

Rを起動し、library(recount)と打ち込んで、recountパッケージがインストールされていることを確認しておいてください。

農学生命情報科学特論I 第2回

¹大学院農学生命科学研究科
アグリバイオインフォマティクス教育研究プログラム

²微生物科学イノベーション連携研究機構

門田幸二(かどた こうじ)

kadota@iu.a.u-tokyo.ac.jp

<http://www.iu.a.u-tokyo.ac.jp/~kadota/>

講義予定

- 第1回(2018年06月12日)
 - カウント情報取得の続き
 - データの正規化(RPK, RPM, RPKM/FPKM)
- 第2回(2018年06月19日)
 - サンプル間クラスタリング、Rのクラスオブジェクト
 - RのReference Manualの読み解き方、クラスタリング結果の客観的な評価
- 第3回(2018年06月26日)
 - 発現変動解析(多重比較問題とFDR)、各種プロット(M-A plot)
 - 発現変動解析(デザイン行列や3群間比較)
- 第4回(2018年07月03日)
 - 機能解析(Gene Ontology解析やパスウェイ解析)

Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?! カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

クラスタリング

①TCCパッケージを用いて、サンプル間クラスタリングを行う。②例題7。③入力ファイルは20,689遺伝子×36サンプルのカウントデータファイル。ヒト(HS)、チンパンジー(PT)、アカゲザル(RM)の3生物種の肝臓(Liver)データ。各12サンプル。コピペ実行

- 解析 | [発現量推定\(トランスクリプトーム配列を利用\)](#) (last modified 2014/07/09)
- 解析 | [クラスタリング | について](#) (last modified 2014/02/05)
- 解析 | [クラスタリング | サンプル間 | hclust](#) (last modified 2015/02/26) **NEW**
- 解析 | [クラスタリング | サンプル間 | TCC\(Sun_2013\)](#) (last modified 2015/03/02) **NEW**
- 解析 | [クラスタリング | 遺伝子間 | MBCluster.Seq\(Si...\)](#) (last modified 2014/02/05)

解析 | クラスタリング | サンプル間 | TCC(Sun_2013) **NEW**

TCCパッケージを用いてサンプル間クラスタリングを行うやり方を示します。clusterSample関数を利用した頑健なクラスタリング結果を返します。
「ファイル名」を「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し以下をコピペ。

1. 59 **②** 7. サンプルデータ41のリアルデータ(sample blekhman 36.txt)の場合:

[Blekhman et al., Genome Res., 2010](#)の 20,689 genes×36 samplesのカウントデータです。 **③**

```

in_f <- "sample_blekhman_36.txt" #入力ファイル名を指定してin_fに格納
out_f <- "hoge7.png" #出力ファイル名を指定してout_fに格納
param_fig <- c(700, 400) #ファイル出力時の横幅と縦幅を指定(単位はピクセル)

#必要なパッケージをロード
library(TCC) #パッケージの読み込み

#入力ファイルの読み込み
data <- read.table(in_f, header=TRUE, row.names=1, sep="\t", quote="") #in_fで指定したファイルの読み込み
dim(data) #オブジェクトdataの行数と列数を表示

#本番
out <- clusterSample(data, dist.method="spearman", #クラスタリング実行結果をoutに格納
                    hclust.method="average", unique.pattern=TRUE) #クラスタリング実行結果をoutに格納

#ファイルに保存
png(out_f, pointsize=13, width=param_fig[1], height=param_fig[2]) #出力ファイルの各種パラメータを指定
par(mar=c(0, 4, 1, 0)) #下、左、上、右の順で余白(行)を指定
plot(out, lab="", xlab="", cex.lab=1.2, #樹形図(デンドログラム)の表示
      cex=1, #樹形図(デンドログラム)の表示
      dev.off())
    
```

①出力は、hoge7.pngという名前のPNGファイル。②サイズは、700×400ピクセル。これは論文の図としても使えるレベル。

クラスタリング

7. サンプルデータ41のリアルデータ(sample blekhman 36.txt)の場合:

Blekhman et al., Genome Res., 2010の 20,689 genes×36 samplesのカウントデータです。

```
in_f <- "sample_blekhman_36.txt"
out_f <- "hoge7.png"
param_fig <- c(700, 400)
```

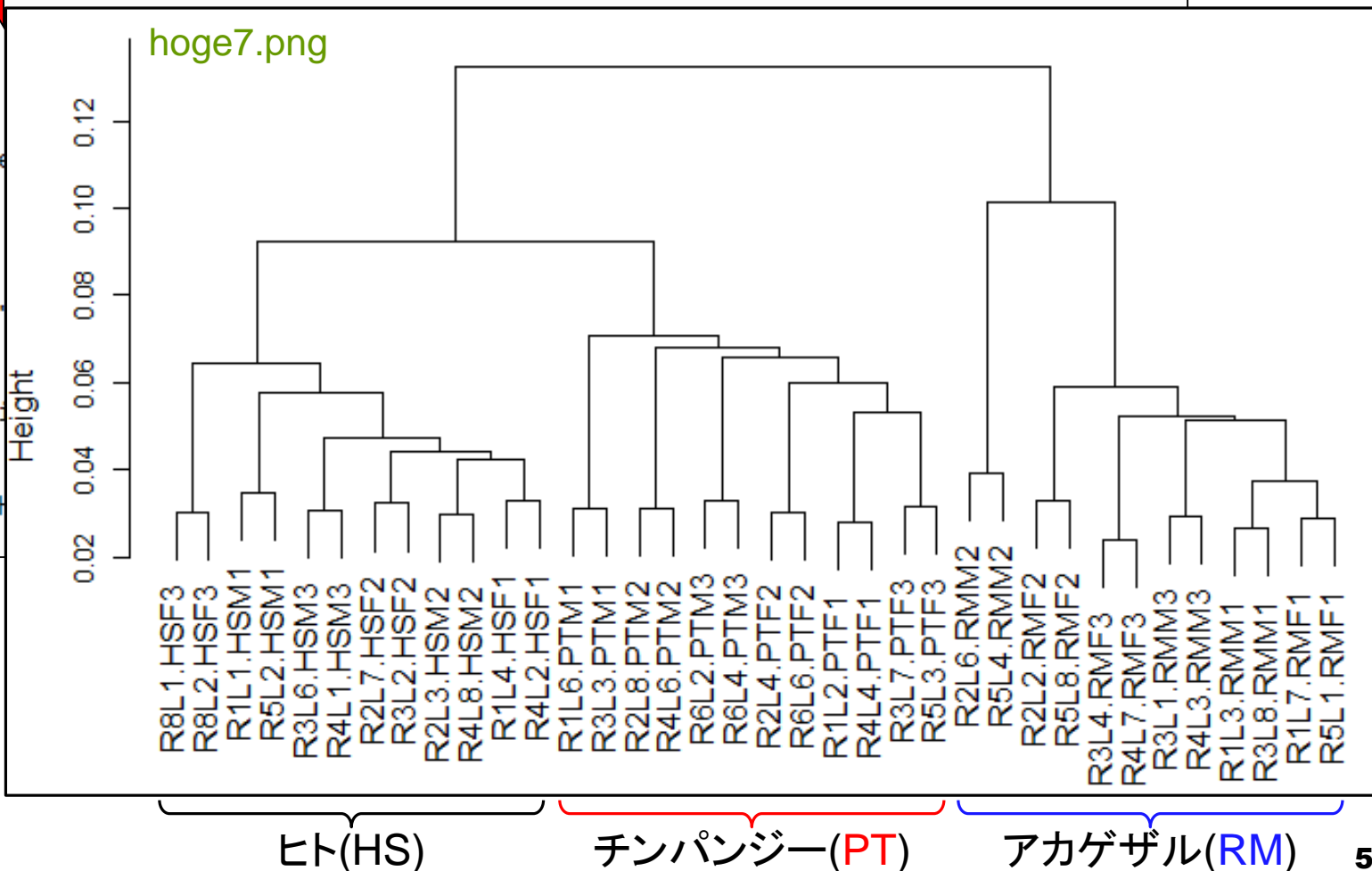
#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納
#ファイル出力時の横幅と縦幅を指定(単位はピクセル)

```
#必要なパッケージをロード
library(TCC)

#入力ファイルの読み込み
data <- read.table(in_f, header=TRUE, as.is=TRUE)
dim(data)

#本番
out <- clusterSample(data, hclust.method="ward.D2")

#ファイルに保存
png(out_f, pointsize=13, width=700, height=400,
    par(mar=c(0, 4, 1, 0)))
plot(out, sub="", xlab="", ylab="Height",
    cex=1.3, main="", ylab="Height",
    dev.off())
```



①出力は、hoge7.pngという名前のPNGファイル。②サイズは、700×400ピクセル。これは論文の図としても使えるレベル。③実際我々の論文中でも使っている。

クラスタリング

7. サンプルデータ41のリアルデータ(sample blekhman 36.txt)の場合:

Blekhman et al., *Genome Res.*, 2010の 20,689 genes×36 samplesのカウントデータです。

```

in_f <- "sample_blekhman_36.txt"
out_f <- "hoge7.png"
param_fig <- c(700, 400)

```

#入力ファイル名を指定してin_fに格納
 #出力ファイル名を指定してout_fに格納
 #ファイル出力時の横幅と縦幅を指定(単位はピクセル)

```

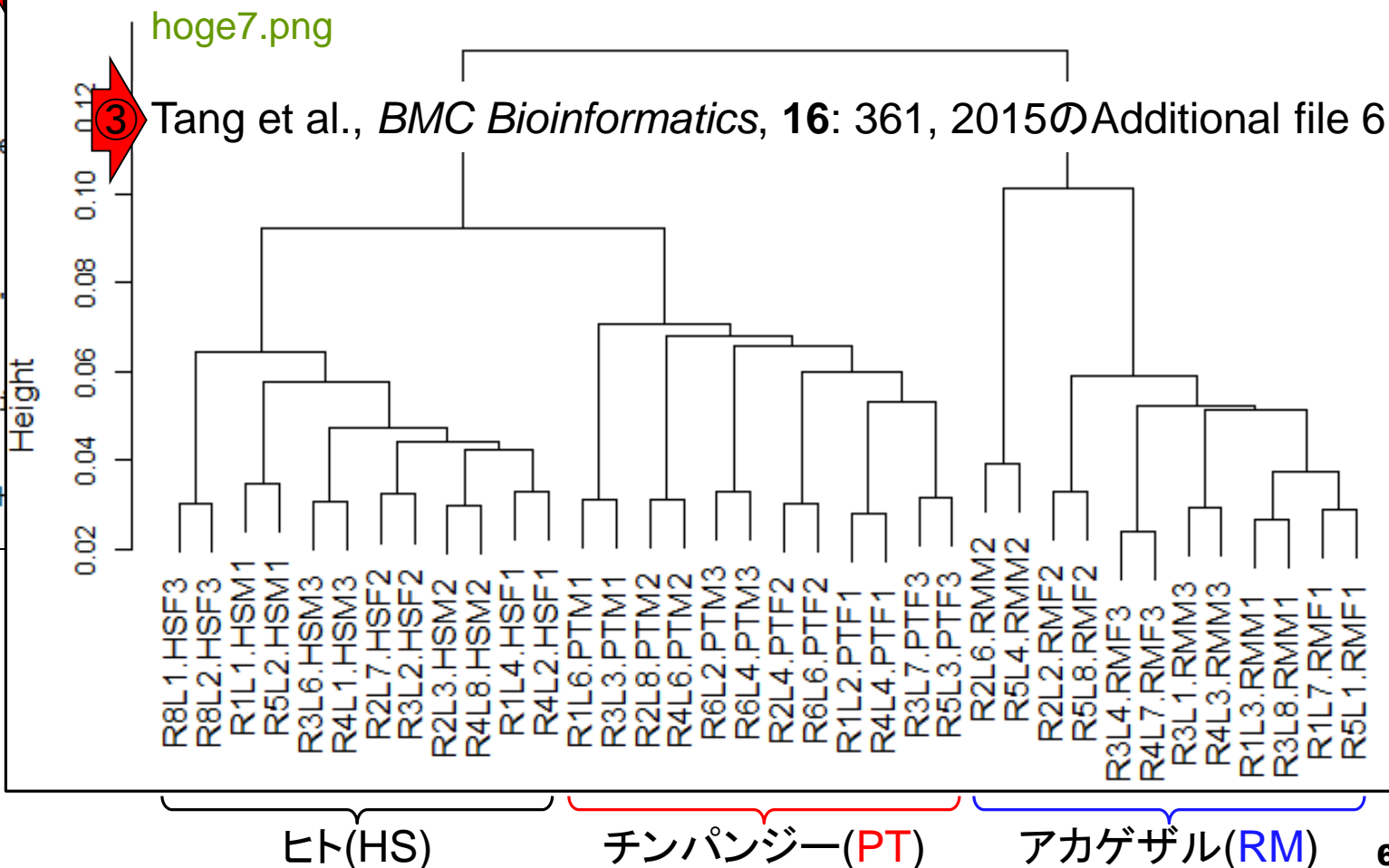
#必要なパッケージをロード
library(TCC)

#入力ファイルの読み込み
data <- read.table(in_f, header=T, as.is=T)
dim(data)

#本番
out <- clusterSample(data, hclust.method="ward.D2")

#ファイルに保存
png(out_f, pointsize=13, width=700, height=400,
    par(mar=c(0, 4, 1, 0)))
plot(out, sub="", xlab="", ylab="Height",
    cex=1.3, main="", dev.off())

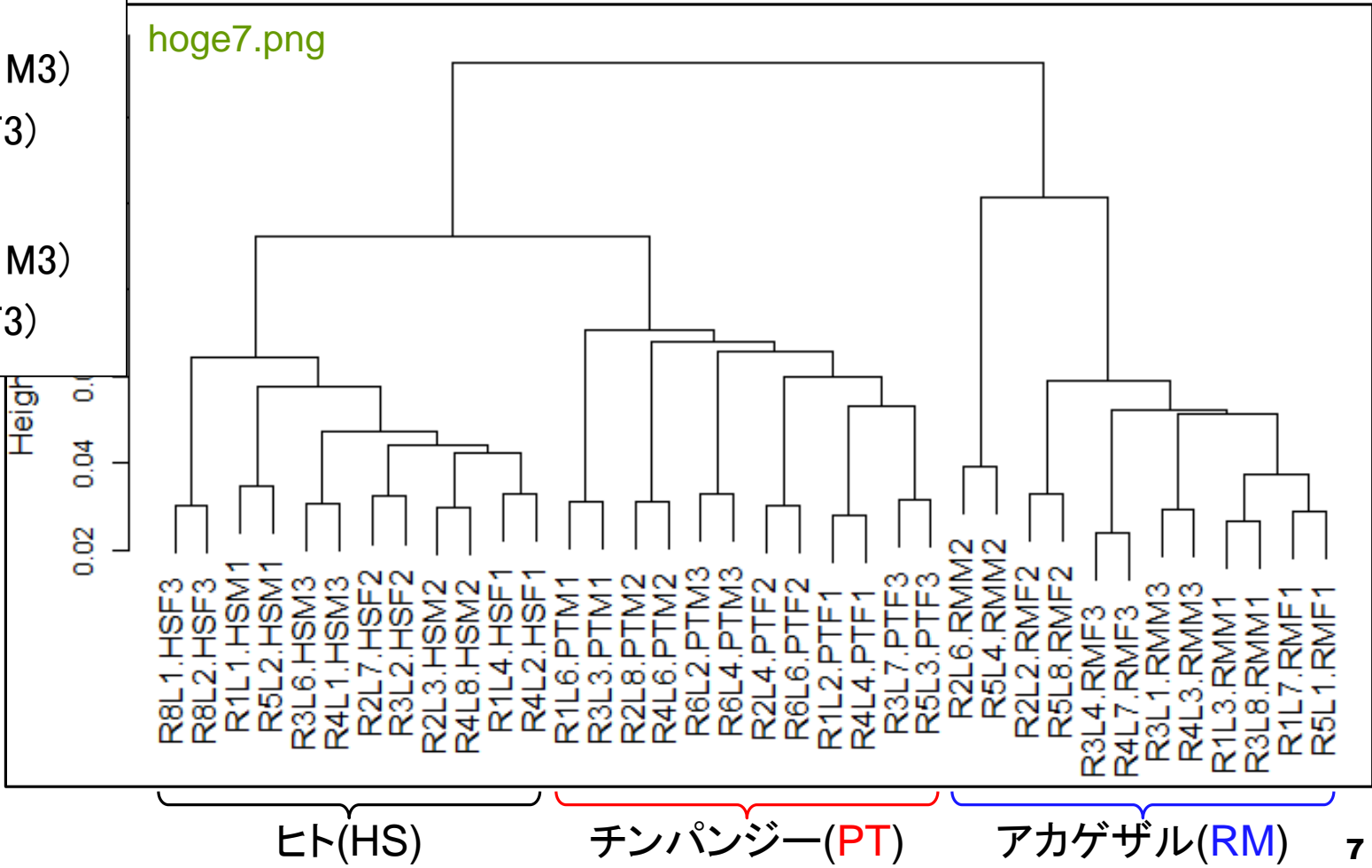
```



入力ファイルは20,689遺伝子 × 36サンプルのカウントデータファイル。ヒト(HS)、チンパンジー(PT)、アカゲザル(RM)の3生物種の肝臓(Liver)データ。各12サンプル。

実験デザイン

- ヒト(HS)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- チンパンジー(PT)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- アカゲザル(RM)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)



6個体 × 2反復

例えば、①アカゲザル(RM)の個体数は6。内訳はオス3匹とメス3匹。各個体につき2反復(technical replicatesは2)とっているので、6個体 × 2反復の計12サンプル。

ヒト(HS)

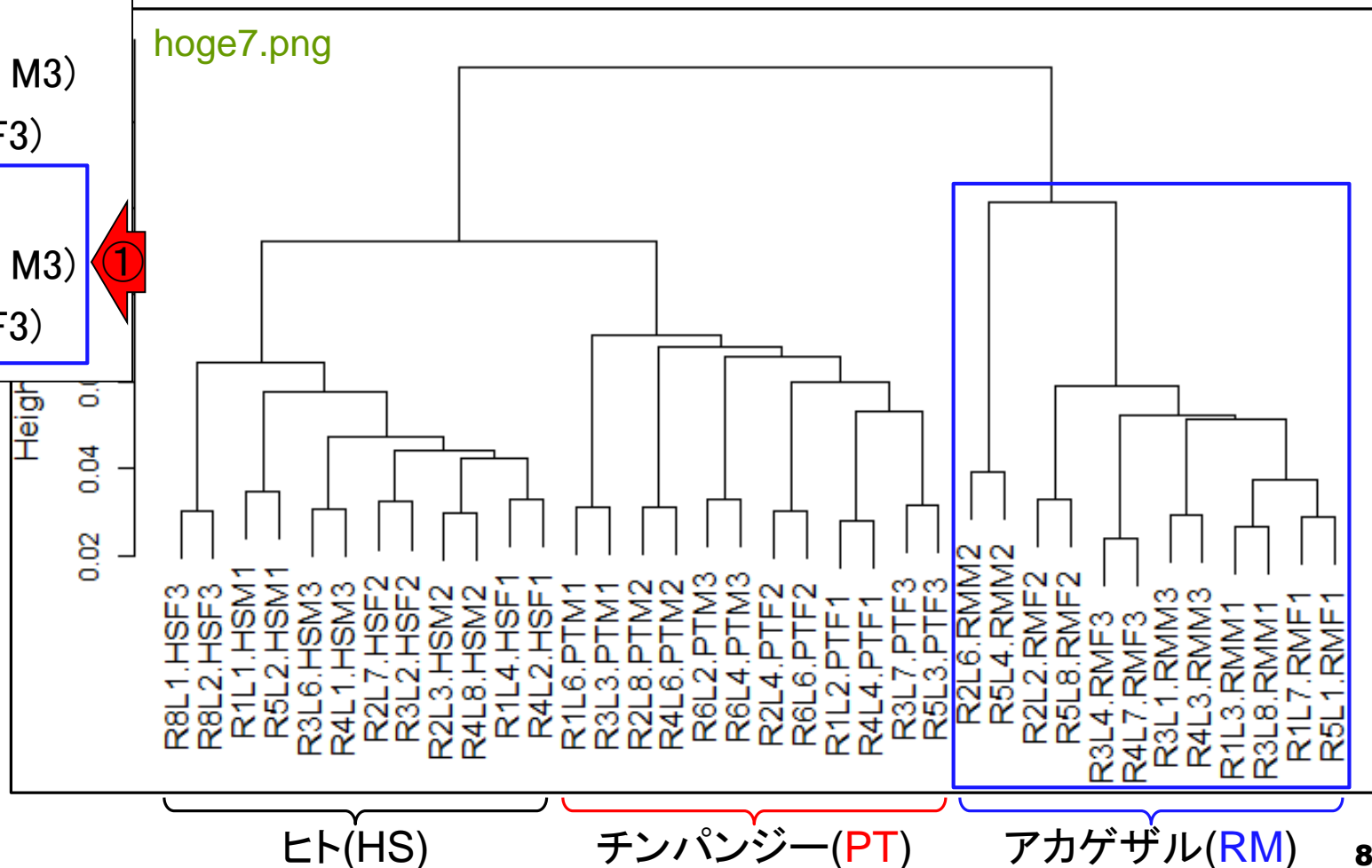
- オス3匹(M1, M2, M3)
- メス3匹(F1, F2, F3)

チンパンジー(PT)

- オス3匹(M1, M2, M3)
- メス3匹(F1, F2, F3)

アカゲザル(RM)

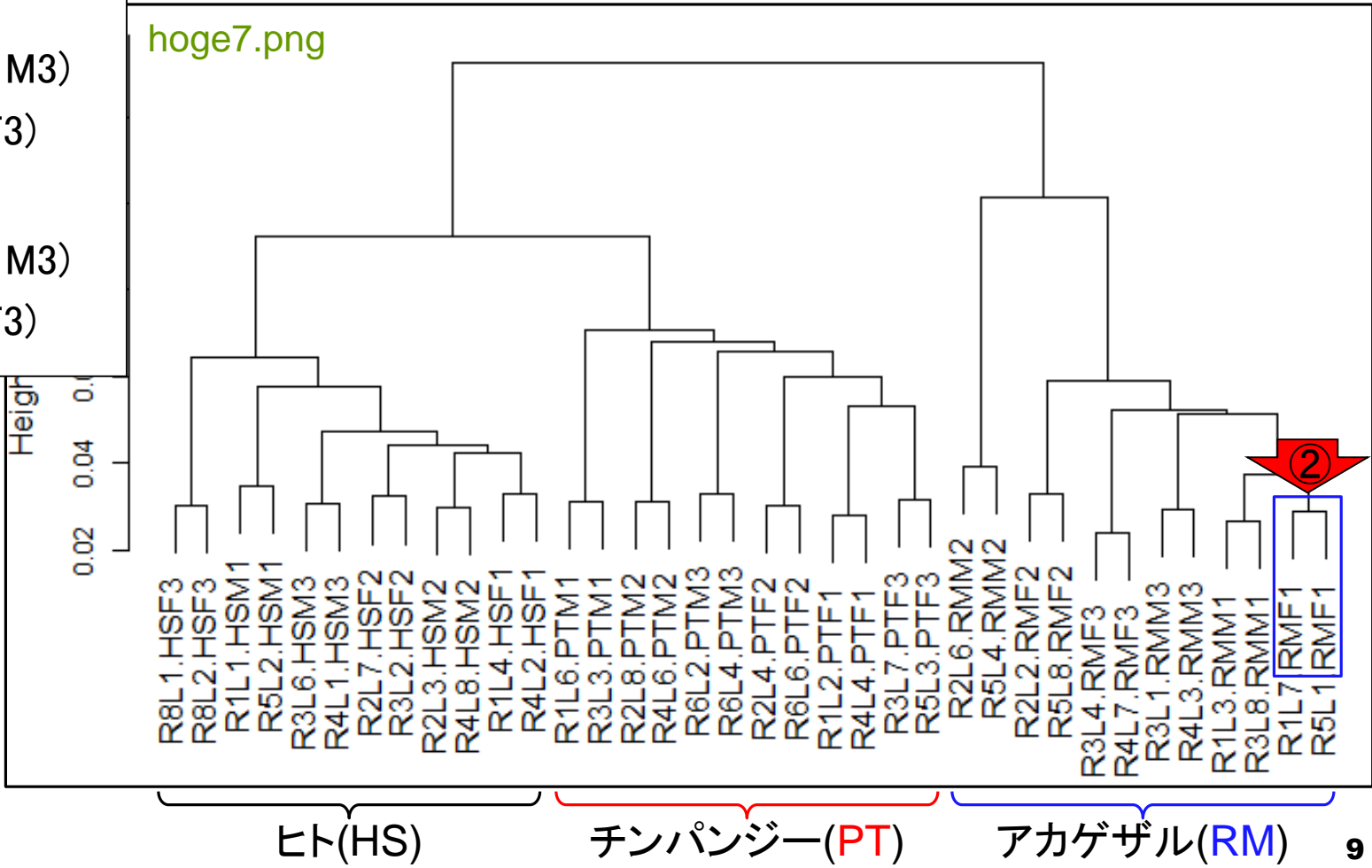
- オス3匹(M1, M2, M3)
- メス3匹(F1, F2, F3)



technical replicates

①アカゲザルのメス1個体(RMF1)の、②デンドログラム上の位置。同一個体の反復データ (technical replicates) で末端のクラスターを形成していることが分かる。これは technical replicates 同士の類似度が非常に高いことを意味します。

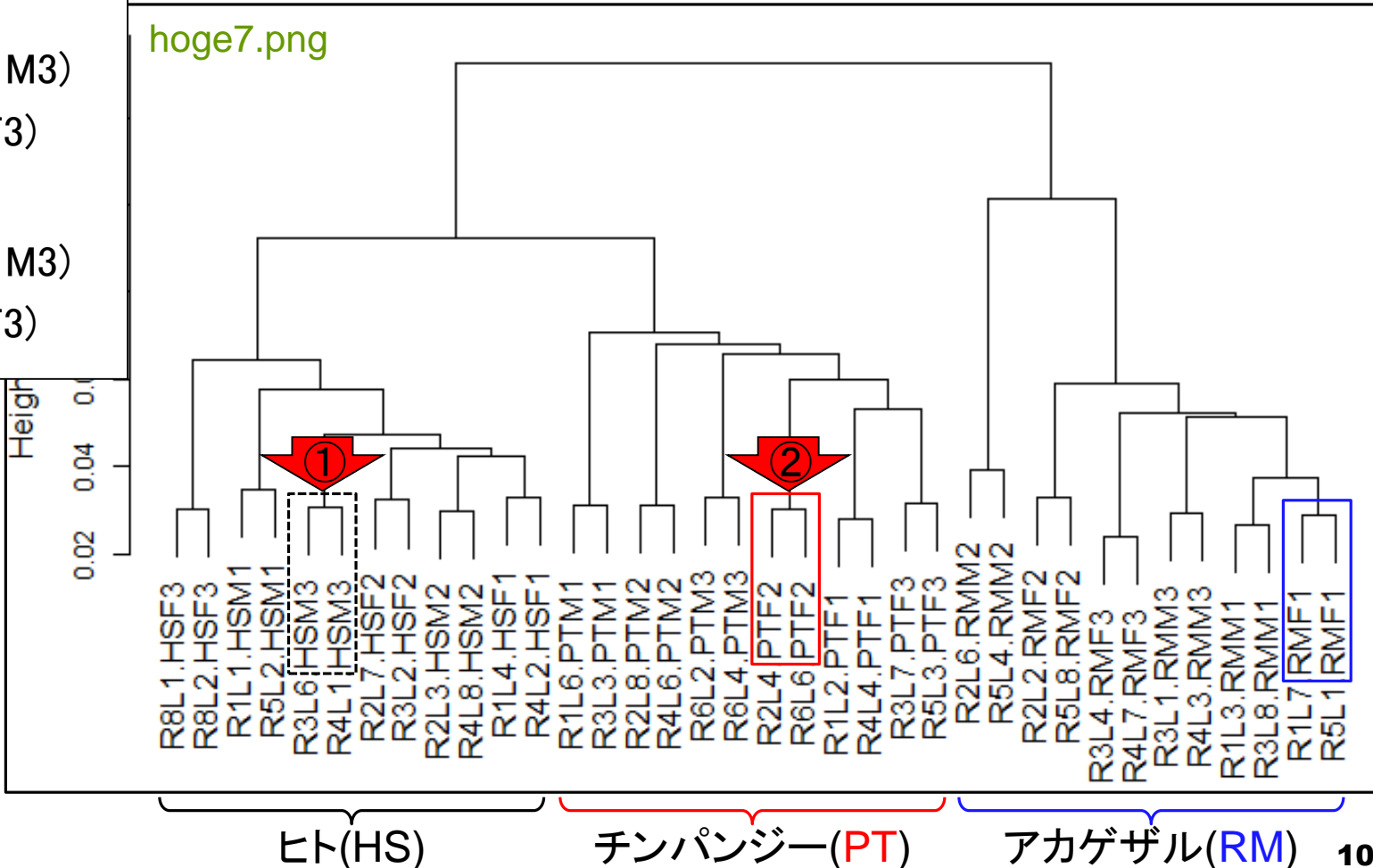
- ヒト(HS)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- チンパンジー(PT)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- アカゲザル(RM)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)



HSM3やPTF2も

他の例として、①ヒトのオス(HSM3)と、②チンパンジーのメス(PTF2)も同様の結果です。全個体についてそのようになっており、妥当ですね。

- ヒト(HS)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- チンパンジー(PT)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- アカゲザル(RM)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)



統計的手法

統計的手法で2群間比較(例えばMales vs. Females)をする目的は、同一群内の別個体(biological replicates)のばらつきの程度を見積もっておき(モデル構築)、比較する2群間で発現に変動がないという前提(帰無仮説)からどれだけ離れているのかをp値で評価することである。p値が低ければ低いほど「発現変動していない(帰無仮説に従う)」とは考えにくく、帰無仮説を棄却して「発現変動している(DEGである)」と判定することになる

■ ヒト(HS)

- オス3匹(M1, M2, M3)
- メス3匹(F1, F2, F3)

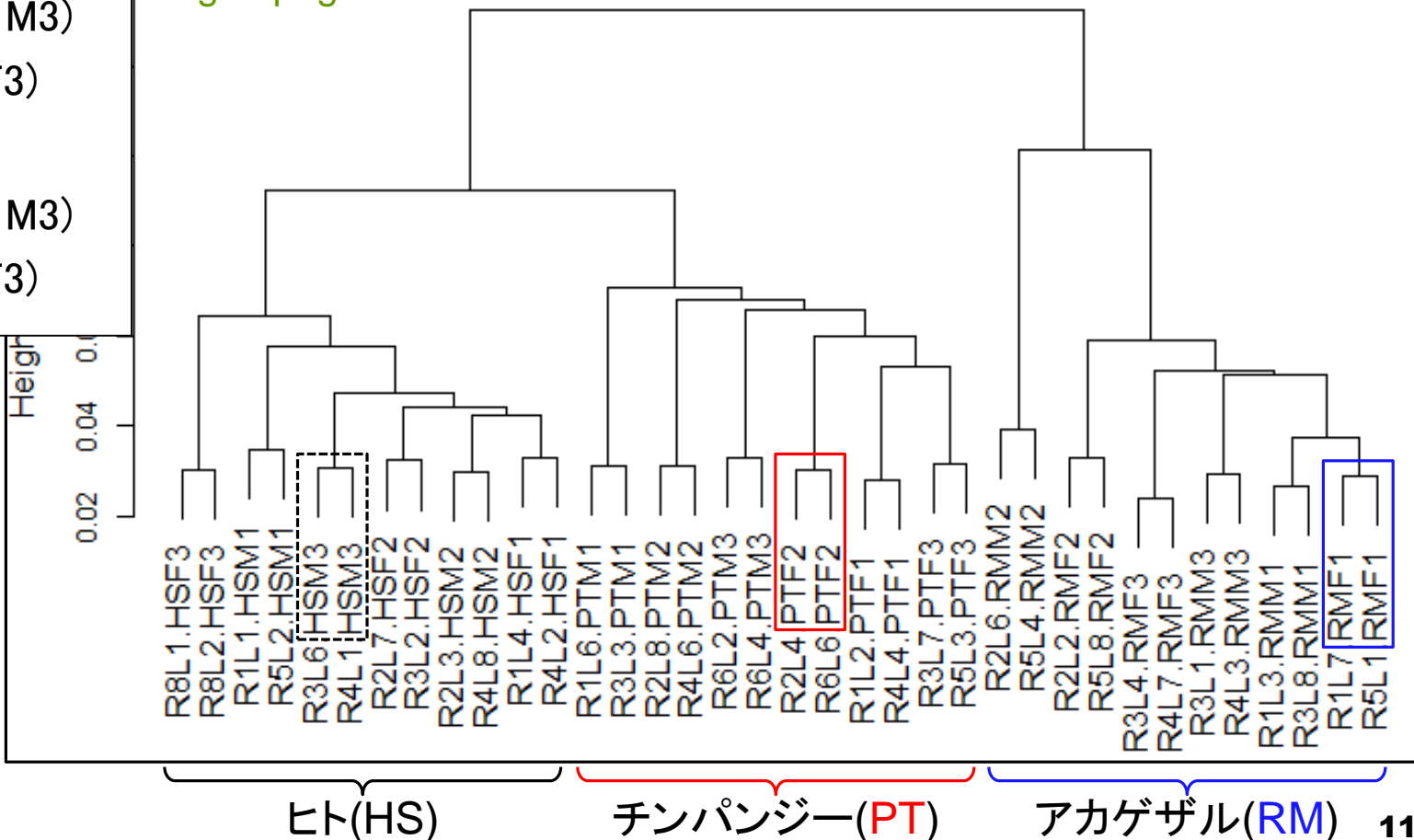
■ チンパンジー(PT)

- オス3匹(M1, M2, M3)
- メス3匹(F1, F2, F3)

■ アカゲザル(RM)

- オス3匹(M1, M2, M3)
- メス3匹(F1, F2, F3)

hoge7.png



Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?! カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

サブセット抽出と整形

①サンプルデータの、②例題42。統計的手法の多くは、biological replicatesのデータを前提としている。technical replicatesのデータをマージ(merge; collapseともいうらしい)したものを作成。③出力ファイルはsample_blekhman_18.txt。サンプル名部分は必要最小限の情報のみになっている。見るだけ。やらない。

- ・ (削除予定)個別パッケージのインストール (last modified 2015/02/20)
- ・ 基本的な利用法 (last modified 2015/04/03)
- ・ サンプルデータ ① (last modified 2015/06/15) NEW
- ・ バイオインフォマティクス人材育成カリキュラム(次世代シーケンサ) | NGSハンズオン
- ・ バイオインフォマティクス人材育成カリキュラム(次世代シーケンサ) | 演習
- ・ 書籍
- ・ 書籍
- ・ 書籍

サンプルデータ NEW

1. ② 42. [Blekhman et al., Genome Res., 2010](#)のリアルカウントデータです。
1つ前の例題41とは違って、technical replicatesの2列分のデータは足して1列分のデータとしています。20,689 genes×18 samplesのカウントデータ(sample_blekhman_18.txt)です。

```
#in_f <- "http://genome.cshlp.org/content/suppl/2009/12/16/gr.099226.109.DC1/suppTable1.xls"#入力ファイル名を指定してin_fに格納
in_f <- "suppTable1.xls" #出力ファイル名を指定してout_fに格納
out_f <- "sample_blekhman_18.txt"

#入力ファイルの読み込み
hoge <- read.table(in_f, header=TRUE, row.names=1, sep="\t", quote="")#in_fで指定したファイルの読み込み
dim(hoge) #行数と列数を表示

#サブセットの取得
data <- cbind( #必要な列名を取得したい列の順番で結合した結果をdataに格納
  hoge$R1L4.HSF1 + hoge$R4L2.HSF1, hoge$R2L7.HSF2 + hoge$R3L2.HSF2, hoge$R8L1.HSF3 + hoge$R8L2.HSF3,
  hoge$R1L1.HSM1 + hoge$R5L2.HSM1, hoge$R2L3.HSM2 + hoge$R4L8.HSM2, hoge$R3L6.HSM3 + hoge$R4L1.HSM3,
  hoge$R1L2.PTF1 + hoge$R4L4.PTF1, hoge$R2L4.PTF2 + hoge$R6L6.PTF2, hoge$R3L7.PTF3 + hoge$R5L3.PTF3,
  hoge$R1L6.PTM1 + hoge$R3L3.PTM1, hoge$R2L8.PTM2 + hoge$R4L6.PTM2, hoge$R6L2.PTM3 + hoge$R6L4.PTM3,
  hoge$R1L7.RMF1 + hoge$R5L1.RMF1, hoge$R2L2.RMF2 + hoge$R5L8.RMF2, hoge$R3L4.RMF3 + hoge$R4L7.RMF3,
  hoge$R1L3.RMM1 + hoge$R3L8.RMM1, hoge$R2L6.RMM2 + hoge$R5L4.RMM2, hoge$R3L1.RMM3 + hoge$R4L3.RMM3)
colnames(data) <- c( #列名を付加
  "HSF1", "HSF2", "HSF3", "HSM1", "HSM2", "HSM3",
  "PTF1", "PTF2", "PTF3", "PTM1", "PTM2", "PTM3",
  "RMF1", "RMF2", "RMF3", "RMM1", "RMM2", "RMM3")
rownames(data) <- rownames(hoge) #行名を付加
dim(data) #行数と列数を表示
```

出力ファイル

出力ファイルは、20,689遺伝子×18サンプルの biological replicatesのみからなる、3生物種間比較用カウントデータ。ヒト(*Homo sapiens*; HS)、チンパンジー(*Pan troglodytes*; PT)、アカゲザル(*Rhesus macaque*; RM)。生物種ごとにメス3匹、オス3匹。雄雌を考慮しなければbiological replicates (生物学的な反復)は6

20,689 genes

	ヒト (<i>Homo sapiens</i> ; HS)						チンパンジー (<i>Pan troglodytes</i> ; PT)						アカゲザル (<i>Rhesus macaque</i> ; RM)					
	メス(Female)			オス(Male)			メス			オス			メス			オス		
	HSF1	HSF2	HSF3	HSM1	HSM2	HSM3	PTF1	PTF2	PTF3	PTM1	PTM2	PTM3	RMF1	RMF2	RMF3	RMM1	RMM2	RMM3
ENSG000000000003	329	300	168	121	421	359	574	429	386	409	685	428	511	464	480	424	1348	705
ENSG000000000005	0	0	0	0	1	0	1	4	1	0	1	1	0	1	2	2	0	0
ENSG000000000419	81	61	56	39	78	62	100	66	65	59	58	93	67	72	57	49	82	90
ENSG000000000457	91	62	76	114	73	95	131	229	87	274	239	149	89	69	118	117	114	163
ENSG000000000460	6	17	12	15	7	17	8	8	5	12	7	10	4	4	10	7	3	4
ENSG000000000938	44	65	210	73	43	65	84	104	76	198	31	58	73	28	54	80	34	72
ENSG000000000971	4765	7225	3405	3600	6383	5546	5382	8331	4335	2568	5019	2653	13566	9964	18247	14236	5196	11834
ENSG000000001036	297	251	189	200	234	249	305	301	313	254	151	331	292	106	379	201	88	140
ENSG000000001084	630	737	306	336	984	459	417	328	885	298	569	218	1062	786	1110	873	664	1752
ENSG000000001167	36	30	36	29	33	28	63	80	25	69	74	41	62	34	108	97	35	61
ENSG000000001460	3	1	5	1	4	2	0	1	1	1	1	3	1	1	1	0	1	3
ENSG000000001461	49	37	34	28	62	32	75	69	40	90	69	60	210	92	176	247	81	117
ENSG000000001487	117	93	88	80	131	110	125	98	75	108	130	131	138	95	187	137	158	172

クラスタリング

- ・解析 | [発現量推定\(トランスクリプトーム配列を利用\)](#) (last modified 2014/07/09)
- ・解析 | [クラスタリング | について](#) (last modified 2014/02/05)
- ・解析 | [クラスタリング | サンプル間 | hclust](#) (last modified 2015/02/26) **NEW**
- ・解析 | [クラスタリング | サンプル間 | TCC\(Sun_2013\)](#) (last modified 2015/03/02) **NEW**
- ・解析 | [クラスタリング | 遺伝子間 | MBCluster.Seq\(SiRNA\)](#) (last modified 2014/02/05)

解析 | クラスタリング | サンプル間 | TCC(Sun_2013) **NEW**

TCCパッケージを用いてサンプル間クラスタリングを行うやり方を示します。clusterSample関数を利用した頑健なクラスタリング結果を返します。
 「ファイル名」を「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し、以下をコピペ

1. 59. ② 8. サンプルデータ42のリアルデータ(sample_blekhman_18.txt)の場合:

[Blekhman et al., Genome Res., 2010](#)の 20,689 genes×18 samplesのカウントデータです。

Neyret-
ンゲ

in_f
out_f
param

#必要
libra

#入力
data
dim(d

#本番
out <

```

in_f <- "sample_blekhman_18.txt" #入力ファイル名を指定してin_fに格納
out_f <- "hoge8.png" #出力ファイル名を指定してout_fに格納
param_fig <- c(700, 400) #ファイル出力時の横幅と縦幅を指定(単位はピクセル)

#必要なパッケージをロード
library(TCC) #パッケージの読み込み

#入力ファイルの読み込み
data <- read.table(in_f, header=TRUE, row.names=1, sep="\t", quote="")#in_fで指定したファイル
dim(data) #オブジェクトdataの行数と列数を表示

#本番
out <- clusterSample(data, dist.method="spearman",#クラスタリング実行結果をoutに格納
                    hclust.method="average", unique.pattern=TRUE)#クラスタリング実行結果をoutに格納

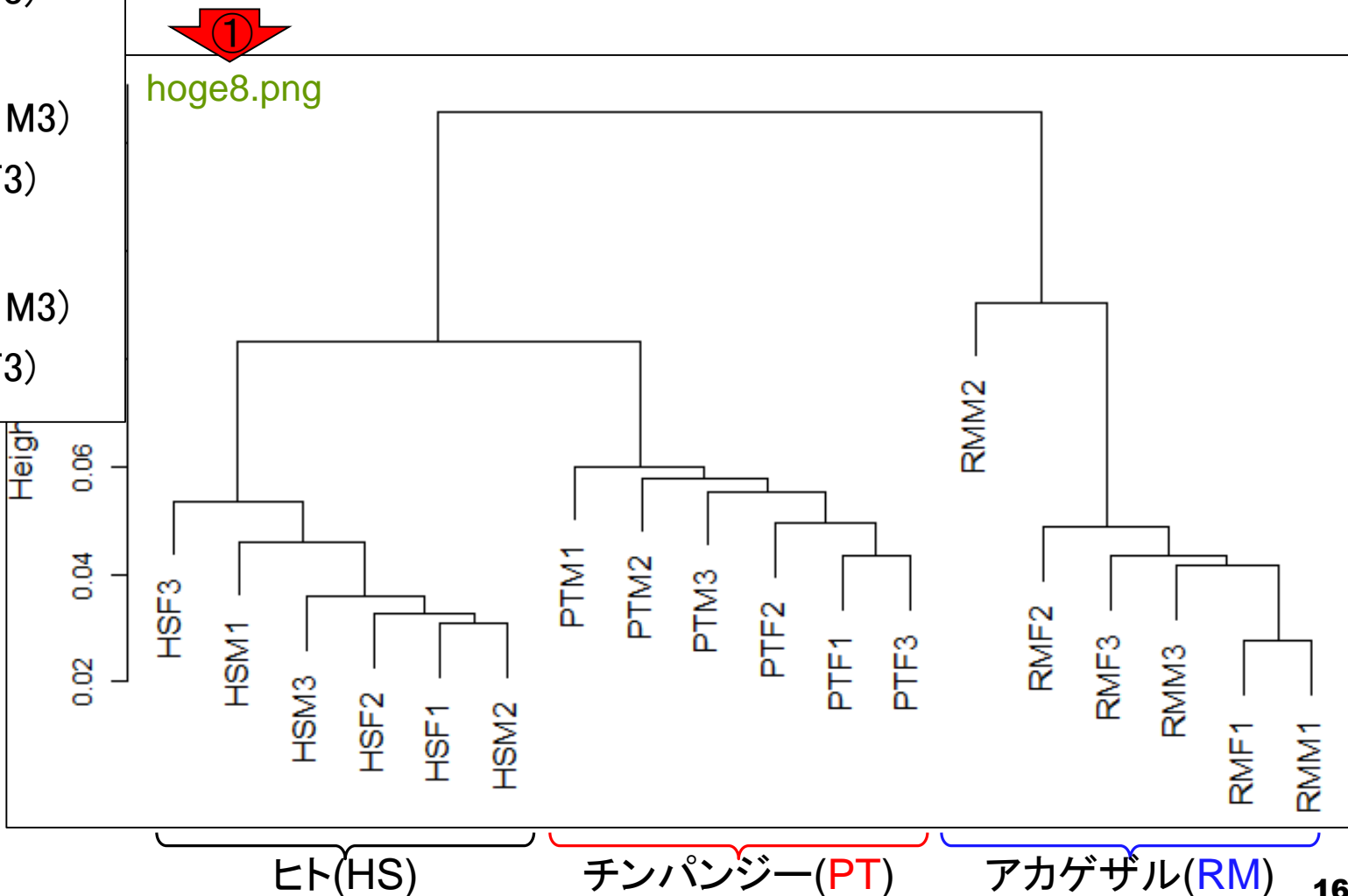
#ファイルに保存
png(out_f, pointsize=13, width=param_fig[1], height=param_fig[2])#出力ファイルの各種パラメー
par(mar=c(0, 4, 1, 0)) #下、左、上、右の順で余白(行)を指定
plot(out, sub="", xlab="", yaxp=1, 2, #樹形図(デンドログラム)の表示

```

結果の解釈

①コピペ実行結果ファイル(hoge8.png)。これは肝臓の発現データでクラスタリングした結果。全体を生物種間比較という観点で眺める。

- ヒト(HS)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- チンパンジー(PT)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- アカゲザル(RM)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)



①の部分で2つのグループに分けると…、②ヒト(HS)とチンパンジー(PT)はよく似ている。

HSとPTは似てる

■ ヒト(HS)

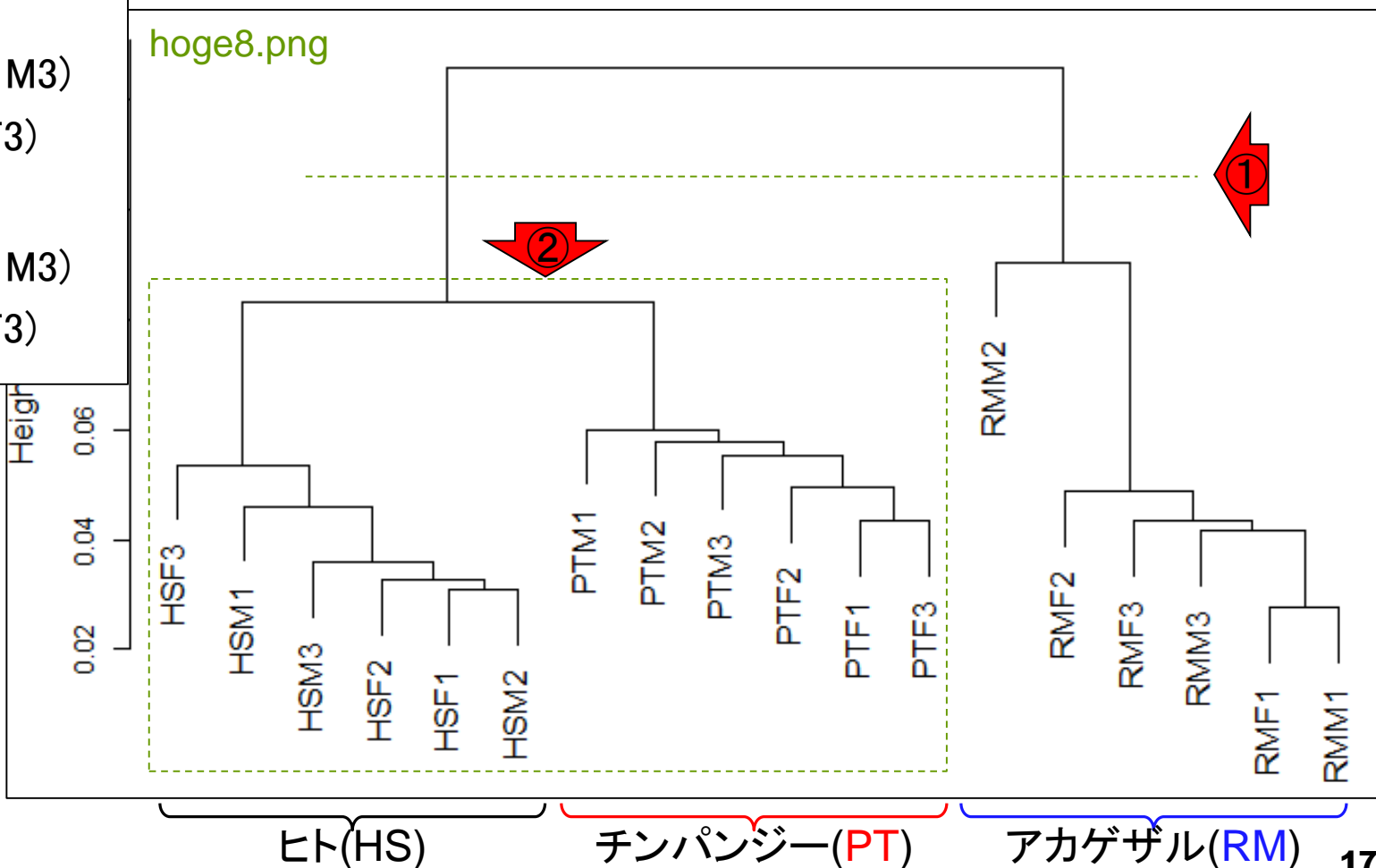
- オス3匹(M1, M2, M3)
- メス3匹(F1, F2, F3)

■ チンパンジー(PT)

- オス3匹(M1, M2, M3)
- メス3匹(F1, F2, F3)

■ アカゲザル(RM)

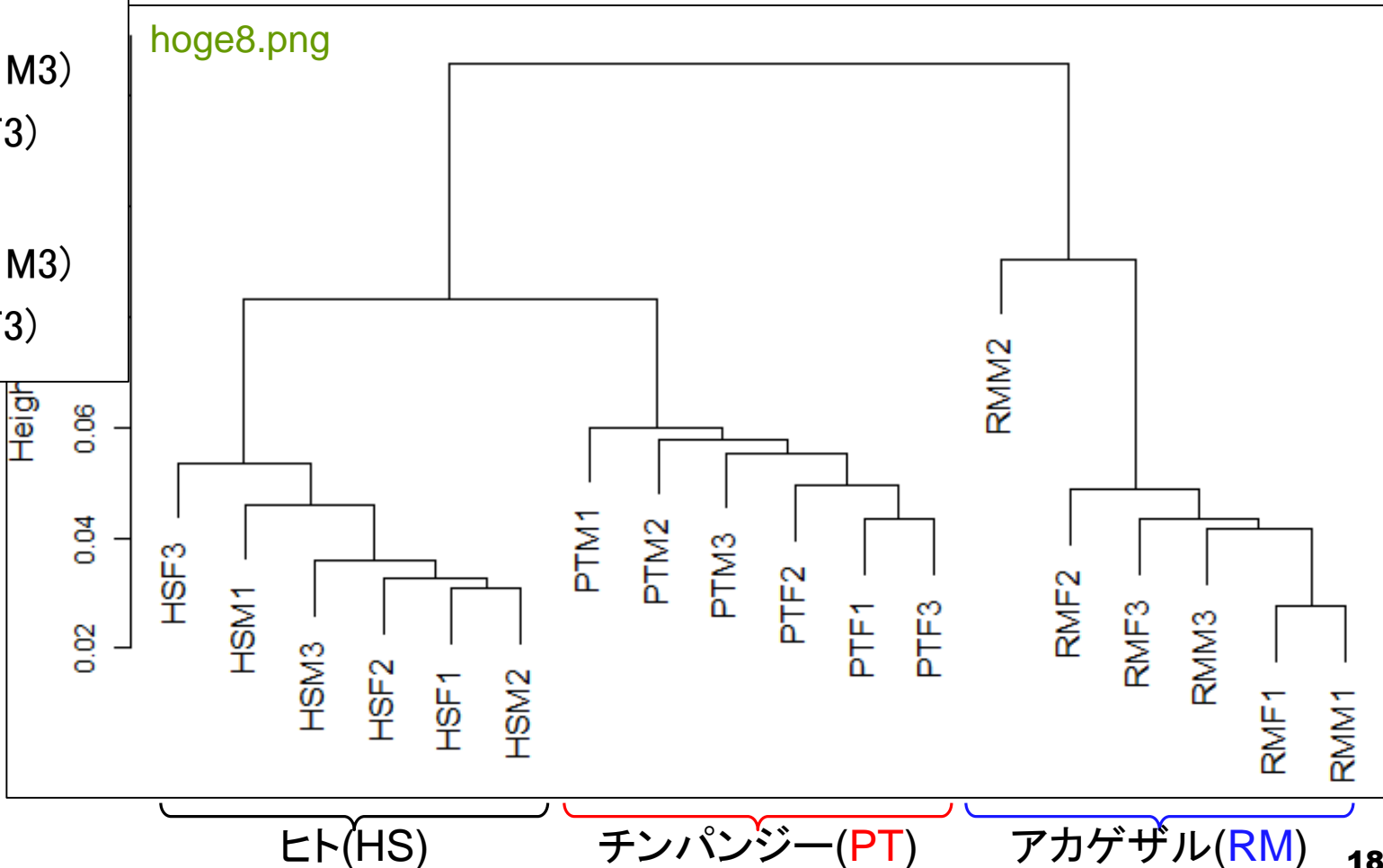
- オス3匹(M1, M2, M3)
- メス3匹(F1, F2, F3)



DEG検出結果の予想

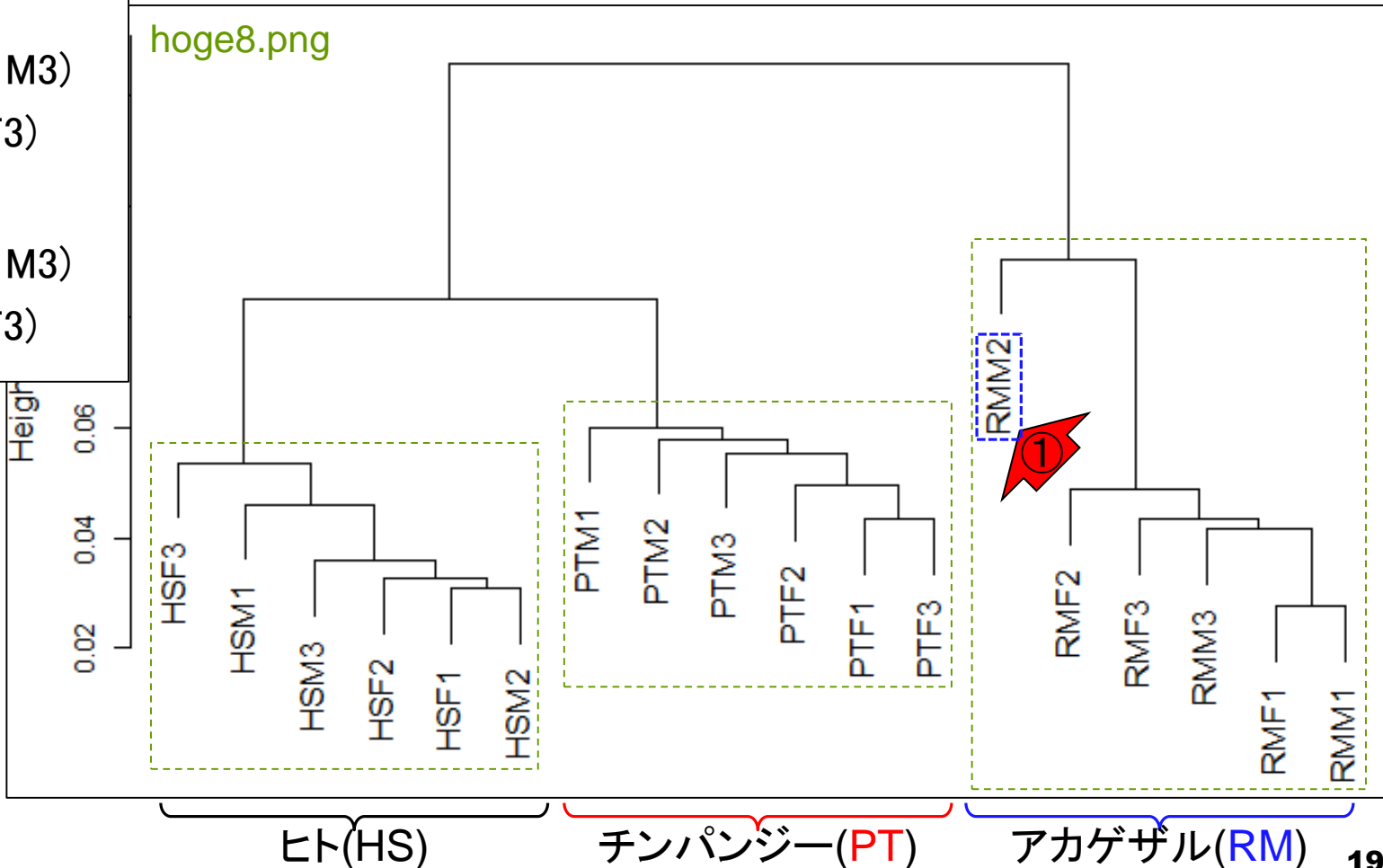
2群間比較(発現変動遺伝子検出; DEG検出)を行うと、「HS vs. RMで得られるDEG数」のほうが「HS vs. PTで得られるDEG数」よりも多そう。

- ヒト(HS)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- チンパンジー(PT)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- アカゲザル(RM)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)



生物種内でクラスター形成

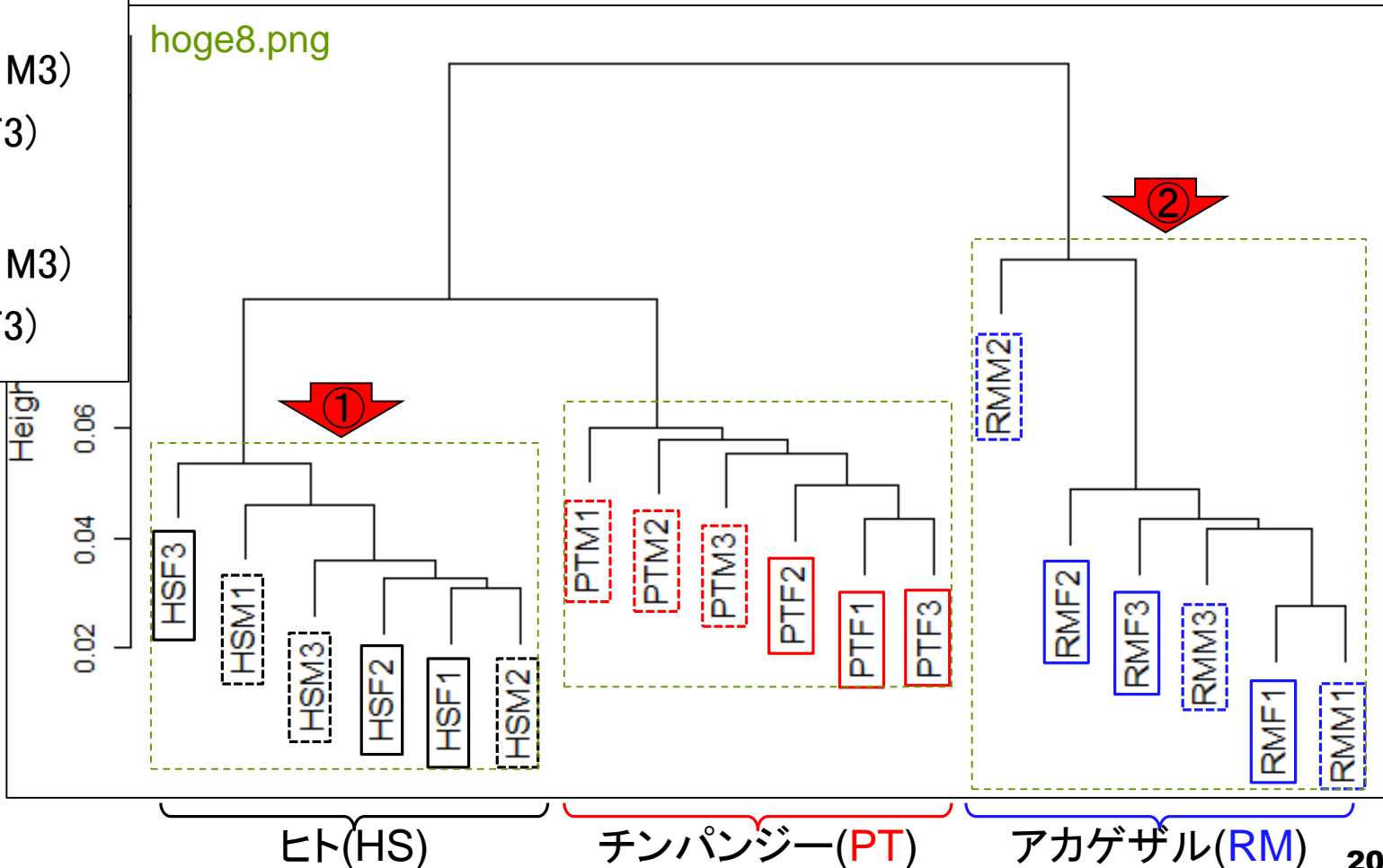
- ヒト(HS)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- チンパンジー(PT)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- アカゲザル(RM)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)



雌雄差はなさそう

①ヒト(HS)と②アカゲザル(RM)は、メスとオスのサンプルが入り混じっている。これらの生物種内で、「メス群 vs. オス群」の2群間比較を行ってもDEGはほとんど検出されないだろう

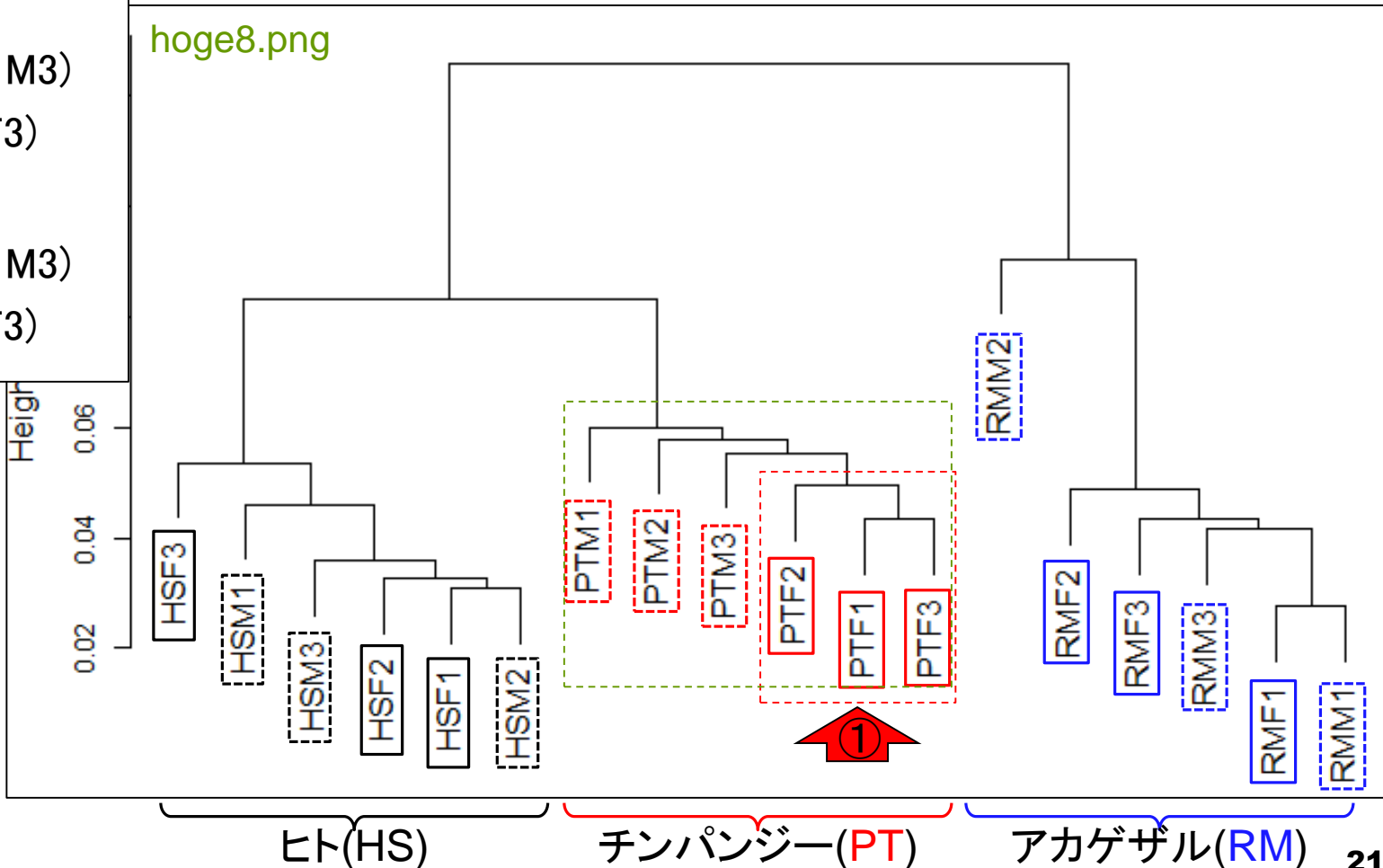
- ヒト(HS)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- チンパンジー(PT)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- アカゲザル(RM)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)



チンパンジー(PT)に限っていえば、①メス3匹がクラスターを形成しているので、「メス群 vs. オス群」の2群間比較結果として、多少なりともDEGが検出されるだろう

結果の解釈

- ヒト(HS)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- チンパンジー(PT)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- アカゲザル(RM)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)



樹形図とDEG数の関係

クラスタリング結果(樹形図; dendrogram)とDEG数の関係性に関する主観的な評価は、①のあたりに書いてます。このようにクラスタリング結果の解釈は往々にして主観的。

BMC Bioinformatics. 2015 Nov 4;16:361. doi: 10.1186/s12859-015-0794-7.

Evaluation of methods for differential expression analysis on multi-group RNA-seq count data.

Tang M¹, Sun J², Shimizu K³, Kadota K⁴.

Author information

Abstract

BACKGROUND: RNA-seq is a powerful tool for measuring transcriptomes, especially for identifying differentially expressed genes or transcripts (DEGs) between sample groups. A number of methods have been developed for this task, and several evaluation studies have also been reported. However, those evaluations so far have been restricted to two-group comparisons. Accumulations of comparative studies for multi-group data are also desired.

METHODS: We compare 12 pipelines available in nine R packages for detecting differential expressions (DE) from multi-group RNA-seq count data, focusing on three-group data with or without replicates. We evaluate those pipelines on the basis of both simulation data and real count data.

RESULTS: As a result, the pipelines in the TCC package performed comparably to or better than other pipelines under various simulation scenarios. TCC implements a multi-step normalization strategy (called DEGES) that internally uses functions provided by other representative packages (edgeR, DESeq2, and so on). We found considerably different numbers of identified DEGs (18.5 ~ 45.7% of all genes) among the pipelines for the same real dataset but similar distributions of the classified expression patterns. We also found that DE results can roughly be estimated by the hierarchical dendrogram of sample clustering for the raw count data.

CONCLUSION: We confirmed the DEGES-based pipelines implemented in TCC performed well in a three-group comparison as well as a two-group comparison. We recommend using the DEGES-based pipeline that internally uses edgeR (here called the EEE-E pipeline) for count data with replicates (especially for small sample sizes). For data without replicates, the DEGES-based pipeline with DESeq2 (called SSS-S) can be recommended.

PMID: 26538400 PMCID: PMC4634584 DOI: 10.1186/s12859-015-0794-7

[Indexed for MEDLINE] [Free PMC Article](#)



PMC **FREE** Full text

Save items

☆ Add to Favorites

Similar articles

TCC: an R package for comparing tag count [BMC Bioinformatics. 2013]

SARTools: A DESeq2- and EdgeR-Based R Pipeline for [PLoS One. 2016]

A comparison of per sample global scaling and per gene [PLoS One. 2017]

Review RNA-Seq differential expression analysis [PLoS One. 2017]

Review A comparison of statistical methods for detecting [Am J Bot. 2012]

[See reviews...](#)

[See all...](#)

Cited by 5 PubMed Central articles

Silhouette Scores for Arbitrary Defined Groups [Biol Proced Online. 2018]

Metastatic ability and the epithelial-mesenchymal transition [Cancer Sci. 2018]

Evaluation of logistic regression models for [BMC Bioinformatics. 2017]



樹形図とDEG数の

クラスタリング結果(樹形図)を眺めて、興味あるグループ間の関係性(特にDEG検出結果)を客観的に評価する指標として、シルエットスコア(Silhouette score)が有用だということを示した論文。これについては後程また言及。

Biol Proced Online. 2018 Mar 1;20:5. doi: 10.1186/s12575-018-0067-8. eCollection 2018.

Silhouette Scores for Arbitrary Defined Groups in Gene Expression Data and Insights into Differential Expression Results.

Zhao S¹, Sun J¹, Shimizu K¹, Kadota K¹.

Author information

Abstract

BACKGROUND: Hierarchical Sample clustering (HSC) is widely performed to examine associations within expression data obtained from microarrays and RNA sequencing (RNA-seq). Researchers have investigated the HSC results with several possible criteria for grouping (e.g., sex, age, and disease types). However, the evaluation of arbitrary defined groups still counts in subjective visual inspection.

RESULTS: To objectively evaluate the degree of separation between groups of interest in the HSC dendrogram, we propose to use *Silhouette* scores. *Silhouettes* was originally developed as a graphical aid for the validation of data clusters. It provides a measure of how well a sample is classified when it was assigned to a cluster by according to both the tightness of the clusters and the separation between them. It ranges from 1.0 to -1.0, and a larger value for the average silhouette (AS) over all samples to be analyzed indicates a higher degree of *cluster* separation. The basic idea to use an AS is to replace the term *cluster* by *group* when calculating the scores. We investigated the validity of this score using simulated and real data designed for differential expression (DE) analysis. We found that larger (or smaller) AS values agreed well with both higher (or lower) degrees of separation between different groups and higher percentages of differentially expressed genes (P_{DEG}). We also found that the AS values were generally independent on the number of replicates (N_{rep}). Although the P_{DEG} values depended on N_{rep} , we confirmed that both AS and P_{DEG} values were close to zero when samples in the data showed an intermingled nature between the groups in the HSC dendrogram.

CONCLUSION: *Silhouettes* is useful for exploring data with predefined group labels. It would help provide both an objective evaluation of HSC dendrograms and insights into the DE results with regard to the compared groups.

KEYWORDS: Bioinformatics; Differential expression analysis; Hierarchical sample clustering; *Silhouettes*

PMID: 29507534 PMCID: PMC5831220 DOI: 10.1186/s12575-018-0067-8

Save items

★ Add to Favorites

Similar articles

How frequently do clusters occur in hierarchical clusterin [J Cheminform. 2016]

Evaluation of methods for differential expression an: [BMC Bioinformatics. 2015]

Knowledge-assisted recognition of cluster boundaries in gene [Artif Intell Med. 2005]

Silhouette scores for assessment of SNP genotype clusters. [BMC Genomics. 2005]

Review [Aiming for zero blindness]. [Nippon Ganka Gakkai Zasshi. 2015]

See reviews...

See all...

Related information

References for this PMC Article

Free in PMC

Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?!カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

ReCount

①ReCountは、②18個のカウントデータを提供しているサイト。自分で1からマッピングなどを行わずに済むので便利。technical replicatesのデータセットについては、biological replicatesにマージしたのも提供してくれている。このスクリーンショットは平成28年度NGSハンズオン講習会(2016年7月21日実施分)の講義資料作成時のものであり、古い。

The screenshot shows the ReCount website interface. A red arrow labeled '1' points to the ReCount logo, and another red arrow labeled '2' points to the main descriptive text.

ReCount
A multi-experiment resource of analysis-ready RNA-seq gene count datasets

ReCount is an online resource consisting of RNA-seq gene count datasets built using the raw data from 18 different studies. The raw sequencing data (.fastq files) were processed with [Myrna](#) to obtain tables of counts for each gene. For ease of statistical analysis, we combined each count table with sample phenotype data to form an R object of class [ExpressionSet](#). The count tables, ExpressionSets, and phenotype tables are ready to use and freely available here. By taking care of several preprocessing steps and combining many datasets into one easily-accessible website, we make finding and analyzing RNA-seq data considerably more straightforward.

All columns of the table below are sortable: clicking on the column title will alphabetize or order the column (keeping the rows properly aligned). The columns are as follows:

Study

With a few exceptions, the datasets are named for the first author of the paper from which the .fastq files were obtained. The Katz paper contained both mouse and human reads, so two separate datasets were created. The "maq" dataset was built from reads obtained from the [MicroArray Quality Control Project](#). The "modencodeworm" and "modencodefly" datasets were generated using reads from papers associated with the [modENCODE Consortium](#).

Site Map

- [Home](#)
- [News and Updates](#)
- [Getting Started with ExpressionSets](#)

Related Tools

- [Myrna: Cloud, differential gene expression](#)

Related Publications

Fraze AC, Langmead B, Leek JT. [ReCount: a multi-experiment resource of analysis-ready RNA-seq gene count datasets](#). *BMC Bioinformatics* 12:449

Fraze et al., *BMC Bioinformatics*, 12: 449, 2011

ReCount

ReCountから取得したカウントデータは、①平成28年度NGSハンズオン講習会の、②2016年7月21日実施分の講義資料(スライド36~61)でも利用しています。

H28年度 NGSハンズオン講習会カリキュラム

①

H28年度日程・講義資料・動画等

カリキュラム (PDF: 72KB)

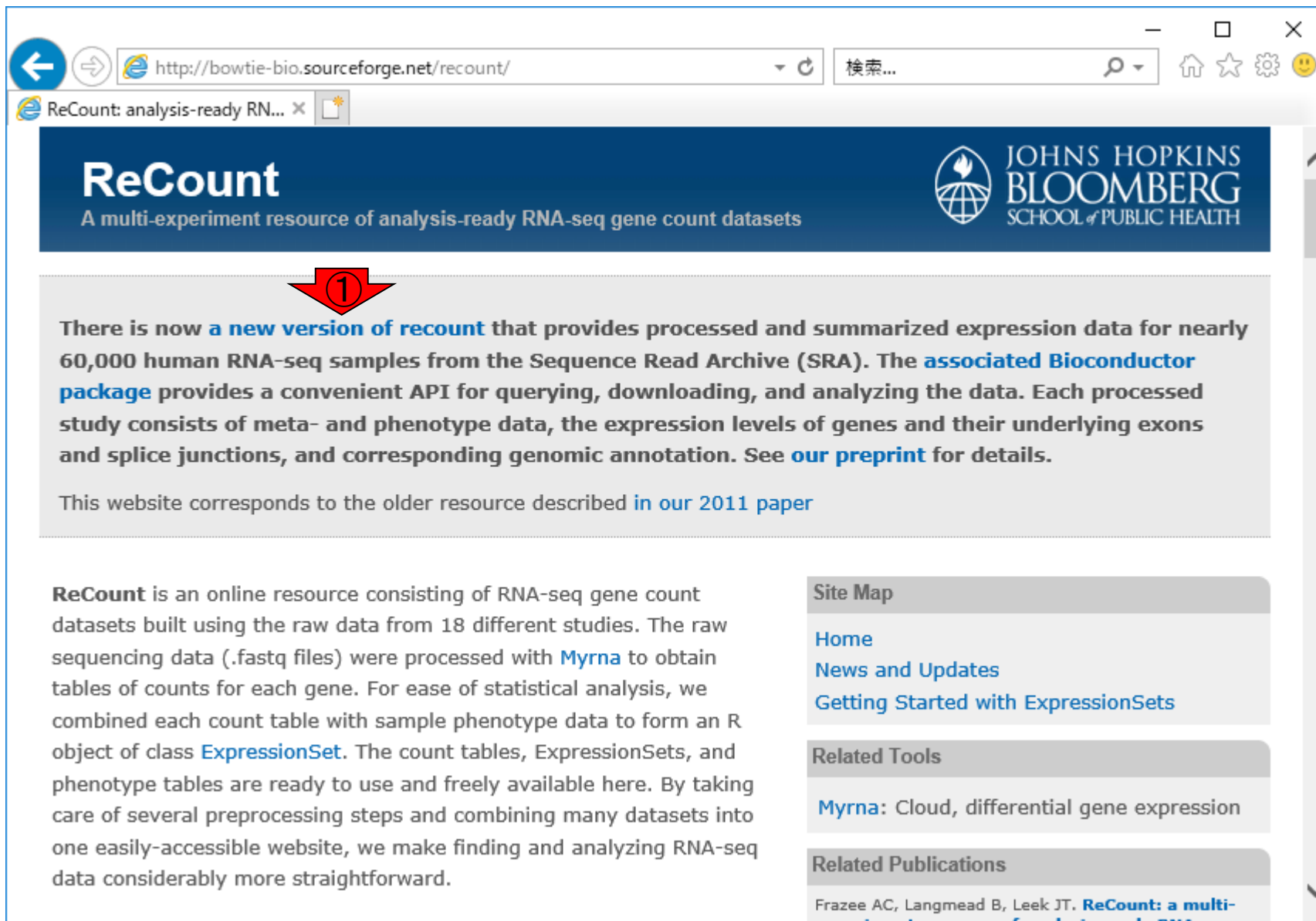
実施日	実施時間	大項目	タイトル	内容 (予定)	担当講師 (敬称略)	講義資料・動画(統合TV)
7月19日 (火)	10:30- 18:15	はじめに (講習会参加者必読) PC環境の構築	Bio-Linux8とRのインストール状況確認	<ul style="list-style-type: none"> ・ Bio-Linux8 (第2部および3部で利用するovaファイル) の導入確認 ・ 共有フォルダ設定完了確認 ・ 基本的なLinuxコマンドの習得状況確認 ・ R本体およびパッケージのインストール確認 ・ 講師指定の事前予習内容の再確認 ・ 講習会期間中に貸与されるノートPCを用いた各種動作確認 	主催・共催機関	講義資料 (PDF:4MB)
7月20日 (水)	10:30- 18:15	第1部 統計解析 (農学生命情報科学特論I)	ゲノム解析、塩基配列解析	<ul style="list-style-type: none"> ・ NGS解析手段、ウェブツール(DDBJ Pipeline)との連携 ・ k-mer解析 (k個の連続塩基に基づく各種解析) の基礎と応用 ・ 塩基ごとの出現頻度解析(k=1)、2連続塩基の出現頻度解析(k=2) ・ 塩基配列解析を行うための基本スキルの復習や作図 ・ de novoアセンブリ時のエラー補正やゲノムサイズ推定の基本的な考え方 	門田 幸二 (東京大学)	講義資料 (PDF:7.3MB) 解析データ (ZIP:2.2MB) 統合TV
7月21日 (木)	10:30- 18:15		トランスクリプトーム解析1	<ul style="list-style-type: none"> ・ カウントデータ取得以降の統計解析(RNA-seq) ・ サンプル間クラスタリング、結果の解釈 ・ 発現変動解析 (反復あり2群間比較) ・ 分布やモデル、実験デザイン ・ 反復なし2群間比較(TCC, DESeq2)、および結果の解釈 		講義資料 (PDF:6.5MB) 解析データ (ZIP:3.2MB) 統合TV

②

https://biosciencedbc.jp/human/human-resources/workshop

ReCount

2018年6月現在のReCountのウェブサイト。
①new version (i.e., recount2)があります。



The screenshot shows a web browser window with the URL <http://bowtie-bio.sourceforge.net/recount/>. The page header features the ReCount logo and the Johns Hopkins Bloomberg School of Public Health logo. A red arrow with the number 1 points to a text block that reads: "There is now a **new version of recount** that provides processed and summarized expression data for nearly 60,000 human RNA-seq samples from the Sequence Read Archive (SRA). The **associated Bioconductor package** provides a convenient API for querying, downloading, and analyzing the data. Each processed study consists of meta- and phenotype data, the expression levels of genes and their underlying exons and splice junctions, and corresponding genomic annotation. See [our preprint](#) for details." Below this, it says "This website corresponds to the older resource described [in our 2011 paper](#)". The main content area describes ReCount as an online resource of RNA-seq gene count datasets. On the right, there are sections for "Site Map" (Home, News and Updates, Getting Started with ExpressionSets), "Related Tools" (Myrna: Cloud, differential gene expression), and "Related Publications" (Frazee AC, Langmead B, Leek JT. [ReCount: a multi-experiment resource of analysis-ready RNA-seq](#)).

recount2

①recount2のウェブサイト。②原著論文。前のバージョン(ReCount)ではgeneレベルのカウントデータのみでしたが、recount2では③exonレベルや④transcriptレベルのカウントデータも利用可能なようです。

The screenshot shows the recount2 website interface. At the top, the title "recount2: analysis-ready RNA-seq gene and exon counts datasets" is displayed. Below it are navigation tabs: "Datasets", "Popular datasets", "GTEx", "TCGA", "Documentation", and "Download data with R". A red arrow labeled "1" points to the title. Another red arrow labeled "3" points to the "Documentation" tab. Below the navigation is a light blue notification box with a red arrow labeled "4" pointing to it. The notification text reads: "Transcript counts are now available thanks to the work of Fu et al, bioRxiv, 2018. Exon counts are now from disjoint exons (v2) instead of reduced ones (v1). Check the Documentation tab for further information." Below the notification is the "recount2" logo and the text "A multi-experiment resource of analysis-ready RNA-seq gene and exon count datasets". A paragraph of text describes the resource. A red arrow labeled "2" points to the end of this paragraph. At the bottom of the screenshot, the date "June 19, 2018" is visible on the left, and the citation "Collado-Torres et al., Nat Biotechnol., 35: 319-321, 2017" is on the right.

recount2

Exonレベルのカウントデータについては、以前の①reduced exons (v1)ではなく、②disjoint exons (v2)というものも提供しているようです。詳細については③Documentationを参照のこと。

The screenshot shows the recount2 website interface. At the top, there is a navigation bar with tabs for "Datasets", "Popular datasets", "GTEx", "TCGA", "Documentation", and "Download data with R". A red arrow labeled "③" points to the "Documentation" tab. Below the navigation bar, there is a light blue notification box with the text: "Transcript counts are now available thanks to the work of Fu et al, bioRxiv, 2018. Exon counts are now from disjoint exons (v2) instead of reduced ones (v1). Check the Documentation tab for further information." Two red arrows labeled "①" and "②" point to the underlined text in the notification box. Below the notification box, the main header of the website is visible, featuring the "recount2" logo and the text "A multi-experiment resource of analysis-ready RNA-seq gene and exon count datasets". The main content area contains a paragraph of text describing the resource, with a red arrow labeled "③" pointing to the word "Documentation" in the text.

recount2


前のバージョン(ReCount)では18個しかありませんでしたが、recount2では①2,041個もあるようです。この数値は大まかにカウントデータセット数に相当します。

recount2: analysis-ready RNA-seq gene and exon counts datasets

Datasets Popular datasets GTEx TCGA Documentation Download data with R

Accessing recount2 via SciServer Contribute your data

Transcript counts are now available thanks to the work of Fu et al, bioRxiv, 2018. Exon counts are now from disjoint exons (v2) instead of reduced ones (v1). Check the Documentation tab for further information.

