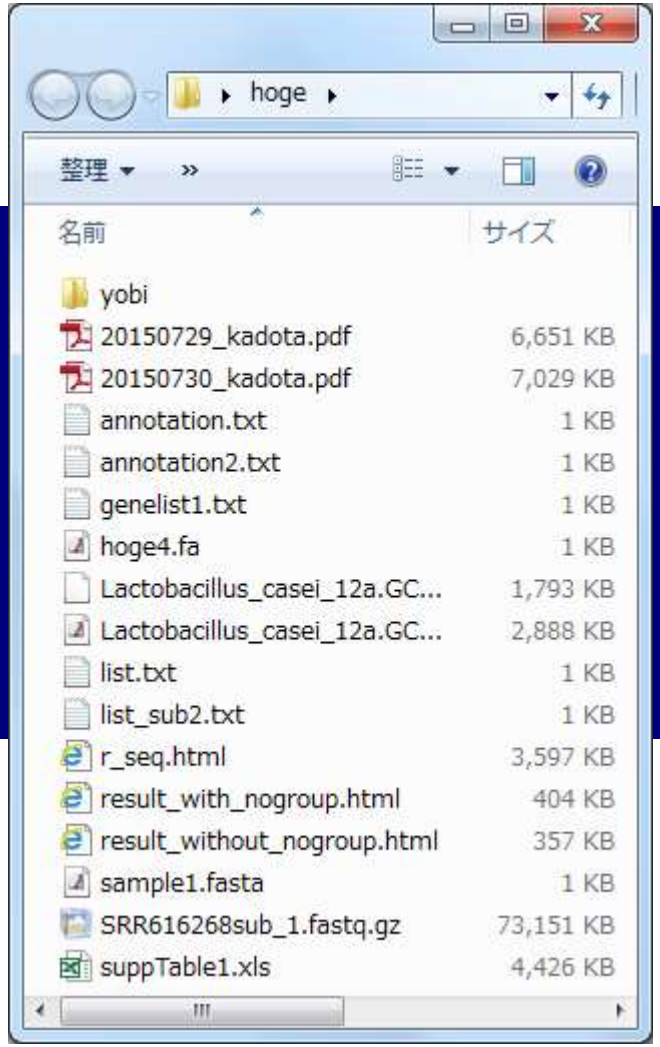
2015.07.31版

配布するUSBメモリ中のhogeフォルダを
デスクトップにコピーしておいてください。



NGSハンズオン講習会
Bioconductorの利用法

東京大学・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
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/

Contents（全体）

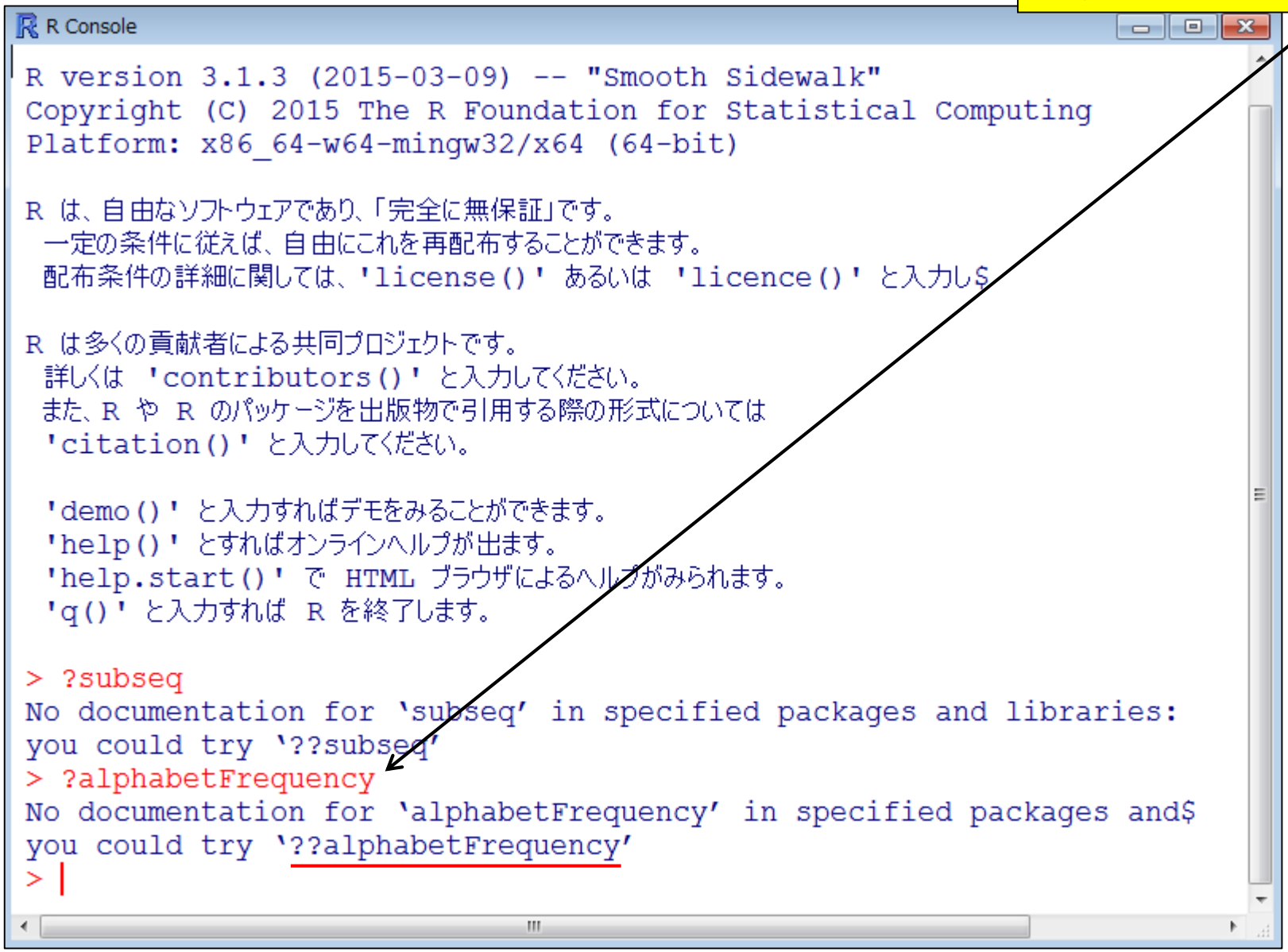
- 7月22日（水）：84→83名。Bio-Linux 8とRのインストール状況確認。基本自習（門田・寺田先生）
- 7月23日（木）：92→90名。Linux基礎。Linuxコマンドなど UNIXの基礎の理解（門田）
- 7月24日（金）：85→83名。スクリプト言語。シェルスクリプト（アメリエフ株式会社 服部恵美先生）
- 7月27日（月）：93→91名。スクリプト言語。Perl（アメリエフ 服部先生）
- 7月28日（火）：91→90名。スクリプト言語。Python（アメリエフ 服部先生）
- 7月29日（水）：94→88名。データ解析環境R（門田）
  - R基礎1(初級)：R言語の基礎（インストールから利用まで）
  - R基礎2(初級)：ファイルの読み込み、行列演算の基本
  - R各種パッケージ(中級)：パッケージのインストール法と代表的なパッケージの利用法
- 7月30日（木）：96→91名。データ解析環境R（門田）
  - Bioconductorの利用法1(中級)：データの型やバージョンの違い
  - Bioconductorの利用法2(中級)：FASTA/FASTQファイルの各種解析
- 8月3日（月）：89→84名。NGS解析。基礎（アメリエフ 山口昌雄先生）
- 8月4日（火）：85→80名。NGS解析。ゲノムReseq、変異解析（アメリエフ 山口先生）
- 8月5日（水）：86 →81名。NGS解析。RNA-seq、統計解析（前半：山口先生、後半：門田）
- 8月6日（木）：104 →98名。NGS解析。ChIP-seq（理研 森岡勝樹先生）

Contents

- パッケージ
  - CRANとBioconductor
  - 推奨パッケージインストール手順のおさらい
  - Bioconductor概観 → ゲノム配列パッケージ(BSgenome)
  - ヒトゲノム情報パッケージの解析
- プロモーター配列取得(sessionInfo、バージョンの違い)
  - Rパッケージ(ゲノムとアノテーションパッケージの併用)
  - FASTA形式ファイルとGFF3形式ファイル
  - データの型
- FASTQファイルの各種解析(LinuxとRを相補的に活用)
  - Linux (FastQC)とR (ShortRead)で同じ:Overrepresented sequences項目
  - Linux (FastQC)とR (QuasR)で異なる見栄え:Per base sequence content項目。FastQCに--nogroupというオプションがあることを知る。
  - FastQCのオプション (デフォルトと--nogroupあり)の違いによるKmer Content項目の結果の違い → Rで検証

パッケージ

R起動直後に「**?関数名**」と打ち込んでも、使用法を記したウェブページが開かずにエラーが出ることがあります。



パッケージ

①「??alphabetFrequency」と打ち込むように勧められているので打ってみる。検索結果のウェブページが表示されるので、②それっぽい関数名のところをクリック。

R Console

```
> ?subseq
No documentation for 'subseq' in specified packages and libraries:
you could try '??subseq'
> ?alphabetFrequency
No documentation for 'alphabetFrequency' in specified packages and$
you could try '??alphabetFrequency'
> ??alphabetFrequency
starting httpd help server ... done
>
```

①

Search Results

The search string was **"alphabetFrequency"**

Help pages:

Biostrings::class:MultipleAlignment	MultipleAlignment objects
Biostrings::letterFrequency	Calculate the frequency of letters in a biological sequence, or the consensus matrix of a set of sequences
GenomicAlignments::stackStringsFromBam	Stack the read sequences stored in a BAM file on a region of interest
ShortRead::QualityScore-class	Quality scores for short reads and their alignments

②

参考

パッケージ

alphabetFrequency関数はBiostringsというパッケージから提供されているものだと読み解く。「??関数名」は、関数名は既知だがどのパッケージから提供されているものかを知りたい場合などに利用する。

letterFrequency {Biostrings}

Calculate the frequency of letters in a biological sequence, or the consensus matrix of a set of sequences

Description

Given a biological sequence (or a set of biological sequences), the `alphabetFrequency` function computes the frequency of each letter of the relevant alphabet.

`letterFrequency` is similar, but more compact if one is only interested in certain letters. It can also tabulate letters "in common".

`letterFrequencyInSlidingView` is a more specialized version of `letterFrequency` for (non-masked) XString objects. It tallys the requested letter frequencies for a fixed-width view, or window, that is conceptually slid along the entire input sequence.

The `consensusMatrix` function computes the consensus matrix of a set of sequences, and the `consensusString` function creates the consensus sequence from the consensus matrix based upon specified criteria.

In this man page we call "DNA input" (or "RNA input") an XString, XStringSet, XStringViews or MaskedXString object of base type DNA (or RNA).

Usage

```
alphabetFrequency(x, as.prob=FALSE, ...)
hasOnlyBaseLetters(x)
uniqueLetters(x)
```

パッケージ

multi-FASTAファイルを読み込んで様々な解析ができるのは、①Biostringsや②seqinrなどの塩基配列解析用パッケージのおかげです。

- イントロ | 一般 | 任意の位置の塩基を置換 (last modified 2013/09/12)
- イントロ | 一般 | 指定した範囲の配列を取得 (last modified 2015/04/06) NEW
- イントロ | 一般 | 指定したID(染色体やdescription)の配列を取得 (last modified 2014/03/10)
- イントロ | 一般 | 翻訳配列(translate)を取得(基礎) | Biostrings (last modified 2015/03/09)
- イントロ | 一般 | 翻訳配列(translate)を取得(応用) | seqinr(Charif_2005) (last modified 2015/03/09)
- イントロ | 一般 | 相補鎖(complement)を取得 (last modified 2013/06/14)
- イントロ | 一般 | 逆相補鎖(reverse complement)を取得 (last modified 2013/06/14)
- イントロ | 一般 | 逆
- イントロ | 一般 | 2
- イントロ | 一般 | 3
- イントロ | 一般 | 任

①

②

イントロ | 一般 | 翻訳配列(translate)を取得(基礎) | Biostrings

Biostringsパッケージを用いて塩基配列を読み込んでアミノ酸配列に翻訳するやり方を示します。翻訳のための遺伝コード(genetic code)は、Standard Genetic Codeだそうです。もちろん生物種?!によって多少違い(variants)があるようで、"Standard", "SGC0", "Vertebrate Mitochondrial", "SGC1"などいろいろ選べるようです。
「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ペ。

1. FASTA形式ファイル(sample1.fasta)の場合:

multi-FASTAではないsingle-FAST

```
in_f <- "sample1.fasta"
out_f <- "hoge1.fasta"

#必要なパッケージをロード
library(Biostrings)

#入力ファイルの読み込み
fasta <- readDNAStringSet
```

イントロ | 一般 | 翻訳配列(translate)を取得(応用) | seqinr(Charif_2005)

seqinrパッケージを用いて塩基配列を読み込んでアミノ酸配列に翻訳するやり方を示します。本気で翻訳配列を取得する場合にはこちらの利用をお勧めします。翻訳できないコドンはアミノ酸X(不明なアミノ酸)に変換してくれたり、translate関数のオプションとしてambiguous=Tとすると、翻訳できるものは可能な限り翻訳してくれます(高橋 広夫 氏提供情報)。lapply関数を用いるやり方(高橋 広夫 氏提供情報)とsapply関数を用いるやり方(甲斐 政親 氏提供情報)を示します。
「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ペ。

1. FASTA形式ファイル(sample1.fasta)の場合:

multi-FASTAではないsingle-FASTA形式ファイルです。

```
in_f <- "sample1.fasta"                    #入力ファイル名を指定してin_fに格納
out_f <- "hoge1.fasta"                     #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(seqinr)                            #パッケージの読み込み

#入力ファイルの読み込み
hoge <- read.fasta(in_f, seqtype="DNA")#in_fで指定したファイルの読み込み
hoge                                       #確認してるだけです
```

パッケージ

Biostringsというパッケージをlibrary関数を用いて読み込むことによって、alphabetFrequencyのようなBiostringsが提供する関数群を利用できるのです。ここでは、意図的に「library(Biostrings)」を2回実行して、2回目は何も表示されないということを思い出させています。実際には1回のみで大丈夫です。「?alp」まで打ってからTabキーを押すなどして「タブ補完」テクを有効利用。

R Console

```
> library(Biostrings)
 要求されたパッケージ BiocGenerics をロード中です
 要求されたパッケージ parallel をロード中です

 次のパッケージを付け加えます: 'BiocGenerics'

The following objects are masked from 'package:parallel':

    clusterApply, clusterApplyLB, clusterCall, cluste
    clusterExport, clusterMap, parApply, parCapply,
    parLapply, parLapplyLB, parRapply, parSapply, par

The following object is masked from 'package:stats':

    xtabs

The following objects are masked from 'package:base':

    anyDuplicated, append, as.data.frame, as.vector,
    colnames, do.call, duplicated, eval, evalq, Filte
    get, intersect, is.unsorted, lapply, Map, mapply,
    mget, order, paste, pmax, pmax.int, pmin, pmin.in
    Position, rank, rbind, Reduce, rep.int, rownames,
    setdiff, sort, table, tapply, union, unique, unli
    unsplit

 要求されたパッケージ S4Vectors をロード中です
 要求されたパッケージ stats4 をロード中です
 要求されたパッケージ IRanges をロード中です
 要求されたパッケージ XVector をロード中です
> library(Biostrings)
> ?alphabetFrequency
```

letterFrequency {Biostrings} R Documentation

Calculate the frequency of letters in a biological sequence, or the consensus matrix of a set of sequences

Description

Given a biological sequence (or a set of biological sequences), the `alphabetFrequency` function computes the frequency of each letter of the relevant alphabet.

`letterFrequency` is similar, but more compact if one is only interested in certain letters. It can also tabulate letters "in common".

`letterFrequencyInSlidingView` is a more specialized version of `letterFrequency` for (non-masked) XString objects. It tallys the requested letter frequencies for a fixed-width view, or window, that is conceptually slid along the entire input sequence.

The `consensusMatrix` function computes the consensus matrix of a set of sequences, and the `consensusString` function creates the consensus sequence from the consensus matrix based upon specified criteria.

In this man page we call "DNA input" (or "RNA input") an XString, XStringSet, XStringViews or MaskedXString object of base type DNA (or RNA).

Usage

```
alphabetFrequency(x, as.prob=FALSE, ...)
hasOnlyBaseLetters(x)
uniqueLetters(x)
```

R本体とパッケージの関係

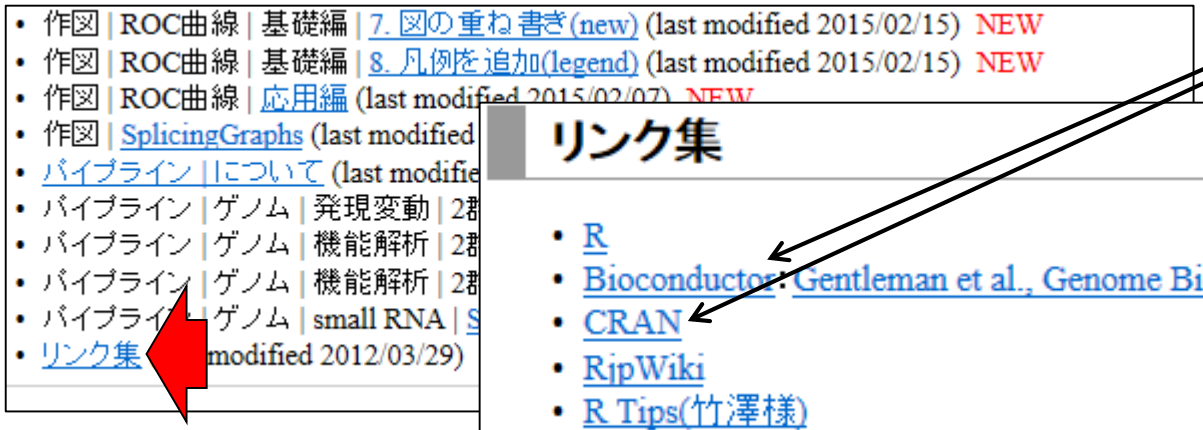
「R本体」と「パッケージ」の関係は、「パソコン」と「ソフト」、「Microsoft EXCEL」と「アドイン」、「Cytoscape」と「プラグイン」のようなものという理解でよい。

- パソコンを購入しただけの状態では、できることが限られています。
  - 通常は、Officeやウイルス撃退ソフトなどをインストールして利用します。
- Linuxをインストールしただけの状態では、できることが限られています。
  - 通常は、マッピングなど各種プログラムをインストールして利用します。
- R本体をインストールしただけの状態では、できることが限られています。
  - NGS解析を行う各種パッケージ（またはライブラリ）をインストールして利用します。

CRANとBioconductor

CRAN提供パッケージは生命科学を含む様々な分野で利用される。NGS解析は、主にBioconductor提供パッケージを利用。

- R上で利用可能なパッケージの2大リポジトリ(貯蔵庫)
  - CRAN (The Comprehensive R Archive Network):6,878パッケージ
  - Bioconductor:1,024パッケージ



定期的にバージョンアップ

バグの修正や新たな機能がどんどん追加されている。最新版の利用をお勧め。毎年5月と11月ごろにバージョンアップするとよいだろう。

- 近年のリリース頻度
  - R本体 (http://www.r-project.org/)
    - 2015-06-18にver. 3.2.1をリリース
    - 2015-04-16にver. 3.2.0をリリース
    - 2015-03-09にver. 3.1.3をリリース
    - 2014-10-31にver. 3.1.2をリリース
    - …
    - 2012-03-30にver. 2.15.0をリリース
    - …
  - Bioconductor (http://bioconductor.org/)は半年ごとにリリース
    - 2015-04にver. 3.1をリリース (R ver. 3.2.1で動作確認)、提供パッケージ数:1,024
    - 2014-10にver. 3.0をリリース (R ver. 3.1.1で動作確認)、提供パッケージ数:934
    - 2014-04にver. 2.14をリリース (R ver. 3.1.0で動作確認)、提供パッケージ数:824
    - 2013-10にver. 2.13をリリース (R ver. 3.0で動作確認)、提供パッケージ数:750
    - 2013-04にver. 2.12をリリース (R ver. 3.0で動作確認)、提供パッケージ数:672
    - 2012-10にver. 2.11をリリース (R ver. 2.15.1で動作確認)、提供パッケージ数:608
    - 2012-04にver. 2.10をリリース (R ver. 2.15.0で動作確認)、提供パッケージ数:553
    - …

Bioconductor

Bioconductorに関する総説(Review)。ゲノム配列やアノテーションパッケージもBioconductorから提供されており、それらに関する言及もあり。

Nat Methods. 2015 Feb;12(2):115-21. doi: 10.1038/nmeth.3252.

Orchestrating high-throughput genomic analysis with Bioconductor.

Huber W[1], Carey VJ[2], Gentleman R[3], Anders S[1], Carlson M[4], Carvalho BS[5], Bravo HC[6], Davis S[7], Gatto L[8], Girke T[9], Gottardo R[10], Hahne F[11], Hansen KD[12], Irizarry RA[13], Lawrence M[3], Love MI[13], MacDonald J[14], Obenchain V[4], Oleś AK[1], Pagès H[4], Reyes A[1], Shannon P[4], Smyth GK[15], Tenenbaum D[4], Waldron L[16], Morgan M[4].

Author information

Abstract

Bioconductor is an open-source, open-development software project for the analysis and comprehension of high-throughput data in genomics and molecular biology. The project aims to enable interdisciplinary research, collaboration and rapid development of scientific software. Based on the statistical programming language R, Bioconductor comprises 934 interoperable packages contributed by a large, diverse community of scientists. Packages cover a range of bioinformatic and statistical applications. They undergo formal initial review and continuous automated testing. We present an overview for prospective users and contributors.

パッケージのインストール

「必要最小限プラスアルファ」の推奨インストール手順を行えば、「(Rで)塩基配列解析」で利用する多くのパッケージがインストールされます。

- はじめに (last modified 2015/03/31)
- 参考資料 (講義、講習会、本など) (last modified 2015/07/07) NEW
- 過去のお知らせ (last modified 2015/07/08) NEW
- インストール | について (last modified 2015/04/04)
- インストール | R本体 | 最新版 | Win用 (last modified 2015/03/22)推奨
- インストール | R本体 | 最新版 | Mac用 (last modified 2015/04/22)推奨
- インストール | R本体 | 過去版 | Win用 (last modified 2015/03/22)
- インストール | R本体 | 過去版 | Mac用 (last modified 2015/03/22)
- インストール | Rパッケージ | ほぼ全て(20GB以上?!) (last modified 2015/05/25)
- インストール | Rパッケージ | 必要最小限プラスアルファ(数GB?!) (last modified 2015/06/02)推奨
- インストール | Rパッケージ | 必要最小限プラスアルファ(アグリバイオ居室のみ) (last modified 2015/06/16)
- インストール | Rパッケージ | 必要最小限(数GB?!) (last modified 2015/05/25)
- インストール | Rパッケージ | 個別 (last modified 2015/06/10)
- (削除予定)Rのインストールと起動 (last modified 2015/04/02)
- (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 (last modified 2015/06/15)
- バイオインフォマティクス人材育成カリキュラム(次世代シークエンサ) | NGSハンズオン講習会2015 (last mo
- バイオインフォマティクス人材育成カリキュラム(次世代シークエンサ) | 速習コース2014 (last modified 2015/
- 書籍 | トランスクリプトーム解析 | について (last modified 2014/05/12)
- 書籍 | トランスクリプトーム解析 | 2.3.1 RNA-seqデータ(FASTQファイル) (last modified 2014/04/15)
- 書籍 | トランスクリプトーム解析 | 2.3.2 リファレンス配列 (last modified 2014/04/16)

パッケージのインストール

これらはCRANから提供されているものたち

インストール | Rパッケージ | 必要最小限プラスアルファ(数GB?!) NEW

(Rで)塩基配列解析、(Rで)マイクロアレイデータ解析中で利用するパッケージ、プラスアルファのパッケージをインストールするやり方です。Rパッケージの2大リポジトリであるCRANとBioconductorから提供されているパッケージ群のうち、一部のインストールに相当しますので、相当短時間でインストールが完了します。

1. R本体を起動

2. CRANから提供されているパッケージ群のインストール

以下を「Rコンソール画面上」でコピー&ペースト。どこからダウンロードするか？と聞かれるので、その場合は自分のいる場所から近いサイトを指定しましょう。

```
#(Rで)塩基配列解析で主に利用
install.packages("limma")                #(Rで)マイクロアレイデータ解析でも利用
install.packages("samr")                 #(Rで)マイクロアレイデータ解析でも利用
install.packages("seqinr")               #(Rで)マイクロアレイデータ解析でも利用

#(Rで)マイクロアレイデータ解析で利用
install.packages("cclust")
install.packages("class")
install.packages("e1071")
install.packages("GeneCycle")
install.packages("gptk")
install.packages("GSA")
install.packages("mixOmics")
install.packages("pvclust")
install.packages("RobLoxBioC")
install.packages("som")
install.packages("st")
install.packages("varSelRF")

#アグリバイオの他の講義科目で利用予定
install.packages("ape")
install.packages("cluster")
install.packages("fields")
```

パッケージのインストール

```
biocLite("DEGseq", suppressUpdates=TRUE)
biocLite("DESeq", suppressUpdates=TRUE)
biocLite("DESeq2", suppressUpdates=TRUE)
biocLite("DiffBind", suppressUpdates=TRUE)
biocLite("doMC", suppressUpdates=TRUE)
biocLite("EBSeq", suppressUpdates=TRUE)
biocLite("EDASeq", suppressUpdates=TRUE)
biocLite("edgeR", suppressUpdates=TRUE)
biocLite("GenomicAlignments", suppressUpdates=TRUE)
```

①

4. Bioconductorから提供されているパッケージ群のインストール

ゲノム配列パッケージです。一つ一つの容量が尋常でないため、必要に応じてテキストエディタなどに予めコピペしておき、いらないゲノムパッケージを削除してからお使いください。

```
source("http://bioconductor.org/biocLite.R")#おまじない
biocLite("BSgenome.Athaliana.TAIR.TAIR9", suppressUpdates=TRUE)#シロイヌナズナゲノム
biocLite("BSgenome.Celegans.UCSC.ce6", suppressUpdates=TRUE)#線虫ゲノム
biocLite("BSgenome.Drerio.UCSC.danRer7", suppressUpdates=TRUE)#ゼブラフィッシュゲノム
biocLite("BSgenome.Hsapiens.NCBI.GRCh38", suppressUpdates=TRUE)#ヒトゲノム(GRCh38)
biocLite("BSgenome.Hsapiens.UCSC.hg19", suppressUpdates=TRUE)#ヒトゲノム(hg19)
biocLite("BSgenome.Mmusculus.UCSC.mm10", suppressUpdates=TRUE)#マウスゲノム(mm10)
```

②

①ゲノム情報のパッケージ群(BSgenome…)はBioconductorから提供されています。ここでは計6パッケージをインストールしています。②例えば赤線部分は、マウスのmm10というバージョンのゲノム配列情報を含むパッケージの名前(BSgenome.Mmusculus.UCSC.mm10)に相当します。biocLiteという関数を用いて該当パッケージをインストールしています。

Contents

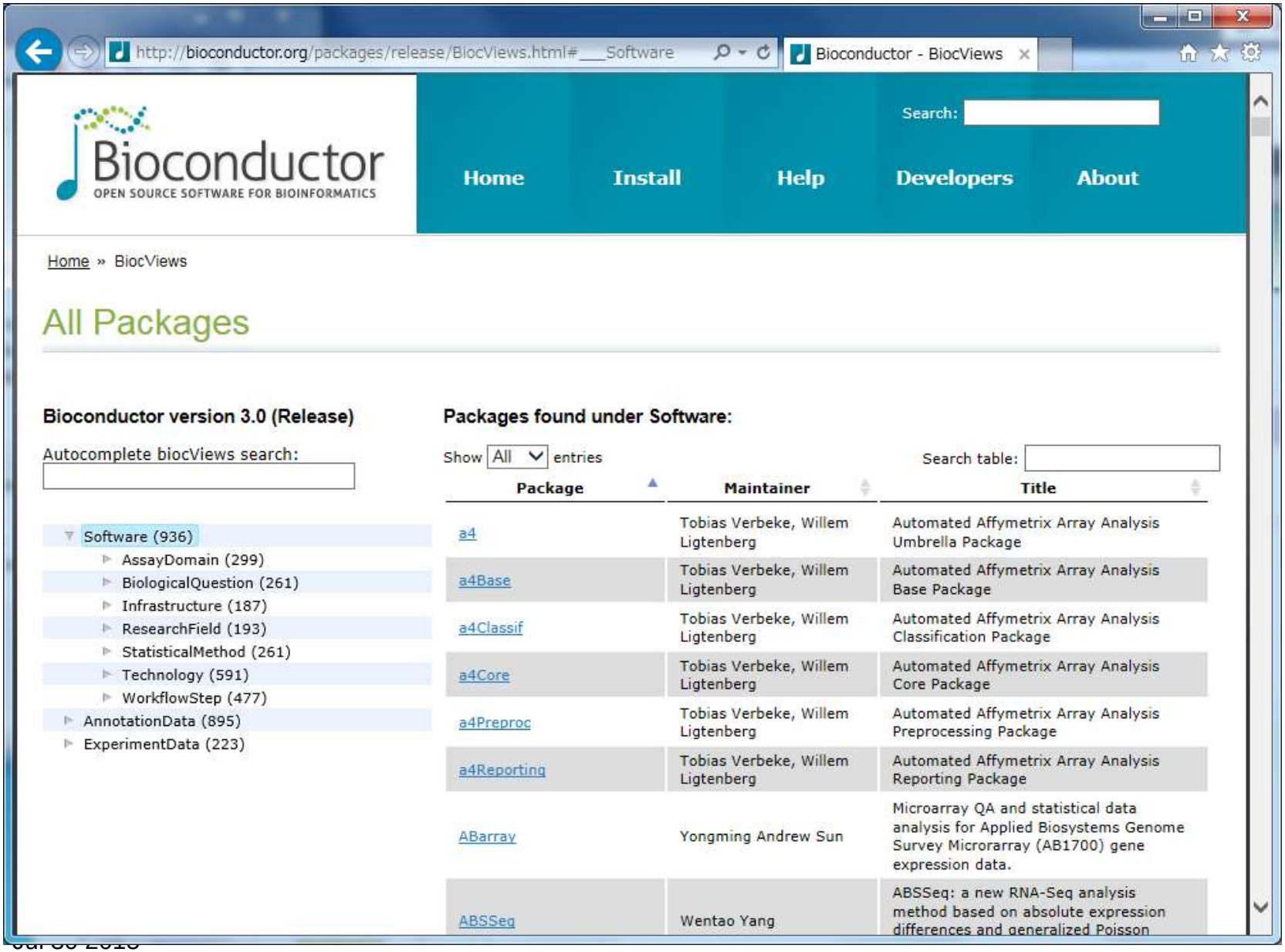
- パッケージ
  - CRANとBioconductor
  - 推奨パッケージインストール手順のおさらい
  - Bioconductor概観 → ゲノム配列パッケージ(BSgenome)
  - ヒトゲノム情報パッケージの解析
- プロモーター配列取得(sessionInfo、バージョンの違い)
  - Rパッケージ(ゲノムとアノテーションパッケージの併用)
  - FASTA形式ファイルとGFF3形式ファイル
  - データの型
- FASTQファイルの各種解析(LinuxとRを相補的に活用)
  - Linux (FastQC)とR (ShortRead)で同じ:Overrepresented sequences項目
  - Linux (FastQC)とR (QuasR)で異なる見栄え:Per base sequence content項目。FastQCに--nogroupというオプションがあることを知る。
  - FastQCのオプション (デフォルトと--nogroupあり)の違いによるKmer Content項目の結果の違い → Rで検証

Bioconductor概観

PubMed上で「R Bioconductor」でキーワード検索し原著論文があるパッケージのみ探すのも一つの戦略ですが、原著論文公開前のパッケージも見つかります。

Bioconductor概観

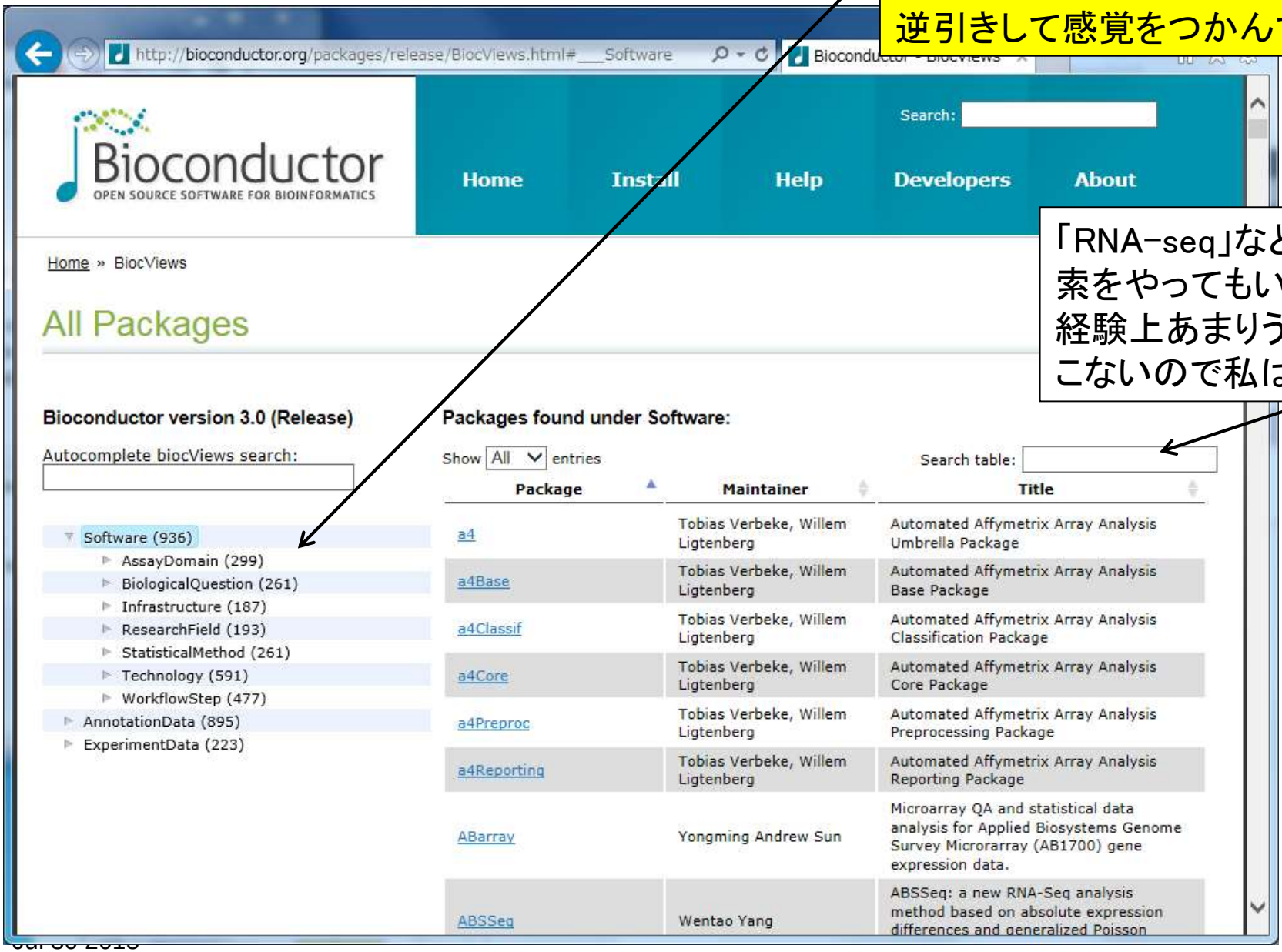
ネットワーク環境にもよりますが、数秒～数分以内にこのような画面になります。



Bioconductor概観

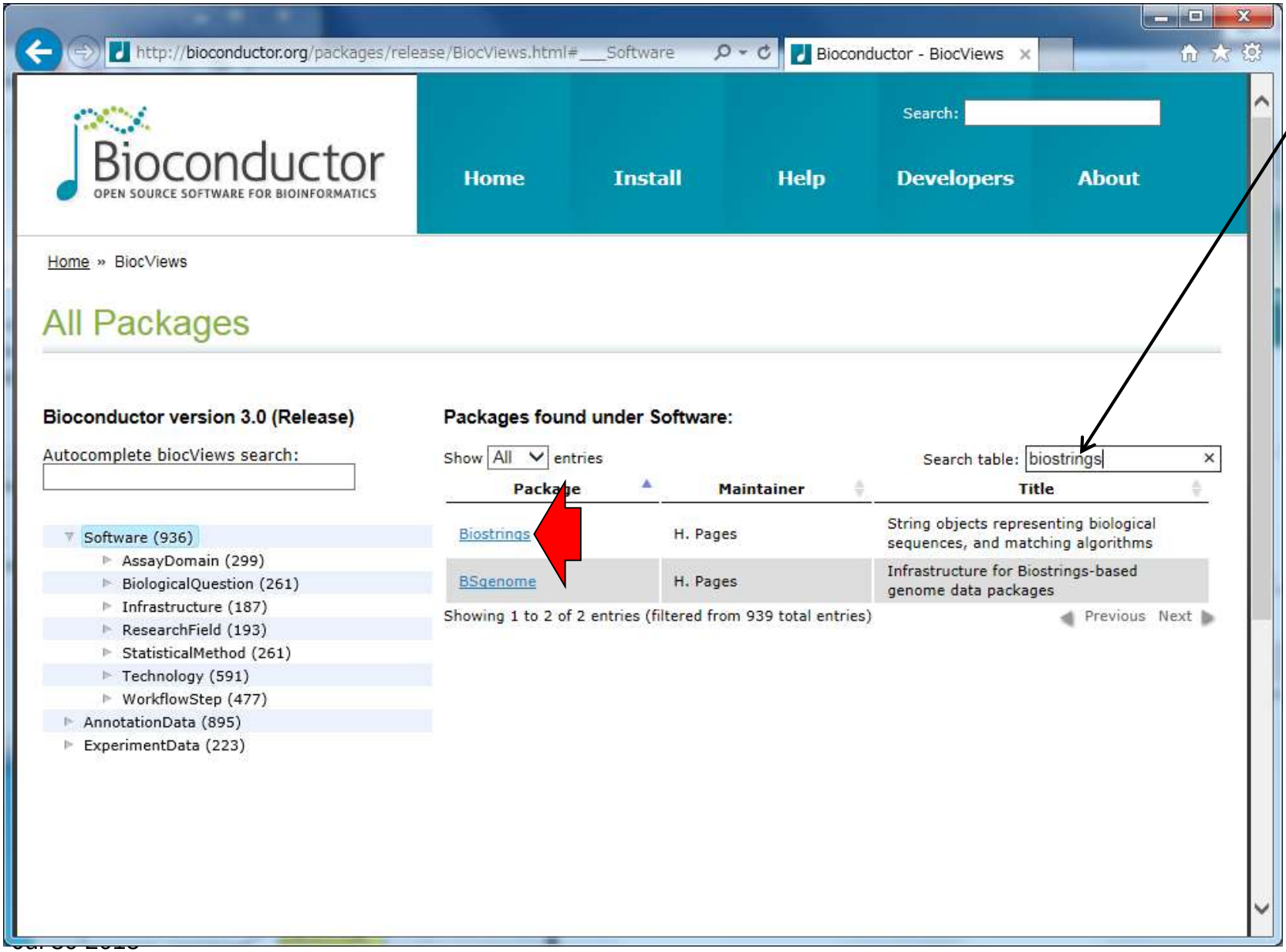
基本的には左側のカテゴリ分けのところを眺めますが、Biostringsなど何を行うパッケージかがある程度分かっているものから逆引きして感覚をつかんでおくとよいでしょう。

