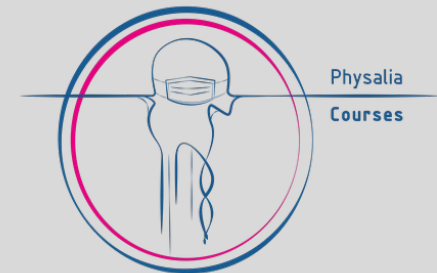


Quantification of gene expression

Epigenomics Data Analysis

Jacques Serizay

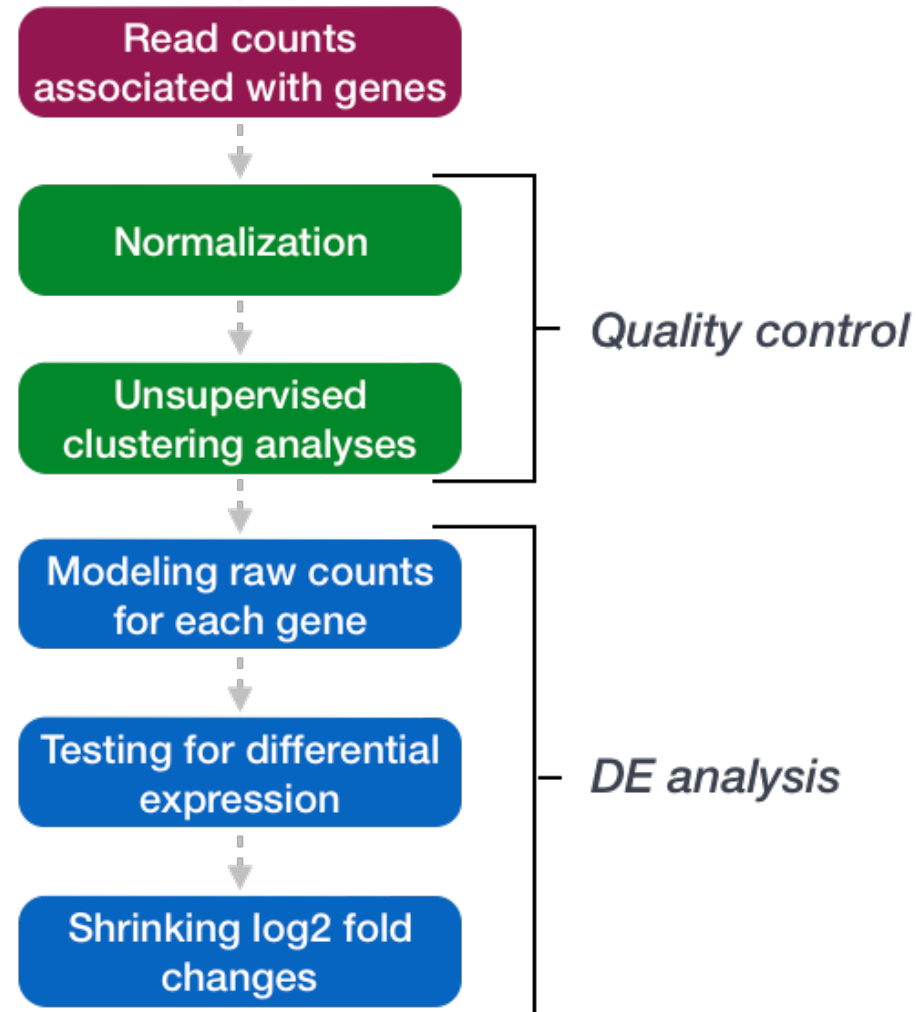
Physalia 2025



RNA-seq downstream analysis

- ✓ Get .bcl files 
- ✓ Create fastq files  Or **bcl2fastq**
- ✓ QC: remove/trim low quality reads E.g. **cutadapt**
- ✓ Align fastq to BAM E.g. **bowtie2**, **STAR**
- ✓ Filter duplicates, artifacts, ... E.g. **samtools**
- ✓ Generate tracks E.g. **deepTools**
- Assay-specific downstream analysis **Transcript abundance quantification**
Differential gene expression analysis
Gene ontology over-representation analysis

DESeq2 analysis workflow

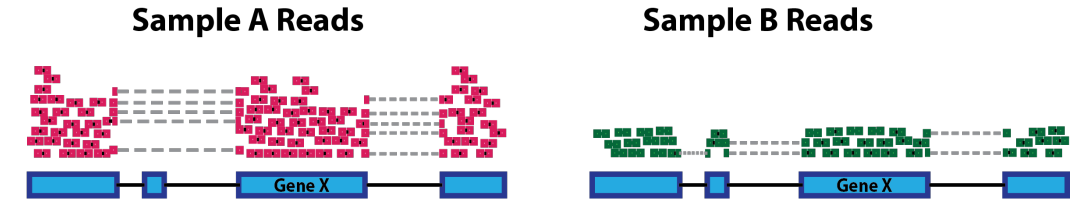


https://hbctraining.github.io/DGE_workshop/

Estimating gene/transcript abundance in RNA-seq

Important points when comparing gene abundance for different datasets/genes are:

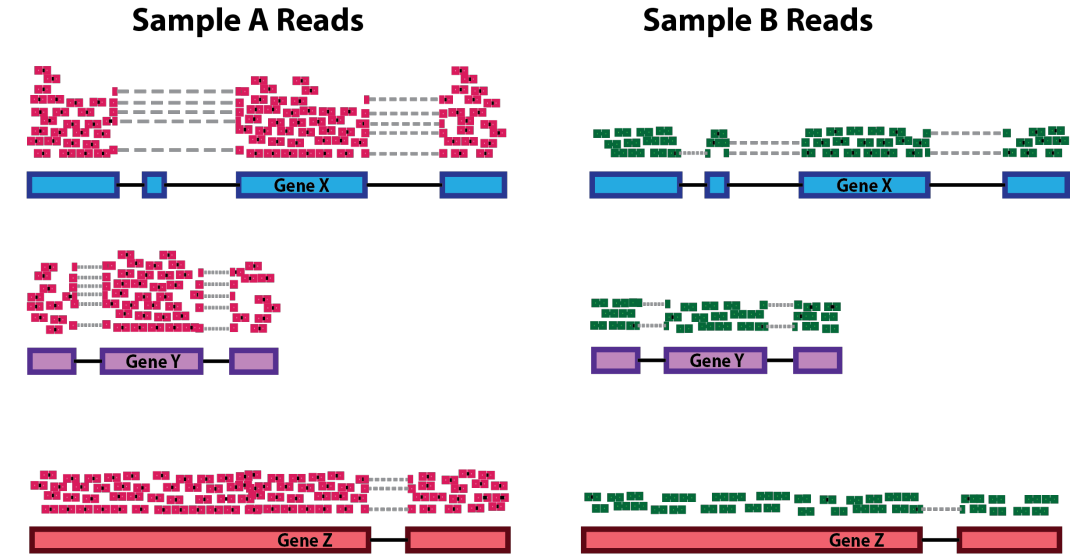
- Sequencing depth



Estimating gene/transcript abundance in RNA-seq

Important points when comparing gene abundance for different datasets/genes are:

