

# CDesk: 1. BulkRNA pipeline

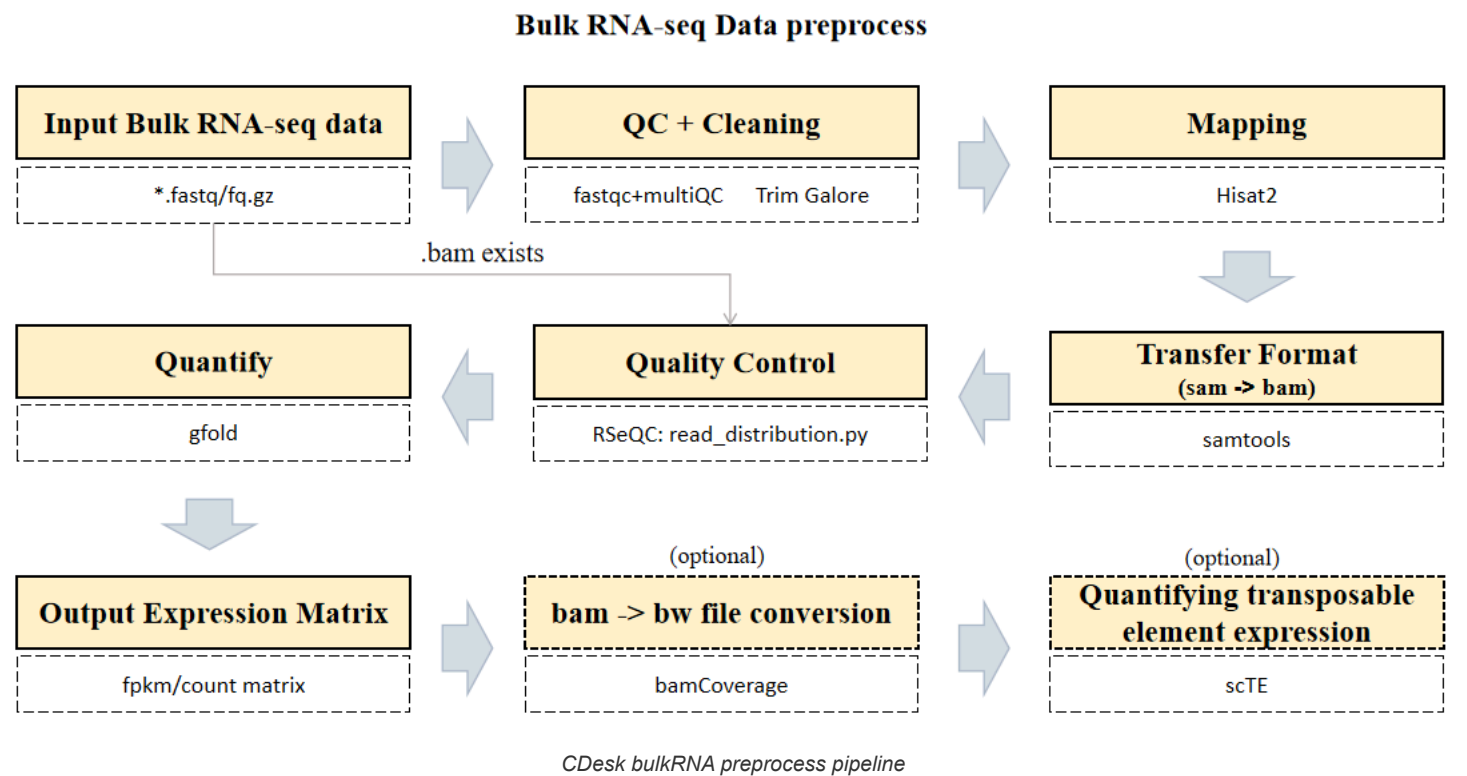
Our CDesk bulkRNA module comprises of 7 function submodules. Here we present you the CDesk bulkRNA working pipeline and how to use it to analyze your bulkRNA data.

- [1.1 bulkRNA: Preprocess](#)
- [1.2 bulkRNA: Correlation](#)
- [1.3 bulkRNA: DEG](#)
- [1.4 bulkRNA: Enrich](#)
- [1.5 bulkRNA: Similarity](#)
- [1.6 bulkRNA: Clustering](#)
- [1.7 bulkRNA: Splice](#)
- [CDesk handbook](#)

## 1.1 bulkRNA: Preprocess

The CDesk bulkRNA preprocess pipeline is illustrated in the figure below. The input consists of a directory containing compressed FASTQ files in either paired-end format (`xxx_1.fastq.gz/xxx_1.fq.gz` and `xxx_2.fastq.gz/xxx_2.fq.gz`), or single-end format (`xxx.fastq.gz/xxx.fq.gz`).

The pipeline first checks whether a BAM file corresponding to each FASTQ sequencing file already exists. If it does, the alignment step is skipped, and quantification proceeds after verifying that the BAM file is properly sorted and indexed, thereby saving computational time. Next, each FASTQ file undergoes quality control using FastQC and MultiQC to assess key metrics such as base quality distribution, GC content, and adapter contamination. Following quality assessment, Trim Galore is applied for data cleaning to remove low-quality bases and adapter sequences, ensuring high accuracy in downstream analyses. Subsequently, HISAT2 is used for sequence alignment, mapping the cleaned RNA-seq reads to the reference genome to determine their genomic origins. The resulting SAM files are then converted into sorted and indexed BAM files using Samtools. After format conversion, gene and transcript expression levels are quantified using GFOLD, which computes expression values based on reference genome annotations or transcript assembly results. Following alignment, read distribution across genomic regions—such as exons, introns, and intergenic regions—is analyzed using RSeQC. Finally, FPKM and count-based gene expression matrices are generated and consolidated for downstream analysis. Optionally, users can generate BigWig files from BAM files using bamCoverage, and transposable element (TE) expression levels can be analyzed and quantified using scTE.