「RNA-seq」など、フリーワード検索をやってもいいとは思いますが、経験上あまりうまく引っかかってこないので私はやりません。



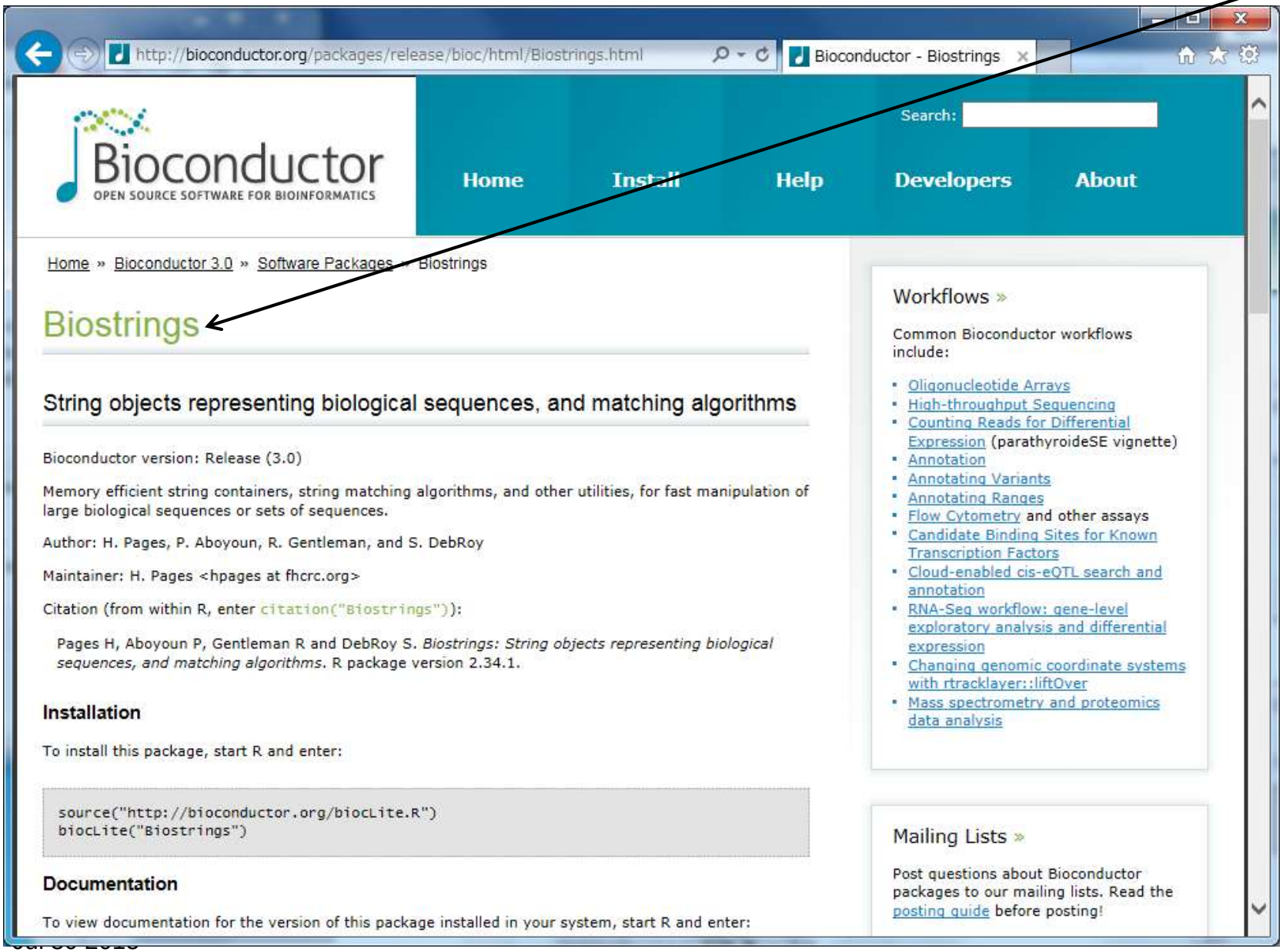
Bioconductor概観

「biostrings」と打つとすぐにリストアップされる。



Bioconductor概観

BioconductorのBiostringsパッケージのページに飛びます。

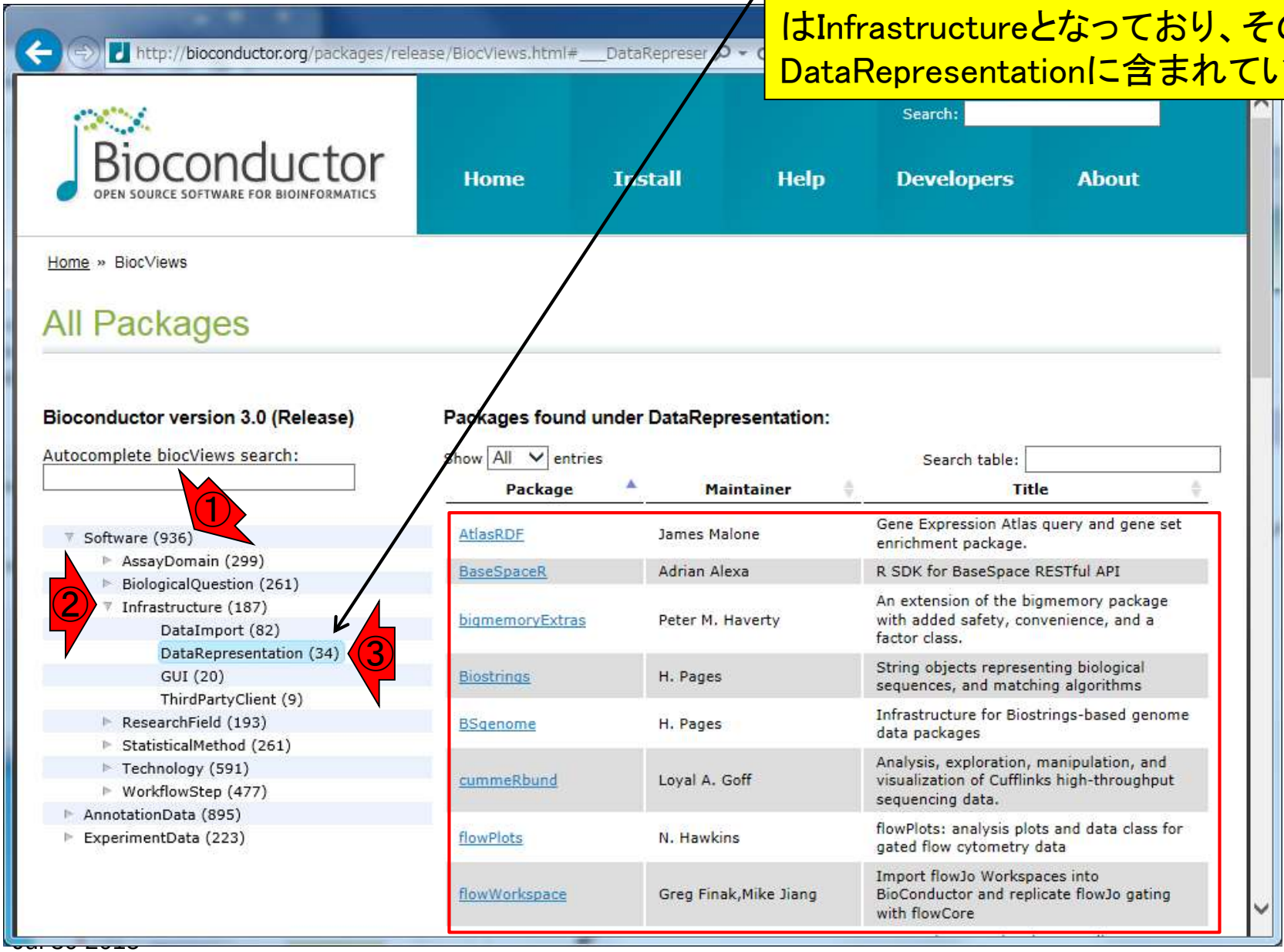


Bioconductor概観

BioconductorのBiostringsパッケージのページで、ちょっと下のほうに移動。biocViewsのところで見えるキーワードっぽいのがさきほどのカテゴリ分けに相当。例えば、DataRepresentationをクリックするとカテゴリ分けの階層関係がわかる。

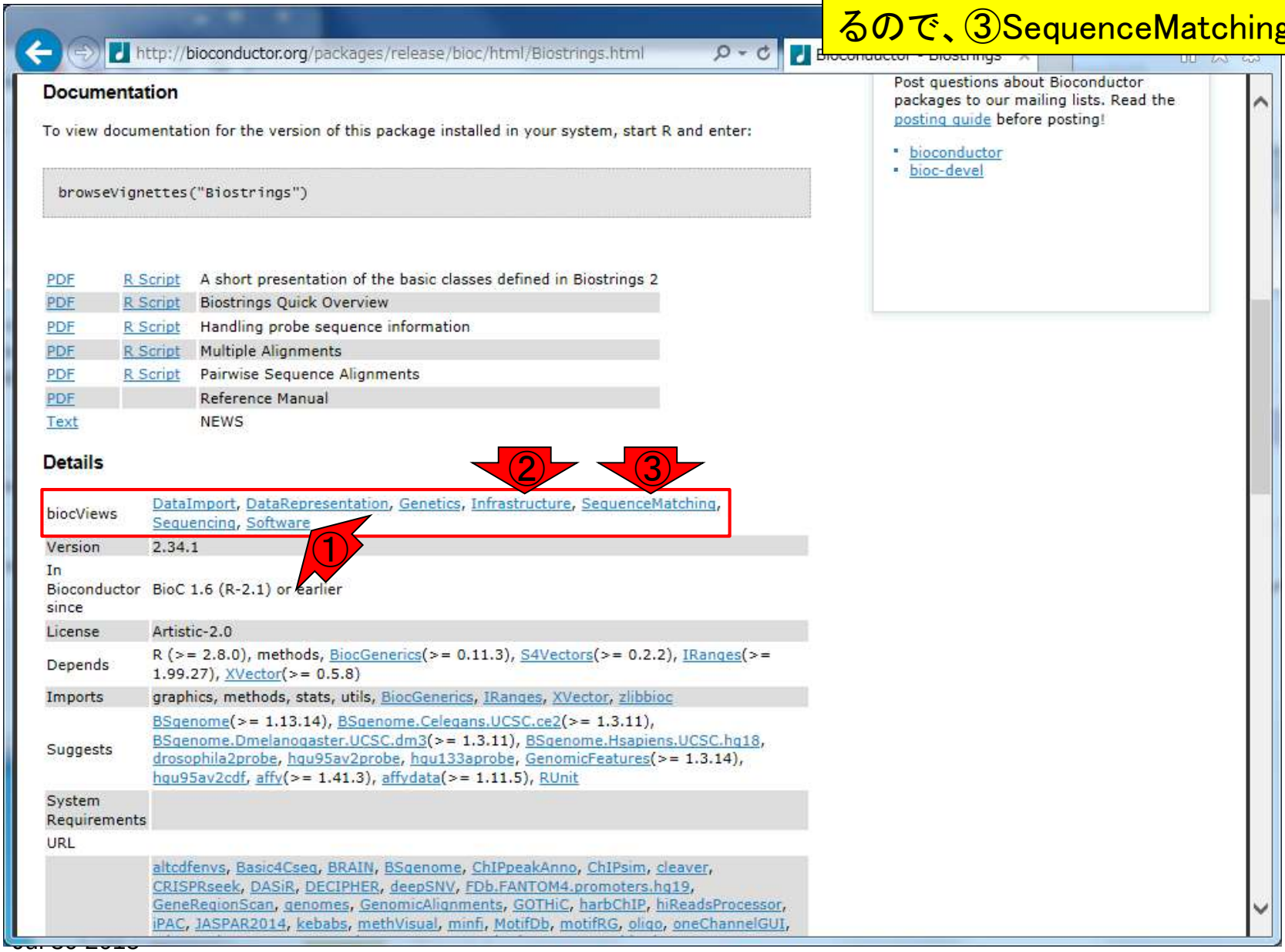
Bioconductor概観

DataRepresentationのカテゴリに含まれるのは34パッケージであることが分かる。赤枠部分がそのリスト。Biostringsは、①大分類はSoftware、②中分類はInfrastructureとなっており、その下の階層の③DataRepresentationに含まれていることがわかる。



Bioconductor概観

①大分類のSoftware、②中分類のInfrastructureも存在する。Biostringsは塩基配列の切り出しや文字列検索系関数も提供しているので、③SequenceMatchingがあるのも妥当。



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

Documentation

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

```
browseVignettes("Biostrings")
```

PDF	R Script	A short presentation of the basic classes defined in Biostrings 2
PDF	R Script	Biostrings Quick Overview
PDF	R Script	Handling probe sequence information
PDF	R Script	Multiple Alignments
PDF	R Script	Pairwise Sequence Alignments
PDF		Reference Manual
Text		NEWS

Post questions about Bioconductor packages to our mailing lists. Read the posting guide before posting!

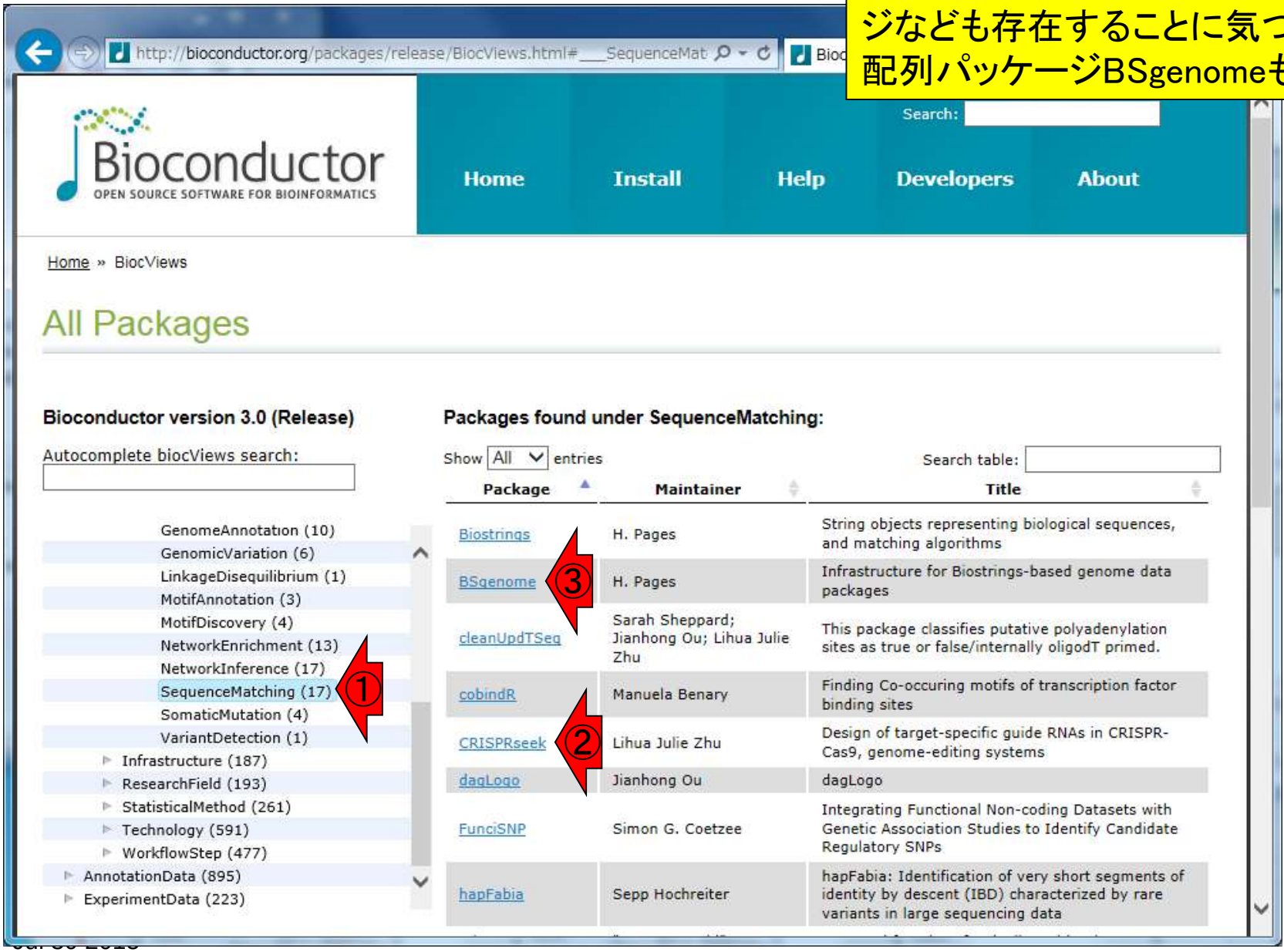
- bioconductor
- bioc-devel

Details

biocViews	DataImport, DataRepresentation, Genetics, Infrastructure, SequenceMatching, Sequencing, Software
Version	2.34.1
In Bioconductor since	BioC 1.6 (R-2.1) or earlier
License	Artistic-2.0
Depends	R (>= 2.8.0), methods, BiocGenerics(>= 0.11.3), S4Vectors(>= 0.2.2), IRanges(>= 1.99.27), XVector(>= 0.5.8)
Imports	graphics, methods, stats, utils, BiocGenerics, IRanges, XVector, zlibbioc
Suggests	BSgenome(>= 1.13.14), BSgenome.Celegans.UCSC.ce2(>= 1.3.11), BSgenome.Dmelanogaster.UCSC.dm3(>= 1.3.11), BSgenome.Hsapiens.UCSC.hg18, drosophila2probe, hgu95av2probe, hgu133aprobe, GenomicFeatures(>= 1.3.14), hgu95av2cdf, affy(>= 1.41.3), affydata(>= 1.11.5), RUnit
System Requirements	
URL	
	altcdfenvs, Basic4Cseq, BRAIN, BSgenome, ChIPpeakAnno, ChIPsim, cleaver, CRISPRseek, DASiR, DECIPHER, deepSNV, FDb.FANTOM4.promoters.hg19, GeneRegionScan, genomes, GenomicAlignments, GOTHiC, harbChIP, hiReadsProcessor, iPAC, JASPAR2014, kebabs, methVisual, minfi, MotifDb, motifRG, oligo, oneChannelGUI,

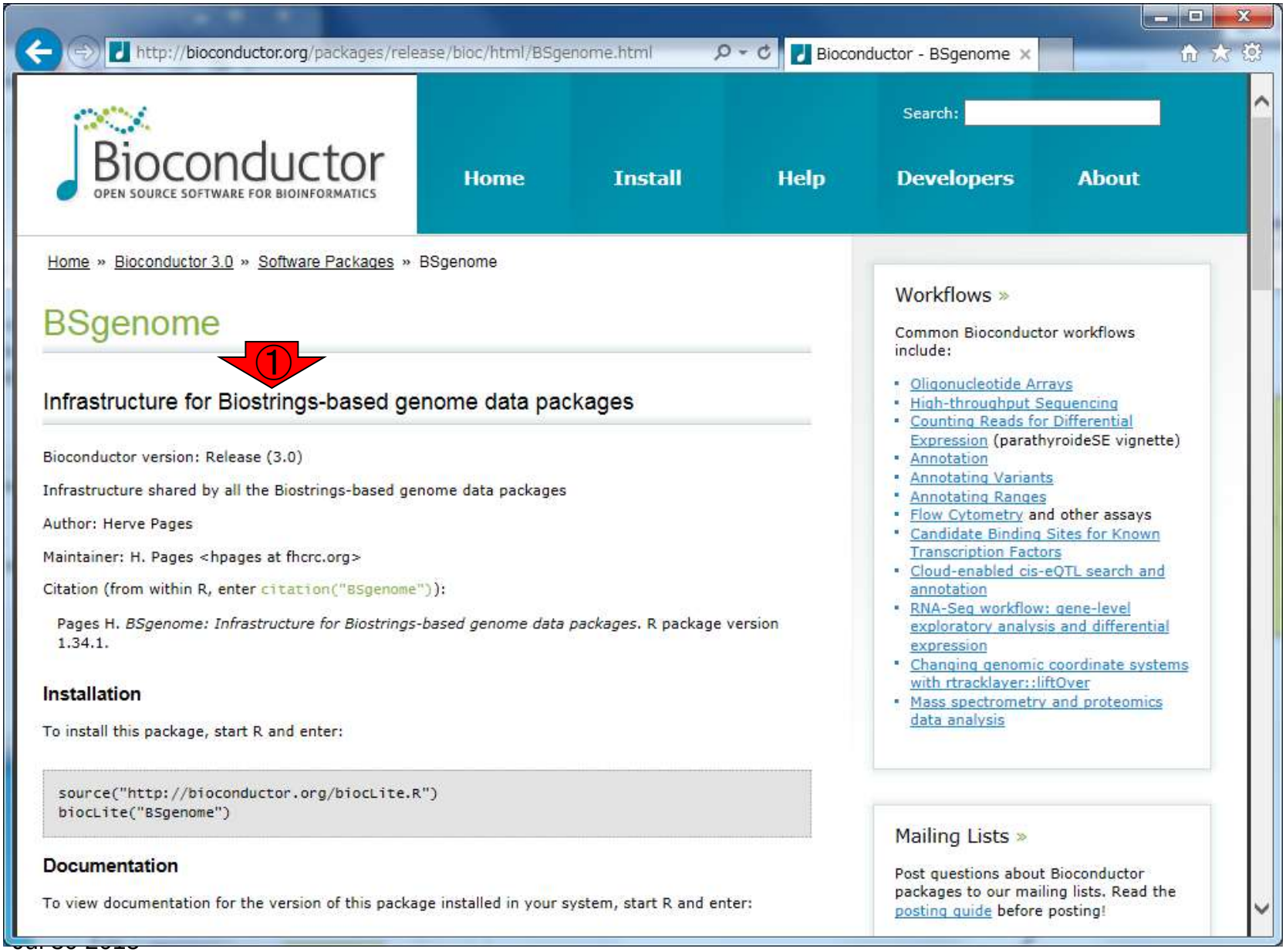
Bioconductor概観

①SequenceMatchingに含まれる17パッケージの一部しか表示されていないが、(Rで)**塩基配列解析**中にはない②CRISPR関連のパッケージなども存在することに気づく。また、③ゲノム配列パッケージBSgenomeもあることがわかる。



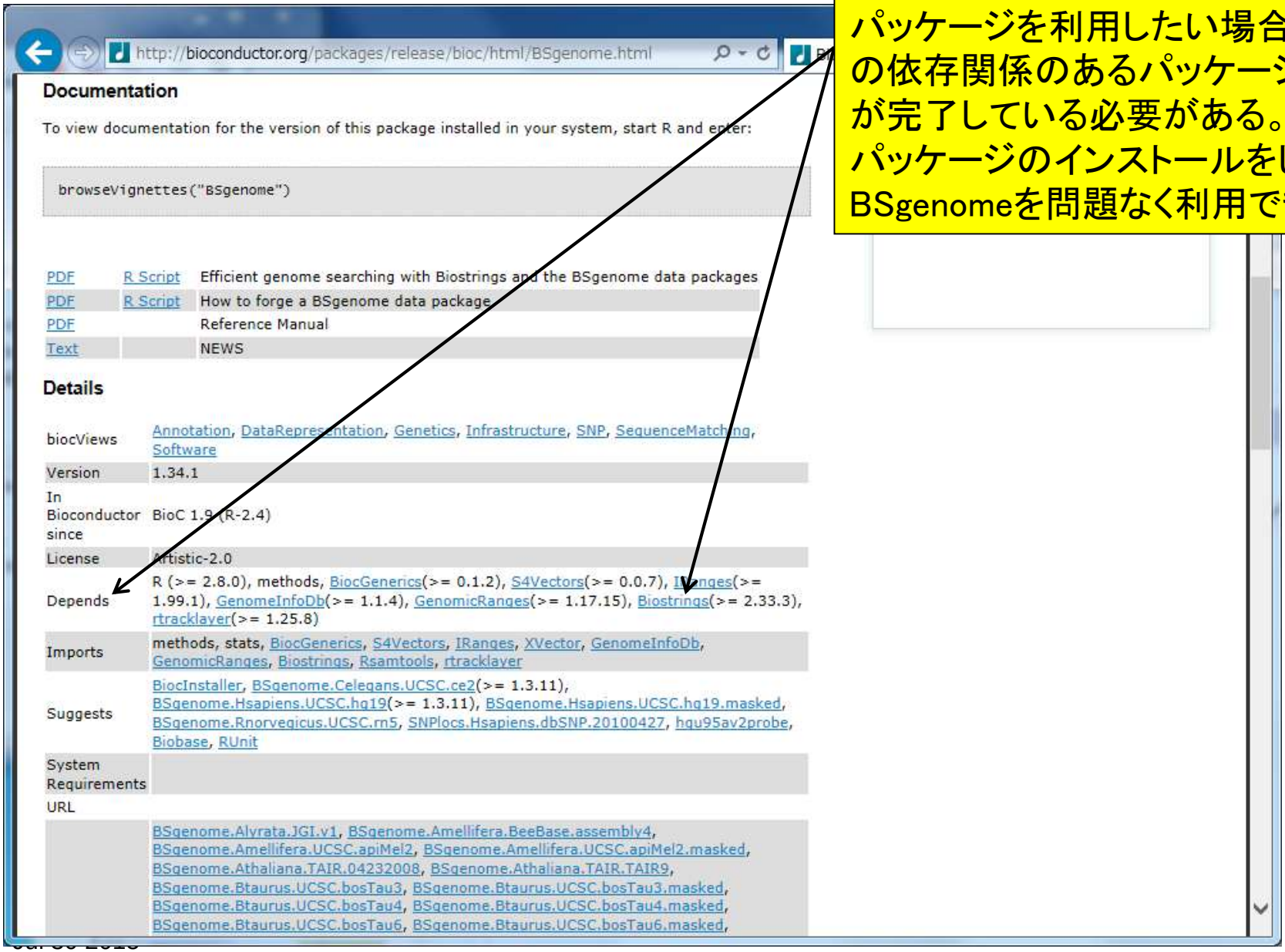
Bioconductor概観

ゲノム配列パッケージBSgenomeは、①内部的にBiostringsパッケージを利用していることがわかる。



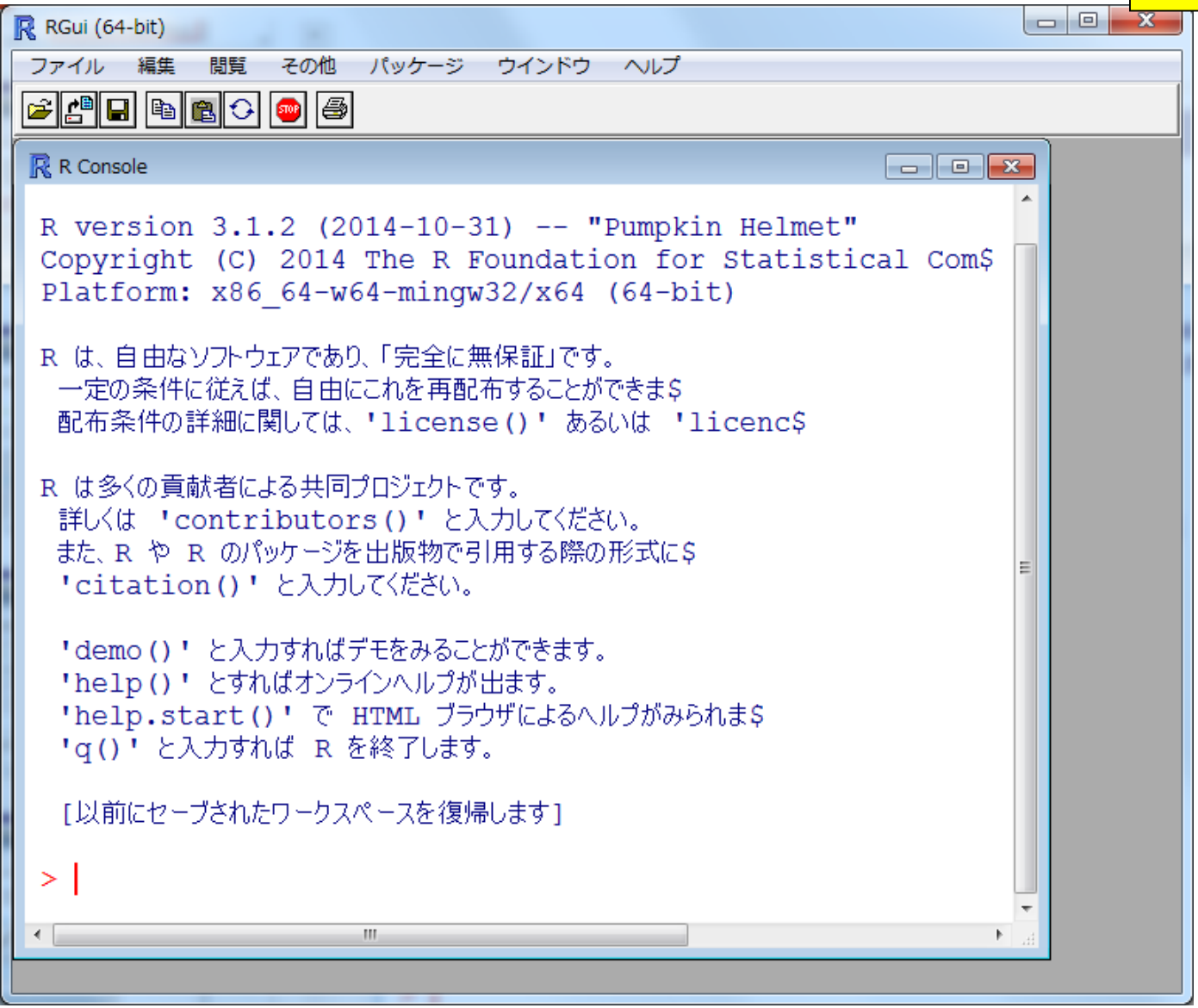
Bioconductor概観

ゲノム配列パッケージBSgenomeのちょっと下のほうに移動。Depends(依存)のところにBiostringsが存在することがわかる。BSgenomeパッケージを利用したい場合には、予めこれらの依存関係のあるパッケージ群のインストールが完了している必要がある。推奨手順通りにパッケージのインストールをしていればBSgenomeを問題なく利用できるはずである。



Bioconductor概観

BSgenomeを問題なく利用できるかは、library(BSgenome)が通るかどうかで判断。R Guiの新規画面を起動。



Bioconductor概観

library(BSgenome)の実行結果中にエラーメッセージが出ていなければOK

RGui (64-bit)

ファイル 編集 閲覧 その他 パッケージ ウインドウ ヘルプ

R Console

```
R version 3.1.2 (2014-10-31) -- "Pumpkin Helmet"
Copyright (C) 2014 The R Foundation for Statisti
Platform: x86_64-w64-mingw32/x64 (64-bit)

R は、自由なソフトウェアであり、「完全に無保証」です。
 一定の条件に従えば、自由にこれを再配布することができま$
 配布条件の詳細に関しては、'license()' あるいは 'licenc$

R は多くの貢献者による共同プロジェクトです。
 詳しくは 'contributors()' と入力してください。
 また、R や R のパッケージを出版物で引用する際の形式に$
 '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。

 [以前にセーブされたワークスペースを復帰します]

> library(BSgenome)
```

R Console

```
> library(BSgenome)
 要求されたパッケージ BiocGenerics をロード中です
 要求されたパッケージ parallel をロード中です

 次のパッケージを付け加えます: 'BiocGenerics'

The following objects are masked from 'package:paralle$

    clusterApply, clusterApplyLB, clusterCall,
    clusterEvalQ, clusterExport, clusterMap,
    parApply, parCapply, parLapply, parLapplyLB,
    parRapply, parSapply, parSapplyLB

The following object is masked from 'package:stats':

    xtabs

The following objects are masked from 'package:base':

    anyDuplicated, append, as.data.frame,
    as.vector, cbind, colnames, do.call,
    duplicated, eval, evalq, Filter, Find, get,
    intersect, is.unsorted, lapply, Map, mapply,
    match, mget, order, paste, pmax, pmax.int,
    pmin, pmin.int, Position, rank, rbind,
    Reduce, rep.int, rownames, sapply, setdiff,
    sort, table, tapply, union, unique, unlist

 要求されたパッケージ IRanges をロード中です
 要求されたパッケージ GenomicRanges をロード中です
 要求されたパッケージ GenomeInfoDb をロード中です
 要求されたパッケージ Biostrings をロード中です
 要求されたパッケージ XVector をロード中です
>
```

Bioconductor概観

library(XXX)をやったときに、XXXパッケージ内部で利用するDependsやImportsにリストアップされているパッケージも同時にロードしている(読み込んでいる)。

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

Documentation

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

```
browseVignettes("BSgenome")
```

PDF	R Script	Efficient genome searching with Biostrings and the BSgenome data packages
PDF	R Script	How to forge a BSgenome data package
PDF		Reference Manual
Text		NEWS

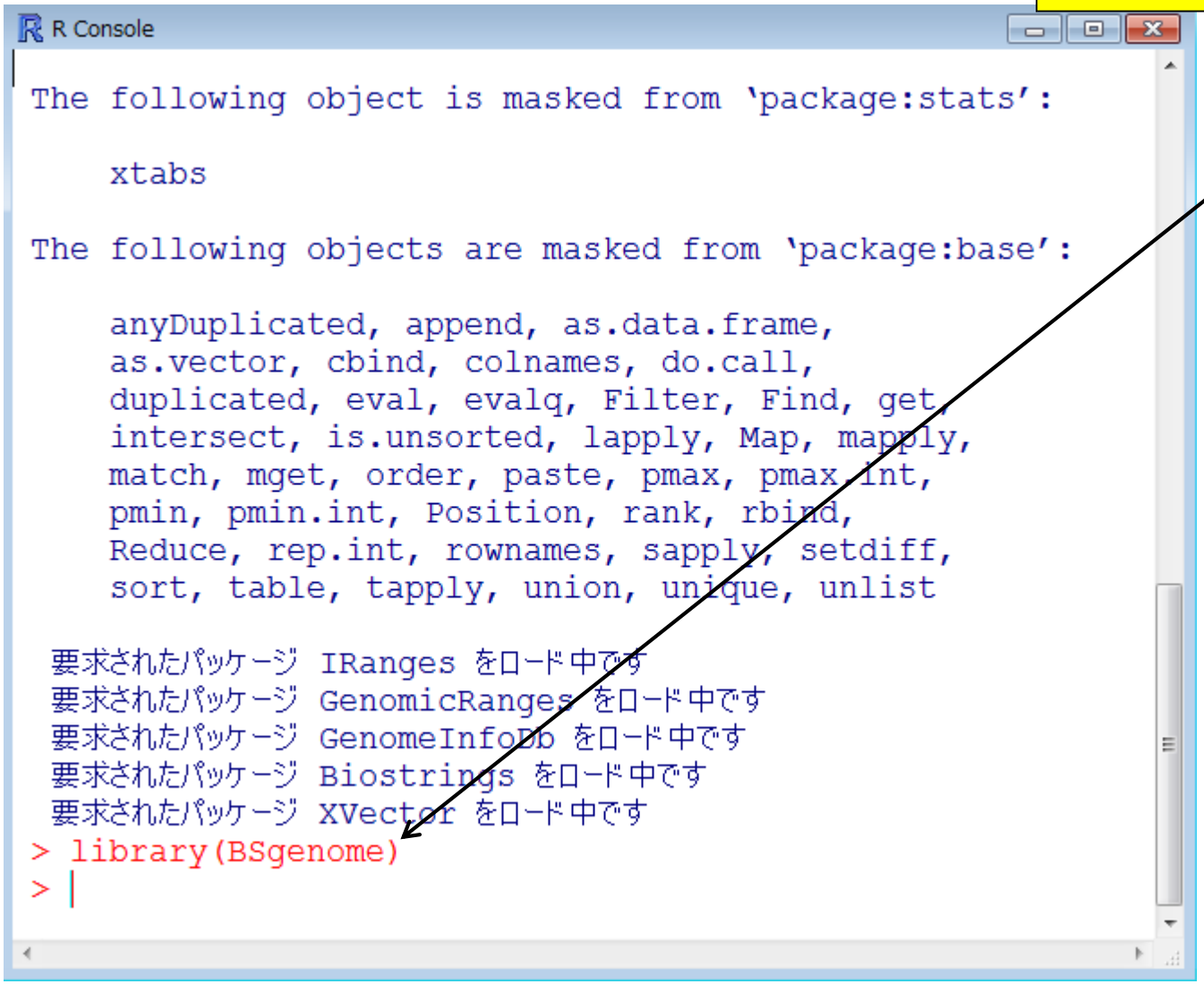
Details

biocViews	Annotation, DataRepresentation, Genetics, Infrastructure, SNP, SequenceMatching, Software
Version	1.34.1
In Bioconductor since	BioC 1.9 (R-2.4)
License	Artistic-2.0
Depends	R (>= 2.8.0), methods, BiocGenerics(>= 0.1.2), S4Vectors(>= 0.0.7), IRanges(>= 1.99.1), GenomeInfoDb(>= 1.1.4), GenomicRanges(>= 1.17.15), Biostrings(>= 2.33 rtracklayer(>= 1.25.8)
Imports	methods, stats, BiocGenerics, S4Vectors, IRanges, XVector, GenomeInfoDb, GenomicRanges, Biostrings, Rsamtools, rtracklayer
Suggests	BiocInstaller, BSgenome.Celegans.UCSC.ce2(>= 1.3.11), BSgenome.Hsapiens.UCSC.hg19(>= 1.3.11), BSgenome.Hsapiens.UCSC.hg19.maske BSgenome.Rnorvegicus.UCSC.rn5, SNPlocs.Hsapiens.dbSNP.20100427, hgu95av2pro Biobase, RUnit
System Requirements	
URL	
	BSgenome.Alyrata.JGI.v1, BSgenome.Amellifera.BeeBase.assembly4, BSgenome.Amellifera.UCSC.apiMel2, BSgenome.Amellifera.UCSC.apiMel2.masked, BSgenome.Athaliana.TAIR.04232008, BSgenome.Athaliana.TAIR.TAIR9, BSgenome.Btaurus.UCSC.bosTau3, BSgenome.Btaurus.UCSC.bosTau3.masked, BSgenome.Btaurus.UCSC.bosTau4, BSgenome.Btaurus.UCSC.bosTau4.masked, BSgenome.Btaurus.UCSC.bosTau6, BSgenome.Btaurus.UCSC.bosTau6.masked,

```
> library(BSgenome)
 要求されたパッケージ BiocGenerics をロード中です
 要求されたパッケージ parallel をロード中です

 次のパッケージを付け加えます: 'BiocGenerics'

The following objects are masked from 'package:paralle$

    clusterApply, clusterApplyLB, clusterCall,
    clusterEvalQ, clusterExport, clusterMap,
    parApply, parCapply, parLapply, parLapplyLB,
    parRapply, parSapply, parSapplyLB

The following object is masked from 'package:stats':

    xtabs

The following objects are masked from 'package:base':

    anyDuplicated, append, as.data.frame,
    as.vector, cbind, colnames, do.call,
    duplicated, eval, evalq, Filter, Find, get,
    intersect, is.unsorted, lapply, Map, mapply,
    match, mget, order, paste, pmax, pmax.int,
    pmin, pmin.int, Position, rank, rbind,
    Reduce, rep.int, rownames, sapply, setdiff,
    sort, table, tapply, union, unique, unlist

 要求されたパッケージ IRanges をロード中です
 要求されたパッケージ GenomicRanges をロード中です
 要求されたパッケージ GenomeInfoDb をロード中です
 要求されたパッケージ Biostrings をロード中です
 要求されたパッケージ XVector をロード中です
>
```