- Sequencing depth
- Gene length

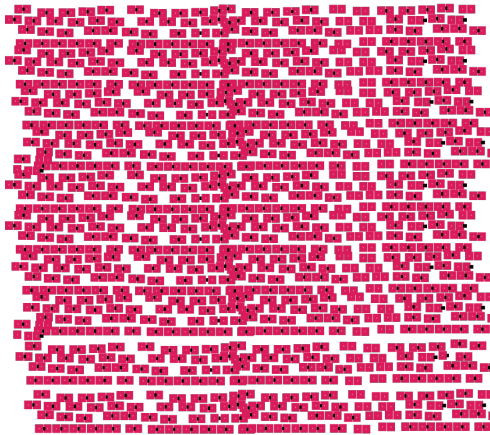
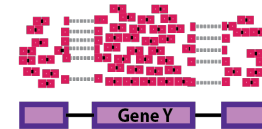
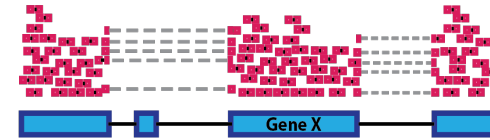


Estimating gene/transcript abundance in RNA-seq

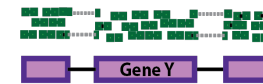
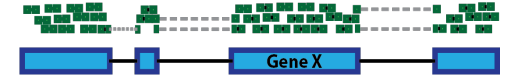
Important points when comparing gene abundance for different datasets/genes are:

- Sequencing depth
- Gene length
- RNA composition

Sample A Reads



Sample B Reads



https://hbctraining.github.io/DGE_workshop/

Estimating gene/transcript abundance in RNA-seq

- featureCounts can be used to count reads overlapping a set of gene annotations.

in bash

```
featureCounts \  
  -g gene_name \  
  -s 2 \  
  -p -B \  
  -T 16 \  
  -o data/counts/RNAseq_counts.tsv \  
  -a data/counts/hg38_Gencodev41.gtf \  
  data/mapping/RNA_ctl_1^hg38^filtered.bam \  
  data/mapping/RNA_ctl_2^hg38^filtered.bam \  
  data/mapping/RNA_ctl_3^hg38^filtered.bam \  
  data/mapping/RNA_foxj1_1^hg38^filtered.bam \  
  data/mapping/RNA_foxj1_2^hg38^filtered.bam
```

Estimating gene/transcript abundance in RNA-seq

- featureCounts can be used to count reads overlapping a set of gene annotations.
- Rsubread is an R package which wraps featureCounts function in R.

in bash

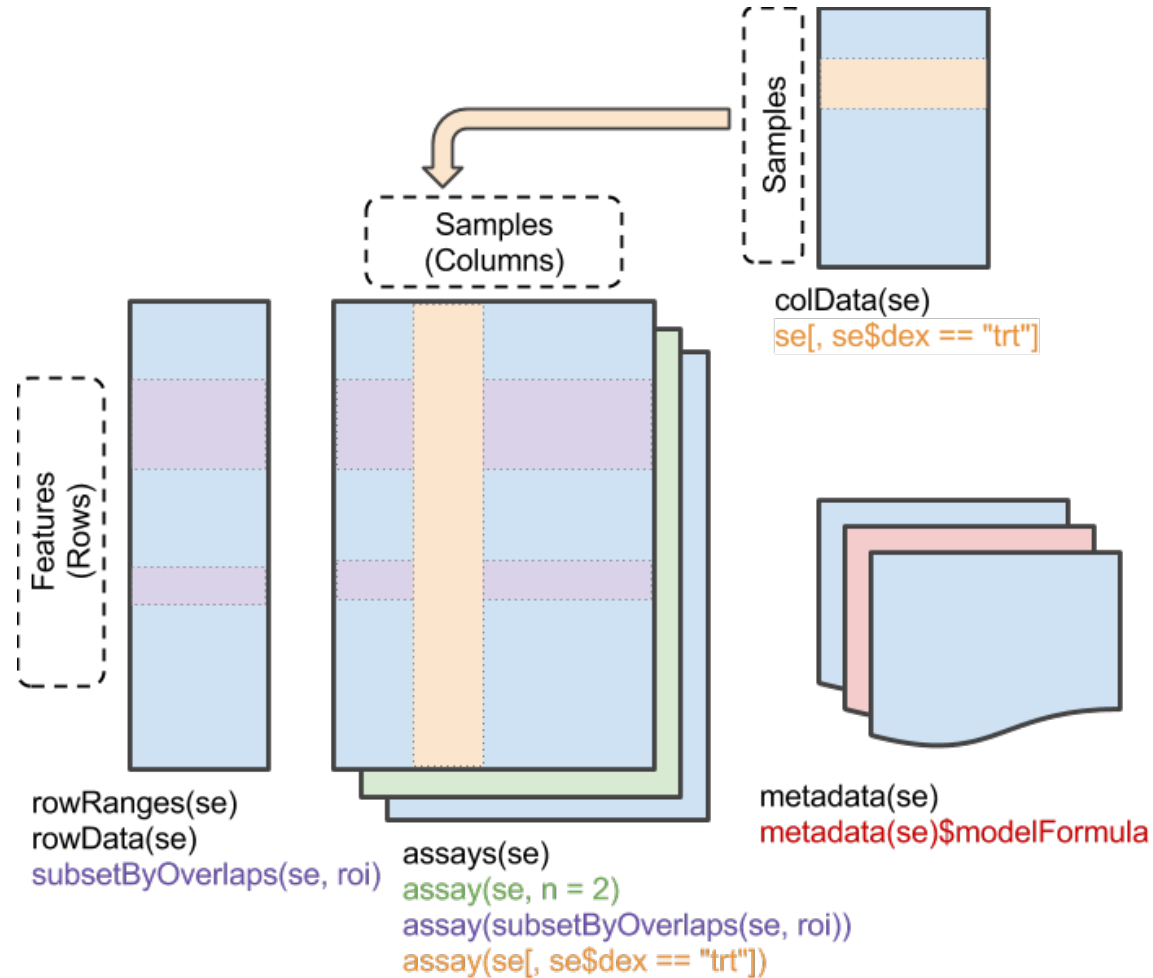
```
featureCounts \  
  -g gene_name \  
  -s 2 \  
  -p -B \  
  -T 16 \  
  -o data/counts/RNAseq_counts.tsv \  
  -a data/counts/hg38_Gencodev41.gtf \  
  data/mapping/RNA_ctl_1^hg38^filtered.bam \  
  data/mapping/RNA_ctl_2^hg38^filtered.bam \  
  data/mapping/RNA_ctl_3^hg38^filtered.bam \  
  data/mapping/RNA_foxj1_1^hg38^filtered.bam \  
  data/mapping/RNA_foxj1_2^hg38^filtered.bam
```

in R

```
cnts <- Rsubread::featureCounts(  
  files = c('...bam', '...bam'),  
  annot.ext = 'annots.gtf',  
  isGTFAnnotationFile = TRUE,  
  GTF.featureType = 'sequence_feature',  
  GTF.attrType = 'id',  
  isPairedEnd = TRUE,  
  nthreads = 16  
)
```


DESeq2 analysis workflow

- DESeq2 is based on a SummarizedExperiment object.