 **recount2** A multi-experiment resource of analysis-ready RNA-seq gene and exon count datasets

recount2 is an online resource consisting of RNA-seq gene and exon counts as well as coverage bigWig files for 2041 different studies. It is the second generation of the [ReCount project](#). The raw sequencing data were processed with [Rail-RNA](#) as described in the recount2 paper and at [Nellore et al, Genome Biology, 2016](#) which created the coverage bigWig files. For ease of statistical analysis, for each study we created count tables at the gene and exon levels and extracted phenotype data, which we provide in their raw formats as well as in RangedSummarizedExperiment R objects (described in the [SummarizedExperiment](#) Bioconductor package). We also computed the mean coverage per study and provide it in a bigWig file, which can be used with the [derfinder](#) Bioconductor package to perform annotation-agnostic differential expression analysis at the expressed regions-level as described at

Search

Transcript counts are now available thanks to the work of Fu et al, bioRxiv, 2018. Exon counts are now from disjoint exons (v2) instead of reduced ones (v1). Check the Documentation tab for further information.

recount2 A multi-experiment resource of analysis-ready RNA-seq gene and exon count datasets

recount2 is an online resource consisting of RNA-seq gene and exon counts as well as coverage bigWig files for 2041 different studies. It is the second generation of the [ReCount project](#). The raw sequencing data were processed with [Rail-RNA](#) as described in the recount2 paper and at [Nellore et al, Genome Biology, 2016](#) which created the coverage bigWig files. For ease of statistical analysis, for each study we created count tables at the gene and exon levels and extracted phenotype data, which we provide in their raw formats as well as in RangedSummarizedExperiment R objects (described in the [SummarizedExperiment](#) Bioconductor package). We also computed the mean coverage per study and provide it in a bigWig file, which can be used with the [derfinder](#) Bioconductor package to perform annotation-agnostic differential expression analysis at the expressed regions-level as described at [Collado-Torres et al, Genome Research, 2017](#). The count tables, RangedSummarizeExperiment objects, phenotype tables, sample bigWigs, mean bigWigs, and file information tables are ready to use and freely available here. We also created the [recount](#) Bioconductor package which allows you to search and download the data for a specific study. By taking care of several preprocessing steps and combining many datasets into one easily-accessible website, we make finding and analyzing RNA-seq data considerably more straightforward.

Main publication

- **Collado-Torres L, Nellore A,** Kammers K, Ellis SE, Taub MA, Hansen KD, Jaffe AE, Langmead B, Leek JT. [Reproducible RNA-seq analysis using recount2](#). *Nature Biotechnology*, 2017. doi: 10.1038/nbt.3838.

Related publications

- **Nellore A,** Collado-Torres L, Jaffe AE, Alquicira-Hernández J, Wilks C, Pritt J, Morton J, Leek JT, Langmead B. [Rail-RNA: scalable analysis of RNA-seq splicing and coverage](#). *Bioinformatics*, 2017. doi: 10.1093/bioinformatics/btw575.



Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?! カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

Search

①このあたりまで移動すると、②検索窓があります。

recount2: analysis-ready R... x

- **Collado-Torres L**, Nellore A, Jaffe AE. [recount workflow: Accessing over 70,000 human RNA-seq samples with Bioconductor](#). *F1000Research*, 2017. doi: 10.12688/f1000research.12223.1.
- **Wilks C**, Gaddipati P, Nellore A, Langmead B. [Snaptron: querying splicing patterns across tens of thousands of RNA-seq samples](#). *Bioinformatics*, 2018. doi: 10.1093/bioinformatics/btx547.
- **Fu J**, Kammers K, Nellore A, Collado-Torres L, Leek JT, Taub MA. [RNA-seq transcript quantification from reduced-representation data in recount2](#). *bioRxiv*, 2018. doi: 10.1101/247346.

The Datasets

Show 10 entries

Search:

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
All	All	All	All			All	All	All	
SRP025982	1720	human	We present primary results from the Sequencing Quality Control (SEQC) project, coordinated by the United States Food and Drug Administration. Examining Illumina HiSeq, Life Technologies SOLiD and Roche 454 platforms at multiple laboratory sites using reference RNA samples with built-in controls, we assess RNA sequencing (RNA-seq) performance for sequence discovery and differential expression profiling and compare it to microarray and quantitative PCR (qPCR) data using complementary metrics. At all sequencing depths, we discover unannotated exon-exon junctions, with >80% validated by qPCR. We find that measurements of relative expression are accurate and reproducible across sites and platforms if specific filters are used. In contrast, RNA-seq and microarrays do not	RSE v2 counts v1 counts v1	RSE v2 counts v1 counts v1	RSE jx_bed jx_cov counts	RSE v2 RSE v1	link	v2 v1

SRP001558

①SRP001558と打ち込むと、このような画面になります。②も切り替わっていることがわかります。

recount2: analysis-ready R...

- **Collado-Torres L**, Nellore A, Jaffe AE. [recount worklow: Accessing over 70,000 human RNA-seq samples with Bioconductor](#). *F1000Research*, 2017. doi: 10.12688/f1000research.12223.1.
- **Wilks C**, Gaddipati P, Nellore A, Langmead B. [Snaptron: querying splicing patterns across tens of thousands of RNA-seq samples](#). *Bioinformatics*, 2018. doi: 10.1093/bioinformatics/btx547.
- **Fu J**, Kammers K, Nellore A, Collado-Torres L, Leek JT, Taub MA. [RNA-seq transcript quantification from reduced-representation data in recount2](#). *bioRxiv*, 2018. doi: 10.1101/247346.

The Datasets

Show 10 entries

Search: SRP001558

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
All	All	All	All	/	/	All	All	All	/
SRP001558	12	human	Comparative studies of gene regulation suggest an important role for natural selection in shaping gene expression patterns within and between species. Most of these studies, however, estimated gene expression levels using microarray probes designed to hybridize to only a small proportion of each gene. Here we used recently-developed RNA sequencing protocols, which side-step this limitation, to assess intra- and inter-species variation in gene regulatory processes in considerably more detail than was previously possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and rhesus macaques, using liver RNA samples from three males and three females from each species.	RSE v2 counts v2 RSE v1 counts v1	RSE v2 counts v2 RSE v1 counts v1	RSE jx_bed jx_cov counts	RSE v2 RSE v1	link	v2 v1

SRP001558

赤枠内の記述をよく見ると…ヒト(HS)・チンパンジー(PT)・アカゲザル(RM)の、Liver(肝臓)サンプルのRNA-seqデータであることが分かります。各生物種につき、メス(Female)3匹、オス(Male)3匹のデータがとられています。①が原著論文

possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and rhesus macaques, using liver RNA samples from three males and three females from each species.

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
SRP001558	12	human	Comparative studies of gene regulation suggest an important role for natural selection in shaping gene expression patterns within and between species. Most of these studies, however, estimated gene expression levels using microarray probes designed to hybridize to only a small proportion of each gene. Here we used recently-developed RNA sequencing protocols, which side-step this limitation, to assess intra- and inter-species variation in gene regulatory processes in considerably more detail than was previously possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and rhesus macaques, using liver RNA samples from three	RSE v2 counts v2 RSE v1 counts v1	RSE v2 counts v2 RSE v1 counts v1	RSE jx_bed jx_cov counts	RSE v2 RSE v1	link	v2 v1

possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and rhesus macaques, using liver RNA samples from three



Blekhman et al., *Genome Res.*, **20**: 180-9, 2010

SRP001558

①の記述からrecount2から提供されているカウント情報は、ヒト(HS)データ限定なのだろうと読み解く。②サンプル数は12と書かれている。メス(Female)3匹、オス(Male)3匹の計6個体で、各個体につき2反復(technical replicatesは2)とついているので、6個体×2反復の計12サンプルとなるのは妥当。

https://jhubiostatistics.shinyapps.io/recount/

recount2: analysis-ready R...

- **Collado-Torres L**, Nellore A, Jaffe AE. [recount workflow: Accessing over 70,000 human RNA-seq samples with Bioconductor](#). *F1000Research*, 2017. doi: 10.12688/f1000research.12223.1.
- **Wilks C**, Gaddipati P, Nellore A, Langmead B. [Snaptron: querying splicing patterns across tens of thousands of RNA-seq samples](#). *Bioinformatics*, 2018. doi: 10.1093/bioinformatics/btx547.
- **Fu J**, Kammers K, Nellore A, Collado-Torres L, Leek JT, Taub MA. [RNA-seq transcript quantification from reduced-representation data in recount2](#). *bioRxiv*, 2018. doi: 10.1101/247346.

The Datasets

Show 10 entries

Search: SRP001558

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
SRP001558	12	human	Comparative studies of gene regulation suggest an important role for natural selection in shaping gene expression patterns within and between species. Most of these studies, however, estimated gene expression levels using microarray probes designed to hybridize to only a small proportion of each gene. Here we used recently-developed RNA sequencing protocols, which side-step this limitation, to assess intra- and inter-species variation in gene regulatory processes in considerably more detail than was previously possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and rhesus macaques, using liver RNA samples from three	RSE v2 counts v2 RSE v1 counts v1	RSE v2 counts v2 RSE v1 counts v1	RSE jx_bed jx_cov counts	RSE v2 RSE v1	link	v2 v1



様々なfeature

①geneレベル、②exonレベル、③transcriptレベルなど、様々なfeatureのカウントデータが提供されていますね。

recount2: analysis-ready R... x

- **Collado-Torres L**, Nellore A, Jaffe AE. [recount workflow: Accessing over 70,000 human RNA-seq samples with BIOCONDUCTOR](#). *F1000Research*, 2017. doi: 10.12688/f1000research.12223.1.
- **Wilks C**, Gaddipati P, Nellore A, Langmead B. [Snaptron: querying splicing patterns across tens of thousands of RNA-seq samples](#). *Bioinformatics*, 2018. doi: 10.1093/bioinformatics/btx547.
- **Fu J**, Kammers K, Nellore A, Collado-Torres L, Leek JT, Taub MA. [RNA-seq transcript quantification from reduced-representation data in recount2](#). *bioRxiv*, 2018. doi: 10.1101/247346.

The Datasets

Show 10 entries Search: SRP001558

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
All	All	All	All	/	/	All	All	All	/
SRP001558	12	human	Comparative studies of gene regulation suggest an important role for natural selection in shaping gene expression patterns within and between species. Most of these studies, however, estimated gene expression levels using microarray probes designed to hybridize to only a small proportion of each gene. Here we used recently-developed RNA sequencing protocols, which side-step this limitation, to assess intra- and inter-species variation in gene regulatory processes in considerably more detail than was previously possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and rhesus macaques, using liver RNA samples from three males and three females from each species.	RSE v2 counts v2 RSE v1 counts v1	RSE v2 counts v2 RSE v1 counts v1	RSE jx_bed jx_cov counts	RSE v2 RSE v1	link	v2 v1

様々なfeature

①geneレベル、②exonレベル、③transcriptレベルなど、様々なfeatureのカウントデータが提供されてますね。④がreduced exons (v1)のデータ、⑤がdisjoint exons (v2)のデータ。

recount2: analysis-ready R... x

- **Collado-Torres L**, Nellore A, Jaffe AE. [recount worklow: Accessing over 70,000 human RNA-seq samples with BIOCONDUCTOR](#). *F1000Research*, 2017. doi: 10.12688/f1000research.12223.1.
- **Wilks C**, Gaddipati P, Nellore A, Langmead B. [Snaptron: querying splicing patterns across tens of thousands of RNA-seq samples](#). *Bioinformatics*, 2018. doi: 10.1093/bioinformatics/btx547.
- **Fu J**, Kammers K, Nellore A, Collado-Torres L, Leek JT, Taub MA. [RNA-seq transcript quantification from reduced-representation data in recount2](#). *bioRxiv*, 2018. doi: 10.1101/247346.

The Datasets

Show 10 entries Search: SRP001558

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
All	All	All	All	/	/	All	All	All	/
SRP001558	12	human	Comparative studies of gene regulation suggest an important role for natural selection in shaping gene expression patterns within and between species. Most of these studies, however, estimated gene expression levels using microarray probes designed to hybridize to only a small proportion of each gene. Here we used recently-developed RNA sequencing protocols, which side-step this limitation, to assess intra- and inter-species variation in gene regulatory processes in considerably more detail than was previously possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and rhesus macaques, using liver RNA samples from three males and three females from each species.	RSE v2 counts v2 RSE v1 counts v1	RSE v2 counts v2 RSE v1 counts v1	RSE jx_bed jx_cov counts	RSE v2 RSE v1	link	v2 v1

様々なfeature

①geneレベル、②exonレベル、③transcriptレベルなど、様々なfeatureのカウントデータが提供されてますね。④がreduced exons (v1)のデータ、⑤がdisjoint exons (v2)のデータ。これらに対応する記述です。

The screenshot shows the website <https://jhubiostatistics.shinyapps.io/recount/>. The main heading is "recount2: analysis-ready RNA-seq gene and exon counts datasets". Below this are navigation tabs: "Datasets", "Popular datasets", "GTEx", "TCGA", "Documentation", and "Download data with R". Red arrows point to "GTEx" (labeled 1), "TCGA" (labeled 2), and "Documentation" (labeled 3). A light blue notification box contains the text: "Transcript counts are now available thanks to the work of Fu et al, bioRxiv, 2018. Exon counts are now from disjoint exons (v2) instead of reduced ones (v1). Check the Documentation tab for further information." Red arrows point to "disjoint exons (v2)" (labeled 5) and "reduced ones (v1)" (labeled 4). The main content area features the "recount2" logo and the text "A multi-experiment resource of analysis-ready RNA-seq gene and exon count datasets". Below this is a detailed paragraph about the resource, including references to the ReCount project, Rail-RNA, and the derfinder Bioconductor package.

提供形式

提供している形式は、①生のカウントデータと、② RangedSummarizedExperimentというRオブジェクトの2種類。

The screenshot shows the website <https://jhubiostatistics.shinyapps.io/recount/>. The main navigation bar includes tabs for "Datasets", "Popular datasets", "GTEx", "TCGA", "Documentation", and "Download data with R". Below the navigation bar, there are links for "Accessing recount2 via SciServer" and "Contribute your data". A light blue notification box states: "Transcript counts are now available thanks to the work of Fu et al, bioRxiv, 2018. Exon counts are now from disjoint exons (v2) instead of reduced ones (v1). Check the Documentation tab for further information." The main content area features the "recount2" logo and the text: "A multi-experiment resource of analysis-ready RNA-seq gene and exon count datasets". Below this, a paragraph describes the resource: "recount2 is an online resource consisting of RNA-seq gene and exon counts as well as coverage bigWig files for 2041 different studies. It is the second generation of the [ReCount project](#). The raw sequencing data were processed with [Rail-RNA](#) as described in the recount2 paper and at [Nellore et al, Genome Biology, 2016](#) which created the coverage bigWig files. For ease of statistical analysis, for each study we created count tables at the gene and exon levels and extracted phenotype data, which we provide in their raw formats as well as in RangedSummarizedExperiment R objects (described in the [SummarizedExperiment](#) Bioconductor package). We also computed the mean coverage per study and provide it in a bigWig file, which can be used with the [derfinder](#) Bioconductor package to perform annotation-agnostic differential expression analysis at the expressed regions-level as described at [derfinder](#)." Two red arrows with circled numbers 1 and 2 point to the phrases "RangedSummarizedExperiment R objects" and "SummarizedExperiment Bioconductor package" respectively.

提供形式

それゆえ、①geneレベルの②disjoint exons (v2)のカウントデータだけでも…

recount2: analysis-ready R... x

- **Collado-Torres L**, Nellore A, Jaffe AE. [recount worklow: Accessing over 70,000 human RNA-seq samples with BIOCONDUCTOR](#). *F1000Research*, 2017. doi: 10.12688/f1000research.12223.1.
- **Wilks C**, Gaddipati P, Nellore A, Langmead B. [Snaptron: querying splicing patterns across tens of thousands of RNA-seq samples](#). *Bioinformatics*, 2018. doi: 10.1093/bioinformatics/btx547.
- **Fu J**, Kammers K, Nellore A, Collado-Torres L, Leek JT, Taub MA. [RNA-seq transcript quantification from reduced-representation data in recount2](#). *bioRxiv*, 2018. doi: 10.1101/247346.

The Datasets

Show 10 entries Search: SRP001558

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
All	All	All	All	/	/	All	All	All	/
SRP001558	12	human	Comparative studies of gene regulation suggest an important role for natural selection in shaping gene expression patterns within and between species. Most of these studies, however, estimated gene expression levels using microarray probes designed to hybridize to only a small proportion of each gene. Here we used recently-developed RNA sequencing protocols, which side-step this limitation, to assess intra- and inter-species variation in gene regulatory processes in considerably more detail than was previously possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and rhesus macaques, using liver RNA samples from three males and three females from each species.	RSE v2 counts v2 RSE v1 counts v1	RSE v2 counts v2 RSE v1 counts v1	RSE jx_bed jx_cov counts	RSE v2 RSE v1	link	v2 v1

RSE形式

RangedSummarizedExperimentの略称の①RSE v2をクリックして得られる、②rse_gene.Rdataと...

recount2: analysis-ready R... x

- **Collado-Torres L**, Nellore A, Jaffe AE. [recount workflow: Accessing over 70,000 human RNA-seq samples with BIOCONDUCTOR](#). *F1000Research*, 2017. doi: 10.12688/f1000research.12223.1.
- **Wilks C**, Gaddipati P, Nellore A, Langmead B. [Snaptron: querying splicing patterns across tens of thousands of RNA-seq samples](#). *Bioinformatics*, 2018. doi: 10.1093/bioinformatics/btx547.
- **Fu J**, Kammers K, Nellore A, Collado-Torres L, Leek JT, Taub MA. [RNA-seq transcript quantification from reduced-representation data in recount2](#). *bioRxiv*, 2018. doi: 10.1101/247346.

The Datasets

Show 10 entries Search: SRP001558

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
All	All	All	All	/	/	All	All	All	/
SRP001558	12	human	Comparative studies of gene regulation suggest an important role for natural selection in shaping gene expression patterns within and between species. Most of these studies, however, estimated gene expression levels using microarray probes designed to hybridize to only a small proportion of each gene. Here we used recently-developed RNA sequencing protocols, which side-step this limitation, to assess intra- and inter-species variation in gene regulatory processes in considerably more detail than was previously possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and macaques, using liver RNA samples from three males and three females from each species.	RSE v2 counts v2 RSE v1 counts v1	RSE v2 counts v2 RSE v1 counts v1	RSE jx_bed jx_cov counts	RSE v2 RSE v1	link	v2 v1

http://duffel.rail.bio/recount/v2/SRP001558/rse_gene.Rdata

生のカウント形式

- ①生のカウントデータのcounts v2をクリックして得られる、
- ②counts_gene.tsv.gzの2種類をダウンロード可能です。

recount2: analysis-ready R... x

- **Collado-Torres L**, Nellore A, Jaffe AE. [recount workflow: Accessing over 70,000 human RNA-seq samples with Bioconductor](#). *F1000Research*, 2017. doi: 10.12688/f1000research.12223.1.
- **Wilks C**, Gaddipati P, Nellore A, Langmead B. [Snaptron: querying splicing patterns across tens of thousands of RNA-seq samples](#). *Bioinformatics*, 2018. doi: 10.1093/bioinformatics/btx547.
- **Fu J**, Kammers K, Nellore A, Collado-Torres L, Leek JT, Taub MA. [RNA-seq transcript quantification from reduced-representation data in recount2](#). *bioRxiv*, 2018. doi: 10.1101/247346.

The Datasets

Show 10 entries Search: SRP001558

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
All	All	All	All	/	/	All	All	All	/
SRP001558	12	human	Comparative studies of gene regulation suggest an important role for natural selection in shaping gene expression patterns within and between species. Most of these studies, however, estimated gene expression levels using microarray probes designed to hybridize to only a small proportion of each gene. Here we used recently-developed RNA sequencing protocols, which side-step this limitation, to assess intra- and inter-species variation in gene regulatory processes in considerably more detail than was previously possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and rhesus macaques, using liver RNA samples from males and three females from each species.	RSE v2 counts v2 RSE v1 counts v1	RSE v2 counts v2 RSE v1 counts v1	RSE jx_bed jx_cov counts	RSE v2 RSE v1	link	v2 v1

http://duffel.rail.bio/recount/v2/SRP001558/counts_gene.tsv.gz

生のカウント形式

①counts_gene.tsv.gzを解凍し、Excel上で②counts_gene.tsvを眺める。なぜか③gene_id列が一番右側になっている(ので気持ち悪い)。また、ここで見られる情報以外は含まれない。

	A	B	C	D	E	F	G	H	I	J	K	L
1	SRR032116	SRR032118	SRR032119	SRR032120	SRR032121	SRR032122	SRR032123	SRR032124	SRR032125	SRR032126	SRR032127	gene_id
2	7690	6538	6780	3359	3702	3201	2812	9053	8005	8237	6866	ENSG00000000003.14
3	0	0	0	0	0	0	0	35	0	0	0	ENSG00000000005.5
4	1501	1224	1503	980	1769	970	1104	2192	1709	1358	1339	ENSG000000000419.12
5	1845	1418	1497	1678	2025	2695	2797	1707	2000	2658	2282	ENSG000000000457.13
6	508	700	962	630	847	902	875	321	844	1206	711	ENSG000000000460.16
7	1615	2028	1963	7241	6495	2081	2655	1983	1263	2238	2504	ENSG000000000938.12
8	208796	249132	271774	141088	150247	157858	175680	245692	235295	246368	217387	ENSG000000000971.15
9	6169	4360	5091	3109	4194	3750	3955	4477	4550	5580	4041	ENSG00000001036.13
10	15747	18358	18146	7409	7320	7153	8522	24045	26675	11906	11058	ENSG00000001084.10
11	1995	1733	1794	1925	2232	1678	1995	1574	1791	2056	2245	ENSG00000001167.14
12	433	140	105	245	209	105	333	259	450	245	175	ENSG00000001460.17
13	1107	854	663	782	855	454	593	1109	1427	840	727	ENSG00000001461.16
14	3367	2900	3043	3004	3275	2391	2298	4168	4012	3939	3580	ENSG00000001497.16

possible. Specifically, we used RNAseq to study
transcript levels in humans, chimpanzees, and
rhesus macaques, using liver RNA samples from
males and three females from each species.

RSE形式を推奨

②rse_gene.Rdataをロードして得られる RangedSummarizedExperiment (RSE)形式のオブジェクトには、サンプルに付随する各種情報(メタデータ)や、geneの染色体上の位置、配列長、gene symbolsなど多くの情報が含まれているのでいろいろと便利です。なので慣れましょう。

https://jhubiostatistics.shinyapps.io/recount/

recount2: analysis-ready R...

- **Collado-Torres L**, Nellore A, Jaffe AE. *recount workflow: Accessing over 70,000 human RNA-seq datasets*. *Bioinformatics*, 2018. doi: 10.1093/bioinformatics/btx547.
- **Wilks C**, Gaddipati P, Nellore A, Langmead B. *Snaptron: querying splicing patterns across tens of thousands of RNA-seq samples*. *Bioinformatics*, 2018. doi: 10.1093/bioinformatics/btx547.
- **Fu J**, Kammers K, Nellore A, Collado-Torres L, Leek JT, Taub MA. *RNA-seq transcript quantification from reduced-representation data in recount2*. *bioRxiv*, 2018. doi: 10.1101/247346.

The Datasets

Show 10 entries

Search: SRP001558

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
All	All	All	All	/	/	All	All	All	/
SRP001558	12	human	Comparative studies of gene regulation suggest an important role for natural selection in shaping gene expression patterns within and between species. Most of these studies, however, estimated gene expression levels using microarray probes designed to hybridize to only a small proportion of each gene. Here we used recently-developed RNA sequencing protocols, which side-step this limitation, to assess intra- and inter-species variation in gene regulatory processes in considerably more detail than was previously possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and rhesus macaques, using liver RNA samples from three males and three females from each species.	RSE v2 counts v2 RSE v1 counts v1	RSE v2 counts v2 RSE v1 counts v1	RSE jx_bed jx_cov counts	RSE v2 RSE v1	link	v2 v1

http://duffel.rail.bio/recount/v2/SRP001558/rse_gene.Rdata

rse_gene.Rdata

① RSE v2をクリックすると、recount2から② rse_gene.Rdataをダウンロードできますが、迷惑をかけるのでここではやらないでください。

https://jhubiostatistics.shinyapps.io/recount/

recount2: analysis-ready R...

- **Collado-Torres L**, Nellore A, Jaffe AE. [recount workflow: Accessing over 70,000 human RNA-seq samples with Bioconductor](#). *F1000Research*, 2017. doi: 10.12688/f1000research.12223.1.
- **Wilks C**, Gaddipati P, Nellore A, Langmead B. [Snaptron: querying splicing patterns across tens of thousands of RNA-seq samples](#). *Bioinformatics*, 2018. doi: 10.1093/bioinformatics/btx547.
- **Fu J**, Kammers K, Nellore A, Collado-Torres L, Leek JT, Taub MA. [RNA-seq transcript quantification from reduced-representation data in recount2](#). *bioRxiv*, 2018. doi: 10.1101/247346.

The Datasets

Show 10 entries Search: SRP001558

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
All	All	All	All	/	/	All	All	All	/
SRP001558	12	human	Comparative studies of gene regulation suggest an important role for natural selection in shaping gene expression patterns within and between species. Most of these studies, however, estimated gene expression levels using microarray probes designed to hybridize to only a small proportion of each gene. Here we used recently-developed RNA sequencing protocols, which side-step this limitation, to assess intra- and inter-species variation in gene regulatory processes in considerably more detail than was previously possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and macaques, using liver RNA samples from three males and three females from each species.	RSE v2 counts v2 RSE v1 counts v1	RSE v2 counts v2 RSE v1 counts v1	RSE jx_bed jx_cov counts	RSE v2 RSE v1	link	v2 v1

http://duffel.rail.bio/recount/v2/SRP001558/rse_gene.Rdata

Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?! カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

Rパッケージrecount

ここで見えているのはrecount2というウェブサイト。①RSE v2をクリックすることで、②rse_gene.Rdataをダウンロードできます。他の手段として、Rパッケージrecountを用いることで、③SRP001558と④geneをオプションとして与えることで、rse_gene.Rdataをダウンロードしたり、カウントデータを取得することができます。

https://jhubiostatistics.shinyapps.io/recount/

- Collado-Torres L, Nellore A, Jaffe AE. [recount workflow: Accessing over 70,000 human RNA-seq](#) 10.12688/f1000research.12223.1.
- Wilks C, Gaddipati P, Nellore A, Langmead B. [Snaptron: querying splicing patterns across tens of](#) 10.1093/bioinformatics/btx547.
- Fu J, Kammers K, Nellore A, Collado-Torres L, Leek JT, Taub MA. [RNA-seq transcript quantification from reduced-representation data in recount2](#). *bioRxiv*, 2018. doi: 10.1101/247346.

The Datasets

Show 10 entries Search: SRP001558

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
-----------	-------------------	---------	----------	------	------	-----------	-------------	-----------	------------

SRP001558	12	human	Comparative studies of gene regulation suggest an important role for natural selection in shaping gene expression patterns within and between species. Most of these studies, however, estimated gene expression levels using microarray probes designed to hybridize to only a small proportion of each gene. Here we used recently-developed RNA sequencing protocols, which side-step this limitation, to assess intra- and inter-species variation in gene regulatory processes in considerably more detail than was previously possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and macaques, using liver RNA samples from three males and three females from each species.	RSE v2 counts v2 RSE v1 counts v1	RSE v2 counts v2 RSE v1 counts v1	RSE jx_bed jx_cov counts	RSE v2 RSE v1	link	v2 v1
-----------	----	-------	---	--------------------------------------	--------------------------------------	--------------------------	---------------	------	-------

http://duffel.rail.bio/recount/v2/SRP001558/rse_gene.Rdata

Rパッケージrecount

(Rで)塩基配列解析

(last modified 2018/05/30, since 2010)

このウェブページは、[recount](#)パッケージのインストールと使用方法について説明しています。

What's new?

- 「マップ後 | カウント 情報取得 | リアルデータ | SRP001558 | [recount\(Collado-Torres 2017\)](#)」
- 「イントロ | [recount](#)パッケージについて」
- 「[H29年度](#)」
- 「Silhouette」

- マップ後 | カウント 情報取得 | paired-end | ゲノム | アノテーション無 | [QuasR\(Gaidatzis 2015\)](#) (last modified 2015/07/02)
- マップ後 | カウント 情報取得 | paired-end | トランスクリプトーム | [QuasR\(Gaidatzis 2015\)](#) (last modified 2016/02/12)
- マップ後 | カウント 情報取得 | トランスクリプトーム | [BEDファイルから](#) (last modified 2014/06/21)
- [カウント 情報取得 | について](#) (last modified 2018/06/10) **NEW**
- カウント 情報取得 | リアルデータ | SRP001558 | [recount\(Collado-Torres 2017\)](#) (last modified 2018/06/10) **NEW**
- カウント 情報取得 | リアルデータ | ERP000546 | [recount\(Collado-Torres 2017\)](#) (last modified 2018/06/10) **NEW**
- カウント 情報取得 | [シミュレーションデータ](#) (last modified 2018/06/09) **NEW**
- [配列長とカウント 数の関係](#) (last modified 2018/06/09) **NEW**
- [正規化 | について](#) (last modified 2018/06/09) **NEW**

カウント情報取得 | リアルデータ | SRP001558 | [recount\(Collado-Torres_2017\)](#) **NEW**

[recount](#)パッケージを用いて、[SRP001558](#)([Blekhman et al., Genome Res., 2010](#); ブラウザはIE以外を推奨)のカウント 情報を含むRangedSummarizedExperimentクラスオブジェクトという形式の.Rdataをダウンロードしたり、カウントデータの数値行列にした状態で保存するやり方を示します。原著論文では、3生物種(ヒト12 samples、チンパンジー12 samples、そしてアカゲザル12 samples)のカウントデータを取得しています。ウェブサイト [recount2](#) 上でSRP001558で検索すると、number of samplesが12、speciesがhumanとなっていることから、提供されているカウントデータはhumanに限定されていることがわかります。例題2までで、なぜか11 samples分のデータしかないことに気づきます。これは、ウェブサイト [recount2](#) 上でSRP001558で検索し、phenotype列の [link](#) をダウンロードして得られる [SRP001558.tsv](#) を眺めることでなんとなくの理由がわかります。私は、「SRR032117のデータがおかしなことになっており、recount2で提供するクオリティに達しなかった。このため、recount2のウェブページ上は12 samplesとなっているものの、カウントデータ自体は11 samples分となっているのだろう。」と予想しました。また、[PRJNA119135](#)・[GSE17274](#)・[SRA010277](#)はENA上にリンク先がありますが、ウェブサイト [recount2](#) 上では引っかけられていませんでした。

「ファイル」-「ディレクトリの変更」でダウンロードしたいディレクトリに移動し以下をコピー。

1. geneレベルカウントデータ情報を得たい場合:

SRP001558という名前のフォルダが作成されます。中にあるrse-gene.Rdataをロードして読み込むとrse-geneというオブジェクト名で取り扱えます。ウェブサイト [recount2](#) 上でSRP001558で検索し、gene列の [RSE v2](#) をダウンロードして得られる rse_gene.Rdataと同じです。

```
param_ID <- "SRP001558" #IDを指定
```

#必要なパッケージをロード

①例題1が、SRP001558のgeneレベルカウントデータを含むrse_gene.Rdataをダウンロードする基本形。このあとダウンロードするので、ここではやらない。

例題1

カウント情報取得 | リアルデータ | SRP001558 | [recount\(Collado-Torres_2017\)](#) **NEW**

[recount](#)パッケージを用いて、[SRP001558\(Blekhman et al., Genome Res., 2010\)](#); ブラウザは正以外を推奨)のカウント情報を含むRangedSummarizedExperimentクラスオブジェクトという形式の.Rdataをダウンロードしたり、カウントデータの数値行列にした状態で保存するやり方を示します。原著論文では、3生物種(ヒト12 samples、チンパンジー12 samples、そしてアカゲザル12 samples)のカウントデータを取得しています。ウェブサイト [recount2](#) 上でSRP001558で検索すると、number of samplesが12、speciesがhumanとなっていることから、提供されているカウントデータはhumanに限定されていることがわかります。例題2までで、なぜか11 samples分のデータしかないことに気づきます。これは、ウェブサイト [recount2](#) 上でSRP001558で検索し、phenotype列の[link](#)をダウンロードして得られる[SRP001558.tsv](#)を眺めることでなんとなくの理由がわかります。私は、「SRR032117のデータがおかしなことになっており、recount2で提供するクオリティに達しなかった。このため、recount2のウェブページ上は12 samplesとなっているものの、カウントデータ自体は11 samples分となっているのだろう。」と予想しました。また、[PRJNA119135](#)・[GSE17274](#)・[SRA010277](#)はENA上にリンク先がありますが、ウェブサイト [recount2](#) 上では引っかかってきませんでした。

「ファイル」-「ディレクトリの変更」でダウンロードしたいディレクトリに移動し以下をコピー。

1. geneレベルカウントデータ情報を得たい場合:


SRP001558という名前のフォルダが作成されます。中にあるrse_gene.Rdataをロードして読み込むとrse_geneというオブジェクト名で取り扱えます。ウェブサイト [recount2](#) 上でSRP001558で検索し、gene列の[RSE v2](#)をダウンロードして得られるrse_gene.Rdataと同じです。

```
param_ID <- "SRP001558"           #IDを指定
#必要なパッケージをロード
library(recount)                  #パッケージの読み込み
#本番(.Rdataをダウンロード)
download_study(param_ID, type="rse-gene", download=T)#ダウンロード
```

①例題3が、手元にあるrse_gene.Rdataを読み込んで、geneレベルカウントデータの数値行列を得る基本形。
②rse_gene.Rdataをデスクトップにダウンロード

例題3

カウント情報取得 | リアルデータ | SRP001558 | [recount\(Collado-Torres_2017\)](#) NEW

[recount](#)パッケージを用いて、[SRP001558\(Blekhman et al., Genome Res., 2010\)](#); ブラウザは正以外を推奨)のカウント情報を含むRangedSummarizedExperimentクラスオブジェクトという形式の.Rdataをダウンロードしたり、カウントデータの数値行列にした状態で保存するやり方を示します。原著論文では、3生物種(ヒト12 samples、チンパンジー12 samples、そしてアカゲザル12 samples)のカウントデータを取得しています。ウェブサイト [recount2](#) 上でSRP001558で検索すると、number of samplesが12、speciesがhumanとなっていることから、提供されているカウントデータはhumanに限定されていることがわかります。例題2までで、なぜか11 samples分のデータしかないことに気づきます。これは、ウェブサイト [recount2](#) 上でSRP001558で検索し、phenotype列の[link](#)をダウンロードして得られるSRP001558.tsvを眺める  ほとんどなくの理由がわかります。私は、

「SRR032117のデータがおかしいこと」
ページ上は12 samplesとなっているも
た、[PRJNA119135](#)・[GSE17274](#)・[SRA010](#)
せんでした。
「ファイル」-「ディレクトリの変更」でダウン

1. geneレベルカウントデータ情報を得たい
SRP001558という名前のフォルダが作成
ト名で取り扱えます。ウェブサイト [recount](#)
rse_gene.Rdataと同じです。

```
param_ID <- "SRP001558"

#必要なパッケージをロード
library(recount)

#本番(.Rdataをダウンロード)
download_study(param_ID, type=
```

3. ダウンロード済みのrse_gene.Rdataを入力として読み込む場合:

ウェブサイト [recount2](#) 上でSRP001558で検索し、gene列のRSE v2のところからダウンロードして得られた geneレベルカウントデータ(rse_gene.Rdata; 約3MB)を読み込んで、カウントの数値行列情報(58,037 genes×11 samples)のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは [hoge3.txt](#) です。

```
in_f <- "rse_gene.Rdata"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt"          #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount)              #パッケージの読み込み

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)                    #in_fで指定した.Rdataをロード
hoge <- rse_gene               #hogeとして取り扱う
hoge                           #確認してるだけです

#本番(カウントデータ取得)
data <- assays(hoge)$counts    #カウントデータ行列を取得してdataに格納
dim(data)                     #行数と列数を表示
head(data)                    #確認してるだけです

#ファイルに保存
tmp <- cbind(rownames(data), data) #保存したい情報をtmpに格納
write.table(tmp, out_f, sep="\t", append=F, quote=F, row.names=F) #tmpの中身を指定したファイル名
```

①作業ディレクトリをデスクトップにして、②rse_gene.Rdataがある状態で、とりあえず③の部分のコピペ。

例題3

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata"
out_f <- "hoge3.txt"

#必要なパッケージをロード
library(recount)

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge

#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)

#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F,

```



#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納

#パッ

#in_
#hog
#確認

#カウ
#行數
#確認

#保存

```

RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ

R Console

'citation()' と入力してください。

'demo()' と入力すればデモをみることができます。
'help()' とすればオンラインヘルプが出ます。
'help.start()' で HTML ブラウザによるヘルプがみられま$
'q()' と入力すれば R を終了します。

> getwd()
[1] "C:/Users/kojik/Desktop"
> list.files()
[1] "BlastViewer.lnk" "desktop.ini"
[3] "FastQC"          "hoge"
[5] "rse_gene.Rdata"  "share"
> |

```



例題3

ここまでが、①rse_gene.Rdataを、②ロードして(取り込んで)、オリジナルのrse_geneというオブジェクト名をhogeに変更したものを表示させた結果です。

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata"
out_f <- "hoge3.txt"

#必要なパッケージをロード
library(recount)

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge

#本番のカウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)

#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F,

```

#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納

#パッ

#in
#hog
#確認

#カ
#行
#確認

#保存

```

> hoge
class: RangedSummarizedExperiment
dim: 58037 11
metadata(0):
assays(1): counts
rownames(58037): ENSG00000000003.14
  ENSG00000000005.5 ... ENSG00000283698.1
  ENSG00000283699.1
rowData names(3): gene_id bp_length symbol
colnames(11): SRR032116 SRR032118 ... SRR032126
  SRR032127
colData names(21): project sample ... title
  characteristics
> |

```

①load関数は、②「ファイル - 作業スペースの読み込み」で③rse_gene.Rdataを選択することと同義です。

Tips: load関数

3. ダウンロード済みの rse_gene.Rdata を入力として読み込む場合:

ウェブサイト [recount2](#) 上でSRP001558で検索し、gene列のRSE v2のところからダウンロードして得られた geneレベルカウントデータ(rse_gene.Rdata; 約3MB)を読み込んで、カウントの数値行列情報(58,037 genes×11 samples)のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルはhoge3.txtです。

```

in_f <- "rse_gene.Rdata"
out_f <- "hoge3.txt"

#必要なパッケージをロード
library(recount)

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge

#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)

#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F,

```

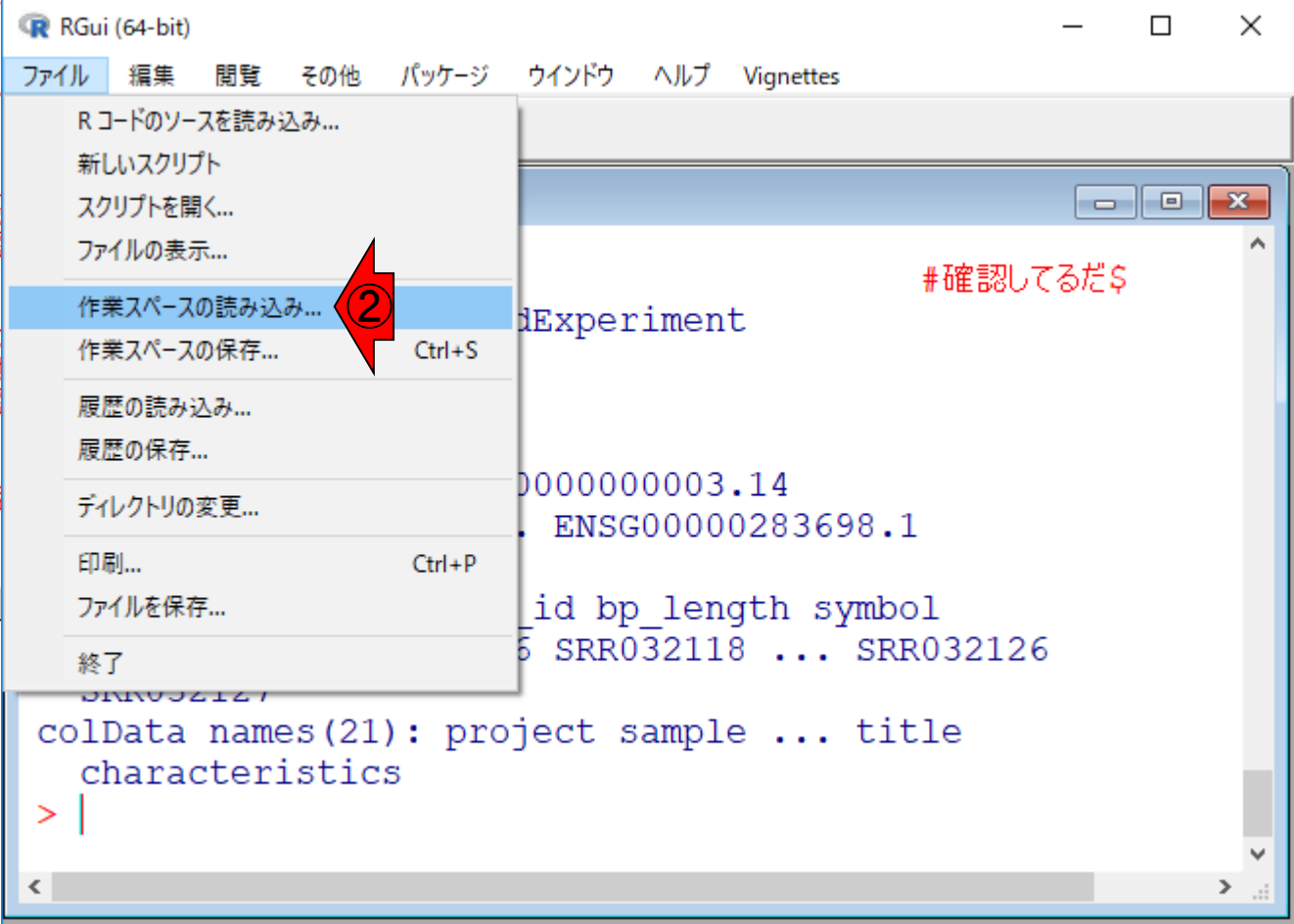
#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納

#パッ

#in_
#hog
#確認

#カウ
#行数
#確認

#保存



#確認してるだ\$

Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?! カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

①hogeが、②RangedSummarizedExperiment (RSE)形式のオブジェクトです。

RSE

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt" #出力ファイル名を指定してout_fに格納
```

```
#必要なパッケージをロード
library(recount)
```

```
#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge
```

```
#本番のカウントデータ取得
data <- assays(hoge)$counts
dim(data)
head(data)
```

```
#ファイルに保存
tmp <- cbind(rownames(data), data) #保存
write.table(tmp, out_f, sep="\t", append=F,
```

The screenshot shows the R Console output for the code above. A red arrow labeled '1' points to the prompt `> hoge`. The output shows the class `RangedSummarizedExperiment` and its dimensions `dim: 58037 11`. Another red arrow labeled '2' points to the `class` line. A third red arrow labeled '1' points to the `assays(1): counts` line, with the comment `#確認してるだ$` next to it.