Here is an example about how to use the CDesk HiC preprocess module.

```
CDesk bulkRNA preprocess \
-i ../../input_directory -o ../../output_directory \
-s mm10 -t 50 -l 2 -bw -te
```

Parameters(*necessary)	Description	Default value
-i,--input*	The input directory	
-o,--output*	The output directory	
-s,--species*	The species specified	
-t,--thread	The number of threads to use	8
-l	1:Single sequencing, 2:Pair sequencing	2
-bw	If specified, perform a bam -> bigwig file transfer (optional)	
-te	If specified, perform a TE expression analysis (optional)	

If the pipeline runs successfully, you will see output similar to the figure shown below.

- Bam: Stores the intermediate sam and bam files.
- BW: Stores the bw files.
- Expression: Stores the count and fpkm expression matrix file and gfold intermediate files.
- TE: Stores the scTE results.
- QC: Stores the fastqc and multiqc result.
- Log: Stores the log files of fastqc and multiqc, trim\_galore, mapping, gfold, bw generation and scTE.



CDesk bulkRNA preprocess example result

► A successful CDesk bulkRNA preprocess running process

# 1.2 bulkRNA: Correlation

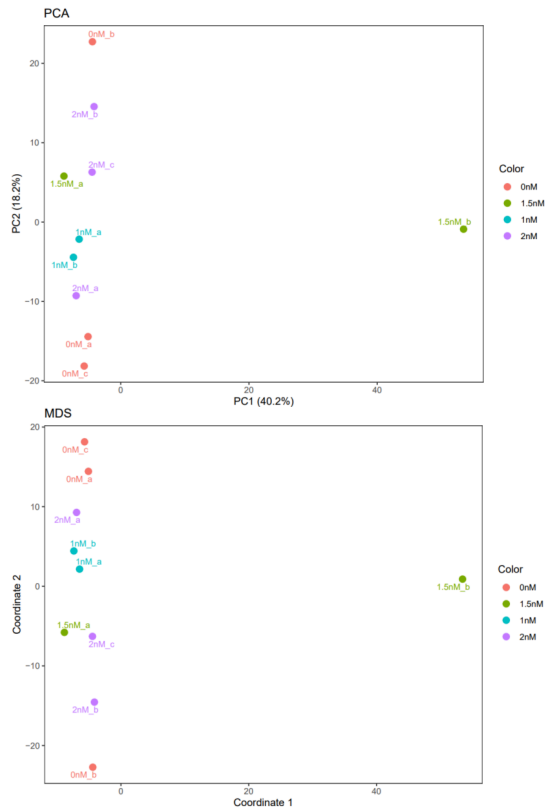
The CDesk bulkRNA correlation module performs correlation analysis on RNA-seq data across different samples to visualize sample relationships. By calculating and displaying sample similarities, it enables researchers to assess overall data structure. Users can optionally apply removeBatchEffect or ComBat batch effect correction methods to adjust for technical variations. Subsequently, Principal Component Analysis (PCA) plots, Multidimensional Scaling (MDS) plot and correlation heatmaps are generated to help researchers interpret sample distribution and underlying patterns in the data.

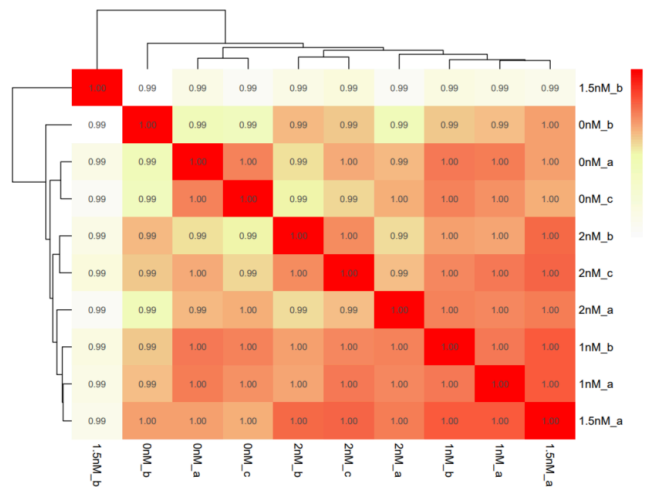
Here is an example about how to use the CDesk bulkRNA correlation module.

```
CDesk bulkRNA correlation \  
-i ../expression.csv \  
-o ../output_directory \  
--group ../group.csv --batch combat
```

Parameters(*necessary)	Description	Default value
-i,--input*	the input gene expression file (.csv), column name as sample, row name as gene	
-O,--output*	The output directory	
--group*	The grouping file	
--batch	Specify the batch effect removing method {no,removeBatchEffect,combat}	no
--width	The plot width	8
--height	The plot height	6

If the pipeline runs successfully, there would be a correlation heatmap pdf file, a PCA plot pdf file and a MDS plot in the output directory.





*CDesk bulkRNA correlation example result*

- A successful CDesk bulkRNA correlation running process
- What should the grouping file look like?

## 1.3 bulkRNA: DEG

The CDesk bulkRNA DEG (differential expressed genes) module performs differential expression analysis on bulk RNA-seq data and provides three analytical methods: DESeq2, adjusted-t, and GFOLD. DESeq2 takes a count-based expression matrix as input, adjusted-t accepts either count or FPKM expression matrices, and GFOLD directly uses BAM files for analysis and is useful when no replicate is available. Following differential expression analysis, results are visualized using heatmaps and volcano plots or MA plots to illustrate gene expression patterns and the significance of changes across different conditions.