SummarizedExperiment

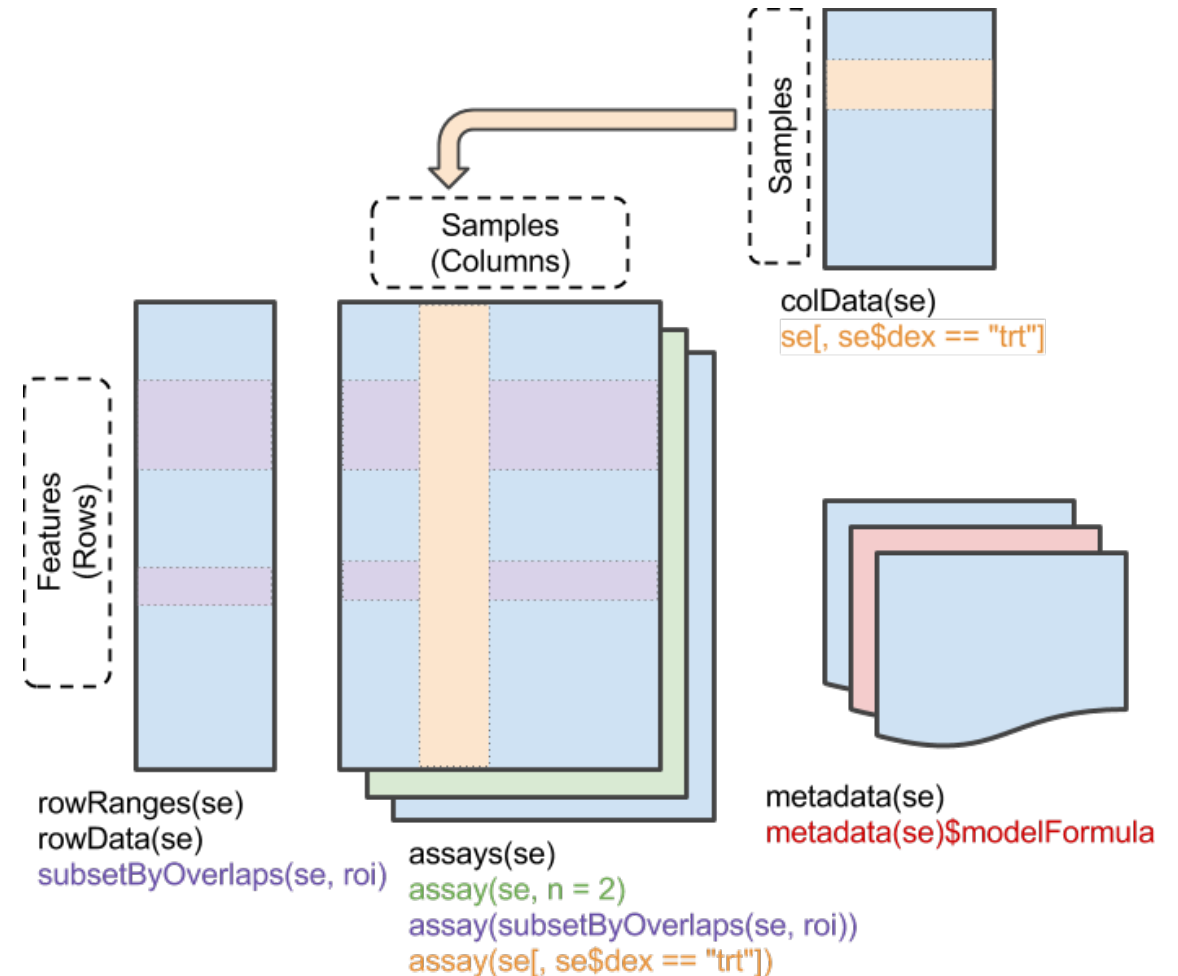
Published: 29 January 2015

Orchestrating high-throughput genomic analysis with Bioconductor

Wolfgang Huber ✉, Vincent J Carey, Robert Gentleman, Simon Anders, Marc Carlson, Benilton S Carvalho, Hector Corrada Bravo, Sean Davis, Laurent Gatto, Thomas Girke, Raphael Gottardo, Florian Hahne, Kasper D Hansen, Rafael A Irizarry, Michael Lawrence, Michael I Love, James MacDonald, Valerie Obenchain, Andrzej K Oleś, Hervé Pagès, Alejandro Reyes, Paul Shannon, Gordon K Smyth, Dan Tenenbaum, Levi Waldron & Martin Morgan –Show fewer authors

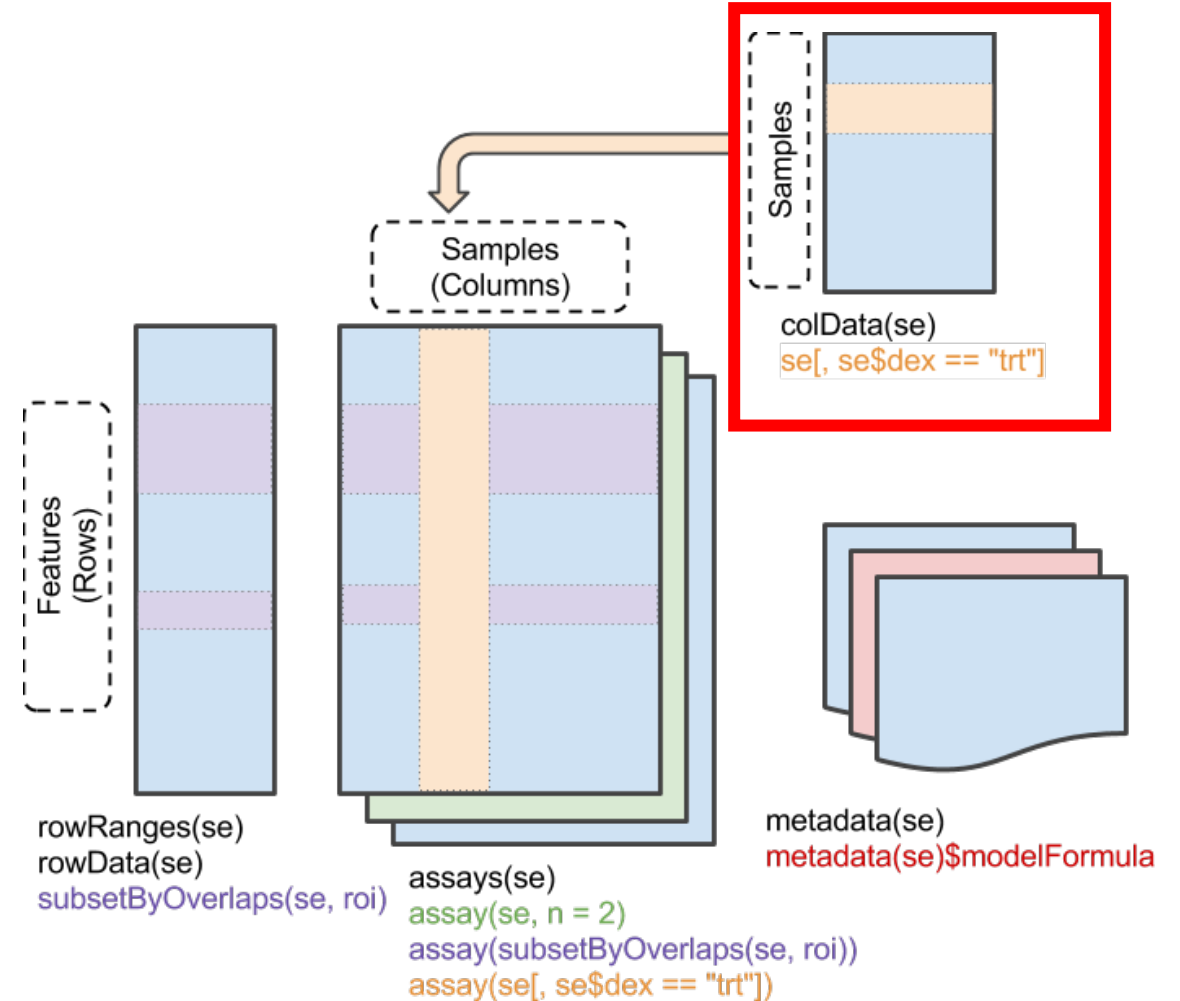
Nature Methods **12**, 115–121(2015) | [Cite this article](#)