```
> hoge
class: RangedSummarizedExperiment
dim: 58037 11
metadata(0):
assays(1): counts
rownames(58037): ENSG00000000003.14
  ENSG00000000005.5 ... ENSG00000283698.1
  ENSG00000283699.1
rowData names(3): gene_id bp_length symbol
colnames(11): SRR032116 SRR032118 ... SRR032126
  SRR032127
colData names(21): project sample ... title
  characteristics
> |
```


カウントデータ格納部分

今とりあえず欲しいのは、カウントデータの数値行列情報。①countsという文字列をたよりに、②assaysというところに格納されているのだな、などと判断する。

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata"
out_f <- "hoge3.txt"

#必要なパッケージをロード
library(recount)

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge

#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)

#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F,

```

#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納

#パッ

#in_
#hog
#確認

#カウ
#行
#確認

#保

```

R RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
> hoge
class: RangedSummarizedExperiment
dim: 58037 11
metadata(0):
assays(1): counts
rownames(58037): ENSG00000000003.14
  ENSG00000000005.5 ... ENSG00000283698.1
  ENSG00000283699.1
rowData names(3): gene_id bp_length symbol
colnames(11): SRR032116 SRR032118 ... SRR032126
  SRR032127
colData names(21): project sample ... title
  characteristics
> |

```

カウントデータ格

私はある程度①classオブジェクトの概念やノリに慣れているので、まずは②str関数実行結果を眺める。そして、hogeオブジェクト内から必要な情報をどのように得るかを試行錯誤する。そして大抵数回程度のトライアルで③のような書き方でよいという結論に至る。慣れないうちは、②の結果に加えてrecountパッケージのマニュアルを眺める必要もある。

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上でSRP001558で検索し、gene列のRSE v2.0のデータ(`rse_gene.Rdata`; 約3MB)を読み込んで、カウントの数値行列をテキストファイルで保存するやり方です。出力ファイルは[hoge3.txt](#)です。

```
in_f <- "rse_gene.Rdata"
out_f <- "hoge3.txt"
```

#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納

```
#必要なパッケージをロード
library(recount)
```

#パッ

```
#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge
```

#in_
#hog
#確認

```
#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)
```



```
#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F,
```

#保存

RGui (64-bit) window showing R Console output:

```
> hoge
class: RangedSummarizedExperiment
dim: 58037 11
metadata(0):
assays(1): counts
rownames(58037): ENSG00000000003.14
  ENSG00000000005.5 ... ENSG00000283698.1
  ENSG00000283699.1
rowData names(3): gene_id bp_length symbol
colnames(11): SRR032116 SRR032118 ... SRR032126
  SRR032127
colData names(21): project sample ... title
  characteristics
> str(hoge)|
```



str実行結果

これが、str(hoge)実行結果の最後のほうの画面。画面がざっと流れる。慣れるとこれを頼りにどんな情報が格納されているかの全貌を知ることができて便利。例えば、①ではassays(hoge)\$countsと書いているが、経験を積んでいくことで、②の部分を見た段階で「hoge@assaysでもOKかも...」と思えるようになる。

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58037 x 14) をテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt"     #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount) #パ

#本番(typeで指定した名前の.Rdataをロード)
load(in_f) #in_
hoge <- rse_gene #hog
hoge #確認

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウ
dim(data) #行数
head(data) #確認

#ファイルに保存
tmp <- cbind(rownames(data), data) #保存
write.table(tmp, out_f, sep="\t", append=F,
```

```
RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
> str(hoge)
.. .. ..@ metadata      : list()
.. .. ..@ assays        :Reference class 'ShallowSimpleLi$
.. .. ..$ data: NULL
.. .. ..and 14 methods.
.. ..@ NAMES            : NULL
.. ..@ elementMetadata:Formal class 'DataFrame' [packag$
.. .. ..@ rownames      : NULL
.. .. ..@ nrows         : int 58037
.. .. ..@ listData      : Named list()
.. .. ..@ elementType   : chr "ANY"
.. .. ..@ elementMetadata: NULL
.. .. ..@ metadata      : list()
.. ..@ metadata        : list()
> |
```

hoge@assays

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt"     #出力ファイル名を指定してout_fに格納
```

```
#必要なパッケージをロード
library(recount)
```

```
#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge
```

```
#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)
```

```
#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F)
```

```
#in_f
#hoge
#確認
```

```
#パッケージ
```

```
#in_f
#hoge
#確認
```

```
#カウント
#行数
#確認
```

```
#保存
```

```
RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
> hoge@assays
Reference class object of class "ShallowSimpleListAssay"
Field "data":
List of length 1
names(1): counts
> assays(hoge)
List of length 1
names(1): counts
> |
```

str(hoge@assays)

再度①strでhoge@assays内部の構造(structure)を眺める。②\$ dataと書かれているので、hoge@assays\$dataをやってみようという思考回路になる。キーボードの上下左右の矢印キーを駆使して効率的に打ち込んでいますよね?!

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上でSRP001558で検索し、`gene`列のRSE v2のところからダウンロードデータ(`rse_gene.Rdata`; 約3MB)を読み込んで、カウントの数値行列情報(58,037 genes x 10,000 cells)をテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt"     #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount) #パ

#本番(typeで指定した名前の.Rdataをロード)
load(in_f) #in_
hoge <- rse_gene #hog
hoge #確認

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウ
dim(data) #行数
head(data) #確認

#ファイルに保存
tmp <- cbind(rownames(data), data) #保存
write.table(tmp, out_f, sep="\t", append=F,
```

```
RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
..@ metadata      : list()
> hoge@assays
Reference class object of class "ShallowSimpleListAssays"
Field "data":
List of length 1
names(1): counts
> assays(hoge)
List of length 1
names(1): counts
> str(hoge@assays)
Reference class 'ShallowSimpleListAssays' [package "SummarizedExperiment"]
 $ data: NULL
 and 14 methods.
> |
```

①

②

hoge@assays\$data

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt"         #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount)            #パッ

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)                  #in_
hoge <- rse_gene            #hog
hoge                         #確認

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カ
dim(data)                   #行
head(data)                  #研

#ファイルに保存
tmp <- cbind(rownames(data), data) #保存
write.table(tmp, out_f, sep="\t", append=F,

```

```

RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
Field "data":
List of length 1
names(1): counts
> assays(hoge)
List of length 1
names(1): counts
> str(hoge@assays)
Reference class 'ShallowSimpleListAssays' [package "Su$
 $ data: NULL
 and 14 methods.
> hoge@assays$data
List of length 1
names(1): counts
> |

```

str(hoge@assays\$data)

①str(hoge@assays\$data)の結果より、②
\$ countsからhoge@assays\$data\$counts
でも、③と同じ意味なのだろうと想像したり
、④dim関数で行数と列数を確認したり
する

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得たデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) をテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt"     #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount)

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge

#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)

#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F,
```

```
R RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
> hoge@assays$data
List of length 1
names(1): counts
> str(hoge@assays$data)
Formal class 'SimpleList' [package "S4Vectors"] with 4$
..@ listData      :List of 1
.. ..$ counts: num [1:58037, 1:11] 7690 0 1501 1845 $
.. .. ..- attr(*, "dimnames")=List of 2
.. .. ..$ : chr [1:58037] "ENSG000000000003.14" "ES
.. .. ..$ : chr [1:11] "SRR032116" "SRR032118" "S$
..@ elementType   : chr "ANY"
..@ elementMetadata: NULL
..@ metadata      : list()
> dim(hoge@assays$data$counts)|
```

str(hoge@assays\$data)

①str(hoge@assays\$data)の結果より、②
\$ countsからhoge@assays\$data\$counts
でも、③と同じ意味なのだろうと想像したり、
④dimで行数と列数を確認したりする。

3. ダウンロード済みの rse_gene.Rdata を入力として読み込む場合:

ウェブサイト [recount2](#) 上で SRP001558 で検索し、gene 列の RSE v2 のところからダウンロードして得られた gene レベルカウントデータ (rse_gene.Rdata; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは hoge3.txt です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt"     #出力ファイル名を指定してout_fに格納
```

```
#必要なパッケージをロード
library(recount)
```

```
#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge
```

```
#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)
```

```
#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F,
```

```
R RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
> hoge@assays$data
List of length 1
names(1): counts
> str(hoge@assays$data)
Formal class 'SimpleList' [package "S4Vectors"] with 4$
..@ listData      :List of 1
.. ..$ counts: num [1:58037, 1:11] 7690 0 1501 1845 $
.. .. ..- attr(*, "dimnames")=List of 2
.. .. ..$ : chr [1:58037] "ENSG000000000003.14" "ES
.. .. ..$ : chr [1:11] "SRR032116" "SRR032118" "S$
..@ elementType   : chr "ANY"
..@ elementMetadata: NULL
..@ metadata      : list()
> dim(hoge@assays$data$counts)|
```


str(hoge@assays\$data)

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt"     #出力ファイル名を指定してout_fに格納
```

```
#必要なパッケージをロード
library(recount)
```

```
#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge
```

```
#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)
```

```
#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F,
```

```
RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
Formal class 'SimpleList' [package "S4Vectors"] with 4$
 ..@ listData      :List of 1
 .. ..$ counts: num [1:58037, 1:11] 7690 0 1501 1845 $
 .. .. ..- attr(*, "dimnames")=List of 2
 .. .. ..$ : chr [1:58037] "ENSG00000000003.14" "ES
 .. .. ..$ : chr [1:11] "SRR032116" "SRR032118" "SS
 ..@ elementType   : chr "ANY"
 ..@ elementMetadata: NULL
 ..@ metadata      : list()

> dim(hoge@assays$data$counts)
[1] 58037    11

> dim(assays(hoge)$counts)
[1] 58037    11

> |
```

最後までコピー

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt"         #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount)            #パッ

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)                  #in_
hoge <- rse_gene            #hog
hoge                         #確認

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウ
dim(data)                   #行数
head(data)                  #確認

#ファイルに保存
tmp <- cbind(rownames(data), data) #保存
write.table(tmp, out_f, sep="\t", append=F,

```

The screenshot shows the RGui interface with the R Console window open. The console displays a table of gene expression data with columns for gene IDs and sample IDs (SRR032126 and SRR032127). Below the table, R code is being executed to save the data to a file.

ENSG00000000460.16	875	321	844
ENSG00000000938.12	2655	1983	1263
		SRR032126	SRR032127
ENSG00000000003.14	8237	6866	
ENSG00000000005.5	0	0	
ENSG00000000419.12	1358	1339	
ENSG00000000457.13	2658	2282	
ENSG00000000460.16	1206	711	
ENSG00000000938.12	2238	2504	

```

>
> #ファイルに保存
> tmp <- cbind(rownames(data), data)      #保存したい情$
> write.table(tmp, out_f, sep="\t", append=F, quote=F,$
> |

```

最後までコピー

58,037行 × 11列からなるカウント行列の、①最初の6行分の行名(rownames)と、②最後の2列分の列名(colnames)。

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上でSRP001558で検索し、`gene`列のRSE v2のところからダウンロードして得られた `gene`レベルカウントデータ(`rse_gene.Rdata`; 約3MB)を読み込んで、カウントの数値行列情報(58,037 genes × 11 samples)のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt" #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount) #パッ

#本番(typeで指定した名前の.Rdataをロード)
load(in_f) #in_
hoge <- rse_gene #hog
hoge #確認

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウ
dim(data) #行数
head(data) #確認

#ファイルに保存
tmp <- cbind(rownames(data), data) #①
write.table(tmp, out_f, sep="\t", append=

```

```

RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
ENSG00000000460.16      875      321      844
ENSG00000000938.12    2655     1983     1263
ENSG00000000003.14    8237     6866
ENSG00000000005.5         0         0
ENSG00000000419.12    1358     1339
ENSG00000000457.13    2658     2282
ENSG00000000460.16    1206      711
ENSG00000000938.12    2238     2504

>
> #ファイルに保存
> tmp <- cbind(rownames(data), data) #保存したい情$
> write.table(tmp, out_f, sep="\t", append=F, quote=F,$
> |

```

Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?! カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

RSE

①例題3で、②RangedSummarizedExperiment (RSE) というクラスオブジェクトである③hogeを再度表示。RSE形式からRSEクラスという表現に変えているが、他にもRSE containerやRSE objectなどいろんな呼び方をする。細かいことは気にしなくてよい。

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

① サイト [recount2](#) 上で SRP001558 で検索し、 `gene` 列の `RSE v2` のところからデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報(58,000行 x 11列)をテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata"
out_f <- "hoge3.txt"

#必要なパッケージをロード
library(recount)

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge

#本番のカウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)

#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F,

```

#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納

#パッ

#in
#hog
#確認

#カ
#行
#確認

#保存

```

RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
> hoge
class: RangedSummarizedExperiment
dim: 58037 11
metadata(0):
assays(1): counts
rownames(58037): ENSG00000000003.14
  ENSG00000000005.5 ... ENSG00000283698.1
  ENSG00000283699.1
rowData names(3): gene_id bp_length symbol
colnames(11): SRR032116 SRR032118 ... SRR032126
  SRR032127
colData names(21): project sample ... title
  characteristics
> |

```

これまで主に着目していたのは、①カウ
ントデータ取得に関するものであった。

RSE

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

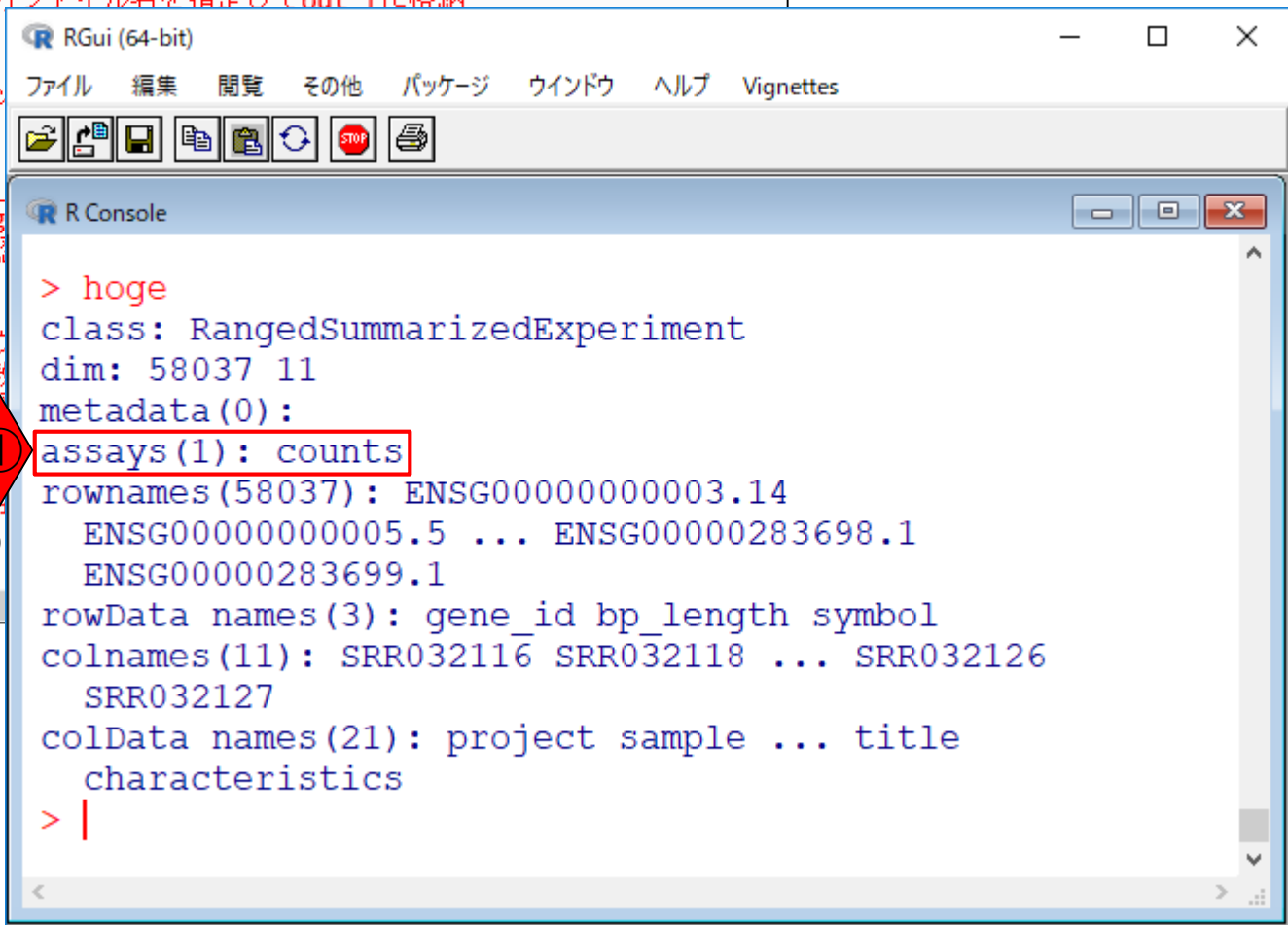
```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt" #出力ファイル名を指定してout_fに格納
```

```
#必要なパッケージをロード
library(recount) #パ
```

```
#本番(typeで指定した名前の.Rdataをロード)
load(in_f) #in_
hoge <- rse_gene #hog
hoge #確認
```

```
#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウ
dim(data) #行
head(data) #確認
```

```
#ファイルに保存
tmp <- cbind(rownames(data), data) #保
write.table(tmp, out_f, sep="\t", append=F,
```



rownames

①はカウントデータ行列の行名情報。②
rownames(hoge)で取り出せる。58,037個の
行名が一気に表示される。やらなくてもよい。

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt"     #出力ファイル名を指定してout_fに格納
```

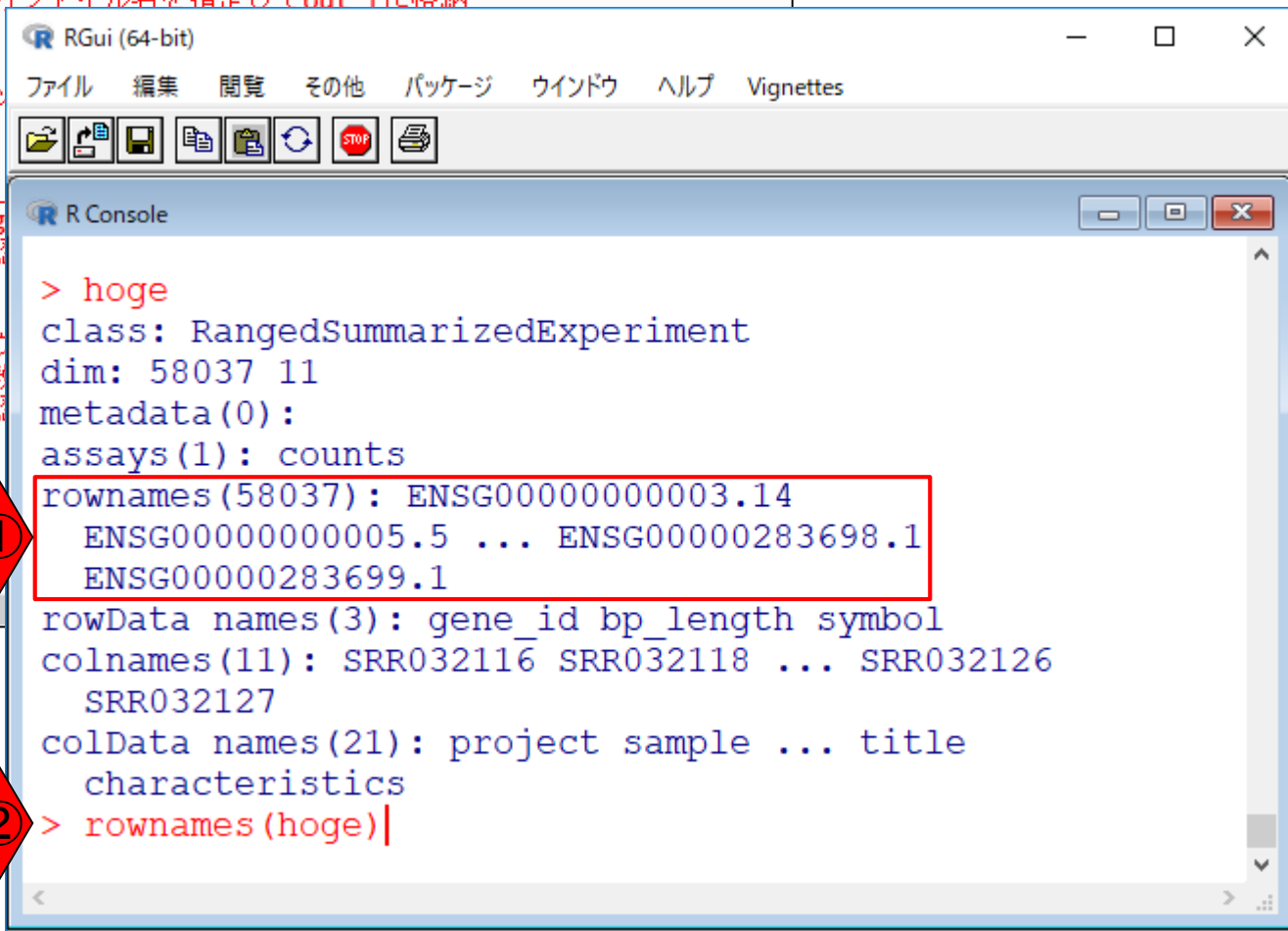
```
#必要なパッケージをロード
library(recount)
```

```
#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge
```

```
#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)
```

```
#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=
```

#パ
#in
#hog
#確認
#カ
#行
#確認
#保



rowData

①はgeneレベルカウントデータ行列の行(gene)ごとの付随情報としてどのようなものがあるかを示している。gene_id, bp_length, symbolの3種類の情報を、②rowData(hoge)で取り出せる。

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上でSRP001558で検索し、gene列のRSE v2のところからダウンロードして得られたgeneレベルカウントデータ(`rse_gene.Rdata`; 約3MB)を読み込んで、カウントの数値行列情報(58,037 genes×11 samples)のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは[hoge3.txt](#)です。

```

in_f <- "rse_gene.Rdata"
out_f <- "hoge3.txt"

#必要なパッケージをロード
library(recount)

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge

#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)

#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F,

```

#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納

#パッ

#in_
#hog
#確認

#カウ
#行数
#確認

#保存



```

RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
> hoge
class: RangedSummarizedExperiment
dim: 58037 11
metadata(0):
assays(1): counts
rownames(58037): ENSG00000000003.14
  ENSG00000000005.5 ... ENSG00000283698.1
  ENSG00000283699.1
rowData names(3): gene_id bp_length symbol
colnames(11): SRR032116 SRR032118 ... SRR032126
  SRR032127
colData names(21): project sample ... title
  characteristics
> rowData(hoge)|

```


rowData(hoge)

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt"         #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount)            #パッ

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)                  #in
hoge <- rse_gene             #hog
hoge                         #確認

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カ
dim(data)                   #行数
head(data)                  #確認

#ファイルに保存
tmp <- cbind(rownames(data), data) #保存
write.table(tmp, out_f, sep="\t", append=F,

```

```

RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
> rowData(hoge)
DataFrame with 58037 rows and 3 columns
      gene_id bp_length      symbol
      <character> <integer> <CharacterList>
1      ENSG00000000003.14      4535      TSPAN6
2      ENSG00000000005.5      1610      TNMD
3      ENSG000000000419.12      1207      DPM1
4      ENSG000000000457.13      6883      SCYL3
5      ENSG000000000460.16      5967      Clorf112
...
58033  ENSG00000283695.1         61      NA
58034  ENSG00000283696.1        997      NA
58035  ENSG00000283697.1       1184      LOC101928917
58036  ENSG00000283698.1        940      NA

```

rowData(hoge)

①gene_idが抽出したカウントデータ行列の行名。②bp_lengthが配列長で、RPKM/FPKM値を得る際の基礎情報として使えます。③symbolがgene symbol情報です。機能解析(GO解析やパスウェイ解析)を行う際には、gene symbol情報で対応付けを行う必要があります。このような情報を保持しているRangedSummarizedExperimentオブジェクトを使いこなせると大変便利。

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上でSRP001558で検索し、gene列のRSE v2のデータ(`rse_gene.Rdata`; 約3MB)を読み込んで、カウントの数値行列情報としてファイルで保存するやり方です。出力ファイルは [hoge3.txt](#) です。

```

in_f <- "rse_gene.Rdata" #入力ファイル名
out_f <- "hoge3.txt" #出力ファイル名

#必要なパッケージをロード
library(recount) #パッケージ

#本番(typeで指定した名前の.Rdataをロード)
load(in_f) #in_f
hoge <- rse_gene #hog
hoge #確認

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウントデータ
dim(data) #行数
head(data) #確認

#ファイルに保存
tmp <- cbind(rownames(data), data) #保存
write.table(tmp, out_f, sep="\t", append=F,

```

R GUI (64-bit) window showing the R Console output:

```

> rowData(hoge)
DataFrame with 58036 rows and 3 columns
      gene_id bp_length symbol
      <character> <integer> <CharacterList>
1      ENSG00000000003.14      4535      TSPAN6
2      ENSG00000000005.5      1610      TNMD
3      ENSG000000000419.12      1207      DPM1
4      ENSG000000000457.13      6883      SCYL3
5      ENSG000000000460.16      5967      Clorf112
...
58033  ENSG00000283695.1      61      NA
58034  ENSG00000283696.1      997      NA
58035  ENSG00000283697.1      1184      LOC101928917
58036  ENSG00000283698.1      940      NA

```

Red arrows point to the `gene_id`, `bp_length`, and `symbol` columns in the output.

colnames

①もう一度hogeを表示。②colnamesという名前と赤枠内に表示されている情報から、カウントデータ行列の列名部分に相当するものだということが分かる。③で確認できるがやらなくてもよい。

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt"      #出力ファイル名を指定してout_fに格納
```

```
#必要なパッケージをロード
library(recount)
```

```
#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge
```

```
#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)
```

```
#ファイルに保存
tmp <- cbind(rownames(data), data) #保存
write.table(tmp, out_f, sep="\t", append=F,
```

```
R RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
> hoge
class: RangedSummarizedExperiment
dim: 58037 11
metadata(0):
assays(1): counts
rownames(58037): ENSG00000000003.14
  ENSG00000000005.5 ... ENSG00000283698.1
  ENSG00000283699.1
rowData names(3): gene_id bp_length symbol
colnames(11): SRR032116 SRR032118 ... SRR032126
  SRR032127
colData names(21): project sample ... title
characteristics
> colnames(hoge)|
```

これまでのノリから、①サンプルに相当する各列ごとに②21個の付随情報があるのではないかと予想する。③を実行すると一気に画面が流れるがやってみる。

colData

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt" #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount) #パッ

#本番(typeで指定した名前の.Rdataをロード)
load(in_f) #in_
hoge <- rse_gene #hog
hoge #確認

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウ
dim(data) #行数
head(data) #確認

#ファイルに保存
tmp <- cbind(rownames(data), data) #保存
write.table(tmp, out_f, sep="\t", append=F,

```

```

RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
> hoge
class: RangedSummarizedExperiment
dim: 58037 11
metadata(0):
assays(1): counts
rownames(58037): ENSG00000000003.14
ENSG00000000005.5 ... ENSG00000283698.1
ENSG00000283699.1
rowData names(3): gene_id bp_length symbol
colnames(11): SRR032116 SRR032118 ... SRR032126
SRR032127
colData names(21): project sample ... title
characteristics
> colData(hoge)

```

colData(hoge)実行結果。この画面サイズだと、最後の①characteristics列しか見えていない。

colData(hoge)

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt" #出力ファイル名を指定してout_fに格納

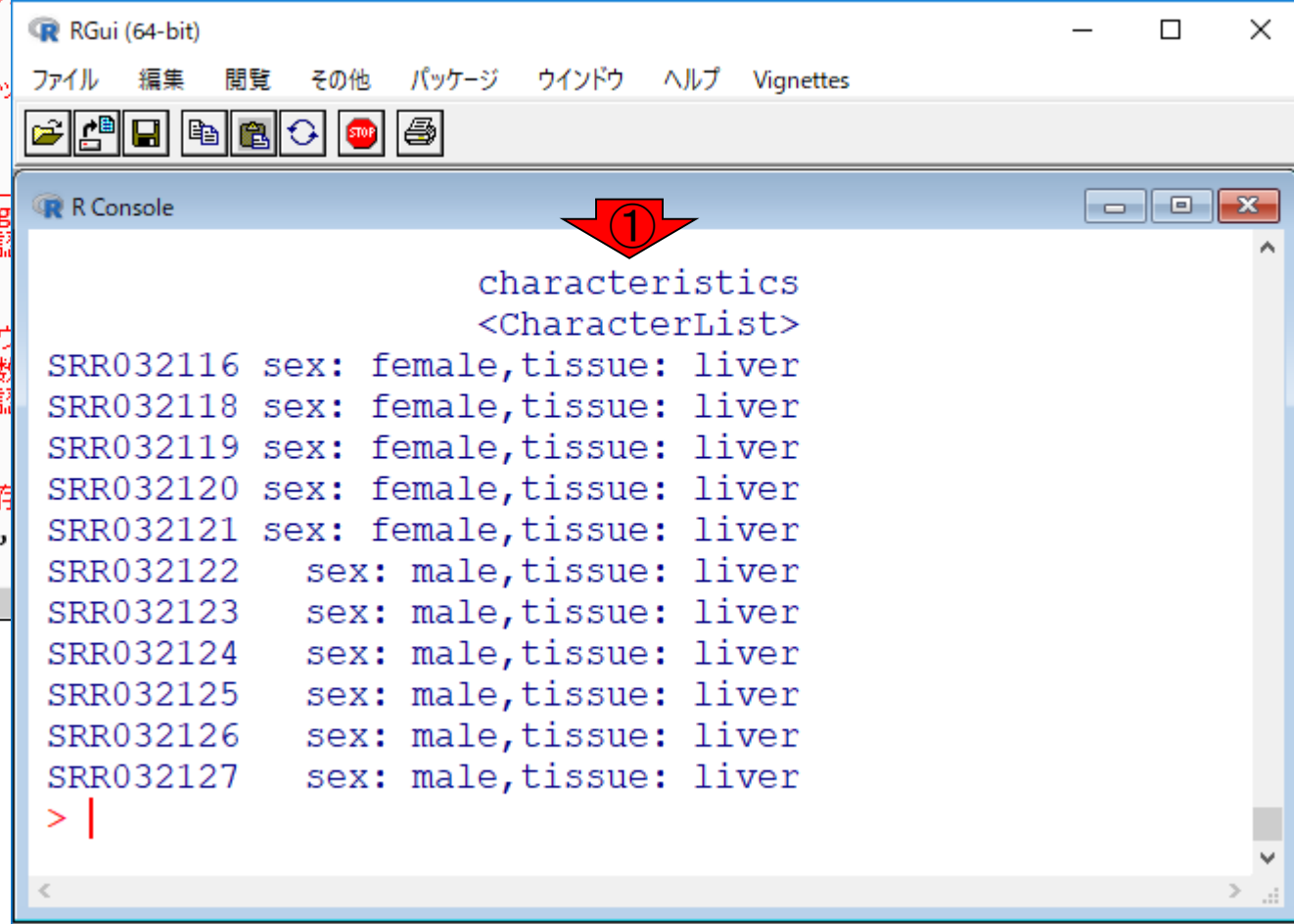
#必要なパッケージをロード
library(recount) #パッ

#本番(typeで指定した名前の.Rdataをロード)
load(in_f) #in_
hoge <- rse_gene #hog
hoge #確認

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウ
dim(data) #行数
head(data) #確認

#ファイルに保存
tmp <- cbind(rownames(data), data) #保存
write.table(tmp, out_f, sep="\t", append=F,

```



colData(hoge)

このあたり(じゃなくてもよいが)で、元々このデータは計12 samplesのはずだったのに、なぜ11 samplesしかないのだろう?!と思い始める。①femaleが5 samplesしかないので、femaleサンプルのうちの1つがカウントデータに含まれていないのだろうと判断する。

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報(58トファイル)で保存するやり方です。出力ファイルは `hoge3.txt` です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt" #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount) #パッ

#本番(typeで指定した名前の.Rdataをロード)
load(in_f) #in_
hoge <- rse_gene #hog
hoge #確認

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウ
dim(data) #行数
head(data) #確認

#ファイルに保存
tmp <- cbind(rownames(data), data) #保存
write.table(tmp, out_f, sep="\t", append=F,
```

```
RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
characteristics
<CharacterList>
SRR032116 sex: female,tissue: liver } ①
SRR032118 sex: female,tissue: liver
SRR032119 sex: female,tissue: liver
SRR032120 sex: female,tissue: liver
SRR032121 sex: female,tissue: liver
SRR032122 sex: male,tissue: liver } ②
SRR032123 sex: male,tissue: liver
SRR032124 sex: male,tissue: liver
SRR032125 sex: male,tissue: liver
SRR032126 sex: male,tissue: liver
SRR032127 sex: male,tissue: liver
> |
```

rse_gene.Rdata

おさらい。ずっと説明しているのは、ウェブサイト recount2から①RSE v2をクリックして得られた、② rse_gene.Rdataを読み込んで得られた

RangedSummarizedExperiment (RSE)形式のhogeオブジェクト。femaleサンプルのうちの1つがカウントデータに含まれていない理由は、③phenotype列のlinkから得られるファイルを眺めることでもなんとなくわかる。

recount2: analysis-ready R... x

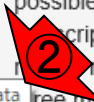
- Collado-Torres L, Nellore A, Jaffe AE. [recount workflow: Accessing over 70,000 NGS datasets](#). *bioRxiv*. 2018. doi: 10.1101/247346.
- Wilks C, Gaddipati P, Nellore A, Langmead B. [Snaptron: querying splicing patterns](#). *Bioinformatics*. 2018. doi: 10.1093/bioinformatics/btx547.
- Fu J, Kammers K, Nellore A, Collado-Torres L, Leek JT, Taub MA. [RNA-seq transcript quantification from reduced-representation data in recount2](#). *bioRxiv*, 2018. doi: 10.1101/247346.

The Datasets

Show 10 entries

Search: SRP001558

accession	number of samples	species	abstract	gene	exon	junctions	transcripts	phenotype	files info
SRP001558	12	human	Comparative studies of gene regulation suggest an important role for natural selection in shaping gene expression patterns within and between species. Most of these studies, however, estimated gene expression levels using microarray probes designed to hybridize to only a small proportion of each gene. Here we used recently-developed RNA sequencing protocols, which side-step this limitation, to assess intra- and inter-species variation in gene regulatory processes in considerably more detail than was previously possible. Specifically, we used RNAseq to study transcript levels in humans, chimpanzees, and macaques, using liver RNA samples from three males and three females from each species.	RSE v2 counts v2	RSE v1 counts v1	RSE v2 counts v2	RSE v1 counts v1	RSE v2 RSE v1	link



http://duffel.rail.bio/recount/v2/SRP001558/rse_gene.Rdata

colData(hoge)

① colData(hoge)の行数と列数を把握すべく、② dimを実行。キーボードの上下左右の矢印キーを駆使して効率的に打ち込んでいますよね?!

