MultiDataSet

Carlos Ruiz

Computational Genomic Seminars
Barcelona

Thursday, June 1, 2017

MultiDataSet

What `MultiDataSet` does

- It encapsulates data from multiple datasets with common samples
- It performs subsetting operations on multiple datasets

MultiDataSet

What `MultiDataSet` does

- It encapsulates data from multiple datasets with common samples
- It performs subsetting operations on multiple datasets

What `MultiDataSet` does **not** do

- Perform data analysis

Outline

1. Design

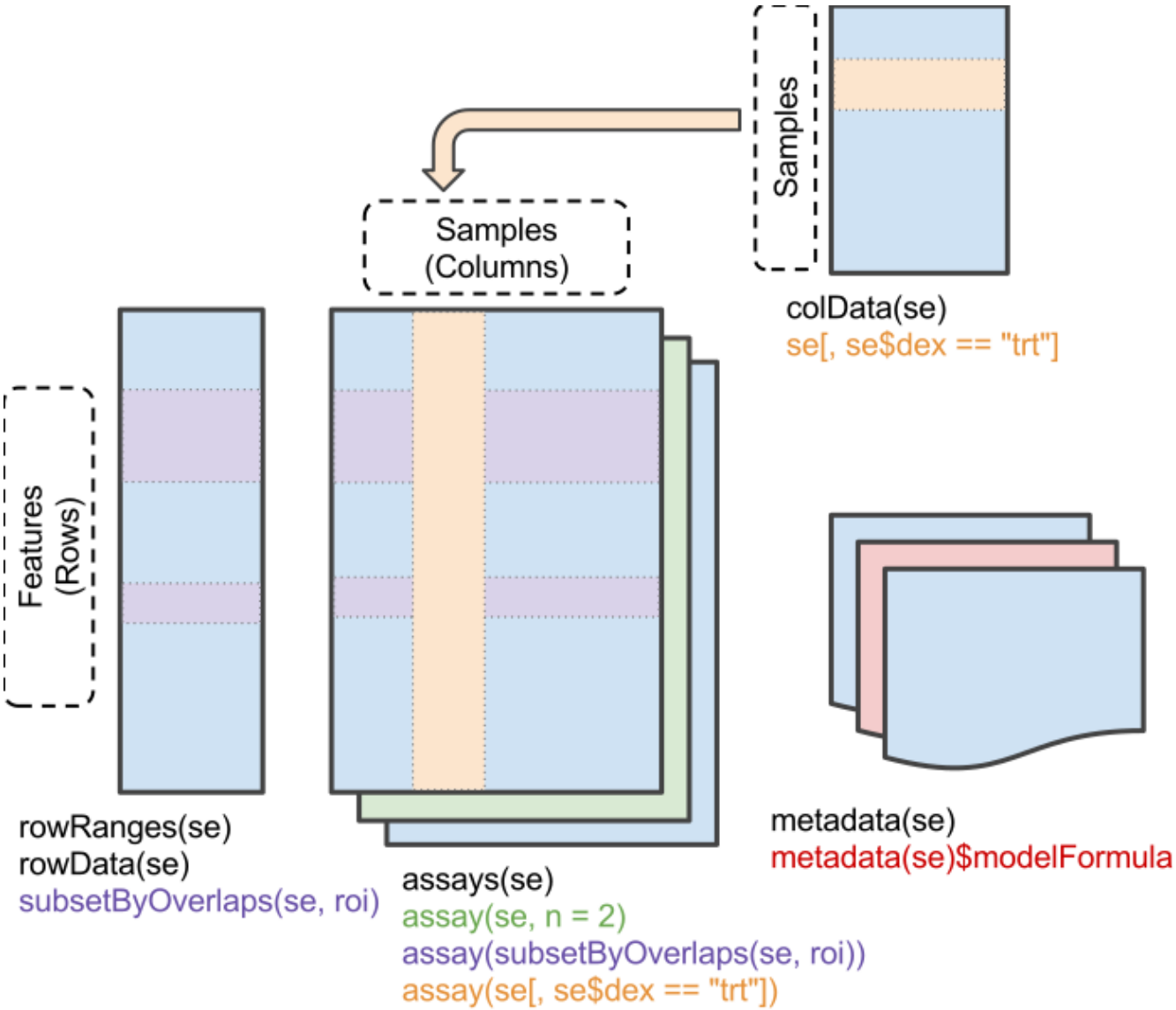
2. Real Applications

   - GTEX
   - TCGA
   - Data Integration

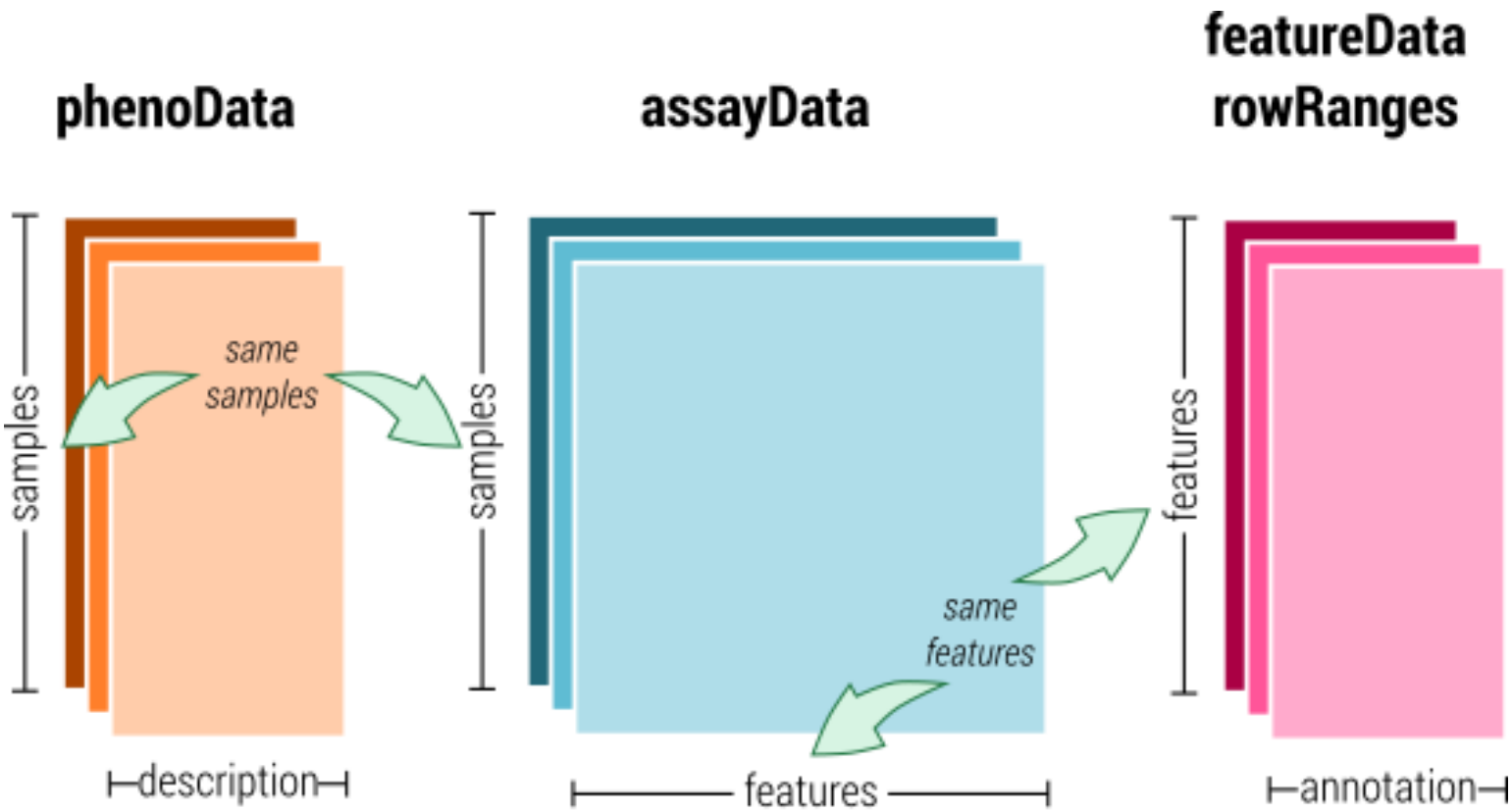