Huber et al., *Nat. Meth.* 2015



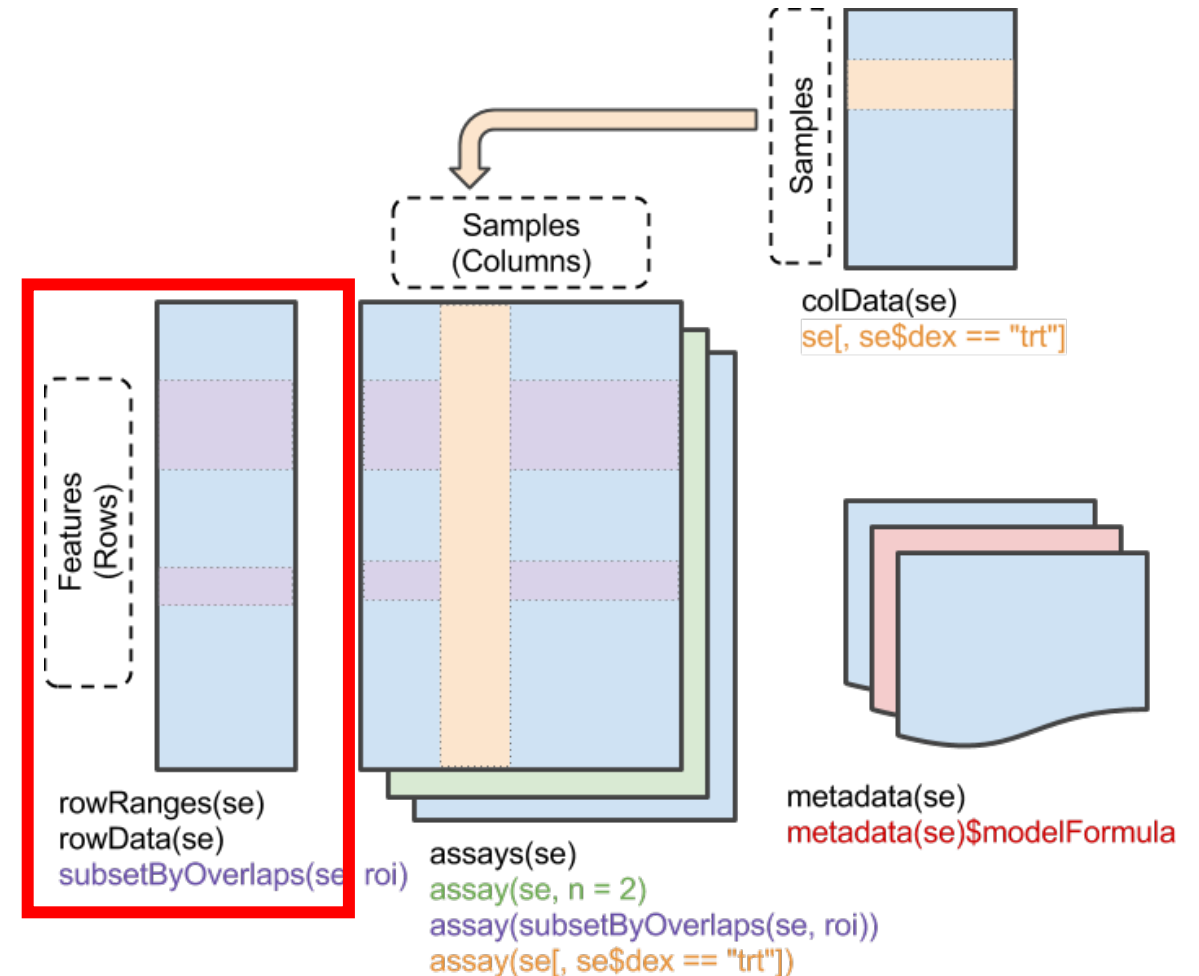
SummarizedExperiment

- `colData()`: Annotations on each column, as a DataFrame.
 - E.g., description of each sample



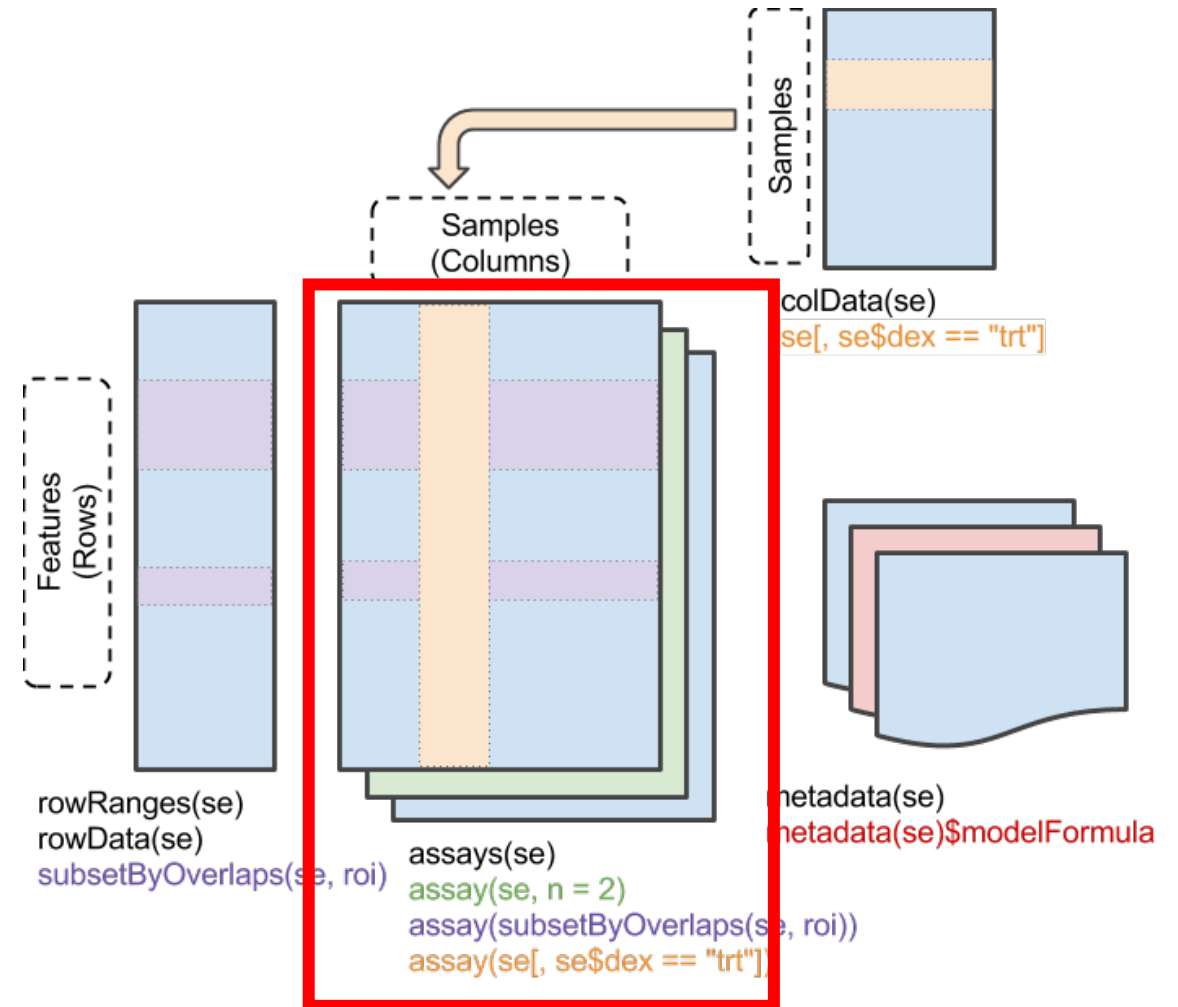
SummarizedExperiment

- `colData()`: Annotations on each column, as a DataFrame.
 - E.g., description of each sample
- `rowData/rowRanges()`: Annotations on each row.
 - E.g., coordinates of gene / exons / peaks in transcripts / etc.



