

2022.05.17版

「(Rで)塩基配列解析」というウェブページを通して、RStudioの基本的な利用法を学ぶためのものです。Chrome(推奨)で動作確認しています。ブラウザによって若干挙動が異なります。スライドの枚数が非常に多いですが、1つ1つの手順を丁寧に解説しているだけですので、中身はそれほどではないです。

(Rで)塩基配列解析の 基本的な利用法 Windows版

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ユニット
<http://www.iu.a.u-tokyo.ac.jp/>

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

前提条件

①の推奨手順通りにインストールを行っていることが前提です。できているつもりでも、実際にはできていなかったという事例が散見されます。実際の作業は②と③ですが、インストールに失敗しているパッケージについては、④を参考にして個別対応しましょう。パッケージ名や関数名のスペルミスだったというオチも散見されます。気を付けましょう。

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#

(Rで)塩基配列解析

(last modified 2022/03/31, since 2010)



このウェブページのR関連部分は、[インストール |](#) についての推奨手順 (Windows2019.10.09版とMac) 従ってフリーソフトRと必要なパッケージをインストール済みであるという前提で記述しています。初心者の方は[基本的な利用法](#) (Windowsの2020.03.06版の[PPTX](#)と[PDF](#); Macintosh2019.03.12版の[PPTX](#)と[PDF](#))で自習してください。2018年7月に[\(Rで\)塩基配列解析](#)の一部 (講習会・書籍・学会誌など) を切り分けて[サブページ](#)に移行しました。(2018/07/18)

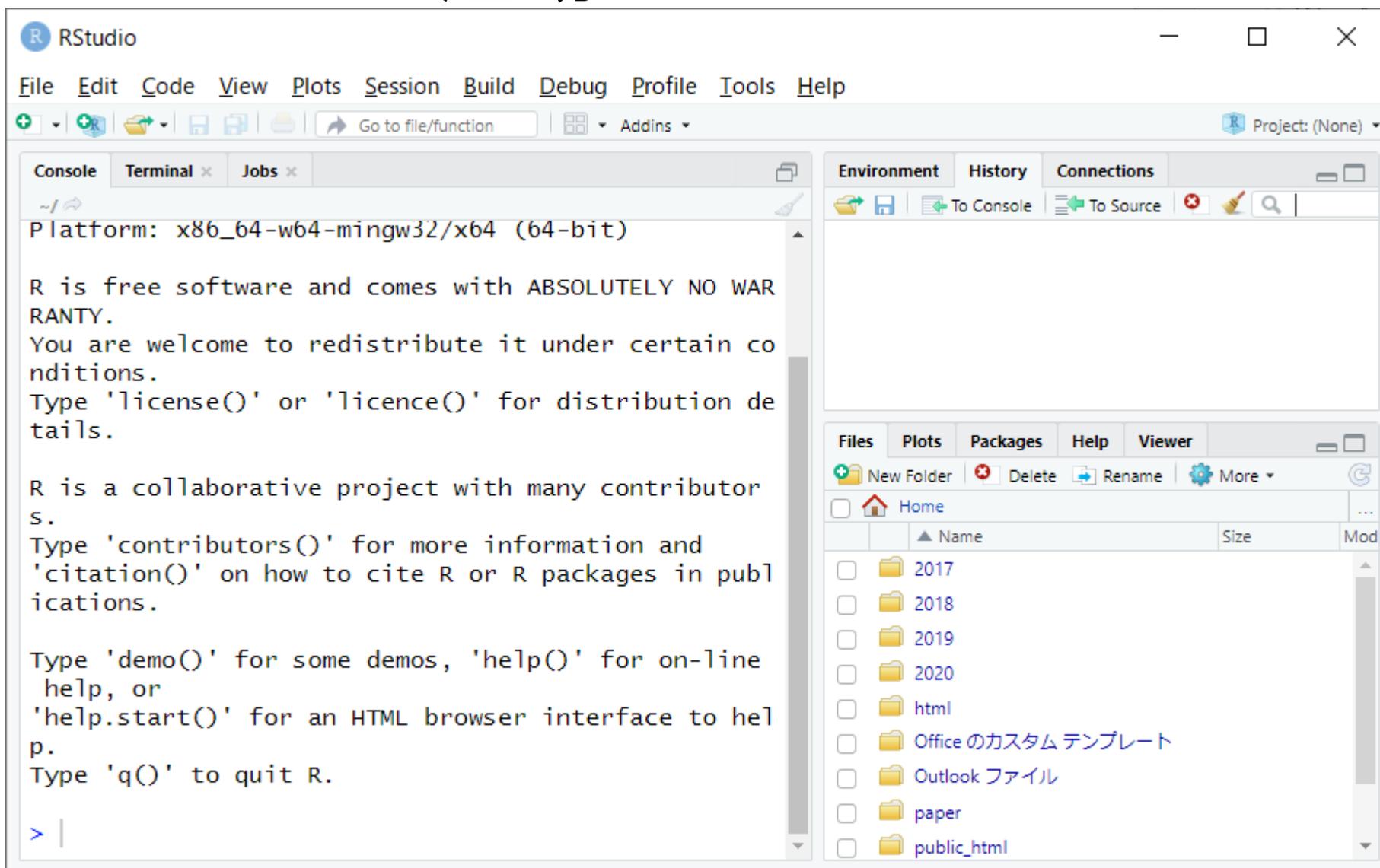
What's new? (過去のお知らせはこちら)