Here is an example about how to use the CDesk HiC matrix module.

```
# Deseq2 (Only accept count integer expression matrix)
CDesk bulkRNA DEG dese22 \
-i ../../count_expression_matrix.csv -o ../../output_directory \
--group ../../grouping.csv -p --fc 1 --pval 0.05 --gene ../../genes.txt

# adjusted-t
CDesk bulkRNA DEG adjusted_t \
-i ../../expression_matrix.csv -o ../../output_directory \
--group ../../grouping.csv -p --fc 1 --pval 0.05 --top_num 1000

# gfold
CDesk bulkRNA DEG gfold \
-i ../../bam_files -o ../../output_directory \
--group ../../grouping_file.csv --readcnt ../../readcnt_files \
-t 50 -s species -p --gene ../../genes.txt
```

Parameters(*necessary)	Description	Default value
<b>deseq2,adjusted_t</b>		
-i,--input*	The input expression matrix file	
-o,--output*	Output directory	
--group*	The grouping file	
-p	Whether to plot or not	
--gene	Interested gene file to mark in the vocalno plot and heatmap	
--top_num	Number of top differential expression genes for heatmap if no gene file provided	500
--fc	fold change threshold of DEGs	1

Parameters <sup>(*necessary)</sup>	Description	Default value
--pval	p_adjusted threshold of DEGs	0.05
--width	Plot width	5
--height	Plot height	5
<b>gfold</b>		
-i,--input*	Input bam files directory	
-o,--output*	Output directory	
--group*	The grouping file	
-s,--species*	The species specified	
-p	Whether to plot or not	
--gene	Interested gene file to mark in the MA plot and heatmap	
--top_num	Number of top differential expression genes for heatmap if no gene file provided	500
--gfold	gfold threshold of DEGs	1
--readcnt	Provide the readcnt file directory to skip bam->readcnt step	
-t,--thread	Number of threads	20
--width	Plot width	5
--height	Plot height	5

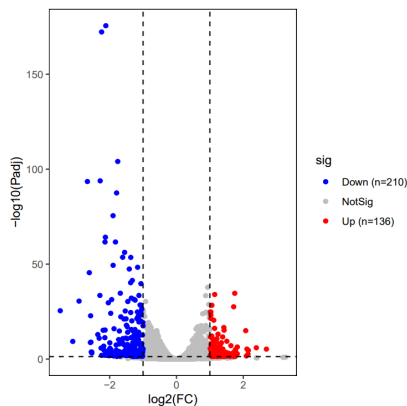
If the pipeline runs successfully, for deseq2 and adjusted\_t analysis, there would be volcano plots, a heatmap plot and deg result csv files in the output directory. For gfold analysis, there would be a heatmap plot, MA plots, deg result files, expression matrix files, intermediate gfold readCnt files and gfold analysis result files.

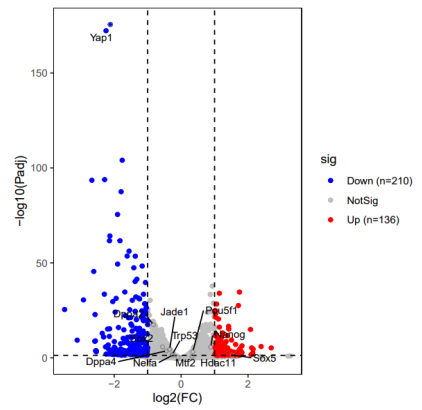
```

├── 2C_vs_4C.csv
├── 2C_vs_4C_MA.pdf
├── 2C_vs_8C.csv
├── 2C_vs_8C_MA.pdf
├── count.csv
├── diff
│   ├── 2C_vs_4C.diff
│   ├── 2C_vs_4C.diff.ext
│   ├── 2C_vs_8C.diff
│   └── 2C_vs_8C.diff.ext
├── fpkm.csv
├── heatmap.pdf
├── readCnt
│   ├── GSM7789776.read_cnt
│   ├── GSM7789777.read_cnt
│   ├── GSM7789778.read_cnt
│   ├── GSM7789779.read_cnt
│   ├── GSM7789780.read_cnt
│   ├── GSM7789781.read_cnt
│   ├── GSM7789782.read_cnt
│   ├── GSM7789783.read_cnt
│   └── GSM7789784.read_cnt

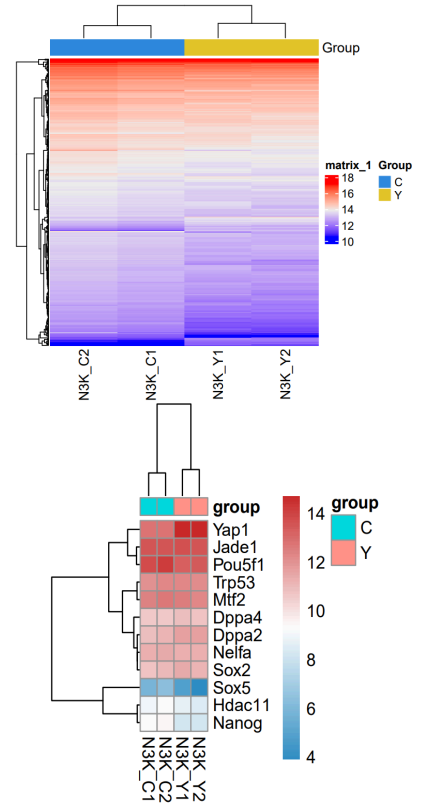
```

*CDesk bulkRNA DEG gfold example result*

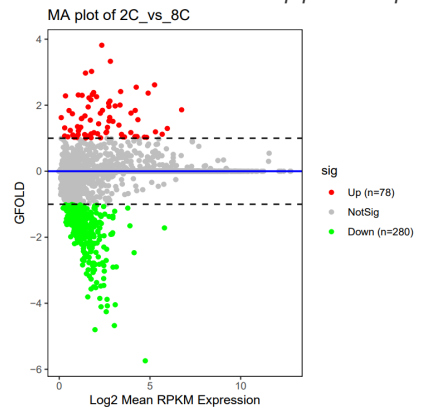


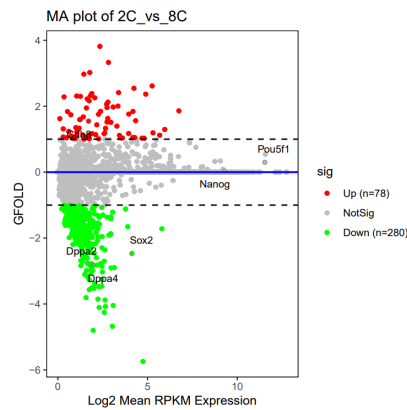


CDesk bulkRNA DEG volcano plot example



CDesk bulkRNA DEG heatmap plot example





*CDesk bulkRNA DEG MA plot example*

- A successful CDesk bulkRNA DEG gfold running process
- What should the grouping file look like?