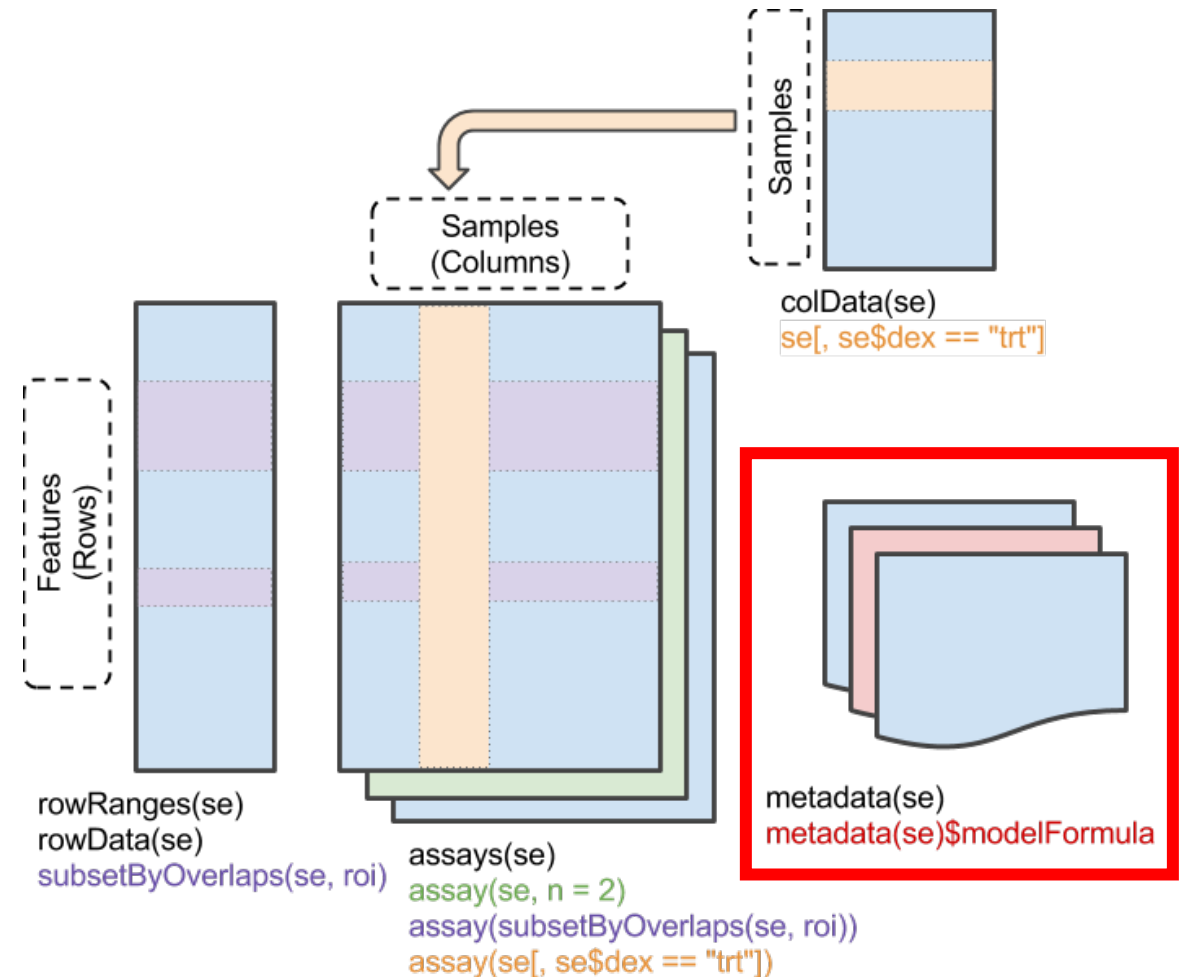
SummarizedExperiment

- `colData()`: Annotations on each column, as a DataFrame.
 - E.g., description of each sample
- `rowData/rowRanges()`: Annotations on each row.
 - E.g., coordinates of gene / exons / peaks in transcripts / etc.
- `assay()`, `assays()`: A matrix-like or list of matrix-like objects of identical dimension
 - rows: refer to **rowRanges**: genes, genomic coordinates, etc.
 - columns: refer to **colData**: samples, cells, etc.
 - Implements `dim()`, `dimnames()` and 2-dimensional `[ , ]`
 - **Can be several assays!!!**



SummarizedExperiment

- `colData()`: Annotations on each column, as a DataFrame.
 - E.g., description of each sample
- `rowData/rowRanges()`: Annotations on each row.
 - E.g., coordinates of gene / exons / peaks in transcripts / etc.
- `assay()`, `assays()`: A matrix-like or list of matrix-like objects of identical dimension
 - rows: refer to **rowRanges**: genes, genomic coordinates, etc.
 - columns: refer to **colData**: samples, cells, etc.
 - Implements `dim()`, `dimnames()` and 2-dimensional [,]
 - **Can be several assays!!!**
- `metadata()`: List of unstructured metadata describing the overall content of the object.



SummarizedExperiment

SummarizedExperiment is a key data structure in **Bioconductor**, reused by or modified in:

- clusterExperiment
- SingleCellExperiment
- LoomExperiment
- MsExperiment
- TreeSummarizedExperiment
- SpatialExperiment
- And 100s more...