- 遺伝子クラスタリングに基づく発現変動遺伝子検出法の論文(Osabe et al., *BMC Bioinformatics*, 2021)のプログラムをおいている[MBCdegのGitHubサイト](#)に中のコードに存在していたミスを修正しました(RNASeq.Data関数実行部分でNormalizerオプションに与える情報がlog2(size.factors)だったのをlog(size.factors)に変更)。(2022/01/22)

- [はじめに](#) (last modified 2022/03/28) **NEW**
- [過去のお知らせ](#) (last modified 2022/03/28) **NEW**
- [インストール |](#) (last modified 2021/04/07)
- インストール | R本体とRStudio | 最新版 | [Win用](#) (last modified 2022/03/28) 推奨 **NEW**
- インストール | R本体とRStudio | 最新版 | [Mac用](#) (last modified 2022/03/28) 推奨 **NEW**
- 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 |](#) (last modified 2017/11/13)
- インストール | Rパッケージ | [必要最小限プラスアルファ](#) (last modified 2022/03/30) 推奨 **NEW**
- インストール | Rパッケージ | [個別\(2018年11月以降\)](#) (last modified 2022/03/31) **NEW**

[トップページへ](#)

RStudioの起動

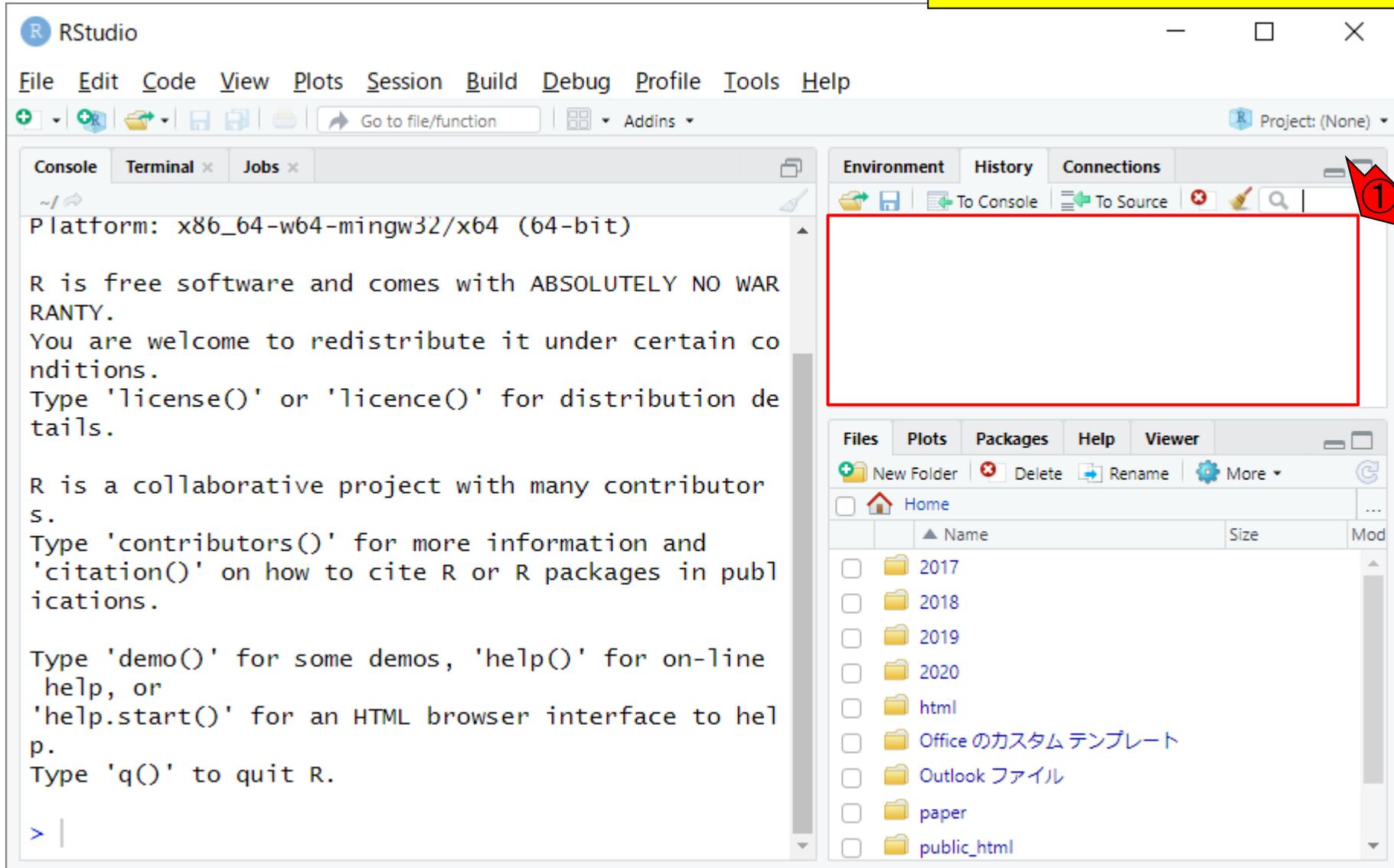


Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

見栄えの統一1

RStudio起動後の見え方はヒトによって異なる。赤枠内が①こんな感じになっているヒトもいれば、



見栄えの統一2

RStudio起動後の見え方はヒトによって異なる。赤枠内が①こんな感じになっているヒトもいれば、こんな感じでゴチャゴチャ文字が見えているヒトもいると思います。

The screenshot shows the RStudio interface with several panes. The top pane is the source editor, showing R code for installing BiocManager. The bottom-left pane is the console, showing help text for the 'install.packages()' function. The bottom-right pane is the environment pane, showing the output of the 'BiocManager::install()' function. The code in the source editor and the output in the environment pane are highlighted with red boxes.

```
5 install.packages("BiocManager") #BiocManager
6
7 #本番
8 BiocManager::install(param, update=F) #paramで
9
10 #後処理(ウェブサイト上で見えているバージョンと
11 BiocManager::valid(param) #TRUE(+
12
```

```
BiocManager::install(param, update=
#後処理(ウェブサイト上で見えているページ...
BiocManager::valid(param) #TRUEはOK...
packageVersion(param) #バージョン情報...
library(rbamtools)
2λ1.5
```

```
~/
type `contributors()` for more information and
`citation()` on how to cite R or R packages in publ
ications.

Type `demo()` for some demos, `help()` for on-line
help, or
`help.start()` for an HTML browser interface to hel
p.
Type `q()` to quit R.

> |
```

見栄えの統一3

①×を押すと、Saveするかどうか聞かれるので②Don't Save。この意味がわかっていて、Saveしたいヒトは自己責任で自由に。

The screenshot shows the RStudio interface with a dialog box titled "Untitled1 - Unsaved Changes" in the foreground. The dialog box contains a yellow warning icon and the text: "The document 'Untitled1' has unsaved changes. Do you want to save these changes?". There are three buttons: "Save", "Don't Save", and "Cancel". A red arrow labeled "1" points to the "Save" button, and another red arrow labeled "2" points to the "Don't Save" button. The background shows the RStudio editor with R code and the console window.