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上でSRP001558で検索し、`gene`列のRSE v2のところからダウンロードして得られた `gene`レベルカウントデータ(`rse_gene.Rdata`; 約3MB)を読み込んで、カウントの数値行列情報(58,037 genes×11 samples)のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge3.txt" #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount) #パ

#本番(typeで指定した名前の.Rdataをロード)
load(in_f) #in_
hoge <- rse_gene #hog
hoge #確認

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウ
dim(data) #行数
head(data) #確認

#ファイルに保存
tmp <- cbind(rownames(data), data) #保存
write.table(tmp, out_f, sep="\t", append=F,

```

RGui (64-bit) window showing the R Console output:

```

characteristics
<CharacterList>
SRR032116 sex: female,tissue: liver
SRR032118 sex: female,tissue: liver
SRR032119 sex: female,tissue: liver
SRR032120 sex: female,tissue: liver
SRR032121 sex: female,tissue: liver
SRR032122 sex: male,tissue: liver
SRR032123 sex: male,tissue: liver
SRR032124 sex: male,tissue: liver
SRR032125 sex: male,tissue: liver
SRR032126 sex: male,tissue: liver
SRR032127 sex: male,tissue: liver
> dim(colData(hoge))

```



dim(colData(hoge))

① colData(hoge)は、11行×21列の情報からなる。これだけの情報量になると②R Console画面上で判断するのは難しいのでファイルに保存してExcelで眺めることにする。それが例題4。

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上でSRP001558で検索し、`gene`列のRSE v2のところからダウンロードして得られた `gene`レベルカウントデータ(`rse_gene.Rdata`; 約3MB)を読み込んで、カウントの数値行列情報(58,037 genes×11 samples)のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata"
out_f <- "hoge3.txt"

#必要なパッケージをロード
library(recount)

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge

#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)

#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F,

```

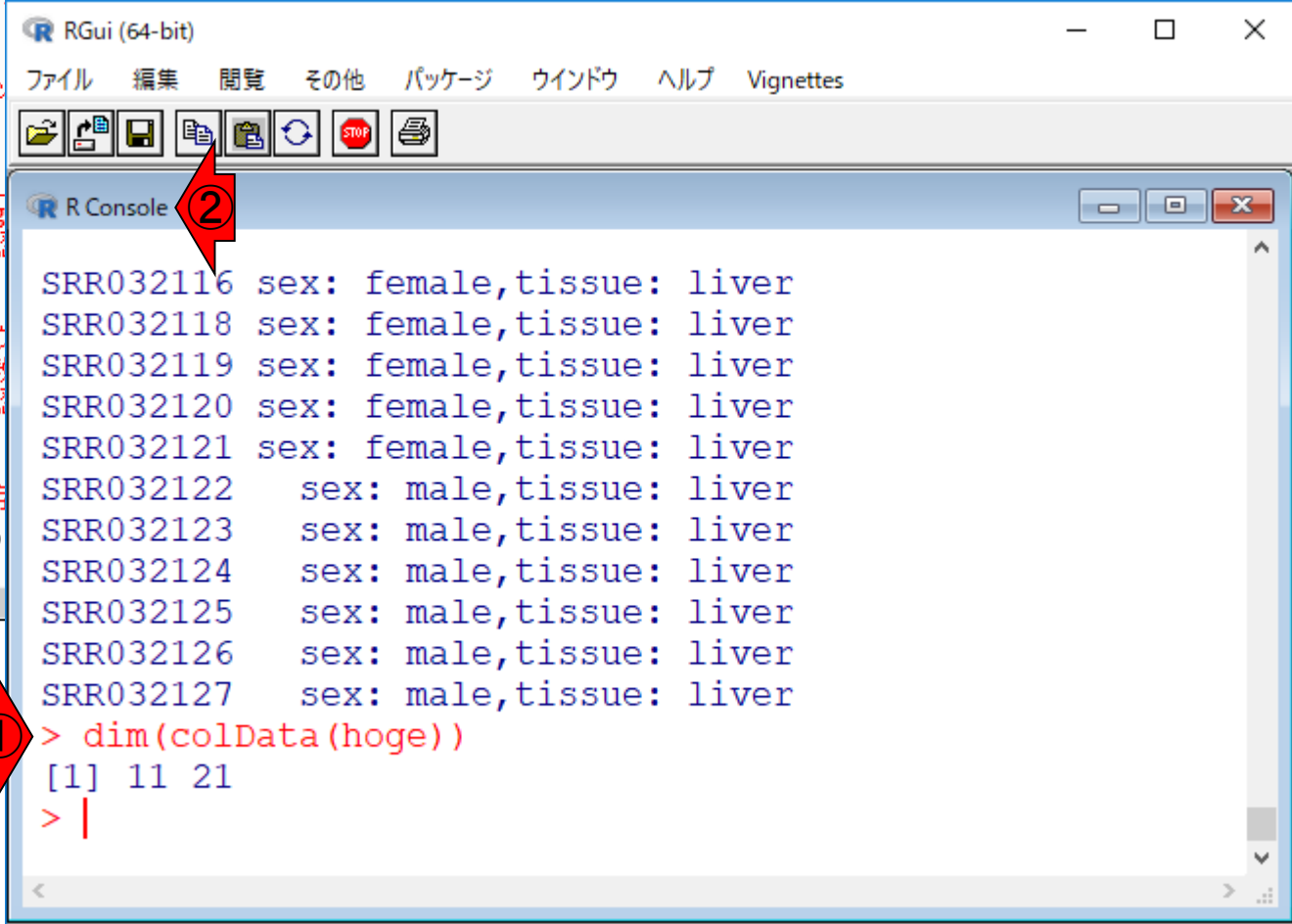
#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納

#パッ

#in_
#hog
#確認

#カウ
#行数
#確認

#保存



①

②

Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?! カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

ここで着目してほしいのは、①RSE形式のhogeオブジェクトから、②サンプルのメタデータ情報をファイル(hoge4_meta_samples.txt)に落とすところのみ。③コードの下部に移動。

例題4

4. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題3とベースとして、さらにサンプルのメタデータ情報ファイル(`hoge4_meta_samples.txt`)と、遺伝子(features)のメタデータ情報ファイル(`hoge4_meta_features.txt`)も出力するやり方です。58,037 genes×11 samplesからなるカウントデータファイル(`hoge4_counts.txt`)は列名をSRR...からSRS...に変更しています。このデータセットの場合は、なぜかtechnical replicatesのサンプルに対して別々のSRS IDが付与されているので、列名変更はほぼ無意味です。

```

in_f <- "rse_gene.Rdata"
out_f1 <- "hoge4_counts.txt"
out_f2 <- "hoge4_meta_samples.txt"
out_f3 <- "hoge4_meta_features.txt"

#必要なパッケージをロード
library(recount)

#入力ファイルの読み込み
load(in_f)
hoge <- rse_gene
hoge

#本データセット(カウントデータ取得)
data <- assays(hoge)$counts
colnames(data) <- colData(hoge)$sample
dim(data)
head(data)

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f1, sep="\t", append=F, quote=F, row.names=F)
    
```

① #入力ファイル名を指定してin_fに格納
② #出力ファイル名を指定してout_f1に格納(カウントデータ)
 #出力ファイル名を指定してout_f2に格納(samplesメタデータ)
 #出力ファイル名を指定してout_f3に格納(featuresメタデータ)
 #パッケージの読み込み
 #in_fで指定した.Rdataをロード
 #hogeとして取り扱う
 #確認してるだけです
 #カウントデータ行列を取得してdataに格納
 #列名をERR...からERS...に変更
 #行数と列数を表示
 #確認してるだけです
 #保存したい情報をtmpに格納
 #tmpの中身を指定したファイル名



例題4

4. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題3とベースとして、さらにサンプルのメタデータ情報ファイル(`hoge4 meta samples.txt`)と、遺伝子(`features`)のメタデータ情報ファイル(`hoge4 meta features.txt`)も出力するやり方です。58,037 genes×11 samplesからなるカウントデータファイル(`hoge4 counts.txt`)は列名をSRR...からSRS...に変更しています。このデータセットの場合は、なぜかtechnical replicatesのサンプルに対して別々のSRS IDが付与されているので、列名変更はほぼ無意味です。

```
load(in_f) #in_fで指定した.Rdataをロード
hoge <- rse_gene #hogeとして取り扱う
hoge #確認してるだけです

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウントデータ行列を取得してdataに格納
colnames(data) <- colData(hoge)$sample #列名をERR...からERS...に変更
dim(data) #行数と列数を表示
head(data) #確認してるだけです

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data) #保存したい情報をtmpに格納
write.table(tmp, out_f1, sep="\t", append=F, quote=F, row.names=F) #tmpの中身を指定したファイル名

#ファイルに保存(サンプルのメタデータ情報)
tmp <- colData(hoge) #保存したい情報をtmpに格納
write.table(tmp, out_f2, sep="\t", append=F, quote=F, row.names=F) #tmpの中身を指定したファイル名

#ファイルに保存(featuresのメタデータ情報)
tmp <- rowData(hoge) #保存したい情報をtmpに格納
write.table(tmp, out_f3, sep="\t", append=F, quote=F, row.names=F) #tmpの中身を指定したファイル名
```

①

例題4

①赤枠部分が今着目してもらいたいところ。②hogeは、`rse_gene.Rdata`を読み込んで得られた `RangedSummarizedExperiment (RSE)`形式のオブジェクト。③`colData(hoge)`の中身を、そのまま④`out_f2 (hoge4_meta_samples.txt)`のことに⑤タブ区切りテキスト形式で保存している。

4. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題3とベースとして、さらにサンプルのメタデータ情報ファイル(`hoge4_meta_samples.txt`)と特徴量ファイル(`hoge4_meta_features.txt`)も出力するやり方です。58,037 genes×11 samplesの `rse_gene.Rdata` (`hoge4_counts.txt`)は列名をSRR...からSRS...に変更しています。このデータセットの場合、サンプルに対して別々のSRS IDが付与されているので、列名変更はほぼ無意味です。

```

load(in_f) #in_fで指定した.Rdataをロード
hoge <- rse_gene #hogeとして取り扱う
hoge #確認してるだけです

#本題(カウントデータ取得)
data <- assays(hoge)$counts #カウントデータ行列を取得してdataに格納
colnames(data) <- colData(hoge)$sample #列名をERR...からERS...に変更
dim(data) #行数と列数を表示
head(data) #確認してるだけです

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data) #保存したい情報をtmpに格納
write.table(tmp, out_f1, sep="\t", append=F, quote=F, row.names=F) #tmpの中身を指定したファイル名

#ファイルに保存(サンプル③メタデータ情報)
tmp <- colData(hoge) #保存したい情報をtmpに格納
write.table(tmp, out_f2, sep="\t", append=F, quote=F, row.names=F) #tmpの中身を指定したファイル名

#ファイルに保存(特徴量④のメタ⑤情報)
tmp <- rowData(hoge) #保存したい情報をtmpに格納
write.table(tmp, out_f3, sep="\t", append=F, quote=F, row.names=F) #tmpの中身を指定したファイル名
    
```

赤枠部分が、ヘッダー行を除くと11行×21列の colData(hoge)の中身を、①(hoge4_meta_samples.txt)に保存した結果。Excelで読み込ませて、今は注目に値しない列の幅を狭めて表示させたものです。

colData(hoge)

4. ダウンロード済みの rse_gene.Rdata を入力として読み込む場合:

例題3とベースとして、さらにサンプルのメタデータ情報ファイル(hoge4_meta_samples.txt)と、遺伝子(features)のメタデータ情報ファイル(hoge4_meta_features.txt)も出力するやり方です。58,037 genes×11 samplesからなるカウントデータファイル(hoge4_counts.txt)は列名をSRR...からSRS...に変更しています。このデータセットの場合は、なぜかtechnical replicatesのサンプルに対して別々のSRS IDが付与されているので、列名変更はほぼ無意味です。

```
in_f <- "rse_gene.Rdata"
out_f1 <- "hoge4_counts.txt"
out_f2 <- "hoge4_meta_samples.txt"
out_f3 <- "hoge4_meta_features.txt"
```



#入力ファイル名を指定してin_fに格納
 #出力ファイル名を指定してout_f1に格納(カウントデータ)
 #出力ファイル名を指定してout_f2に格納(samplesメタデータ)
 #出力ファイル名を指定してout_f3に格納(featuresメタデータ)

```
#必要なパッケージをロード
library(recount) #パッケージの読み込み
```

#入力ファイルの読み込み

```
load(in_f)
hoge <- readRDS("hoge4.Rdata")
colnames(hoge) <- c("A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M", "N", "O", "P", "Q", "R", "S", "T", "U")
dim(hoge)
head(hoge)
#ファイル名をtmpに保存
write.csv(hoge, "tmp.csv")
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	pro	sample	exp	run	rea	rea	prc	pai	sra	ma	auc	sh	sh	bio	bio	bio	avg	gc	big	title	characteristics
2	SRF	SRS009313	SR	SRR032116	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 1 rep1	c("sex: female", "tissue: liver")
3	SRF	SRS009315	SR	SRR032118	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 2 rep1	c("sex: female", "tissue: liver")
4	SRF	SRS009316	SR	SRR032119	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 2 rep2	c("sex: female", "tissue: liver")
5	SRF	SRS009317	SR	SRR032120	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 3 rep1	c("sex: female", "tissue: liver")
6	SRF	SRS009318	SR	SRR032121	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 3 rep2	c("sex: female", "tissue: liver")
7	SRF	SRS009319	SR	SRR032122	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 1 rep1	c("sex: male", "tissue: liver")
8	SRF	SRS009320	SR	SRR032123	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 1 rep2	c("sex: male", "tissue: liver")
9	SRF	SRS009321	SR	SRR032124	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 2 rep1	c("sex: male", "tissue: liver")
10	SRF	SRS009322	SR	SRR032125	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 2 rep2	c("sex: male", "tissue: liver")
11	SRF	SRS009323	SR	SRR032126	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 3 rep1	c("sex: male", "tissue: liver")
12	SRF	SRS009324	SR	SRR032127	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 3 rep2	c("sex: male", "tissue: liver")

colData(hoge)

赤枠部分が、ヘッダー行を除くと11行×21列の colData(hoge)の中身を、①(hoge4_meta_samples.txt)に保存した結果。Excelで読み込ませて、今は注目に値しない列の幅を狭めて表示させたものです。②が colData(hoge)\$sample、③が colData(hoge)\$run、そして④が colData(hoge)\$title で取り出せる情報となる。

4. ダウンロード済みの rse_gene.Rdata を入力として読み込む場合:

例題3とベースとして、さらにサンプルのメタデータ情報ファイル(hoge4_meta_samples.txt)も出力するやり方です。58,037 genes×11 samples(hoge4_counts.txt)は列名をSRR...からSRS...に変更しています。このデータセットに対して別々のSRS IDが付与されているので、列名変更はほぼ無意味

```
in_f <- "rse_gene.Rdata"
out_f1 <- "hoge4_counts.txt"
out_f2 <- "hoge4_meta_samples.txt"
out_f3 <- "hoge4_meta_features.txt"
```

```
#必要なパッケージをロード
library(recount) #パッケージの読み込み
```

#入力ファイルの読み込み

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	pro	sample	exp	run	rea	rea	prc	pai	sra	ma	auc	sh	sh	bio	bio	bio	avg	gc	big	title	characteristics
2	SRF	SRS009313	SR	SRR032116	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 1 rep1	c("sex: female", "tissue: liver")
3	SRF	SRS009315	SR	SRR032118	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 2 rep1	c("sex: female", "tissue: liver")
4	SRF	SRS009316	SR	SRR032119	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 2 rep2	c("sex: female", "tissue: liver")
5	SRF	SRS009317	SR	SRR032120	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 3 rep1	c("sex: female", "tissue: liver")
6	SRF	SRS009318	SR	SRR032121	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 3 rep2	c("sex: female", "tissue: liver")
7	SRF	SRS009319	SR	SRR032122	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 1 rep1	c("sex: male", "tissue: liver")
8	SRF	SRS009320	SR	SRR032123	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 1 rep2	c("sex: male", "tissue: liver")
9	SRF	SRS009321	SR	SRR032124	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 2 rep1	c("sex: male", "tissue: liver")
10	SRF	SRS009322	SR	SRR032125	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 2 rep2	c("sex: male", "tissue: liver")
11	SRF	SRS009323	SR	SRR032126	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 3 rep1	c("sex: male", "tissue: liver")
12	SRF	SRS009324	SR	SRR032127	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 3 rep2	c("sex: male", "tissue: liver")

colData(hoge)\$sample

例えば、①colData(hoge)\$sampleの実行結果は、確かに②列の情報と同じです。これは、例題4のカウントデータファイル③hoge4_counts.txtの列名として使われています。

4. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題3とベースとして、さらにサンプルのメタデータ情報ファイル(`hoge4 meta samples.txt`)と、遺伝子(`hoge4 meta features.txt`)も出力するやり方です。58,037 genes×11 samplesからなるカウントデータ(`hoge4 counts.txt`)は列名をSRR...からSRS...に変更しています。このデータセットの場合は、なぜかtechnical replicatesのサンプルに対して別々のSRS IDが付与されているので、列名変更はほぼ無意味です。

```
in_f <- "rse_gene.Rdata"
out_f1 <- "hoge4_counts.txt"
out_f2 <- "hoge4_meta_samples.txt"
out_f3 <- "hoge4_meta_features.txt"
```

```
#必要なパッケージをロード
library(recount)
```

```
#入力ファイルの読み込み
```

	A	B	C	D	E	F
1	pro	sample	exp	run		rearea
2	SRF	SRS009313	SR	SRR032116	##	##
3	SRF	SRS009315	SR	SRR032118	##	##
4	SRF	SRS009316	SR	SRR032119	##	##
5	SRF	SRS009317	SR	SRR032120	##	##
6	SRF	SRS009318	SR	SRR032121	##	##
7	SRF	SRS009319	SR	SRR032122	##	##
8	SRF	SRS009320	SR	SRR032123	##	##
9	SRF	SRS009321	SR	SRR032124	##	##
10	SRF	SRS009322	SR	SRR032125	##	##
11	SRF	SRS009323	SR	SRR032126	##	##
12	SRF	SRS009324	SR	SRR032127	##	##

#入力ファイル名を指定してin_fに格納

#出力

#出力

#出力

#パッ

```
R Console
SRR032121 sex: female,tissue: liver
SRR032122 sex: male,tissue: liver
SRR032123 sex: male,tissue: liver
SRR032124 sex: male,tissue: liver
SRR032125 sex: male,tissue: liver
SRR032126 sex: male,tissue: liver
SRR032127 sex: male,tissue: liver
> dim(colData(hoge))
[1] 11 21
> colData(hoge)$sample
[1] "SRS009313" "SRS009315" "SRS009316" "SRS009317"
[5] "SRS009318" "SRS009319" "SRS009320" "SRS009321"
[9] "SRS009322" "SRS009323" "SRS009324"
> |
```


①の情報で、行列dataの列名に相当するcolnames(data)を置換しているの...

カウントデータの列名

4. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題3とベースとして、さらにサンプルのメタデータ情報ファイル(`hoge4 meta samples.txt`)と、遺伝子(features)のメタデータ情報ファイル(`hoge4 meta features.txt`)も出力するやり方です。58,037 genes×11 samplesからなるカウントデータファイル(`hoge4 counts.txt`)は列名をSRR...からSRS...に変更しています。このデータセットの場合は、なぜかtechnical replicatesのサンプルに対して別々のSRS IDが付与されているので、列名変更はほぼ無意味です。

```

in_f <- "rse_gene.Rdata"
out_f1 <- "hoge4_counts.txt"
out_f2 <- "hoge4_meta_samples.txt"
out_f3 <- "hoge4_meta_features.txt"

#必要なパッケージをロード
library(recount)

#入力ファイルの読み込み
load(in_f)
hoge <- rse_gene
hoge

#本番(カウントデータ取得)
data <- assays(hoge)$counts
colnames(data) <- colData(hoge)$sample
dim(data)
head(data)

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f1, sep="\t", append=TRUE)

```

#入力ファイル名を指定してin_fに格納

#出力
#出力
#出力

#必要なパッケージをロード

#パッケージをロード

#入力ファイルの読み込み

#in_fに格納
#hogeに代入
#確認

#本番(カウントデータ取得)

#カウントデータ取得
#列名変更
#行数確認
#確認



```

RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
SRR032121 sex: female,tissue: liver
SRR032122 sex: male,tissue: liver
SRR032123 sex: male,tissue: liver
SRR032124 sex: male,tissue: liver
SRR032125 sex: male,tissue: liver
SRR032126 sex: male,tissue: liver
SRR032127 sex: male,tissue: liver
> dim(colData(hoge))
[1] 11 21
> colData(hoge)$sample
[1] "SRS009313" "SRS009315" "SRS009316" "SRS009317"
[5] "SRS009318" "SRS009319" "SRS009320" "SRS009321"
[9] "SRS009322" "SRS009323" "SRS009324"
> |

```

カウントデータの列名

①の情報で、行列dataの列名に相当するcolnames(data)を置換している...
②列名変更後の行列dataの最初の2行分で見えているような状態に、③がなっています。

4. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題3とベースとして、さらにサンプルのメタデータ情報ファイル(`hoge4 meta samples.txt`)と、遺伝子ファイル(`hoge4 meta features.txt`)も出力するやり方です。58,037 genes×11 samplesからなるカウントデータ(`hoge4 counts.txt`)は列名をSRR...からSRS...に変更しています。このデータセットの場合は、なぜかtechnical replicatesのサンプルに対して別々のSRS IDが付与されているので、列名変更はほぼ無意味です。

```

in_f <- "rse_gene.Rdata"
out_f1 <- "hoge4_counts.txt"
out_f2 <- "hoge4_meta_samples.txt"
out_f3 <- "hoge4_meta_features.txt"

#必要なパッケージをロード
library(recount)

#入力ファイルの読み込み
load(in_f)
hoge <- rse_gene
hoge

#本番(カウントデータ取得)
data <- assays(hoge)$counts
colnames(data) <- colData(hoge)$sample
dim(data)
head(data)

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f1, sep="\t", append=F)

```



#入力ファイル名を指定してin_fに格納

#出力

#出力

#出力

#パッ

#i

#h

#確認

#カウ

#列名

#行数

#確認

#保存



```

> head(data, n=2)
      SRS009313 SRS009315 SRS009316
ENSG00000000003.14 7690      6538      6780
ENSG00000000005.5      0         0         0
      SRS009317 SRS009318 SRS009319
ENSG00000000003.14 3359      3702      3201
ENSG00000000005.5      0         0         0
      SRS009320 SRS009321 SRS009322
ENSG00000000003.14 2812      9053      8005
ENSG00000000005.5      0         35         0
      SRS009323 SRS009324
ENSG00000000003.14 8237      6866
ENSG00000000005.5      0         0
> |

```

Tips

参考

①の列をみればわかるが、例えば②はHSF2に相当する同一サンプル、つまりtechnical replicatesである。しかしながら、別々のSRS ID(SRS009315とSRS009316)が割り振られている。おそらくこれはNGSデータを公共DBに登録し始めた初期のデータだから、完全に方針が定まっていなかったことに起因すると思われる。どのように登録するかはsubmitterに大きく依存する。recountは公共DB中の情報を取りに行っているだけ。

4. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題3とベースとして、さらにサンプルのメタデータ情報ファイル(`hoge4 meta` ファイル(`hoge4 meta features.txt`)も出力するやり方です。58,037 genes×11 (`hoge4 counts.txt`)は列名をSRR...からSRS...に変更しています。このデータに対して別々のSRS IDが付与されているので、列名変更はほぼ無意味

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してout_f3に格納(featuresメタデータ)
out_f1 <- "hoge4_counts.txt" #出力ファイル名を指定してout_f1に格納(countsメタデータ)
out_f2 <- "hoge4_meta_samples.txt" #出力ファイル名を指定してout_f2に格納(samplesメタデータ)
out_f3 <- "hoge4_meta_features.txt" #出力ファイル名を指定してout_f3に格納(featuresメタデータ)
```

```
#必要なパッケージをロード
library(recount) #パッケージの読み込み
```

#入力ファイルの読み込み

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	pro	sample	exp	run	rea	rea	prc	pai	sra	ma	auc	sh	sh	bio	bio	bio	avg	gc	big	title	characteristics
2	SRF	SRS009313	SR	SRR032116	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 1 rep1	c("sex: female", "tissue: liver")
3	SRF	SRS009315	SR	SRR032118	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 2 rep1	c("sex: female", "tissue: liver")
4	SRF	SRS009316	SR	SRR032119	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 2 rep2	c("sex: female", "tissue: liver")
5	SRF	SRS009317	SR	SRR032120	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 3 rep1	c("sex: female", "tissue: liver")
6	SRF	SRS009318	SR	SRR032121	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 3 rep2	c("sex: female", "tissue: liver")
7	SRF	SRS009319	SR	SRR032122	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 1 rep1	c("sex: male", "tissue: liver")
8	SRF	SRS009320	SR	SRR032123	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 1 rep2	c("sex: male", "tissue: liver")
9	SRF	SRS009321	SR	SRR032124	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 2 rep1	c("sex: male", "tissue: liver")
10	SRF	SRS009322	SR	SRR032125	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 2 rep2	c("sex: male", "tissue: liver")
11	SRF	SRS009323	SR	SRR032126	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 3 rep1	c("sex: male", "tissue: liver")
12	SRF	SRS009324	SR	SRR032127	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 3 rep2	c("sex: male", "tissue: liver")

```
#本番の読み込み
data <- load(in_f)
colnames(data) <- out_f1
dim(data) <- out_f2
head(data)
#ファイルの書き込み
tmp <- write.csv(data, out_f3)
```

① colData(hoge)\$titleの情報が、サンプル間クラスタリングを行った際にわかりやすいと判断したので、それを行っているのが例題5。

colData(hoge)\$title

4. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題3とベースとして、さらにサンプルのメタデータ情報ファイル(`hoge4 meta samples.txt`)と、遺伝子(features)のメタデータ情報ファイル(`hoge4 meta features.txt`)も出力するやり方です。58,037 genes×11 samplesからなるカウントデータファイル(`hoge4 counts.txt`)は列名をSRR...からSRS...に変更しています。このデータセットの場合は、なぜかtechnical replicatesのサンプルに対して別々のSRS IDが付与されているので、列名変更はほぼ無意味です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f1 <- "hoge4_counts.txt" #出力ファイル名を指定してout_f1に格納(カウントデータ)
out_f2 <- "hoge4_meta_samples.txt" #出力ファイル名を指定してout_f2に格納(samplesメタデータ)
out_f3 <- "hoge4_meta_features.txt" #出力ファイル名を指定してout_f3に格納(featuresメタデータ)
```

```
#必要なパッケージをロード
library(recount) #パッケージの読み込み
```

#入力ファイルの読み込み

```
load(in_f)
hoge <- readRDS("hoge4_counts.txt")
hoge$colnames <- readLines(out_f2)
dim(hoge)
head(hoge)
#ファイル名をtmpに書き出す
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	pro	sample	exp	run	rea	rea	prc	pai	sra	ma	auc	sh	sh	bio	bio	bio	avg	gc	big	title	characteristics
2	SRF	SRS009313	SR	SRR032116	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 1 rep1	c("sex: female", "tissue: liver")
3	SRF	SRS009315	SR	SRR032118	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 2 rep1	c("sex: female", "tissue: liver")
4	SRF	SRS009316	SR	SRR032119	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 2 rep2	c("sex: female", "tissue: liver")
5	SRF	SRS009317	SR	SRR032120	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 3 rep1	c("sex: female", "tissue: liver")
6	SRF	SRS009318	SR	SRR032121	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human female 3 rep2	c("sex: female", "tissue: liver")
7	SRF	SRS009319	SR	SRR032122	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 1 rep1	c("sex: male", "tissue: liver")
8	SRF	SRS009320	SR	SRR032123	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 1 rep2	c("sex: male", "tissue: liver")
9	SRF	SRS009321	SR	SRR032124	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 2 rep1	c("sex: male", "tissue: liver")
10	SRF	SRS009322	SR	SRR032125	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 2 rep2	c("sex: male", "tissue: liver")
11	SRF	SRS009323	SR	SRR032126	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 3 rep1	c("sex: male", "tissue: liver")
12	SRF	SRS009324	SR	SRR032127	##	##	1	##	##	##	##	live	cor	20	(20	(20	35	GS	SR	Human male 3 rep2	c("sex: male", "tissue: liver")



Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?! カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

例題5

例題5では①のようにしてtitle列の情報を採用したが…②でも書いているように、いつもここに有意義な情報があるとは限らないので注意。

5. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題4で得られたサンプルのメタデータ情報ファイル([hoge4 meta samples.txt](#))中のtitle列に相当する情報で置き換えています。これは、[hoge4 meta samples.txt](#)をExcelで眺めたときに、たまたまtitle列情報がdiscriminable(容易に識別可能である)だと主観的に判断したためです。このあたりの情報のクオリティとかどのような情報が提供されているかは、submitter依存です。したがって、一筋縄ではいきません。まるで有益な情報のない残念なものも結構あるからです。58,037 genes×11 samplesからなる出力ファイルは[hoge5.txt](#)です。



```

in_f <- "rse_gene.Rdata"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge5.txt"         #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount)           #パッケージの読み込み

#入力ファイルの読み込み(.Rdata)
load(in_f)                 #in_fで指定した.Rdataをロード
hoge <- rse_gene           #hogeとして取り扱う
                             #確認してるだけです

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウントデータ行列を取得してdataに格納
dim(data)                  #行数と列数を表示
head(data)                 #確認してるだけです

#後処理(列名を変更)
colnames(data) <- colData(hoge)$title #列名を変更
head(data)                 #確認してるだけです

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data) #保存したい情報をtmpに格納
    
```



例題5をコピー実行して、①のような列名になった、②hoge5.txtを得ておきましょう。

例題5をコピー実行

5. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題4で得られたサンプルのメタデータ情報ファイル([hoge4 meta samples.txt](#))中のtitle列に相当する情報で置き換えています。これは、[hoge4 meta samples.txt](#)をExcelで眺めたときに、たまたまtitle列情報がdiscriminable(容易に識別可能である)だと主観的に判断したためです。このあたりの情報のクオリティとかどのような情報が提供されているかは、submitter依存です。したがって、一筋縄ではいきません。まるで有益な情報のない残念なものも結構あるからです。58,037 genes×11 samplesからなる出力ファイルは[hoge5.txt](#)です。

```

in_f <- "rse_gene.Rdata"
out_f <- "hoge5.txt"

#必要なパッケージをロード
library(recount)

#入力ファイルの読み込み(.Rdata)
load(in_f)
hoge <- rse_gene
hoge

#本番(カウントデータ取得)
data <- assays(hoge)$counts
dim(data)
head(data)

#後処理(列名を変更)
colnames(data) <- colData(hoge)$title
head(data)

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data)

```

```

RGui (64-bit)
ファイル 編集 閲覧 その他 パッケージ ウィンドウ ヘルプ Vignettes

R Console
ENSG00000000460.16      321      844
ENSG00000000938.12    1983    1263
                        Human male 3 rep1 Human male 3 rep2
ENSG00000000003.14    8237    6866
ENSG00000000005.5      0        0
ENSG00000000419.12    1358    1339
ENSG00000000457.13    2658    2282
ENSG00000000460.16    1206     711
ENSG00000000938.12    2238    2504
>
> #ファイルに保存(カウントデータ)
> tmp <- cbind(rownames(data), data)      #保存したい情$
> write.table(tmp, out_f, sep="\t", append=F, quote=F,$
> |

```



①例題6は、②technical replicates(同一個体の反復データ)をマージして、③58,037 genes × 6 samples のカウントデータ行列にするコード。コピペ実行。

例題6

①

②

③

6. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題5の続きのようなものですが、technical replicatesのデータをマージした結果を出力しています。例えば、"Human female 2 rep1"列と"Human female 2 rep2"列のカウント数の和をとり、列名を"HSF2"のようにしています。この列名の表記法は、「サンプルデータ42の20,689 genes×18 samplesのリアルカウントデータ ([sample blekman 18.txt](#))」中のヒトサンプル名と同じにしています。58,037 genes×6 samplesからなる出力ファイルは `hoge6.txt` です。

```

in_f <- "rse_gene.Rdata"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge6.txt"         #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount)            #パッケージの読み込み

#入力ファイルの読み込み(.Rdata)
load(in_f)                  #in_fで指定した.Rdataをロード
hoge <- rse_gene             #hogeとして取り扱う
hoge                         #確認してるだけです

#本番(カウントデータ取得)
uge <- assays(hoge)$counts  #カウントデータ行列を取得してugeに格納
dim(uge)                    #行数と列数を表示
head(uge)                   #確認してるだけです

#後処理(technical replicatesの列をマージ)
uge <- as.data.frame(uge)   #行列形式からデータフレーム形式に変更
data <- cbind(              #必要な列名を取得したい列の順番で結合した結果:
  uge$SRR032116,           #HSF1
  uge$SRR032118 + uge$SRR032119, #HSF2
  uge$SRR032120 + uge$SRR032121, #HSF3
)
    
```


例題6をコピー実行して、①のような列名になった、②hoge6.txtを得ておきましょう。

例題6

6. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題5の続きのようなものですが、technical replicatesのデータをマージした結果を出力しています。例えば、"Human female 2 rep1"列と"Human female 2 rep2"列のカウント数の和をとり、列名を"HSF2"のようにしています。この列名の表記法は、「サンプルデータ42の20,689 genes×18 samplesのリアルカウントデータ ([sample blekhman 18.txt](#))」中のヒトサンプル名と同じにしています。58,037 genes×6 samplesからなる出力ファイルは `hoge6.txt` です。

```

in_f <- "rse_gene.Rdata"
out_f <- "hoge6.txt"

#必要なパッケージをロード
library(recount)

#入力ファイルの読み込み(.Rdata)
load(in_f)
hoge <- rse_gene
hoge

#本番(カウントデータ取得)
uge <- assays(hoge)$counts
dim(uge)
head(uge)

#後処理(technical replicatesの列をマージ)
uge <- as.data.frame(uge)
data <- cbind(
  uge$SRR032116,
  uge$SRR032118 + uge$SRR032119,
  uge$SRR032120 + uge$SRR032121,

```

#入力ファイル名を指定してin_fに格納

#出力

#パッ

#in_

#hog

#確認

#カウ

#行数

#確認

#行列

#必要

#HSF

#HSF

#HSF

```

> head(data)
#確認してるだ$
      HSF1  HSF2  HSF3  HSM1  HSM2  HSM3
ENSG000000000003.14 7690 13318 7061 6013 17058 15103
ENSG000000000005.5    0     0     0     0     35     0
ENSG0000000000419.12 1501 2727 2749 2074 3901 2697
ENSG0000000000457.13 1845 2915 3703 5492 3707 4940
ENSG0000000000460.16 508 1662 1477 1777 1165 1917
ENSG0000000000938.12 1615 3991 13736 4736 3246 4742
>
> #ファイルに保存(カウントデータ)
> tmp <- cbind(rownames(data), data) #保存したい情$
> write.table(tmp, out_f, sep="\t", append=F, quote=F,$
> |

```

遺言

参考

利用したいRパッケージのマニュアルでは、サンプルデータの列名変更などは最初のほうに説明されている。本来①のあたりは本質的なところではない。しかしながら、慣れないと非常に難解であり、しかもマニュアル中の説明はそれほど丁寧ではない。それゆえ、今回詳述したようなcolData(hoge)を自分で眺めてうまく対処するノリに慣れるのが重要です！



5. ダウンロード済みのrse_gene.Rdataを入力として読み込む場合:

例題4で得られたサンプルのメタデータ情報ファイル([hoge4 meta samples.txt](#))に代わっています。これは、[hoge4 meta samples.txt](#)をExcelで眺めたときに、たまたまに識別可能である)だと主観的に判断したためです。このあたりの情報のク提供されているかは、submitter依存です。したがって、一筋縄ではいきません。のも結構あるからです。58,037 genes×11 samplesからなる出力ファイルは[hoge](#)

```

in_f <- "rse_gene.Rdata"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge5.txt"         #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount)           #パッケージの読み込み

#入力ファイルの読み込み(.Rdata)
load(in_f)                 #in_fで指定した.Rdataをロード
hoge <- rse_gene           #hogeとして取り扱う
                             #確認してるだけです

#本番(カウントデータ取得)
data <- assays(hoge)$counts #カウントデータ行列を取得してdataに格納
dim(data)                  #行数と列数を表示
head(data)                  #確認してるだけです

#後処理(列名を変更)
colnames(data) <- colData(hoge)$title #列名を変更
head(data)                 #確認してるだけです

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data) #保存したい情報をtmpに格納
    
```



①

Rパッケージrecount

パッケージのマニュアルの読み解きが難解である例を示します。①(Rパッケージの)recount。②ページ下部に移動。

2. 平成30年06月19日 (PC使用)

講義資料PDF(約2MB; 2018.06.12版)

(Rで)塩基配列解析

Blekhman et al., *Genome Res.*, 2010

TCC : Sun et al., *BMC Bioinformatics*, 2013

Tang et al., *BMC Bioinformatics*, 2015

Zhao et al., *Biol. Proc. Online*, 2018

ReCount(website) : Frazee et al., *BMC Bioinformatics*

平成28年度NGSハンズオン講習会

recount2(website) : Collado-Torres et al., *Nature*

recount(R package) : Collado-Torres et al., *Nature*

[rse.Rdata](#)(SRP001558)

[rse_gene.Rdata](#)(ERP000546)

http://bioconductor.org/packages/release/bioc/html/recount.html

Bioconductor - recount

Bioconductor
OPEN SOURCE SOFTWARE FOR BIOINFORMATICS

Home Install Help Developers About

Home » Bioconductor 3.7 » Software Packages » recount

recount

platforms all downloads top 20% posts 2 / 2 / 1 / 0 in Bioc 1.5 years
build ok

DOI: [10.18129/B9.bioc.recount](https://doi.org/10.18129/B9.bioc.recount)

Explore and download data from the recount project

Bioconductor version: Release (3.7)

Explore and download data from the recount project available at <https://jhubiostatistics.shinyapps.io/recount/>. Using the recount package you can download RangedSummarizedExperiment objects at the gene, exon or exon-exon junctions level, the raw counts, the phenotype metadata used, the urls to the sample coverage bigWig files or the mean coverage bigWig file for a particular study. The RangedSummarizedExperiment objects can be used by different packages for performing differential expression analysis. Using <http://bioconductor.org/packages/derfinder> you can perform annotation-agnostic differential expression analyses with the data from the recount project as described at <http://www.nature.com/nbt/journal/v35/n4/full/nbt.3838.html>.

Author: Leonardo Collado-Torres [aut, cre], Abhinav Nellore [ctb], Andrew E. Jaffe [ctb], Margaret A. Taub [ctb], Kai Kammers [ctb], Shannon E. Ellis [ctb], Kasper Daniel Hansen [ctb], Ben Langmead [ctb], Jeffrey T. Leek [aut, ths]

Maintainer: Leonardo Collado-Torres <lcollado@jhu.edu>

Citation (from within R, enter `citation("recount")`):

Collado-Torres L, Nellore A, Kammers K, Ellis SE, Taub MA, Hansen KD, Jaffe AE, Langmead B, Leek JT

Documentation »

Bioconductor

- Package [vignettes](#) and manuals.
- [Workflows](#) for learning and use.
- [Course and conference](#) material.
- [Videos](#).
- Community [resources](#) and [tutorials](#).

R / [CRAN](#) packages and [documentation](#)

Support »

Please read the [posting guide](#). Post questions about Bioconductor to one of the following locations:

- [Support site](#) - for questions about Bioconductor packages
- [Bioc-devel](#) mailing list - for package developers

Rパッケージrecount

← → <http://bioconductor.org/packages/release/bioc/html/recount.html> 検索...

Bioconductor - recount

Installation

To install this package, start R and enter:

```
## try http:// if https:// URLs are not supported
source("https://bioconductor.org/biocLite.R")
biocLite("recount")
```

Documentation

To view documentation for the version of this package installed in your system, start R and enter:

```
browseVignettes("recount")
```

HTML	Script	Basic DESeq2 results explor
HTML	Script	recount quick start guide
PDF		Reference Manual
Text		NEWS

Details

biocViews [Coverage](#), [DataImport](#), [DifferentialExpression](#), [GeneExpression](#), [RNASeq](#), [Sequencing](#), [Software](#)

Version 1.6.2

In Bioconductor since BioC 3.4 (R-3.3) (1.5 years)

License Artistic-2.0

Depends R (>= 3.3.0), [SummarizedExperiment](#)

Imports [BiocParallel](#), [derfinder](#), [downloader](#), [GEOquery](#), [GenomeInfoDb](#), [GenomicRanges](#), [IRanges](#), methods, [RCurl](#), [rentrez](#), [rtracklayer](#)(>= 1.35.3), [S4Vectors](#), stats, utils

LinkingTo

Suggests [AnnotationDbi](#), [BiocStyle](#)(>= 2.5.19), [DESeq2](#), [devtools](#)(>= 1.6), [EnsDb.Hsapiens.v79](#), [GenomicFeatures](#), [knitcitations](#), [knitr](#)(>= 1.6), [org.Hs.eq.db](#), [regionReport](#)(>= 1.9.4), [rmarkdown](#)(>= 0.9.5), [testthat](#)

SystemRequirements

Enhances

Rパッケージrecount

The screenshot shows a web browser window displaying the Bioconductor website. The address bar shows the URL: `http://bioconductor.org/packages/release/bioc/vignettes/recount/inst/doc/recount-qui`. The page title is "recount quick start guide".

On the left side, there is a navigation menu with the following items:

- 1 Basics
- 2 Quick start to using to recount
- 3 Introduction
- 4 Sample DE analysis
- 5 Sample derfinder analysis
- 6 Annotation used
- 7 Candidate gene fusions
- 8 Snaptron
- 9 Download all the data
- 10 Accessing recount via SciServer
- 11 Reproducibility
- 12 Bibliography

A red arrow points to the "2 Quick start to using to recount" link. The main content area shows the title "recount quick start guide" and the author "Leonardo Collado-Torres^{1,2*}". Below the author information, the date "14 May 2018" is displayed. The "Package" section indicates "recount 1.6.2".

The main heading is "1 Basics", and the sub-heading is "1.1 Install *recount*". The text explains that R is an open-source statistical environment and that *recount* is a package available via the Bioconductor repository. It states that R can be installed on any operating system from CRAN after which you can install *recount* by using the following commands in your R session:

```
## try http:// if https:// URLs are not supported
source("https://bioconductor.org/biocLite.R")
biocLite("recount")

## Check that you have a valid Bioconductor installation
```

Rパッケージrecount

①2 Quick start to using to recountに移動したことがわかります。②赤枠あたりから下のほうを順に読んでいってください。非常に難解であることが分かります。

recount quick start guide

2 Quick start to using to recount

Main updates:

- As of January 30, 2017 the annotation used for the exon and gene counts is Gencode v25.
- As of January 12, 2018 transcripts counts are available via `recount2` thanks to the work of Fu et al. Disjoint exon counts (version 2) were also released as described in detail in the [recount website](#) documentation tab.

recount2

Here is a very quick example of how to download a `RangedSummarizedExperiment` object with the gene counts for a 2 groups project (12 samples) with SRA study id [SRP009615](#) using the `recount` package (Collado-Torres, Nellore, Kammers, Ellis, et al., 2017). The `RangedSummarizedExperiment` object is defined in the [SummarizedExperiment](#) (Morgan, Obenchain, Hester, and Pagès, 2017) package and can be used for differential expression analysis with different packages. Here we show how to use `DESeq2` (Love, Huber, and Anders, 2014) to perform the differential expression analysis.

This quick analysis is explained in more detail later on in this document. Further information about the recount project can be found in the [main publication](#). Check the [recount website](#) for related publications.

