


**Integrative Genomics**

### 3. Gene Expression Exercises with R



ggibson.gt@gmail.com  
<http://www.gibsongroup.biology.gatech.edu>

**WIKIPEDIA**  
The Free Encyclopedia

View page  
 Contents  
 Featured content  
 Current events  
 Random article  
 Donate to Wikipedia  
 Privacy policy

Interaction  
 Help  
 About Wikipedia  
 Community portal  
 Recent changes  
 Contact us

Tools  
 What links here  
 Related changes  
 Special pages  
 Permanent link  
 Page information  
 Wikidata item  
 Cite this page

Development  
 Create a new account  
 Download our PDF files  
 Privacy policy

Language en

#### List of RNA-Seq bioinformatics tools

From Wikipedia, the free encyclopedia

**RNA-Seq** (/rɪnɑːˈsiːq/) is a technique that performs transcriptome studies based on next-generation sequencing technologies. This technique is largely dependent on bioinformatic tools developed to support the different steps of the process. Here are listed some of the principal tools commonly employed and links to some related web resources.

To follow an integrated guide to the analysis of RNA-Seq data, please see: [Roadmap for Integrating RNA-Seq Data](#), [RNA-Seq Tutorial](#) or [RNA-Seq workflow](#). Also, important tools are [BBTools](#), [RNA-SeqQC](#), [RNA-SeQC](#), [RSeQC](#), [RSEM](#), [Rsubread](#) and [Rsubread](#).

Category	Tools
1 Quality control and pre-processing data	1.1 Quality control and filtering data
	1.2 Detection of chimeric reads
	1.3 Pre-processing data
2 Alignment Tools	2.1 Short (Illumina) aligners
	2.2 Spliced aligners
	2.2.1 Aligners based on known splice junctions (junction-guided aligners)
	2.2.2 De novo Splice Aligners
	2.2.2.1 De novo Splice Aligners that also use annotation optionally
	2.2.2.2 Other Spliced Aligners
3 Quantitative analysis and Differential Expression	3.1 Read-count solutions
	3.2 Software (analytic pipeline / integrated solutions)
	4.1 Commercial Solutions
	4.2 Open-Source Bioinformatics Solutions
	5 Alternative Splicing Analysis
	6 Bias Correction
	7 Fusion genes/transcriptome reconstruction/interstructural variations
	8 Copy Number Variation identification
	9 RNA-Seq simulation
	10 Transcriptome assemblies
	10.1 Genome-guided assemblies
	10.2 Genome-independent (de novo) assemblies
	11 Co-expression networks
	12 miRNA prediction
	13 Visualization tools
	14 Functional Network & Pathway Analysis Tools
	15 Further simulation tools for RNA-Seq data
	16 RNA-Seq Databases
	17 Resources and Publications

# qvalue

qvalue is an R package for determining the false discovery rate from a list of p-values, adjusted for an estimate of the number of true nulls

Input: Rin3\_p.txt

The screenshot shows the Bioconductor website for the `qvalue` package. The page includes a navigation bar with links for Home, Install, Help, Developers, and About. The main content area displays the package name `qvalue` and provides information about its version (3.11), download statistics (top 5%), and platform compatibility (Linux, Windows, Mac OS X). A section titled "Q-value estimation for false discovery rate control" describes the package's functionality, including its authors (John Storey, Andrew Bass, Alan Dobney, David Robinson) and maintainer (John D. Storey). The page also includes an installation instruction: "To install this package, start R and enter: source('http://bioconductor.org/packages/2.10/bioc/html/qvalue/'); install.packages('qvalue')".

The screenshot shows an R console window on the left and a web browser window on the right. The R console displays the following text:

```
R version 3.1.2 (2014-10-31) -- "Pie in the Sky"
Copyright (C) 2014 The R Foundation
Platform: x86_64-w64-mingw32/x64 (64-bit)
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
Natural language support but running in an English locale
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> library(qvalue)
> help(package=qvalue)
starting httpd help server ... done
>
```

The web browser window shows the documentation for the 'qvalue' package version 1.0.0. The title is 'Q-value estimation for false discovery rate control'. The page includes a description of the package, a list of authors (Benjamini, Hochberg, and Tibshirani), and a list of functions: `qvalue`, `qvalue.p`, `qvalue.v`, `qvalue.w`, `qvalue.z`, `qvalue.m`, `qvalue.f`, `qvalue.g`, `qvalue.h`, `qvalue.i`, `qvalue.j`, `qvalue.k`, `qvalue.l`, `qvalue.m`, `qvalue.n`, `qvalue.o`, `qvalue.p`, `qvalue.q`, `qvalue.r`, `qvalue.s`, `qvalue.t`, `qvalue.u`, `qvalue.v`, `qvalue.w`, `qvalue.x`, `qvalue.y`, `qvalue.z`.