3. Profiling

Design

Structure


Samples
Samples
(Columns)
colData(se)
se[, se$dex == "trt"]
Features
(Rows)
rowRanges(se)
rowData(se)
subsetByOverlaps(se, roi)
assays(se)
assay(se, n = 2)
assay(subsetByOverlaps(se, roi))
assay(se[, se$dex == "trt"])
metadata(se)
metadata(se)$modelFormula

Structure


phenoData
assayData
featureData
rowRanges
samples
same
samples
description
samples
features
same
features
features
annotation

Add sets to MultiDataSet

1. Create Empty `MultiDataset`

2. Add Sets

- `eSet`
- `SummarizedExperiment`
- `RangedSummarizedExperiment`

Add sets to MultiDataSet

1. Create Empty `MultiDataset`

```
multi <- createMultiDataSet()
```

2. Add Sets

- `eSet`
- `SummarizedExperiment`
- `RangedSummarizedExperiment`

Add sets to MultiDataSet

1. Create Empty `MultiDataset`

2. Add Sets

- `eSet`

```
multi <- add_eset(multi, eset, "Expression")
```

- `SummarizedExperiment`

```
multi <- add_se(multi, se, "Expression")
```

- `RangedSummarizedExperiment`

```
multi <- add_rse(multi, rse, "Expression")
```

Typical workflow

Add set

```
multi <- createMultiDataSet()
multi <- add_eset(multi, eset, "Expression")
```

Perform operations

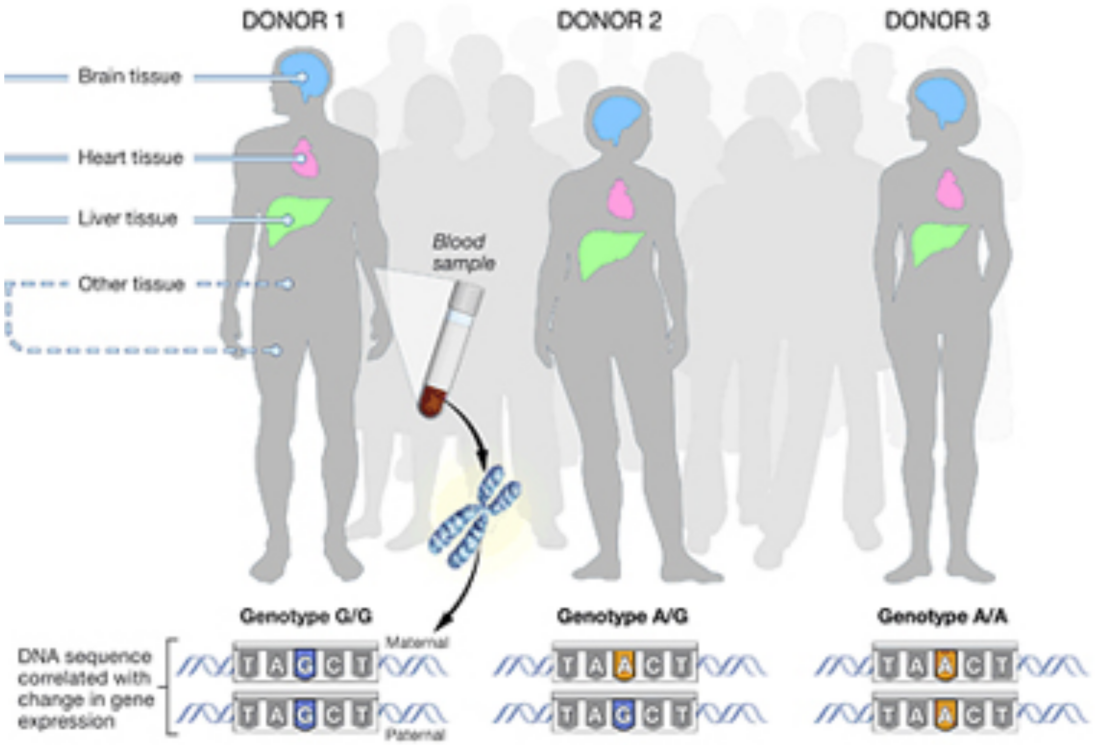
```
multi <- multi[, c("A", "B", "C")]
```