## 1.4 bulkRNA: Enrich

CDesk bulkRNA enrich module performs gene functional enrichment analysis. It accepts either a user-specified gene list for GO and KEGG enrichment analysis, or a ranked gene list file (e.g., DEG result file) for GSEA analysis. The module outputs enrichment results along with visualization plots of enriched functional pathways. By default, it generates plots for the top 10 most significant GO and KEGG terms of each ontology and the top 5 most significant GSEA pathways. Users can customize the visualizations by modifying the enrichment result file or by specifying particular pathways of interest.

Here is an example about how to use the CDesk bulkRNA enrich module.

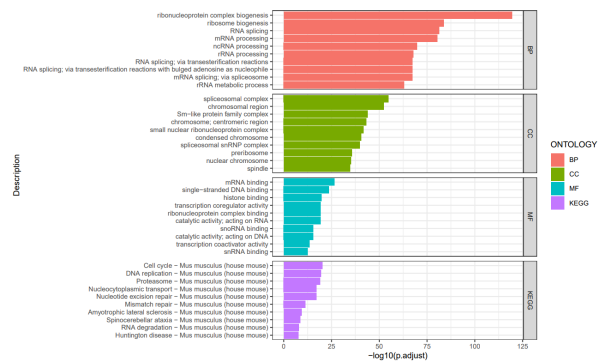
```
# GO and KEGG
# single sample
CDesk bulkRNA enrich analyze \
-i ../genes.txt -o ../output_directory \
-s mouse --type single
# single sample custom (Specify the reference customer file instead of species)
CDesk bulkRNA enrich analyze \
-i ../genes.txt -o ../output_directory \
--custom ../custom.txt --type single
# multiple sample
CDesk bulkRNA enrich analyze \
-i ../multi.csv -o ../output_directory \
-s mouse --type multi
# multiple sample custom
CDesk bulkRNA enrich analyze \
-i ../multi.csv -o ../output_directory \
--custom ../custom.txt --type multi

# GO and KEGG custom plot
# single sample
CDesk bulkRNA enrich plot \
-i ../enrichment_results.csv \
-o ../output_directory --type single
# multiple sample
CDesk bulkRNA enrich plot \
-i ../enrichment_results_combine.csv \
-o ../output_directory --type multi

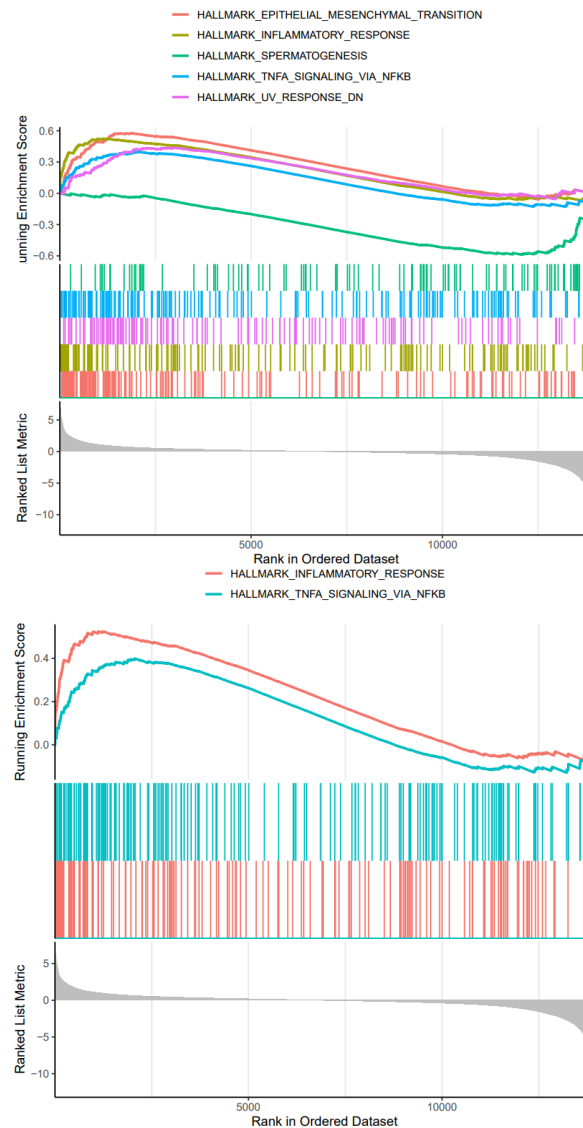
# GSEA (custom plot)
CDesk bulkRNA enrich GSEA \
-i ../ranked_gene_list_file.csv -o ../output_directory \
-s pig (--path ../paths.txt)
```