The screenshot shows an R console window on the left and an R Graphics Device window on the right. The R console displays the following code and output:

```
> library(qvalue)
> help(package=qvalue)
starting httpd help server ... done
> setwd("C:/Users/gjibson/My Documents/Teaching/SISG/2100 GRP/qvalue")
> data=read.table("Rin3_p.txt")
> head(data)
      V1
1 0.1100
2 0.0053
3 0.3700
4 0.0058
5 0.0001
6 0.0140
> pvalues <- data[,1]
> qobj <- qvalue(pvalues)
> summary(qobj)

Call:
qvalue(p = pvalues)

pi0: 0.2480381

Cumulative number of significant calls:

<1e-04 <0.001 <0.01 <0.025 <0.05 <0.1 <1
p-value 240 959 2412 3137 3788 4528 7991
q-value 62 604 2650 3758 4809 5974 8000

> hist(qobj$pvalues)
> plot(qobj)
> hist(qobj$pvalues, nclass=20)
> [write(qobj, file="Rin3q.txt")]
```

The R Graphics Device window shows four plots. The top-left plot is a line graph of  $\hat{\pi}_0$  vs  $\lambda$ , with  $\hat{\pi}_0 = 0.248$ . The top-right plot is a line graph of  $q$ -value vs  $p$ -value. The bottom-left plot is a line graph of significant calls vs  $q$ -value cut-off. The bottom-right plot is a line graph of rejected false positives vs significant calls.

## snm

snm is an R package for supervised normalization of microarray (or RNASeq) data, that simultaneously adjusts biological, technical and array effects

Input:

CAD8000.csv	data file
CAD_bio.csv	biological variable
CAD_adj.csv	adjustment variable (to fit)
CAD_adjrm	adjustment variable (to remove)
CAD_int	intensity-dependent (array) variable

DATA	A	B	C	D	E	F	G	H	I	J	K	L
1	EUH02661	EUH01927	EUH02357	CLH00229	EUH02482	CLH00189	EUH02317	EUH02121	EUH02210	CLH00238	EUH02394	
2	ILMN_238	14.68244	13.98202	14.46552	14.44341	13.87446	14.39205	13.8535	14.29878	14.03563	13.41536	13.69829
3	ILMN_166	14.51808	13.98157	15.04292	14.69002	13.74107	14.64559	13.61555	14.21511	13.5964	13.50156	13.47871
4	ILMN_168	14.09896	13.73664	13.98173	13.91768	13.85829	13.88826	13.88211	14.13509	13.9924	13.65327	13.81957
5	ILMN_210	14.88958	13.71523	14.93602	14.42121	13.55921	14.44894	13.45433	14.2198	13.53969	13.4686	13.42105
6	ILMN_224	14.35569	13.90495	14.45346	14.17866	13.62601	14.23437	13.51523	14.29185	13.69174	13.42759	13.58778
7	ILMN_212	14.67233	13.86557	15.08908	14.35686	13.54718	14.54751	13.61004	14.27196	13.5793	13.34899	13.41936
8	ILMN_223	14.15323	13.81096	14.06618	14.00923	13.7164	13.89014	13.54152	14.29942	13.57882	13.42607	13.56903
9	ILMN_209	13.92744	13.84332	13.89218	13.83482	13.81746	13.56429	13.85589	14.17429	13.79296	13.60181	13.64221
10	ILMN_222	13.81189	13.59864	13.79147	13.66698	13.80799	13.60064	13.73122	14.02785	13.92121	13.61169	13.75067