Retrieve set

```
finalset <- multi[["Expression"]]
```

Real applications

GTEX


DONOR 1
DONOR 2
DONOR 3
Brain tissue
Heart tissue
Liver tissue
Other tissue
Blood sample
Genotype G/G
Genotype A/G
Genotype A/A
DNA sequence correlated with change in gene expression
Maternal
Paternal

GTEX

Scientific question

- Does the age modifies gene expression in all tissues?

GTEX

Scientific question

- Does the age modifies gene expression in all tissues?

Problem

- Each tissue has different samples.

Requirement

- We need complete cases in the analysis.

GTEX

Starting point

- `ExpressionSets` with the data of each tissue.
- Objects named with the tissue source: blood, brain, lung...

Create MultiDataset and add sets

```
multi <- createMultiDataSet()
multi <- add_eset(multi, blood, "Blood")
multi <- add_eset(multi, brain, "Brain")
multi <- add_eset(multi, lung, "Lung")
```

GTEX

Standard code

```
commonNames <- Reduce(intersect,
list(sampleNames(blood), sampleNames(brain),
sampleNames(lung)))
blood[, commonNames]
brain[, commonNames]
lung[, commonNames]
```

GTEX

Standard code

```
commonNames <- Reduce(intersect,
list(sampleNames(blood), sampleNames(brain),
sampleNames(lung)))
blood[, commonNames]
brain[, commonNames]
lung[, commonNames]
```

MultiDataSet code

```
multi <- commonSamples(multi)
```

GTEX

Scientific question

- Does the age modifies gene expression **in females** in all tissues?

Requirements

- We need to select samples that are females.
- We need complete cases in the analysis.

GTEX

Starting point

- `ExpressionSets` with the data of each tissue.
- Objects named with the origin source: blood, brain, lung...
- `ExpressionSets`' pData contains a column called sex (male/female).

GTEX

Standard code

```
blood <- blood[, blood$sex == "female"]
brain <- brain[, blood$sex == "female"]
lung <- lung[, blood$sex == "female"]
commonNames <- Reduce(intersect,
list(sampleNames(blood), sampleNames(brain),
sampleNames(lung)))
blood[, commonNames]
brain[, commonNames]
lung[, commonNames]
```