Depends On Me

[AffixCan](#), [airway](#), [alabaster.se](#), [AllelicImbalance](#), [ASpediaFI](#), [atena](#), [bambu](#), [BDMMAcorrect](#), [benchmarkfdrData2019](#), [BiocSlearn](#), [BioPlex](#), [BiSeq](#), [bnbc](#), [bodymapRat](#), [bsseq](#), [CAGEfightR](#), [celaref](#), [celldex](#), [clusterExperiment](#), [compartmap](#), [CoreGx](#), [coseq](#), [csaw](#), [CSSQ](#), [curatedAdipoChIP](#), [curatedAdipoRNA](#), [curatedMetagenomicData](#), [DaMiRseq](#), [deco](#), [deepSNV](#), [DeMixT](#), [DESeq2](#), [DEXSeq](#), [DiffBind](#), [diffcoexp](#), [diffHic](#), [divergence](#), [DMCFB](#), [DMCHMM](#), [ENmix](#), [EnrichmentBrowser](#), [epigenomix](#), [evalumeR](#), [EventPointer](#), [exomePeak2](#), [ExperimentSubset](#), [ExpressionAtlas](#), [extraChIPs](#), [FEAST](#), [fission](#), [FRASER](#), [GenomicAlignments](#), [GenomicFiles](#), [GenomicSuperSignature](#), [GRmetrics](#), [GSEABenchmarkR](#), [HDCytoData](#), [HelloRanges](#), [hermes](#), [HiCDOC](#), [HighlyReplicatedRNASeq](#), [hipathia](#), [HMP16SDData](#), [IntAD](#), [InteractionSet](#), [InterEst](#), [ISEE](#), [ISEEhex](#), [ISEEhub](#), [ISLET](#), [isomiRs](#), [ivygapSE](#), [lefser](#), [lipidr](#), [LoomExperiment](#), [Macarron](#), [made4](#), [MatrixQCvis](#), [MBASED](#), [MetaGxOvarian](#), [MetaGxPancreas](#), [methrix](#), [methylPipe](#), [MethylSeqData](#), [MetNet](#), [mia](#), [miaViz](#), [MicrobiomeBenchmarkData](#), [microbiomeDataSets](#), [microRNAome](#), [minfi](#), [miRmine](#), [moanin](#), [MouseGastrulationData](#), [MouseThymusAgeing](#), [mpa](#), [MultiAssayExperiment](#), [NADfinder](#), [NBAMSeq](#), [NewWave](#), [ObMITI](#), [OUTRIDER](#), [padma](#), [parathyroidSE](#), [PDATK](#), [phenomis](#), [PhIPData](#), [profileplyr](#), [qmtools](#), [qsvaR](#), [recount](#), [recount3](#), [RegEnrich](#), [REMP](#), [restfulSE](#), [restfulSEData](#), [ROCpAI](#), [rqt](#), [runibic](#), [sampleClassifierData](#), [Scale4C](#), [scAnnotatR](#), [scGPS](#), [scMultiome](#), [scone](#), [screenCounter](#), [scTreeViz](#), [SDAMS](#), [SeqGate](#), [SEtools](#), [SGSeq](#), [signatureSearch](#), [SingleCellExperiment](#), [singleCellITK](#), [SingleR](#), [soGGi](#), [spatialDmexsim](#), [spqn](#), [spqnData](#), [sRACIPE](#), [ssPATHS](#), [stageR](#), [SummarizedBenchmark](#), [survtype](#), [TENxIO](#), [tidySummarizedExperiment](#), [timecoursedata](#), [TissueEnrich](#), [TNBC.CMS](#), [TREG](#), [tuberculosis](#), [UMI4Cats](#), [VanillaICE](#), [VariantAnnotation](#), [VariantExperiment](#), [velociraptor](#), [weitrax](#), [yamss](#), [zinbwave](#)

Imports Me

[ADAM](#), [ADImpute](#), [aggregateBioVar](#), [airpart](#), [alabaster.spatial](#), [ALDEx2](#), [alpine](#), [ANCOMBC](#), [animalcules](#), [anota2seq](#), [APalyzer](#), [apeglm](#), [APL](#), [appreci8R](#), [ASICS](#), [ASURAT](#), [ATACseqTFFA](#), [AUCcell](#), [autonomics](#), [awst](#), [barcodetrackR](#), [BASICS](#), [BASICStan](#), [batchelor](#), [BayesSpace](#), [bayNorm](#), [BBCAnalyzer](#), [beer](#), [benchdamic](#), [bigPint](#), [BiocOncoTK](#), [BioNERO](#), [biosigner](#), [biotmle](#), [biolvizBase](#), [biscuiteer](#), [BiSeq](#), [blacksheepR](#), [BloodCancerMultiOmics2017](#), [BloodGen3Module](#), [brgedata](#), [BRGenomics](#), [BUMHMM](#), [BUScorrect](#), [BUSseq](#), [CAEN](#), [CAGER](#), [CATALYST](#), [CBEA](#), [cBioPortalData](#), [ccfindR](#), [celda](#), [CellID](#), [CellMixS](#), [CellTrails](#), [censcyt](#), [Cepo](#), [CeTF](#), [CHETAH](#), [ChIPpeakAnno](#), [ChromScape](#), [chromVAR](#), [CiteFuse](#), [CLLMethylation](#), [clustifyr](#), [cmapR](#), [CNVfilter](#), [CNVRanger](#), [CoGAPS](#), [comapr](#), [combi](#), [condiments](#), [consensusDE](#), [consICA](#), [CopyNumberPlots](#), [corral](#), [COSMIC.67](#), [countsimQC](#), [CTSV](#), [curatedTCGAData](#), [cydar](#), [CyTOFpower](#), [cytofQC](#), [cytoKernel](#), [cytomapper](#), [cytoviewer](#), [DAMEfinder](#), [dasper](#), [debCAM](#), [debrowser](#), [decompTumor2Sig](#), [DEFormats](#), [DEGreport](#), [DELocal](#), [deltaCaptureC](#), [DEP](#), [DEScan2](#), [DESpace](#), [destiny](#), [DEWSeq](#), [diffcyt](#), [DifferentialRegulation](#), [diffFUTR](#), [Dino](#), [DiscoRhythm](#), [distinct](#), [dittoSeq](#), [DMRcate](#), [DominoEffect](#), [doppelgangR](#), [doseR](#), [DropletUtils](#), [Dune](#), [easierData](#), [easyRNASeq](#), [eisaR](#), [ELMER](#), [emtdata](#), [ensemblVEP](#), [epialleleR](#), [epigraHMM](#), [EpiMix](#), [epimutations](#), [epistack](#), [epivizrData](#), [erma](#), [escape](#), [escherR](#), [EWCE](#), [ExpHunterSuite](#), [FCBF](#), [fcScan](#), [FeatSeekR](#), [FieldEffectCirc](#), [FindIT2](#), [fishpond](#), [FLAMES](#), [FlowSorted.Blood.EPIC](#), [FlowSorted.CordBloodCombined.450k](#), [fluentGenomics](#), [FuseSOM](#), [GARS](#), [gCrisprTools](#), [gemma.R](#), [GeneTonic](#), [genomicInstability](#), [GeoTcgaData](#), [getDEE2](#), [ggbio](#), [ggspavis](#), [Glimma](#), [glmGamPoi](#), [glmSparseNet](#), [GNET2](#), [GRaNIE](#), [GreyListChIP](#), [gscreend](#), [GSE13015](#), [GSVA](#), [gwasurvivr](#), [GWENA](#), [HiContacts](#), [HMP2Data](#), [HTSeqGenie](#), [HumanTranscriptomeCompendium](#), [hummingbird](#), [iasva](#), [icetate](#), [ideal](#), [IFAA](#), [IgGeneUsage](#), [IHWpaper](#), [ILoReg](#), [lmcRtools](#), [infercnv](#), [INSPeCT](#), [InterMineR](#), [IntOMICS](#), [ISEEU](#), [IsoformSwitchAnalyzer](#), [LACE](#), [LineagePulse](#), [lineagespot](#), [lionessR](#), [lisaClust](#), [MADSEQ](#), [MAI](#), [mariner](#), [marr](#), [MAST](#), [mastR](#), [mbkmeans](#), [MBQN](#), [mCSEA](#), [MEAL](#), [MEAT](#), [MEB](#), [MerfishData](#), [MetaboAnnotation](#), [metabolomicsWorkbenchR](#), [MetaGxBreast](#), [MetaNeighbor](#), [MetaScope](#), [metaseqR2](#), [MethReg](#), [MethylAid](#), [methylscaper](#), [methylumi](#), [miaSim](#), [MicrobiotaProcess](#), [midashHLA](#), [miloR](#), [MinimumDistance](#), [miRLAB](#), [miRSM](#), [missMethyl](#), [MLInterfaces](#), [MLSeq](#), [monaLisa](#), [MoonlightR](#), [motifbreakR](#), [motifmatchr](#), [MPRAnalyze](#), [MsExperiment](#), [MsFeatures](#), [msgbsR](#), [MSPrep](#), [msgrob2](#), [MuData](#), [MultiDataSet](#), [multiOmicsViz](#), [mumosa](#), [muscat](#), [musicatK](#), [MWASTools](#), [NanoMethViz](#), [Nebulosa](#), [NetActivity](#), [netSmooth](#), [nnSVG](#), [NoRCE](#), [NormalyzerDE](#), [NxtIRFcore](#), [oligoClasses](#), [omicRexposome](#), [OmicsLonDA](#), [omicsPrint](#), [omicsViewer](#), [oncomix](#), [ontoProc](#), [ORFIk](#), [OVESEG](#), [PAIRADISE](#), [pairedGSEA](#), [pairkat](#), [pcaExplorer](#), [peco](#), [PharmacoGx](#), [phemd](#), [phenopath](#), [PhosR](#), [pipeComp](#), [plantthunter](#), [pmp](#), [POMA](#), [POWSC](#), [proActiv](#), [proDA](#), [psichomics](#), [pulsedSilac](#), [PureCN](#), [QFeatures](#), [qsmooth](#), [quantiseqr](#), [R453Plus1Toolbox](#), [RadioGx](#), [RaggedExperiment](#), [RareVariantVis](#), [RcisTarget](#), [receptLoss](#), [regionReport](#), [regsplice](#), [rgsepd](#), [rifi](#), [rifiComparative](#), [Rmmquant](#), [RNAAgeCalc](#), [RNAsense](#), [RnaSeqSampleSize](#), [roar](#), [RoIDE](#), [ropls](#), [rScudo](#), [RTCGAToolbox](#), [RTN](#), [saturn](#), [SBGNview](#), [SC3](#), [SCArray](#), [SCArray.sat](#), [scater](#), [scBFA](#), [scCB2](#), [scDbiFinder](#), [scDD](#), [scDDboost](#), [scds](#), [scFeatures](#), [schOT](#), [scmap](#), [scMerge](#), [scMET](#), [scmeth](#), [SCnorm](#), [scoreInvHap](#), [scp](#), [scPipe](#), [scran](#), [scReClassify](#), [scRepertoire](#), [scRNAseq](#), [scruff](#), [scry](#), [scTensor](#), [scTGF](#), [scuttle](#), [scvIR](#), [sechm](#), [segmenter](#), [seqCAT](#), [sesame](#), [SGCP](#), [sigFeature](#), [signifinder](#), [SigPack](#), [SimBu](#), [simpleSeg](#), [SingleCellMultiModal](#), [singscore](#), [SingscoreAMLMutations](#), [slalom](#), [slingshot](#), [snapcount](#), [SNPhood](#), [Spaniel](#), [spaSim](#), [SpatialCPie](#), [spatialDE](#), [SpatialExperiment](#), [SpatialFeatureExperiment](#), [spatialHeatmap](#), [spatialLBD](#), [spatzle](#), [SPIAT](#), [spicyR](#), [splatter](#), [SpliceWiz](#), [SplicingFactory](#), [SpotClean](#), [srnadiff](#), [sSNAPPY](#), [standR](#), [stJoincount](#), [struct](#), [StructuralVariantAnnotation](#), [supersigs](#), [SVMDO](#), [switchde](#), [systemPipeR](#), [systemPipeTools](#), [TabulaMurisSenisData](#), [TBSignatureProfiler](#), [TCGAbiolinks](#), [TCGAutils](#), [TCGAWorkflow](#), [TCGAWorkflowData](#), [TCseq](#), [TEKRABber](#), [tenXplore](#), [tidybulk](#), [tidySingleCellExperiment](#), [TOAST](#), [tomoda](#), [ToxicoGx](#), [tradeSeq](#), [TrajectoryUtils](#), [transformGamPoi](#), [traviz](#), [TreeSummarizedExperiment](#), [Trendy](#), [tricycle](#), [TSCAN](#), [tsr](#), [TTMap](#), [TVTB](#), [tximeta](#), [UCell](#), [VAExprs](#), [VariantFiltering](#), [VDJdive](#), [vidger](#), [Voyager](#), [wpm](#), [xcms](#), [zellkonverter](#), [zFPKM](#)