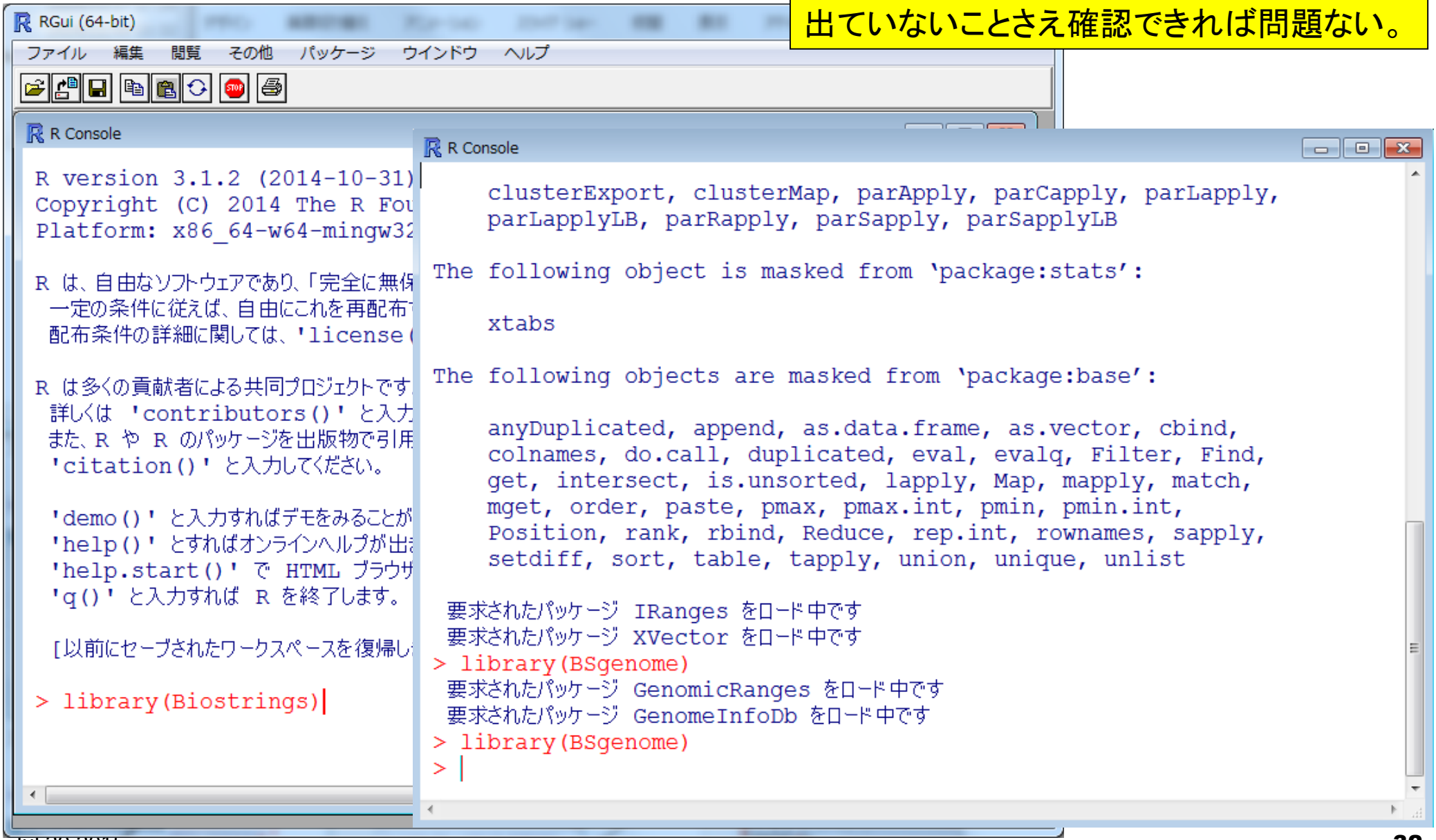
Bioconductor概観

もう一度library(BSgenome)をやってもメッセージは出ません。エラーメッセージが出ていなければ特に気にする必要はありません。



Bioconductor概観

先にBSgenomeが内部的に利用しているBiostringsパッケージのロードを行っておくと、若干見栄えが異なるが、エラーメッセージが出ていないことさえ確認できれば問題ない。



Contents

- パッケージ
  - CRANとBioconductor
  - 推奨パッケージインストール手順のおさらい
  - Bioconductor概観 → ゲノム配列パッケージ(BSgenome)
  - ヒトゲノム情報パッケージの解析
- プロモーター配列取得(sessionInfo、バージョンの違い)
  - Rパッケージ(ゲノムとアノテーションパッケージの併用)
  - FASTA形式ファイルとGFF3形式ファイル
  - データの型
- FASTQファイルの各種解析(LinuxとRを相補的に活用)
  - Linux (FastQC)とR (ShortRead)で同じ:Overrepresented sequences項目
  - Linux (FastQC)とR (QuasR)で異なる見栄え:Per base sequence content項目。FastQCに--nogroupというオプションがあることを知る。
  - FastQCのオプション (デフォルトと--nogroupあり)の違いによるKmer Content項目の結果の違い → Rで検証

BSgenome利用の意義

ゲノム配列情報はUCSCやEnsemblなどのウェブサイトから取得するのが一般的ではあるが、Rの生物種ごとに提供されているBSgenomeで取得、あるいは取り扱うことも可能。ChIP-seq用パッケージMEDIPSはBSgenomeを利用。

- 解析 | 機能解析 | パスウェイ(Pathway)解析 | SeqGSEA(Wang 2014) (last modified 2014/
- 解析 | 菌叢解析 | について (last modified 2014/12/19)
- 解析 | 菌叢解析 | phyloseq(McMurdie 2013) (last modified 2014/05/29)
- 解析 | エクソーム解析 | について (last modified 2
- 解析 | ChIP-seq | について modified 2015/02
- 解析 | ChIP-seq | DiffBind(Ross-Innes 2012) (las
- 解析 | ChIP-seq | ChIPseqR(Humburg 2011) (last
- 解析 | ChIP-seq | chipseq (last modified 2011/12/1
- 解析 | ChIP-seq | PICS(Zhang 2011) (last modifie

解析 | ChIP-seq | について

このあたりはほとんどノータッチです。SraTailor (Oki et al., 2014)は、実験医学2014年12月号の「Close Up実験法」中で日本語による解説記事があります(沖 真弥氏提供情報)。
2015年2月に調査した結果をリストアップします。

R用:

- ChIPsim: Zhang et al., PLoS Comput. Biol., 2008
- PeakSeq法: Rozowsky et al., Nat Biotechnol., 2009
- CSAR: Kaufmann et al., PLoS Biol., 2009
- rMAT: Droit et al., Bioinformatics, 2010
- ChIPpeakAnno: Zhu et al., BMC Bioinformatics, 2010
- PICS: Zhang et al., Biometrics, 2011
- ChIPseqR: Humburg et al., BMC Bioinformatics, 2011
- DiffBind: Ross-Innes et al., Nature, 2012
- MEDIPS: Lienhard et al., Bioinformatics, 2014
- DSS: Feng et al., Nucleic Acids Res., 2014
- methylSig: Park et al., Bioinformatics, 2014

R以外:

- bwtool: Pohl and Beato, Bioinformatics, 2014
- SraTailor: Oki et al., Genes Cells., 2014

Review、ガイドライン、パイプライン系:

- ガイドライン: Bailey et al., PLoS Comput Biol., 2013
- Review: Robinson et al., Front Genet., 2014

BSgenome

プロモーター配列取得ではなく、
①ゲノム配列取得のほうです！

- イントロ | 一般 | 任意の長さの連続塩基の出現頻度情報を取得 (last modified 2013/06/14)
- イントロ | 一般 | Tips | 任意の拡張子でファイルを保存 (last modified 2013/09/26)
- イントロ | 一般 | Tips | 拡張子は同じで任意の文字を追加して保存 (last modified 2013/09/26)
- イントロ | 一般 | 配列取得 | ゲノム配列 | 公共DBから (last modified 2014/05/28)
- イントロ | 一般 | 配列取得 | ゲノム配列 | BSgenome (last modified 2015/02/18) NEW
- イントロ | 一般 | 配列取得 | プロモーター配列 | 公共DBから (last modified 2014/04/02)
- イントロ | 一般 | 配列取得 | プロモーター配列 | BSgenome (last modified 2014/04/25)
- イントロ | 一般 | 配列取得 | プロモーター配列 | GenomicFeatures(Lawrence 2013) (last modified 2014/04/23)
- イントロ | 一般 | 配列取得 | トランスクリ
- イントロ | 一般 | 配列取得 | トランスクリ
- イントロ | NGS | 様々なプラットフォーム
- イントロ | NGS | qPCRやmicroarrayなど

①

イントロ | 一般 | 配列取得 | ゲノム配列 | BSgenome NEW

BSgenomeパッケージを用いて様々な生物種のゲノム配列を取得するやり方を示します。ミヤマハタザオ (A. lyrata)、セイヨウミツバチ (A. mellifera)、シロイヌナズナ(A.thaliana)、ウシ(B.taurus)、線虫(C.elegans)、犬(C.familiaris)、キイロショウジョウバエ(D.melanogaster)、ゼブラフィッシュ(D.rerio)、大腸菌(E.coli)、イトヨ(G.aculeatus)、セキショクヤケイ(G.gallus)、ヒト(H.sapiens)、アカゲザル(M.mulatta)、マウス(M.musculus)、チンパンジー(P.troglodytes)、ラット(R.norvegicus)、出芽酵母(S.cerevisiae)、トキソプラズマ(T.gondii)と実に様々な生物種が利用可能であることがわかります。getSeq関数はBSgenomeオブジェクト 中の「single sequences」というあたりにリストアップされているchr...というものを全て抽出しています。したがって、例えばマウスゲノムは「chr1」以外に「chr1_random」や「chrUn_random」なども等価に取扱っている点に注意してください。
「ファイル」-「ディレクトリの変更」でファイルを保存したいディレクトリに移動し以下をコピペ。

1. 利用可能な生物種とRにインストール済みの生物種をリストアップしたい場合:

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

#本番（利用可能なパッケージをリストアップ；インストール済みとは限らない）
available.genomes()                        #このパッケージ中で利用可能なゲノムをリストアップ

#本番（インストール済みの生物種をリストアップ）
installed.genomes()                        #インストール済みの生物種をリストアップ

#後処理（パッケージ名でだいたいわかるがproviderやversionを分割して表示したい場合）
installed.genomes(splitNameParts=TRUE) #インストール済みの生物種をリストアップ
```

BSgenome

黒枠部分のコードをコピペ。R ver. 3.1.3 (Bioconductor ver. 3.0)で利用可能な生物種のパッケージ名をリストアップ。71個あることが分かる。Rのバージョンが古いとパッケージ数は少なくなる。例えばR ver. 3.2.0 (Bioconductor ver. 3.1)では77個。

イントロ | 一般 | 配列取得 | ゲノム配列 | BSgenome NEW

BSgenomeパッケージを用いて様々な生物種のゲノム配列を取得するやり方を示します。ミヤマハ
lyrata)、セイヨウミツバチ (A. mellifera)、シロイヌナズナ(A.thaliana)、ウシ(B.taurus)、線虫(C.ele
(C.familiaris)、キイロショウジョウバエ(D.melanogaster)、ゼブラフィッシュ(D.rerio)、大腸菌(E.c
(G.aculeatus)、セキショクヤケイ(G.gallus)、ヒト(H.sapiens)、アカゲザル(M.mulatta)、マウス(M.musculus)、チ
ンパンジー(P.troglodytes)、ラット(R.norvegicus)、出芽酵母(S.cerevisiae)、トキソプラズマ(T.gondii)と実に様々
な生物種が利用可能であることがわかります。getSeq関数はBSgenomeオブジェクト 中の「single sequences」という
あたりにリストアップされているchr...というものを全て抽出しています。したがって、例えばマウスゲノムは「chr1」
以外に「chr1_random」や「chrUn_random」なども等価に取扱っている
「ファイル」-「ディレクトリの変更」でファイルを保存したいディレクトリに

1. 利用可能な生物種とRにインストール済みの生物種をリストアップ

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

#本番（利用可能なパッケージをリストアップ；インストール
available.genomes()                            #このパッケー

#本番（インストール済みの生物種をリストアップ）
installed.genomes()                            #インストー

#後処理（パッケージ名でだいたいわかるがproviderやversi
installed.genomes(splitNameParts=TRUE) #インストー
```

R Console

```
[56] "BSgenome.Ptroglodytes.UCSC.panTro2"
[57] "BSgenome.Ptroglodytes.UCSC.panTro2.masked"
[58] "BSgenome.Ptroglodytes.UCSC.panTro3"
[59] "BSgenome.Ptroglodytes.UCSC.panTro3.masked"
[60] "BSgenome.Rnorvegicus.UCSC.rn4"
[61] "BSgenome.Rnorvegicus.UCSC.rn4.masked"
[62] "BSgenome.Rnorvegicus.UCSC.rn5"
[63] "BSgenome.Rnorvegicus.UCSC.rn5.masked"
[64] "BSgenome.Scerevisiae.UCSC.sacCer1"
[65] "BSgenome.Scerevisiae.UCSC.sacCer2"
[66] "BSgenome.Scerevisiae.UCSC.sacCer3"
[67] "BSgenome.Sscrofa.UCSC.susScr3"
[68] "BSgenome.Sscrofa.UCSC.susScr3.masked"
[69] "BSgenome.Tgondii.ToxoDB.7.0"
[70] "BSgenome.Tguttata.UCSC.taeGut1"
[71] "BSgenome.Tguttata.UCSC.taeGut1.masked"
> |
```

赤枠のヒトゲノムは4バージョン(hg17, 18, 19, GRCh38)を利用可能。2013年12月にリリースされた①最新版(GRCh38)のRパッケージも利用可能。

R Console

```
> available.genomes()
 [1] "BSgenome.Alyrata.JGI.v1"
 [2] "BSgenome.Amellifera.BeeBase.asse
 [3] "BSgenome.Amellifera.UCSC.apiMel2
 [4] "BSgenome.Amellifera.UCSC.apiMel2
 [5] "BSgenome.Athaliana.TAIR.04232008
 [6] "BSgenome.Athaliana.TAIR.TAIR9"
 [7] "BSgenome.Btaurus.UCSC.bosTau3"
 [8] "BSgenome.Btaurus.UCSC.bosTau3.ma
 [9] "BSgenome.Btaurus.UCSC.bosTau4"
[10] "BSgenome.Btaurus.UCSC.bosTau4.ma
[11] "BSgenome.Btaurus.UCSC.bosTau6"
[12] "BSgenome.Btaurus.UCSC.bosTau6.ma
[13] "BSgenome.Celegans.UCSC.ce10"
[14] "BSgenome.Celegans.UCSC.ce2"
[15] "BSgenome.Celegans.UCSC.ce6"
[16] "BSgenome.Cfamiliaris.UCSC.canFam
[17] "BSgenome.Cfamiliaris.UCSC.canFam
[18] "BSgenome.Cfamiliaris.UCSC.canFam
[19] "BSgenome.Cfamiliaris.UCSC.canFam
[20] "BSgenome.Dmelanogaster.UCSC.dm2"
[21] "BSgenome.Dmelanogaster.UCSC.dm2.
[22] "BSgenome.Dmelanogaster.UCSC.dm3"
[23] "BSgenome.Dmelanogaster.UCSC.dm3.
[24] "BSgenome.Drerio.UCSC.danRer5"
[25] "BSgenome.Drerio.UCSC.danRer5.mas
[26] "BSgenome.Drerio.UCSC.danRer6"
[27] "BSgenome.Drerio.UCSC.danRer6.mas
[28] "BSgenome.Drerio.UCSC.danRer7"
```

R Console

```
[29] "
[30] "BSgenome.Ecoli.NCBI.20080805"
[31] "BSgenome.Gaculeatus.UCSC.gasAcu1"
[32] "BSgenome.Gaculeatus.UCSC.gasAcu1.masked"
[33] "BSgenome.Ggallus.UCSC.galGal3"
[34] "BSgenome.Ggallus.UCSC.galGal3.masked"
[35] "BSgenome.Ggallus.UCSC.galGal4"
[36] "BSgenome.Ggallus.UCSC.galGal4.masked"
[37] "BSgenome.Hsapiens.NCBI.GRCh38"
[38] "BSgenome.Hsapiens.UCSC.hg17"
[39] "BSgenome.Hsapiens.UCSC.hg17.masked"
[40] "BSgenome.Hsapiens.UCSC.hg18"
[41] "BSgenome.Hsapiens.UCSC.hg18.masked"
[42] "BSgenome.Hsapiens.UCSC.hg19"
[43] "BSgenome.Hsapiens.UCSC.hg19.masked"
[44] "BSgenome.Mfuro.UCSC.musFur1"
[45] "BSgenome.Mmulatta.UCSC.rheMac2"
[46] "BSgenome.Mmulatta.UCSC.rheMac2.masked"
[47] "BSgenome.Mmulatta.UCSC.rheMac3"
[48] "BSgenome.Mmulatta.UCSC.rheMac3.masked"
[49] "BSgenome.Mmusculus.UCSC.mm10"
[50] "BSgenome.Mmusculus.UCSC.mm10.masked"
[51] "BSgenome.Mmusculus.UCSC.mm8"
[52] "BSgenome.Mmusculus.UCSC.mm8.masked"
[53] "BSgenome.Mmusculus.UCSC.mm9"
[54] "BSgenome.Mmusculus.UCSC.mm9.masked"
[55] "BSgenome.Osativa.MSU.MSU7"
```

①

BSgenome

黒枠部分のコードをコピペ。10秒程度かかる。実際にインストール済みのものは(このPC環境では)7パッケージであることがわかる。講習会ではヒトゲノムの最新版(BSgenome.Hsapiens.NCBI.GRCh38)を使いますが、①これが入ってない場合は、②各自installed.genomes()で見られる他のパッケージに置き換えて実行してください。

イントロ | 一般 | 配列取得 | ゲノム配列 | BSgenome NE

BSgenomeパッケージを用いて様々な生物種のゲノム配列を取得するやり方を示し
lyrata)、セイヨウミツバチ (A. mellifera)、シロイヌナズナ(A.thaliana)、ウシ(B.taur
(C.familiaris)、キイロショウジョウバエ(D.melanogaster)、ゼブラフィッシュ(D.rerio
(G.aculeatus)、セキショクヤケイ(G.gallus)、ヒト(H.sapiens)、アカゲザル(M.mula
ンパンジー(P.troglodytes)、ラット(R.norvegicus)、出芽酵母(S.cerevisiae)、トキソ
な生物種が利用可能であることがわかります。getSeq関数はBSgenomeオブジェクト 中の「single sequences」というあたりにリストアップされているchr...というものを全て抽出しています。したがって、例えばマウスゲノムは「chr1」以外に「chr1_random」や「chrUn_random」なども等価に取扱っている点に注意してください。
「ファイル」-「ディレクトリの変更」でファイルを保存したいディレクトリに移動し以下をコピペ。

1. 利用可能な生物種とRにインストール済みの生物種をリストアップしたい場合:

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

#本番（利用可能なパッケージをリストアップ；インストール済みと
available.genomes()                        #このパッケージ中で

#本番（インストール済みの生物種をリストアップ）
installed.genomes()                        #インストール済み

#後処理（パッケージ名でだいたいわかるがproviderやversion等
installed.genomes(splitNameParts=TRUE) #インストール済みの
```

②

R Console

```
[68] "BSgenome.Sscrofa.UCSC.susScr3.masked"
[69] "BSgenome.Tgondii.ToxoDB.7.0"
[70] "BSgenome.Tguttata.UCSC.taeGut1"
[71] "BSgenome.Tguttata.UCSC.taeGut1.masked"
> installed.genomes()                      #イン$
[1] "BSgenome.Athaliana.TAIR.TAIR9"
[2] "BSgenome.Celegans.UCSC.ce2"
[3] "BSgenome.Celegans.UCSC.ce6"
[4] "BSgenome.Drerio.UCSC.danRer7"
[5] "BSgenome.Hsapiens.NCBI.GRCh38"
[6] "BSgenome.Hsapiens.UCSC.hg19"
[7] "BSgenome.Mmusculus.UCSC.mm10"
>
```

①

Contents

- パッケージ
  - CRANとBioconductor
  - 推奨パッケージインストール手順のおさらい
  - Bioconductor概観 → ゲノム配列パッケージ(BSgenome)
  - ヒトゲノム情報パッケージの解析
- プロモーター配列取得(sessionInfo、バージョンの違い)
  - Rパッケージ(ゲノムとアノテーションパッケージの併用)
  - FASTA形式ファイルとGFF3形式ファイル
  - データの型
- FASTQファイルの各種解析(LinuxとRを相補的に活用)
  - Linux (FastQC)とR (ShortRead)で同じ:Overrepresented sequences項目
  - Linux (FastQC)とR (QuasR)で異なる見栄え:Per base sequence content項目。FastQCに--nogroupというオプションがあることを知る。
  - FastQCのオプション (デフォルトと--nogroupあり)の違いによるKmer Content項目の結果の違い → Rで検証

BSgenome

①例題9。2013年12月にリリースされたヒトゲノム最新版(GRCh38)のRパッケージを入力、multi-FASTAファイルを出力として得る。作業ディレクトリはどこでもよいが基本はデスクトップ上のhoge。数分かかるが、約3.3GBのファイルが生成される。**決してテキストエディタで開かないで!**

イントロ | 一般 | 配列取得 | ゲノム配列 | BSgenome NEW

BSgenomeパッケージを用いて様々な生物種のゲノム配列を取得するやり方を示しま
lyrata)、セイヨウミツバチ (A. mellifera)、シロイヌナズナ(A.thaliana)、ウシ(B.taurus)、
(C.familiaris)、キイロショウジョウバエ(D.melanogaster)、ゼブラフィッシュ(D.rerio)、
(G.aculeatus)、セキショクヤケイ(G.gallus)、ヒト(H.sapiens)、アカゲザル(M.mulatta)、マウス(M.musculus)、チ
ンパンジー(P.troglodytes)、ラット(R.norvegicus)、出芽酵母(S.cerevisiae)、トキソプラズマ(T.gondii)と実に様々
な生物種が利用可能であることがわかります。getSeq関数はBSgenomeオブジェクト 中の「single sequences」という
あたりにリストアップされているchr、というものを全て抽出しています。したがって、例えばマウスゲノムは「chr1」
以外に「chr1_random
「ファイル」-「ディレクト

1. 利用可能な生物種とR

```
#必要なパッケージを
library(BSgenome)

#本番（利用可能なパ
available.genomes

#本番（インストール
installed.genomes

#後処理（パッケージ
installed.genomes
```

①

9. インストール済みのヒト("BSgenome.Hsapiens.NCBI.GRCh38")のゲノム配列をmulti-FASTAファイルで保存したい場合:

2013年12月にリリースされたGenome Reference Consortium GRCh38です。R ver. 3.1.0とBioconductor ver. 2.14以上の環境で実行可能です。

```
out_f <- "hoge9.fasta"                    #出力ファイル名を指定してout_fに格納
param <- "BSgenome.Hsapiens.NCBI.GRCh38"#パッケージ名を指定

#必要なパッケージをロード
library(param, character.only=T)          #paramで指定したパッケージの読み込み

#前処理(paramで指定したパッケージ中のオブジェクト名をgenomeに統一)
#tmp <- unlist(strsplit(param, ".", fixed=TRUE))[2]#paramで指定した文字列からオブジェクト名を取得し
tmp <- ls(paste("package", param, sep=":"))#paramで指定したパッケージで利用可能なオブジェクト名を取得
genome <- eval(parse(text=tmp))           #文字列tmpをRオブジェクトとしてgenomeに格納(パッケージ中には
genome                                    #確認してるだけです

#本番
fasta <- getSeq(genome)                   #ゲノム塩基配列情報を抽出した結果をfastaに格納
names(fasta) <- seqnames(genome)          #description情報を追加している

#ファイルに保存
writeXStringSet(fasta, file=out_f, format="fasta", width=50)#fastaの中身を指定したファイル名で保存
```

BSgenome

①出力ファイルの内容は、fastaオブジェクトに格納されている。慣れれば②fastaオブジェクトの中身を眺めるほうが全体像をつかみやすい。

9. インストール済みのヒト("BSgenome.Hsapiens.NCBI.GRCh38")のゲノム配列をmulti-FASTAファイルで保存したい場合:

2013年12月にリリースされたGenome Reference Consortium GRCh38です。R ver. 3.1.0とBioconductor ver. 2.14以上の環境で実行可能です。

```
out_f <- "hoge9.fasta"                    #出力ファイル名を指定してout_fに格納
param <- "BSgenome.Hsapiens.NCBI.GRCh38"#パッケージ名を指定

#必要なパッケージをロード
library(param, character.only=T)

#前処理(paramで指定したパッケージ中のオ
#tmp <- unlist(strsplit(param, "."),
tmp <- ls(paste("package", param, s
genome <- eval(parse(text=tmp))
genome

#本番
fasta <- getSeq(genome)
names(fasta) <- seqnames(genome)

#ファイルに保存
writeXStringSet(fasta, file=out_f, f
```

①

②

R Console

```
> #ファイルに保存
> writeXStringSet(fasta, file=out_f, format="fasta", widt$
> fasta
  A DNAStringSet instance of length 455
          width seq                    names
  [1] 248956422 NNNNNNNNN...NNNNNNNNN 1
  [2] 242193529 NNNNNNNNN...NNNNNNNNN 2
  [3] 198295559 NNNNNNNNN...NNNNNNNNN 3
  [4] 190214555 NNNNNNNNN...NNNNNNNNN 4
  [5] 181538259 NNNNNNNNN...NNNNNNNNN 5
  ...       ... ...
[451]    200773 TCTACTCTC...GGGGAATTC HSCHR19KIR_FH08_B...
[452]    170148 TTTCTTTCT...GGGGAATTC HSCHR19KIR_FH13_A...
[453]    215732 TGTGGTGAG...GGGGAATTC HSCHR19KIR_FH13_B...
[454]    170537 TCTACTCTC...GGGGAATTC HSCHR19KIR_FH15_A...
[455]    177381 GATCTATCT...GGGGAATTC HSCHR19KIR_RP5_B_...
> |
```

BSgenome

①1～22番染色体のみ取扱いたい場合。染色体番号の数が大きくなるほど配列長が短くなっている傾向が一目瞭然ですね。

9. インストール済みのヒト("BSgenome.Hsapiens.NCBI.GRCh38")のゲノム配列をmulti-FASTAファイルで保存したい場合:

2013年12月にリリースされたGenome Reference Consortium GRCh38です。R ver. 3.1.0とBioconductor ver. 2.14以上の環境で実行可能です。

```
out_f <- "hoge9.fasta"                    #出力ファイル名を指定してout_fに格納
param <- "BSgenome.Hsapiens.NCBI.GRCh38"#パッケージ名を指定

#必要なパッケージをロード
library(param, character.only=T)

#前処理(paramで指定したパッケージ中のオ
#tmp <- unlist(strsplit(param, "."),
tmp <- ls(paste("package", param, s
genome <- eval(parse(text=tmp))
genome

#本番
fasta <- getSeq(genome)
names(fasta) <- seqnames(genome)

#ファイルに保存
writeXStringSet(fasta, file=out_f, f
```

①

R Console

```
[454]    170537 TCTACTCTC...GGGGAATTC HSCHR19KIR_FH15_A...
[455]    177381 GATCTATCT...GGGGAATTC HSCHR19KIR_RP5_B_...
> fasta[1:22]
  A DNAStringSet instance of length 22
      width seq                    names
 [1] 248956422 NNNNNNNNNN...NNNNNNNNN 1
 [2] 242193529 NNNNNNNNNN...NNNNNNNNN 2
 [3] 198295559 NNNNNNNNNN...NNNNNNNNN 3
 [4] 190214555 NNNNNNNNNN...NNNNNNNNN 4
 [5] 181538259 NNNNNNNNNN...NNNNNNNNN 5
 ...       ... ...
[18]  80373285 NNNNNNNNNN...NNNNNNNNN 18
[19]  58617616 NNNNNNNNNN...NNNNNNNNN 19
[20]  64444167 NNNNNNNNNN...NNNNNNNNN 20
[21]  46709983 NNNNNNNNNN...NNNNNNNNN 21
[22]  50818468 NNNNNNNNNN...NNNNNNNNN 22
>
```

BSgenome

①X, Y, およびミトコンドリア配列も含めたい場合。②配列の並びの確認は試行錯誤。最初からわかっていたわけではありません。R画面上で眺めるほうが、全体像を把握しやすい。

9. インストール済みのヒト("BSgenome.Hsapiens.NCBI.GRCh38")のゲノム配列をmulti-FASTAファイルで保存したい場合:

2013年12月にリリースされたGenome Reference Consortium GRCh38です。R ver. 3.1.0とBioconductor ver. 2.14以上の環境で実行可能です。

```
out_f <- "hoge9.fasta"                    #出力ファイル名を指定してout_fに格納
param <- "BSgenome.Hsapiens.NCBI.GRCh38"#パッケージ名を指定

#必要なパッケージをロード
library(param, character.only=T)

#前処理(paramで指定したパッケージ中のオ
#tmp <- unlist(strsplit(param, "."),
tmp <- ls(paste("package", param, s
genome <- eval(parse(text=tmp))
genome

#本番
fasta <- getSeq(genome)
names(fasta) <- seqnames(genome)

#ファイルに保存
writeXStringSet(fasta, file=out_f, f
```

R Console

```
[21]  46709983 NNNNNNNNNN...NNNNNNNNN 21
[22]  50818468 NNNNNNNNNN...NNNNNNNNN 22
> fasta[1:25]                                              ①
  A DNAStringSet instance of length 25                     ②
         width seq                                         names
 [1] 248956422 NNNNNNNNNN...NNNNNNNNN 1
 [2] 242193529 NNNNNNNNNN...NNNNNNNNN 2
 [3] 198295559 NNNNNNNNNN...NNNNNNNNN 3
 [4] 190214555 NNNNNNNNNN...NNNNNNNNN 4
 [5] 181538259 NNNNNNNNNN...NNNNNNNNN 5
 ...         ... ...
[21]  46709983 NNNNNNNNNN...NNNNNNNNN 21
[22]  50818468 NNNNNNNNNN...NNNNNNNNN 22
[23] 156040895 NNNNNNNNNN...NNNNNNNNN X
[24]  57227415 NNNNNNNNNN...NNNNNNNNN Y
[25]     16569 GATCACAGGT...ATCACGATG MT
>
```

BSgenome

X, Y, およびミトコンドリア配列までのサブセットを hoge10.fastaで保存したい場合。①上矢印キーを何回か押して、ファイルに保存するためのコマンドを出し、②と③の水色下線部分を変更すればよい。

9. インストール済みのヒト("BSgenome.Hsapiens.NCBI.GRCh38")のゲノ

2013年12月にリリースされたGenome Reference Consortium GRCh38です。R ver. 3.1.0とBioconductor ver. 2.14以上の環境で実行可能です。

```
out_f <- "hoge9.fasta"                    #出力ファイル名を指定してout_fに格納
param <- "BSgenome.Hsapiens.NCBI.GRCh38"#パッケージ名を指定

#必要なパッケージをロード
library(param, character.only=T)

#前処理(paramで指定したパッケージ中のオ
#tmp <- unlist(strsplit(param, "."),
tmp <- ls(paste("package", param, sep
genome <- eval(parse(text=tmp))
genome

#本番
fasta <- getSeq(genome)
names(fasta) <- seqnames(genome)

#ファイルに保存
writeXStringSet(fasta, file=out_f, fo
```

R Console

```
[21]   46709983 NNNNNNNNNN...NNNNNNNNN 21
[22]   50818468 NNNNNNNNNN...NNNNNNNNN 22
> fasta[1:25]
  A DNAStringSet instance of length 25
         width seq                    names
 [1] 248956422 NNNNNNNNNN...NNNNNNNNN 1
 [2] 242193529 NNNNNNNNNN...NNNNNNNNN 2
 [3] 198295559 NNNNNNNNNN...NNNNNNNNN 3
 [4] 190214555 NNNNNNNNNN...NNNNNNNNN 4
 [5] 181538259 NNNNNNNNNN...NNNNNNNNN 5
 ...       ... ...
[21]   46709983 NNNNNNNNNN...NNNNNNNNN 21
[22]   50818468 NNNNNNNNNN...NNNNNNNNN 22
[23] 156040895 NNNNNNNNNN...NNNNNNNNN X
[24]   57227415 NNNNNNNNNN...NNNNNNNNN Y
[25]      16569 GATCACAGGT...ATCACGATG MT
> writeXStringSet(fasta, file=out_f, format="fasta", widt$
```

①

②

③

BSgenome

②と③をこんな感じで変更。実行後にhoge9.fastaよりも若干ファイルサイズの小さいhoge10.fastaが生成されていることが確認できるはず。**決してテキストエディタで開かないで!**

9. インストール済みのヒト("BSgenome.Hsapiens.NCBI.GRCh38")のゲノム配列をmulti-FASTAファイルで保存したい場合:

2013年12月にリリースされたGenome Reference Consortium GRCh38です。R ver. 3.1.0とBioconductor ver. 2.14以上の環境で実行可能です。

```
out_f <- "hoge9.fasta"                    #出力ファイル名を指定してout_fに格納
param <- "BSgenome.Hsapiens.NCBI.GRCh38"#パッケージ名を指定

#必要なパッケージをロード
library(param, character.only=T)

#前処理(paramで指定したパッケージ中のオ
#tmp <- unlist(strsplit(param, "."),
tmp <- ls(paste("package", param, sep
genome <- eval(parse(text=tmp))
genome

#本番
fasta <- getSeq(genome)
names(fasta) <- seqnames(genome)

#ファイルに保存
writeXStringSet(fasta, file=out_f, fo
```

R Console

```
[22]  50818468 NNNNNNNNNN...NNNNNNNNN 22
> fasta[1:25]
  A DNAStringSet instance of length 25
         width seq                    names
 [1] 248956422 NNNNNNNNNN...NNNNNNNNN 1
 [2] 242193529 NNNNNNNNNN...NNNNNNNNN 2
 [3] 198295559 NNNNNNNNNN...NNNNNNNNN 3
 [4] 190214555 NNNNNNNNNN...NNNNNNNNN 4
 [5] 181538259 NNNNNNNNNN...NNNNNNNNN 5
 ...       ... ...
[21]  46709983 NNNNNNNNNN...NNNNNNNNN 21
[22]  50818468 NNNNNNNNNN...NNNNNNNNN 22
[23] 156040895 NNNNNNNNNN...NNNNNNNNN X
[24]  57227415 NNNNNNNNNN...NNNNNNNNN Y
[25]     16569 GATCACAGGT...ATCACGATG MT
> writeXStringSet(fasta[1:25], file="hoge10.fasta", forma$
>
```

②

③

参考

①例題10。②様々な記述形式があります。やらなくていいです。**決してテキストエディタで開かないで!**

BSgenome

イントロ | 一般 | 配列取得 | ゲノム配列 | BSgenome NEW

BSgenomeパッケージを用いて様々な生物種のゲノム配列を取得するやり方を示します。ミヤマハタザオ (A. lyr… セイヨウミツバチ (A. mellifera)、シロイヌナズナ(A.thaliana)、ウシ(B.taurus)、線虫(C.elegans)、ボ… (C… (G… ンパンジ… な生物… あたり… 以外に… 「ファイ…

①

10. インストール済みのヒト("BSgenome.Hsapiens.NCBI.GRCh38")のゲノム配列のmulti-FASTAファイルで保存したい場合:

一部を抽出して保存するやり方です。このパッケージ中の染色体の並びが既知(chr1, 2, ..., chr22, chrX, chrY, and MT)であるという前提です。

```
out_f <- "hoge10.fasta"                  #出力ファイル名を指定してout_fに格納
param <- "BSgenome.Hsapiens.NCBI.GRCh38"#パッケージ名を指定
param_range <- 1:25                      #抽出したい範囲を指定

#必要なパッケージのロード
library(param, character.only=T)

#前処理(paramで指定したパッケージ中…
#tmp <- unlist(strsplit(param, "…
tmp <- ls(paste("package", param,…
genome <- eval(parse(text=tmp))
genome

#本番
fasta <- getSeq(genome)
names(fasta) <- seqnames(genome)

#後処理(フィルタリング)
obj <- param_range
fasta <- fasta[obj]
fasta
```

②

1. 利用…

```
#必…
libr…

#本…
avai…

#本…
inst…

#後…
inst…
```

R Console

```
          width seq                                    names
 [1] 248956422 NNNNNNNNNNN...NNNNNNNNNN 1
 [2] 242193529 NNNNNNNNNNN...NNNNNNNNNN 2
 [3] 198295559 NNNNNNNNNNN...NNNNNNNNNN 3
 [4] 190214555 NNNNNNNNNNN...NNNNNNNNNN 4
 [5] 181538259 NNNNNNNNNNN...NNNNNNNNNN 5
 ...       ... ...
[21]  46709983 NNNNNNNNNNN...NNNNNNNNNN 21
[22]  50818468 NNNNNNNNNNN...NNNNNNNNNN 22
[23] 156040895 NNNNNNNNNNN...NNNNNNNNNN X
[24]  57227415 NNNNNNNNNNN...NNNNNNNNNN Y
[25]     16569 GATCACAGGTC...CATCACGATG MT
>
> #ファイルに保存
> writeXStringSet(fasta, file=out_f, format="fasta", width=$
>
```

BSgenome

例題9。26番目以降の配列は、ヒトゲノムの一部ではあるものの、まだ割り当てられる染色体が定まっていないものたちです。メタゲノム解析などでヒトゲノムにマップされないリードのみ取扱いたい場合には、利用可能な全配列をマッピング時のリファレンスとして用いるのが自然だと思います。

9. インストール済みのヒト("BSgenome.Hsapiens.NCBI.GRCh

2013年12月にリリースされたGenome Reference Consortium
実行可能です。

```
out_f <- "hoge9.fasta"                    #出力ファイル名を指定してout_fに格納
param <- "BSgenome.Hsapiens.NCBI.GRCh38"#パッケージ名を指定

#必要なパッケージをロード
library(param, character.only=T)

#前処理(paramで指定したパッケージ中のオ
#tmp <- unlist(strsplit(param, "."),
tmp <- ls(paste("package", param, sep
genome <- eval(parse(text=tmp))
genome

#本番
fasta <- getSeq(genome)
names(fasta) <- seqnames(genome)