GTEX

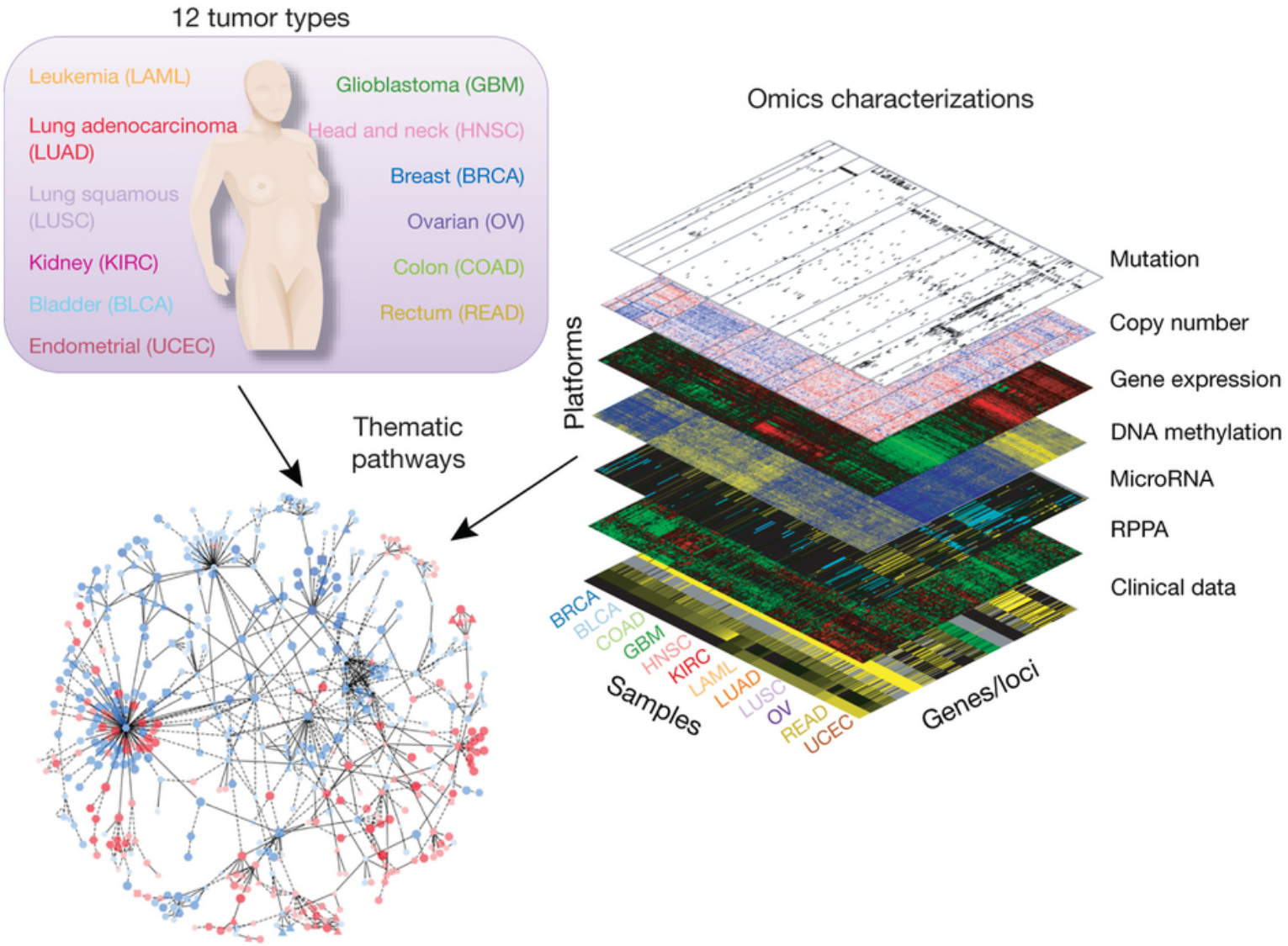
Standard code

```
blood <- blood[, blood$sex == "female"]
brain <- brain[, blood$sex == "female"]
lung <- lung[, blood$sex == "female"]
commonNames <- Reduce(intersect,
list(sampleNames(blood), sampleNames(brain),
sampleNames(lung)))
blood[, commonNames]
brain[, commonNames]
lung[, commonNames]
```

MultiDataSet code

```
multi <- subset(multi, , sex == "female")
multi <- commonSamples(multi)
```

TCGA


12 tumor types
Leukemia (LAML)
Glioblastoma (GBM)
Lung adenocarcinoma (LUAD)
Head and neck (HNSC)
Breast (BRCA)
Lung squamous (LUSC)
Ovarian (OV)
Kidney (KIRC)
Colon (COAD)
Bladder (BLCA)
Rectum (READ)
Endometrial (UCEC)
Thematic pathways
Omics characterizations
Platforms
Mutation
Copy number
Gene expression
DNA methylation
MicroRNA
RPPA
Clinical data
BRCA
BLCA
COAD
GBM
HNSC
KIRC
LAML
LUAD
LUSC
OV
READ
UCEC
Samples
Genes/loci

TCGA

Scientific question

- Does SNP variation in target region modify gene expression and methylation in that region in lung cancer?

