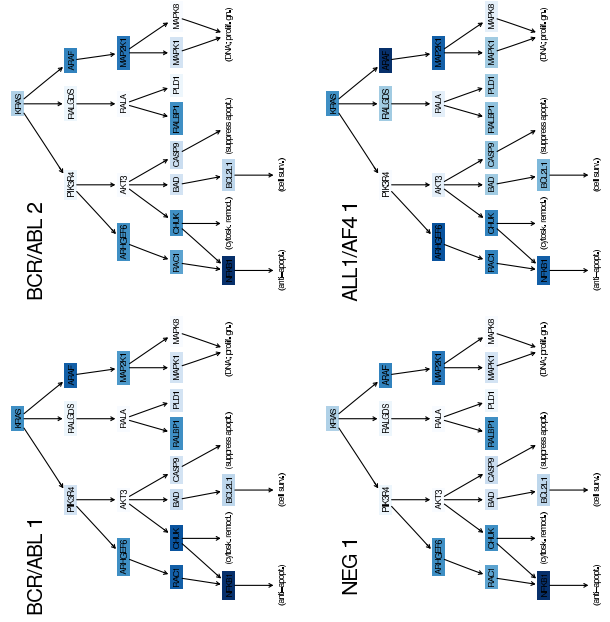
<?xml version="1.0"?>
<gxl>
  <graph id="colorectalFrag" edgemode="directed">
    <node id="DOC"/>
    <node id="CASP3"/>
    <node id="CASP9"/>
    <node id="KRAS"/>
    <node id="PI3K"/>
    <node id="RAF"/>
    <node id="RALGDS"/>
    <node id="Rac"/>
    <node id="JNK"/>
    <edge id="d1" from="CASP3" to="DOC"/>
    <edge id="d2" from="DOC" to="CASP9"/>
    <edge id="d3" from="KRAS" to="PI3K"/>
    <edge id="d4" from="KRAS" to="RAF"/>
    <edge id="d4" from="KRAS" to="RALGDS"/>
    <edge id="d4" from="Rac" to="Rac"/>
    <edge id="d4" from="Rac" to="JNK"/>
  </graph>
</gxl>

```

28



25



27

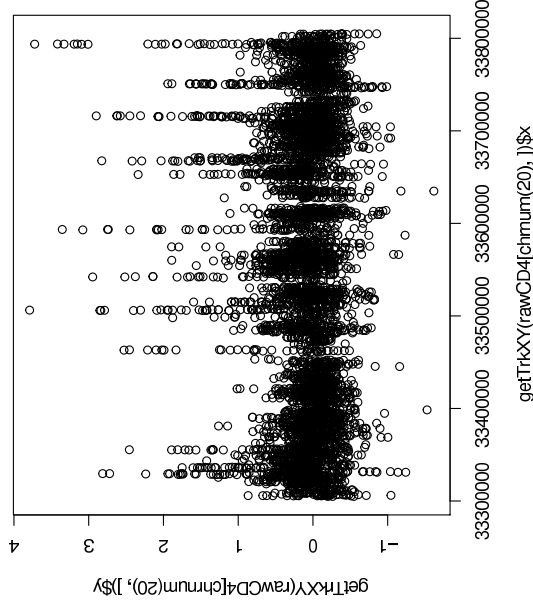
hg18 track data

1. UCSC genome browser widely used
2. annotation 'tracks' are simple files consisting of coordinates and values

```
> library(encoDnaseI)
> data(rawCD4)
> rawCD4

hg18track (storageMode: lockedEnvironment)
assayData: 382713 features, 1 samples
element names: dataVals
phenoData
  sampleNames: 1
varLabels and varMetadata description: none
featureData
  featureNames: 1, 2, ..., 382713 (382713 total)
  fvarLabels and fvarMetadata description:
  bin: given bin
  chrom: chr...
  chromStart: numeric origin
  chromEnd: numeric close
  experimentData: use 'experimentData(object)'
  pubMedIds: 16791207
Annotation:
```

33



35

```
> plot(getTrkXY(rawCD4[chrnum(20), ]))
```

34

Biostring containers

```
> library(Biostrings)
> d <- DNASTring("TTGAAAA-CTC-N")
> length(d)
[1] 13
> alphabet(d)
[1] "A" "C" "G" "T" "M" "R" "W" "S" "Y" "K" "V" "H"
> views(d, c(1, 2, 3, 4), c(6, 7, 8, 9))
Views on a 13-letter DNASTring subject
subject: TTGAAAA-CTC-N
views:
      start end width
[1] 1 6 6 [TTGAAA]
[2] 2 7 6 [TGAAAA]
```

36

```
[3] 3 8 6 [GAAAA-]
[4] 4 9 6 [AAAA-C]
> mm = matchPattern("AA", d)
> start(mm)
[1] 4 5 6
```

37

A last type of container related to analysis

```
> library(MLInterfaces)
> library(golubEsets)
> data(Golub_Train)
> ldl = MLearn(ALL.AML ~ ., Golub_Train[400:800, ], ldlI, xvalSpec("LOO"))
> ldl
> class(ldl)
> confuMat(ldl)
```

39

```
> pom = readFASTA("pombe_chr02_region.fasta")
> names(pom)
NULL
> names(pom[[1]])
[1] "desc" "seq"
> poms = DNAString(pom[[1]]$seq)
> poms
200000-letter "DNAString" instance
seq: TGATAAAAATCCTGATCTTTGGCCCTAATTGTATTC...ACACGAAAAAATCATGTTTAAGGTCGTGACT
> aam = matchPattern("AA", poms)
> class(aam)
[1] "BStringViews"
attr(,"package")
[1] "Biostrings"
```

38

The classifierOutput container

```
> getClass(class(ldl))
Slots:
Name: testOutcomes testPredictions testScores trainOutcomes
Class: factor factor factor ANY factor
Name: trainPredictions trainScores fsHistory
Class: factor factor ANY list RObject
Name: call embeddedCV
Class: call logical ANY
```

40

Summary

1. interactive computing: objects should have concise and suggestive representations
2. 'show' methods control this
3. getClass shows the anatomy of a class; class definitions can employ other class definitions
4. setValidity for a class imposes unbreakable conditions on class structure
 - an object will not be created if these are not met
5. array data, gene sets, networks, tracks, strings, classifiers all occupy containers with specified structure
6. strings, matrices, vectors, lists can have arbitrary sizes and have only very rudimentary introspection
7. contrast annotation-identified ExpressionSet with a matrix or list
8. formal containers for downstream results facilitate linkage of workflow components (e.g., classifierOutput)