```
5 install.packages("BiocManager") #BiocManag
6
7 #本番
8 BiocManager::install(par
9
10 #後処理(ウェブサイト上で
11 BiocManager::valid(param
12
```

Console

```
~/
type 'contributors()' for more
'citation()' on how to cite R or R packages in public
ications.

Type 'demo()' for some demos, 'help()' for on-line
help, or
'help.start()' for an HTML browser interface to hel
p.
Type 'q()' to quit R.

> |
```

見栄えの統一4

①×を押すと、Saveするかどうかが聞かれるので②Don't Save。この意味がわかっていて、Saveしたいヒトは自己責任で自由に。こんな感じになります。左下の画面が上のほうに伸びて、左側全体を占拠したような感じになっていることが（直前のスライドとの比較でなんとなく）わかります。

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. Below the menu bar is a toolbar with icons for saving, opening, and navigating. The main workspace is divided into several panes:

- Console:** Displays the R startup message and instructions. The text includes: "Platform: x86_64-w64-mingw32/x64 (64-bit)", "R is free software and comes with ABSOLUTELY NO WARRANTY. You are welcome to redistribute it under certain conditions. Type 'license()' or 'licence()' for distribution details.", "R is a collaborative project with many contributors. Type 'contributors()' for more information and 'citation()' on how to cite R or R packages in publications.", "Type 'demo()' for some demos, 'help()' for on-line help, or 'help.start()' for an HTML browser interface to help.", "Type 'q()' to quit R."
- Environment:** Shows the loaded package "BiocManager" with version 2.1.5 and its functions: "#後処理(ウェブサイト上で見えているページ...", "BiocManager::valid(param) #TRUEはOK...", "packageVersion(param) #バージョン情報...", and "library(rbamtools)".
- Files:** A file explorer showing the current directory structure. It includes folders for years (2017, 2018, 2019, 2020), "html", "Officeのカスタムテンプレート", "Outlook ファイル", and "paper". A file ".Rhistory" is also visible with a size of 11.7 KB.

見栄えの統一5

①×次に、①を押して、赤枠内にごちゃごちゃ書かれている(記載事項はヒトによって異なる)のをすっきりさせます。

RStudio interface showing the Environment pane. The Environment pane displays the following code:

```
BiocManager::install(param, update=...
#後処理(ウェブサイト上で見えているパーズ...
BiocManager::valid(param) #TRUEはOK...
packageVersion(param) #バージョン情報...
library(rbamtools)
2^1.5
```

A red box highlights the code in the Environment pane, and a red arrow labeled '1' points to the 'update=' argument in the `BiocManager::install()` function call.