Parameters(*necessary)	Description	Default value
<b>analyze</b>	GO and KEGG functional encichment analysis	
-i,--input*	The single column file of interested genes (SYMBOL) for single sample or the multiple sample gene list file	
-o,--output*	The output directory	
--type*	Analyze type: single sample/multiple samples	{simple,multi}
--custom	The reference customer file containing two columns for custom analysis (every row is consisted of two factors: the customer term name and gene of interests separated by tab)	
--width	Plot width	10
--height	Plot height	6
<b>plot</b>	GO and KEGG custom plot(You can manually modify the GO and KEGG result file then plot)	
-i,--input*	The enrichment result file	
-o,--output*	The output directory	
--type*	Analyze type: single sample/multiple samples	{simple,multi}
--width	Plot width	10
--height	Plot height	6
<b>GSEA</b>	GSEA functional encichment analysis	
-i,--input*	The ranked gene list file (e.g., DEG result file)	
-o,--output*	The output directory	
-s,--species*	The species specified	{human,mouse,pig,chicken,rat}
--cols	Columns used	gene_name,log2FoldChange
--path	Specify the paths to plot	
--width	Plot width	7
--height	Plot height	7

If the pipeline runs successfully, there would be a functional enrichment analysis result file, a bubble plot and bar plot for GO and KEGG single sample analysis. There would be a combined functional enrichment analysis result file and a combined bubble plot for GO and KEGG multiple samples analysis. There would be a GSEA plot and GSEA analysis result for GSEA analysis. There would be new plots if you run the plot for GO/KEGG or rerun GSEA with specified functional pathways.







*CDesk bulkRNA gsea enrichment example result*

- A successful CDesk bulkRNA enrich running process
- What should the multiple sample gene list file look like?
- What should the gsea input file look like?

## 1.5 bulkRNA: Similarity

CDesk bulkRNA similarity module performs batch effect correction on multiple RNA-seq datasets, followed by gene expression analysis, PCA, and sample correlation analysis, generating corresponding visualizations. It outputs a sample correlation heatmap and PCA plot. If a gene list is provided, the module analyzes the expression patterns of these genes across groups, and visualizes their batch-corrected expression levels and  $CV^2$  (square of the coefficient of variation) for each group. When more than 20 genes are provided, the module displays the overall expression pattern across all genes rather than individual gene profiles, ensuring clarity and interpretability.

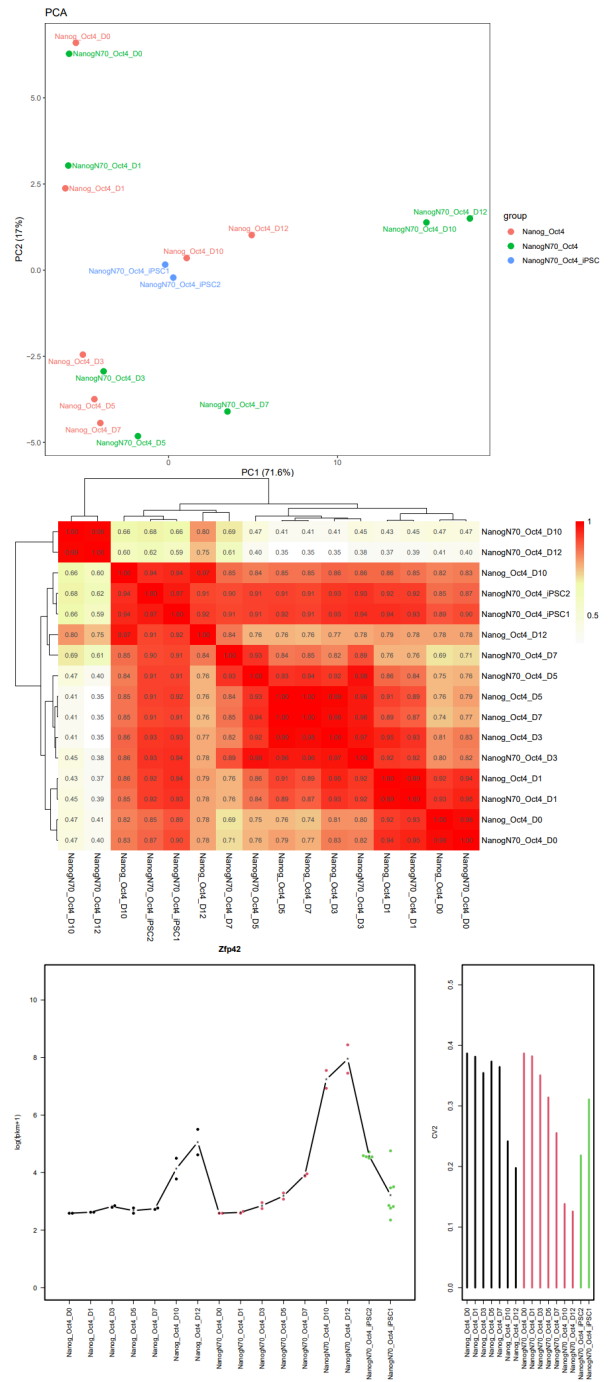
Here is an example about how to use the CDesk bulkRNA similarity module.

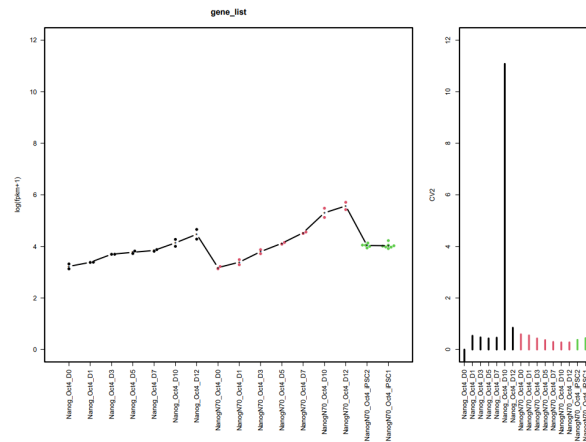
```
CDesk bulkRNA similarity \
-i ../../file_list.txt -o ../../output_directory \
--batch removeBatchEffect --group ../../grouping.csv \
--gene ../../genes.txt
```

Parameters <sup>(*necessary)</sup>	Description	Default value
-i,--input*	The gene expression list file	
-o,--output*	Output directory	

Parameters(*necessary)	Description	Default value
--group*	The grouping file	
--batch	The batch effect removing method {no,removeBatchEffect,combat}	no
--gene	Specify the gene list txt file	ALL
--width	Plot width	10
--height	Plot height	8

If the pipeline runs successfully, it would output a correlation heatmap, a PCA plot and a bar plot showing CV and expression in each group if genes provided.