#ファイルに保存
writeXStringSet(fasta, file=out_f, fo
```

R Console

```
> #ファイルに保存
> writeXStringSet(fasta, file=out_f, format="fasta", widt$
> fasta
  A DNAStringSet instance of length 455
          width seq                    names
  [1] 248956422 NNNNNNNNN...NNNNNNNNN 1
  [2] 242193529 NNNNNNNNN...NNNNNNNNN 2
  [3] 198295559 NNNNNNNNN...NNNNNNNNN 3
  [4] 190214555 NNNNNNNNN...NNNNNNNNN 4
  [5] 181538259 NNNNNNNNN...NNNNNNNNN 5
  ...       ... ...
[451]    200773 TCTACTCTC...GGGGAATTC HSCHR19KIR_FH08_B...
[452]    170148 TTTCTTTCT...GGGGAATTC HSCHR19KIR_FH13_A...
[453]    215732 TGTGGTGAG...GGGGAATTC HSCHR19KIR_FH13_B...
[454]    170537 TCTACTCTC...GGGGAATTC HSCHR19KIR_FH15_A...
[455]    177381 GATCTATCT...GGGGAATTC HSCHR19KIR_RP5_B_...
>
```

Contents

- パッケージ
  - CRANとBioconductor
  - 推奨パッケージインストール手順のおさらい
  - Bioconductor概観 → ゲノム配列パッケージ(BSgenome)
  - ヒトゲノム情報パッケージの解析
- プロモーター配列取得(sessionInfo、バージョンの違い)
  - Rパッケージ(ゲノムとアノテーションパッケージの併用)
  - FASTA形式ファイルとGFF3形式ファイル
  - データの型
- FASTQファイルの各種解析(LinuxとRを相補的に活用)
  - Linux (FastQC)とR (ShortRead)で同じ:Overrepresented sequences項目
  - Linux (FastQC)とR (QuasR)で異なる見栄え:Per base sequence content項目。FastQCに--nogroupというオプションがあることを知る。
  - FastQCのオプション (デフォルトと--nogroupあり)の違いによるKmer Content項目の結果の違い → Rで検証

プロモーター配列取得

①ゲノム配列と②アノテーション情報があれば、任意の領域の配列を取り出すことができます。

イントロ | 一般 | 配列取得 | プロモーター配列 | BSgenomeとTxDbから

NEW

①

②

ゲノム配列(BSgenome)パッケージとアノテーション情報(TxDb)パッケージを用いて様々な生物種のプロモーター配列(転写開始点近傍配列;上流配列)を取得するやり方を示します。2014年4月リリースのBioconductor 2.14以降の推奨手順では、ゲノムのパッケージ(例:BSgenome.Hsapiens.UCSC.hg19)と対応するアノテーションパッケージ(例:TxDb.Hsapiens.UCSC.hg19.knownGene)の両方を読み込ませる必要がありますので、2015年2月に記述内容を大幅に変更しました。ヒトなどの主要なパッケージ以外はおそらくデフォルトではインストールされていませんので、「パッケージがインストールされていません」的なエラーが出た場合は、個別パッケージのインストールを参考にして予め利用したいパッケージのインストールを行ってから再挑戦してください。出力はmulti-FASTAファイルです。現状では、ゼブラフィッシュ(danRer7)はゲノムパッケージ(BSgenome.Drerio.UCSC.danRer7)は存在しますが、対応するTxDbパッケージが存在しないので、どこかからGFFファイルを取得してmakeTranscriptDbFromGFF関数などを利用してTranscriptDbオブジェクトを得るなどする必要があります。

プロモーター配列取得

例題1。作業ディレクトリはどこでもよいが基本はデスクトップ上のhoge。転写開始点上流1000塩基を取得したい場合。width列は配列長に相当。出力ファイルを見なくてもうまくいっているだろうと思える。

1. ヒト(hg19)の場合:

ゲノムパッケージ(BSgenome.Hsapiens.UCSC.hg19)と対応するアノテーションパッケージ(TxDb.Hsapiens.UCSC.hg19.knownGene)を読み込んで、転写開始点上流1000塩基分を取得する

```
out_f <- "hoge1.fasta"                          #出力ファイル名を指定してout_fに格納
param_bsgenome <- "BSgenome.Hsapiens.UCSC.hg19"#パッケージ名を指定(BSgenome系のゲノムパ
param_txdb <- "TxDb.Hsapiens.UCSC.hg19.knownGene"#パッケージ名を指定(TxDB系のアノテーシ
param_upstream <- 1000                          #転写開始点上流の塩基配列数を指定

#前処理(指定したパッケージ中のオブジェクト
library(param_bsgenome, character.only
tmp <- ls(paste("package", param_bsger
genome <- eval(parse(text=tmp))

library(param_txdb, character.only=T)
tmp <- ls(paste("package", param_txdb,
txdb <- eval(parse(text=tmp))

#本番
gn <- sort(genes(txdb))
hoge <- flank(gn, width=param_upstream
fasta <- getSeq(genome, hoge)
fasta

#ファイルに保存
writeXStringSet(fasta, file=out_f, for
```

R Console

```
> fasta <- getSeq(genome, hoge)            #指定した範囲の$
> fasta                                    #確認してるだけ$
  A DNAStringSet instance of length 23056
        width seq                    names
    [1]  1000 ACACATGCTA...TTTCGTTAA 100287102
    [2]  1000 GCTATTATCA...AATAACTCT 79501
    [3]  1000 GAATTAGGCT...AAGGCGGGG 643837
    [4]  1000 CGGGGAGCCC...CTTGGGCCA 148398
    [5]  1000 CGGCGGGGCT...AGCGGCGGG 339451
    ...   ... ...
[23052]  1000 GGTGAGCCAA...ATCTCAGCC 283788
[23053]  1000 AGCCCTCCAC...CTCTCCAAC 100507412
[23054]  1000 CGGGGCCCAG...CCTGGCTGC 728410
[23055]  1000 CGGGGCCCAG...CCTGGCTGC 100653046
[23056]  1000 CAGGCTGAGC...CTCACCGCG 100288687
>
> #ファイルに保存
> writeXStringSet(fasta, file=out_f, format="fasta", wid$
>
```

プロモーター配列取得

例題3。線虫はBSgenomeとTxDbの両方が提供されています。予め2つのパッケージを個別にインストールしておけば、上流500～下流20塩基の範囲の配列を取得できる。

3. 線虫(ce6)の場合:

ゲノムパッケージ(BSgenome.Celegans.UCSC.ce6)と 対応するアノテーションパッケージ(TxDb.Celegans.UCSC.ce6.ensGene)を読み込んで、転写開始点上流500塩基から下流20塩基までの範囲を取得するやり方です。

```
out_f <- "hoge3.fasta"                    #出力ファイル名を指定してout_fに格納
param_bsgenome <- "BSgenome.Celegans.UCSC.ce6"#パッケージ名を指定(BSgenome系のゲノムパッ
param_txdb <- "TxDb.Celegans.UCSC.ce6.ensGene"#パッケージ名を指定(TxDB系のアノテーション
param_upstream <- 500
param_downstream <- 20

#前処理(指定したパッケージ中のオブジェクト
library(param_bsgenome, character.only
tmp <- ls(paste("package", param_bsger
genome <- eval(parse(text=tmp))

library(param_txdb, character.only=T)
tmp <- ls(paste("package", param_txdb,
txdb <- eval(parse(text=tmp))

#本番
gn <- sort(genes(txdb))
hoge <- promoters(gn, upstream=param_u
          downstream=param_downstream
fasta <- getSeq(genome, hoge)
fasta
```

R Console

```
> fasta <- getSeq(genome, hoge)            #指定した範囲の$
> fasta                                    #確認してるだけ$
  A DNAStringSet instance of length 27928
        width seq                          names
    [1]   520 TTGCGCGTAA...CCGCGTCAC Y74C9A.2
    [2]   520 GCAGATAATT...ACGATGGAA Y74C9A.1
    [3]   520 AGAGGAATTT...CGAGACAAA Y48G1C.12
    [4]   520 TTACCTCCAG...CCCGATCCC Y48G1C.4
    [5]   520 CTTCATTCCC...GCAGCAGCA Y48G1C.2
    ...   ... ...
[27924]   520 GGGTGTTACA...ATAAAAAAG MTCE.29
[27925]   520 ATATCTGCAG...ATAATTCAG MTCE.30
[27926]   520 GTAGTATTTT...TTATATTAA MTCE.32
[27927]   520 TTTATATTAT...TGTTTAAAT MTCE.33
[27928]   520 GATTAATATT...TTAATAAAA MTCE.36
>
> #ファイルに保存
> writeXStringSet(fasta, file=out_f, format="fasta", wid$
>
```

Tips: sessionInfo

自分の解析環境を知る。エラーが出てどうにもならない場合は、sessionInfo()実行結果をメール添付か本文中に記載して助けを求める。見る側は、②R本体のバージョン(この場合、3.2.0)や、パッケージが読み込まれているか、あるいはパッケージのバージョンをチェックする。

①

②

```
> sessionInfo()
R version 3.2.0 (2015-04-16)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 7 x64 (build 7601) Service Pack 1

locale:
[1] LC_COLLATE=Japanese_Japan.932  LC_CTYPE=Japanese_Japan.932
[3] LC_MONETARY=Japanese_Japan.932 LC_NUMERIC=C
[5] LC_TIME=Japanese_Japan.932

attached base packages:
[1] stats4    parallel  stats     graphics  grDevices utils     datasets  methods
[9] base

other attached packages:
 [1] TxDb.Celegans.UCSC.ce6.ensGene_3.1.2    BSgenome.Celegans.UCSC.ce6_1.4.0
 [3] BSgenome.Hsapiens.UCSC.hg19_1.4.0       BSgenome_1.36.0
 [5] rtracklayer_1.28.3                      Biostrings_2.36.1
 [7] XVector_0.8.0                           RCurl_1.95-4.6
 [9] bitops_1.0-6                            TxDb.Hsapiens.UCSC.hg19.knownGene_3.1.2
[11] GenomicFeatures_1.20.1                  AnnotationDbi_1.30.1
[13] Biobase_2.28.0                          GenomicRanges_1.20.3
[15] GenomeInfoDb_1.4.0                      IRanges_2.2.1
[17] S4Vectors_0.6.0                         BiocGenerics_0.14.0

loaded via a namespace (and not attached):
```

Tips: sessionInfo

これは、うまくプロモーター配列を取得できたときのR環境。①例題1のヒト(H. sapiens)プロモーター配列を取得したときと、②例題3の線虫(C. elegans)プロモーター配列を取得したときに読み込んだパッケージが表示されているのがわかる。

R Console

```
> sessionInfo()
R version 3.2.0 (2015-04-16)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 7 x64 (build 7601) Service Pack 1

locale:
[1] LC_COLLATE=Japanese_Japan.932  LC_CTYPE=Japanese_Japan.932
[3] LC_MONETARY=Japanese_Japan.932 LC_NUMERIC=C
[5] LC_TIME=Japanese_Japan.932

attached base packages:
[1] stats4    parallel  stats     graphics  grDevices utils     datasets  methods
[9] base

other attached packages:
 [1] TxDb.Celegans.UCSC.ce6.ensGene_3.1.2    BSgenome.Celegans.UCSC.ce6_1.4.0
 [3] BSgenome.Hsapiens.UCSC.hg19_1.4.0       BSgenome_1.36.0
 [5] rtracklayer_1.28.3                      Biostrings_2.36.1
 [7] XVector_0.8.0                           RCurl_1.95-4.6
 [9] bitops_1.0-6                            TxDb.Hsapiens.UCSC.hg19.knownGene_3.1.2
[11] GenomicFeatures_1.20.1                  AnnotationDbi_1.30.1
[13] Biobase_2.28.0                          GenomicRanges_1.20.3
[15] GenomeInfoDb_1.4.0                      IRanges_2.2.1
[17] S4Vectors_0.6.0                         BiocGenerics_0.14.0

loaded via a namespace (and not attached):
```

パッケージのインストール

「必要最小限プラスアルファ」の推奨インストール手順通りにインストールしたヒトの多くはうまくいっている。

- インストール | R本体 | 過去版 | Mac用 (last modified 2015/03/22)
- インストール | Rパッケージ | ほぼ全て(20GB以上?!) (last modified 2015/05/25)
- インストール | Rパッケージ | 必要最小限プラスアルファ(数GB?!) (last modified 2015/07/24)推奨
- インストール | Rパッケージ | 必要最小限プラスアルファ(アグリバイオ居室のみ) (last modified 20
- インストール | Rパッケージ | 必要最小限(数GB?!) (last modified 2015/05/25)
- インストール
- (削除予定)R
- (削除予定)
- 基本的な利

インストール | Rパッケージ | 必要最小限プラスアルファ(数GB?!) NEW

(Rで)塩基配列解析、(Rで)マイクロアレイデータ解析中で利用するパッケージ、プラスアルファのパッケージをインストールするやり方です。Rパッケージの2大リポジトリであるCRANとBioconductor から提供されているパッケージ群のうち、一部のインストールに相当しますので、相当短時間でインストールが完了します。
SAFEではなくsafeパッケージの間違い
(_ _)m limmaの取得先が間違ってCRA
追加しました。

1. R本体を起動

2. CRANから提供されているパッケー

以下を「R コンソール画面上」でコピー
合は自分のいる場所から近いサイトを

```
#(Rで)塩基配列解析で主に利用
install.packages("PoissonSe
install.packages("samr")
install.packages("seqinr")

#(Rで)マイクロアレイデータ解析
```

```
biocLite("SRAdb", suppressUpdates=TRUE)
biocLite("TCC", suppressUpdates=TRUE)  #(Rで)マイクロアレイデータ解析でも利用
biocLite("TxDb.Celegans.UCSC.ce6.ensGene", suppressUpdates=TRUE)
biocLite("TxDb.Hsapiens.UCSC.hg19.knownGene", suppressUpdates=TRUE)
biocLite("TxDb.Hsapiens.UCSC.hg38.knownGene", suppressUpdates=TRUE)
biocLite("TxDb.Mmusculus.UCSC.mm10.knownGene", suppressUpdates=TRUE)
biocLite("TxDb.Rnorvegicus.UCSC.rn5.refGene", suppressUpdates=TRUE)
#biocLite("yeastRNASeq", suppressUpdates=TRUE)

#(Rで)マイクロアレイデータ解析で利用
biocLite("affy", suppressUpdates=TRUE)
biocLite("agilp", suppressUpdates=TRUE)
```

4. Bioconductorから提供されているパッケージ群のインストール

ゲノム配列パッケージです。一つ一つの容量が尋常でないため、必要に応じてテキストエディタなどに予めコピペしておき、いらないゲノムパッケージを削除してからお使いください。

```
source("http://bioconductor.org/biocLite.R")#おまじない
biocLite("BSgenome.Athaliana.TAIR.TA", suppressUpdates=TRUE)#シロイヌナズナゲノ
biocLite("BSgenome.Celegans.UCSC.ce6", suppressUpdates=TRUE)#線虫ゲノム
biocLite("BSgenome.Drerio.UCSC.danRer7", suppressUpdates=TRUE)#ゼブラフィッシュゲノ
biocLite("BSgenome.Hsapiens.NCBI.GRCh38", suppressUpdates=TRUE)#ヒトゲノム(GRCh38)
biocLite("BSgenome.Hsapiens.UCSC.hg19", suppressUpdates=TRUE)#ヒトゲノム(hg19)
biocLite("BSgenome.Mmusculus.UCSC.mm10", suppressUpdates=TRUE)#マウスゲノム(mm10)
```

空白部分は後に埋まっていきます

全体像

リポジトリ(パッケージ提供元)	パッケージ名	R本体	
		R ver. 3.1.3	R ver. 3.2.0
Bioconductor	TxDb.Celegans.UCSC.ce6.ensGene		3.1.2
Bioconductor	BSgenome.Celegans.UCSC.ce6		1.4.0
CRAN	XML		
CRAN	RCurl		1.95-4.6
CRAN	checkmate		-

R ver. 3.1.3でこけた

①例題3を門田のR ver. 3.1.3でやった結果。②XMLおよび③RCurlパッケージがないためにエラー祭りとなった(爆)

R Console

```
> out_f <- "hoge3.fasta"                          #出力ファイル名を指定してo$
> param_bsgenome <- "BSgenome.Celegans.UCSC.ce6"#パッケージ名を指定$
> param_txdb <- "TxDb.Celegans.UCSC.ce6.ensGene"ケージ名を指定$
> param_upstream <- 500                           #転写開始点上流の塩基配列$
> param_downstream <- 20                          #転写開始点下流の塩基配列$
>
> #前処理(指定したパッケージ中のオブジェクト名をgenomeおよびtxdbに$
> library(param_bsgenome, character.only=T)#指定したパッケージの読$
 要求されたパッケージ BSgenome をロード中です
 要求されたパッケージ rtracklayer をロード中です
Error in loadNamespace(j <- i[[L]], c(lib.loc, .libPaths()), versi$
    'XML' という名前のパッケージはありません
 エラー:  パッケージ 'rtracklayer' をロードできませんでした
> tmp <- ls(paste("package", param_bsgenome, sep=":"))#指定したパッ$
 以下にエラー as.environment(pos) :
    検索リストに "package:BSgenome.Celegans.UCSC.ce6" という項目はあ$
> genome <- eval(parse(text=tmp))                 #文字列tmpをRオブジェクト$
 以下にエラー eval(parse(text = tmp)) :
    引数 'expr' の評価中にエラーが起きました (関数 'eval' に対するメ$

>
> library(param_txdb, character.only=T)   #指定したパッケージの読み$
 要求されたパッケージ GenomicFeatures をロード中です
Error in loadNamespace(j <- i[[1L]], c(lib.loc, .libPaths()), versi$
    'RCurl' という名前のパッケージはありません
 エラー:  パッケージ 'GenomicFeatures' をロードできませんでした
```

XMLのインストール

①XMLパッケージのインストール。XML自体はCRANから提供されているが、CRANパッケージ用のinstall.packages関数ではなく、Bioconductor用のbiocLite関数でもCRANパッケージのXMLをインストール可能であるというTipsを兼ねて紹介。

R Console

```
> source("http://bioconductor.org/biocLite.R")#
Bioconductor version 3.0 (BiocInstaller 1.16.5), ?biocLite for help
A new version of Bioconductor is available after installing the
  most recent version of R; see http://bioconductor.org/install
> biocLite("XML", suppressUpdates=TRUE)
BioC_mirror: http://bioconductor.org
Using Bioconductor version 3.0 (BiocInstaller 1.16.5), R version
  3.1.3.
Installing package(s) 'XML'
 URL 'http://cran.fhcrc.org/bin/windows/contrib/3.1/XML_3.98-1.3.z$
Content type 'application/zip' length 4293309 bytes (4.1 MB)
 開かれた URL
downloaded 4.1 MB

package 'XML' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\kadota\AppData\Local\Temp\RtmpER54eM\downloaded_p$
>
```

RCurlのインストール

①RCurlパッケージのインストール。
RCurlもCRANから提供されているが、
biocLite関数でインストール可能。

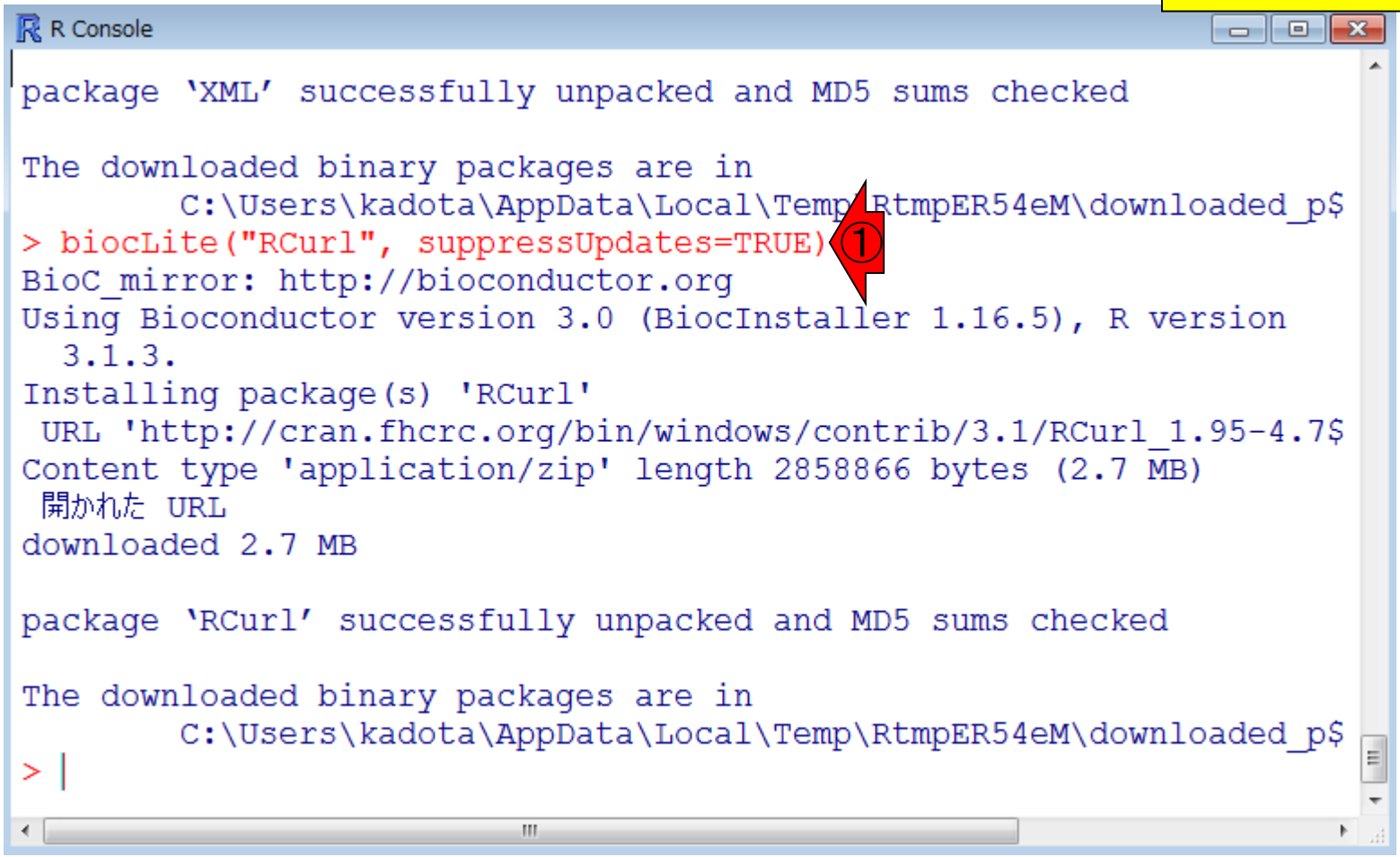
R Console

```
> source("http://bioconductor.org/biocLite.R")#おまじない
Bioconductor version 3.0 (BiocInstaller 1.16.5), ?biocLite for help
A new version of Bioconductor is available after installing the
  most recent version of R; see http://bioconductor.org/install
> biocLite("XML", suppressUpdates=TRUE)
BioC_mirror: http://bioconductor.org
Using Bioconductor version 3.0 (BiocInstaller 1.16.5), R version
  3.1.3.
Installing package(s) 'XML'
 URL 'http://cran.fhcrc.org/bin/windows/contrib/3.1/XML_3.98-1.3.z$
Content type 'application/zip' length 4293309 bytes (4.1 MB)
 開かれた URL
downloaded 4.1 MB

package 'XML' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\kadota\AppData\Local\Temp\RtmpER54eM\downloaded_p$
> biocLite("RCurl", suppressUpdates=TRUE)
```

①

RCurlのインストール

①RCurlパッケージのインストール。
RCurlもCRANから提供されているが、
biocLite関数でインストール可能。



R ver. 3.1.3で再挑戦

意気揚々と、再度例題3をコピペ。しかし①今度はcheckmateというパッケージがないからダメだと言われていることがわかる。

R Console

```
> out_f <- "hoge3.fasta"                          #出力ファイル名を指定して$
> param_bsgenome <- "BSgenome.Celegans.UCSC.ce6"#パッケージ名を指$
> param_txdb <- "TxDb.Celegans.UCSC.ce6.ensGene"#パッケージ名を指$
> param_upstream <- 500                           #転写開始点上流の塩基配列$
> param_downstream <- 20                          #転写開始点下流の塩基配列$
>
> #前処理(指定したパッケージ中のオブジェクト名をgenomeおよびtxdbに$
> library(param_bsgenome, character.only=T)#指定したパッケージの読$
 要求されたパッケージ BSgenome をロード中です
 要求されたパッケージ rtracklayer をロード中です
Error in loadNamespace(i, c(lib.loc, .libPaths()), versionCheck = $
    'checkmate' という名前のパッケージはありません
 エラー:  パッケージ 'rtracklayer' をロードできませんでした
> tmp <- ls(paste("package", param_bsgenome, sep=":"))#指定したパ$
 以下にエラー as.environment(pos) :
    検索リストに "package:BSgenome.Celegans.UCSC.ce6" という項目は$
> genome <- eval(parse(text=tmp))                 #文字列tmpをRオブジェクト$
 以下にエラー eval(parse(text = tmp)) :
    引数 'expr' の評価中にエラーが起きました (関数 'eval' に対する$

>
> library(param_txdb, character.only=T)   #指定したパッケージの読み$
 要求されたパッケージ GenomicFeatures をロード中です
Error in loadNamespace(i, c(lib.loc, .libPaths()), versionCheck = $
```

checkmateのインストール

①checkmateパッケージのインストール。
checkmateもCRANから提供されているが、biocLite関数でインストール可能。

R Console

```
> fasta <- getSeq(genome, hoge)            #指定した範囲の塩基配列情$
 以下にエラー (function (classes, fdef, mtable)  :
  unable to find an inherited method for function 'getSeq' for s$
> fasta                                    #確認してるだけです
 エラー:  オブジェクト 'fasta' がありません
>
> #ファイルに保存
> writeXStringSet(fasta, file=out_f, format="fasta", width=50)#fas$
 以下にエラー is(x, "XStringSet") :  オブジェクト 'fasta' がありま$
> biocLite("checkmate", suppressUpdates=TRUE)
BioC_mirror: http://bioconductor.org
Using Bioconductor version 3.0 (BiocInstaller 1.16.5), R version
  3.1.3.
Installing package(s) 'checkmate'
 URL 'http://cran.fhcrc.org/bin/windows/contrib/3.1/checkmate_1.6.$
Content type 'application/zip' length 393202 bytes (383 KB)
 開かれた URL
downloaded 383 KB

package 'checkmate' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\kadota\AppData\Local\Temp\RtmpER54eM\downloaded_p$
>
```

R ver. 3.1.3で再々挑戦

意気揚々と、再度例題3をコピペ。
無事パッケージのロードに成功!!

R Console

```
> out_f <- "hoge3.fasta"                          #出力ファイル名を指定して$
> param_bsgenome <- "BSgenome.Celegans.UCSC.ce6"#パッケージ名を指$
> param_txdb <- "TxDb.Celegans.UCSC.ce6.ensGene"#パッケージ名を指$
> param_upstream <- 500                           #転写開始点上流の塩基配列$
> param_downstream <- 20                          #転写開始点下流の塩基配列$
>
> #前処理(指定したパッケージ中のオブジェクト名を
> library(param_bsgenome, charac
 要求されたパッケージ BSgenome をロード中です
 要求されたパッケージ rtracklayer をロード中
> tmp <- ls(paste("package", par
> genome <- eval(parse(text=tmp)
>
> library(param_txdb, character.
 要求されたパッケージ GenomicFeatures をロ
> tmp <- ls(paste("package", par
> txdb <- eval(parse(text=tmp))
>
> #本番
> gn <- sort(genes(txdb))
> hoge <- promoters(gn, upstream=
+             downstream=param_dow
> fasta <- getSeq(genome, hoge)
> fasta
```

R Console

```
> fasta                                           #確認してるだけ$
  A DNAStringSet instance of length 27928
        width seq                                         names    $
    [1]   520 TTGCGCGTAAAATAT...TCTTGCCGCGTCAC Y74C9A.2
    [2]   520 GCAGATAATTGAGGA...TCGAAACGATGGAA Y74C9A.1
    [3]   520 AGAGGAATTTCACCG...TGCGACGAGACAAA Y48G1C.12
    [4]   520 TTACCTCCAGTGATT...AAATTCCCGATCCC Y48G1C.4
    [5]   520 CTTCATTCCCACACT...TGGAAGCAGCAGCA Y48G1C.2
    ...   ... ...
[27924]   520 GGGTGTTACACTATG...TTAGTATAAAAAAG MTCE.29
[27925]   520 ATATCTGCAGTATTT...TTAGTATAATTCAG MTCE.30
[27926]   520 GTAGTATTTTTCCAA...ATAGTTTATATTAA MTCE.32
[27927]   520 TTTATATTATTACGG...TTACTTGTTTAAAT MTCE.33
[27928]   520 GATTAATATTTTTTG...TTAGTTTAATAAAA MTCE.36
>
> #ファイルに保存
> writeXStringSet(fasta, file=out_f, format="fasta", wid$
>
```

①ただの復習ですw

この程度は…

バイオインフォマティクス人材育成カリキュラム(次世代シークエンサ) | NGSハンズオン講習会2015 NEW

NGSハンズオン講習会を2015年7月22日-8月6日の11日間で実施します。予定通り受講申込多数のため、予備日(8月26日、27日、28日)も実施することになっています。

はじめに(全

- 講習
  基本
  ズム
- 平成

2015年7月22日(2015.07.22):Bio-Linux 8とRのインストール状況確認(門田幸二、寺田 透)

書籍 | 日本乳酸菌学会誌 | について で示した通りのPC環境を構築しておきましょう。連載第1-3回、および第4回のウェブ資料W6-5までの予習は必須です。Rについても同様です。一週間程度はきっちり時間をかけて予習しておきましょう。7/22は、以下に示すようなことが**できる(わかる)ようになっていることの確認**を自分でしてもらう日です。門田自身全てを完全に把握しているわけではありませんし、ウェブ資料のページ数も膨大ですので、どこにどういうことが書かれていたかの全体像の俯瞰やチェックリストという位置づけでもあります(頻繁に更新しているのでときどきリロードしましょう):

- Bio-Linux 8
- R
  1. インストール | について をよく読み、ここに書いてある手順に従って2015年4月4日以降にインストールを行った
  2. インストール | について で書いてある内容は**Bio-Linux8(ゲストOS)とは無関係であり、WindowsやMacintosh(つまりホストOS)上で行う作業**である
  3. ファイルの拡張子(.txtや.docxなど)はちゃんと表示されている
  4. Rの起動と終了ができる。終了時に表示されるメッセージにうろたえない
  5. **R本体だけでなくRパッケージ群のインストールもちゃんと行った**
  6. **Rパッケージ群のインストール確認も行い、エラーが出ないことを確認した**
  7. ① library関数を用いたRパッケージのロード中に、別のパッケージがないことに起因するエラーメッセージが出ることもあるが、必要なパッケージを個別にインストールするやり方を知っている
  8. 基本的な利用法をよく読み、予習を行った
  9. 作業ディレクトリの変更ができる
  10. 例題ファイルのダウンロード時に、拡張子が勝手に変わることがあるので注意する
  11. 慣れないうちは、getwd()とlist.files()を打ち込むことで、作業ディレクトリと入力ファイルの存在確認を行う
  12. エラーに遭遇した際、「ありがちなミス1-4」に当てはまっていないかどうか自分で確認

sessionInfo (R 3.1.3)

R ver. 3.1.3での、ロードされているパッケージ群のバージョン情報に注目!

R Console

①

②

```
> sessionInfo()
R version 3.1.3 (2015-03-09)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 7 x64 (build 7601) Service Pack 1

locale:
[1] LC_COLLATE=Japanese_Japan.932  LC_CTYPE=Japanese_Japan.932
[3] LC_MONETARY=Japanese_Japan.932 LC_NUMERIC=C
[5] LC_TIME=Japanese_Japan.932

attached base packages:
[1] stats4    parallel  stats     graphics  grDevices utils     datasets  methods
[9] base

other attached packages:
 [1] TxDb.Celegans.UCSC.ce6.ensGene_3.0.0 GenomicFeatures_1.18.7
 [3] BSgenome.Celegans.UCSC.ce6_1.4.0     BSgenome_1.34.1
 [5] rtracklayer_1.26.3                   BiocInstaller_1.16.5
 [7] Biostrings_2.34.1                    XVector_0.6.0
 [9] AnnotationDbi_1.28.2                 Biobase_2.26.0
[11] GenomicRanges_1.18.4                 GenomeInfoDb_1.2.4
[13] IRanges_2.0.1                        S4Vectors_0.4.0
[15] BiocGenerics_0.12.1

loaded via a namespace (and not attached):
```

sessionInfo (R 3.1.3)

特に①～③のパッケージは、④2015年7月24日にインストールしたばかりであり、前述のR ver. 3.2.0実行結果よりも後にインストールしたものである。

R Console

```
other attached packages:
 [1] TxDb.Celegans.UCSC.ce6.ensGene_3.0.0 GenomicFeatures_1.18.7
 [3] BSgenome.Celegans.UCSC.ce6_1.4.0     BSgenome_1.34.1
 [5] rtracklayer_1.26.3                   BiocInstaller_1.16.5
 [7] Biostrings_2.34.1                    XVector_0.6.0
 [9] AnnotationDbi_1.28.2                 Biobase_2.26.0
[11] GenomicRanges_1.18.4                 GenomeInfoDb_1.2.4
[13] IRanges_2.0.1                        S4Vectors_0.4.0
[15] BiocGenerics_0.12.1

loaded via a namespace (and not attached):
 [1] base64enc_0.1-2          BatchJobs_1.6            BBmisc_1.9
 [4] BiocParallel_1.0.3       biomaRt_2.22.0           bitops_1.0-6
 [7] brew_1.0-6               checkmate_1.6.1          codetools_0.2-11
[10] crayon_1.3.0             DBI_0.3.1                digest_0.6.8
[13] fail_1.2                 foreach_1.4.2            GenomicAlignments_1.2.2
[16] iterators_1.0.7          magrittr_1.5             memoise_0.2.1
[19] RCurl_1.95-4.7           Rsamtools_1.18.3         RSQLite_1.0.0
[22] sendmailR_1.2-1          stringi_0.4-1            stringr_1.0.0
[25] testthat_0.10.0          tools_3.1.3              XML_3.98-1.3
[28] zlibbioc_1.12.0
> date()
[1] "Fri Jul 24 21:21:59 2015"
>
```

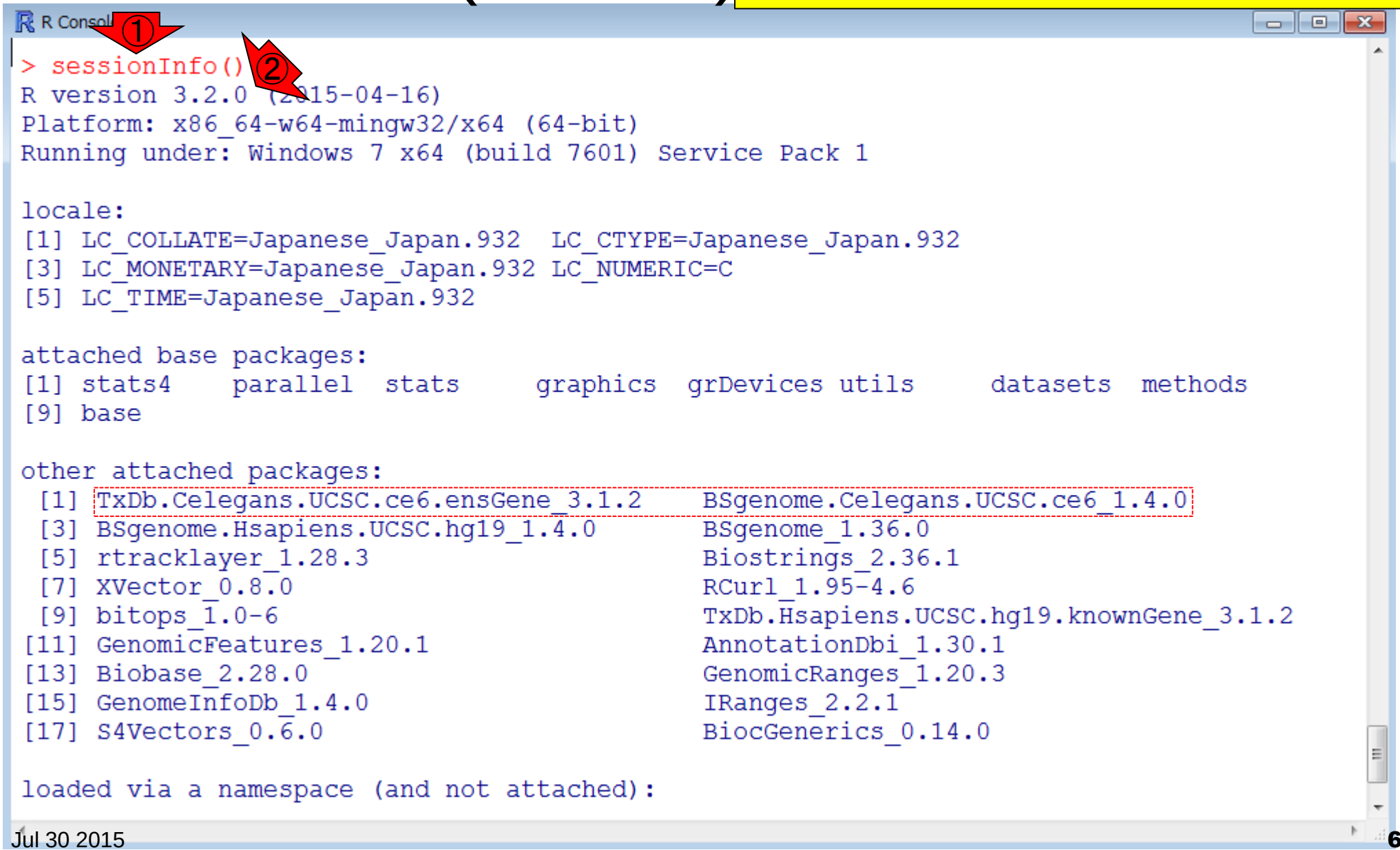
ほぼ埋まった

全体像

リポジトリ(パッケージ提供元)	パッケージ名	R本体	
		R ver. 3.1.3	R ver. 3.2.0
Bioconductor	TxDb.Celegans.UCSC.ce6.ensGene	3.0.0	3.1.2
Bioconductor	BSgenome.Celegans.UCSC.ce6	1.4.0	1.4.0
CRAN	XML	3.98-1.3	
CRAN	RCurl	1.95-4.7	1.95-4.6
CRAN	checkmate	1.6.1	-

sessionInfo (R 3.2.0)

BSgenomeパッケージはver. 1.4.0で同じ。しかしTxDbパッケージは、ver. 3.1.2 (R ver. 3.2.0)とver. 3.0.0 (R ver. 3.1.3)で異なっていることがわかる。



sessionInfo (R 3.2.0)

①XML、および②RCurlのバージョン。
checkmateパッケージは、存在の有無は不明であるが、少なくともロードされておらず、R ver. 3.2.0では必要とされていないようだ。

R Console