The Console pane shows the following text:

```
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

The Files pane shows the following file list:

Name	Size	Mod
.Rhistory	11.7 KB	Oct
2017		
2018		
2019		
2020		
html		
Officeのカスタムテンプレート		
Outlook ファイル		
paper		

見栄えの統一7

①×次に、①を押して、赤枠内にごちゃごちゃ書かれている(記載事項はヒトによって異なる)のをすっきりさせます。こんな感じになります。

The screenshot shows the RStudio interface. The console window on the left displays the R startup message, which has been formatted to be clean and readable. The file explorer on the right shows a directory listing with columns for Name, Size, and Mod.

Console Output:

```
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

File Explorer:

	Name	Size	Mod
<input type="checkbox"/>	2017		
<input type="checkbox"/>	2018		
<input type="checkbox"/>	2019		
<input type="checkbox"/>	2020		
<input type="checkbox"/>	html		
<input type="checkbox"/>	Officeのカスタム テンプレート		
<input type="checkbox"/>	Outlook ファイル		
<input type="checkbox"/>	paper		
<input type="checkbox"/>	public_html		

RStudioの終了

普通のソフトウェアと同様、①で終了。もし①×を押した後に、「Save workspace …」的なことを聞かれたら②Don't Save。

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins Project: (None)

Console Terminal x Jobs x

Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY. You are welcome to redistribute it under certain conditions. Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors. Type 'contributors()' for more information and 'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or 'help.start()' for an HTML browser interface to help. Type 'q()' to quit R.

> |

Environment History Connections

To Console To Source

Quit R Session

Save workspace image to ~/.RData?

Save Don't Save Cancel

Name Size Mod

- 2017
- 2018
- 2019
- 2020
- html
- Officeのカスタム テンプレート
- Outlook ファイル
- paper
- public_html

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

Console画面の見栄え1

おそらくまだ、①Console画面の②赤枠部分の文字の大きさや見栄えが異なっているヒトがいらっしゃると思います。

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins Project: (None)

Console Terminal x Jobs x

Environment History Connections

Files Plots Packages Help Viewer

```
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

	Name	Size	Mod
<input type="checkbox"/>	2017		
<input type="checkbox"/>	2018		
<input type="checkbox"/>	2019		
<input type="checkbox"/>	2020		
<input type="checkbox"/>	html		
<input type="checkbox"/>	Officeのカスタムテンプレート		
<input type="checkbox"/>	Outlook ファイル		
<input type="checkbox"/>	paper		
<input type="checkbox"/>	public_html		

Console画面の見栄え2

おそらくまだ、①Console画面の②赤枠部分の文字の大きさや見栄えが異なっているヒトがいらっしゃると思います。RStudioのGUI画面を③の方向で右下に広げるなどしてみてください。

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. Below the menu bar is a toolbar with icons for file operations and a search bar. The main window is divided into several panes. The left pane is the Console, which contains the following text:

```
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

The text in the Console is enclosed in a red rectangular box. A red arrow labeled '1' points to the top-left corner of the RStudio window. Another red arrow labeled '2' points to the top-right corner of the Console window. A third red arrow labeled '3' points to the bottom-right corner of the RStudio window. The right pane shows the Environment, History, and Connections tabs. Below that is the Files pane, which displays a file explorer view of the Home directory with folders for years (2017-2020), html, Office templates, Outlook files, paper, and public_html.

Console画面の見栄え3

おそらくまだ、①Console画面の②赤枠部分の文字の大きさや見栄えが異なっているヒトがいらっしゃると思います。RStudioのGUI画面を③の方向で右下に広げるなどしてみてください。RStudioのGUI画面を③の方向で右下に広げて、全体を縮小した結果。②の部分の見栄えが自分のGUI画面と同じになって安心しているヒトもいると思います。