*CDesk bulkRNA similarity example result*

► What should the similarity grouping file look like?

## 1.6 bulkRNA: Clustering

CDesk bulkRNA cluster module performs clustering analysis of genes. The WGCNA function implements the weighted gene co-expression network analysis (WGCNA) pipeline to explore relationships between gene expression patterns and phenotypic traits. It includes a comprehensive workflow: data preprocessing, sample and gene quality control, filtering of lowly expressed genes, cluster analysis, soft-threshold selection, co-expression network construction, module detection, correlation analysis between modules and traits, and extraction and visualization of key modules and hub genes. The results are output in both graphical and file formats to a specified directory. The kmeans function performs unsupervised k-means clustering on the gene expression matrix, grouping genes with similar expression patterns. It outputs gene membership for each cluster and generates visualizations of expression profiles across different groups, facilitating the identification of co-regulated gene sets.

Here is an example about how to use the CDesk bulkRNA cluster module.

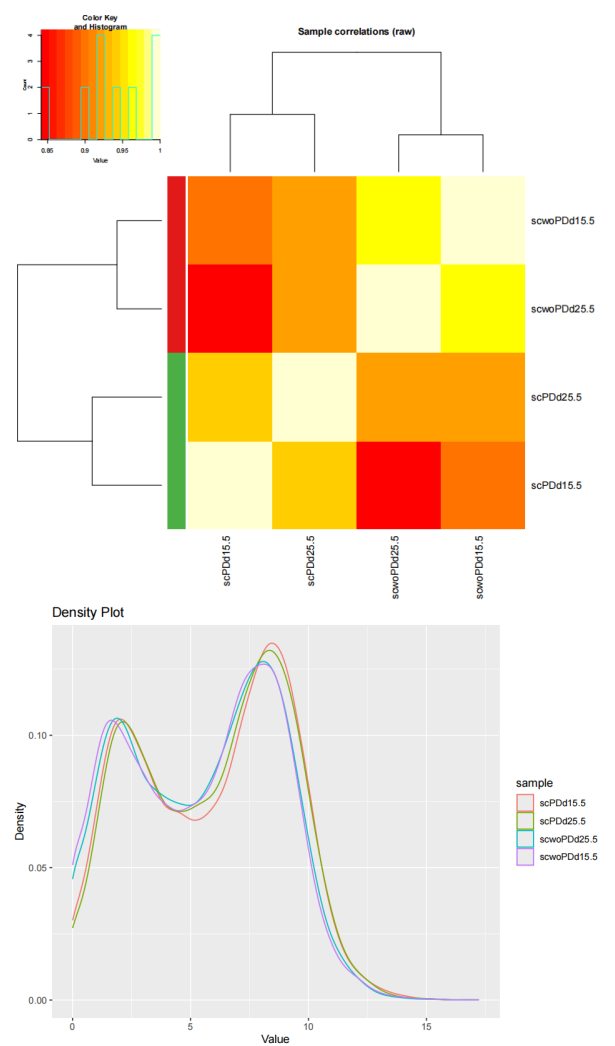
```
# WGCNA
CDesk bulkRNA cluster WGCNA \
-i ../../expression_matrix.csv \
--pheno ../pheno.csv \
--trait trait -o ../../output_directory

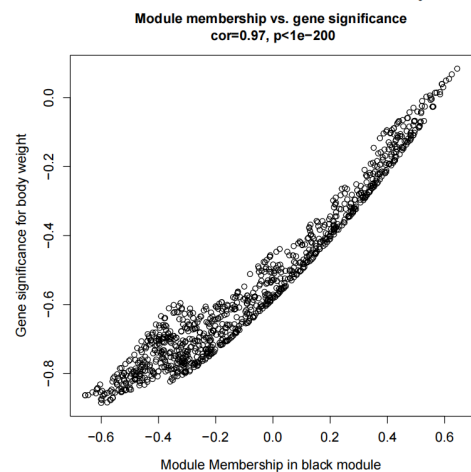
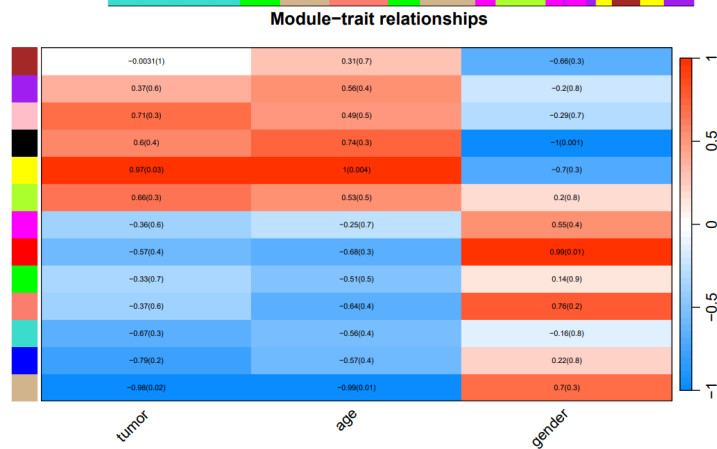
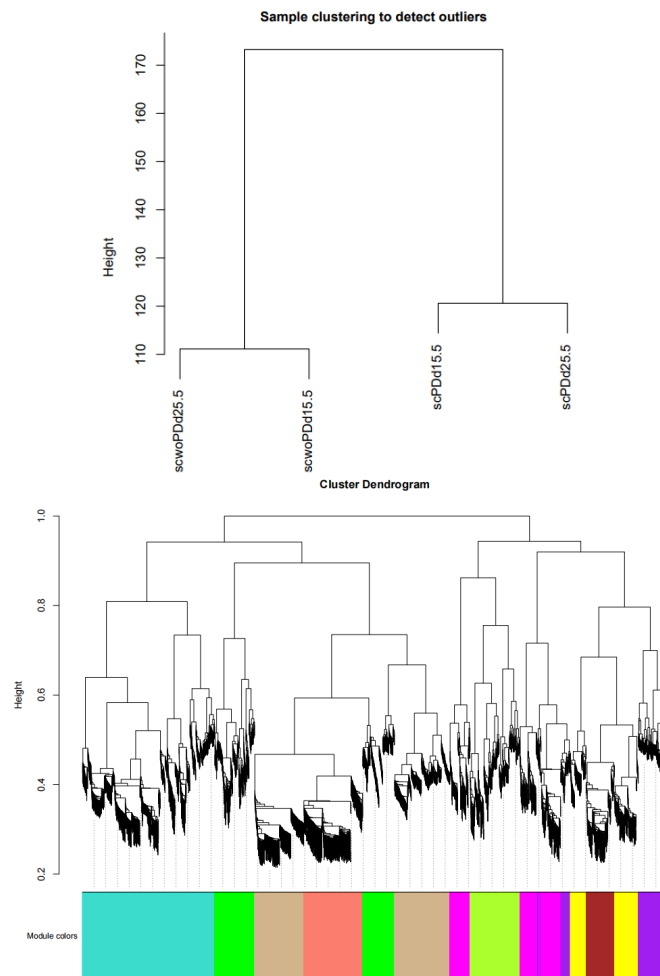
# Kmean
CDesk bulkRNA cluster kmean \
-i ../../expression_matrix.csv -o ../../output_directory \
--group ../group.csv --gene ../genes.txt
```