TCGA

Scientific question

- Does SNP variation in target region modify gene expression and methylation in that region in lung cancer?

Problem

- Features in the sets are very different have different names and lengths.
- Each dataset is stored in different classes.

Requirement

- For each set, we need to select those features in our target region.

TCGA

Starting point

- `SummarizedExperiment` with the expression data: exprs
- `SnpSet` with the SNP data: snps
- `GenomicMethylationSet` with the methylation data: meth
- `GenomicRanges` with the target region: gr

TCGA

Create MultiDataset and add sets

- exprs: `SummarizedExperiment`
- snps: `eSet`, `SnpSet`
- meth: `RangedSummarizedExperiment`, `GenomicMethylationSet`

```
multi <- createMultiDataSet()
multi <- add_se(multi, exprs, "exprs")
multi <- add_eset(multi, snps, "snps")
multi <- add_rse(multi, meth, "meth")
```

TCGA

Standard code

```
gr_exprs <- makeGRangesFromDataFrame(rowData(exprs))
gr_exprsfilt <- subsetByOverlaps(gr_exprs, gr)
exprs_filt <- exprs[names(gr_exprsfilt),]
gr_snps <- makeGRangesFromDataFrame(fData(snps))
gr_snpfilt <- subsetByOverlaps(gr_snps, gr)
snps_filt <- snps[names(gr_snpfilt),]
meth_filt <- subsetByOverlaps(meth, gr)
```

TCGA

Standard code

```
gr_exprs <- makeGRangesFromDataFrame(rowData(exprs))
gr_exprsfilt <- subsetByOverlaps(gr_exprs, gr)
exprs_filt <- exprs[names(gr_exprsfilt),]
gr_snps <- makeGRangesFromDataFrame(fData(snps))
gr_snpfilt <- subsetByOverlaps(gr_snps, gr)
snps_filt <- snps[names(gr_snpfilt),]
meth_filt <- subsetByOverlaps(meth, gr)
```

MultiDataSet code

```
multi <- multi[, , gr]
```

TCGA

Scientific question

- Is the expression of a gene correlated with its methylation and SNPs in breast cancer?

TCGA

Scientific question

- Is the expression of a gene correlated with its methylation and SNPs in breast cancer?

Problem

- SNPs and CpGs can be mapped to different genes.
- Each dataset is stored in different classes.

Requirement

- For each set, we need to select those features mapped to our target gene.

TCGA

Starting point

- `SummarizedExperiment` with the expression data: exprs
- `SnpSet` with the SNP data: snps
- `GenomicMethylationSet` with the methylation data: meth
- exprs, snps and meth have the column geneNames in their feature data
- In meth, geneNames contains all the genes mapped to a feature separated by semicolons (e.g. BRCA;HER2)

TCGA

Standard code

```
exprs[grepl("BRCA", rowData(exprs)$geneNames), ]
snps[grepl("BRCA", fData(snps)$geneNames), ]
meth[grepl("BRCA", rowData(meth)$geneNames), ]
```

TCGA

Standard code

```
exprs[grepl("BRCA", rowData(exprs)$geneNames), ]
snps[grepl("BRCA", fData(snps)$geneNames), ]
meth[grepl("BRCA", rowData(meth)$geneNames), ]
```

MultiDataSet code

```
subset(multi, grepl("BRCA", geneNames))
```

TCGA

We use more complex logical filters:

- Select all features to BRCA **OR** HER2 genes

```
subset(multi, grepl("BRCA", geneNames) |
grepl("HER2", geneNames))
```

- Select all features to BRCA **AND** HER2 genes

```
subset(multi, grepl("BRCA", geneNames) &
grepl("HER2", geneNames))
```

Data Integration

Scientific question

- Are there specific profiles of methylation, gene expression and miRNAs for the different clinical subtypes of breast cancer?

Data Integration

Scientific question

- Are there specific profiles of methylation, gene expression and miRNAs for the different clinical subtypes of breast cancer?