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function

Console Terminal x Jobs x

Environment

```
R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

Files Plots Packages Help Viewer

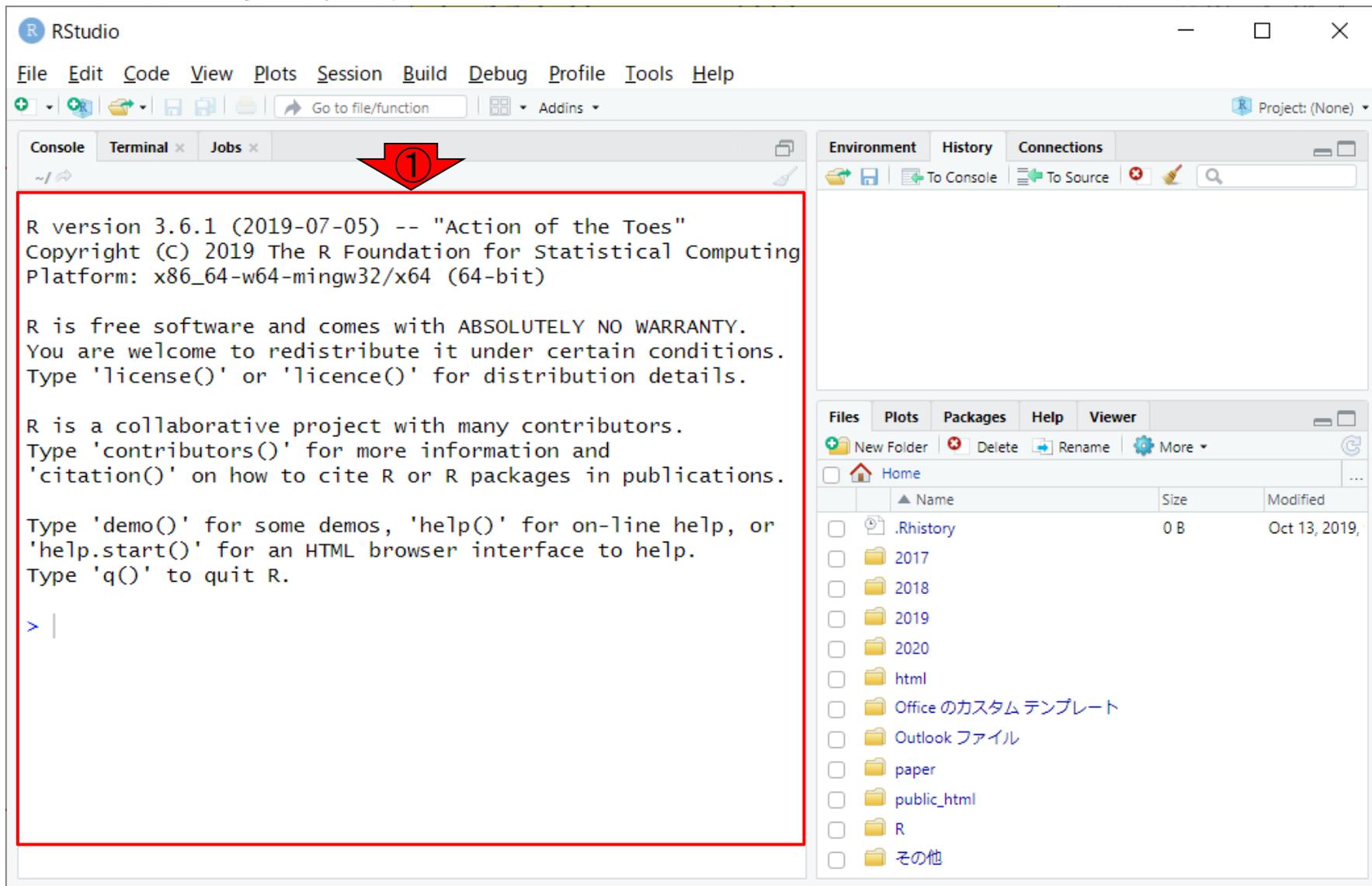
New Folder Delete Rename More

Home

	Name	Size	Modified
<input type="checkbox"/>	.Rhistory	0 B	Oct 13, 2019,
<input type="checkbox"/>	2017		
<input type="checkbox"/>	2018		
<input type="checkbox"/>	2019		
<input type="checkbox"/>	2020		
<input type="checkbox"/>	html		
<input type="checkbox"/>	Officeのカスタム テンプレート		
<input type="checkbox"/>	Outlook ファイル		
<input type="checkbox"/>	paper		
<input type="checkbox"/>	public_html		
<input type="checkbox"/>	R		
<input type="checkbox"/>	その他		

設定変更1

まずは①赤枠内の文字サイズの変更方法から。



The screenshot shows the RStudio interface. The console window is highlighted with a red border, and a red arrow with the number 1 points to it. The console displays the R startup message and a prompt. The Environment, History, and Connections panels are visible on the right. The Files panel at the bottom right shows a file explorer view of the user's home directory.

```
R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

Name	Size	Modified
.Rhistory	0 B	Oct 13, 2019,
2017		
2018		
2019		
2020		
html		
Officeのカスタム テンプレート		
Outlook ファイル		
paper		
public_html		
R		
その他		

設定変更2

まずは①赤枠内の文字サイズの変更方法から。②Toolsメニューの、③Global Options。

The screenshot shows the RStudio interface with the following elements:

- Terminal Panel (Red Box ①):** Contains R version information and help text. A red arrow points to the text area.
- Tools Menu (Red Arrow ②):** Opened from the top menu bar. The 'Global Options...' option is highlighted.
- Global Options Dialog (Red Arrow ③):** A dialog box for configuring RStudio settings, currently showing the 'Home' tab with a file explorer view.

Terminal Panel Content:

```
R version 3.6.1 (2019-07-05) -- "Action of the  
Copyright (C) 2019 The R Foundation for Statist  
Platform: x86_64-w64-mingw32/x64 (64-bit)  
  
R is free software and comes with ABSOLUTELY NO  
You are welcome to redistribute it under certain  
Type 'license()' or 'licence()' for distributio  
  
R is a collaborative project with many contribu  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.  
  
Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
> |
```

Tools Menu Items:

- Install Packages...
- Check for Package Updates...
- Version Control
- Shell...
- Terminal
- Jobs
- Addins
- Keyboard Shortcuts Help Alt+Shift+K
- Modify Keyboard Shortcuts...
- Project Options...
- Global Options...