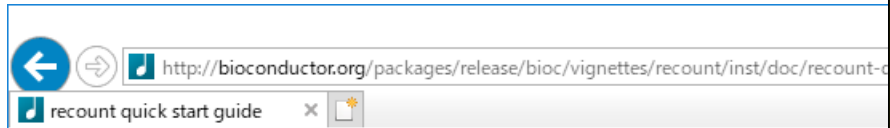
```
## Load library
library('recount')

## Find a project of interest
project_info <- abstract_search('GSE32465')

## Download the gene-level RangedSummarizedExperim
```

Rパッケージrecount

赤枠部分も見ただけで嫌になりますが、落ち着いてよく眺めると、①GSE32465というIDの、②rse_gene.Rdataを取得しているんだらうな、というのわかります。また、colData実行結果にどのような情報が含まれているかがわかっていれば、③colDataを駆使して有意義な列名情報を得ようとしているんだらうな、という程度はわかります。



- 1 Basics
- 2 Quick start to using to recount
- 3 Introduction
- 4 Sample DE analysis
- 5 Sample derfinder analysis
- 6 Annotation used
- 7 Candidate gene fusions
- 8 Snaptron
- 9 Download all the data
- 10 Accessing recount via SciServer
- 11 Reproducibility
- 12 Bibliography

```

## Load library
Library('recount')

## Find a project of interest
project_info <- abstract_search('GSE32465')

## Download the gene-level RangedSummarizedExperiment data
download_study(project_info$project)

## Load the data
load(file.path(project_info$project, 'rse_gene.Rdata'))

## Browse the project at SRA
browse_study(project_info$project)

## View GEO ids
colData(rse_gene)$geo_accession

## Extract the sample characteristics
geochar <- lapply(split(colData(rse_gene), seq_len(
  nrow(colData(rse_gene)))), geo_characteristics)

## Note: The information for this study is a little inconsistent, so we
## have to fix it.
geochar <- do.call(rbind, lapply(geochar, function(x) {
  if('cells' %in% colnames(x)) {
    colnames(x)[colnames(x) == 'cells'] <- 'cell.line'
  }
  return(x)
} else {
  return(x)
}
}))

```



DESeq2との連結

recount quick start guide

1 Basics

2 Quick start to using to recount

3 Introduction

4 Sample DE analysis

5 Sample derfinder analysis

6 Annotation used

7 Candidate gene fusions

8 Snaptron

9 Download all the data

10 Accessing recount via SciServer

11 Reproducibility

12 Bibliography

```

## We can now define some sample information to use
e
sample_info <- data.frame(
  run = colData(rse_gene)$run,
  group = ifelse(grep1('uninduced', colData(rse_gene)$title), 'uninduced', 'induced'),
  gene_target = sapply(colData(rse_gene)$title,
function(x) { strsplit(strsplit(x,
    'targeting ')[[1]][2], ' gene')[[1]]
  [1] }},
  cell.line = geochar$cell.line
)

## Scale counts by taking into account the total c
verage per sample
rse <- scale_counts(rse_gene)

## Add sample information for DE analysis
colData(rse)$group <- sample_info$group
colData(rse)$gene_target <- sample_info$gene_target

## Perform differential expression analysis with DESeq2
library('DESeq2')

## Specify design and switch to DESeq2 format
dds <- DESeqDataSet(rse, ~ gene_target + group)

## Perform DE analysis
dds <- DESeq(dds, test = 'LRT', reduced = ~ gene_target, fitType = 'local')
res <- results(dds)

## Explore results
plotMA(res, main="DESeq2 results for SRP009615")

## Make a report with the results
library('regionReport')

```


DESeq2との連結

まずは①rse_geneを得るところまでコピー実行して、rse_geneの中身を様々な視点で眺め、これより上の行で一体何をやっているかを解読するような戦略もあり。

http://bioconductor.org/packages/release/bioc/vignettes/recount/inst/doc/recount-quick-start-guide

recount quick start guide

- 1 Basics
- 2 Quick start to using to recount
- 3 Introduction
- 4 Sample DE analysis
- 5 Sample derfinder analysis
- 6 Annotation used
- 7 Candidate gene fusions
- 8 Snaptron
- 9 Download all the data
- 10 Accessing recount via SciServer
- 11 Reproducibility
- 12 Bibliography

```

## We can now define some sample information to use
e
sample_info <- data.frame(
  run = colData(rse_gene)$run,
  group = ifelse(grep1('uninduced', colData(rse_gene)$title), 'uninduced', 'induced'),
  gene_target = sapply(colData(rse_gene)$title,
function(x) { strsplit(strsplit(x,
  'targeting ')[[1]][2], ' gene')[[1]]
[1] }},
  cell.line = geochar$cell.line
)

## Scale counts by ① to account the total c
overage per sample
rse <- scale_counts(rse_gene)

## Add ② information for DE analysis
colData(rse)$group <- sample_info$group
colData(rse)$gene_target <- sample_info$gene_target

## Perform differential expression analysis with D
ESeq2
library('DESeq2')

## Specify design and switch to DESeq2 format
dds <- DESeqDataSet(rse, ~ gene_target + group)

## Perform DE analysis
dds <- DESeq(dds, test = 'LRT', reduced = ~ gene_t
arget, fitType = 'local')
res <- results(dds)

## Explore results
plotMA(res, main="DESeq2 results for SRP009615")

## Make a report with the results
library('regionReport')

```

scale_counts

まずは①rse_geneを得るところまでコピー実行して、rse_geneの中身を様々な視点で眺め、これより上の行で一体何をやっているかを解読するような戦略もあり。また、DESeq2への受け渡し前に②scale_countsを実行している点も見逃してはいけない。が、私も挙動を完全に掌握できているわけではないのでとりあえず保留。

http://bioconductor.org/packages/release/bioc/vignettes/recount/inst/doc/recount-quick-start-guide

- 1 Basics
- 2 Quick start to using to recount
- 3 Introduction
- 4 Sample DE analysis
- 5 Sample derfinder analysis
- 6 Annotation used
- 7 Candidate gene fusions
- 8 Snaptron
- 9 Download all the data
- 10 Accessing recount via SciServer
- 11 Reproducibility
- 12 Bibliography

```
## We can now define some sample information
sample_info <- data.frame(
  run = colData(rse_gene)$run,
  group = ifelse(grep1('uninduced', colData(rse_gene)$title), 'uninduced', 'induced'),
  gene_target = sapply(colData(rse_gene)$title,
function(x) { strsplit(strsplit(x,
    'targeting ')[[1]][2], ' gene')[[1]]
  [1] }},
  cell.line = geochar$cell.line
)

## Scale counts by ① to account the total coverage per sample
rse <- scale_counts(rse_gene)

## Add ② information for DE analysis
colData(rse)$group <- sample_info$group
colData(rse)$gene_target <- sample_info$gene_target

## Perform differential expression analysis with DESeq2
library('DESeq2')

## Specify design and switch to DESeq2 format
dds <- DESeqDataSet(rse, ~ gene_target + group)

## Perform DE analysis
dds <- DESeq(dds, test = 'LRT', reduced = ~ gene_target, fitType = 'local')
res <- results(dds)

## Explore results
plotMA(res, main="DESeq2 results for SRP009615")

## Make a report with the results
library('regionReport')
```

Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?! カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

サンプル間クラスタリング

②例題1でもなんでもいいので、テンプレートとして利用し、[hoge5.txt](#)と[hoge6.txt](#)のサンプル間クラスタリングをやってみましょう。

(Rで)塩基配列解析

(last modified 2018/06/11, since 2010)

このウェブページのR関連部分は、[インストール](#)についての推奨手順 ([Windows](#) [Linux](#) [Mac](#))
ツール済みであるという前提で記述しています。初心者の方は[基本的な利用](#)
系的にまと

- ・解析 | [発現量推定\(トランスクリプトーム配列を利用\)](#) (last modified 2014/07/09)
- ・解析 | [クラスタリング](#) | [サンプル間](#) | [hclust](#) (last modified 2014/02/05)
- ・解析 | [クラスタリング](#) | [サンプル間](#) | [hclust](#) (last modified 2015/02/26) **NEW**
- ・解析 | [クラスタリング](#) | [サンプル間](#) | [TCC\(Sun_2013\)](#) **①** (last modified 2015/03/02) **NEW**
- ・解析 | [クラスタリング](#) | [遺伝子間](#) | [MBCluster.Seq \(Seq-IT4\)](#) (last modified 2014/02/05)
- ・解析 | [シミュレーションカウントデータ](#) | [シミュレーション](#) | [シミュレーション](#) (last modified 2015/01/25)

What's new
・以下の
の.Rdat
SRP001

解析 | クラスタリング | サンプル間 | [TCC\(Sun_2013\)](#) **NEW**

[TCC](#)パッケージを用いてサンプル間クラスタリングを行うやり方を示します。clusterSample関数を利用した頑健なクラスタリング結果を返します。
「ファイル」-「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し以下をコピー。

② 1. 59,857 genes×6 samplesのリアルデータ([srp017142_count_bowtie.txt](#))の場合:

[Neyret-Kahn et al., Genome Res., 2013](#)の2群間比較用(3 proliferative samples vs. 3 Ras samples)ヒトRNA-seqカウントデータです。[パイプライン](#)
[ゲノム](#) | [発現変動](#) | [2群間](#) | [対応なし](#) | [複製あり](#) | [SRP017142\(Neyret-Kahn_2013\)](#)から得られます。

```

in_f <- "srp017142_count_bowtie.txt" #入力ファイル名を指定してin_fに格納
out_f <- "hoge1.png" #出力ファイル名を指定してout_fに格納
param_fig <- c(500, 400) #ファイル出力時の横幅と縦幅を指定(単位はピクセル)

#必要なパッケージをロード
library(TCC) #パッケージの読み込み

#入力ファイルの読み込み
data <- read.table(in_f, header=TRUE, row.names=1, sep="\t", quote="") #in_fで指定したファイルの読み込み
dim(data) #オブジェクトdataの行数と列数を表示

#本番
out <- clusterSample(data, dist.method="spearman", #クラスタリング実行結果をoutに格納
                     hclust.method="average", unique.pattern=TRUE) #クラスタリング実行結果をoutに格納

```

サンプル間クラスタリング

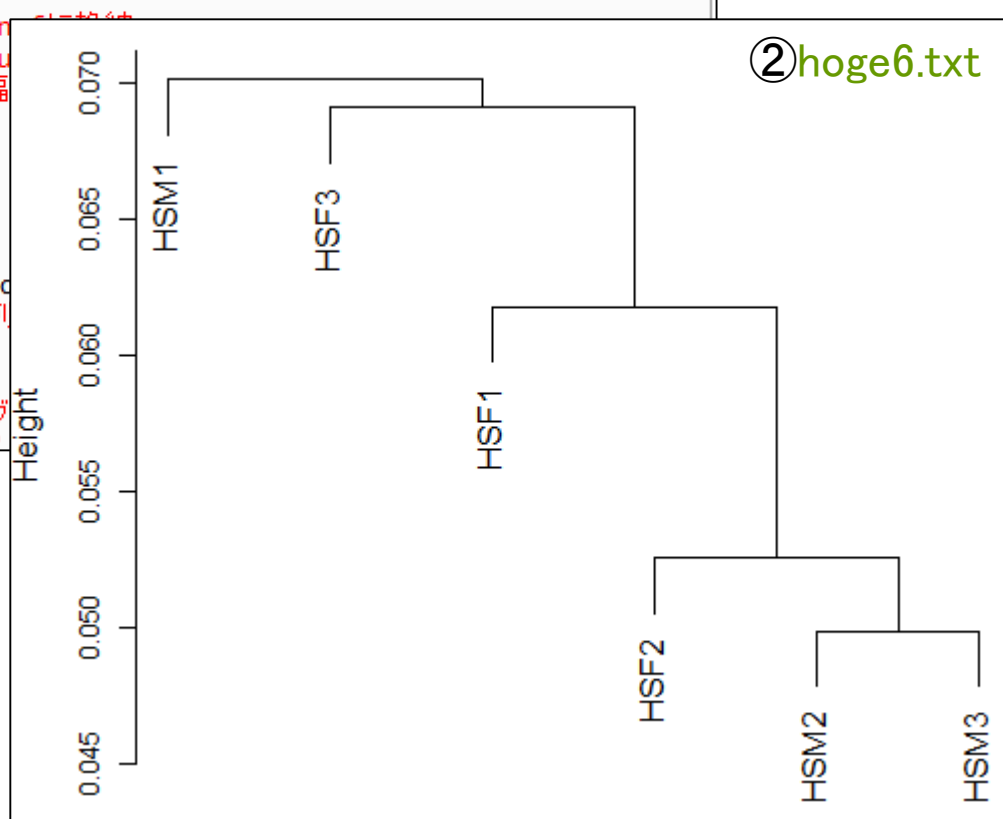
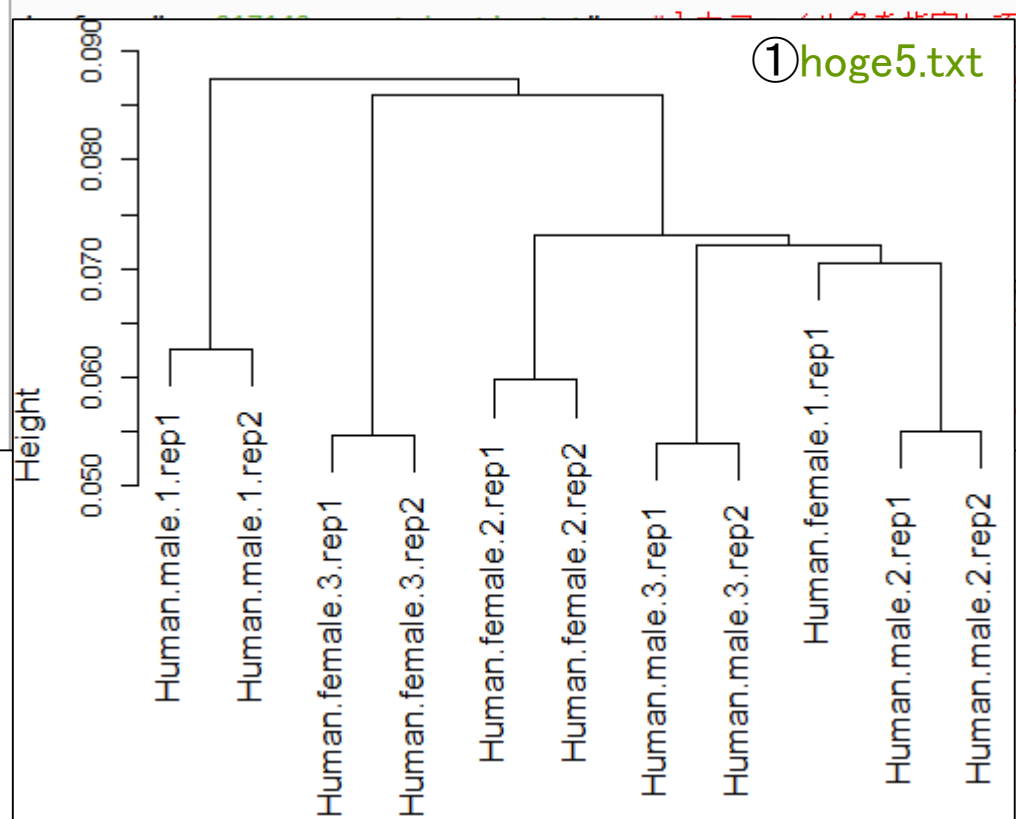
例題7をテンプレートとして、①hoge5.txt、②hoge6.txtを入力として、pngファイルの大きさを500×400にして実行した結果。

解析 | クラスタリング | サンプル間 | TCC(Sun_2013) NEW

TCCパッケージを用いてサンプル間クラスタリングを行うやり方を示します。clusterSample関数を利用した頑健なクラスタリング結果を返します。「ファイル」-「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し以下をコピー。

1. 59,857 genes×6 samplesのリアルデータ([srp017142 count bowtie.txt](#))の場合:

[Neyret-Kahn et al. Genome Res. 2013](#)の2群間比較用(3 proliferative samples vs. 3 Ras samples)ヒトRNA-seqカウントデータです。[パイプライン | ゲノム | 発現変動 | 2群間 | 対応なし | 複製あり | SRP017142\(Neyret-Kahn_2013\)](#)から得られます。



サンプル間クラスタリング

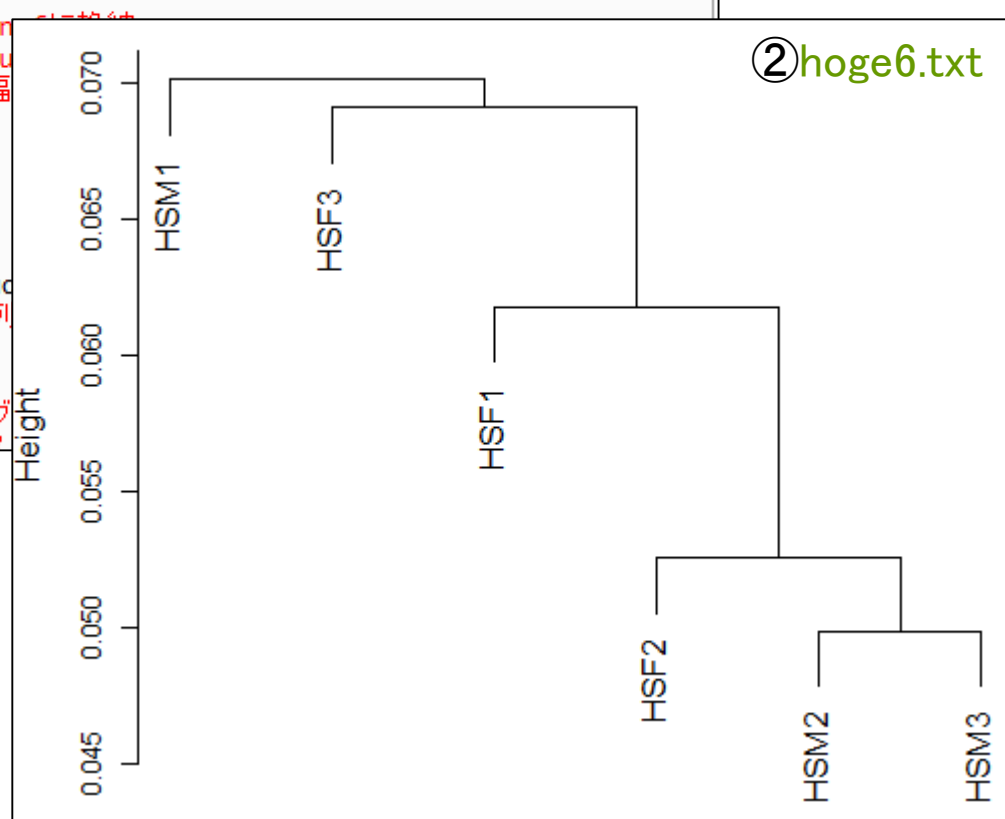
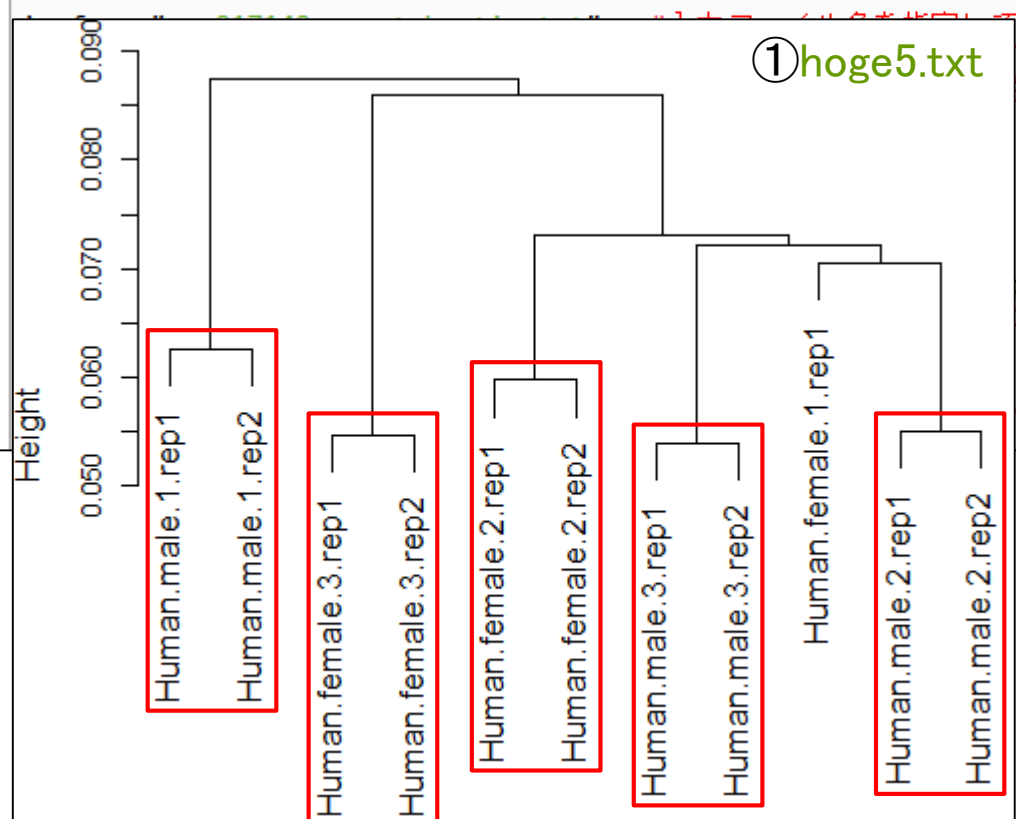
解析 | クラスタリング | サンプル間 | TCC(Sun_2013) NEW

TCCパッケージを用いてサンプル間クラスタリングを行うやり方を示します。clusterSample関数を利用。「ファイル」-「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し以下をコピー。

1. 59,857 genes×6 samplesのリアルデータ([srp017142 count bowtie.txt](#))の場合:

[Neyret-Kahn et al. Genome Res. 2013](#)の2群間比較用(3 proliferative samples vs. 3 Ras samples)ヒトRNA-seqカウントデータです。[パイプライン | ゲノム | 発現変動 | 2群間 | 対応なし | 複製あり | SRP017142\(Neyret-Kahn_2013\)](#)から得られます。

赤枠より、同一個体の反復データ (technical replicates) で末端のクラスターを形成していることが分かる。これは technical replicates 同士の類似度が 非常に高い ことを意味しており、妥当。



サンプル間クラスタリング

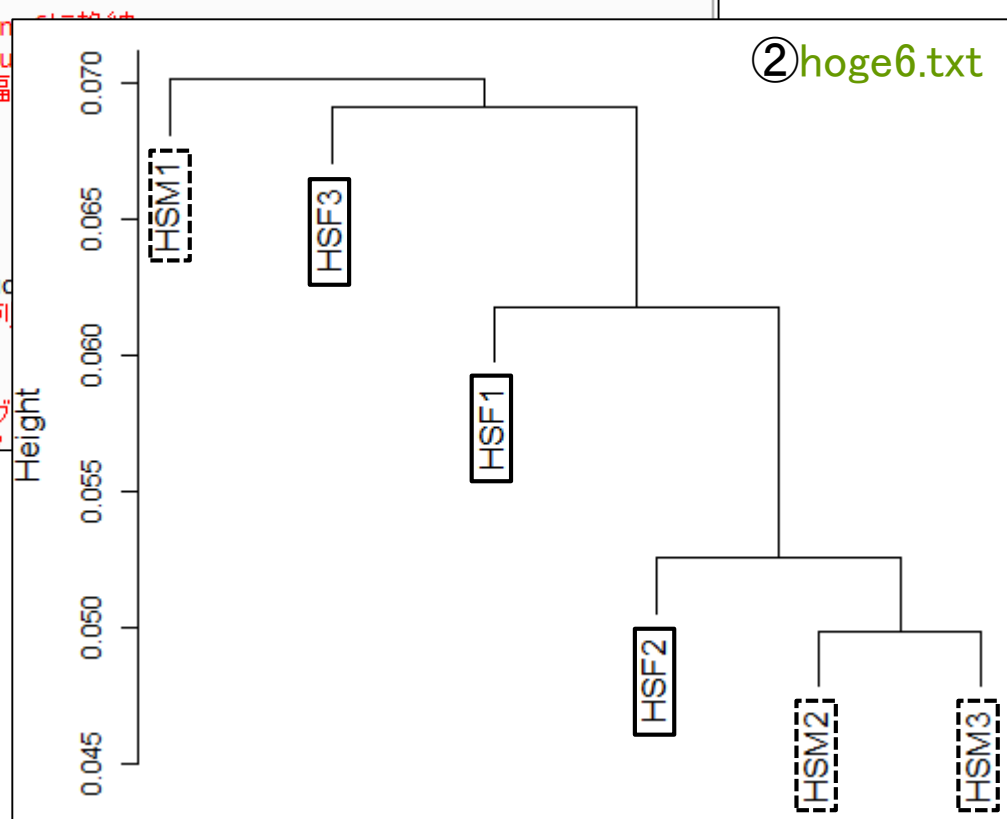
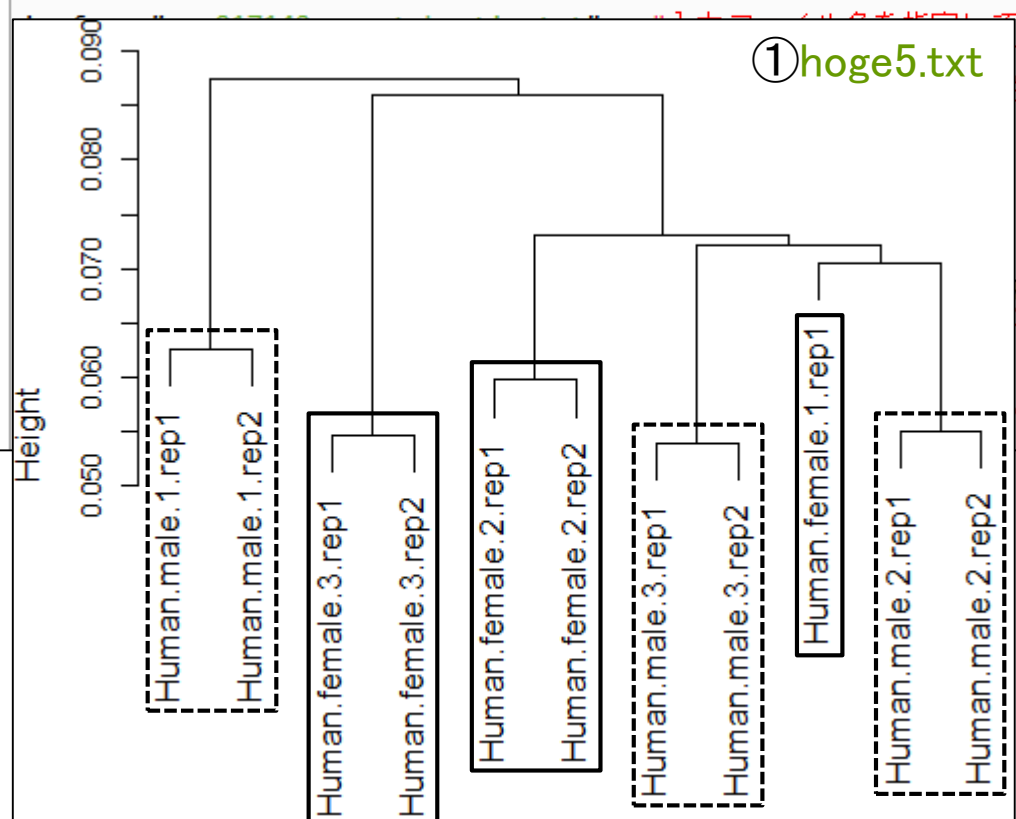
ヒトのメス(HSF)とヒトのオス(HSM)では、肝臓(Liver)の発現パターンに差がないことがわかる。つまり雌雄差はなさそう。

解析 | クラスタリング | サンプル間 | TCC(Sun_2013) NEW

TCCパッケージを用いてサンプル間クラスタリングを行うやり方を示します。clusterSample関数を利用した頑健なクラスタリング結果を返します。「ファイル」-「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し以下をコピー。

1. 59,857 genes×6 samplesのリアルデータ(srp017142 count bowtie.txt)の場合:

Neyret-Kahn et al. Genome Res. 2013の2群間比較用(3 proliferative samples vs. 3 Ras samples)ヒトRNA-seqカウントデータです。パイプライン | ゲノム | 発現変動 | 2群間 | 対応なし | 複製あり | SRP017142(Neyret-Kahn_2013)から得られます。



おさらい

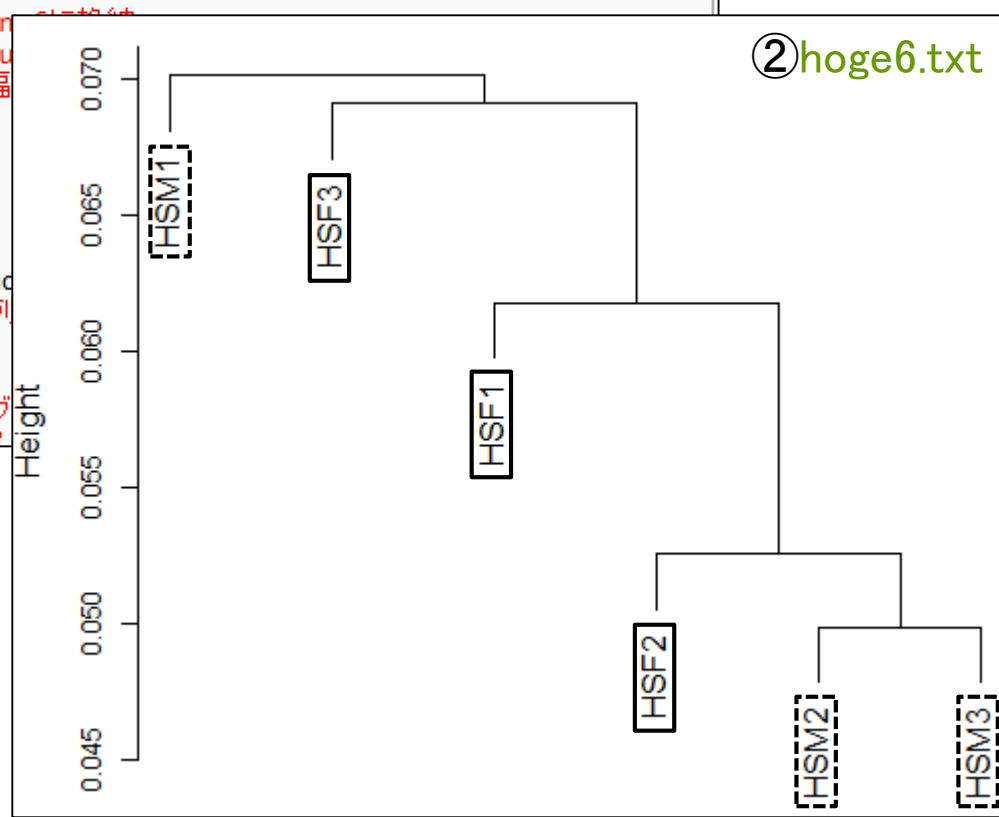
解析 | クラスタリング | サンプル間 | TCC(Sun

TCCパッケージを用いてサンプル間クラスタリングを行うやり方を示した「ファイル」-「ディレクトリの変更」で解析したいファイルを置いてあるデータ

1. 59,857 genes×6 samplesのリアルデータ([srp017142 count bowtie.t](#)

[Neyret-Kahn et al., Genome Res., 2013](#)の2群間比較用(3 proliferative samples vs. 3 Ras samples)のRNA-seqカウントデータです。[バイオインフォマティクス | 発現変動 | 2群間 | 対応なし | 複製あり | SRP017142\(Neyret-Kahn 2013\)](#)から得られます。

このクラスタリング結果の元データ(58,037 genesからなるカウントデータ)は、recountのグループが①この原著論文の、②公共DBのIDであるSRP001558(の生リードデータ)を独自のパイプラインを実行した結果をRangedSummarizedExperiment(RSE)形式のオブジェクトとしてrse_gene.Rdataとして提供しているもの。サンプルのメタデータ情報もSRP001558の記載内容をベースとしている。



SRP001558(Blekhman et al., *Genome Res.*, **20**: 180-9, 2010)

おさらい

スライド20あたりまでで取り扱っていた20,689 genesからなるカウントデータは、①この原著論文の②Supplementary Table 1で提供されているものです。遺伝子数も異なるうえ、当時とはアノテーション(遺伝子の座標)情報も異なると思われるので、クラスタリング結果の単純な比較はできない。

- マップ後 | カウント 情報取得 | paired-end | トランスクリプトーム | [Qualimap](#)
- マップ後 | カウント 情報取得 | トランスクリプトーム | [BEDファイルから](#)
- [カウント 情報取得 | について](#) (last modified 2018/06/10) **NEW**
- カウント 情報取得 | リアルデータ | SRP001558 | [recount](#)(Collado-Torres, 2017)(last modified 2017/07/04)
- カウント 情報取得 | リアルデータ | ERP000546 | [recount](#)(Soneson, 2014)(last modified 2014/06/22)
- カウント 情報取得 | シミュレーションデータ (last modified 2018/06/10)
- [配列長とカウント数の関係](#) (last modified 2018/06/09)
- [正規化 | について](#) (last modified 2014/06/22)
- 正規化 | 基礎 | [RPK or CPK \(配列長補正\)](#) (last modified 2014/06/22)
- 正規化 | 基礎 | [RPM or CPM \(総リード数補正\)](#) (last modified 2014/06/22)
- 正規化 | 基礎 | [RPKM](#) (last modified 2015/07/04)
- 正規化 | サンプル内 | [EDASeq\(Risso 2011\)](#) (last modified 2011/07/04)
- 正規化 | サンプル内 | [RNASeqBias\(Zheng 2011\)](#) (last modified 2011/07/04)

カウント情報取得 | について **NEW**

ここではSAM/BAMなどのマッピング結果ファイルからのカウント 情報取得ではなく、最初からカウント 情報になっているもののありかや、それらを提供しているデータベースから取得するやり方、そしてシミュレーションカウントデータを生成するやり方を示します。

R用(リアルデータ):

- [recount](#): [Collado-Torres et al., Nat Biotechnol., 2017](#)

R用(シミュレーションデータ):

- [TCC](#): [Sun et al., BMC Bioinformatics, 2013](#)
- [compcoder](#): [Soneson C., Bioinformatics, 2014](#)
- [SimSeq](#): [Benidt and Nettleton, Bioinformatics, 2015](#)
- [Polyester](#)(single-cell RNA-seq用): [Frazee et al., Bioinformatics, 2015](#)
- [Splatter](#)(single-cell RNA-seq用): [Zappia et al., Genome Biol., 2017](#)
- [powsimR](#)(single-cell RNA-seq用): [Vieth et al., Bioinformatics, 2017](#)

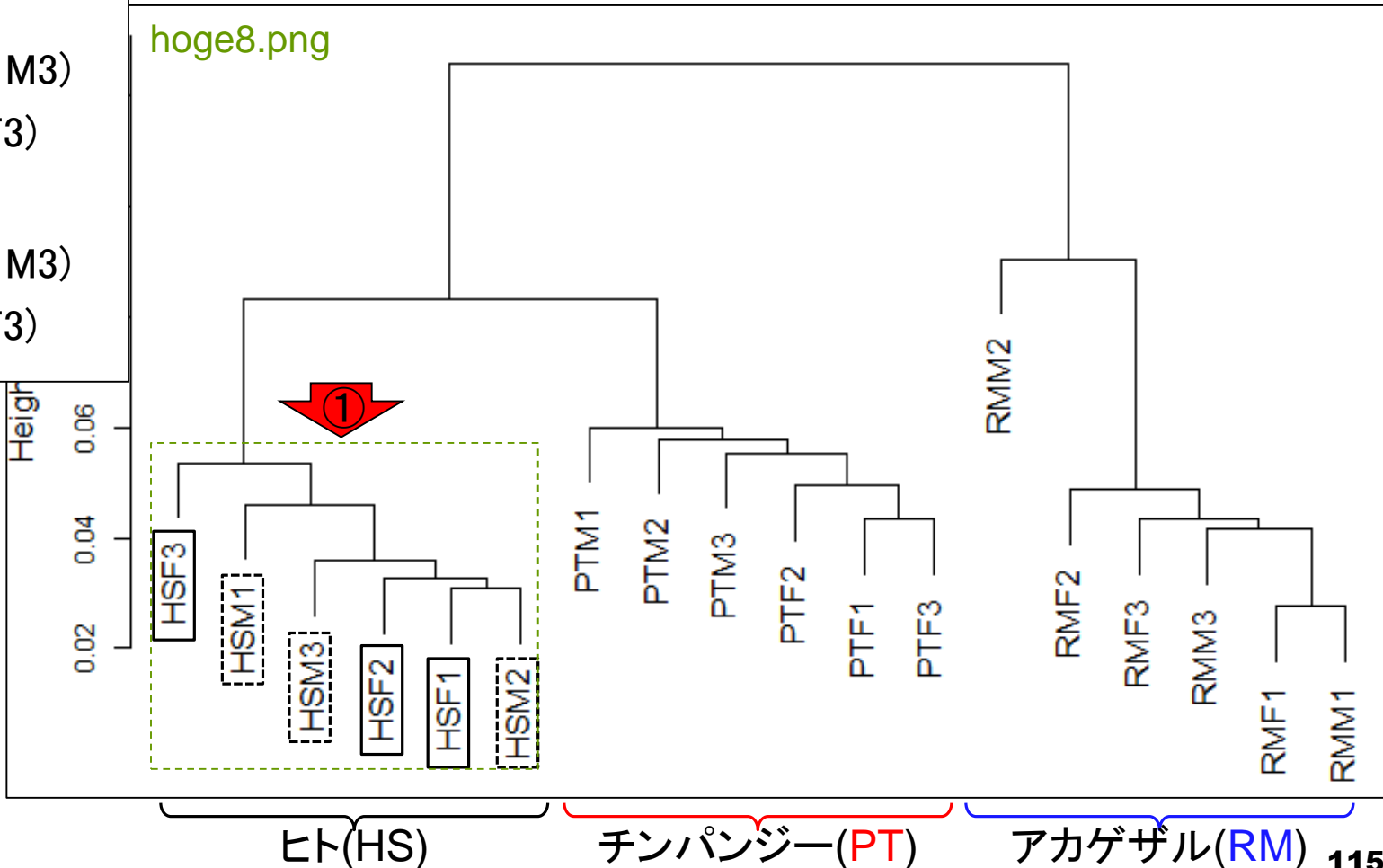
R以外:

- Supplementary Table1([suppTable1.xls](#)): [Blekhman et al., Genome Res., 2010](#)
この公共DB内のIDは[GSE17274](#)や[SRP001558](#)です。以下に整形したデータもあります:
 - サンプルデータ41で作成した20,689 genes×36 samplesのカウントデータ([sample blekhman 36.txt](#))
 - サンプルデータ42で作成した20,689 genes×18 samplesのカウントデータ([sample blekhman 18.txt](#))
- [ReCount](#)(website): [Frazee et al., BMC Bioinformatics, 2011](#)
- [recount2](#)(website): [Collado-Torres et al., Nat Biotechnol., 2017](#)

おさらい

サンプルデータの例題42で作成した20,689 genes × 18 samplesのカウントデータのクラスタリング結果(スライド15-20)。メスとオスのサンプルが入り混じっており、雌雄差はなさそうという結論は、recountの58,037 genesのときと変わらない。

- ヒト(HS)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- チンパンジー(PT)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)
- アカゲザル(RM)
 - オス3匹(M1, M2, M3)
 - メス3匹(F1, F2, F3)



Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?!カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

Rパッケージrecount

- ①(Rパッケージの)recount。
- ②ページ下部に移動。

2. 平成30年06月19日 (PC使用)
講義資料PDF(約2MB; 2018.06.12版)
(Rで)塩基配列解析
Blekhman et al., Genome Res., 2010
TCC : Sun et al., BMC Bioinformatics, 2013
Tang et al., BMC Bioinformatics, 2015
Zhao et al., Biol. Proc. Online, 2018
ReCount(website) : Frazee et al., BMC Bioinformatics, 2018
平成28年度NGSハンズオン講習会
recount2(website) : Collado-Torres et al., Nature Methods, 2018
recount(R package) : Collado-Torres et al., Nature Methods, 2018
rse.Rdata(SRP001558)
rse_gene.Rdata(ERP000546)

http://bioconductor.org/packages/release/bioc/html/recount.html

Bioconductor - recount

Bioconductor
OPEN SOURCE SOFTWARE FOR BIOINFORMATICS

Home Install Help Developers About

Home » Bioconductor 3.7 » Software Packages » recount

recount

platforms all downloads top 20% posts 2 / 2 / 1 / 0 in Bioc 1.5 years
build ok

DOI: [10.18129/B9.bioc.recount](https://doi.org/10.18129/B9.bioc.recount)

Explore and download data from the recount project

Bioconductor version: Release (3.7)

Explore and download data from the recount project available at <https://jhubiostatistics.shinyapps.io/recount/>. Using the recount package you can download RangedSummarizedExperiment objects at the gene, exon or exon-exon junctions level, the raw counts, the phenotype metadata used, the urls to the sample coverage bigWig files or the mean coverage bigWig file for a particular study. The RangedSummarizedExperiment objects can be used by different packages for performing differential expression analysis. Using <http://bioconductor.org/packages/derfinder> you can perform annotation-agnostic differential expression analyses with the data from the recount project as described at <http://www.nature.com/nbt/journal/v35/n4/full/nbt.3838.html>.

Author: Leonardo Collado-Torres [aut, cre], Abhinav Nellore [ctb], Andrew E. Jaffe [ctb], Margaret A. Taub [ctb], Kai Kammers [ctb], Shannon E. Ellis [ctb], Kasper Daniel Hansen [ctb], Ben Langmead [ctb], Jeffrey T. Leek [aut, ths]

Maintainer: Leonardo Collado-Torres <lcollado@jhu.edu>

Citation (from within R, enter `citation("recount")`):

Collado-Torres L, Nellore A, Kammers K, Ellis SE, Taub MA, Hansen KD, Jaffe AE, Langmead B, Leek JT

Reference Manual

http://bioconductor.org/packages/release/bioc/html/recount.html

Installation

To install this package, start R and enter:

```
## try http:// if https:// URLs are not supported
source("https://bioconductor.org/biocLite.R")
biocLite("recount")
```

Documentation

To view documentation for the version of this package installed in your system, start R and enter:

```
browseVignettes("recount")
```

[HTML](#) [R Script](#) Basic DESeq2 results exploration
[HTML](#) [R Script](#) recount quick start guide
[PDF](#) **②** Reference Manual **①**
[Text](#) NEWS

Details

biocViews [Coverage](#), [DataImport](#), [DifferentialExpression](#), [GeneExpression](#), [RNASeq](#), [Sequencing](#), [Software](#)

Version 1.6.2

In Bioconductor since BioC 3.4 (R-3.3) (1.5 years)

License Artistic-2.0

Depends R (>= 3.3.0), [SummarizedExperiment](#)

Imports [BiocParallel](#), [derfinder](#), [downloader](#), [GEOquery](#), [GenomeInfoDb](#), [GenomicRanges](#), [IRanges](#), methods, [RCurl](#), [rentrez](#), [rtracklayer](#)(>= 1.35.3), [S4Vectors](#), stats, utils

LinkingTo

Suggests [AnnotationDbi](#), [BiocStyle](#)(>= 2.5.19), [DESeq2](#), [devtools](#) (>= 1.6), [EnsDb.Hsapiens.v79](#), [GenomicFeatures](#), [knitcitations](#), [knitr](#) (>= 1.6), [org.Hs.eq.db](#), [regionReport](#)(>= 1.9.4), [rmarkdown](#) (>= 0.9.5), [testthat](#)

SystemRequirements

Enhances

Reference ManualのPDF

①Reference ManualのPDFが開きます。

The screenshot shows a web browser window with the URL `http://bioconductor.org/packages/release/bioc/manuals/recount/man/recount.pdf`. The page content is the Reference Manual for the 'recount' package, including sections for Title, Date, Depends, Imports, Suggests, Version, License, and Keywords. A PDF viewer overlay is active on the right side of the browser, showing options to 'PDFを書き出し' (Export PDF), 'Adobe Export PDF', and 'PDFを作成' (Create PDF). The viewer also displays the filename 'recount.pdf' and a dropdown menu for 'Microsoft Word (*.docx)'. A blue '変換' (Convert) button is visible. The browser interface includes a search bar, navigation icons, and a sidebar with a list of bookmarked pages.