Problem

- Apply Multi Coinertia Analysis, implemented in omicade4 in function mcia.
- Input of mcia has a specific format.
  - List of matrices
  - All matrices must have the same samples in the same order

Data Integration

Starting point

- `SummarizedExperiment` with the expression data: exprs
- `ExpressionSet` with the miRNAs data: miRNAs
- `GenomicMethylationSet` with the methylation data: meth

Data Integration

Create MultiDataset and add sets

```
multi <- createMultiDataSet()
multi <- add_se(multi, exprs, "exprs")
multi <- add_eset(multi, miRNAs, "mirna")
multi <- add_rse(multi, meth, "meth")
```

Data Integration

Standard code

```
commonNames <- Reduce(intersect,
list(colnames(exprs), sampleNames(miRNAs), colnames(meth)))
input <- list(exprs = assay(exprs),
mirna = exprs(miRNAs),
meth = betas(meth))
input <- lapply(input, function(x) x[, commonNames])
mcia_res <- mcia(input)
```

Data Integration

Standard code

```
commonNames <- Reduce(intersect,
list(colnames(exprs), sampleNames(miRNAs), colnames(meth)))
input <- list(exprs = assay(exprs),
mirna = exprs(miRNAs),
meth = betas(meth))
input <- lapply(input, function(x) x[, commonNames])
mcia_res <- mcia(input)
```

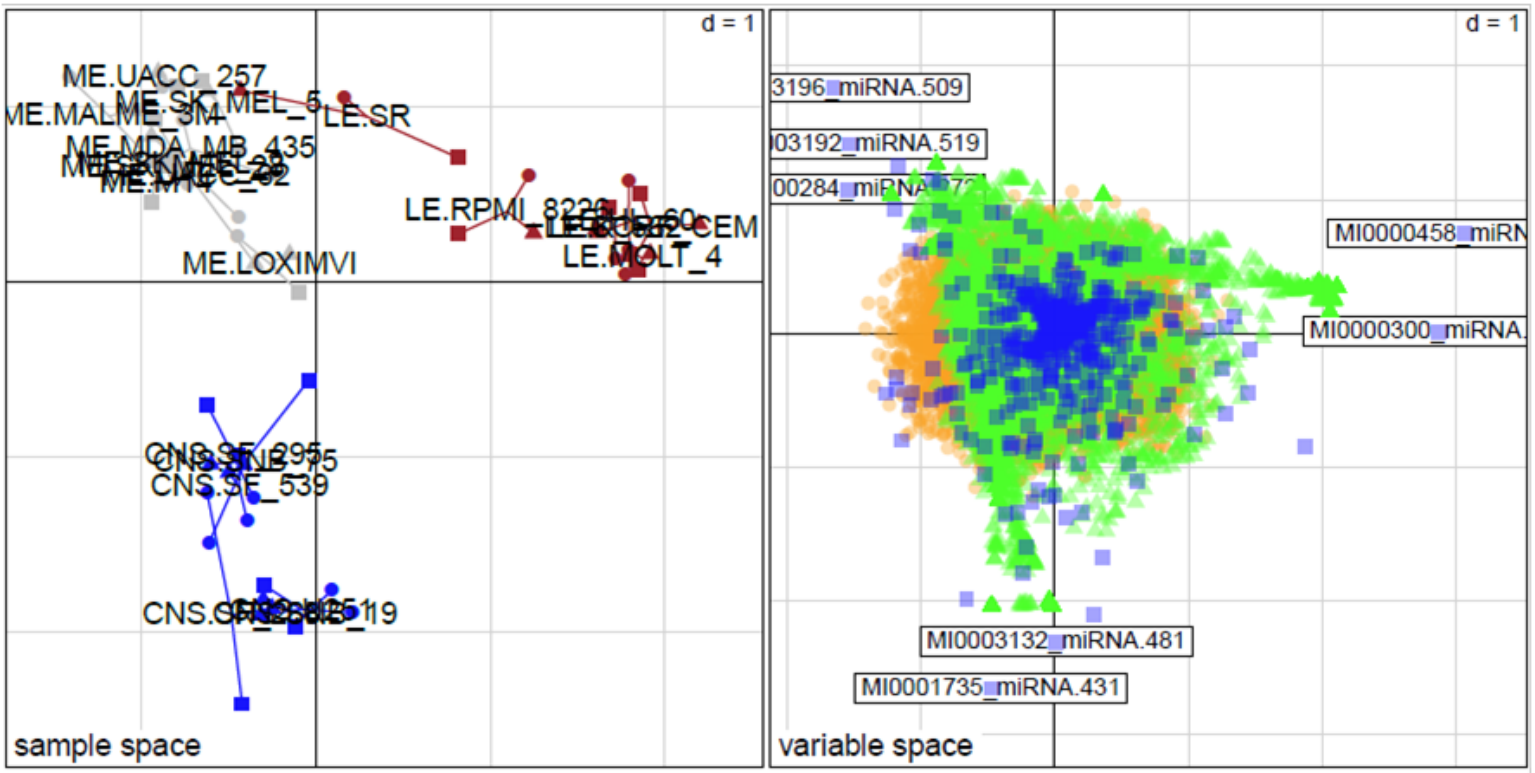
MultiDataSet code

```
mcia_res <- w_mcia(multi)
```