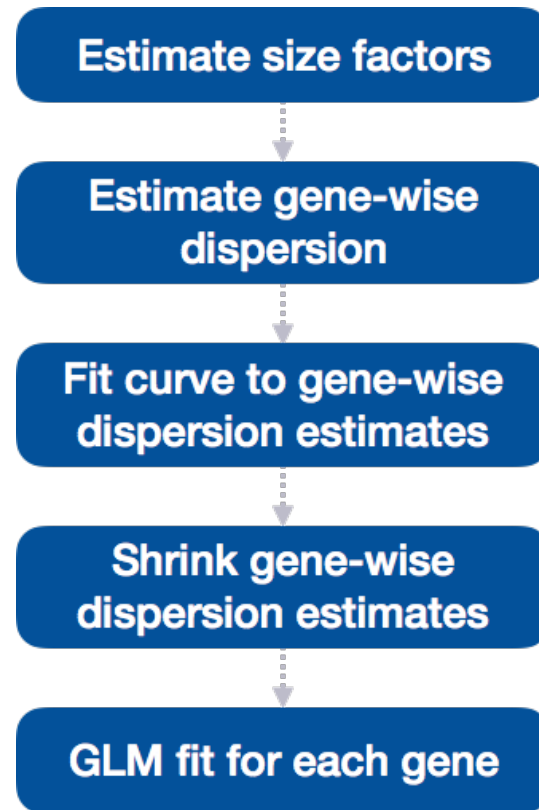
DESeq2 analysis workflow

- DESeq2 is based on a SummarizedExperiment object.
- A **formula** needs to be specified

```
> dds <- DESeqDataSet(counts, design = ~ timepoint)
```


DESeq2 analysis workflow

- Differential expression analysis is as simple as `DESeq()`



	sample2	WT	2	No drug	
	sample3	KO	1	Drug	
• DE	sample4	KO	2	drug	
	sample5	WT	1	drug	
• Af	sample6	WT	2	drug	

- Differential expression analysis is as simple as DESeq ()

```
> dds <- DESeqDataSet(counts, design = ~ Condition + Treatment)
> dds <- DESeq(dds)
estimating size factors
estimating dispersions
gene-wise dispersion estimates
mean-dispersion relationship
final dispersion estimates
fitting model and testing
> dds
class: DESeqDataSet
dim: 6513 10
metadata(1): version
assays(4): counts mu H cooks
rownames: NULL
rowData names(37): summit peakID ... deviance maxCooks
colnames: NULL
colData names(5): sample timepoint replicate bam sizeFactor
```

DESeq2 analysis workflow

4) Extract results

- ❖ We can use `DESeq2::results()`
- ❖ DESeq2 extracts results for pairwise comparison between 2 conditions

```
> contrasts
      15_v_00  30_v_00  45_v_00  60_v_00  30_v_15  45_v_15  60_v_15  45_v_30  60_v_30  60_v_45
[1,] "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint"
[2,] "15"        "30"        "45"        "60"        "30"        "45"        "60"        "45"        "60"        "60"
[3,] "00"        "00"        "00"        "00"        "15"        "15"        "15"        "30"        "30"        "45"
```