Reference ManualのPDF

①の赤枠がPDFの中身。②のところを押して、赤枠の横幅を広げる。

The screenshot shows a web browser window displaying a PDF document from <http://bioconductor.org/packages/release/bioc/manuals/recount/man/recount.pdf>. The browser's address bar shows the URL and a search icon. The page content is titled "Package 'recount'" and includes metadata such as "June 13, 2018" and "Version 1.4.2". The document text is highlighted with a red box labeled "1". A red arrow labeled "2" points to a zoom-in icon in the browser's navigation bar.

bioconductor.org

1 / 25 34.2%

PDFを書き出し

Adobe Export PDF

オンラインで PDF ファイルを Word または Excel に変換します

PDF ファイルを選択

recount.pdf

次に変換:

Microsoft Word (*.docx)

文書の言語:

日本語 変更

変換

PDF を作成

PDF を編集

Document Cloud でファイルを保存および共有
さらに詳しく

Reference Manualの印刷

多少文字が大きくなりましたが、まだ見えづらい場合は、①などを押してうまく調整してください。

bioconductor.org

http://bioconductor.org/packages/release/bioc/manuals/recount/man/recount.pdf

1 / 25 49.3%

Package 'recount'

June 13, 2018

Title Explore and download data from the recount project

Version 1.6.2

Date 2018-05-14

Depends R (>= 3.3.0), SummarizedExperiment

Imports BiocParallel, dplyr, downloader, GEOquery, GenomicInfoDB, GenomicRanges, IRanges, methods, RCurl, rstatix, tracklayer (>= 1.35.3), S4Vectors, stats, utils

Suggests AnnotationDbI, BiocStyle (>= 2.5.19), DESeq2, devtools (>= 1.6), Ensembl.Hsapiens.v79, GenomicFeatures, knitr (>= 1.6), org.Hs.eg.db, regionReport (>= 1.9.4), rmarkdown (>= 0.9.5), testthat

VignetteBuilder knitr

Description Explore and download data from the recount project available at <https://hubstatistics.shinyapps.io/recount/>. Using the recount package you can download RangedSummarizedExperiment objects at the gene, exon or exon-exon junctions level, the raw counts, the phenotype metadata used, the urls to the sample coverage bigWig files or the mean coverage bigWig file for a particular study. The RangedSummarizedExperiment objects can be used by different packages for performing differential expression analysis. Using <http://bioconductor.org/packages/dplyr/> you can perform annotation-agnostic differential expression analyses with the data from the recount project as described at <http://www.nature.com/nbt/journal/v35/n04/full/nbt3838.html>.

License Artistic-2.0

Encoding UTF-8

LazyData true

URL <https://github.com/loekgroup/recount>

BugReports <https://support.bioconductor.org/t/recount/>

biocViews Coverage, DifferentialExpression, GeneExpression, RNASeq, Sequencing, Software, DataImport

RoxigenNote 6.0.1

Author Leonardo Collado-Torres [aut, cre],
Abhinav Nellore [cib],
Andrew E. Jaffe [cib],
Margaret A. Tanh [cib],
Kai Kammerer [cib],

Reference ManualのP

ここでは①100%まで拡大しました。他にも、②のあたりを左にずらすことで、③しおりが占める領域を狭めることができます。

The screenshot shows a web browser window displaying a PDF document from bioconductor.org. The address bar shows the URL: <http://bioconductor.org/packages/release/bioc/manuals/recount/man/recount.pdf>. The browser's toolbar includes navigation icons, a search bar, and a zoom level of 100%. A red arrow labeled '1' points to the zoom level dropdown. On the left side, there is a sidebar with a 'しおり' (Bookmarks) section containing a list of document sections. A red arrow labeled '3' points to the bookmark icon in the sidebar. The main content area of the PDF is titled 'Package 'recount'' and includes the following information:

- Package 'recount'**
- June 13, 2018
- Title** Explore and download data from the recount project
- Version** 1.6.2
- Date** 2018-05-14
- Depends** R (>= 3.3.0), SummarizedExperiment
- Imports** BiocParallel, derfinder, downloader, GEOquery, GenomeInfoDb, GenomicRanges, IRanges, methods, RCurl, rentrez, rtracklayer (>= 1.35.3), S4Vectors, stats, utils
- Suggests** AnnotationDbi, BiocStyle (>= 2.5.19), DESeq2, devtools (>= 1.6), EnsDb.Hsapiens.v79, GenomicFeatures, knitcitations, knitr (>= 1.6), org.Hs.eg.db, regionReport (>= 1.9.4), rmarkdown (>= 0.9.5), testthat

A red arrow labeled '2' points to the left edge of the PDF content area, indicating the area where the sidebar can be moved to reduce its width.

Reference ManualのP

ここでは①100%まで拡大しました。他にも、②のあたりを左にずらすことで、③しおりが占める領域を狭めることができます。こんな感じ。

The screenshot shows a web browser window displaying a PDF document from bioconductor.org. The address bar shows the URL: <http://bioconductor.org/packages/release/bioc/manuals/recount/man/recount.pdf>. The browser's toolbar includes a search bar with the text "検索...", a zoom level of 100%, and a "サインイン" button. The PDF content is titled "Package 'recount'" and includes the following information:

- Title** Explore and download data from the recount project
- Version** 1.6.2
- Date** 2018-05-14
- Depends** R (>= 3.3.0), SummarizedExperiment
- Imports** BiocParallel, derfinder, downloader, GEOquery, GenomeInfoDb, GenomicRanges, IRanges, methods, RCurl, rentrez, rtracklayer (>= 1.35.3), S4Vectors, stats, utils
- Suggests** AnnotationDbi, BiocStyle (>= 2.5.19), DESeq2, devtools (>= 1.6), EnsDb.Hsapiens.v79, GenomicFeatures, knitcitations, knitr (>= 1.6), org.Hs.eg.db, regionReport (>= 1.9.4), rmarkdown (>= 0.9.5), testthat

Annotations on the image indicate:

- ①: A red arrow pointing to the zoom level dropdown menu (100%).
- ②: A red arrow pointing to the left edge of the PDF content area.
- ③: A red arrow pointing to the bookmark sidebar on the left.

Reference ManualのP

①ここにgetRPKMという興味ある関数が見えているが、とりあえずは②でページ下部に移動し、「recount quick start guide」のHTMLファイル中でDESeq2を用いた発現変動解析を行う際にscale_countsを実行しなければならないと書かれていたので、それを見してみる。

http://bioconductor.org/packages/release/bioc/manuals/recount/man/recount.pdf

bioconductor.org

1 / 25 100%

しおり

- recount-package
- abstract_search
- add_metadata
- add_predictions
- all_metadata
- browse_study
- coverage_matrix
- download_study
- expressed_regions
- find_geo
- geo_characteristics
- geo_info
- getRPKM
- read_counts
- recount_abstract
- recount_exons
- recount_genes

Package 'recount'

June 13, 2018

Title Explore and download data from the recount project

Version 1.6.2

Date 2018-05-14

Depends R (>= 3.3.0), SummarizedExperiment

Imports BiocParallel, derfinder, downloader, GEOquery, GenomeInfoDb, GenomicRanges, IRanges, methods, RCurl, rentrez, rtracklayer (>= 1.35.3), S4Vectors, stats, utils

Suggests AnnotationDbi, BiocStyle (>= 2.5.19), DESeq2, devtools (>= 1.6), EnsDb.Hsapiens.v79, GenomicFeatures, knitcitations, knitr (>= 1.6), org.Hs.eg.db, regionReport (>= 1.9.4), rmarkdown (>= 0.9.5), testthat

210 x 297 mm

scale_counts

http://bioconductor.org/packages/release/bioc/manuals/recount/man/recount.pdf

bioconductor.org

1 / 25 100%

サインイン

Package 'recount'

June 13, 2018

Title Explore and download data from the recount project

Version 1.6.2

Date 2018-05-14

Depends R (>= 3.3.0), SummarizedExperiment

Imports BiocParallel, derfinder, downloader, GEOquery, GenomeInfoDb, GenomicRanges, IRanges, methods, RCurl, rentrez, rtracklayer (>= 1.35.3), S4Vectors, stats, utils

Suggests AnnotationDbi, BiocStyle (>= 2.5.19), DESeq2, devtools (>= 1.6), EnsDb.Hsapiens.v79, GenomicFeatures, knitcitations, knitr (>= 1.6), org.Hs.eg.db, regionReport (>= 1.9.4), rmarkdown (>= 0.9.5), testthat

scale_counts

scale_counts

こんな感じになって、①scale_countsの説明部分に②ページが自動的に飛んでいるのが分かります。③がタイトルで、大まかな関数の説明部分。これを見た段階で「RPMのような総カウント数を揃えるものなのだろう」と予想できる。

The screenshot shows a web browser window displaying the Bioconductor manual page for the `scale_counts` function. The browser's address bar shows the URL `http://bioconductor.org/packages/release/bioc/manuals/recount/man/recount.pdf`. The page title is `scale_counts` and the subtitle is `Scale the raw counts provided by the recount project`. The page content includes a **Description** section, a **Usage** section with the function signature `scale_counts(rse, by = "auc", targetSize = 4e+07, L = 100, factor_only = FALSE, round = TRUE)`, and an **Arguments** section listing parameters like `rse`, `by`, `targetSize`, `L`, `factor_only`, and `round`. A sidebar on the left contains a list of links, with `scale_counts` highlighted. Red arrows are overlaid on the image: arrow 1 points to the search bar, arrow 2 points to the page number '22 / 25', and arrow 3 points to the title.

scale_counts

①Descriptionが、もう少し詳細な説明部分。これまで特に言及してきませんでした。②Blekhmanらの原著論文のSupplementary Table 1から得られるカウント数は、数十から数百というオーダーでした(スライド14)。しかし、recountから得られたカウント数は数千というオーダーで一桁大きく、なぜだろう?!とっていました(スライド65)。

bioconductor.org

scale_counts *Scale the raw counts provided by the recount project*

Description

In preparation for a differential expression analysis, you will have to choose how to scale the raw counts provided by the recount project. Note that the raw counts are the sum of the base level coverage so you have to take into account the read length or simply the total coverage for the given sample (default option). You might want to do some further scaling to take into account the gene or exon lengths. If you prefer to calculate read counts without scaling check the function [read_counts](#).

Usage

```
scale_counts(rse, by = "auc", targetSize = 4e+07, L = 100, factor_only = FALSE, round = TRUE)
```

Arguments

rse	A RangedSummarizedExperiment -class object as downloaded with download_study .
by	Either <code>auc</code> or <code>mapped_reads</code> . If set to <code>auc</code> it will scale the counts by the total coverage of the sample. That is, the area under the curve (AUC) of the coverage. If set to <code>mapped_reads</code> it will scale the counts by the number of mapped reads, whether the library was paired-end or not, and the desired read length (L).
targetSize	The target library size in number of single end reads.
L	The target read length. Only used when <code>by = 'mapped_reads'</code> since it cancels out in the calculation when using <code>by = 'auc'</code> .
factor_only	Whether to only return the numeric scaling factor or to return a RangedSummarizedExperiment class object with the counts scaled. If set to <code>TRUE</code> , you have to multiply the sample counts by this scaling factor.
round	Whether to round the counts to integers or not.

SRP001558 (Blekhman et al., *Genome Res.*, **20**: 180-9, 2010)

scale_counts

①の部分の記述を見て納得。Coverageの意味はよく分からないが、`recount`で提供している生のカウント数は、遺伝子領域内にマップされたリードの総塩基数(マップされたリード数ではない)をカウントしたもの(をベースとしている)だと判断した。総塩基数であれば、総リード数の10倍以上の数値になって然るべきだからです。

scale_counts *Scale the raw counts provided by the recount project*

Description

In preparation for a differential expression analysis, you will have to choose how to scale the raw counts provided by the recount project. **Note that the raw counts are the sum of the base level coverage** so you have to take into account the read length or simply the total coverage for the given sample (default option). You might want to do some further scaling to take into account the gene or exon lengths. If you prefer to calculate read counts without scaling check the function `read_counts`.

Usage

```
scale_counts(rse, by = "auc", targetSize = 4e+07, L = 100,
             factor_only = FALSE, round = TRUE)
```

Arguments

<code>rse</code>	A <code>RangedSummarizedExperiment</code> -class object as downloaded with <code>download_study</code> .
<code>by</code>	Either <code>auc</code> or <code>mapped_reads</code> . If set to <code>auc</code> it will scale the counts by the total coverage of the sample. That is, the area under the curve (AUC) of the coverage. If set to <code>mapped_reads</code> it will scale the counts by the number of mapped reads, whether the library was paired-end or not, and the desired read length (<code>L</code>).
<code>targetSize</code>	The target library size in number of single end reads.
<code>L</code>	The target read length. Only used when <code>by = 'mapped_reads'</code> since it cancels out in the calculation when using <code>by = 'auc'</code> .
<code>factor_only</code>	Whether to only return the numeric scaling factor or to return a <code>RangedSummarizedExperiment</code> class object with the counts scaled. If set to <code>TRUE</code> , you have to multiply the sample counts by this scaling factor.
<code>round</code>	Whether to round the counts to integers or not.

UsageとArguments

①Usage (利用法)と②Arguments (引数)。③rse というものが入カデータに相当し、それ以外の④byや⑤targetSizeなどがオプションに相当する。

The screenshot shows a PDF viewer displaying the documentation for the `scale_counts` function. The left sidebar contains a list of documents, with `scale_counts` selected. The main content area is titled "scale_counts" and "Scale the raw counts provided by the recount project". It includes a "Description" section, a "Usage" section, and an "Arguments" section. Red arrows are overlaid on the image to highlight specific parts: arrow 1 points to the "Usage" section header; arrow 2 points to the "Arguments" section header; arrow 3 points to the `rse` parameter in the function signature; arrow 4 points to the `by` parameter; and arrow 5 points to the `targetSize` parameter.

Usage

```
scale_counts(rse, by = "auc", targetSize = 4e+07, L = 100, factor_only = FALSE, round = TRUE)
```

Arguments

<code>rse</code>	A <code>RangedSummarizedExperiment</code> -class object as downloaded with <code>download_study</code> .
<code>by</code>	Either <code>auc</code> or <code>mapped_reads</code> . If set to <code>auc</code> it will scale the counts by the total coverage of the sample. That is, the area under the curve (AUC) of the coverage. If set to <code>mapped_reads</code> it will scale the counts by the number of mapped reads, whether the library was paired-end or not, and the desired read length (L).
<code>targetSize</code>	The target library size in number of single end reads.
<code>L</code>	The target read length. Only used when <code>by = 'mapped_reads'</code> since it cancels out in the calculation when using <code>by = 'auc'</code> .
<code>factor_only</code>	Whether to only return the numeric scaling factor or to return a <code>RangedSummarizedExperiment</code> class object with the counts scaled. If set to <code>TRUE</code> , you have to multiply the sample counts by this scaling factor.
<code>round</code>	Whether to round the counts to integers or not.

UsageとArguments

①入力のrseは、②RangedSummarizedExperiment-classオブジェクトと判断。rse_gene.Rdataをロードした後に使えるようになるオブジェクトのことですね。

scale_counts *Scale the raw counts provided by the recount project*

Description

In preparation for a differential expression analysis, you will have to choose how to scale the raw counts provided by the recount project. Note that the raw counts are the sum of the base level coverage so you have to take into account the read length or simply the total coverage for the given sample (default option). You might want to do some further scaling to take into account the gene or exon lengths. If you prefer to calculate read counts without scaling check the function [read_counts](#).

Usage

```
scale_counts(rse, by = "auc", targetSize = 4e+07, L = 100,
             factor_only = FALSE, round = TRUE)
```

Arguments

rse	A RangedSummarizedExperiment -class object as downloaded with download_study .
by	Either <code>auc</code> or <code>mapped_reads</code> . If set to <code>auc</code> it will scale the counts by the total coverage of the sample. That is, the area under the curve (AUC) of the coverage. If set to <code>mapped_reads</code> it will scale the counts by the number of mapped reads, whether the library was paired-end or not, and the desired read length (L).
targetSize	The target library size in number of single end reads.
L	The target read length. Only used when <code>by = 'mapped_reads'</code> since it cancels out in the calculation when using <code>by = 'auc'</code> .
factor_only	Whether to only return the numeric scaling factor or to return a RangedSummarizedExperiment -class object with the counts scaled. If set to <code>TRUE</code> , you have to multiply the sample counts by this scaling factor.
round	Whether to round the counts to integers or not.

①hogeが、②RangedSummarizedExperiment (RSE)形式のオブジェクトです。なので、scale_counts(hoge)とやれば実行できる。

hogeのこと

3. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

ウェブサイト [recount2](#) 上で `SRP001558` で検索し、`gene` 列の `RSE v2` のところからダウンロードして得られた `gene` レベルカウントデータ (`rse_gene.Rdata`; 約3MB) を読み込んで、カウントの数値行列情報 (58,037 genes × 11 samples) のみをタブ区切りテキストファイルで保存するやり方です。出力ファイルは `hoge3.txt` です。

```

in_f <- "rse_gene.Rdata"
out_f <- "hoge3.txt"

#必要なパッケージをロード
library(recount)

#本番(typeで指定した名前の.Rdataをロード)
load(in_f)
hoge <- rse_gene
hoge

#本番のカウントデータ取得
data <- assays(hoge)$counts
dim(data)
head(data)

#ファイルに保存
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", append=F,

```

#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納

#パッ

#in
#hog
#確認

#カ
#行
#確認

#保存

```

> hoge
class: RangedSummarizedExperiment
dim: 58037 11
metadata(0):
assays(1): counts
rownames(58037): ENSG00000000003.14
                ENSG00000000005.5 ... ENSG00000283698.1
                ENSG00000283699.1
rowData names(3): gene_id bp_length symbol
colnames(11): SRR032116 SRR032118 ... SRR032126
              SRR032127
colData names(21): project sample ... title
                   characteristics
> |

```

#確認してるだ\$

Value (返り値)

(本当は次のスライドの結果を見てから改めて確認するのだが...) ①factor_onlyオプションのデフォルトがFALSEであることを知ったうえで、②半ページほど下に移動。

bioconductor.org

scale_counts *Scale the raw counts provided by the recount project*

Description

In preparation for a differential expression analysis, you will have to choose how to scale the raw counts provided by the recount project. Note that the raw counts are the sum of the base level coverage so you have to take into account the read length or simply the total coverage for the given sample (default option). You might want to do some further scaling to take into account the gene or exon lengths. If you prefer to calculate read counts without scaling check the function [read_counts](#).

Usage

```
scale_counts(rse, by = "auc", targetSize = 4e+07, L = 100,
             factor_only = FALSE, round = TRUE)
```

Arguments

rse	A <code>RangedSummarizedExperiment</code> -class object as downloaded with <code>download_study</code> .
by	Either <code>auc</code> or <code>mapped_reads</code> . If set to <code>auc</code> it will scale the counts by the total coverage of the sample. That is, the area under the curve (AUC) of the coverage. If set to <code>mapped_reads</code> it will scale the counts by the number of mapped reads, whether the library was paired-end or not, and the desired read length (L).
targetSize	The target library size in number of single end reads.
L	The target read length. Only used when <code>by = 'mapped_reads'</code> since it cancels out in the calculation when using <code>by = 'auc'</code> .
factor_only	Whether to only return the numeric scaling factor or to return a <code>RangedSummarized</code> class object with the counts scaled. If set to <code>TRUE</code> , you have to multiply the sample counts by this scaling factor.
round	Whether to round the counts to integers or not.

SRP001558 (Blekhman et al., *Genome Res.*, **20**: 180-9, 2010)

Value (返り値)

①ここがscale_counts実行結果としてどのようなものが返されるかを記したところ。②出力は、カウントデータのところがスケールされた状態のRangedSummarizedExperimentオブジェクトだということがわかる。

The screenshot shows a PDF viewer displaying the documentation for the `scale_counts` function. The left sidebar contains a list of functions, with `scale_counts` selected. The main content area shows the following details:

round Whether to round the counts to integers or not.

Details

Rail-RNA <http://rail.bio> uses soft clipping when aligning which is why we recommend using `by = 'auc'`.

If the reads are from a paired-end library, then the `avg_read_length` is the average fragment length. This is taken into account when using `by = 'mapped_reads'`.

Value

If `factor_only = TRUE` it returns a numeric vector with the scaling factor for each sample. If `factor_only = FALSE` it returns a `RangedSummarizedExperiment`-class object with the counts already scaled.

Author(s)

210 x 297 mm

Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?!カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ(SRP001558)をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明(前半)
 - RangedSummarizedExperimentオブジェクトの説明(後半)、例題4
 - 例題5、例題6、ヒト(計6人分)のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方(scale_countsを例に)、例題7と8
 - 課題1(getRPKM関の入出力)、課題2(RSE)、課題3(例題7と8のクラスタリング結果)
 - ERP000546(ヒトの様々な器官由来のRNA-seqカウントデータ)からの情報抽出
- クラスタリング結果の客観的な評価指標(Silhouettes)

例題7

(Rで)塩基配列解析

(last modified 2018/06/11, since 2010)

このウェブ
ツール済
系的に

- マップ後 | カウント 情報取得 | トランスクリプトーム | [BEDファイルから](#) (last modified 2014/0)
- [カウント 情報取得 | について](#) (last modified 2018/06/10) **NEW**
- カウント 情報取得 | リアルデータ | SRP001558 | [recount\(Collado-Torres 2017\)](#) (last modified 2018/06/11) **NEW**
- カウント 情報取得 | リアルデータ | ERP000546 | [recount\(Collado-Torres 2017\)](#) (last modified 2018/06/11) **NEW**
- カウント 情報取得 | [シミュレーションデータ](#) (last modified 2018/06/09) **NEW**



What's

• 以下の
の.Rd
SRP0

カウント情報取得 | リアルデータ | SRP001558 | [recount\(Collado-Torres_2017\)](#) **NEW**

[recount](#)パッケージを用いて、[SRP001558\(Blekhman et al., Genome Res., 2010\)](#); ブラウザはIE以外を推奨)のカウント 情報を含む RangedSummarizedExperimentクラスオブジェクトという形式の.Rdataをダウンロードしたり、カウントデータの数値行列にした状態で保存するやり方を示します。原著論文では、3生物種(ヒト12 samples、チンパンジー12 samples、そしてアカゲザル12 samples)のカウントデータを取得しています。ウェブサイト [recount2](#) 上でSRP001558で検索すると、number of samplesが12、speciesが



7. ダウンロード済みの [rse_gene.Rdata](#) を入力として読み込む場合:

例題5の発展形として、[recount\(R package\)](#) の [recount quick start guide](#) のHTML で書かれているscale_counts関数実行結果を返すやり方です。58,037 genes×11 samplesからなる出力ファイルは[hoge7.txt](#)です。

```

in_f <- "rse_gene.Rdata"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge7.txt"          #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount)              #パッケージの読み込み

#入力ファイルの読み込み(.Rdata)
load(in_f)                     #in_fで指定した.Rdataをロード
hoge <- rse_gene                #hogeとして取り扱う
colSums(assays(hoge)$counts)    #列ごとの総和を表示

#本番(scaling)
hoge2 <- scale_counts(hoge)     #scale_counts実行結果をhoge2に格納
colSums(assays(hoge2)$counts)  #列ごとの総和を表示

#後処理(scaling後のカウントデータ取得と列名変更)

```

例題7

7. ダウンロード済みの `rse_gene.Rdata` を入力として

例題5の発展形として、`recount`(R package) の `reco` です。58,037 genes×11 samplesからなる出力ファイル

```
in_f <- "rse_gene.Rdata"
out_f <- "hoge7.txt"

#必要なパッケージをロード
library(recount)

#入力ファイルの読み込み(.Rdata)
load(in_f)
hoge <- rse_gene
colSums(assays(hoge)$counts)

#本番(scaling)
hoge2 <- scale_counts(hoge)
colSums(assays(hoge2)$counts)

#後処理(scaling後のカウントデータ取得と列名変更)
data <- assays(hoge2)$counts
colnames(data) <- colData(hoge2)$title

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", app
```

`scale_counts`実行①前と②後で、総カウント数が1/5程度になっていることがわかる。計算の詳細は不明であるが、RPMで総カウント数が100万になるはずである。そして、このデータのリード長が36 bpであり、リード全長が遺伝子領域内にマップされるわけではないことを鑑みると、②スケーリング後に3300万程度という結果は妥当といえるだろう。などと、とりあえずつじつまを合わせて納得する。

```
> colSums(assays(hoge)$counts) #列ごとの総和を表示
SRR032116 SRR032118 SRR032119 SRR032120 SRR032121 SRR032122
148579016 140429303 158304396 171031517 178046675 126119617
SRR032123 SRR032124 SRR032125 SRR032126 SRR032127
139987135 196322399 192811889 185154022 171021549
>
> #本番(scaling)
> hoge2 <- scale_counts(hoge) #scale_counts実行$
> colSums(assays(hoge2)$counts) #列ごとの総和を表示
SRR032116 SRR032118 SRR032119 SRR032120 SRR032121 SRR032122
33318293 33522269 33622869 31356176 31462101 33546520
SRR032123 SRR032124 SRR032125 SRR032126 SRR032127
33588861 32491430 32533849 30336521 30375223
>
> #後処理(scaling後のカウントデータ取得と列名変更)
> data <- assays(hoge2)$counts #カウントデータ行$
> colnames(data) <- colData(hoge2)$title #列名を変更
>
> #ファイルに保存(カウントデータ)
> tmp <- cbind(rownames(data), data) #保存したい情報をt$
> write.table(tmp, out_f, sep="\t", append=F, quote=F, row.$
> |
```


例題7

① `scale_counts` 実行結果の `hoge2` オブジェクトは、入力の `hoge` オブジェクトと同じく `RangedSummarizedExperiment (RSE)` オブジェクトです。その証拠が、② 列の総和を算出する `colSums` 関数実行時の入力が、③ `assays(hoge2)$counts` です。逆にいえば、`colSums` 関数は `RSE` オブジェクトを入力として受け付けません。

7. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む

例題5の発展形として、`recount` (R package) の `recount quick` です。58,037 genes × 11 samples からなる出力ファイルは `hoge`

```
in_f <- "rse_gene.Rdata"
out_f <- "hoge7.txt"

#必要なパッケージをロード
library(recount)

#入力ファイルの読み込み(.Rdata)
load(in_f)
hoge <- rse_gene
colSums(assays(hoge)$counts)

#本番 (scaling)
hoge2 <- scale_counts(hoge)
colSums(assays(hoge2)$counts)

#② 処理 (scaling) 後 ③ カウントデータ取得と列名変更
data <- assays(hoge2)$counts
colnames(data) <- colData(hoge2)$title

#ファイルに保存 (カウントデータ)
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", app
```

```
R Console
> colSums(assays(hoge)$counts) #列ごとの総和を表示
SRR032116 SRR032118 SRR032119 SRR032120 SRR032121 SRR032122
148579016 140429303 158304396 171031517 178046675 126119617
SRR032123 SRR032124 SRR032125 SRR032126 SRR032127
139987135 196322399 192811889 185154022 171021549
>
> #本番 (scaling)
> hoge2 <- scale_counts(hoge) #scale_counts実行$
> colSums(assays(hoge2)$counts) #列ごとの総和を表示
SRR032116 SRR032118 SRR032119 SRR032120 SRR032121 SRR032122
33318293 33522269 33622869 31356176 31462101 33546520
SRR032123 SRR032124 SRR032125 SRR032126 SRR032127
33588861 32491430 32533849 30336521 30375223
>
> #後処理 (scaling後のカウントデータ取得と列名変更)
> data <- assays(hoge2)$counts #カウントデータ行$
> colnames(data) <- colData(hoge2)$title #列名を変更
>
> #ファイルに保存 (カウントデータ)
> tmp <- cbind(rownames(data), data) #保存したい情報をt$
> write.table(tmp, out_f, sep="\t", append=F, quote=F, row.$
> |
```

例題8

8. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題5の発展形として、`recount`(R package) の [recount quick start guide](#) の [HTML](#) で書かれている `getRPKM` 関数実行結果を返すやり方です。58,037 genes×11 samplesからなる出力ファイルは `hoge8.txt` です。

```

in_f <- "rse_gene.Rdata"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge8.txt"         #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount)             #パッケージの読み込み

#入力ファイルの読み込み(.Rdata)
load(in_f)                   #in_fで指定した.Rdataをロード
hoge <- rse_gene              #hogeとして取り扱う
colSums(assays(hoge)$counts)  #列ごとの総和を表示

#本番(RPKM値の取得)
data <- getRPKM(hoge)        #getRPKM実行結果をdataに格納
colSums(data)                #列ごとの総和を表示

#後処理(列名変更)
colnames(data) <- colData(hoge)$title #列名を変更

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data) #保存したい情報をtmpに格納
write.table(tmp, out_f, sep="\t", append=F, quote=F, row.names=F) #tmpの中身を指定したファイル名で保存

```



例題8

①getRPKM関数の実行結果であるdataオブジェクトは、②colSums関数の入力としてそのまま使われている。このことから、③dataオブジェクトはRangedSummarizedExperiment (RSE)オブジェクトではないことがわかる。

8. ダウンロード済みの [rse_gene.Rdata](#) を入力として読み込む場合:

例題5の発展形として、[recount](#) (R package) の [recount quick start guide](#) の [HTML](#) で書かれているgetRPKM関数実行結果を返すやり方です。58,037 genes×11 samplesからなる出力ファイルは [hoge8.txt](#) です。

```

in_f <- "rse_gene.Rdata"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge8.txt"          #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount)              #パッケージの読み込み

#入力ファイルの読み込み(.Rdata)
load(in_f)                    #in_fで指定した.Rdataをロード
hoge <- rse_gene               #hogeとして取り扱う
colSums(assays(hoge)$counts)   #列ごとの総和を表示

#本番(RPKM値の取得)
data <- getRPKM(hoge)         #getRPKM実行結果をdataに格納
colSums(data)                 #列ごとの総和を表示

#(変更)
colnames(data) <- colData(hoge)$title #列名を変更

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data) #保存したい情報をtmpに格納
write.table(tmp, out_f, sep="\t", append=F, quote=F, row.names=F) #tmpの中身を指定したファイル名で保存
    
```



例題8実行結果

こんな感じになります。①colSums実行結果はRPKM値っぽい値になっていることがわかります。RPMでカウントの総和が100万となるが、RPKで1000 bpの長さだったときの長さにも補正しているのがRPKM。平均的なヒト遺伝子の長さが2000 bp程度だったのだと考えれば妥当でしょう。

8. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題5の発展形として、`recount`(R package) の `recount quick start guide` を参照。58,037 genes×11 samplesからなる出力ファイルは `hoge8.txt` です。

```
in_f <- "rse_gene.Rdata"
out_f <- "hoge8.txt"

#必要なパッケージをロード
library(recount)

#入力ファイルの読み込み(.Rdata)
load(in_f)
hoge <- rse_gene
colSums(assays(hoge)$counts)

#本番(RPKM値の取得)
data <- getRPKM(hoge)
colSums(data)

#後処理(列名変更)
colnames(data) <- colData(hoge)$title

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", app
```

```
R Console

> hoge <- rse_gene #hogeとして取り扱う
> colSums(assays(hoge)$counts) #列ごとの総和を表示
SRR032116 SRR032118 SRR032119 SRR032120 SRR032121 SRR032122
148579016 140429303 158304396 171031517 178046675 126119617
SRR032123 SRR032124 SRR032125 SRR032126 SRR032127
139987135 196322399 192811889 185154022 171021549
>
> #本番 (RPKM値の取得)
> data <- getRPKM(hoge) #getRPKM実行結果を$
> colSums(data) #列ごとの総和を表示
SRR032116 SRR032118 SRR032119 SRR032120 SRR032121 SRR032122
464186.3 454446.2 454888.3 632497.6 636818.8 475629.5
SRR032123 SRR032124 SRR032125 SRR032126 SRR032127
477651.7 488678.6 489803.6 487942.1 488657.3
>
> #後処理 (列名変更)
> colnames(data) <- colData(hoge)$title #列名を変更
>
> #ファイルに保存 (カウントデータ)
> tmp <- cbind(rownames(data), data) #保存したい情報をt$
> write.table(tmp, out_f, sep="\t", append=F, quote=F, row.$
> |
```

Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?! カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

課題1

①getRPKM関数の入力として与えるものをRのクラスオブジェクト名で答えよ(大文字小文字の区別やスペルミスに注意)。また、出力(getRPKMの返り値)の形式についても答えよ(Value欄の記載内容そのままでもよい)。②ヒント

8. ダウンロード済みの [rse_gene.Rdata](#) を入力として読み込む場合:

例題5の発展形として、[recount](#)(R package)の[recount quick start guide](#)のHTMLで書かれているgetRPKM関数実行結果を返すやり方です。58,037 genes×11 samplesからなる出力ファイルは[hoge8.txt](#)です。

```
in_f <- "rse_gene.Rdata"
out_f <- "hoge8.txt"

#必要なパッケージをロード
library(recount)

#入力ファイルの読み込み(.Rdata)
load(in_f)
hoge <- rse_gene
colSums(assays(hoge)$counts)

#本番(RPKM値の取得)
data <- getRPKM(hoge)
colSums(data)

#後処理(列名変更)
colnames(data) <- colData(hoge)$title

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", app
```



```
R Console
>
> #本番 (RPKM値の取得)
> data <- getRPKM(hoge) #getRPKM実行結果を$
> colSums(data) #列ごとの総和を表示
SRR032116 SRR032118 SRR032119 SRR032120 SRR032121 SRR032122
464186.3 454446.2 454888.3 632497.6 636818.8 475629.5
SRR032123 SRR032124 SRR032125 SRR032126 SRR032127
477651.7 488678.6 489803.6 487942.1 488657.3
>
> #後処理 (列名変更)
> colnames(data) <- colData(hoge)$title #列名を変更
>
> #ファイルに保存 (カウントデータ)
> tmp <- cbind(rownames(data), data) #保存したい情報をt$
> write.table(tmp, out_f, sep="\t", append=F, quote=F, row.$
> ?getRPKM
starting httpd help server ... done
> is.data.frame(data)
[1] FALSE
> is.matrix(data)
[1] TRUE
> |
```

課題2

①getRPKM関数がRPKM値を出力可能なのは、課題1で答えたRのクラスオブジェクトがカウントデータ以外に遺伝子の何の情報を保持しているためか答えよ。

8. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題5の発展形として、`recount`(R package) の [recount quick start guide](#) の HTML で書かれている `getRPKM` 関数実行結果を返すやり方です。58,037 genes×11 samples からなる出力ファイルは `hoge8.txt` です。

```
in_f <- "rse_gene.Rdata"
out_f <- "hoge8.txt"

#必要なパッケージをロード
library(recount)

#入力ファイルの読み込み(.Rdata)
load(in_f)
hoge <- rse_gene
colSums(assays(hoge)$counts)

#本番(RPKM値の取得)
data <- getRPKM(hoge)
colSums(data)

#後処理(列名変更)
colnames(data) <- colData(hoge)$title

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data)
write.table(tmp, out_f, sep="\t", app
```



```
R Console
>
> #本番 (RPKM値の取得)
> data <- getRPKM(hoge) #getRPKM実行結果を$
> colSums(data) #列ごとの総和を表示
SRR032116 SRR032118 SRR032119 SRR032120 SRR032121 SRR032122
464186.3 454446.2 454888.3 632497.6 636818.8 475629.5
SRR032123 SRR032124 SRR032125 SRR032126 SRR032127
477651.7 488678.6 489803.6 487942.1 488657.3
>
> #後処理 (列名変更)
> colnames(data) <- colData(hoge)$title #列名を変更
>
> #ファイルに保存(カウントデータ)
> tmp <- cbind(rownames(data), data) #保存したい情報をt$
> write.table(tmp, out_f, sep="\t", append=F, quote=F, row.$
> ?getRPKM
starting httpd help server ... done
> is.data.frame(data)
[1] FALSE
> is.matrix(data)
[1] TRUE
> |
```

課題3

①例題7と②例題8で得られた、③scaling後のカウントデータ (hoge7.txt) および④RPKMデータ (hoge8.txt) を入力としてサンプル間クラスタリングを行い、対応する例題5の生のカウントデータ (hoge5.txt) の結果と比較し、簡単に考察せよ。

7. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合

①の発展形として、`recount`(R package) の [recount quick start guide](#) のHTML で書かれている `scale_counts` 関数実行結果を返すやり方です。58,037 genes×11 samples からなる出力ファイルは `hoge7.txt` です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge7.txt" #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount) #パッケージの読み込み
```

8. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

②の発展形として、`recount`(R package) の [recount quick start guide](#) のHTML で書かれている `getRPKM` 関数実行結果を返すやり方です。58,037 genes×11 samples からなる出力ファイルは `hoge8.txt` です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge8.txt" #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount) #パッケージの読み込み

#入力ファイルの読み込み(.Rdata)
load(in_f) #in_fで指定した.Rdataをロード
hoge <- rse_gene #hogeとして取り扱う
colSums(assays(hoge)$counts) #列ごとの総和を表示

#本番(RPKM値の取得)
data <- getRPKM(hoge) #getRPKM実行結果をdataに格納
colSums(data) #列ごとの総和を表示

#後処理(列名変更)
colnames(data) <- colData(hoge)$title #列名を変更

#ファイルに保存(カウントデータ)
tmp <- cbind(rownames(data), data) #保存したい情報をtmpに格納
write.table(tmp, out_f, sep="\t", append=F, quote=F, row.names=F) #tmpの中身を指定したファイル名で保存
```

```
#入力ファイルの読み込み
load(in_f)
hoge <- rse_gene
colSums(assays(hoge))

#本番(scaling)
hoge2 <- scale_count
colSums(assays(hoge2))

#後処理(scaling後のカ
data <- assays(hoge2)
colnames(data) <- col

#ファイルに保存(カウ
tmp <- cbind(rowname
write.table(tmp, out
```


Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?! カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

ERP000546

①ERP000546(ヒトの様々な器官由来のRNA-seqカウントデータ)の場合についても、是非！自分で例題1から順番にやってみてください。ERP000546の例題5で得られるカウントデータと、SRP001558の例題6で得られるカウントデータはマージ可能(列方向で結合してサンプル間クラスタリングできる)です。ERP000546にもliverサンプルが含まれています。SRP001558の例題6で得られる6 liver samplesが、どのような位置にくるのか見てみてください。

(Rで)塩基配列解析

(last modified 2018/05/30, since 2010)

このウェブページを
なパッケージを
版)で自習して

- マップ後 | カウント情報取得 | paired-end
- マップ後 | カウント情報取得 | paired-end
- マップ後 | カウント情報取得 | トランスクリプト
- [カウント情報取得 | について](#) (last modified 2018/06/10) **NEW**
- [カウント情報取得 | リアルデータ | SRP001558 | recount\(Collado-Torres 2017\)](#) (last modified 2018/06/10) **NEW**
- [カウント情報取得 | リアルデータ | ERP000546 | recount\(Collado-Torres 2017\)](#) (last modified 2018/06/10) **NEW**
- [カウント情報取得 | シミュレーションデータ](#) (last modified 2018/06/09) **NEW**
- [配列長とカウント数の関係](#) (last modified 2018/06/09) **NEW**
- [正規化 | について](#) (last modified 2014/06/22)
- [正規化 | 基礎 | RPK or CPK \(配列長補正\)](#) (last modified 2015/07/04)
- [正規化 | 基礎 | RPM or](#)
- [正規化 | 基礎 | RPKM](#)
- [正規化 | サンプル内 | H](#)
- [正規化 | サンプル内 | H](#)



What's new?