```
other attached packages:
 [1] TxDb.Celegans.UCSC.ce6.ensGene_3.1.2    BSgenome.Celegans.UCSC.ce6_1.4.0
 [3] BSgenome.Hsapiens.UCSC.hg19_1.4.0        BSgenome_1.36.0
 [5] rtracklayer_1.28.3                       Biostrings_2.36.1
 [7] XVector_0.8.0                            RCurl_1.95-4.6
 [9] bitops_1.0-6                             TxDb.Hsapiens.UCSC.hg19.knownGene_3.1.2
[11] GenomicFeatures_1.20.1                   AnnotationDbi_1.30.1
[13] Biobase_2.28.0                           GenomicRanges_1.20.3
[15] GenomeInfoDb_1.4.0                       IRanges_2.2.1
[17] S4Vectors_0.6.0                          BiocGenerics_0.14.0

loaded via a namespace (and not attached):
 [1] zlibbioc_1.14.0         GenomicAlignments_1.4.1 BiocParallel_1.2.1
 [4] tools_3.2.0             DBI_0.3.1               lambda.r_1.1.7
 [7] futile.logger_1.4.1     futile.options_1.0.0    biomaRt_2.24.0
[10] RSQLite_1.0.0           Rsamtools_1.20.2        XML_3.98-1.1
> |
```

②

①

全体像を把握

①赤枠がごく最近(2015年7月24日)インストールしたCRAN提供パッケージたち。バージョン番号がR ver. 3.2.0のものに比べて大きいので、確かに最新版なのだろう。

リポジトリ(パッケージ提供元)	パッケージ名	R本体	
		R ver. 3.1.3	R ver. 3.2.0
Bioconductor	TxDb.Celegans.UCSC.ce6.ensGene	3.0.0	3.1.2
Bioconductor	BSgenome.Celegans.UCSC.ce6	1.4.0	1.4.0
CRAN	XML	3.98-1.3	3.98-1.1
CRAN	RCurl	1.95-4.7	1.95-4.6
CRAN	checkmate	1.6.1	-

全体像を把握

これらの結果から、2つの結論が導き出される。①CRAN提供パッケージは、R本体のバージョンに関係なく、常に最新版がインストールされる。②R本体のバージョンが上がると必要とされないパッケージが出ることもある(checkmate)

リポジトリ(パッケージ提供元)	パッケージ名	R本体	
		R ver. 3.1.3	R ver. 3.2.0
Bioconductor	TxDb.Celegans.UCSC.ce6.ensGene	3.0.0	3.1.2
Bioconductor	BSgenome.Celegans.UCSC.ce6	1.4.0	1.4.0
CRAN	XML	3.98-1.3	3.98-1.1
CRAN	RCurl	1.95-4.7	1.95-4.6
CRAN	checkmate	1.6.1	-

①

②

R ver. 3.1.3で再挑戦

ここで、もう一度「R ver. 3.1.3で再挑戦」したときのエラーメッセージを読み返す。①
黒枠は、BSgenome.Celegans.UCSC.ce6 パッケージのロード部分。

R Console

```
> out_f <- "hoge3.fasta"                          #出力ファイル名を指定して$
> param_bsgenome <- "BSgenome.Celegans.UCSC.ce6"#パッケージ名を指$
> param_txdb <- "TxDb.Celegans.UCSC.ce6.ensGene"#パッケージ名を指$
> param_upstream <- 500                           #転写開始点上流の塩基配列$
> param_downstream <- 20                          #転写開始点下流の塩基配列$
>
> #前処理(指定したパッケージ中のオブジェクト名をgenomeおよびtxdbに$
> library(param_bsgenome, character.only=T)#指定したパッケージの読$
 要求されたパッケージ BSgenome をロード中です
 要求されたパッケージ rtracklayer をロード中です
Error in loadNamespace(i, c(lib.loc, .libPaths()), versionCheck = $
    'checkmate' という名前のパッケージはありません
 エラー:  パッケージ 'rtracklayer' をロードできませんでした
> tmp <- ls(paste("package", param_bsgenome, sep=":"))#指定したパ$
 以下にエラー as.environment(pos) :
    検索リストに "package:BSgenome.Celegans.UCSC.ce6" という項目は$
> genome <- eval(parse(text=tmp))                 #文字列tmpをRオブジェクト$
 以下にエラー eval(parse(text = tmp)) :
    引数 'expr' の評価中にエラーが起きました (関数 'eval' に対する$

>
> library(param_txdb, character.only=T)   #指定したパッケージの読み$
 要求されたパッケージ GenomicFeatures をロード中です
Error in loadNamespace(i, c(lib.loc, .libPaths()), versionCheck = $
```

R ver. 3.1.3で再挑戦

文脈から、以下のように理解する。①BSgenome.Celegans.UCSC.ce6は内部的にrtracklayerを利用しているため、rtracklayerもロードしようとしている。しかし、②rtracklayerは内部的にcheckmateを利用している。このためcheckmateもロードしようとしたが、ckeckmateが未インストールのためこけた、というのが事の顛末。

R Console

```
> out_f <- "hoge3.fasta"                          #出力ファイル名を
> param_bsgenome <- "BSgenome.Celegans.UCSC.ce6"#パッケ
> param_txdb <- "TxDb.Celegans.UCSC.ce6.ensGene"#パッケ
> param_upstream <- 500                           #転写開始点上流
> param_downstream <- 20                          #転写開始点下流
>
> #前処理(指定したパッケージ中のオブジェクト名をgenomeおよびtxdbに$
> library(param_bsgenome, character.only=T)#指定したパッケージの読$
 要求されたパッケージ BSgenome をロード中です
 要求されたパッケージ rtracklayer をロード中です
Error in loadNamespace(i, c(lib.loc, .libPaths()), versionCheck = $
   'checkmate' という名前のパッケージはありません
エラー:  パッケージ 'rtracklayer' をロードできませんでした
> tmp <- ls(paste("package", param_bsgenome, sep=":"))#指定したパ$
 以下にエラー as.environment(pos) :
   検索リストに "package:BSgenome.Celegans.UCSC.ce6" という項目は$
> genome <- eval(parse(text=tmp))                 #文字列tmpをRオブジェクト$
 以下にエラー eval(parse(text = tmp)) :
   引数 'expr' の評価中にエラーが起きました (関数 'eval' に対する$

>
> library(param_txdb, character.only=T)   #指定したパッケージの読み$
 要求されたパッケージ GenomicFeatures をロード中です
Error in loadNamespace(i, c(lib.loc, .libPaths()), versionCheck = $
```

依存関係を含めると…

複雑。rtracklayer ver. 1.28.3ではcheckmate パッケージを内部的に使わなくなったのだろう、と理解する。たとえBsgenomeとTxDbのバージョンがR本体の間で不変であったとしても、内部的に用いるパッケージのバージョンや、それに起因する実行結果の違いもありうる。

リポジトリ(パッケージ提供元)	パッケージ名	R本体 R ver. 3.1.3	R本体 R ver. 3.2.0
Bioconductor	TxDb.Celegans.UCSC.ce6.ensGene	3.0.0	3.1.2
CRAN	RCurl	1.95-4.7	1.95-4.6
Bioconductor	BSgenome.Celegans.UCSC.ce6	1.4.0	1.4.0
Bioconductor	rtracklayer	1.26.3	1.28.3 ①
CRAN	XML	3.98-1.3	3.98-1.1
CRAN	checkmate	1.6.1	-

現実的な対策

私は少しの結果の違いは（突き詰めていっても不毛なので）気にしない。論文中に記載するのはR本体と主要なパッケージ（この場合BSgenomeとTxDb）のみが基本。

リポジトリ（パッケージ提供元）	パッケージ名	R本体	
		R ver. 3.1.3	R ver. 3.2.0
Bioconductor	TxDb.Celegans.UCSC.ce6.ensGene	3.0.0	3.1.2
CRAN	RCurl	1.95-4.7	1.95-4.6
Bioconductor	BSgenome.Celegans.UCSC.ce6	1.4.0	1.4.0
Bioconductor	rtracklayer	1.26.3	1.28.3
CRAN	XML	3.98-1.3	3.98-1.1
CRAN	checkmate	1.6.1	-

R本体のバージョンは重要

- 近年のリリース頻度
  - R本体 (http://www.r-project.org/)
    - 2015-06-18にver. 3.2.1をリリース
    - 2015-04-16にver. 3.2.0をリリース
    - 2015-03-09にver. 3.1.3をリリース
    - 2014-10-31にver. 3.1.2をリリース
    - …
    - 2012-03-30にver. 2.15.0をリリース
    - …
  - Bioconductor (http://bioconductor.org/)は半年ごとにリリース
    - 2015-04にver. 3.1をリリース (R ver. 3.2.1で動作確認)、提供パッケージ数:1,024
    - 2014-10にver. 3.0をリリース (R ver. 3.1.1で動作確認)、提供パッケージ数:934 ①
    - 2014-04にver. 2.14をリリース (R ver. 3.1.0で動作確認)、提供パッケージ数:824
    - 2013-10にver. 2.13をリリース (R ver. 3.0で動作確認)、提供パッケージ数:750
    - 2013-04にver. 2.12をリリース (R ver. 3.0で動作確認)、提供パッケージ数:672
    - 2012-10にver. 2.11をリリース (R ver. 2.15.1で動作確認)、提供パッケージ数:608
    - 2012-04にver. 2.10をリリース (R ver. 2.15.0で動作確認)、提供パッケージ数:553
    - …

R本体のバージョンがわかれば、Bioconductor提供パッケージのバージョンも概ね定まる。例えば、①R ver. 3.1.2では、どのタイミングで(例えば2015年8月に)パッケージのインストールを行おうが、Bioconductor ver. 3.0で提供されているパッケージのバージョンとなる。決して2015年4月に公開されたBioconductor ver. 3.1で提供されているパッケージがインストールされることはない。

Contents

- パッケージ
  - CRANとBioconductor
  - 推奨パッケージインストール手順のおさらい
  - Bioconductor概観 → ゲノム配列パッケージ(BSgenome)
  - ヒトゲノム情報パッケージの解析
- プロモーター配列取得(sessionInfo、バージョンの違い)
  - Rパッケージ(ゲノムとアノテーションパッケージの併用)
  - FASTA形式ファイルとGFF3形式ファイル
  - データの型
- FASTQファイルの各種解析(LinuxとRを相補的に活用)
  - Linux (FastQC)とR (ShortRead)で同じ:Overrepresented sequences項目
  - Linux (FastQC)とR (QuasR)で異なる見栄え:Per base sequence content項目。FastQCに--nogroupというオプションがあることを知る。
  - FastQCのオプション (デフォルトと--nogroupあり)の違いによるKmer Content項目の結果の違い → Rで検証

プロモーター配列取得

multi-FASTA形式のゲノム配列ファイルとGFF3形式などの一般的なアノテーションファイルがあれば、それを利用することもできる。②例題7の乳酸菌のプロモーター配列取得を示します。

- イントロ | 一般 | Tips | 拡張子は同じで任意の文字を追加して保存 (last modified 2013/09/26)
- イントロ | 一般 | 配列取得 | ゲノム配列 | 公共DBから (last modified 2014/05/28)
- イントロ | 一般 | 配列取得 | ゲノム配列 | BSgenome (last modified 2015/02/19) NEW
- イントロ | 一般 | 配列取得 | プロモーター配列 | 公共DBから (last modified 2014/04/02)
- イントロ | 一般 | 配列取得 | プロモーター配列 | BSgenomeとTxDbから (last modified 2015/02/20) NEW
- イントロ | 一般 | 配列取得 | プロモーター配列 | GenomicFeatures(Lawrence_2013) (last modified 2015/02/20)
- イントロ | 一般 | 配列取得 | トランスクリプトーム配列 | 公共DBから (last modified 2014/04/02)

①

イントロ | 一般 | 配列取得 | プロモーター配列 | GenomicFeatures(Lawrence_2013) NEW

GenomicFeaturesパッケージを主に用いてプロモーター配列(転写開始点近傍配列)を得るやり方を示します。
ここでは、イントロ | 一般 | 配列取得 | ゲノム配列(BSgenomeから)で指定可能なゲノムとイントロ | NGS | アノテーション情報取得 | TranscriptDb | GenomicFeatures(Lawrence_2013)で作成可能なTranscriptDbオブジェクトを入力として、getPromoterSeqという関数を用いて、転写開始点から任意の[上流xxx塩基, 下流yyy塩基]分の塩基配列を取得して、FASTA形式ファイルで保存するやり方を示しています。multi-FASTA形式のゲノム配列ファイルとGFF3形式のアノテーションファイルのみからプロモータ配列を取得するやり方も示しています。

②

7. 乳酸菌ゲノム(Lactobacillus casei 12A)の[上流500塩基, 下流20塩基]のプロモーター配列を取得する場合:

Ensembl (Flicek et al., Nucleic Acids Res., 2014)から提供されている Lactobacillus casei 12Aの multi-FASTA形式ゲノム配列ファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa)と GFF3形式のアノテーションファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3) を読み込むやり方です。

```
in_f1 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa"#入力ファイル
in_f2 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3"#入力ファイル名
out_f <- "hoge7.fasta"                    #出力ファイル名を指定してout_fに格納
param_upstream <- 500                     #転写開始点上流の塩基配列数を指定
param_downstream <- 20                    #転写開始点下流の塩基配列数を指定

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

プロモーター配列取得

2015年3月5-6日のHPCIハンズオン講習会時のスライド(2015.03.05)です。このときはR ver. 3.1.2でうまくいった。

7. 乳酸菌ゲノム(Lactobacillus casei 12A)の[上流500塩基, 下流20塩基]のプロモーター配列を取得する場合:

Ensembl (Flicek et al., Nucleic Acids Res., 2014)から提供されている Lactobacillus casei 12Aの multi-FASTA形式ゲノム配列ファイル (Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3)と GFF3形式のアノテーションファイル (Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa) を読み込むやり方です。
makeTranscriptDbFromGFF関数実行時に用いているuseGenesAsTranscripts=Tは、原核生物(prokaryotes)に代表される「転写物＝遺伝子」の場合に指定します。デフォルトはFです。

```
in_f1 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa"#入力ファイル名を指
in_f2 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3"#入力ファイル名を指定
out_f <- "hoge7.fasta"                    #出力ファイル名を指定してout_fに格納
param_upstream <- 500                     #転写開始点上流の塩基配列数を指定
param_downstream <- 20                    #転写開始点下流の塩基配列数を指定

#必要なパッケージをロード
library(Rsamtools)                        #パッケージの
library(Biostrings)                       #パッケージの
library(GenomicFeatures)                  #パッケージの

#前処理(アノテーション情報を取得)
txdb <- makeTranscriptDbFromGFF(in_f2, #TranscriptD
          format="gff3", useGenesAsTranscripts=T)#
txdb                                      #確認してるだ

#前処理(欲しい領域の座標情報取得)
gn <- sort(genes(txdb))                   #遺伝子の座標
hoge <- promoters(gn, upstream=param_upstream,#指定
          downstream=param_downstream)#指定した範囲
hoge                                      #確認してるだ
obj <- (ranges(hoge)@start >= 0)          #条件を満たす
```

R Console

```
> #本番(配列取得)
> fasta <- getSeq(FaFile(in_f1), hoge)    #指定した範囲の塩基配列情$
> names(fasta) <- names(hoge)             #description情報を追加して$
> fasta                                   #確認してるだけです
  A DNAStringSet instance of length 2680
       width seq                                          names
   [1]   520 CTTGAAAGCCTTGAAA...ATTTACGATCACCCG gene:LCA12A_0618
   [2]   520 GAACTGATAATGTGCG...AACAATCGAAATCAC gene:LCA12A_0619
   [3]   520 GTTTCACGATCAATCG...ACTGGATCACTTGGT gene:LCA12A_0620
   [4]   520 CCACAGCATTGGCTGT...GGACAAGAAAGAAAC gene:LCA12A_0621
   [5]   520 ACTGATCATTATGACC...TGATCGCCAAGAAAG gene:LCA12A_0622
   ...   ... ...
[2676]   520 AAGTGACCGTTGCTTA...TGCATCTATCACTGC gene:LCA12A_0612
[2677]   520 ACACAGAAACTTTATG...AACCTTTCAAGGCAG gene:LCA12A_0613
[2678]   520 TGCGCAAGTCATATCG...AAATCGTCAAATCAA gene:LCA12A_0614
[2679]   520 CGAATTTCCTGTGATT...TGCGCGTTTCGGCGG gene:LCA12A_0615
[2680]   520 GCGCCAATTCACGCTC...AACCAAACGGACTTT gene:LCA12A_0616
>
> #ファイルに保存
> writeXStringSet(fasta, file=out_f, format="fasta", width=50)#fast$
>
```

プロモーター配列取得

当時(2015年3月; R ver. 3.1.2)は、GFF3形式のアノテーションファイル読み込み時に①makeTranscriptDbFromGFF関数、および②useGenesAsTranscriptsオプション(デフォルトはF)を利用していた。

7. 乳酸菌ゲノム(Lactobacillus casei 12A)の[上流500塩基, 下流20塩基]のプロモ

Ensembl (Flicek et al., Nucleic Acids Res., 2014)から提供されている Lactobacillu
(Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3)と
(Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa) を読み込むやり方です。
makeTranscriptDbFromGFF関数実行時に用いているuseGenesAsTranscripts=Tは、原核生物(prokaryotes)に代表される「転写物＝遺伝子」の場合に指定します。デフォルトはFです。

```
in_f1 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa"#入力ファイル名を指
in_f2 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3"#入力ファイル名を指定
out_f <- "hoge7.fasta"                    #出力ファイル名を指定してout_fに格納
param_upstream <- 500                     #転写開始点上流の塩基配列数を指定
param_downstream <- 20                    #転写開始点下流の塩基配列数を指定

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

#前処理(アノテーション情報を取得)
txdb <- makeTranscriptDbFromGFF(in_f2, #TranscriptDbオブジェクトを取得してtxdbに格納
          format="gff3", useGenesAsTranscripts=T)#TranscriptDbオブジェクトを取得してtxdbに格納
txdb                                      #確認してるだけです

#前処理(欲しい領域の座標情報取得)
gn <- sort(genes(txdb))                   #遺伝子の座標情報を取得
hoge <- promoters(gn, upstream=param_upstream,#指定した範囲の座標情報を取得
          downstream=param_downstream)#指定した範囲の座標情報を取得
hoge                                      #確認してるだけです
obj <- (ranges(hoge)@start >= 0)          #条件を満たすかどうかを判定した結果をobjに格納(座標が0よりも小さい
```

プロモーター配列取得

2015年7月3日にR ver. 3.2.0で黒枠部分をコピペすると警告メッセージが出た

```
in_f1 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa"#入力ファイル
in_f2 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3"#入力ファイル名
out_f <- "hoge7.fasta"                    #出力ファイル名を指定してout_fに格納
param_upstream <- 500                     #転写開始点上流の塩基配列数を指定
param_downstream <- 20                    #転写開始点下流の塩基配列数を指定

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

#前処理(アノテーション情報を取得)
txdb <- makeTranscriptDbFromGFF(in_f2, #TranscriptDbオブジェクトを取得してtxdbに格納
          format="gff3", useGenesAsTranscripts=T)#TranscriptDbオブジェクトを取得してtxdbに格納
txdb                                      #確認してるだけです
```

```
#前処理(欲しい領域の座標情報取
gn <- sort(genes(txdb))
hoge <- promoters(gn, upstr
          downstream=param
hoge
obj <- (ranges(hoge)@start
```

R Console

```
> #前処理(アノテーション情報を取得)
> txdb <- makeTranscriptDbFromGFF(in_f2, #TranscriptDbオブジェクトを$
+           format="gff3", useGenesAsTranscripts=T)#TranscriptDbオ$
Prepare the 'metadata' data frame ... metadata: OK
Warning messages:
1: 'makeTranscriptDbFromGFF' is deprecated.
Use 'makeTxDbFromGFF' instead.
See help("Deprecated")
2: 'useGenesAsTranscripts' is ignored and deprecated
3: In matchCircularity(seqlevels(gr), circ_seqs) :
  None of the strings in your circ_seqs argument match your seqnames.
> txdb                                    #確認してるだけです
TxDb object:
```

①

②

プロモーター配列取得

R Console

```
> #前処理(アノテーション情報を取得)
> txdb <- makeTranscriptDbFromGFF(in_f2, #TranscriptDb
+             format="gff3", useGenesAsTranscripts=T)#T
Prepare the 'metadata' data frame ... metadata: OK
Warning messages:
1: 'makeTranscriptDbFromGFF' is deprecated.
Use 'makeTxDbFromGFF' instead.
See help("Deprecated")
2: 'useGenesAsTranscripts' is ignored and deprecated
3: In matchCircularity(seqlevels(gr), circ_seqs) :
  None of the strings in your circ_seqs argument match your seqnames.
> txdb                                    #確認してるだけです
TxDb object:
# Db type: TxDb
# Supporting package: GenomicFeatures
# Data source: Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3
# Organism: NA
# miRBase build ID: NA
# Genome: NA
# transcript_nrow: 2799
# exon_nrow: 2800
# cds_nrow: 2681
# Db created by: GenomicFeatures package from Bioconductor
# Creation time: 2015-07-25 15:30:24 +0900 (Sat, 25 Jul 2015)
# GenomicFeatures version at creation time: 1.20.1
# RSQLite version at creation time: 1.0.0
# DBSCHEMAVERSION: 1.1
>
```

①

②

③

警告メッセージの中身。①makeTranscriptDbFromGFF関数は削除予定だ。makeTxDbFromGFF関数を使用せよ。②useGenesAsTranscriptsオプションを与えても無視するし、このオプション自体も削除予定だよ。といった具合。R ver. 3.2.0では、今のところ③で示すようにうまく読み込めてはいる。

プロモーター配列取得

2015年7月25日現在。③R ver. 3.1.2で正常動作したコードを(#のコメントアウトで)残しつつ、④R ver. 3.2.0で動作確認済みのコードがデフォルトとなるようにしている。

- イントロ | 一般 | 配列取得 | ゲノム配列 | BSgenome(last modified 2015/04/22)
- イントロ | 一般 | 配列取得 | プロモーター配列 | 公共DBから(last modified 2014/04/02)
- イントロ | 一般 | 配列取得 | プロモーター配列 | BSgenomeとTxDbから(last modified 2015/02/20)
- イントロ | 一般 | 配列取得 | プロモーター配列 | GenomicFeatures(Lawrence_2013) NEW (last modified
- イントロ | 一般 | 配列取得 | トランスクリプトーム配列 | 公共DBから(last modified 2015/05/09)

①

イントロ | 一般 | 配列取得 | プロモーター配列 | GenomicFeatures(Lawrence_2013) NEW

GenomicFeaturesパッケージを主に用いてプロモーター配列(転写開始点近傍配列)を得るやり方を示します。
ここでは、イントロ | 一般 | 配列取得 | ゲノム配列(BSgenomeから)で指定可能なゲノムとイントロ | NGS | アノテーション情報取得 | TranscriptDb

②

7. 乳酸菌ゲノム(Lactobacillus casei 12A)の[上流500塩基, 下流20塩基]のプロモーター配列を取得する場合:

Ensembl (Flicek et al., Nucleic Acids Res., 2014)から提供されている Lactobacillus casei 12Aの multi-FASTA形式ゲノム配列ファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa)と GFF3形式のアノテーションファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3) を読み込むやり方です。

```
in_f1 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa"#入力ファイル
in_f2 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3"#入力ファイル名
out_f <- "hoge7.fasta"                    #出力ファイル名を指定してout_fに格納
param_upstream <- 500                     #転写開始点上流の塩基配列数を指定
param_downstream <- 20                    #転写開始点下流の塩基配列数を指定

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

#前処理(アノテーション情報を取得)
#txdb <- makeTranscriptDbFromGFF(in_f2,#TranscriptDbオブジェクトを取得してtxdbに格納
#          format="gff3", useGenesAsTranscripts=T)#TranscriptDbオブジェクトを取得してtxdbに格納
txdb <- makeTxDbFromGFF(in_f2, format="gff3")#TranscriptDbオブジェクトを取得してtxdbに格納
txdb                                      #確認してるだけです
```

③

④

R ver. 3.2.0で実行

makeTranscriptDbFromGFF関数やuseGenesAsTranscriptsオプションを使っていないので、①それ系の警告は出ていないことがわかる。

R Console

```
> #前処理(アノテーション情報を取得)
> #txdb <- makeTranscriptDbFromGFF(in_f2,#TranscriptDbオブ$
> #           format="gff3", useGenesAsTranscripts=T)#Trans$
> txdb <- makeTxDbFromGFF(in_f2, format="gff3")#Transcript$
Prepare the 'metadata' data frame ... metadata: OK
Warning message:
In matchCircularity(seqlevels(
  None of the strings in your
> txdb
TxDb object:
# Db type: TxDb
# Supporting package: GenomicFe
# Data source: Lactobacillus_ca
# Organism: NA
# miRBase build ID: NA
# Genome: NA
# transcript_nrow: 2799
# exon_nrow: 2800
```

①

R Console

```
> fasta                                          #確認してるだけで$
  A DNAStringSet instance of length 2726
        width seq                                      names        $
    [1]   520 CTTGAAAGCCTTGA...TTTACGATCACCCG LCA12A_0618
    [2]   520 GAACTGATAATGTG...ACAATCGAAATCAC LCA12A_0619
    [3]   520 GTTTCACGATCAAT...CTGGATCACTTGGT LCA12A_0620
    [4]   520 CCACAGCATTGGCT...GACAAGAAAGAAAC LCA12A_0621
    [5]   520 ACTGATCATTATGA...GATCGCCAAGAAAG LCA12A_0622
    ...   ... ...
[2722]   520 AAGTGACCGTTGCT...GCATCTATCACTGC LCA12A_0612
[2723]   520 ACACAGAAACTTTA...ACCTTTCAAGGCAG LCA12A_0613
[2724]   520 TGCGCAAGTCATAT...AATCGTCAAATCAA LCA12A_0614
[2725]   520 CGAATTTCCTGTGA...GCGCGTTTCGGCGG LCA12A_0615
[2726]   520 GCGCCAATTCACGC...ACCAAACGGACTTT LCA12A_0616
>
> #ファイルに保存
> writeXStringSet(fasta, file=out_f, format="fasta", width$
>
```

R ver. 3.1.2で実行

R ver. 3.1.2(正確にはGenomicFeatures ver. 1.18.7)当時は、まだmakeTxDbFromGFF関数が実装されていないので、①makeTxDbFromGFF関数を見つけることができないのは当たり前。

```
in_f1 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chr
in_f2 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.chromos
out_f <- "hoge7.fasta"                    #出力ファイル名を指定してout_fに格納
param_upstream <- 500                     #転写開始点上流の塩基配列数を指定
param_downstream <- 20                    #転写開始点下流の塩基配列数を指定

#必要なパッケージをロード
library(Rsamtools)
library(Biostrings)
library(GenomicFeatures)

#前処理(アノテーション情報を取得)
#txdb <- makeTranscriptDbFromGFF(in_f
#          format="gff3", useGenesAsT
txdb <- makeTxDbFromGFF(in_f2, format
txdb

#前処理(欲しい領域の座標情報取得)
gn <- sort(genes(txdb))
hoge <- promoters(gn, upstream=param_
          downstream=param_downstrea
hoge
```

①

```
R Console
The following object is masked from 'package:GenomeInfoDb':

    species

 警告メッセージ:
1:  パッケージ 'GenomicFeatures' はバージョン 3.1.3 の R の下で$
2:  パッケージ 'AnnotationDbi' はバージョン 3.1.3 の R の下で造$
>
> #前処理(アノテーション情報を取得)
> #txdb <- makeTranscriptDbFromGFF(in_f2,#TranscriptDbオブジェク$
> #          format="gff3", useGenesAsTranscripts=T)#Transcript
> txdb <- makeTxDbFromGFF(in_f2, format="gff3")#TranscriptDbオブ
エラー:  関数 "makeTxDbFromGFF" を見つけることができませんでした
> txdb                                          #確認してるだけです
 エラー:  オブジェクト 'txdb' がありません
> package.version("GenomicFeatures")
[1] "1.18.7"
> packageVersion("GenomicFeatures")
[1] '1.18.7'
>
```

R ver. 3.1.3で実行

R ver. 3.1.3(正確にはGenomicFeatures ver. 1.18.7)当時は、まだmakeTxDbFromGFF関数が実装されていないので、①makeTxDbFromGFF関数を見つけることができないのは当たり前。

```
in_f1 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chr
in_f2 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.chromos
out_f <- "hoge7.fasta"                    #出力ファイル名を指定してout_fに格納
param_upstream <- 500                     #転写開始点上流の塩基配列数を指定
param_downstream <- 20                    #転写開始点下流の塩基配列数を指定

#必要なパッケージをロード
library(Rsamtools)
library(Biostrings)
library(GenomicFeatures)

#前処理(アノテーション情報を取得)
#txdb <- makeTranscriptDbFromGFF(in_f
#          format="gff3", useGenesAsT
txdb <- makeTxDbFromGFF(in_f2, format
txdb

#前処理(欲しい領域の座標情報取得)
gn <- sort(genes(txdb))
hoge <- promoters(gn, upstream=param_
          downstream=param_downstrea
hoge
```

①

```
R Console

 次のパッケージを付け加えます: 'AnnotationDbi'

The following object is masked from 'package:GenomeInfoDb':

    species

>
> #前処理(アノテーション情報を取得)
> #txdb <- makeTranscriptDbFromGFF(in_f2,#TranscriptDbオブジェク$
> #          format="gff3", useGenesAsTranscripts=T)#Transcript
> txdb <- makeTxDbFromGFF(in_f2, format="gff3")#TranscriptDbオブ
エラー:  関数 "makeTxDbFromGFF" を見つけることができませんでした
> txdb                                          #確認してるだけです
 エラー:  オブジェクト 'txdb' がありません
> package.version("GenomicFeatures")
[1] "1.18.7"
> packageVersion("GenomicFeatures")
[1] '1.18.7'
>
```

定期的にバージョンアップ

バグの修正や新たな機能がどんどん追加されている。最新版の利用をお勧め。推奨の毎年5月と11月ごろにバージョンアップをちゃんとやったヒトは、R ver. 3.2.0以上なはず。

- 近年のリリース頻度
  - R本体 (http://www.r-project.org/)
    - 2015-06-18にver. 3.2.1をリリース
    - 2015-04-16にver. 3.2.0をリリース
    - 2015-03-09にver. 3.1.3をリリース
    - 2014-10-31にver. 3.1.2をリリース
    - …
    - 2012-03-30にver. 2.15.0をリリース
    - …
  - Bioconductor (http://bioconductor.org/)は半年ごとにリリース
    - 2015-04にver. 3.1をリリース (R ver. 3.2.1で動作確認)、提供パッケージ数:1,024
    - 2014-10にver. 3.0をリリース (R ver. 3.1.1で動作確認)、提供パッケージ数:934
    - 2014-04にver. 2.14をリリース (R ver. 3.1.0で動作確認)、提供パッケージ数:824
    - 2013-10にver. 2.13をリリース (R ver. 3.0で動作確認)、提供パッケージ数:750
    - 2013-04にver. 2.12をリリース (R ver. 3.0で動作確認)、提供パッケージ数:672
    - 2012-10にver. 2.11をリリース (R ver. 2.15.1で動作確認)、提供パッケージ数:608
    - 2012-04にver. 2.10をリリース (R ver. 2.15.0で動作確認)、提供パッケージ数:553
    - …

Contents

- パッケージ
  - CRANとBioconductor
  - 推奨パッケージインストール手順のおさらい
  - Bioconductor概観 → ゲノム配列パッケージ(BSgenome)
  - ヒトゲノム情報パッケージの解析
- プロモーター配列取得(sessionInfo、バージョンの違い)
  - Rパッケージ(ゲノムとアノテーションパッケージの併用)
  - FASTA形式ファイルとGFF3形式ファイル
  - データの型
- FASTQファイルの各種解析(LinuxとRを相補的に活用)
  - Linux (FastQC)とR (ShortRead)で同じ:Overrepresented sequences項目
  - Linux (FastQC)とR (QuasR)で異なる見栄え:Per base sequence content項目。FastQCに--nogroupというオプションがあることを知る。
  - FastQCのオプション (デフォルトと--nogroupあり)の違いによるKmer Content項目の結果の違い → Rで検証

データの型

簡単にいうと、「型　＝　見た目」です。例えば、①makeTxDbFromGFF関数は、(GFF3形式の)アノテーションファイルを読み込んだものを②TxDbという形式のオブジェクトとして格納しています。重要なのは、細かい中身の理解ではなく、ちゃんと読み込めているっぽいことを確認すること

7. 乳酸菌ゲノム(Lactobacillus casei 12A)の[上流500塩基, 下流20

Ensembl (Flicek et al., Nucleic Acids Res., 2014)から提供されてい
配列ファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.dna.c
ションファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.

```
in_f1 <- "Lactobacillus_casei_12a.GCA_00030956
in_f2 <- "Lactobacillus_casei_12a.GCA_00030956
out_f <- "hoge7.fasta"                          #出力フ
param_upstream <- 500                           #転写開始
param_downstream <- 20                          #転写開始

#必要なパッケージをロード
library(Rsamtools)                              #パッケ
library(Biostrings)                             #パッ
library(GenomicFeatures)                        #パッ

#前処理(アノテーション情報を取得)
#txdb <- makeTranscriptDbFromGFF(in_f2,#Transc
#          format="gff3", useGenesAsTranscript
txdb <- makeTxDbFromGFF(in_f2, format="gff3")#
txdb                                            #確認し

#前処理(欲しい領域の座標情報取得)
gn <- sort(genes(txdb))                         #遺伝子
hoge <- promoters(gn, upstream=param_upstream,
          downstream=param_downstream)#指定し
hoge                                            #確認し
```

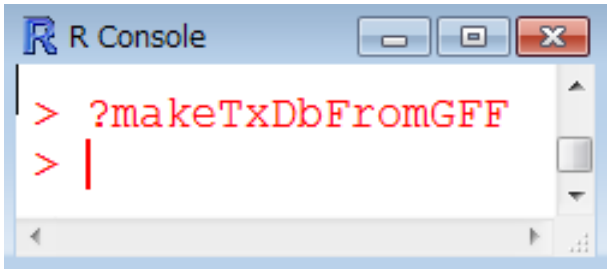
①

R Console

```
> txdb <- makeTxDbFromGFF(in_f2, format="gff3")#Trans$
Prepare the 'metadata' data frame ... metadata: OK
Warning message:
In matchCircularity(seqlevels(gr), circ_seqs) :
  None of the strings in your circ_seqs argument matc$
> txdb                                          #確認してる$
TxDb object:
# Db type: TxDb
# Supporting package: GenomicFeatures
# Data source: Lactobacillus_casei_12a.GCA_000309565.$
# Organism: NA
# miRBase build ID: NA
# Genome: NA
# transcript_nrow: 2799
# exon_nrow: 2800
# cds_nrow: 2681
# Db created by: GenomicFeatures package from Biocond$
# Creation time: 2015-07-25 17:25:09 +0900 (Sat, 25 J$
# GenomicFeatures version at creation time: 1.20.1
# RSQLite version at creation time: 1.0.0
# DBSCHEMAVERSION: 1.1
>
```

データの型

「?*関数名*」で見られるhtmlマニュアル中に型に関する記載あり。



makeTxDbFromGFF {GenomicFeatures} R Documentation

Make a TxDb object from annotations available as a GFF3 or GTF file

Description

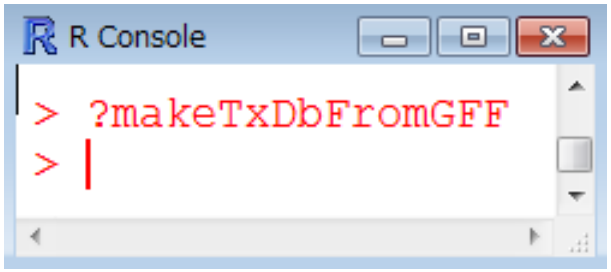
The `makeTxDbFromGFF` function allows the user to make a TxDb object from transcript annotations available as a GFF3 or GTF file.

Usage

```
makeTxDbFromGFF(file,
                format=c("gff3","gtf"),
                exonRankAttributeName=NA,
                gffGeneIdAttributeName=NA,
                chrominfo=NA,
                dataSource=NA,
                organism=NA,
                circ_seqs=DEFAULT_CIRC_SEQS,
                miRBaseBuild=NA,
                useGenesAsTranscripts=FALSE,
                gffTxName="mRNA",
                species=NA)
```

データの型

①これはGenomicFeaturesパッケージが提供している関数。②GFF3形式以外にGTF形式も読み込めるようだ。③GTF形式ファイルの場合はformat="gtf"と書けばよさそう



makeTxDbFromGFF {GenomicFeatures} R Documentation

①

Make a TxDb object from annotations available as a GFF3 or GTF file

②

Description

The `makeTxDbFromGFF` function allows the user to make a TxDb object from transcript annotations available as a GFF3 or GTF file.

Usage

```
makeTxDbFromGFF(file,
                format=c("gff3","gtf"),
                exonRankAttributeName=NA,
                gffGeneIdAttributeName=NA,
                chrominfo=NA,
                dataSource=NA,
                organism=NA,
                circ_seqs=DEFAULT_CIRC_SEQS,
                miRBaseBuild=NA,
                useGenesAsTranscripts=FALSE,
                gffTxName="mRNA",
                species=NA)
```

③

データの型

makeTxDbFromGFF is a convenience function tha
file to the makeTxDbFromGRanges function.

Value

A TxDb object. ①

Author(s)

M. Carlson and H. Pages

See Also

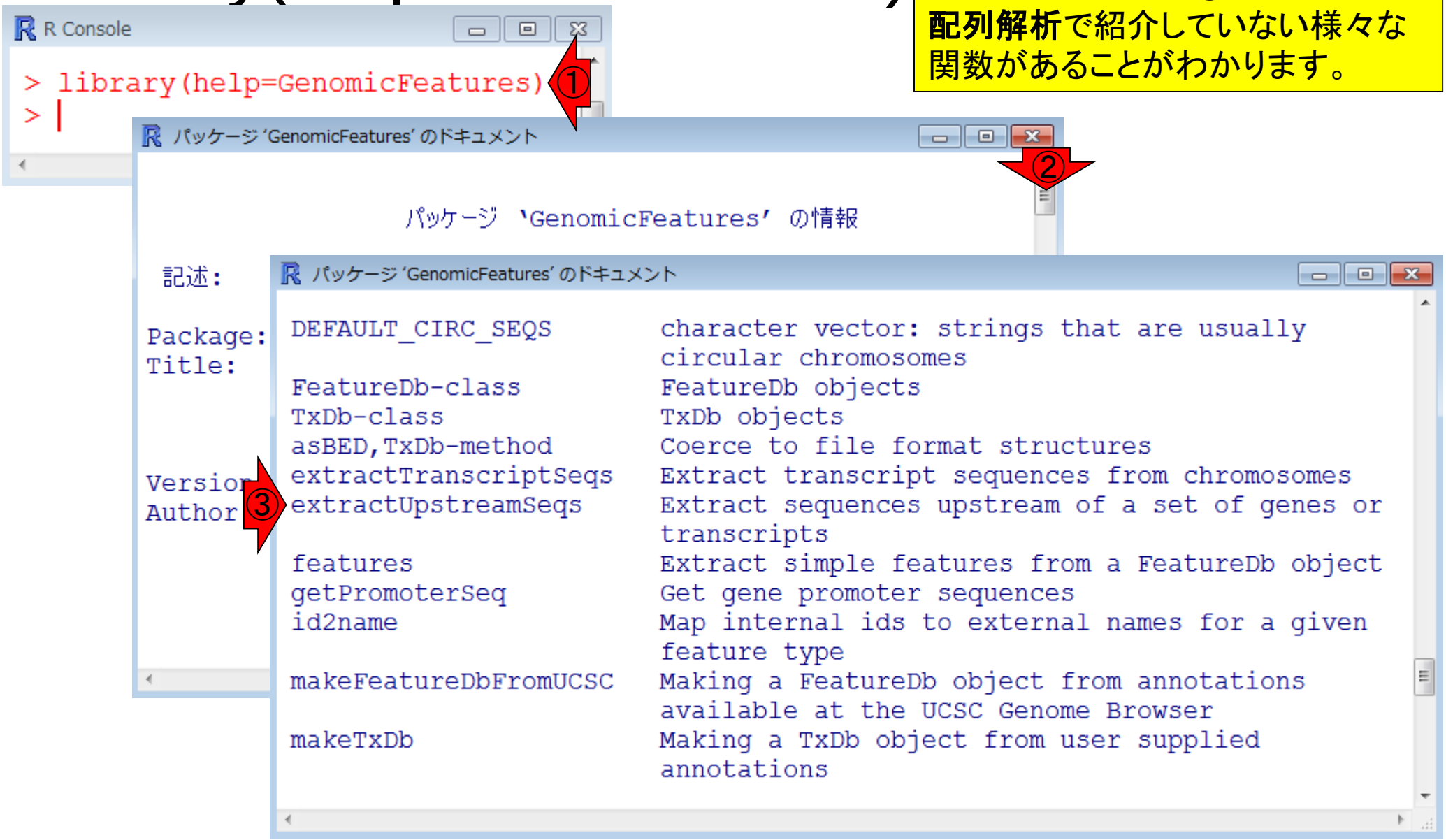
- makeTxDbFromGRanges, which makeTxDbFromGFF is based on, for making a TxDb object from a GRanges object. ②
- The import function in the **rtracklayer** package (also used by makeTxDbFromGFF internally).
- makeTxDbFromUCSC and makeTxDbFromBiomart for convenient ways to make a TxDb object from UCSC or BioMart online resources. ③ ④

htmlマニュアルの下部に移動。①makeTxDbFromGFF関数実行結果として得られるものはTxDbという形式のオブジェクト。②もちろんR上でアノテーション情報を格納する形式はTxDbだけではなく、GRangesという形式も存在する。GRanges → TxDbへの型変換はmakeTxDbFromGRanges関数で可能であることがわかる。アノテーション情報源は、③UCSCや④BioMartが有名だが、そこからうまく情報を抽出してTxDbオブジェクトを得るための関数も存在することが分かる。こんな感じで関数の知識や利用の幅を広げていく。

参考

library(help=パッケージ名)

①指定したパッケージ中で利用可能な関数を概観できます。②下のほうに移動すると、③など(Rで)**塩基配列解析**で紹介していない様々な関数があることがわかります。



パッケージ概観

②Bioconductor (ver. 3.1)上でのGenomicFeaturesのトップページ。③ダウンロード数はトップ5%であることがわかる

- イントロ | 一般 | 配列取得 | プロモーター配列 | GenomicFeatures(Lawrence 2013) 参考

- イントロ | 一般 | 配列取得 | ゲノム配列 | BSgenome(last modified 2015/04/22)
- イントロ | 一般 | 配列取得 | プロモーター配列 | 公共DBから(last modified 2014/04/02)
- イントロ | 一般 | 配列取得 | プロモーター配列 | BSgenomeとTxDbから(last modified 2015/02/20)
- イントロ | 一般 | 配列取得 | プロモーター配列 | GenomicFeatures(Lawrence 2013) (last modified
- イントロ | 一般 | 配列取得 | トランスクリプトーム配列 | 公共DBから(last modified 2015/05/09)

①

イントロ | 一般 | 配列取得 | プロモーター配列 | GenomicFeatures(Lawrence_2013) NEW

GenomicFeaturesパッケージを主に用いてプロモーター配列(転写開始点近傍配列)を得るやり方を示します。

②

Bioconductor
OPEN SOURCE SOFTWARE FOR BIOINFORMATICS

Search:

Home | Install | Help | Developers | About

Home » Bioconductor 3.1 » Software Packages » GenomicFeatures

GenomicFeatures

③

platforms all | downloads top 5% | posts 32 / 2 / 3 / 8 | in Bioc 5.5 years
build ok | commits 15.17 | test coverage 70%

Tools for making and manipulating transcript centric annotations

Bioconductor version: Release (3.1)

A set of tools and methods for making and manipulating transcript centric annotations. With these tools

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

パッケージ概観

少し下のほうに移動。①原著論文はきっちり引用。②パッケージのインストール法。③パッケージのドキュメント(マニュアルではない)をローカル環境で開く方法。④具体的なやり方。④の中身は⑤と同じだが、ネットワークかローカルかの違いがある。⑥利用可能な関数マニュアルはここでも見られる。

Author: M. Carlson, H. Pages, P. Aboyoun, S. Falcon, M.