DESeq2 analysis workflow

```
> contrasts
  15_v_00  30_v_00  45_v_00  60_v_00  30_v_15  45_v_15  60_v_15  45_v_30  60_v_30  60_v_45
[1,] "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint"
[2,] "15"        "30"        "45"        "60"        "30"        "45"        "60"        "45"        "60"        "60"
[3,] "00"        "00"        "00"        "00"        "15"        "15"        "15"        "30"        "30"        "45"
```

```
> results(dds, contrast = contrasts[, 2])
log2 fold change (MLE): timepoint 30 vs 00
Wald test p-value: timepoint 30 vs 00
DataFrame with 6513 rows and 6 columns
```

	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
1	1643.7457	-0.900131	0.225114	-3.99856	6.37297e-05	0.002809493
2	106.1707	-0.876365	0.313066	-2.79930	5.12139e-03	0.066589377
3	134.4141	-0.753498	0.311437	-2.41942	1.55451e-02	0.133487340
4	203.8053	-1.094019	0.227780	-4.80296	1.56338e-06	0.000140911
5	67.2922	-1.001061	0.336647	-2.97362	2.94309e-03	0.046569804
...	...	...	...	...	...	...
6509	36.1437	-0.0313828	0.417826	-0.0751098	0.9401274	0.979001
6510	603.5468	-0.4020281	0.181778	-2.2116386	0.0269916	0.185475
6511	42.8792	-0.3776449	0.378523	-0.9976807	0.3184342	0.633629
6512	15.2541	-0.5818438	0.642764	-0.9052221	0.3653477	NA
6513	37.8941	-0.8396196	0.427293	-1.9649731	0.0494173	0.258224

DESeq2 analysis workflow

```
> contrasts
      15_v_00 30_v_00 45_v_00 60_v_00 30_v_15 45_v_15 60_v_15 45_v_30 60_v_30 60_v_45
[1,] "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint"
[2,] "15"        "30"        "45"        "60"        "30"        "45"        "60"        "45"        "60"        "60"
[3,] "00"        "00"        "00"        "00"        "15"        "15"        "15"        "30"        "30"        "45"
```

```
> results(dds, contrast = contrasts[, 2])
log2 fold change (MLE): timepoint 30 vs 00
Wald test p-value: timepoint 30 vs 00
DataFrame with 6513 rows and 6 columns
```

	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
1	1643.7457	-0.900131	0.225114	-3.99856	6.37297e-05	0.002809493
2	106.1707	-0.876365	0.313066	-2.79930	5.12139e-03	0.066589377
3	134.4141	-0.753498	0.311437	-2.41942	1.55451e-02	0.133487340
4	203.8053	-1.094019	0.227780	-4.80296	1.56338e-06	0.000140911
5	67.2922	-1.001061	0.336647	-2.97362	2.94309e-03	0.046569804
...	...	...	...	...	...	...
6509	36.1437	-0.0313828	0.417826	-0.0751098	0.9401274	0.979001
6510	603.5468	-0.4020281	0.181778	-2.2116386	0.0269916	0.185475
6511	42.8792	-0.3776449	0.378523	-0.9976807	0.3184342	0.633629
6512	15.2541	-0.5818438	0.642764	-0.9052221	0.3653477	NA
6513	37.8941	-0.8396196	0.427293	-1.9649731	0.0494173	0.258224

DESeq2 analysis workflow

```
> contrasts
  15_v_00  30_v_00  45_v_00  60_v_00  30_v_15  45_v_15  60_v_15  45_v_30  60_v_30  60_v_45
[1,] "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint"
[2,] "15"        "30"        "45"        "60"        "30"        "45"        "60"        "45"        "60"        "60"
[3,] "00"        "00"        "00"        "00"        "15"        "15"        "15"        "30"        "30"        "45"
```

```
> results(dds, contrast = contrasts[, 2])
log2 fold change (MLE): timepoint 30 vs 00
Wald test p-value: timepoint 30 vs 00
DataFrame with 6513 rows and 6 columns
```

	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
1	1643.7457	-0.900131	0.225114	-3.99856	6.37297e-05	0.002809493
2	106.1707	-0.876365	0.313066	-2.79930	5.12139e-03	0.066589377
3	134.4141	-0.753498	0.311437	-2.41942	1.55451e-02	0.133487340
4	203.8053	-1.094019	0.227780	-4.80296	1.56338e-06	0.000140911
5	67.2922	-1.001061	0.336647	-2.97362	2.94309e-03	0.046569804
...	...	...	...	...	...	...
6509	36.1437	-0.0313828	0.417826	-0.0751098	0.9401274	0.979001
6510	603.5468	-0.4020281	0.181778	-2.2116386	0.0269916	0.185475
6511	42.8792	-0.3776449	0.378523	-0.9976807	0.3184342	0.633629
6512	15.2541	-0.5818438	0.642764	-0.9052221	0.3653477	NA
6513	37.8941	-0.8396196	0.427293	-1.9649731	0.0494173	0.258224

DESeq2 analysis workflow

```
> contrasts
  15_v_00  30_v_00  45_v_00  60_v_00  30_v_15  45_v_15  60_v_15  45_v_30  60_v_30  60_v_45
[1,] "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint" "timepoint"
[2,] "15"        "30"        "45"        "60"        "30"        "45"        "60"        "45"        "60"        "60"
[3,] "00"        "00"        "00"        "00"        "15"        "15"        "15"        "30"        "30"        "45"
```

```
> results(dds, contrast = contrasts[, 2])
log2 fold change (MLE): timepoint 30 vs 00
Wald test p-value: timepoint 30 vs 00
DataFrame with 6513 rows and 6 columns
```

	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
1	1643.7457	-0.900131	0.225114	-3.99856	6.37297e-05	0.002809493
2	106.1707	-0.876365	0.313066	-2.79930	5.12139e-03	0.066589377
3	134.4141	-0.753498	0.311437	-2.41942	1.55451e-02	0.133487340
4	203.8053	-1.094019	0.227780	-4.80296	1.56338e-06	0.000140911
5	67.2922	-1.001061	0.336647	-2.97362	2.94309e-03	0.046569804
...	...	...	...	...	...	...
6509	36.1437	-0.0313828	0.417826	-0.0751098	0.9401274	0.979001
6510	603.5468	-0.4020281	0.181778	-2.2116386	0.0269916	0.185475
6511	42.8792	-0.3776449	0.378523	-0.9976807	0.3184342	0.633629
6512	15.2541	-0.5818438	0.642764	-0.9052221	0.3653477	NA
6513	37.8941	-0.8396196	0.427293	-1.9649731	0.0494173	0.258224

Regularized log counts

- DESeq2 is primarily designed to **estimate fold-change** between conditions, **NOT** actual abundance!

Regularized log counts

- DESeq2 is primarily designed to **estimate fold-change** between conditions, **NOT** actual abundance!
- Rlog (regularized log) can be used to approximate gene expression levels.

Regularized log counts

- DESeq2 is primarily designed to **estimate fold-change** between conditions, **NOT** actual abundance!
- Rlog (regularized log) can be used to approximate gene expression levels.
- Rlog transforms the count data to the **log2 scale** in a way which
 - 1) minimizes differences between samples for rows with small counts
 - 2) normalizes with respect to library size

DESeq2::rlog()