- 「マップ後 |
- 「イントロ |
- 「[H29年度](#) |
- [Silhouette](#) |

カウント情報取得 | リアルデータ | ERP000546 | recount(Collado-Torres_2017) **NEW**

recountパッケージを用いて、[ERP000546](#)(原著論文なし;ブラウザはIE以外を推奨)のカウント情報を含む RangedSummarizedExperimentクラスオブジェクトという形式の.Rdataをダウンロードしたり、カウントデータの数値行列にした状態で保存するやり方を示します。RangedSummarizedExperimentというのがよくわからないとは思いますが、この中にEnsemblなどのgene IDだけでなくgene symbolsや配列長情報なども含まれているので何かと便利なのです。
「ファイル」-「ディレクトリの変更」でダウンロードしたいディレクトリに移動し以下をコピー。

1. geneレベルカウントデータ情報を得たい場合:

[ERP000546](#)という名前のフォルダが作成されます。中にあるrse-gene.Rdataをロードして読み込むとrse-geneというオブジェクト名で取り扱えます。ウェブサイト [recount2](#) 上でERP000546で検索し、gene列の [RSE v2](#) をダウンロードして得られる rse_gene.Rdataと同じです。

```
param_ID <- "ERP000546" #IDを指定

#必要なパッケージをロード
library(recount) #パッケージの読み込み

#本番(.Rdataをダウンロード)
download_study(param_ID, type="rse-gene", download=T)#ダウンロード
```

ERP000546(ヒトの様々な器官由来のRNA-seqカウントデータ)の①例題5。②入力のrse_gene.RdataはERP000546用ですのでご注意ください。

例題5

②

5. ダウンロード済みのrse_gene.Rdataを入力として読み込む場合:

① 4で得られたサンプルのメタデータ情報ファイル(hoge4 meta samples.txt)中の ERR...からERS...の情報を手かりにして、hoge4 meta samples added.txtの1番右の列で示すような「これがheartで、これがkidneyで...」という対応関係をENAで1つ1つ調べたもので置き換えています。出力ファイルはhoge5.txtです。

```
in_f <- "rse_gene.Rdata"           #入力ファイル名を指定してin_fに格納
out_f <- "hoge5.txt"                #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount)                   #パッケージの読み込み

#入力ファイルの読み込み(.Rdata)
load(in_f)                          #in_fで指定した.Rdataをロード
hoge <- rse_gene                     #hogeとして取り扱う
hoge                                  #確認してるだけです

#本番(カウントデータ取得)
uge <- assays(hoge)$counts           #カウントデータ行列を取得してugeに格納
dim(uge)                             #行数と列数を表示
head(uge)                             #確認してるだけです

#後処理(同一ERS IDの列をマージ)
uge <- as.data.frame(uge)            #行列形式からデータフレーム形式に変更
data <- cbind(                       #必要な列名を取得したい列の順番で結合した新
  uge$ERR030885 + uge$ERR030893,     #kidney(ERS025081)
  uge$ERR030894 + uge$ERR030886,     #heart(ERS025082)
  uge$ERR030874 + uge$ERR030901,     #ovary(ERS025083)
```

例題5

①出力ファイル(hoge5.txt)は、②58,037 genes × 19 samplesからなるカウントデータ。

5. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む場合:

例題4で得られたサンプルのメタデータ情報ファイル(`hoge4 meta samples.txt`)中の ERR...から ERS...の情報を手がかかりにして、`hoge4 meta samples added.txt`の1番右の列で示すような「これがheartで、これがkidneyで...」という対応関係を `ENA` で1つ1つ調べたもので置き換えています。出力ファイルは `hoge5.txt` です。

```
in_f <- "rse_gene.Rdata" #入力ファイル名を指定してin_fに格納
out_f <- "hoge5.txt" #出力ファイル名を指定してout_fに格納

#必要なパッケージをロード
library(recount) #パッケージの読み込み

#入力ファイルの読み込み(.Rdata)
load(in_f)
hoge <- rse_gene
hoge

#本番(カウントデータ取得)
uge <- assays(hoge)$counts
dim(uge)
head(uge)

#後処理(同一ERS IDの列をマージ)
uge <- as.data.frame(uge)
data <- cbind(
  uge$ERR030885 + uge$ERR030893,
  uge$ERR030894 + uge$ERR030886,
  uge$ERR030874 + uge$ERR030901,
  #in_f #hogef #確認
  > colnames(data) <- c( #列名を付加
+ "kidney", "heart", "ovary", #列名を付加
+ "mixture1", "brain", "lymphnode", #列名を付加
+ "mixture2", "breast", "colon", #列名を付加
+ "thyroid", "white_blood_cells", #列名を付加
+ "adrenal", "mixture3", "testes", #列名を付加
+ "prostate", "liver", #列名を付加
+ "skeletal_muscle", "adipose", "lung") #列名を付加
  > rownames(data) <- rownames(uge) #行名を付加
  > dim(data) #行数と列数を$
  [1] 58037 19
  >
  > #ファイルに保存(カウントデータ)
  > tmp <- cbind(rownames(data), data) #保存したい情$
  > write.table(tmp, out_f, sep="\t", append=F, quote=F,$
  > |
```

```
R Console
> colnames(data) <- c(
+ "kidney", "heart", "ovary",
+ "mixture1", "brain", "lymphnode",
+ "mixture2", "breast", "colon",
+ "thyroid", "white_blood_cells",
+ "adrenal", "mixture3", "testes",
+ "prostate", "liver",
+ "skeletal_muscle", "adipose", "lung")
> rownames(data) <- rownames(uge)
> dim(data)
[1] 58037 19
>
> #ファイルに保存(カウントデータ)
> tmp <- cbind(rownames(data), data)
> write.table(tmp, out_f, sep="\t", append=F, quote=F,$
> |
```

例題5

①出力ファイル(hoge5.txt)は、②58,037 genes × 19 samplesからなるカウントデータ。③ここで見えているようなヒト組織のデータです。④58,037という数字に着目！Recountは生のリードデータから統一的な手順でカウントデータを得ているので、他のデータ(例:SRP001558)とほぼ直接的な比較ができます!

5. ダウンロード済みの `rse_gene.Rdata` を入力として読み込む

例題4で得られたサンプルのメタデータ情報ファイル(hoge4)がかりに、`hoge4 meta samples added.txt`の1番右の列と対応関係をENAで1つ1つ調べたもので置き換えています。出力ファイルは `hoge5.txt` です。

```
in_f <- "rse_gene.Rdata"
out_f <- "hoge5.txt"
```

```
#必要なパッケージをロード
library(recount)
```

```
#入力ファイルの読み込み(.Rdata)
load(in_f)
hoge <- rse_gene
hoge
```

```
#本番(カウントデータ取得)
uge <- assays(hoge)$counts
dim(uge)
head(uge)
```

```
#後処理(同一ERS IDの列をマージ)
uge <- as.data.frame(uge)
data <- cbind(
  uge$ERR030885 + uge$ERR030893,
  uge$ERR030894 + uge$ERR030886,
  uge$ERR030874 + uge$ERR030901,
```

```
#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納
```

```
#パッケージの読み込み
```

```
#in_f
#hoge
#確認
```

```
#カ
#行
#確認
```

```
#行列
#必要?
#k
#h
#ova
```

```
> colnames(data) <- c(
+   "kidney", "heart", "ovary",
+   "mixture1", "brain", "lymphnode",
+   "mixture2", "breast", "colon",
+   "thyroid", "white_blood_cells",
+   "adrenal", "mixture3", "testes",
+   "prostate", "liver",
+   "skeletal_muscle", "adipose", "lung")
```

```
#列名を付加
#列名を付加
#列名を付加
#列名を付加
#列名を付加
#列名を付加
#列名を付加
#列名を付加
#列名を付加
#列名を付加
#列名を付加
#行名を付加
#行数と列数を$
```

```
> rownames(data) <- rownames(uge)
> dim(data)
[1] 58037 19
>
> #ファイルに保存(カウントデータ)
> tmp <- cbind(rownames(data), data) #保存したい情$
> write.table(tmp, out_f, sep="\t", append=F, quote=F,$
> |
```

サンプル間クラスタリング

解析 | クラスタリング | サンプル間 | TCC(Sun_2013) ① W

①の例題5をテンプレートに、さきほどの
②hoge5.txtを入力として、pngファイルの
大きさを500×400にして実行した結果。
③liverはここに位置しています。

TCCパッケージを用いてサンプル間クラスタリングを行うやり方を示します。clusterSample関数を利用した頑健なクラスタリング結果を返します。

「ファイル」-「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し

1. 59,857 genes×6 samplesのリアルデータ(srp017142_count_bowtie.txt)の場合:

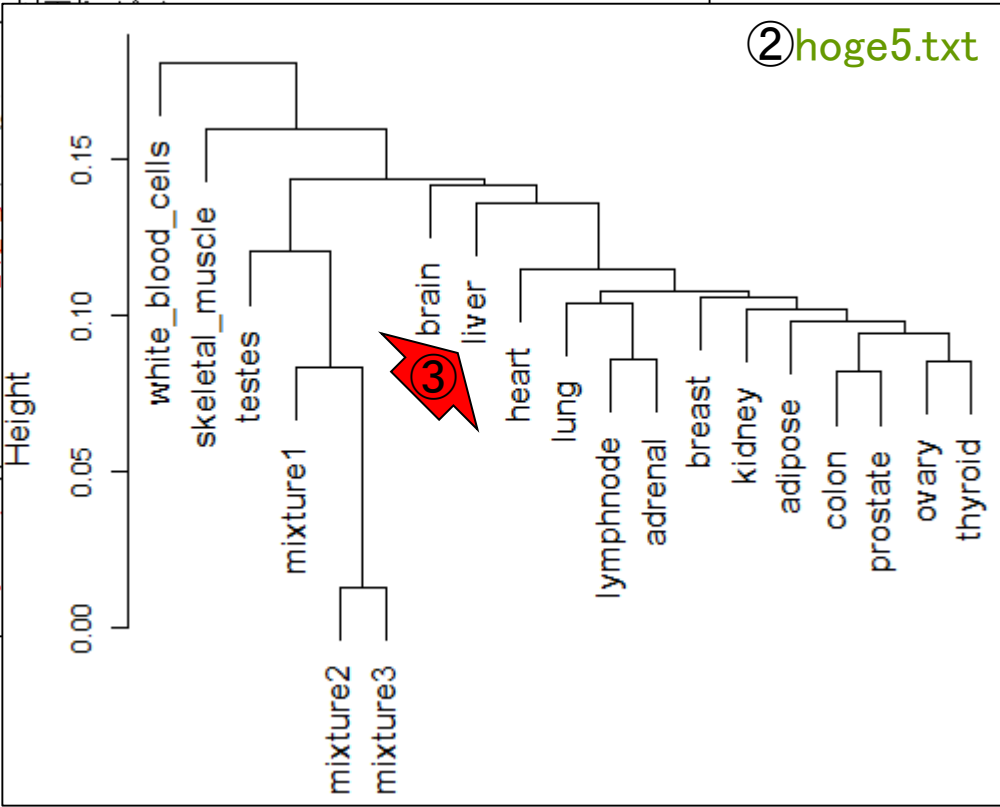
[Neyret-Kahn et al. Genome Res. 2013の2群間比較用\(3 proliferative samples vs. 3 Reproductive samples\)](#)
[ゲノム | 発現変動 | 2群間 | 対応なし | 複製あり | SRP017142\(Neyret-Kahn_2013\)](#)から

```
in_f <- "srp017142_count_bowtie.txt" #入力ファイル名を指定してin
out_f <- "hoge1.png" #出力ファイル名を指定してou
param_fig <- c(500, 400) #ファイル出力時の横幅と縦幅
```

```
#必要なパッケージをロード
library(TCC) #パッケージの読み込み
```

```
#入力ファイルの読み込み
data <- read.table(in_f, header=TRUE, row.names=1, sep="\t", quote="\"", as.is=TRUE)
dim(data) #オブジェクトdataの行数と列数
```

```
#本番
out <- clusterSample(data, dist.method="spearman", #クラスタリング
                     hclust_method="average", unique_patterns=TRUE) #クラスタリング
```



サンプル間クラスタ

解析 | クラスタリング | サンプル間 | TCC(Sun_2013)

①Blekhmanらの6 liver samplesのデータをマージ(列方向で結合)して、改めてクラスタリングを行った結果。②オリジナルの19 samples中のliverの位置は変わっていますが、ものの見事にliverの計7 samplesでクラスターを形成していることが分かります。

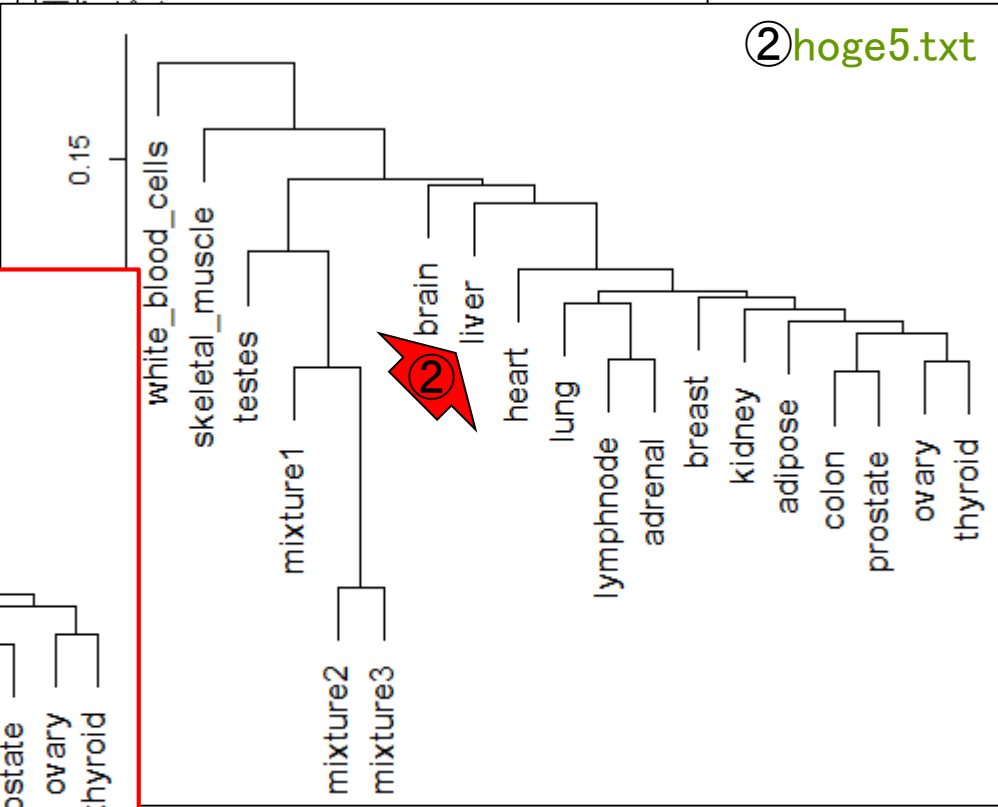
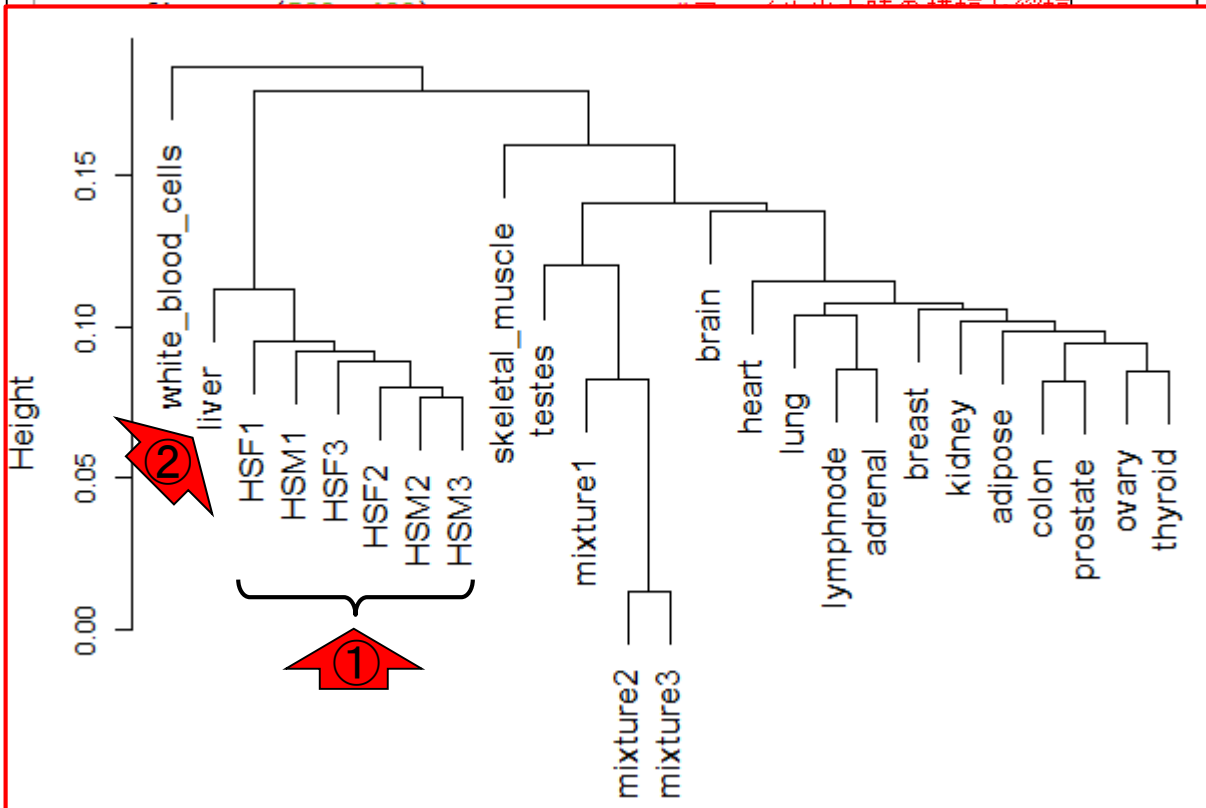
TCCパッケージを用いてサンプル間クラスタリングを行うやり方を示します。clus

「ファイル」-「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し

1. 59,857 genes×6 samplesのリアルデータ([srp017142_count_bowtie.txt](#))の場合:

[Neyret-Kahn et al., Genome Res., 2013](#)の2群間比較用(3 proliferative samples vs. 3 Reproductive samples)のRNA-seqデータ(2群間比較用)の発現変動(2群間比較)の対応なし(複製あり)のSRP017142(Neyret-Kahn_2013)の

```
in_f <- "srp017142_count_bowtie.txt" #入力ファイル名を指定してin
out_f <- "hoge1.png" #出力ファイル名を指定してou
```



②hoge5.txt

Contents

- サンプル間クラスタリング
 - Liverの3生物種間比較データ (technical replicates マージ前)
 - Liverの3生物種間比較データ (technical replicates マージ後)
- 公共?! カウントデータセット
 - Recount、recount2
 - Liverの3生物種間比較データ (SRP001558) をrecount2で眺める
 - SRP001558のrse_gene.Rdataからの情報抽出
 - 例題1と3、RangedSummarizedExperimentオブジェクトの説明 (前半)
 - RangedSummarizedExperimentオブジェクトの説明 (後半)、例題4
 - 例題5、例題6、ヒト (計6人分) のデータのみでサンプル間クラスタリング
 - RパッケージのReference Manualの見方 (scale_countsを例に)、例題7と8
 - 課題1 (getRPKM関の入出力)、課題2 (RSE)、課題3 (例題7と8のクラスタリング結果)
 - ERP000546 (ヒトの様々な器官由来のRNA-seqカウントデータ) からの情報抽出
- クラスタリング結果の客観的な評価指標 (Silhouettes)

客観的な評価

クラスタリング結果(樹形図)を眺めて、興味あるグループ間の関係性(特にDEG検出結果)を客観的に評価する指標として、シルエットスコア(Silhouette score)が有用だということを示した論文。

Biol Proced Online. 2018 Mar 1;20:5. doi: 10.1186/s12575-018-0067-8. eCollection 2018.

Silhouette Scores for Arbitrary Defined Groups in Gene Expression Data and Insights into Differential Expression Results.

Zhao S¹, Sun J¹, Shimizu K¹, Kadota K¹.

Author information

Abstract

BACKGROUND: Hierarchical Sample clustering (HSC) is widely performed to examine associations within expression data obtained from microarrays and RNA sequencing (RNA-seq). Researchers have investigated the HSC results with several possible criteria for grouping (e.g., sex, age, and disease types). However, the evaluation of arbitrary defined groups still counts in subjective visual inspection.

RESULTS: To objectively evaluate the degree of separation between groups of interest in the HSC dendrogram, we propose to use *Silhouette* scores. *Silhouettes* was originally developed as a graphical aid for the validation of data clusters. It provides a measure of how well a sample is classified when it was assigned to a cluster by according to both the tightness of the clusters and the separation between them. It ranges from 1.0 to -1.0, and a larger value for the average silhouette (AS) over all samples to be analyzed indicates a higher degree of *cluster* separation. The basic idea to use an AS is to replace the term *cluster* by *group* when calculating the scores. We investigated the validity of this score using simulated and real data designed for differential expression (DE) analysis. We found that larger (or smaller) AS values agreed well with both higher (or lower) degrees of separation between different groups and higher percentages of differentially expressed genes (P_{DEG}). We also found that the AS values were generally independent on the number of replicates (N_{rep}). Although the P_{DEG} values depended on N_{rep} , we confirmed that both AS and P_{DEG} values were close to zero when samples in the data showed an intermingled nature between the groups in the HSC dendrogram.

CONCLUSION: *Silhouettes* is useful for exploring data with predefined group labels. It would help provide both an objective evaluation of HSC dendrograms and insights into the DE results with regard to the compared groups.

KEYWORDS: Bioinformatics; Differential expression analysis; Hierarchical sample clustering; Silhouettes

PMID: 29507534 PMCID: PMC5831220 DOI: 10.1186/s12575-018-0067-8

Read free
full text at



PMC Full text

FREE

Save items

★ Add to Favorites

Similar articles

How frequently do clusters occur in hierarchical clusterin [J Cheminform. 2016]

Evaluation of methods for differential expression analysis [BMC Bioinformatics. 2015]

Knowledge-assisted recognition of cluster boundaries in gene [Artif Intell Med. 2005]

Silhouette scores for assessment of SNP genotype clusters. [BMC Genomics. 2005]

Review [Aiming for zero blindness]. [Nippon Ganka Gakkai Zasshi. 2015]

See reviews...

See all...

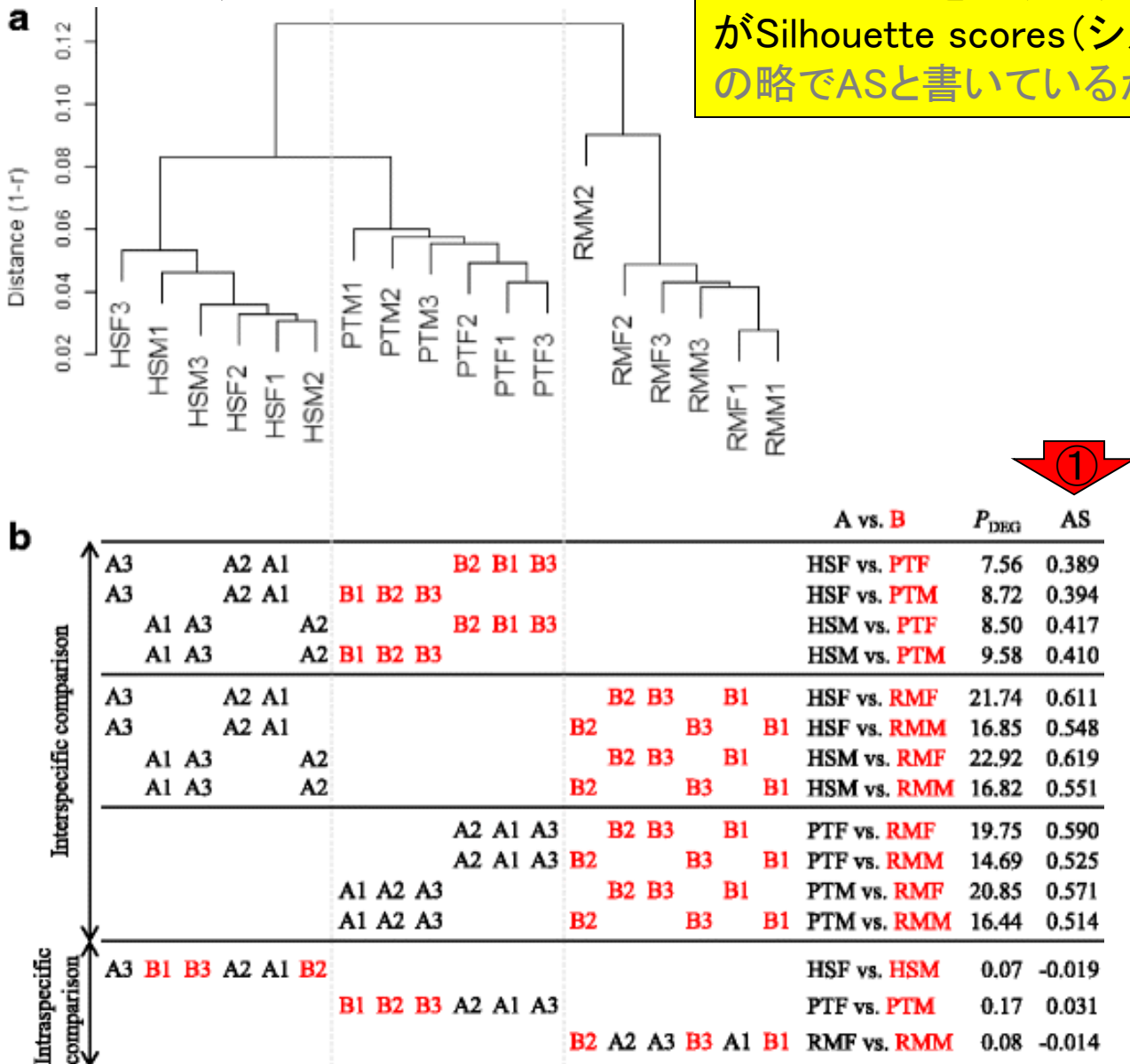
Related information

References for this PMC Article

Free in PMC

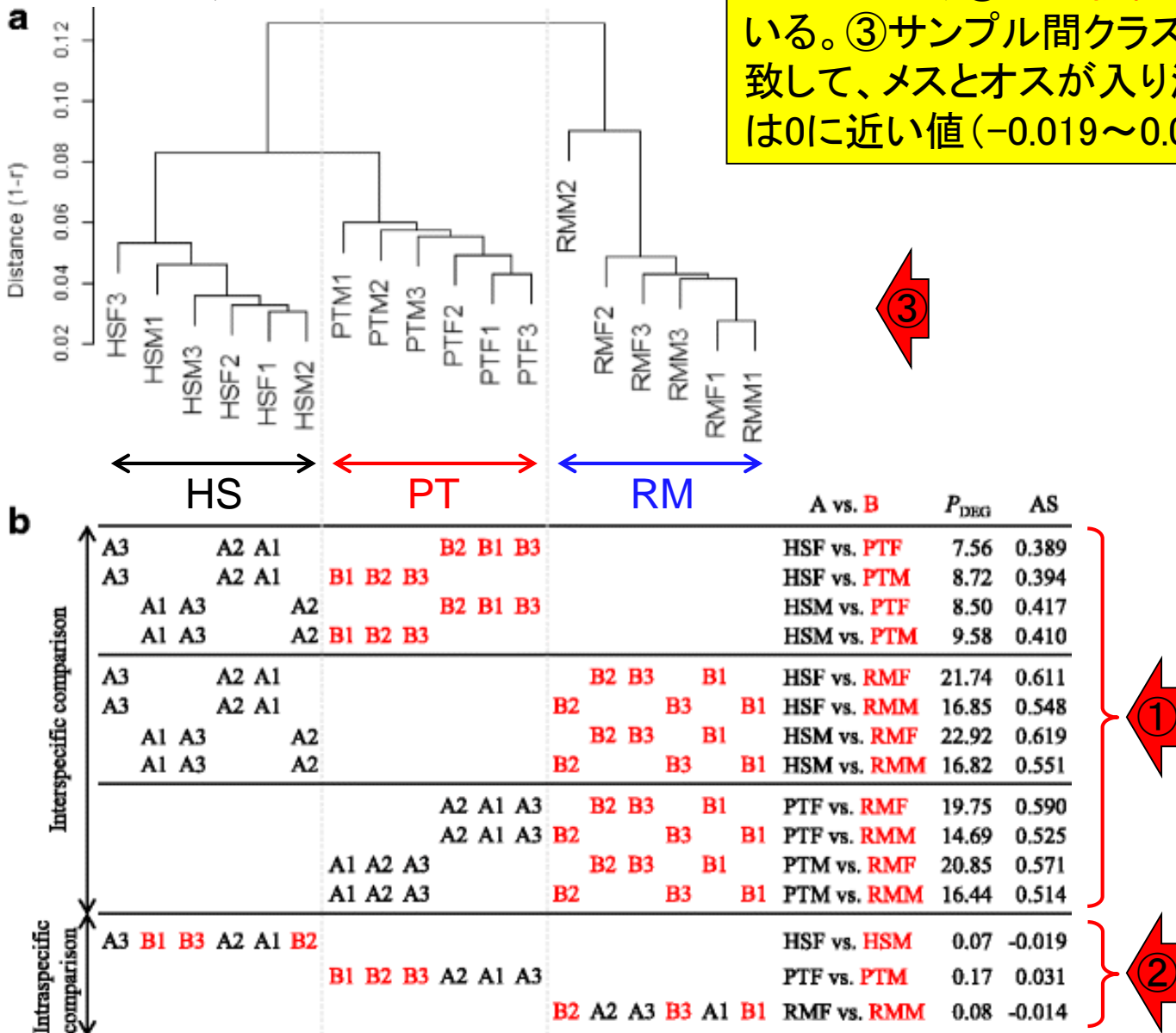
客観的な評価

興味あるグループ間(グループA vs. **グループB**)が区別できないほど似た発現パターンなら0に近い値、明瞭に区別できるなら1に近い値を返すような指標があればあるとうれしい。①これがSilhouette scores(シルエットスコア)。Average silhouettesの略でASと書いているが細かいところは気にしなくてもよい。



客観的な評価

①は生物種間の比較を行っているので、シルエットスコア (AS値)が0.389~0.619と比較的大きな値をとっている。それに対して、②は生物種内の比較(メス vs. オス)を行っている。③サンプル間クラスタリング結果の見た目と完全に一致して、メスとオスが入り混じっているのでシルエットスコアは0に近い値(-0.019~0.031)となっていることがわかる。



①では、②ヒト(HS)、チンパンジー(PT)、アカゲザル(RM)の3生物種間比較用のデータ(sample_blekhman_18.txt)を用いて説明しています。③がこのファイル中のサンプルの並びです。

Silhouettes

(Rで)塩基配列解析

(last modified 2018/05/30, since 2010)

このウェブページのR関連部分は、[インストール](#)についての推奨手順(Windows2018.0)なパッケージをインストール済みであるという前提で記述しています。初心者の方は[基本版](#)で自習して

What's new?

- 「マップ後」
- 「イントロ」
- 「H29年度」
- Silhouetteス

- ・ 解析 | 一般 | アラインメント | ペアワイズ | 基礎2 | [Biostrings](#) (last modified 2016/12/29)
- ・ 解析 | 一般 | アラインメント | ペアワイズ | 応用 | [Biostrings](#) (last modified 2016/12/29)
- ・ 解析 | 一般 | アラインメント | マルチプル | [DECIPHER\(Wright 2015\)](#) (last modified 2016/12/29)
- ・ 解析 | 一般 | アラインメント | マルチプル | [msa\(Bowyer 2015\)](#) (last modified 2016/12/29)
- ・ 解析 | 一般 | [Silhouette scores\(シルエットスコア\)](#) (last modified 2018/03/01)
- ・ 解析 | 一般 | [パターンマッチング](#) (last modified 2017/06/19)
- ・ 解析 | 一般 | [GC含量\(GC contents\)](#) (last modified 2015/09/12)
- ・ 解析 | 一般 | [Sequence logos\(Schneider 1990\)](#) (last modified 2014/07/23)
- ・ 解析 | 一般 | 上流配列解析 | [LDSS\(Yamamoto 2007\)](#) (last modified 2015/02/19)

解析 | 一般 | Silhouette scores(シルエットスコア)

Silhouetteスコアの新たな使い道提唱論文(Zhao et al., [Biol. Proc. Online, 2018](#))の利用法を説明します。入力に「解析 | 発現変動 | 2群間 | 対応なし | 複製あり | [TCC\(Sun 2013\)](#)」などと同じく、遺伝子発現行列データと比較したいグループラベル情報(Group1が1、Group2が2みたいなやつ)です。出力は、Average Silhouette(AS値)というスカラー情報(1つの数値)です。AS値の取り得る範囲は[-1, 1]で、数値が大きいほど指定したグループ間の類似度が低いことを意味し、発現変動解析結果としてDifferentially Expressed Genes (DEGs)が沢山得られる傾向にあります。逆に、AS値が低い(通常は-1に近い値になることはほぼ皆無で、相関係数と同じく0に近い)ほど指定したグループ間の類似度が高いことを意味し、DEGがほとんど得られない傾向にあります。論文中で提案している使い道としては、「発現変動解析を行ってDEGがほとんど得られなかった場合に、サンプル間クラスターリング(SC)結果とAS値を提示して、「客観的な数値情報である)AS値が0に近い値だったのでDEGがないのは妥当だね」みたいなdiscussionに使ってもらえればと思います。RNA-seqカウントデータでもマイクロアレイデータでも使えます。

例題の多くは、[サンプルデータ42](#)の20,689 genes×18 samplesのリアルカウントデータ([sample_blekhman_18.txt](#))を入力としています。ヒト(Homo sapiens; HS)のメス3サンプル(HSF1-3)とオス3サンプル(HSM1-3)、チンパンジー(Pan troglodytes; PT)のメス3サンプル(PTF1-3)とオス3サンプル(PTM1-3)、アカゲザル(Rhesus macaque; RM)のメス3サンプル(RMF1-3)とオス3サンプル(RMM1-3)の並びになっています。つまり、以下のような感じです。FはFemale(メス)、MはMale(オス)を表します。

ヒト(1-6列目): HSF1, HSF2, HSF3, HSM1, HSM2, and HSM3
 チンパンジー(7-12列目): PTF1, PTF2, PTF3, PTM1, PTM2, and PTM3
 アカゲザル(13-18列目): RMF1, RMF2, RMF3, RMM1, RMM2, and RMM3

「ファイル」-「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し以下をコピー。

例題1

①例題1のHSF vs. PTFの2群間比較。②シルエットスコアは0.389となっており、原著論文と同じ結果が得られていることがわかります。



1. HSF vs. PTFの場合:

HSF(ヒトメス)データが存在する1-3列目と、PTF(チンパンジーメス)データが存在する7-9列目のデータのみ抽出してAS値を算出しています。[Zhao et al., Biol. Proc. Online, 2018](#)のFig. 1bのHSF vs. PTFのAS値と同じ結果(AS = 0.389)が得られていることが分かります。尚、このZhao論文中では、先に18サンプルの全データを用いてフィルタリング(低発現遺伝子の除去とユニークパターンのみにする作業)を行ったのち、解析したい計6サンプルのサブセット抽出を行っているのでその手順に従っています。

```
in_f <- "sample_blekhman_18.txt" #入力ファイル名を指定してin_fに格納
param_subset <- c(1:3, 7:9) #取り扱いたいサブセット情報を指定
param_G1 <- 3 #G1群のサンプル数を指定
param_G2 <- 3 #G2群のサンプル数を指定
```

```
#必要なパッケージをロード
library(cluster) #パッケージをロード
```

```
#入力ファイルの読み込みとラベル情報の作成
data <- read.table(in_f, header=TRUE, row.names=1)
data.cl <- c(rep(1, param_G1), rep(2, param_G2))
dim(data) #オブジェクトの次元
```

```
#前処理(フィルタリング)
obj <- as.logical(rowSums(data) > 0) #条件を満たすオブジェクト
data <- unique(data[obj,]) #objがTRUEと$オブジェクト$
dim(data) #オブジェクト$
```

```
#前処理(サブセットの抽出)
data <- data[,param_subset] #param_subset$
```

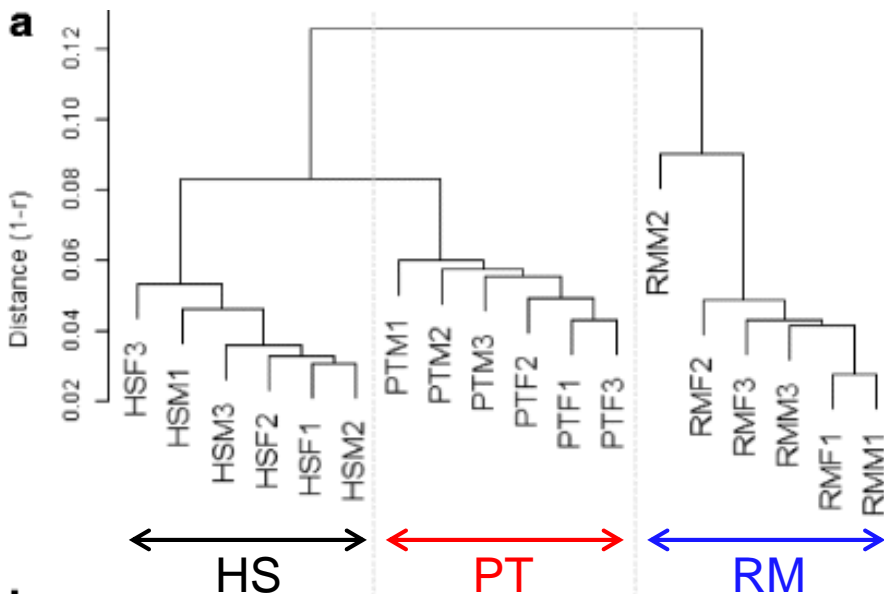
```
#本番(AS値の計算)
```

```
R Console
>
> #前処理(フィルタリング)
> obj <- as.logical(rowSums(data) > 0) #条件を満たす$
> data <- unique(data[obj,]) #objがTRUEと$
> dim(data) #オブジェクト$
[1] 16560 18
>
> #前処理(サブセットの抽出)
> data <- data[,param_subset] #param_subset$
>
> #本番(AS値の計算)
> d <- as.dist(1 - cor(data, method="spearman")) #サン$
> AS <- mean(silhouette(data.cl, d)[, "sil_width"])#$
> AS #AS値を表示
[1] 0.3893772
> |
```

②

①この結果のことです。ここでは2群間比較にフォーカスしていますが、シルエットスコアは原理的に3群間、4群間比較にも拡張して適用可能です。

HSF vs. PTF



			A vs. B	P_{DEG}	AS	
Interspecific comparison	A3	A2 A1	B2 B1 B3	HSF vs. PTF	7.56	0.389
	A3	A2 A1	B1 B2 B3	HSF vs. PTM	8.72	0.394
	A1 A3	A2	B2 B1 B3	HSM vs. PTF	8.50	0.417
	A1 A3	A2	B1 B2 B3	HSM vs. PTM	9.58	0.410
	A3	A2 A1	B2 B3 B1	HSF vs. RMF	21.74	0.611
	A3	A2 A1	B2 B3 B1	HSF vs. RMM	16.85	0.548
	A1 A3	A2	B2 B3 B1	HSM vs. RMF	22.92	0.619
	A1 A3	A2	B2 B3 B1	HSM vs. RMM	16.82	0.551
		A2 A1 A3	B2 B3 B1	PTF vs. RMF	19.75	0.590
		A2 A1 A3	B2 B3 B1	PTF vs. RMM	14.69	0.525
		A1 A2 A3	B2 B3 B1	PTM vs. RMF	20.85	0.571
		A1 A2 A3	B2 B3 B1	PTM vs. RMM	16.44	0.514
Intraspecific comparison	A3	B1 B3 A2 A1 B2		HSF vs. HSM	0.07	-0.019
		B1 B2 B3 A2 A1 A3		PTF vs. PTM	0.17	0.031
		B2 A2 A3 B3 A1 B1		RMF vs. RMM	0.08	-0.014



例題3

3. HS vs PT vs. RMの場合:

HS(ヒト)6 samples, PT(チンパンジー)6 samples, RM(アカゲザル)6 samplesの3生物種間のAS値を算出しています。全サンプルのデータを使っているため、サブセットの抽出は行っていません。このデータのサンプル間クラスタリング結果([Zhao et al., Biol. Proc. Online, 2018](#)のFig. 1a)でも3生物種明瞭に分離されていますが、高いAS値(= 0.4422661)が得られていることが分かります。

```
in_f <- "sample_blekman_18.txt" #入力ファイル名を指定してin_fに格納
param_G1 <- 6 #G1群のサンプル数を指定
param_G2 <- 6 #G2群のサンプル数を指定
param_G3 <- 6 #G3群のサンプル数を指定
```

```
#必要なパッケージをロード
library(cluster) #パッケージをロード
```

```
#入力ファイルの読み込みとラベル情報の作成
data <- read.table(in_f, header=TRUE, row.names=1)
data.cl <- c(rep(1, param_G1), rep(2, param_G2), rep(3, param_G3))
dim(data) #オブジェクト$
```

```
#前処理(フィルタリング)
obj <- as.logical(rowSums(data) > 0) #条件を満たす$
data <- unique(data[obj,]) #objがTRUEと$
dim(data) #オブジェクト$
```

```
#本番(AS値の計算)
d <- as.dist(1 - cor(data, method="spearman")) #サン$
AS <- mean(silhouette(data.cl, d)[, "sil_width"]) #AS値を表示
AS
```

```
R Console
> data.cl <- c(rep(1, param_G1), rep(2, param_G2), rep(3, param_G3))
> dim(data) #オブジェクト$
[1] 20689 18
>
> #前処理(フィルタリング)
> obj <- as.logical(rowSums(data) > 0) #条件を満たす$
> data <- unique(data[obj,]) #objがTRUEと$
> dim(data) #オブジェクト$
[1] 16560 18
>
> #本番(AS値の計算)
> d <- as.dist(1 - cor(data, method="spearman")) #サン$
> AS <- mean(silhouette(data.cl, d)[, "sil_width"]) #AS値を表示
> AS
[1] 0.4422661
> |
```



例題5

①例題5の1人目 vs. 2人目 vs. 3人目の3群間比較。②生物種やメスオスの区別は全くしていないので、0に近いスコアが得られていますね。

5. 1人目 vs 2人目 vs. 3人目の場合: ①

3群間比較ですが、生物種やメスオスに関係なく、1人目をG1群、2人目をG2群、3人目をG3群としたdata.clを作成しています。予想通り0に近いAS値(= 0.4422661)が得られます。

```
in_f <- "sample_blekhan_18.txt" #入力ファイル名を指定してin_fに格納
data.cl <- c(1,2,3,1,2,3,1,2,3,1,2,3,1,2,3,1,2,3,1,2,3) #G1群を1、G2群を2、G3群を3としたベクトルdata.clを作成
```

```
#必要なパッケージをロード
library(cluster) #パッケージの読み込み
```

```
#入力ファイルの読み込みとラベル情報の作成
data <- read.table(in_f, header=TRUE, row.names=1)
dim(data)
```

```
#前処理(フィルタリング)
obj <- as.logical(rowSums(data) > 0) #条件を満たすオブジェクト
data <- unique(data[obj,]) #objがTRUEとされたオブジェクト
dim(data)
```

```
#本番(AS値の計算)
d <- as.dist(1 - cor(data, method="spearman")) #サン$
AS <- mean(silhouette(data.cl, d)[, "sil_width"]) #AS値を表示
AS
```

```
R Console
> data <- read.table(in_f, header=TRUE, row.names=1, sep="$")
> dim(data)
[1] 20689 18
> #前処理(フィルタリング)
> obj <- as.logical(rowSums(data) > 0) #条件を満たす$
> data <- unique(data[obj,]) #objがTRUEとされた$
> dim(data)
[1] 16560 18
> #本番(AS値の計算)
> d <- as.dist(1 - cor(data, method="spearman")) #サン$
> AS <- mean(silhouette(data.cl, d)[, "sil_width"]) #AS値を表示
> AS
[1] -0.10115
> |
```