Global Options Dialog - Home Tab:

Name	Size	Modified
.Rhistory	0 B	Oct 13, 2019,
2017		
2018		
2019		
2020		
html		
Officeのカスタム テンプレート		
Outlook ファイル		
paper		
public_html		
R		
その他		

設定変更3

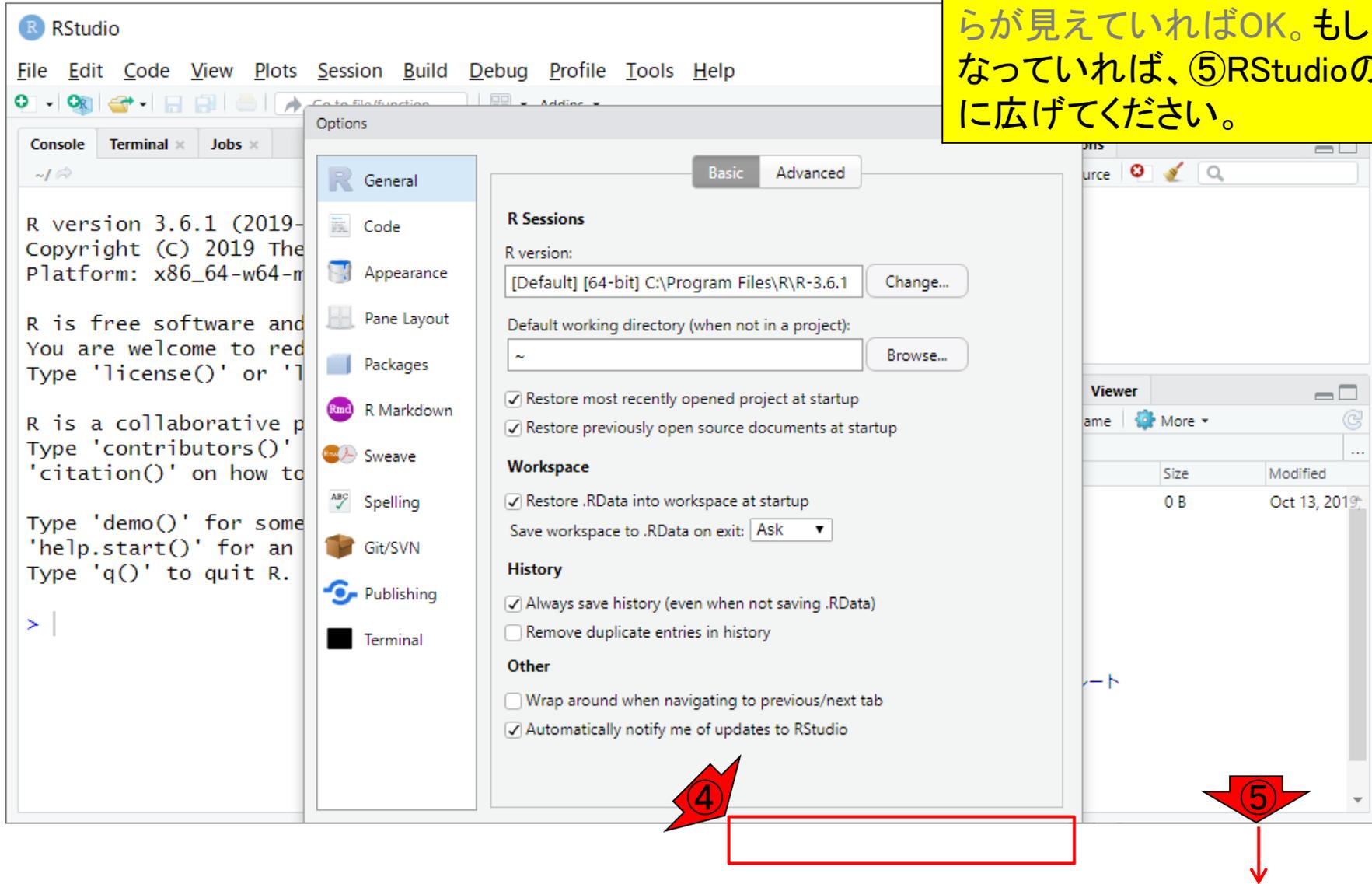
まずは①赤枠内の文字サイズの変更方法から。②Toolsメニューの、③Global Options。こんな感じになります。④これらが見えていればOK。

The screenshot shows the RStudio Options dialog box, Basic tab. The dialog is overlaid on the RStudio interface. The 'General' section is selected in the left sidebar. The 'R Sessions' section shows the R version as [Default] [64-bit] C:\Program Files\R\R-3.6.1. The 'Workspace' section has checkboxes for 'Restore most recently opened project at startup' and 'Restore previously open source documents at startup', both checked. The 'History' section has 'Always save history (even when not saving .RData)' checked. The 'Other' section has 'Automatically notify me of updates to RStudio' checked. A red arrow with the number 4 points to the 'OK' button at the bottom right of the dialog box.