Maintainer: Bioconductor Package Maintainer <maintaine

Citation (from within R, enter citation("GenomicFeatures")):

Lawrence M, Huber W, Pagès H, Aboyoun P, Carlson M, Gentleman R, Morgan M and Carey V (2013). "Software for Computing and Annotating Genomic Ranges." *PLoS Computational Biology*, **9**. http://dx.doi.org/10.1371/journal.pcbi.1003118, http://www.ploscompbiol.org/article/info%3Adoi%2F10.1371%2Fjournal.pcbi.1003118

①

Installation

To install this package, start R and enter:

```
source("http://bioconductor.org/biocLite.R")
biocLite("GenomicFeatures")
```

②

Documentation

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

```
browseVignettes("GenomicFeatures")
```

③

R Console

```
> browseVignettes("GenomicFeatures")
starting httpd help server ... done
>
```

④

⑤

PDF	R Script	Making and Utilizing TxDb Objects
PDF		Reference Manual
Text		NEWS

⑥

目的をおさらい

乳酸菌*L. casei 12A*株のゲノム配列とアノテーションファイルを読み込んで、全遺伝子の上流500塩基、および下流20塩基分の塩基配列を抽出してプロモーター解析（するための入力データを取得）したい!
黒枠までで行ったことは、アノテーションファイルを読み込んでtxdbオブジェクトに格納したところまで。

7. 乳酸菌ゲノム(Lactobacillus casei 12A)の[上流500塩基, 下流20塩基]のプロ

Ensembl (Flicek et al., Nucleic Acids Res., 2014)から提供されている Lactobac
配列ファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome
ションファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.

```
in_f1 <- "Lactobacillus_casei_12a.GCA_00030956
in_f2 <- "Lactobacillus_casei_12a.GCA_00030956
out_f <- "hoge7.fasta"                          #出力フ
param_upstream <- 500                           #転写開始
param_downstream <- 20                          #転写開始

#必要なパッケージをロード
library(Rsamtools)                              #パッケ
library(Biostrings)                             #パッケ
library(GenomicFeatures)                        #パッケ

#前処理(アノテーション情報を取得)
#txdb <- makeTranscriptDbFromGFF(in_f2,#Transc
#          format="gff3", useGenesAsTranscript
txdb <- makeTxDbFromGFF(in_f2, format="gff3")#
txdb                                            #確認して

#前処理(欲しい領域の座標情報取得)
gn <- sort(genes(txdb))                         #遺伝子
hoge <- promoters(gn, upstream=param_upstream,
          downstream=param_downstream)#指定した
hoge                                            #確認して
```

R Console

```
> txdb <- makeTxDbFromGFF(in_f2, format="gff3")#Trans$
Prepare the 'metadata' data frame ... metadata: OK
Warning message:
In matchCircularity(seqlevels(gr), circ_seqs) :
  None of the strings in your circ_seqs argument matc$
> txdb                                          #確認してる$
TxDb object:
# Db type: TxDb
# Supporting package: GenomicFeatures
# Data source: Lactobacillus_casei_12a.GCA_000309565.$
# Organism: NA
# miRBase build ID: NA
# Genome: NA
# transcript_nrow: 2799
# exon_nrow: 2800
# cds_nrow: 2681
# Db created by: GenomicFeatures package from Biocond$
# Creation time: 2015-07-25 17:25:09 +0900 (Sat, 25 J$
# GenomicFeatures version at creation time: 1.20.1
# RSQLite version at creation time: 1.0.0
# DBSCHEMAVERSION: 1.1
>
```

残りのコードを概説

```
#前処理(アノテーション情報を取得)
#txdb <- makeTranscriptDbFromGFF(in_f2,#TranscriptDbオ
#          format="gff3", useGenesAsTranscripts=T)#Tran
txdb <- makeTxDbFromGFF(in_f2, format="gff3")#Transcrip
txdb                                          #確認してるだけで

#前処理(欲しい領域の座標情報取得)
gn <- sort(genes(txdb))                       #遺伝子の座標情報
hoge <- promoters(gn, upstream=param_upstream,#指定した範
          downstream=param_downstream)#指定した範囲の座
hoge                                          #確認してるだけで
obj <- (ranges(hoge)@start >= 0)              #条件を満たすかどうかを判定した結果をobjに格納(座標が0よりも小
hoge <- hoge[obj]                             #objがTRUEとなる要素のみ抽出した結果をhogeに格納
hoge                                          #確認してるだけです

#本番(配列取得)
fasta <- getSeq(FaFile(in_f1), hoge)          #指定した範囲の塩基配列情報を取得
names(fasta) <- names(hoge)                   #description情報を追加している
fasta                                         #確認してるだけです

#ファイルに保存
writeXStringSet(fasta, file=out_f, format="fasta", width=50)#fastaの中身を指定したファイル名で保存
```

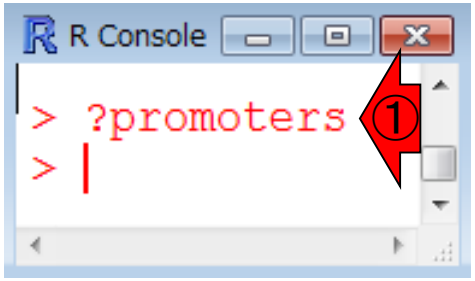
①TxDbオブジェクト(オブジェクト名txdb)を出発点として、欲しい領域(上流500塩基、下流20塩基)情報を取得。②ゲノム配列情報FaFile(in_f1)と欲しい領域情報hogeを入力として塩基配列を取得。③FASTA形式ファイルで保存。このあたりは、H26年度受講生追跡調査の「**配列情報を使った次の解析へのinput fileの加工が難関だと感じでいる**」に対する回答。特にプロモーター解析などは「…で、その入力ファイルをどうやって作るの?」かが実務担当者の直面する課題。

データの型

①黒枠部分は感覚的にpromotersという関数を用いて転写開始点上流と下流の範囲を取得した結果をhogeに格納してるんだろうな、くらいは分かりますよね。そしてなぜ②のところでtxdbを使っていないのか?という素朴な疑問があるでしょう。理由はpromoters関数が入力としてTxDbオブジェクトを想定していないからです(おそらく2014年10月リリースのBioconductor ver. 3.0までは正解)。2015年4月リリースのBioconductor ver. 3.1では、GenomicFeaturesパッケージから提供されているpromoters関数はTxDbに対応済み。

```
#前処理(アノテーション情報を取得)
#txdb <- makeTranscriptDbFromGFF(in_
#           format="gff3", useGenesAs
txdb <- makeTxDbFromGFF(in_f2, forma
txdb

#前処理(欲しい領域の座標情報取得)
gn <- sort(genes(②))
hoge <- promoters(gn, upstream=param_upstream,#指定した範囲の座標情報を取得
           downstream=param_downstream)#指定した範囲の座標情報を取得
hoge                                   #確認してるだけです
obj <- (ranges(hoge)@start >= 0)       #条件を満たすかどうかを判定した結果をobjに格納(座標が0よりも小
hoge <- hoge[obj]                      #objがTRUEとなる要素のみ抽出した結果をhogeに格納
hoge                                   #確認してるだけです

#本番(配列取得)
fasta <- getSeq(FaFile(in_f1), hoge)   #指定した範囲の塩基配列情報を取得
names(fasta) <- names(hoge)            #description情報を追加している
fasta                                  #確認してるだけです

#ファイルに保存
writeXStringSet(fasta, file=out_f, format="fasta", width=50)#fastaの中身を指定したファイル名で保存
```

データの型



①「?promoters」。門田のR ver. 3.2.0環境では、promotersという同じ名前の関数が3つのパッケージ(IRanges, GenomicFeatures, and GenomicRanges)から提供されていることがわかる。②GenomicRangesパッケージから提供されているpromoters関数は入力としてTxDbオブジェクトを受け付けないことがわかる。しかし、③GenomicFeaturesパッケージから提供されているpromoters関数は、TxDbを入力としていることが2015年7月26日に判明(爆)。

Help on topic 'promoters' was found in the following packages:

Intra range transformations of a Ranges, Views, RangesList, or MaskCollection object
(in package IRanges in library C:/Users/kadota/Documents/R/win-library/3.2)
Extract genomic features from an object
(in package GenomicFeatures in library C:/Users/kadota/Documents/R/win-library/3.2)
② Intra range transformations of a GRanges or GRangesList object
(in package GenomicRanges in library C:/Users/kadota/Documents/R/win-library/3.2)

③

データの型

③GenomicFeaturesパッケージから提供されているpromoters関数は、④TxDbを入力としていることが2015年7月26日に判明(爆)。

Help on topic 'promoters' was found in the following packages:

Intra range transformations of a Ranges, Views, RangesList, or MaskCollection object
(in package IRanges in library C:/Users/kadota/Documents/R/win-library/3.2)
Extract genomic features from an object
(in package GenomicFeatures in library C:/Users/kadota/Documents/R/win-library/3.2)
Intra range transformations of a GRanges or GRangesList object
(in package Genom

③

transcripts {GenomicFeatures} R Documentation

Extract genomic features from an object

Description

④

Generic functions to extract genomic features from an object. This page documents the methods for TxDb objects only.

Usage

```
transcripts(x, ...)
## S4 method for signature 'TxDb'
transcripts(x, vals=NULL, columns=c("tx_id", "tx_name"))

exons(x, ...)
## S4 method for signature 'TxDb'
exons(x, vals=NULL, columns="exon_id")

cds(x, ...)
## S4 method for signature 'TxDb'
cds(x, vals=NULL, columns="cds_id")

genes(x, ...)
## S4 method for signature 'TxDb'
genes(x, vals=NULL, columns="gene_id", single.strand.genes.only=TRUE)

## S4 method for signature 'TxDb'
promoters(x, upstream=2000, downstream=200, ...)
```

⑤

プロモーター配列取得

②例題8に、③TxDbオブジェクトを入力としてpromoters関数を実行するコードを追加。1つ上の行は過去の遺物(legacy)として残しているだけ。(Rで)**塩基配列解析**自体がデカすぎて、もはや全体の動作確認もままならないし、修正・変更もしきれていないのが現状

① ②

イントロ | 一般 | 配列取得 | プロモーター配列 | Genomi

GenomicFeaturesパッケージを主に用いてプロモーター配列(転写開始点近傍配列)を得るやり方を示します。
ここでは、イン
TranscriptDb
xxx塩基, 下流
は、どの生物種に
状の指定が可能
Chr4, Chr5, ChrM
genome@seqinfo(
形式のアノテーシ

1. ヒト("BSgenom
ヒトゲノム ver. 1

8. 乳酸菌ゲノム(Lactobacillus casei 12A)の[上流100塩基, 下流0塩基]のプロモーター配列を取得する場合:

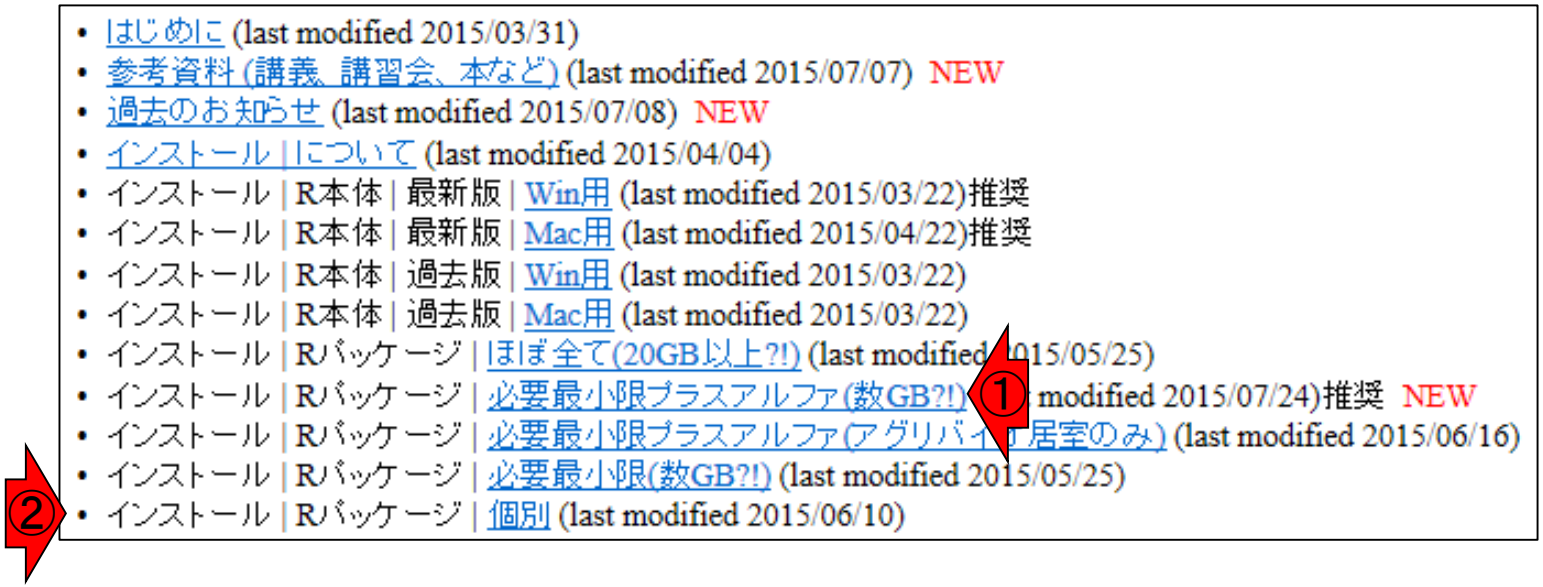
7.と基本的に同じですが、少なくとも2015年4月リリースのBioconductor ver. 3.1以降で、GenomicFeaturesパッケージが提供するpromoters関数がTxDbオブジェクトをそのまま読み込めることが判明したので、それを反映させています。

```
in_f1 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa"#入力フ
in_f2 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3"#入力ファ
out_f <- "hoge8.fasta"                          #出力ファイル名を指定してout_fに格納
param_upstream <- 100                           #転写開始点上流の塩基配列数を指定
param_downstream <- 0                           #転写開始点下流の塩基配列数を指定

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

#前処理(アノテーション情報を取得)
#txdb <- makeTranscriptDbFromGFF(in_f2,#TranscriptDbオブジェクトを取得してtxdbに格納
#          format="gff3", useGenesAsTranscripts=T)#TranscriptDbオブジェクトを取得してtxdbに
txdb <- makeTxDbFromGFF(in_f2, format="gff3")#TranscriptDbオブジェクトを取得してtxdbに格納
txdb                                            #確認してるだけです

#前処理(欲しい領域の座標情報取得)
#gn <- sort(genes(txdb))                        #遺伝子の座標情報を取得
hoge <- promoters(txdb, upstream=param_upstream,#指定した範囲の座標情報を取得
            downstream=param_downstream)#指定した範囲の座標情報を取得
hoge                                            #確認してるだけです
```

③

遺言1

- R本体とパッケージのバージョンアップを毎年5月と11月に行うべし
  - 依存関係など様々な問題はあるものの、落ち着いて必要なパッケージを個別にインストールしていけば大丈夫
  - 第一義的に重要なのは、Linuxでプログラムのインストールができることと同様、library(XXX)の部分でこけないようにすること。



遺言2

- R最新版でうまくいったなら、過去版のエラー原因究明は時間の無駄!
  - ①R ver. 3.1.2のときはmakeTranscriptDbFromGFF関数でうまくいくはずだが、エラーが出る現象に遭遇した(2015.07.07のアグリバイオ大学院講義)。しかし、②R ver. 3.2.1でmakeTxDbFromGFF関数で正常動作することがわかり、その時点で思考停止

7. 乳酸菌ゲノム(Lactobacillus casei 12A)の[上流500塩基, 下流20塩基]のプロモーター配列を取得する場合:

Ensembl (Flicek et al., Nucleic Acids Res., 2014)から提供されている Lactobacillus casei 12Aの multi-FASTA形式ゲノム配列ファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa)と GFF3形式のアノテーションファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3) を読み込むやり方です。

```
in_f1 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa"#入力ファ
in_f2 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3"#入力ファイ
out_f <- "hoge7.fasta"                    #出力ファイル名を指定してout_fに格納
param_upstream <- 500                     #転写開始点上流の塩基配列数を指定
param_downstream <- 20                    #転写開始点下流の塩基配列数を指定

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

#前処理(アノテーション情報を取得)
#txdb <- makeTranscriptDbFromGFF(in_f2,#TranscriptDbオブジェクトを取得してtxdbに格納
#          format="gff3", useGenesAsTranscripts=T)#TranscriptDbオブジェクトを取得してtxdbに格
txdb <- makeTxDbFromGFF(in_f2, format="gff3")#TranscriptDbオブジェクトを取得してtxdbに格納
txdb                                      #確認してるだけです

#前処理(欲しい領域の座標情報取得)
gn <- sort(genes(txdb))                   #遺伝子の座標情報を取得
hoge <- promoters(gn, upstream=param_upstream,#指定した範囲の座標情報を取得
          downstream=param_downstream)#指定した範囲の座標情報を取得
hoge                                      #確認してるだけです
```

遺言3

- 得られた結果を無条件に受け入れるな！様々な角度で検証せよ。
  - 実はプログラムのバグで、GFF3形式ファイルの読み込みに失敗していただけ、みたいなオチもよくあるだろう。フリーソフトの利用は自己責任である。

7. 乳酸菌ゲノム(Lactobacillus casei 12A)の[上流500塩基, 下流20塩基]のプロモーター配列を取得する場合:

Ensembl (Flicek et al., Nucleic Acids Res., 2014)から提供されている Lactobacillus casei 12Aの multi-FASTA形式ゲノム配列ファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa)と GFF3形式のアノテーションファイル(Lactobacillus_casei_12a.GCA_000309565

```
in_f1 <- "Lactobacillus_casei_12a.GCA_0
in_f2 <- "Lactobacillus_casei_12a.GCA_0
out_f <- "hoge7.fasta"                  #出
param_upstream <- 500                   #転
param_downstream <- 20                  #転

#必要なパッケージをロード
library(Rsamtools)                      #パ
library(Biostrings)                     #パ
library(GenomicFeatures)                #パ

#前処理(アノテーション情報を取得)
#txdb <- makeTranscriptDbFromGFF(in_f2,#Tr
#          format="gff3", useGenesAsTransc
txdb <- makeTxDbFromGFF(in_f2, format="gff
txdb                                    #確

#前処理(欲しい領域の座標情報取得)
gn <- sort(genes(txdb))                 #遺
hoge <- promoters(gn, upstream=param_upstr
          downstream=param_downstream)#指
hoge                                    #確
```

① > txdb

```
TxDb object:
# Db type: TxDb
# Supporting package: GenomicFeatures
# Data source: Lactobacillus_casei_12a.GCA_000309565.2.2$
# Organism: NA
# miRBase build ID: NA
# Genome: NA
# transcript_nrow: 2799
# exon_nrow: 2800
# cds_nrow: 2681
# Db created by: GenomicFeatures package from Bioconduct$
# Creation time: 2015-07-26 11:26:55 +0900 (Sun, 26 Jul $
# GenomicFeatures version at creation time: 1.20.1
# RSQLite version at creation time: 1.0.0
# DBSCHEMAVERSION: 1.1
>
```

Rを用いた検証例

txdbオブジェクトの元データは、①のGFF3ファイル。②乳酸菌L. casei 12Aの遺伝子数(=転写物数)は、2,799個となっている。これが正しいかを検証する。

7. 乳酸菌ゲノム(Lactobacillus casei 12A)の[上流500塩基, 下流20塩基]のプロモーター配列を取得

Ensembl (Flicek et al., Nucleic Acids Res., 2014)から提供されている Lactobacillus casei 12Aの multi-FASTA形式のゲノム配列ファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa)と GFF3形式のアノテーションファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3)を読み込むやり方です。

```
in_f1 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa"  #入力ファ
in_f2 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3"  #力ファイ
out_f <- "hoge7.fasta"                    #出力ファイル名を指定してout_fに格納
param_upstream <- 500                     #転写開始点上流の塩基配列数を指定
param_downstream <- 20                    #転写開始点下流の塩基配列数を指定

#必要なパッケージをロード
library(Rsamtools)                        #パ
library(Biostrings)                       #パ
library(GenomicFeatures)                  #パ

#前処理(アノテーション情報を取得)
#txdb <- makeTranscriptDbFromGFF(in_f2,#Tr
#          format="gff3", useGenesAsTransc
txdb <- makeTxDbFromGFF(in_f2, format="gff
txdb                                      #確

#前処理(欲しい領域の座標情報取得)
gn <- sort(genes(txdb))                   #遺
hoge <- promoters(gn, upstream=param_upstr
          downstream=param_downstream)#指
hoge                                      #確
```

①

R Console

```
> txdb
TxDb object:
# Db type: TxDb
# Supporting package: GenomicFeatures
# Data source: Lactobacillus_casei_12a.GCA_000309565.2.2$
# Organism: NA
# miRBase build ID: NA
# Genome: NA
# transcript_nrow: 2799
# exon_nrow: 2800
# cds_nrow: 2681
# Db created by: GenomicFeatures package from Bioconduct$
# Creation time: 2015-07-26 11:26:55 +0900 (Sun, 26 Jul $
# GenomicFeatures version at creation time: 1.20.1
# RSQLite version at creation time: 1.0.0
# DBSCHEMAVERSION: 1.1
>
```

②

Rを用いた検証例

(9列目の)"ID=gene"という文字列を含む行数を調べてみるといいのでは、という思想のもので検証する。

7. 乳酸菌ゲノム(Lactobacillus casei 12A)の[上流500塩基, 下流20塩基]のプロモーター配列を取得する場合:

Ensembl (Flicek et al., Nucleic Acids Res., 2014)から提供されている Lactobacillus casei 12Aの multi-FASTA形式ゲノム配列ファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa)と GFF3形式のアノテーションファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3) を読み込むやり方です。

```
in_f1 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.dna.chromosome.Chromosome.fa"#入力ファ
in_f2 <- "Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3"#入力ファイ
out_f <- "hoge7.fasta"                    #出力ファイル名を指定してout_fに格納
```

```
##gff-version 3
##sequence-region   Chromosome 1 2907892
```

Chromosome	ena	gene	1	1350	.	+	.	ID=gene:LCA12A_0617;assembly_name=GCA_000309565.2;b
Chromosome	ena	gene	1523	2662	.	+	.	ID=gene:LCA12A_0618;assembly_name=GCA_000309565.2;b
Chromosome	ena	gene	3240	3452	.	+	.	ID=gene:LCA12A_0619;assembly_name=GCA_000309565.2;b
Chromosome	ena	gene	3449	4564	.	+	.	ID=gene:LCA12A_0620;assembly_name=GCA_000309565.2;b
Chromosome	ena	gene	4817	6778	.	+	.	ID=gene:LCA12A_0621;assembly_name=GCA_000309565.2;b
Chromosome	ena	gene	6840	9461	.	+	.	ID=gene:LCA12A_0622;assembly_name=GCA_000309565.2;b
Chromosome	ena	gene	9566	10270	.	−	.	ID=gene:LCA12A_0623;assembly_name=GCA_000309565.2;b
Chromosome	ensembl	CDS	1	1350	.	+	0	ID=CDS:EKP96483;Parent=transcript:EKP96483;assembly_n
Chromosome	ensembl	exon	1	1350	.	+	.	Name=EKP96483-1;Parent=transcript:EKP96483;assembly_
Chromosome	ensembl	CDS	1523	2662	.	+	0	ID=CDS:EKP96484;Parent=transcript:EKP96484;assembly_n
Chromosome	ensembl	exon	1523	2662	.	+	.	Name=EKP96484-1;Parent=transcript:EKP96484;assembly_
Chromosome	ensembl	CDS	3240	3452	.	+	0	ID=CDS:EKP96485;Parent=transcript:EKP96485;assembly_n
Chromosome	ensembl	exon	3240	3452	.	+	.	Name=EKP96485-1;Parent=transcript:EKP96485;assembly_
Chromosome	ensembl	CDS	3449	4564	.	+	0	ID=CDS:EKP96486;Parent=transcript:EKP96486;assembly_n
Chromosome	ensembl	exon	3449	4564	.	+	.	Name=EKP96486-1;Parent=transcript:EKP96486;assembly_
Chromosome	ensembl	CDS	4817	6778	.	+	0	ID=CDS:EKP96487;Parent=transcript:EKP96487;assembly_n
Chromosome	ensembl	exon	4817	6778	.	+	.	Name=EKP96487-1;Parent=transcript:EKP96487;assembly_
Chromosome	ensembl	CDS	6840	9461	.	+	0	ID=CDS:EKP96488;Parent=transcript:EKP96488;assembly_n

Rを用いた検証例

txdbオブジェクトの元データは、①のGFF3ファイル。乳酸菌L. casei 12Aの遺伝子数(=転写物数)は、2,799個となっている。Rで③"ID=gene"を含む行数も④2,799個だったので、大丈夫だろうと判断する。

- 書籍 | 日本乳酸菌学会誌 | 第4回クオリティコントロールとプログラムのインストール
- イントロ | 一般 | ランダムに行を抽出 (last modified 2014/07/17)
- イントロ | 一般 | 任意の文字列を行の最初に挿入 (last modified 2014/07/17)
- イントロ | 一般 | 任意のキーワードを含む行を抽出(基礎) (last modified 2014/04/11)
- イントロ | 一般 | ランダムな塩基配列を生成 (last modified 2014/06/16)

①

②

14. GFF3形式のタブ区切りテキストファイル(Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3)に対して、"ID=gene"という文字列が含まれる行全体を出力したい場合:

乳酸菌ゲノム(Lactobacillus casei 12A)のアノテーションファイルです。4.をベースに作成。

```
in_f <- "Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3"#入力ファイル名を
out_f <- "hoge14.txt"                    #出力ファイル名を指定してout_fに格納
param <- "ID=gene"                       #検索したい文字列を指定

#入力ファイルの読み込み
data <- readLines(in_f)                  #
length(data)                             #

#本番(リストファイル中の要素一つ一つに対して
hoge <- sapply(param, grep, x=data)      #
hoge <- unique(hoge)                     #
out <- data[hoge]                        #
length(out)                              #

#ファイルに保存
writeLines(out, out_f)                   #
```

③

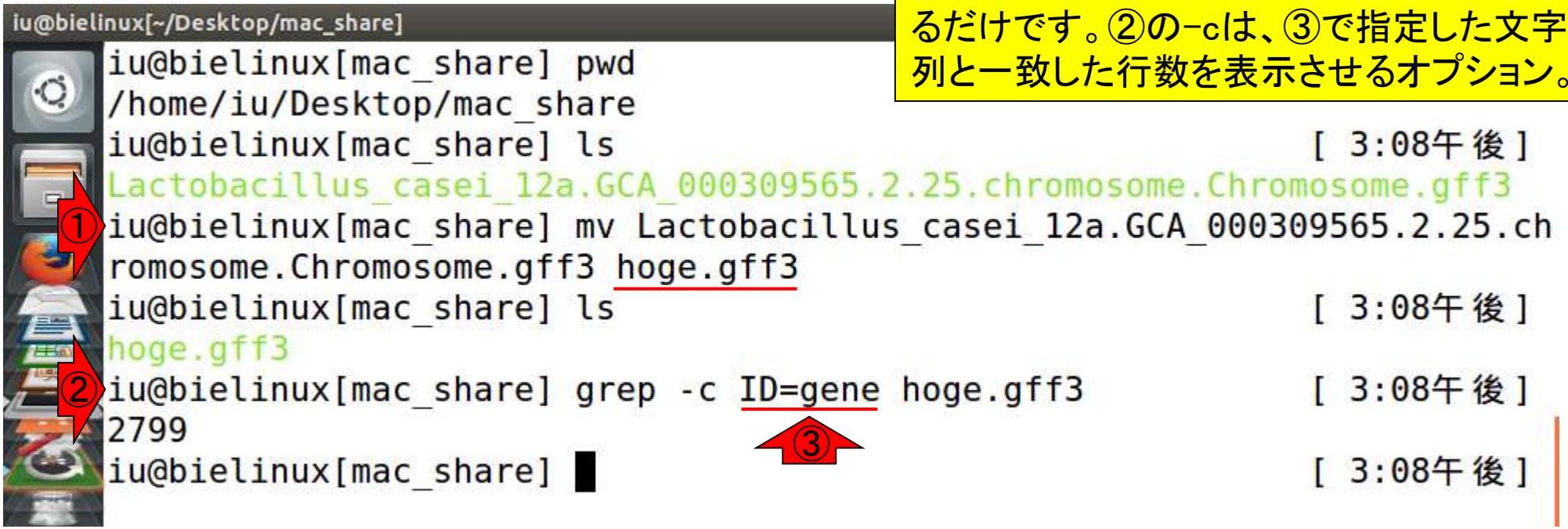
④

R Console

```
> data <- readLines(in_f)                    #in_fで指$
> length(data)                               #オブジェ$
[1] 11081
>
> #本番(リストファイル中の要素一つ一つに対して、要$
> hoge <- sapply(param, grep, x=data)        #リストフ$
> hoge <- unique(hoge)                       #得られるh$
> out <- data[hoge]                          #hogeで指$
> length(out)                                #オブジェ$
[1] 2799
>
> #ファイルに保存
> writeLines(out, out_f)                     #outの中身$
>
```

Linuxを用いた検証例

Linuxをある程度使えるヒトは、普通grepコマンドを使います。①のmvコマンドは、ファイル名が長いので、hoge.gff3にrenameしているだけです。②の-cは、③で指定した文字列と一致した行数を表示させるオプション。



Linuxを用いた検証例

①-vは、一致しないものを表示するオプション。-cvと併用することで、一致しない行数を表示。②Linuxだと他の気になるキーワードも簡単に調査可能。③^をつけることで「Chromosomeを含む行数」ではなく「行頭にChromosomeを含む行数」をカウントすることもできる。

```
iu@bielinux[~/Desktop/mac_share]
iu@bielinux[mac_share] pwd
/home/iu/Desktop/mac_share
iu@bielinux[mac_share] ls
Lactobacillus_casei_12a.GCA_000309565.2.25.chromosome.Chromosome.gff3
iu@bielinux[mac_share] mv Lactobacillus_casei_12a.GCA_000309565.2.25.ch
romosome.Chromosome.gff3 hoge.gff3
iu@bielinux[mac_share] ls                                    [ 3:19午後 ]
hoge.gff3
iu@bielinux[mac_share] grep -c ID=gene hoge.gff3             [ 3:19午後 ]
2799
①iu@bielinux[mac_share] grep -cv ID=gene hoge.gff3           [ 3:19午後 ]
8282
iu@bielinux[mac_share] grep -c gene hoge.gff3                [ 3:19午後 ]
5598
iu@bielinux[mac_share] grep -c CDS hoge.gff3                 [ 3:19午後 ]
2681
iu@bielinux[mac_share] grep -c exon hoge.gff3                [ 3:19午後 ]
2807
③iu@bielinux[mac_share] grep -c ^Chromosome hoge.gff3        [ 3:20午後 ]
11079
iu@bielinux[mac_share]                                       [ 3:20午後 ]
```

②

④

Contents

- パッケージ
  - CRANとBioconductor
  - 推奨パッケージインストール手順のおさらい
  - Bioconductor概観 → ゲノム配列パッケージ(BSgenome)
  - ヒトゲノム情報パッケージの解析
- プロモーター配列取得(sessionInfo、バージョンの違い)
  - Rパッケージ(ゲノムとアノテーションパッケージの併用)
  - FASTA形式ファイルとGFF3形式ファイル
  - データの型
- FASTQファイルの各種解析(LinuxとRを相補的に活用)
  - Linux (FastQC)とR (ShortRead)で同じ:Overrepresented sequences項目
  - Linux (FastQC)とR (QuasR)で異なる見栄え:Per base sequence content項目。FastQCに--nogroupというオプションがあることを知る。
  - FastQCのオプション (デフォルトと--nogroupあり)の違いによるKmer Content項目の結果の違い → Rで検証

FASTQファイル解析

①ファイル形式の変換(FASTQ → FASTA)や、FastQCのいくつかの項目と同じ解析をRでも実行可能である:②Overrepresented sequences、③Per base sequence content(リードのposition ごとの出現確率)、④Kmer Content。

- イントロ | NGS | 読み込み | FASTQ形式 | 基礎 (last modified 2015/06/24)
- イントロ | NGS | 読み込み | FASTQ形式 | 応用 (last modified 2015/06/18)
- イントロ | NGS | 読み込み | FASTQ形式 | description行の記述を整形 (last mo
- イントロ | NGS | 読み込み | Illuminaの* seq.txt (last modified 2013/06/13)
- イントロ | NGS | 読み込み | Illuminaの* qseq.txt (last modified 2013/06/17)
- イントロ | ファイル形式の変換 | について (last modified 2014/06/09)
- イントロ | ファイル形式の変換 | BAM --> BED (last modified 2014/06/21)
- イントロ | ファイル形式の変換 | FASTQ --> FASTA (last modified 2015/06/15)
- イントロ | ファイル形式の変換 | Genbank --> FASTA (last modified 2014/03/10)
- イントロ | ファイル形式の変換 | qseq --> FASTA (last modified 2013/06/17)
- イントロ | ファイル形式の変換 | qseq --> Illumina FASTQ (last modified 2013/06/17)
- イントロ | ファイル形式の変換 | qseq --> Sanger FASTQ (last modified 2013/08/19)
- 前処理 | クオリティコントロール | について (last modified 2015/06/25)
- 前処理 | クオリティチェック | QuasR(Gaidatzis_2015) (last modified 2015/06/15)
- 前処理 | クオリティチェック | qrqc (last modified 2014/06/17)
- 前処理 | クオリティチェック | PHREDスコアに変換 (last modified 2013/06/18)
- 前処理 | クオリティチェック | 配列長分布を調べる (last modified 2015/06/22)
- 前処理 | クオリティチェック | Overrepresented sequences | ShortRead(Morgan_2009) (last modified 2015/07/21)
- 前処理 | トリミング | ポリA配列除去 | ShortRead(Morgan_2009) (last modified 2014/06/11)
- 前処理 | トリミング | アダプター配列除去(基礎) | QuasR(Gaidatzis_2015) (last modified 2015/06/26)推奨
- 前処理 | トリミング | アダプター配列除去(基礎) | girafe(Toedling_2010) (last modified 2014/06/11)
- 前処理 | トリミング | アダプター配列除去(基礎) | ShortRead(Morgan_2009) (last modified 2014/06/21)
- 前処理 | トリミング | アダプター配列除去(応用) | QuasR(Gaidatzis_2015) (last modified 2015/06/26)推奨
- 前処理 | トリミング | アダプター配列除去(応用) | ShortRead(Morgan_2009) (last modified 2014/06/18)
- 前処理 | トリミング | 指定した末端塩基数だけ除去 (last modified 2015/06/29) NEW

FastQC実行結果

①頻出する配列をリストアップ。②トップは「CCCCGGTATA…」という50塩基の配列で14,383回出現。Percentageは1.4383%。全部で100万リードなので妥当。オリジナル107 bpのうち最初の50 bpで解析している。復習

FastQC Report

Summary

- Basic Statistics
- Per base sequence quality
- Per tile sequence quality
- Per sequence quality scores
- Per base sequence content
- Per sequence GC content
- Per base N content
- Sequence Length Distribution
- Sequence Duplication Levels
- Overrepresented sequences ①
- Adapter Content
- Kmer Content