BIOL		ADJ_RM		ADJ			INT	
1	CVD_TYPE	49	Study Rin3	1	Gender Age BMI	1	Array	
2	EUH02661 ACUTE	50	EUH02253 A MOD	2	EUH02661 FEM 56 31	2	1	
3	EUH01927 ACUTE	51	EUH02381 A HIGH	3	EUH01927 FEM 54 26.9	3	2	
4	EUH02357 ACUTE	52	EUH01996 A MOD	4	EUH02357 MAL 51 20.7	4	3	
5	CLH00229 ACUTE	53	CLH00187 A MOD	5	CLH00229 MAL 52 24.4	5	4	
6	EUH02482 ACUTE	54	GG_0372 B MOD	6	EUH02482 MAL 62 26.1	6	5	
7	CLH00189 ACUTE	55	GG_0380 B HIGH	7	CLH00189 MAL 65 29.9	7	6	
8	EUH02317 ACUTE	56	GG_0396 B MOD	8	EUH02317 MAL 65 28.4	8	7	
9	EUH02121 ACUTE	57	GG_0398 B LOW	9	EUH02121 MAL 64 33.1	9	8	
10	EUH02210 FINE	58	GG_0399 B LOW	10	EUH02210 FEM 69 29.8	10	9	



```

source("http://biocconductor.org/biocLite.R")
biocLite("pvca")

> setwd("C:/Users/ggibson3/My Documents/Teaching/SISG/SISG GEP/pvca")
> library("Biobase")
Loading required package: BiocGenerics
Loading required package: parallel

> library(pvca)
> exprs <- as.matrix(read.table("CAD8000.csv", header=TRUE, sep=",", row.names=1, as.is=TRUE))
> class(exprs)
[1] "matrix"
> dim(exprs)
[1] 8000 100
> colnames(exprs)
 [1] "EUH02661" "EUH01927" "EUH02357" "CLR00229" "EUH02482" "CLR00189" "EUH02317" "EUH02121"
 [9] "EUH02210" "CLR00238" "EUH02394" "EUH02095" "EUH02638" "CLR00143" "EUH02362" "EUH02300"
[17] "EUH02515" "EUH03390" "EUH04433" "EUH01977" "EUH02063" "EUH02614" "EUH02054" "EUH02490"
[25] "EUH02465" "EUH02404" "EUH02686" "EUH02468" "EUH02391" "EUH02158" "EUH02271" "EUH02296"
[33] "EUH02287" "EUH02659" "EUH02475" "EUH02520" "CLR00094" "EUH02070" "EUH02396" "EUH02297"
[41] "EUH02276" "EUH02113" "EUH02062" "EUH02366" "CLR00190" "EUH02102" "EUH02253"
[49] "EUH02381" "EUH01996" "CLR00187" "GG_0372" "GG_0380" "GG_0396" "GG_0398" "GG_0399"
[57] "GG_0400" "GG_0401" "GG_0413" "GG_0415" "GG_0420" "GG_0423" "GG_0425" "GG_0428"
[65] "GG_0431" "GG_0433" "GG_0434" "GG_0435" "GG_0436" "GG_0437" "GG_0442" "GG_0443"
[73] "GG_0445" "GG_0452" "GG_0454" "GG_0464" "GG_0469" "GG_0473" "GG_0475" "GG_0478"
[81] "GG_0484" "GG_0489" "GG_0491" "GG_0503" "GG_0504" "GG_0506" "GG_0507" "GG_0508"
[89] "GG_0509" "GG_0510" "GG_0511" "GG_0514" "GG_0516" "GG_0517" "GG_0518" "GG_0519"
[97] "GG_0520" "GG_0525" "GG_0531" "GG_0532"

> head(exprs[,1:5])
      EUH02661 EUH01927 EUH02357 CLR00229 EUH02482
ILMN_2389211 14.091  13.892  13.747  14.056  14.243
ILMN_1667796 13.872  13.903  14.273  14.276  14.125
ILMN_1683271 13.419  13.665  13.180  13.487  14.252
ILMN_2106437 14.237  13.638  14.160  14.004  13.945
ILMN_2242491 13.692  13.830  13.667  13.756  14.015
ILMN_2127842 13.992  13.794  14.287  13.926  13.941

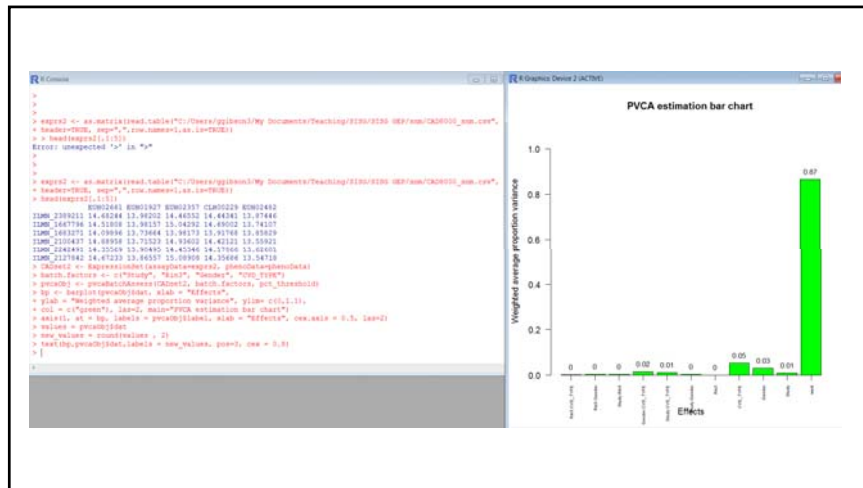
```

```

R Console
> pData <- read.table("CAD_ExpDes.csv", row.names=1, header=TRUE, sep=",")
> all(row.names(pData) == colnames(exprs))
[1] TRUE
> HGT010Data <- class(
+   Study      Bin      Gender  Age3      BMI3     CVD_TYPE  Array
+   factors    factors    factors    factors    factors    factor    integer
+   pData[c(1:5,20), c("Gender", "Age3", "BMI3")]
+   Gender  Age3  BMI3
EUH02342  FEM  MATURE  OVER
EUH01977  FEM  MATURE  OVER
+   methods <- data.frame(labelDescription=c("Batch", "QualClass", "Gender",
+   "AgeClass", "BMIClass", "CVD status", "Number"),
+   rownames=c("Study", "Bin", "Gender", "Age3", "BMI3", "CVD_TYPE", "Array"))
+   pData <- new("AnnotatedDataFrame", data=pData, varMetadata=metaData)
+   pData
An object of class "AnnotatedDataFrame"
rownames: EUH01977 ... GG_0532 (100 total)
varLabels: Study Bin ... Array (7 total)
varMetadata: labelDescription columns
+   CVDstat <- RegressionCoefficients(pData$exprs, pData$CVDTYPE)
+   pct_threshhold = 0.7
+   batch.factors <- c("Study", "Bin", "Gender", "Age3", "BMI3", "CVD_TYPE")
+   pData <- pData[batch.factors[batch.factors, pct_threshhold]]
+   HGT <- bagplot(pData$batch, xlab = "Effects")
+   ylim = "Weighted average proportion variance", ylim=c(0,1.1),
+   col = c("blue", "red", "black") #PCA estimation bar chart"
+   axis(1, at = HGT$label, las=2, main="PCA estimation bar chart")
+   values = pData$batch
+   new.values = round(values, 2)
+   text(HGT$plot$stat.label + new.values, pos=1, cex = 0.8)
+ }