RStudio Options dialog box, Basic tab. The 'General' section is selected. The 'R Sessions' section shows the R version as [Default] [64-bit] C:\Program Files\R\R-3.6.1. The 'Workspace' section has checkboxes for 'Restore most recently opened project at startup' and 'Restore previously open source documents at startup', both checked. The 'History' section has 'Always save history (even when not saving .RData)' checked. The 'Other' section has 'Automatically notify me of updates to RStudio' checked. A red arrow with the number 4 points to the 'OK' button at the bottom right of the dialog box.

設定変更4

まずは①赤枠内の文字サイズの変更方法から。②Toolsメニューの、③Global Options。こんな感じになります。④これらが見えていればOK。もしこんな感じになっていれば、⑤RStudioのGUI画面を下に広げてください。



設定変更5

まずは①赤枠内の文字サイズの変更方法から。②Toolsメニューの、③Global Options。こんな感じになります。④これらが見えていればOK。もしこんな感じになっていれば、⑤RStudioのGUI画面を下に広げてください。こんな感じです。

The screenshot shows the RStudio Options dialog box with the 'General' tab selected. The 'Basic' sub-tab is active, showing settings for R Sessions, Workspace, History, and Other. The 'Apply' button is highlighted with a red box and a red arrow labeled '4'. A second red arrow labeled '5' points to the bottom of the RStudio GUI window.

Options

Basic Advanced

R Sessions

R version: [Default] [64-bit] C:\Program Files\R\R-3.6.1 Change...

Default working directory (when not in a project): ~ Browse...

Restore most recently opened project at startup

Restore previously open source documents at startup

Workspace

Restore .RData into workspace at startup

Save workspace to .RData on exit: Ask

History

Always save history (even when not saving .RData)

Remove duplicate entries in history

Other

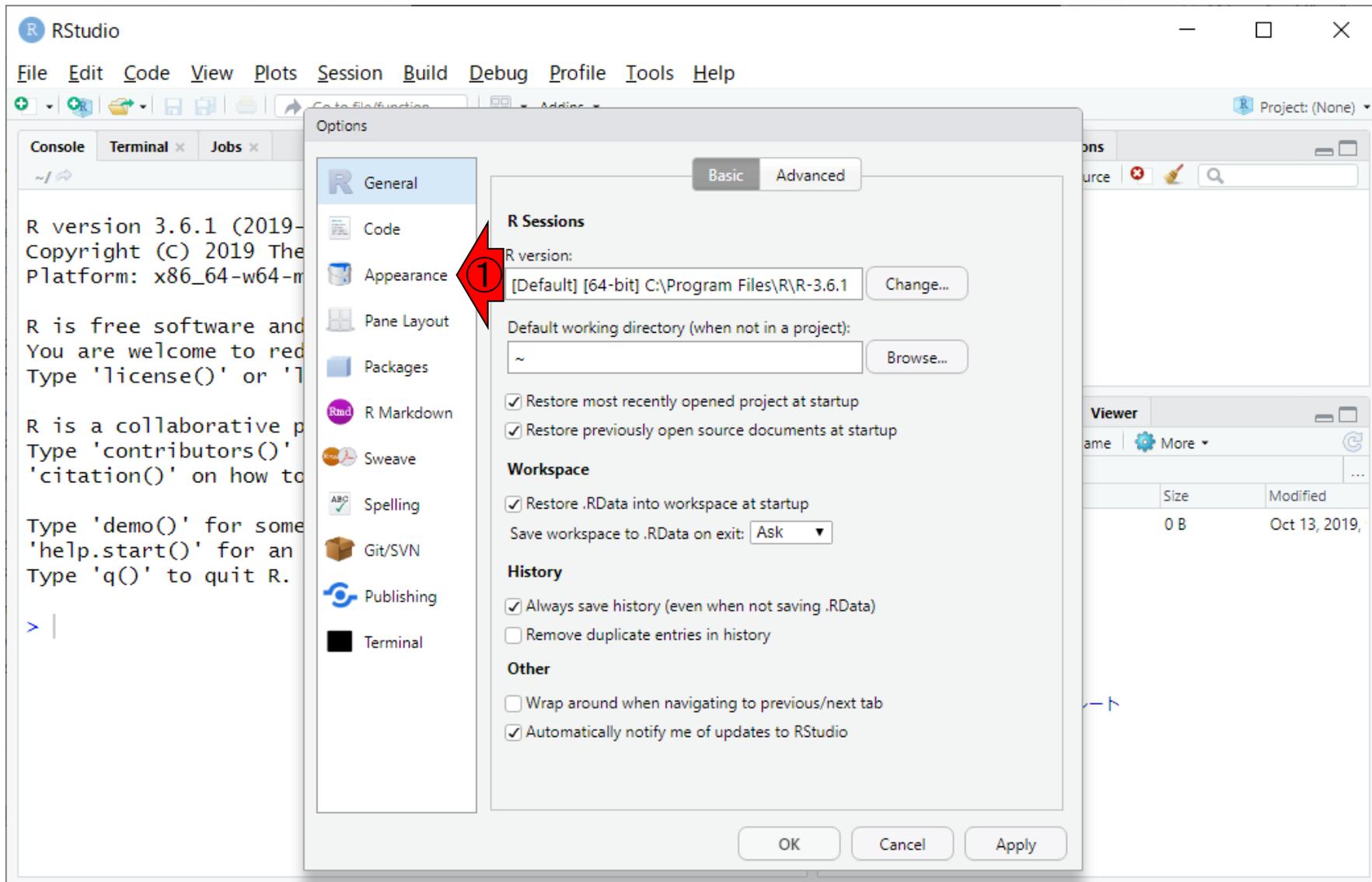
Wrap around when navigating to previous/next tab