Overrepresented sequences

②

Sequence	Count	Percentage	Possible Source
CCCCGGTATATTTTCGGCGCAGTGCCACTCGACTAGTGAGCTATTACGCA	14383	1.4383	No Hit
GGCCTATTCACTGCGGCTGACCTTGCGGTCAGCACCCCTTCTTCCGAAGT	11044	1.1044	No Hit
GTGCTTTTCACCTTTCCCTCACGGTACTGGTTCACTATCGGTCACTAGGG	8892	0.8892000000000001	No Hit
CCCGGTATATTTTCGGCGCAGTGCCACTCGACTAGTGAGCTATTACGCAC	8474	0.8474	No Hit
GTCACTAGGGAGTATTTAGCCTTGGGAGATGGTCCTCCCGGATTCCGACG	8189	0.8189	No Hit
GCCGGCATTCTCACTTCTAAGCGCTCCAGCCGTCCTCACGATCGACCTTC	8132	0.8132	No Hit
GTCCAGTCCTACAACCCCGAGAAGCAAGCTTCTCGGTTTGGGCTCTTCCC	6663	0.6663	No Hit
GTCGGTTTGCGGTACGGGTAGTTTATTTCTCACTAGAAGCTTTTCTTGGC	6411	0.6411	No Hit
GGTCACTTGGTTTCGGGTCTACATCTGCTTACTCATTCGCCCTGTTCAGA	5502	0.5502	No Hit
GCCGGCATTCTCACTTCTAAGCGCTCCAGCCGTCCTCACGATCAACCTTC	4845	0.48450000000000004	No Hit
CCCTCCATCGCTTAAACAAAATAAACTAGTGCAGGAATCTCAACCTGCTT	4395	0.43949999999999995	No Hit
CCGGTATATTTTCGGCGCAGTGCCACTCGACTAGTGAGCTATTACGCACT	4385	0.4385	No Hit
CCCGCGTCTGCCGCCGGCCAGCTATGTATTCACTGACAAGCAATACACTG	4366	0.4366	No Hit
CCACAGTTTCGGTATTATGCTTAGCCCCGGTATATTTTCGGCGCAGTGCC	4314	0.4314	No Hit
CTGGGCTGTTCCCCTTTCGACAATGGACCTTATCGCTCACTGTCTGACTC	4113	0.41130000000000005	No Hit
CCGCCGTACTCAGGATCCTGGACGGAGGGTTCGACGTTTCGCTTACAGGG	4081	0.4081	No Hit
CCGGCATTCTCACTTCTAAGCGCTCCAGCCGTCCTCACGATCGACCTTCA	3846	0.3846	No Hit
GTAGGTCACTTGGTTTCGGGTCTACATCTGCTTACTCATTCGCCCTGTTC	3823	0.3823	No Hit

ShortRead実行結果

subseqとtable関数の併用で同じ結果が得られる例として提示。これもただの復習。

- 前処理 | クオリティチェック | 配列長分布を調べる (last modified 2015/06/22)
- 前処理 | クオリティチェック | Overrepresented sequences | ShortRead(Morgan 2009)
- 前処理 | トリミング | ポリA配列除去 | ShortRead(Morgan 2009) (last modified 2014/

①

前処理 | クオリティチェック | Overrepresented sequences | ShortRead

ShortRead パッケージを用いてリードの種類ごとの出現回数をを得るやり方を示します。FastQCの Over

②

4. gzip圧縮FASTQ形式ファイル(SRR616268sub_1.fastq.gz)の場合:

乳酸菌RNA-seqデータSRR616268の最初の100万リード分(約73MB)です。長さは全て107 bpです。3.と基本的に同じですが、FastQCのデフォルトと同じく、最初の50bp分のみで解析するやり方です。「イントロ | 一般 | 指定した範囲の配列を取得」のテクニ

```
in_f <- "SRR616268sub
out_f <- "hoge4.txt"
param <- c(1, 50)

#必要なパッケージをロー
library(ShortRead)

#入力ファイルの読み込み
fastq <- readFastq(in
fasta <- sread(fastq)
fasta

#前処理(指定した範囲の
fasta <- subseq(fasta
```

R Console

```
> head(out)                                    #確認してるだけ$
fasta
CCCCGGTATATTTTCGGCGCAGTGCCACTCGACTAGTGAGCTATTACGCA
                                                14383
GGCCTATTCACTGCGGCTGACCTTGCGGTCAGCACCCCTTCTTCCGAAGT
                                                11044
GTGCTTTTCACCTTTCCCTCACGGTACTGGTTCACTATCGGTCACTAGGG
                                                 8892
CCCGGTATATTTTCGGCGCAGTGCCACTCGACTAGTGAGCTATTACGCAC
                                                 8474
GTCACTAGGGAGTATTTAGCCTTGGGAGATGGTCCTCCCGGATTCCGACG
                                                 8189
GCCGGCATTCTCACTTCTAAGCGCTCCAGCCGTCCTCACGATCGACCTTC
                                                 8132

>
> #ファイルに保存
> tmp <- cbind(names(out), out)                #保存したい情報$
> write.table(tmp, out f, sep="\t", append=F, quote=F, ro$
```

FastQC実行結果

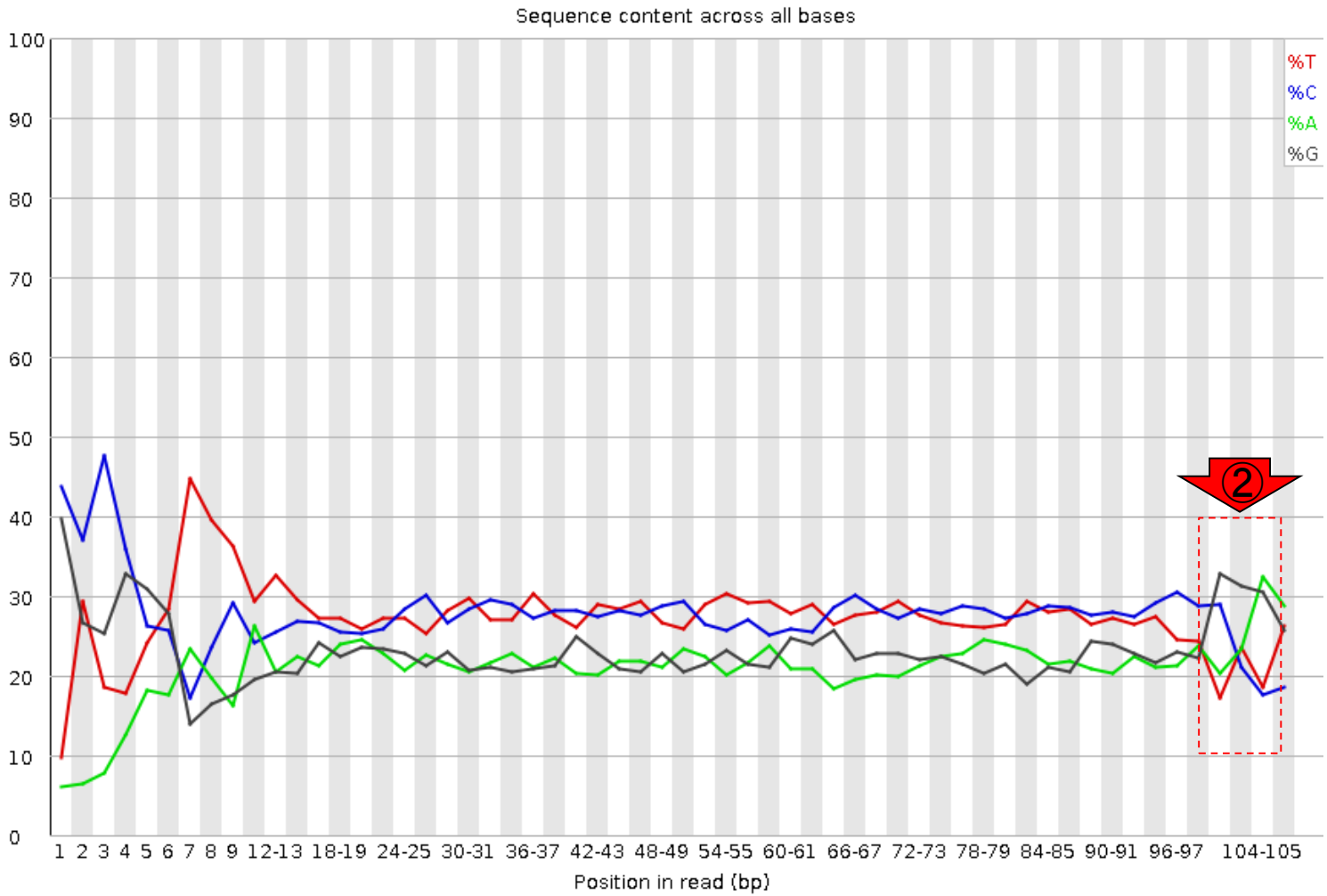
①ポジションごとの塩基の出現確率。FastQC (ver. 0.11.3)のデフォルトオプションでの実行結果は、最初の10塩基までは塩基ごとに、それ以降は適当に数塩基分を平均化した出現確率を表示(していることにQuasR実行結果と比較することで後に気づいた)。②赤枠部分に注目!

FastQC Report

Summary

- Basic Statistics
- Per base sequence quality
- Per tile sequence quality
- Per sequence quality scores
- Per base sequence content ①
- Per sequence GC content
- Per base N content
- Sequence Length Distribution
- Sequence Duplication Levels
- Overrepresented sequences
- Adapter Content
- Kmer Content

Per base sequence content



QuasR実行結果

①QuasRというパッケージは、Quality Control部分だけで比較すると、FastQCの簡易版のような位置づけ。②例題4実行結果のhoge4.pdfの一部が右下プロットで、FastQCのPer base sequence contentと同じもの。③の赤枠部分に注目!

- イントロ | ファイル形式の変換 | qseq --> Sanger FASTQ (last modified 2013/08/19)
- 前処理 | クオリティコントロール | について (last modified 2015/06/25)
- 前処理 | クオリティチェック | QuasR(Gaidatzis_2015) (last modified 2015/06/15)
- 前処理 | クオリティチェック | qrqc (last modified 2014/0[illegible]17)
- 前処理 | クオリティチェック | PHREDスコアに変換 (last modified 2013/06/18)
- 前処理 |
- 前処理 |
- 前処理 |

①

前処理 | クオリティチェック | QuasR(Gaidatzis_2015)

QuasRパッケージを用いてQCレポートファイルを出力するやり方を示します。FastQCのR版のようなqrqcよりも[illegible]

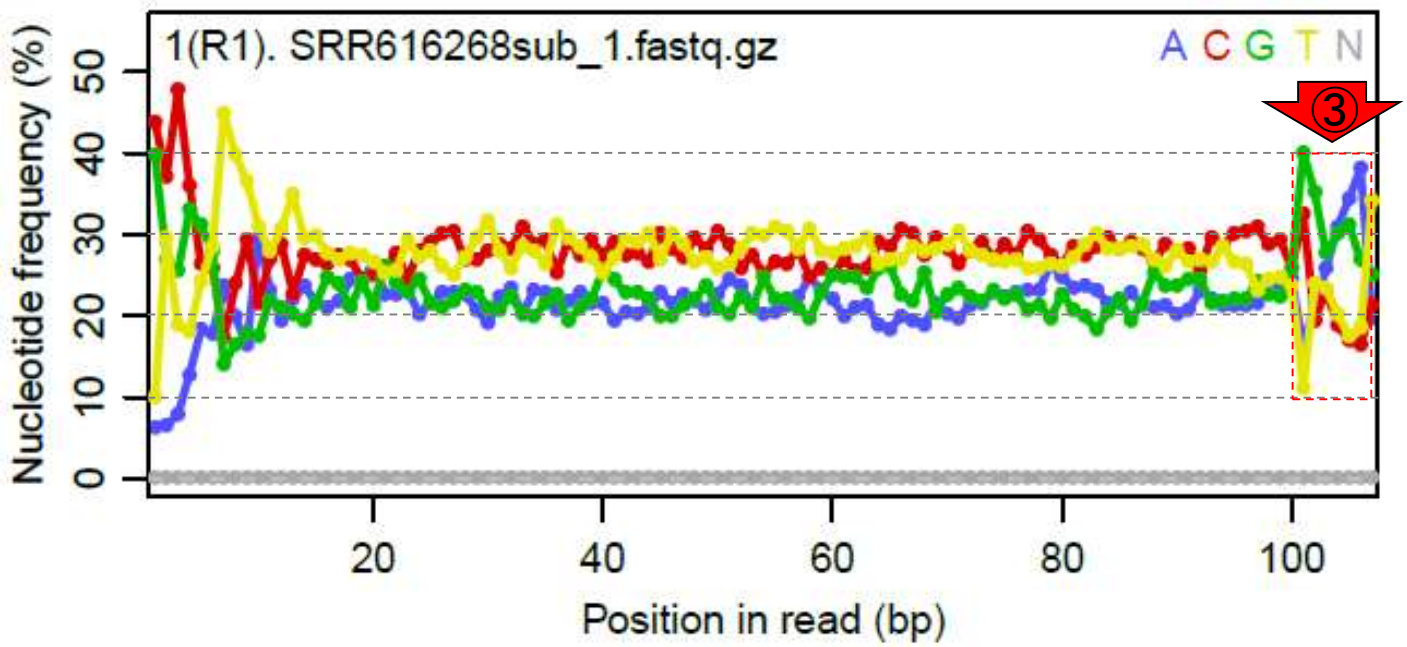
「ファイル

②

4. FASTQ形式ファイル(SRR616268sub_1.fastq.gz)の場合:

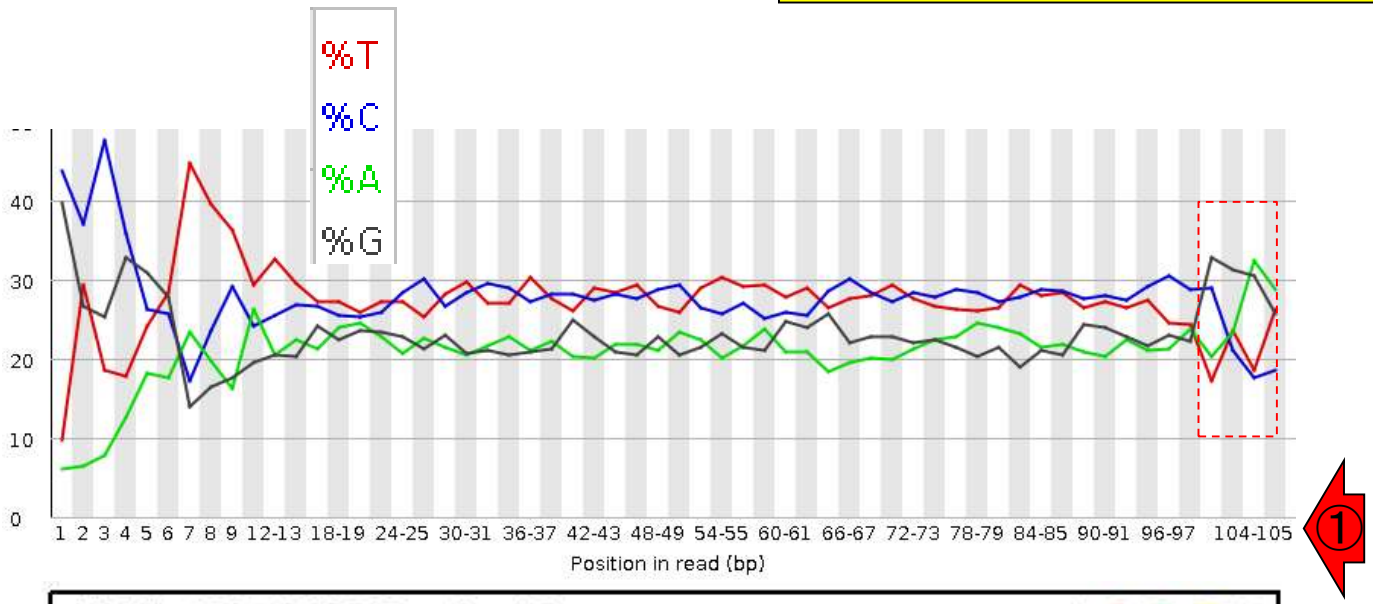
乳酸菌RNA-seqデータSRR616268の最初の100万リード分(約73MB)です。

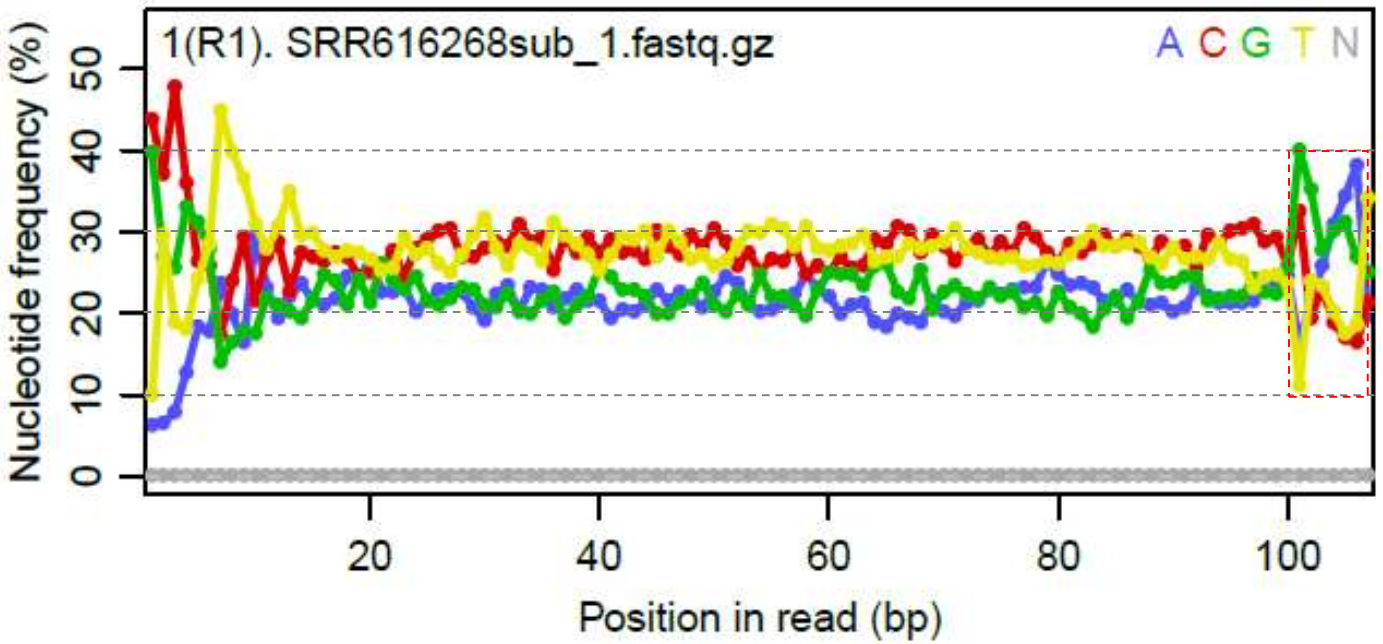
```
in_f <- "SRR616268sub_1.fastq.gz"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge4.pdf"                   #出力ファイル名を指定してout_fに格納

#必要なパッケージを
library(QuasR)

#本番
qQCReport(in_f, p
```



FastQC vs. QuasR

①FastQCのほうの横軸の数値をよく眺めると、こちらは2塩基分づつ平均化したものをプロットしているのではないかと思うに至る



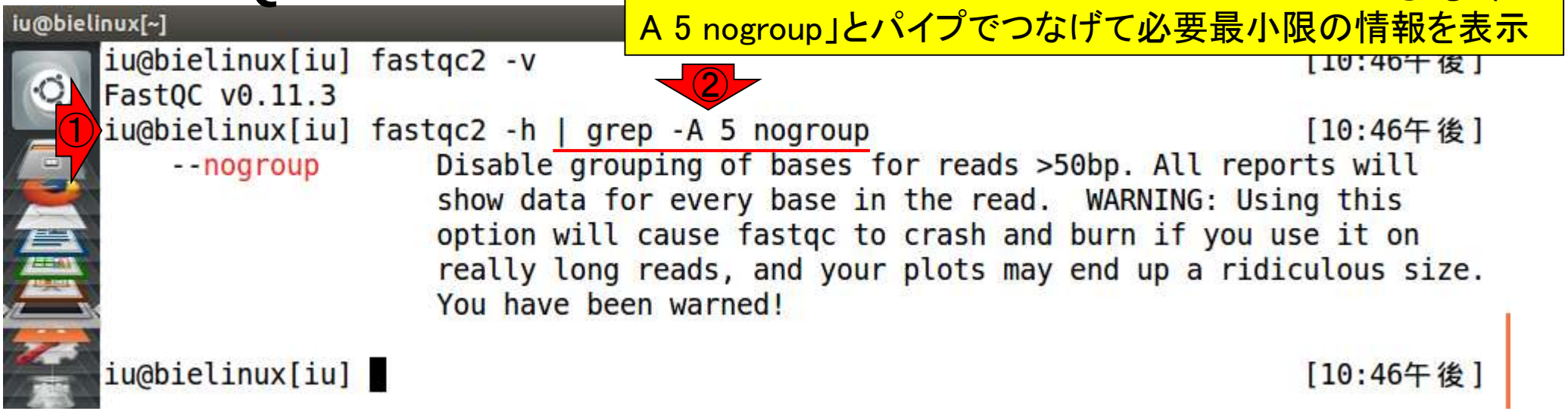


Contents

- パッケージ
  - CRANとBioconductor
  - 推奨パッケージインストール手順のおさらい
  - Bioconductor概観 → ゲノム配列パッケージ(BSgenome)
  - ヒトゲノム情報パッケージの解析
- プロモーター配列取得(sessionInfo、バージョンの違い)
  - Rパッケージ(ゲノムとアノテーションパッケージの併用)
  - FASTA形式ファイルとGFF3形式ファイル
  - データの型
- FASTQファイルの各種解析(LinuxとRを相補的に活用)
  - Linux (FastQC)とR (ShortRead)で同じ:Overrepresented sequences項目
  - Linux (FastQC)とR (QuasR)で異なる見栄え:Per base sequence content項目。FastQCに--nogroupというオプションがあることを知る。
  - FastQCのオプション (デフォルトと--nogroupあり)の違いによるKmer Content項目の結果の違い → Rで検証

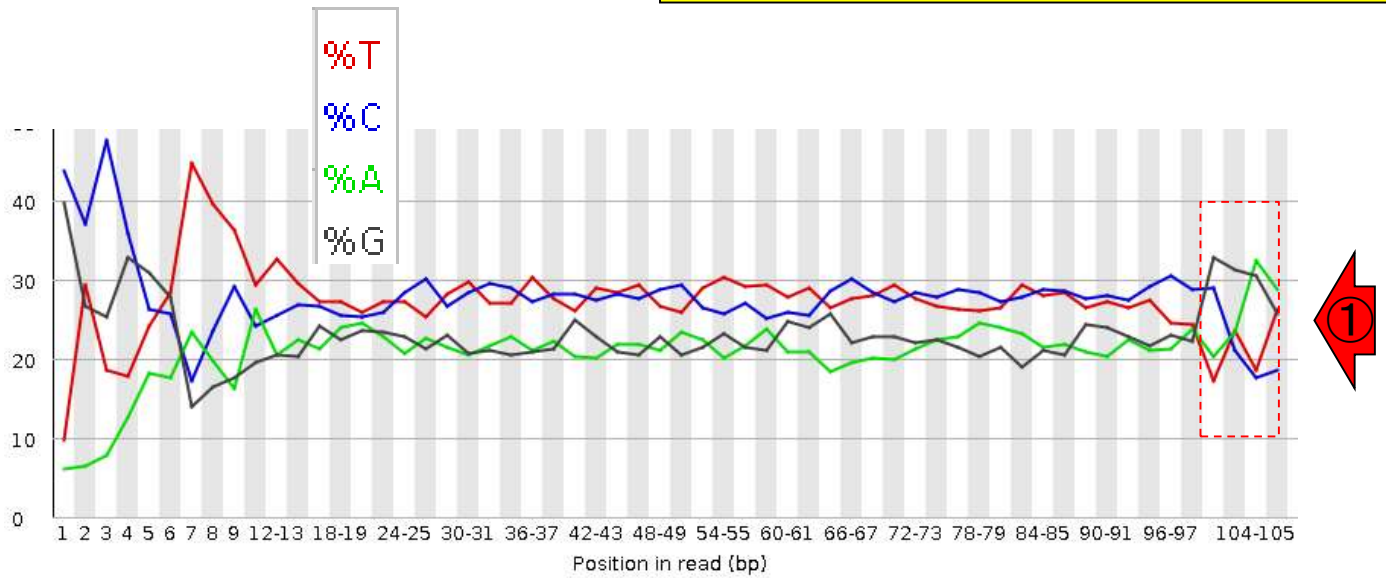
FastQCマニュア

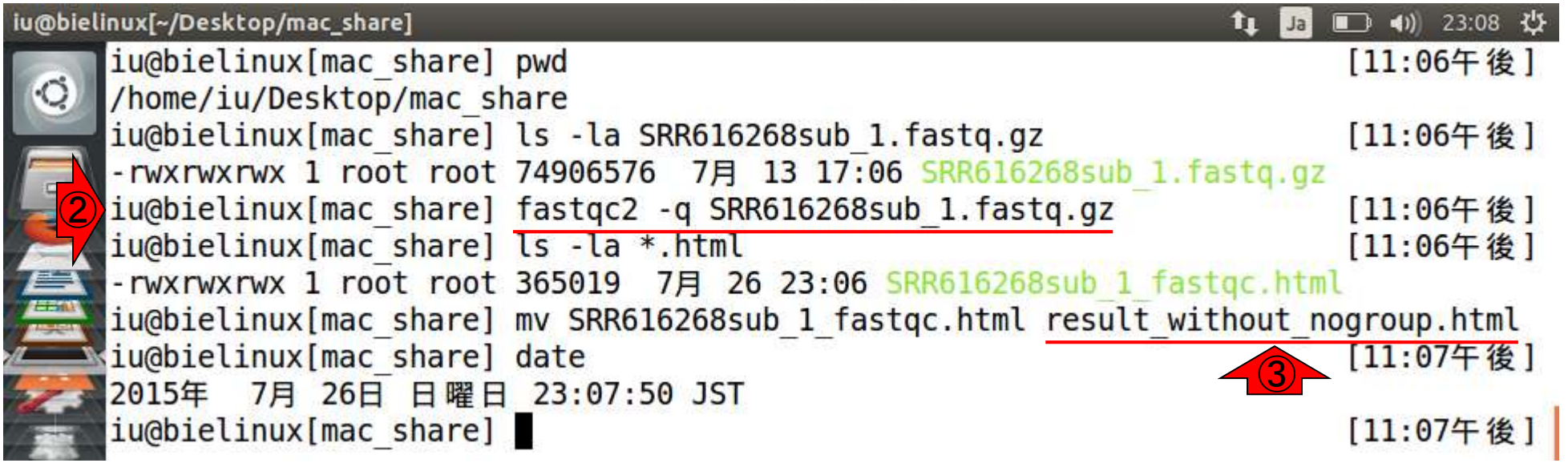
①fastqc2コマンドのマニュアルを表示。予めマニュアル全体を眺めて、--nogroupオプションを用いれば平均化(grouping)せず塩基ごとに表示できることが分かったうえで、②「grep -A 5 nogroup」とパイプでつなげて必要最小限の情報を表示



FastQC

①このFastQCレポートHtmlは、②FastQCをデフォルトオプションで実行した結果。③得られたHtmlファイル名をresult_without_nogroup.htmlに変更。



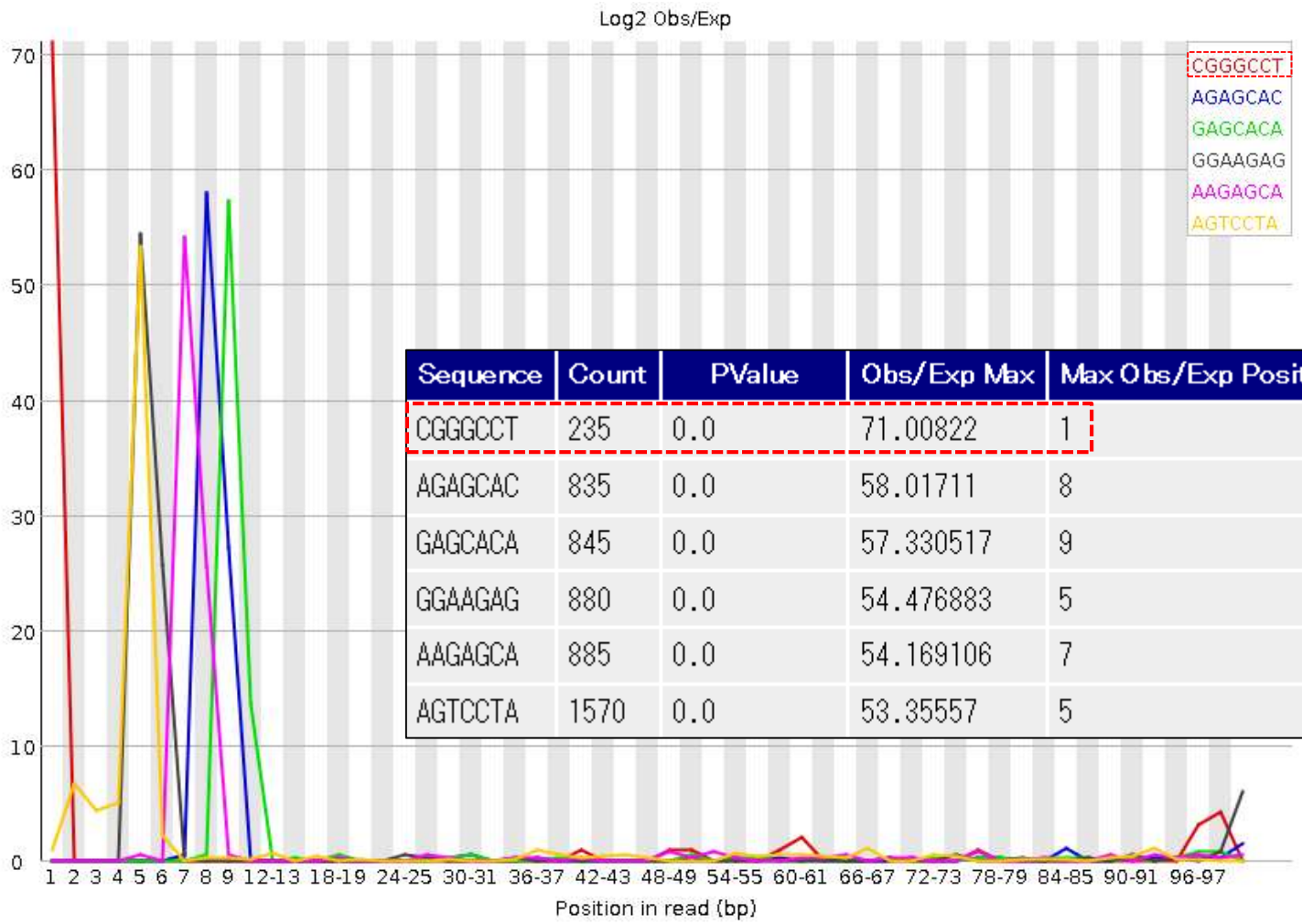


①Kmer Contentの項目はこんな感じでした。(2015.06.23の講義資料も参照のこと)

FastQC

Summary

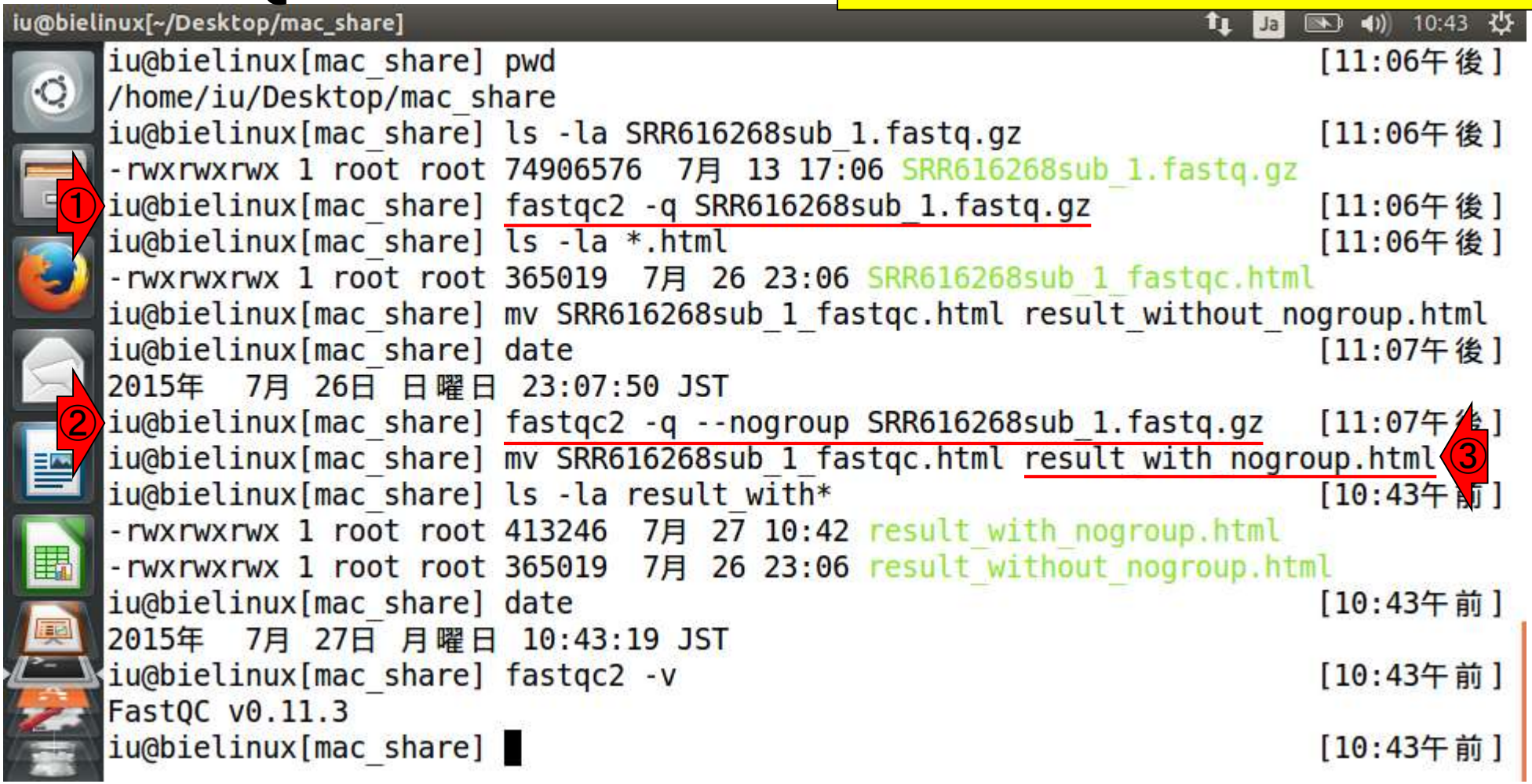
- Basic Statistics
- Per base sequence quality
- Per tile sequence quality
- Per sequence quality scores
- Per base sequence content
- Per sequence GC content
- Per base N content
- Sequence Length Distribution
- Sequence Duplication Levels
- Overrepresented sequences
- Adapter Content
- Kmer Content ①



Sequence	Count	PValue	Obs/Exp Max	Max Obs/Exp Position
CGGGCCT	235	0.0	71.00822	1
AGAGCAC	835	0.0	58.01711	8
GAGCACA	845	0.0	57.330517	9
GGAAGAG	880	0.0	54.476883	5
AAGAGCA	885	0.0	54.169106	7
AGTCCTA	1570	0.0	53.35557	5

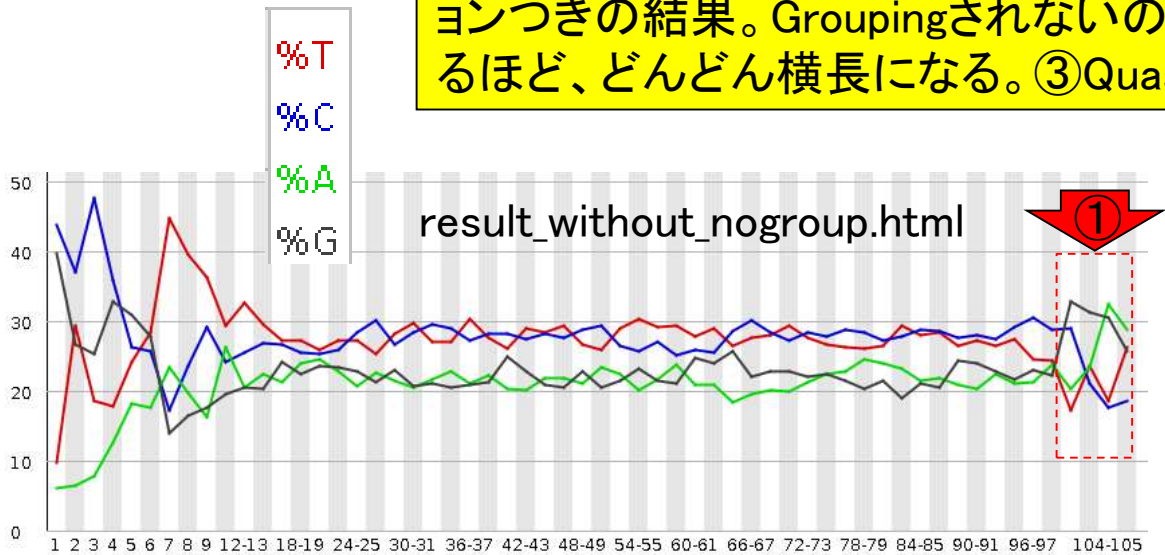
FastQC

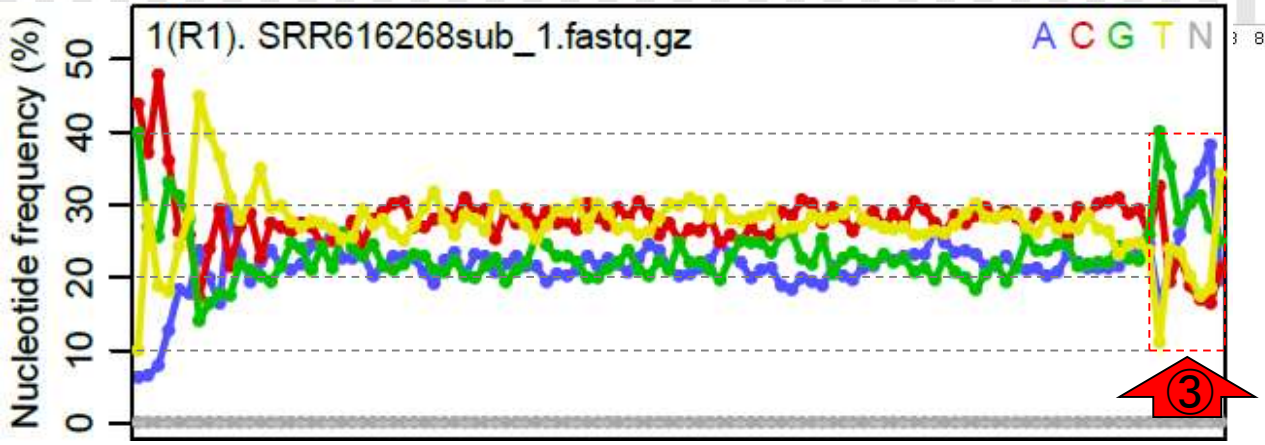
FastQC (ver. 0.11.3)を①デフォルトで実行、②--nogroupオプションつきで実行。③得られたhtmlファイル名をresult_with_nogroup.htmlに変更。



FastQC

①FastQCデフォルトの結果。②FastQC --nogroupオプションつきの結果。Groupingされないので、配列長が長くなるほど、どんどん横長になる。③QuasRデフォルトの結果