Data Integration

MultiDataSet code

```
mcia_res <- w_mcia(multi)
plot(mcia_res)
```



Data Integration

Object standardization eases development of wrappers:

```
as.list.MultiDataSet <- function(x) {
    ll <- lapply(names(x), function(dtype) {
        elm <- assayDataElementNames(assayData(x)[[dtype]])[1
        assayDataElement(assayData(x)[[dtype]], elm)
    })
    names(ll) <- names(x)
    return(ll)
}
```

Profiling

Profiling

`MultiDataSet` has a similar efficiency than other R packages designed to manage multiple omic datasets.

Time spent in different operations (s):

	List	MDS	MAE
Create Object	-	13.25	6.46
GRanges filter	0.72	1.83	0.92
Common Samples	5.80	2.52	5.47

*MDS is `MultiDataSet`, MAE is `MultiAssayExperiment`

Summary

MultiDataSet can do much more

Subsetting operations

- Select samples by name
- Combine selection of samples and features

MultiDataSet can do much more

Subsetting operations

- Select samples by name
- Combine selection of samples and features

Make easy using other methods

- iClusterPlus' wrapper
- Develop wrappers for other methods
- Develop adding functions for non-expert users
- Develop adding functions for standardizing input in integration functions

MultiDataSet can do much more

Under development features

- Functions to add more complex objects
- Wrapper to use Generalized Canonical Correlation Analysis
- Download data from public repositories in MultiDataSet
- Increase data management efficiency

Take-home message

- MultiDataSet is a class to encapsulate data from multiple datasets.
- It facilitates data management.
- It can work with most data types.
- It eases applying mcia and iClusterPlus to your data.
- It can save you a lot of time and effort!

Availability

MultiDataSet is available at Bioconductor since version 3.3 (R>= 3.3)

`https://bioconductor.org/packages/release/bioc/html/MultiDataSet.html`

More information can be found in the paper:

Hernandez-Ferrer C, Ruiz-Arenas C, Beltran-Gomila A and Gonzalez J (2017). "MultiDataSet: an R package for encapsulating multiple data sets with application to omic data integration." BMC bioinformatics, 18(1), pp. 36

If you have any question, doubt or suggestion to improve the package, contact us at

- carlos.ruiz@isglobal.org
- carles.hernandez@isglobal.org
- juanr.gonzalez@isglobal.org.

Acknowledgements

This work has been partly funded by the Spanish Ministry of Economy and Competitiveness (MTM2015-68140-R). CH-F was supported by a grant from European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no308333 – the HELIX project. CR-A was supported by a FI fellowship from Catalan Government (#016FI_B 00272). The funding body had no role in the design of the study, the collection, analysis, and interpretation of data or in writing the manuscript.