Parameters(*necessary)	Description	Default value
<b>WGCNA</b>		
-i,--input*	The input expression matrix file	
-o,--output*	Output directory	
--pheno*	The sample trait information file	
--trait*	The phenotypes specified to calculate correlations with gene modules	
<b>kmean</b>		
-i,--input*	The input expression matrix file	
-o,--output*	Output directory	
--group*	The grouping file	
--cluster	Number of clusters for K-means clustering	6

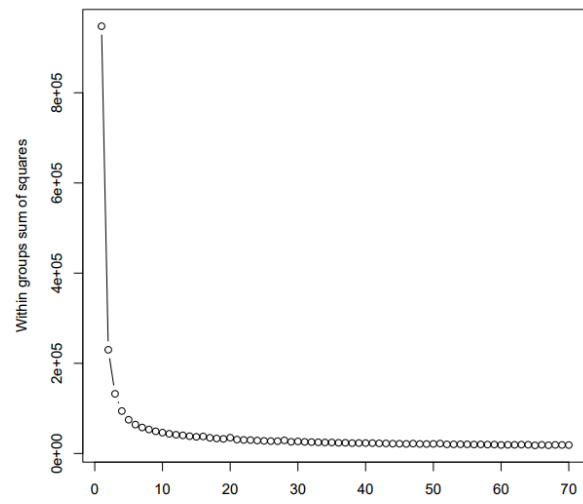
Parameters(*necessary)	Description	Default value
--method	Distance measure method to be used for clustering {euclidean,maximum,manhattan,canberra,binary,pearson,abspearson,abscorrelation,correlation,spearman,kendall}	correlati
--gene	Specify the gene list txt file	
--width	Plot width	10
--height	Plot height	8

If the pipeline runs successfully, for WGCNA analysis, it would output: gene members of each identified module, scatter plots showing the correlation between module eigengenes and trait profiles, heatmap of sample-to-sample similarity, dendrogram (hierarchical clustering tree) of genes, density plot of gene count distribution across samples, heatmap of module-trait associations and cluster plots visualizing gene expression patterns across modules. For kmean analysis, it would output: the list of genes assigned to the cluster, line plots and heatmaps showing the average expression profile of genes within each cluster and an elbow plot for evaluating the optimal number of clusters.