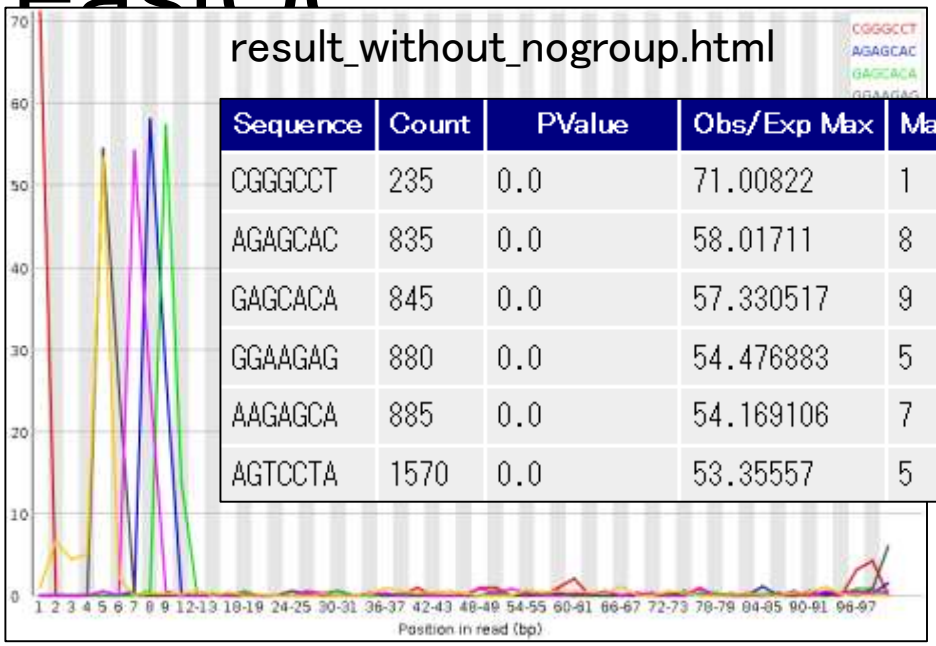
Contents

- パッケージ
  - CRANとBioconductor
  - 推奨パッケージインストール手順のおさらい
  - Bioconductor概観 → ゲノム配列パッケージ(BSgenome)
  - ヒトゲノム情報パッケージの解析
- プロモーター配列取得(sessionInfo、バージョンの違い)
  - Rパッケージ(ゲノムとアノテーションパッケージの併用)
  - FASTA形式ファイルとGFF3形式ファイル
  - データの型
- FASTQファイルの各種解析(LinuxとRを相補的に活用)
  - Linux (FastQC)とR (ShortRead)で同じ:Overrepresented sequences項目
  - Linux (FastQC)とR (QuasR)で異なる見栄え:Per base sequence content項目。FastQCに--nogroupというオプションがあることを知る。
  - FastQCのオプション (デフォルトと--nogroupあり)の違いによるKmer Content項目の結果の違い → Rで検証

FastQC

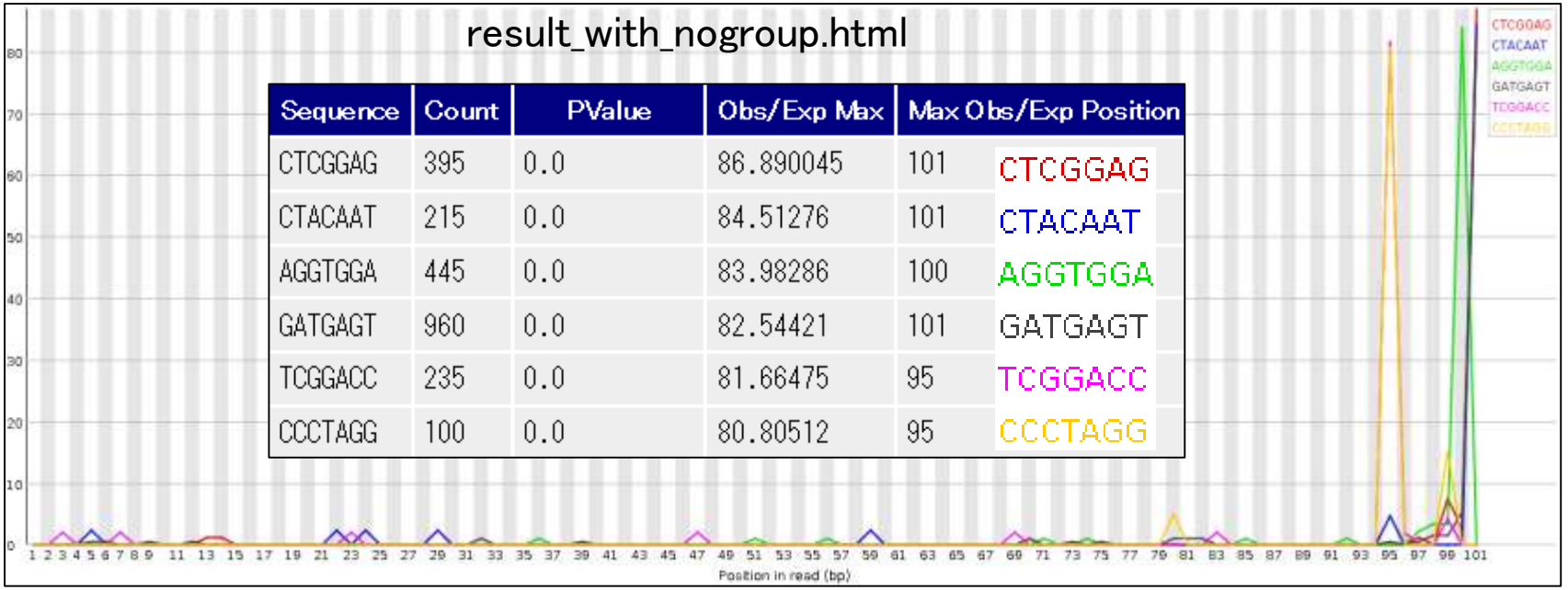
Kmer Contentの項目を比較。違いは--nogroupの有無のみ…何か変。結論としては、--nogroupをつけないときの結果は(門田の感覚では)バグ

result_without_nogroup.html



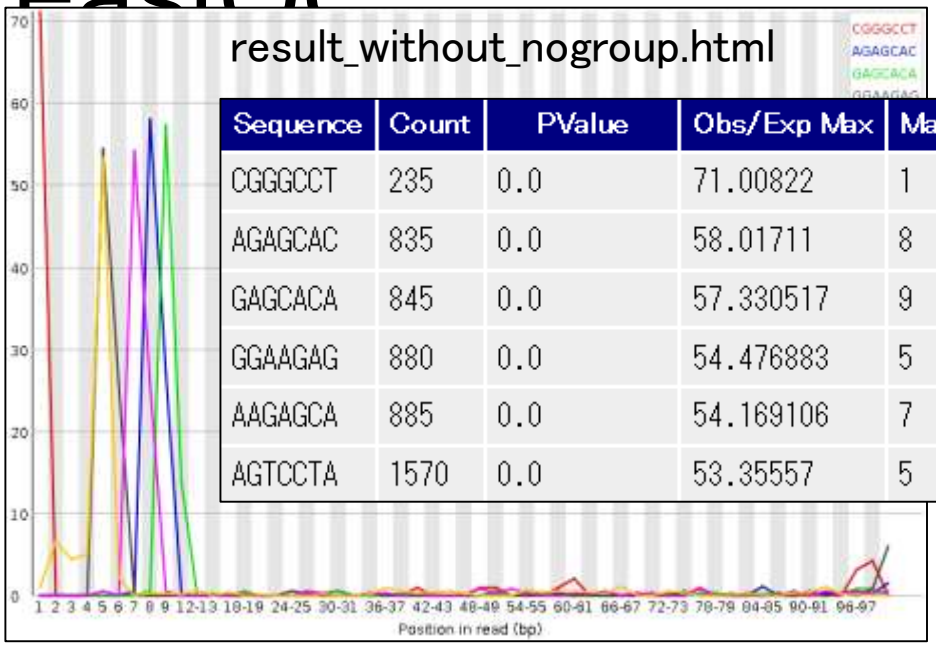
Sequence	Count	PValue	Obs/Exp Max	Max Obs/Exp Position
CGGGCCT	235	0.0	71.00822	1
AGAGCAC	835	0.0	58.01711	8
GAGCACA	845	0.0	57.330517	9
GGAAGAG	880	0.0	54.476883	5
AAGAGCA	885	0.0	54.169106	7
AGTCCTA	1570	0.0	53.35557	5

result_with_nogroup.html



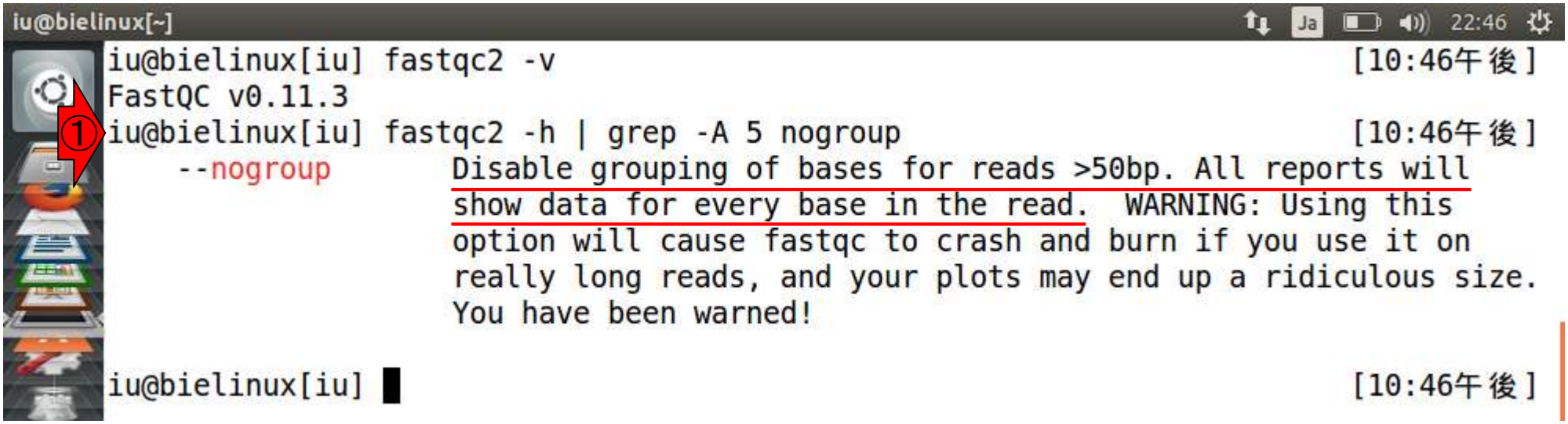
Sequence	Count	PValue	Obs/Exp Max	Max Obs/Exp Position
CTCGGAG	395	0.0	86.890045	101
CTACAAT	215	0.0	84.51276	101
AGGTGGA	445	0.0	83.98286	100
GATGAGT	960	0.0	82.54421	101
TCGGACC	235	0.0	81.66475	95
CCCTAGG	100	0.0	80.80512	95

FastQC



Sequence	Count	PValue	Obs/Exp Max	Max Obs/Exp Position	
CGGGCCT	235	0.0	71.00822	1	CGGGCCT
AGAGCAC	835	0.0	58.01711	8	AGAGCAC
GAGCACA	845	0.0	57.330517	9	GAGCACA
GGAAGAG	880	0.0	54.476883	5	GGAAGAG
AAGAGCA	885	0.0	54.169106	7	AAGAGCA
AGTCCTA	1570	0.0	53.35557	5	AGTCCTA

開発者側の論理としては、デフォルト(--nogroupオプションをつけない)だとKmer Content解析結果も塩基をgroupingする、という思想なのかもしれない。ただしこれは実質的に5'側が上位に来る傾向となり、3'側に除去すべきアダプターやプライマー配列が存在していてもそれに気づけない(実際門田がこれにハマった)。

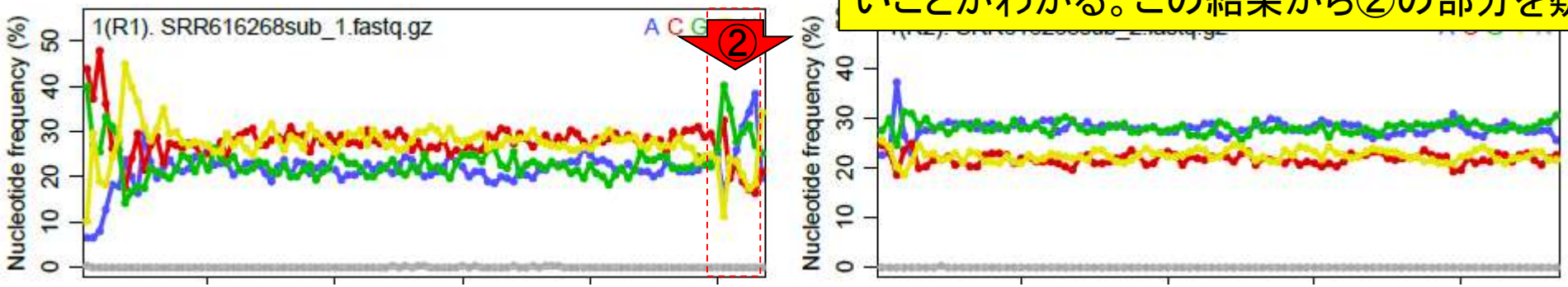


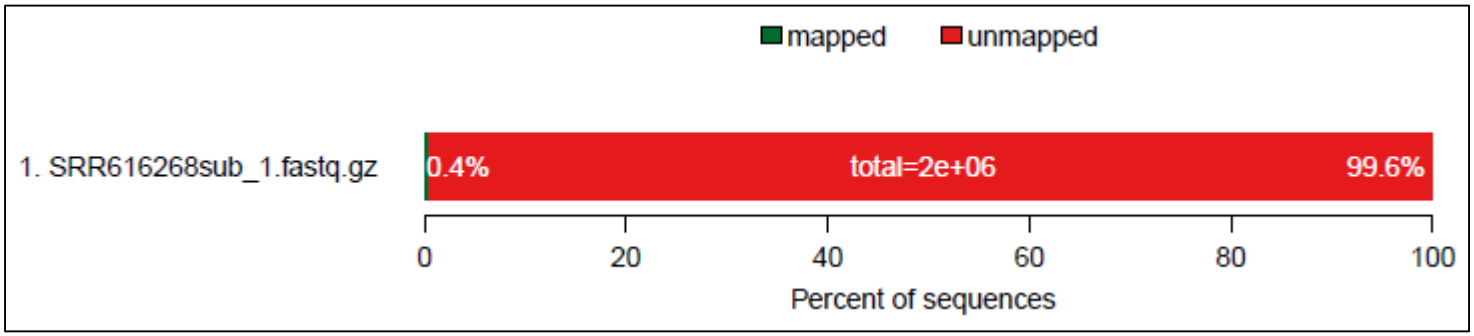
様々な状況証拠を収集

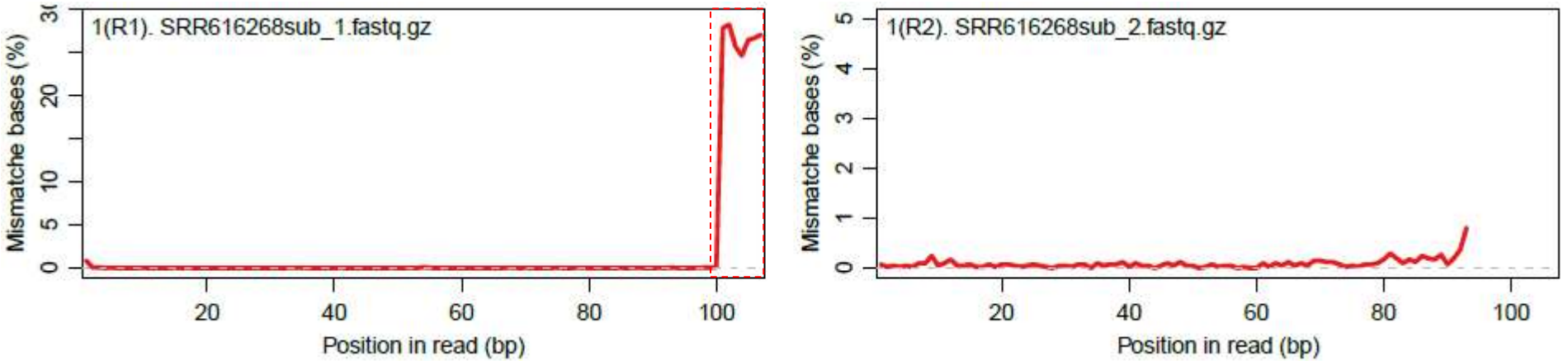
- リファレンス配列がある場合（マッピング）
  1. 診断：(比較的多めのミスマッチ数で)マッピングを行い、マップされたものの中でミスマッチがあったポジションの分布を眺める。もし3’側にミスマッチの偏りがあれば、それはプライマー/アダプター配列由来塩基という判断を下せる。
  2. 対策：該当部分の塩基を一定数トリム
  3. 検証：マップ率が向上するかどうかを調べる。
- リファレンス配列がない場合（アセンブル）
  1. 診断：paired-endデータの場合は、single-endデータとしてde novo assembleするとある程度アセンブルされる（一定数のコンティグが得られる）一方、paired-endデータで実行すると途端にアセンブルされなくなる。
  2. 対策：1塩基づつトリムしてはアセンブル、みたいなことを繰り返して、一定範囲のデータを得る。
  3. 検証：アセンブルが劇的に改善されるところが出てくる(はず)。

QuasR(トリム前)

QuasRはマッピングもできる。乳酸菌ゲノム配列にマッピングした結果のPDFファイルの一部を表示。
①ほとんどマップされていない。100万リード中、わずか0.4% (約4,000リード)しかマップされていないことがわかる。この結果から②の部分を疑う。

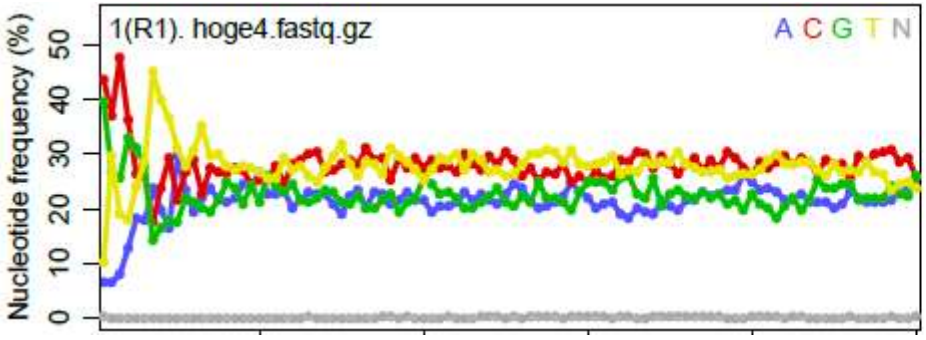


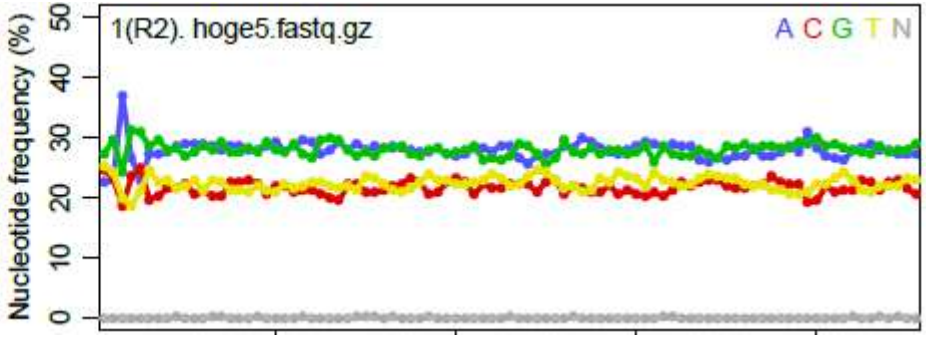


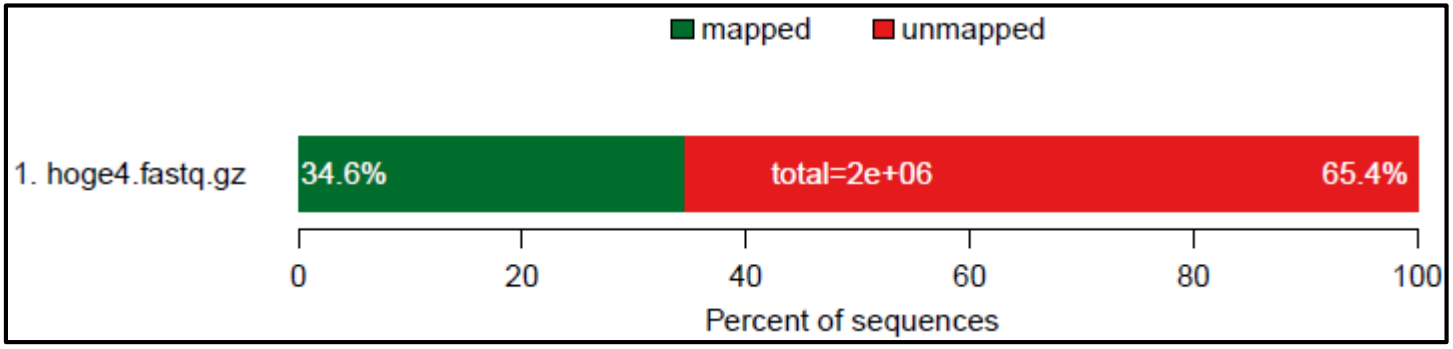


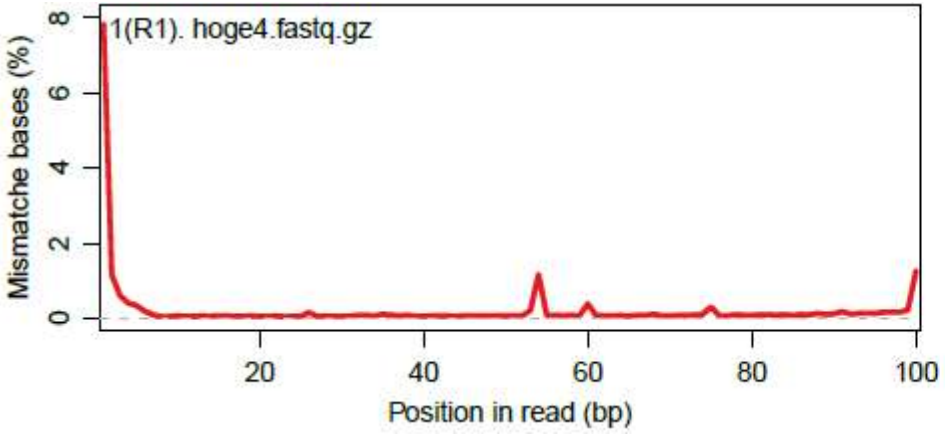
これだけ結果が変わります。かなりイケてます

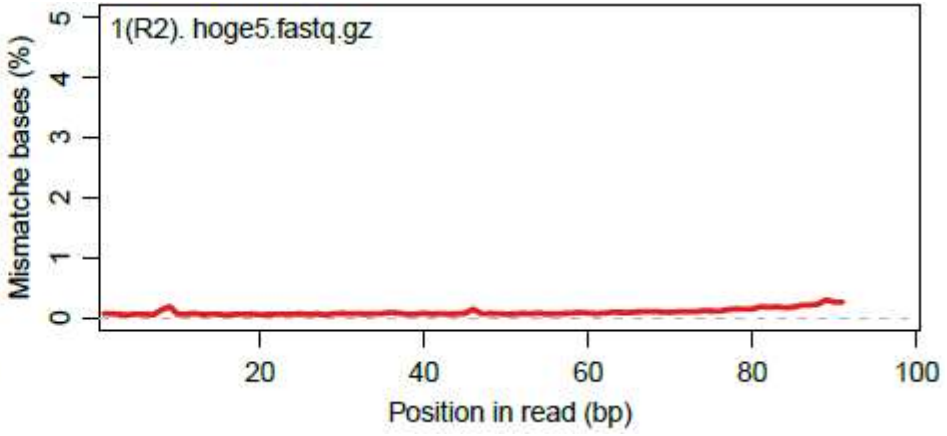
QuasR(トリム後)











②例題4。Rで実行可能です

末端塩基のトリム

- 前処理 | トリミング | アダプター配列除去(応用) | QuasR(Gaidatzis 2015) (last modified 2
- 前処理 | トリミング | アダプター配列除去(応用) | ShortRead(Morgan 2009) (last modified
- 前処理 | トリミング | 指定した末端塩基数だけ除去 (last modified 2015/06/29) NEW
- 前処理 | フィルタリング | PHREDスコアが低い塩基をNに置換 (last modified 2014/03/03)
- 前処理 | フィルタリング | PHREDスコアが低い配列(リード)を除去 (last modified 2014/08
- 前処理 | フィ
- 前処理 | フィ
- 前処理 | フィ
- 前処理 | フィ
- 前処理 | フィ
- 前処理 | フィ

①

前処理 | トリミング | 指定した末端塩基数だけ除去 NEW

3'末端を指定塩基数分だけトリムするやり方を示します。
「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ペ。

1. multi-FAST

②

イントロ | 一般

```
in_f <- "hog
out_f <- "ho
param_trim <

#必要なパッケ
library(Bios

#入力ファイル
fasta <- rea
fasta

#本番
hoge <- widt
```

4. FASTQ形式ファイル(SRR616268sub_1.fastq.gz)の場合:

乳酸菌RNA-seqデータSRR616268の最初の100万リード分(約73MB; 74,906,576 bytes)です。paired-endのforward側です。長さは全て107 bpです。3'側の7 bp分をトリムするので、出力ファイル中の長さは100 bplになります。

```
in_f <- "SRR616268sub_1.fastq.gz"       #入力ファイル名を指定してin_fに格納
out_f <- "hoge4.fastq.gz"               #出力ファイル名を指定してout_fに格納
param_trim <- 7                         #3'末端のトリムしたい塩基数を指定

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

#入力ファイルの読み込み
fastq <- readFastq(in_f)                #in_fで指定したファイルの読み込み
sread(fastq)                            #配列情報を表示

#本番
hoge <- width(sread(fastq)) - param_trim#トリム後のend位置情報を取得
hoge[hoge < 1] <- 1                     #トリム後のend位置が1未満の場合には1にしている
hoge1 <- DNAStringSet(sread(fastq), start=1, end=hoge)#sread(fastq)から指定した範囲を抽
hoge2 <- BStringSet(quality(quality(fastq)), start=1, end=hoge)#quality(quality(fasto
fasta <- ShortRead0(hoge1, hoge2, id(fastq))#ShortRead0というクラスオブジェクトを作成し
```

末端塩基のトリム

①トリム前のリード情報。②7塩基をトリムするので、③赤枠部分をトリムした結果が返される(はず)。

4. FASTQ形式ファイル(SRR616268sub_1.fastq.gz)の場合:

乳酸菌RNA-seqデータSRR616268の最初の100万リード分(約73MB; 74,906,576 bytes)です。paired-endのforward側です。長さは全て107 bpです。3'側の7 bp分をトリムするので、出力ファイル中の長さは100 bplになります。

```
in_f <- "SRR616268sub_1.fastq.gz"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge4.fastq.gz"              #出力ファイル名を指定してout_fに格納
param_trim <- 7                        #3'末端のトリムしたい塩基数を指定

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

#入力ファイルの読み込み
fastq <- readFastq(in_f)
sread(fastq)

#本番
hoge <- width(sread(fastq))
hoge[hoge < 1] <- 1
hoge1 <- DNAStringSet(sread(
hoge2 <- BStringSet(quality(
```

②

①

R Console

```
> fastq <- readFastq(in_f)                #in_fで指定したファイ$
> sread(fastq)                            #配列情報を表示
  A DNAStringSet instance of length 1000000
          width seq
      [1]   107 AGCCCGACTTTCGTCCCTGCTC...TTCTGAGGGAACCTTCTCTAAC
      [2]   107 GATCTGGGCTGTTCCCCTTTCG...CTGAATTCAGTAACCTCCGAAA
      [3]   107 CCGGTATATTTTCGGCGCAGTG...GTTGTCTGTGCAACGATCTTAC
      [4]   107 CTTCGATACCGCCCAAGAACTT...GCCATCTGTGGCCATAAAGACC
      [5]   107 CCCCGGTATATTTTCGGCGCAG...TAGTTGTCTGTGCAAGGTCCAT
      ...   ... ...
 [999996]   107 CCCCGGTATATTTTCGGCGCAG...TAGTTGTCTGTGCAAGGGTTAC
 [999997]   107 TTCGGGTCTACATCTGCTTACT...AACGTAACTCGCCGGGTGAAAT
 [999998]   107 CGTCCATCCCGGTCCTCTCGTA...CGTTCTGAACCCAGCCAAGAGT
 [999999]   107 CTAGGGAGTATTTAGCCTTGGG...AGGGTTCGACGTTTCCTGGACA
[1000000]   107 GCCTTGTCAATCAAGGTGAGCA...TCTTTTCACCTTGACAGCAGCT
>
```

①

③

末端塩基のトリム

①トリム後のリード情報。赤枠部分の塩基がなくなっているのが分かる。また、②配列長も100 bpになっていることがわかる。

```
#入力ファイルの読み込み
fastq <- readFastq(in_f)                    #in_fで指定したファイルの読み込み
sread(fastq)                                #配列情報を表示

#本番
hoge <- width(sread(fastq)) - param_trim#トリム後のend位置情報を取得
hoge[hoge < 1] <- 1                         #トリム後のend位置が1未満の場合には1にしている
hoge1 <- DNAStringSet(sread(fastq), start=1, end=hoge)#sread(fastq)から指定した範囲を抽
hoge2 <- BStringSet(quality(quality(fastq)), start=1, end=hoge)#quality(quality(fastq
fastq <- ShortReadQ(hoge1, hoge2, id(fastq))#ShortReadQというクラスオブジェクトを作成し
sread(fastq)
quality(fastq)

#ファイルに保存
writeFastq(fastq, out_f, compre
```

①

①

R Console

```
> sread(fastq)                              #配列情報を表示
  A DNAStringSet instance of length 1000000
          width seq
      [1]   100 AGCCCGACTTTCGTCCCTGCTC...CCAACCATTCTGAGGGAACCTT
      [2]   100 GATCTGGGCTGTTCCCCTTTCG...AGTTTATCTGAATTCAGTAACC
      [3]   100 CCGGTATATTTTCGGCGCAGTG...CATCCTAGTTGTCTGTGCAACG
      [4]   100 CTTCGATACCGCCCAAGAACTT...CAAAGGTGCCATCTGTGGCCAT
      [5]   100 CCCCGGTATATTTTCGGCGCAG...AACATCCTAGTTGTCTGTGCAA
      ...   ... ...
 [999996]   100 CCCCGGTATATTTTCGGCGCAG...AACATCCTAGTTGTCTGTGCAA
 [999997]   100 TTCGGGTCTACATCTGCTTACT...GCAAGCAAACGTAACTCGCCGG
 [999998]   100 CGTCCATCCCGGTCCTCTCGTA...CTCACGACGTTCTGAACCCAGC
 [999999]   100 CTAGGGAGTATTTAGCCTTGGG...TGGACGGAGGGTTCGACGTTTC
[1000000]   100 GCCTTGTCAATCAAGGTGAGCA...GCATCCATCTTTTCACCTTGAC
>
```

②

Tips：例外処理

```
#入力ファイルの読み込み
fastq <- readFastq(in_f)                    #in_fで指定したファイル
sread(fastq)                                #配列情報を表示

#本番
hoge <- width(sread(fastq)) - param_trim#トリム後のend位置情報を取得
hoge[hoge < 1] <- 1                         #トリム後のend位置が1未満の場合には1にしている
hoge1 <- DNAStringSet(sread(fastq), start=1, end=hoge)#sread(fastq)から指定した範囲を抽
hoge2 <- BStringSet(quality(quality(fastq)), start=1, end=hoge)#quality(quality(fastq
fastq <- ShortReadQ(hoge1, hoge2, id(fastq))#ShortReadQというクラスオブジェクトを作成し
sread(fastq)                                #配列情報を表示
quality(fastq)                              #quality情報を表示

#ファイルに保存
writeFastq(fastq, out_f, compress=T)    #fastqの中身を指定したファイル名で保存
```

①

実用上は、①このあたりの例外処理が重要。このプログラムはリードごとにトリム後の配列長情報をhogeに予め格納している。そしてトリム後の配列長がもし1 bp未満になったら、1 bpは残すような処理をしている。このあたりの想定外への対処が、いわゆる「bug fix」。

R Console

```
> length(hoge)
[1] 1000000
> hoge[1:10]
 [1] 100 100 100 100 100 100 100 100 100 100
>
```

Contents

- パッケージ
  - CRANとBioconductor
  - 推奨パッケージインストール手順のおさらい
  - Bioconductor概観 → ゲノム配列パッケージ(BSgenome)
  - ヒトゲノム情報パッケージの解析
- プロモーター配列取得(sessionInfo、バージョンの違い)
  - Rパッケージ(ゲノムとアノテーションパッケージの併用)
  - FASTA形式ファイルとGFF3形式ファイル
  - データの型
- FASTQファイルの各種解析(LinuxとRを相補的に活用)
  - Linux (FastQC)とR (ShortRead)で同じ:Overrepresented sequences項目
  - Linux (FastQC)とR (QuasR)で異なる見栄え:Per base sequence content項目。FastQCに--nogroupというオプションがあることを知る。
  - FastQCのオプション (デフォルトと--nogroupあり)の違いによるKmer Content項目の結果の違い → Rで検証

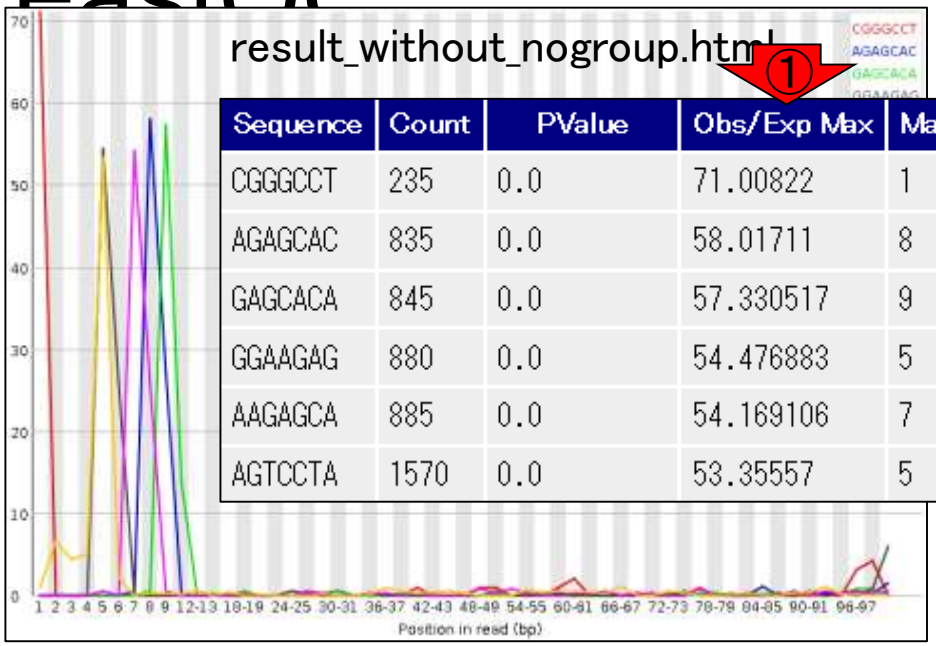
FastQC

Kmer Contentの項目で、①Obs/Exp Maxやプロファイルの図を(若干数値が代わりますが)Rで再現することができます。

result_without_nogroup.html

①

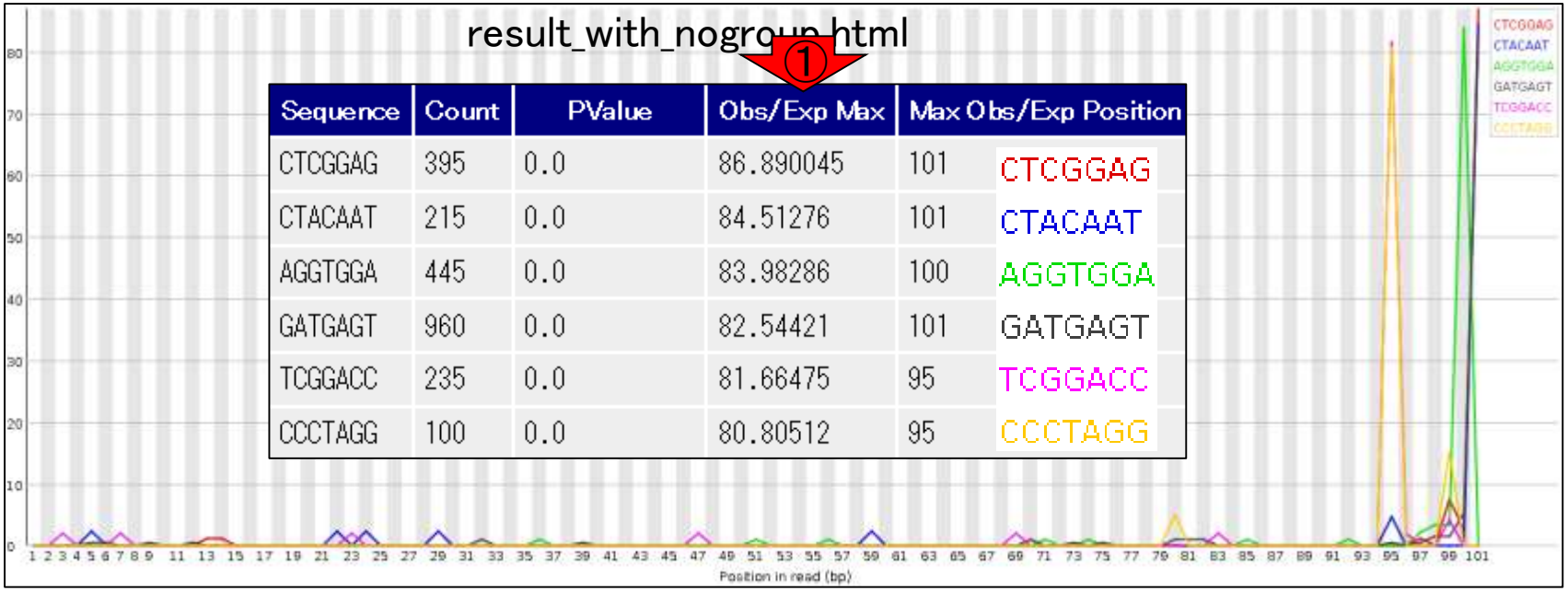
Sequence	Count	PValue	Obs/Exp Max	Max Obs/Exp Position	
CGGGCCT	235	0.0	71.00822	1	CGGGCCT
AGAGCAC	835	0.0	58.01711	8	AGAGCAC
GAGCACA	845	0.0	57.330517	9	GAGCACA
GGAAGAG	880	0.0	54.476883	5	GGAAGAG
AAGAGCA	885	0.0	54.169106	7	AAGAGCA
AGTCCTA	1570	0.0	53.35557	5	AGTCCTA



result_with_nogroup.html

①

Sequence	Count	PValue	Obs/Exp Max	Max Obs/Exp Position	
CTCGGAG	395	0.0	86.890045	101	CTCGGAG
CTACAAT	215	0.0	84.51276	101	CTACAAT
AGGTGGA	445	0.0	83.98286	100	AGGTGGA
GATGAGT	960	0.0	82.54421	101	GATGAGT
TCGGACC	235	0.0	81.66475	95	TCGGACC
CCCTAGG	100	0.0	80.80512	95	CCCTAGG



参考資料

アグリバイオ大学院講義の枠組みで、FastQC、マッピング、アセンブル、Kmer ContentのRでの検証などについて講義しています。2015.0707, 2015.06.30, 2015.06.23などでキーワード検索。