```

Effect	Weighted average proportion variance
Batch	0.01
Bin	0.01
Gender	0.01
Age3	0.25
BMI3	0.01
CVD_TYPE	0.01
Array	0.54



## edgeR

edgeR is an R package for normalization of RNASeq counts using the negative binomial distribution to adjust for high variance at low expression

Input:

EM1_week2.txt	data file
EM1_week2.txt	data file
EM1_week2.txt	data file
targets_EM1.txt	design file

```

source("http://biocconductor.org/bioclite.R")
biocLite("edgeR")

> library(edgeR)
Loading required package: limma
Warning messages:
1: package 'edgeR' was built under R version 3.1.3
2: package 'limma' was built under R version 3.1.3
> setwd("~/Users/gjgilson/My Documents/Teaching/SISG/SISG Q&P/edgeR")
> targets <- readTargets("targets_EMI.txt")
> targets
  files group description
1 EMI_Week1.txt Sick Respiratory Infection
2 EMI_Week2.txt Sick Respiratory Infection
3 EMI_Week3.txt Better Health Improvement
> ed <- readDGE(targets, skip=0, comment.char = "#")
> ed$samples
  files group description lib.size norm.factors
1 EMI_Week1.txt Sick Respiratory Infection 40396453 1
2 EMI_Week2.txt Sick Respiratory Infection 47774592 1
3 EMI_Week3.txt Better Health Improvement 28294470 1
> head(ed$counts)
  1 2 3
42064 1091 903 1623
42065 24 25 21
42066 170 173 129
42067 1 2 1
42068 1270 1276 1283
42069 7655 8545 7388
> summary(ed$counts)
  1 2 3
Min. : 0 Min. : 0 Min. : 0
1st Qu.: 0 1st Qu.: 0 1st Qu.: 0
Median : 40 Median : 49 Median : 37
Mean : 1556 Mean : 1840 Mean : 1475
3rd Qu.: 1041 3rd Qu.: 1200 3rd Qu.: 975
Max. : 11093284 Max. : 1295951 Max. : 1315122
> dim(ed)
[1] 25963 3
  > keep <- rowSums(cpm(ed)) >= 2
  > ed <- ed[keep, keep.lib.sizes=FALSE]
  > head(ed$counts)
  1 2 3
42069 7655 8545 7388
42070 4431 7028 6784
42071 5553 5945 4441
42248 3945 5314 3461
42249 13088 15971 12261
42253 11005 14132 8969
  > dim(ed)
[1] 4356 3

```