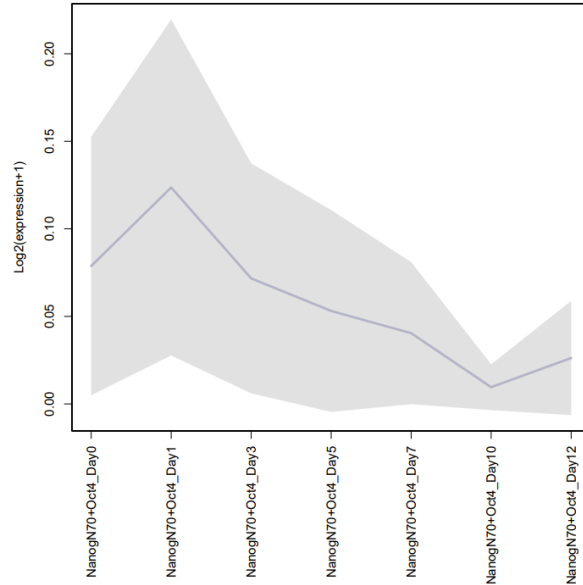




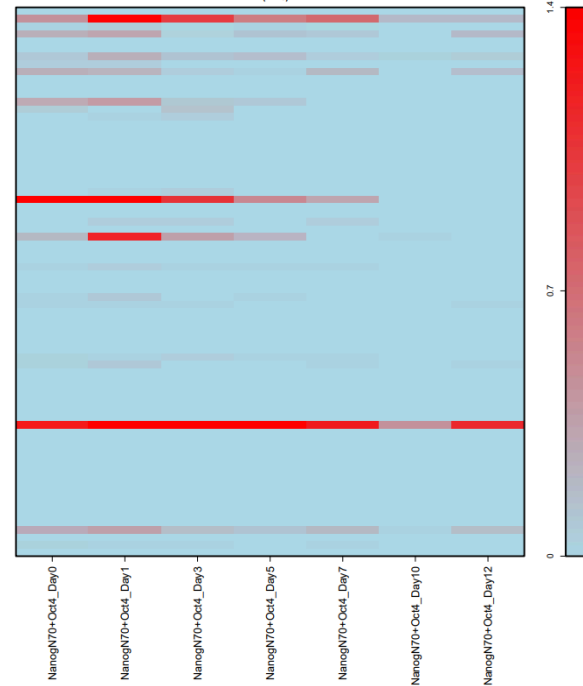
*CDesk bulkRNA cluster WGCNA example result*



C1(n=73)



C1(n=73)



CDesk bulkRNA cluster kmean example result

► A successful CDesk bulkRNA cluster running process

- What should the WGCNA input sample trait information file look like?
- What should the kmean grouping file look like?

## 1.7 bulkRNA: Splice

CDesk bulkRNA splice module provides two main functions. The detect function identifies differential splicing events from BAM files using [rMATS](#), a widely used tool for analyzing alternative splicing in RNA-seq data. rMATS employs a statistical model to quantify splicing event expression in samples (with biological replicates), and uses a likelihood ratio test to compute P-values reflecting differences in Inclusion Level (IncLevel) between two groups. IncLevel is analogous to Percent Spliced In (PSI) in definition. P-values are subsequently adjusted using the Benjamini-Hochberg procedure to control the false discovery rate (FDR). The draw function generates Sashimi plots from BAM files to visually represent differential splicing events.

rMATS can detect five major types of alternative splicing events:

1. Skipped Exon (SE)
2. Alternative 5' Splice Site (A5SS)
3. Alternative 3' Splice Site (A3SS)
4. Mutually Exclusive Exons (MXE)
5. Retained Intron (RI)

The [rmats2sashimiplot](#) tool creates Sashimi plot visualizations from rMATS output. It can also generate plots using a gene annotation file (e.g., GTF) and user-specified genomic coordinates, enabling flexible and intuitive visualization of splicing patterns.

Here is an example about how to use the CDesk bulkRNA splice module.

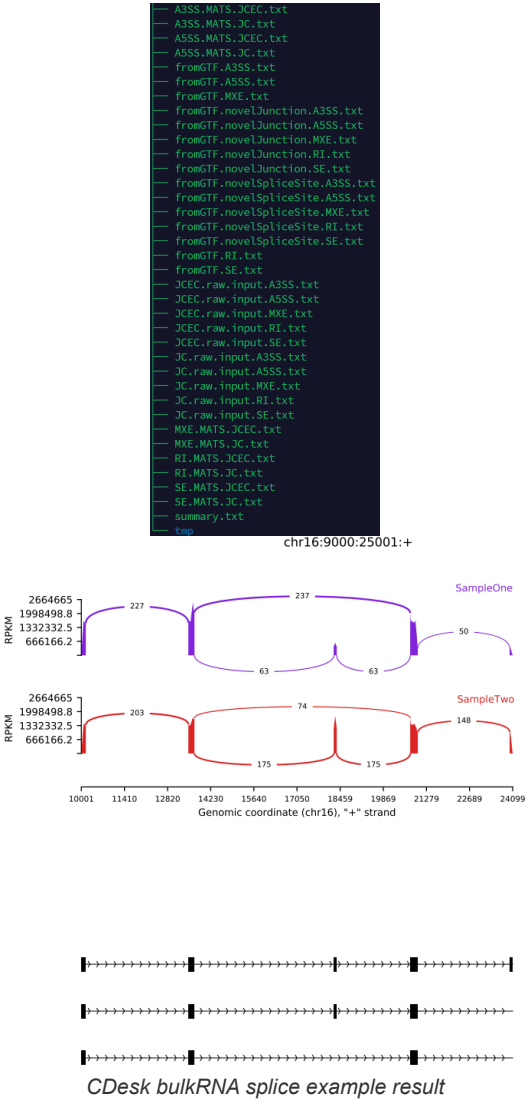
```
# detect
CDesk bulkRNA splice detect \
--b1 /.../sample1.txt --b2 /.../sample2.txt \
--species species -o /.../output_directory

# draw
CDesk bulkRNA splice draw \
--b1 /.../sample1.txt --b2 /.../sample2.txt -o /.../output_directory \
--region chr16:+:9000:25000 --species species --group /.../grouping.gf
```

Parameters(*necessary)	Description	Default value
<b>detect</b>		
--b1*	The first txt file containing BAM file paths seprated by comma	
--b2*	The second txt file containing BAM file paths seprated by comma	
-o,--output*	Output directory	
--species*	Specify the species	
-t,--thread	Number of threads	10
--length	Length of each read	150
--variable_read_length	Allow reads with lengths that differ from read Length	
--allow_clipping	Allow alignments with soft or hard clipping to be used	
<b>draw</b>		
--b1*	The first txt file containing BAM file paths seprated by comma	
--b2*	The second txt file containing BAM file paths seprated by comma	
-o,--output*	Output directory	
--species*	Specify the species	
--region*	The genome region coordinates: format:{chromosome}:{strand}:{start}:{end}	

Parameters(*necessary)	Description	Default value
--group*	The path to a .gf file which groups the replicates	
--exon_s	How much to scale down exon	1
--intron_s	How much to scale down introns	1

If the pipeline runs successfully, for detect function, it would output the rrmats analysis result. For draw function, it would output the Sashimi plot and Sashimi index result folder in the output directory.



- What does the grouping gf file mean?
- What does the rrmats result mean?

## CDesk handbook