Automatically notify me of updates to RStudio

OK Cancel Apply

設定変更6

文字サイズ変更は、①Appearanceのところで可能です。



設定変更7

文字サイズ変更は、①Appearanceのところで可能です。こんな感じになります。②Editor font sizeのところでサイズ変更を行います。確かデフォルトは10だったと思いますが、私はすでに12に変更済みなのでこんな感じになっています。

The screenshot shows the RStudio Options dialog box with the 'Appearance' tab selected. The 'Editor font size' is set to 12. The 'Editor theme' is set to 'Material'. The 'Code' editor shows a snippet of R code. The 'Viewer' pane shows a table with columns 'Size' and 'Modified'.

Options dialog box settings:

- RStudio theme: Modern
- Zoom: 100%
- Editor font: Lucida Console
- Editor font size: 12
- Editor theme: Material

```
# plotting of R objects
plot <- function(x, y, ...)
{
  if (is.function(x) &&
      is.null(attr(x, "class")))
  {
    if (missing(y))
      y <- NULL

    # check for ylab argument
    hasylab <- function(...)
      !all(is.na(
        pmatch(names(list(...)),
                "ylab")))

    if (hasylab(...))
      plot.function(x, y, ...)

    else
      plot.function(
        x, y,
        ylab = paste(
          deparse(substitute(x)),

```

Size	Modified
0 B	Oct 13, 2019,

設定変更8

文字サイズ変更は、①Appearanceのところで可能です。こんな感じになります。②Editor font sizeのところでサイズ変更を行います。確かデフォルトは10だっと思いますが、私はすでに12に変更済みなのでこんな感じになっています。③例えば14に変更して、④Applyすると…

The screenshot shows the RStudio Options dialog box with the 'Appearance' tab selected. The 'Editor font size' is set to 14. The 'Apply' button is highlighted with a red arrow labeled 4. The background shows the RStudio interface with a code editor and a terminal window.

Options dialog box settings:

- General: RStudio theme: Modern
- Code: Zoom: 100%
- Appearance: Editor font: Lucida Console
- Editor font size: 14
- Editor theme: (list of themes)

Code editor content:

```
# plotting of R
plot <- function (x, y,
{
  if (is.function(x) &&
      is.null(attr(x, "
{
  if (missing(y))
    y <- NULL

# check for ylab ar
hasylab <- function
!all(is.na(
  pmatch(names(li
    "ylab"))))

if (hasylab(...))
  plot.function(x,

else
  plot.function(
    x, y,
```

設定変更9

文字サイズ変更は、①Appearanceのところで可能です。こんな感じになります。②Editor font sizeのところでサイズ変更を行います。確かデフォルトは10だったと思いますが、私はすでに12に変更済みなのでこんな感じになっています。③例えば14に変更して、④Applyすると、こんな感じになります。⑤Console画面と呼ばれる赤枠部分の文字が大きくなったのがわかります。こんな感じで、見やすい大きさに変更してご利用ください。⑥OK。

The screenshot shows the RStudio Options dialog box with the 'Appearance' tab selected. The 'Editor font size' is set to 14. A red arrow labeled '5' points to the console area, and another red arrow labeled '6' points to the 'Apply' button.

```
R version 3.6.1 (2019-05-09)
Copyright (c) 2019 R Core Team
Platform: x86_64-w64-mingw32/x64

R is free software under the GNU General Public License (GPL).
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

>
```

```
# plotting of R plot
plot <- function(x, y) {
  if (is.function(x) || is.null(a)) {
    if (missing(y)) {
      y <- NULL
    }
  }

  # check for ylab argument
  hasylab <- function(x) {
    !all(is.na(c(
      pmatch(names(list(...)),
        "ylab"))))
  }

  if (hasylab(...)) {
    plot.function(x, y)
  } else {
    plot.function(x, y)
  }
}
```

Size	Modified
0 B	Oct 13, 2019,

設定変更10

ここまでで、文字の大きさを調整できるようになりました。本質的な部分ではありませんが、様々な年代の方が利用しますので配慮が必要な事柄です。



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins Project: (None)

Environment History Connections

Files Plots Packages Help Viewer

Home

	Name	Size	Modified
<input type="checkbox"/>	.Rhistory	0 B	Oct 13, 2019,
<input type="checkbox"/>	2017		
<input type="checkbox"/>	2018		
<input type="checkbox"/>	2019		
<input type="checkbox"/>	2020		
<input type="checkbox"/>	html		
<input type="checkbox"/>	Office のカスタム テンプレート		
<input type="checkbox"/>	Outlook ファイル		
<input type="checkbox"/>	paper		
<input type="checkbox"/>	public_html		
<input type="checkbox"/>	R		
<input type="checkbox"/>	その他		

```
R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

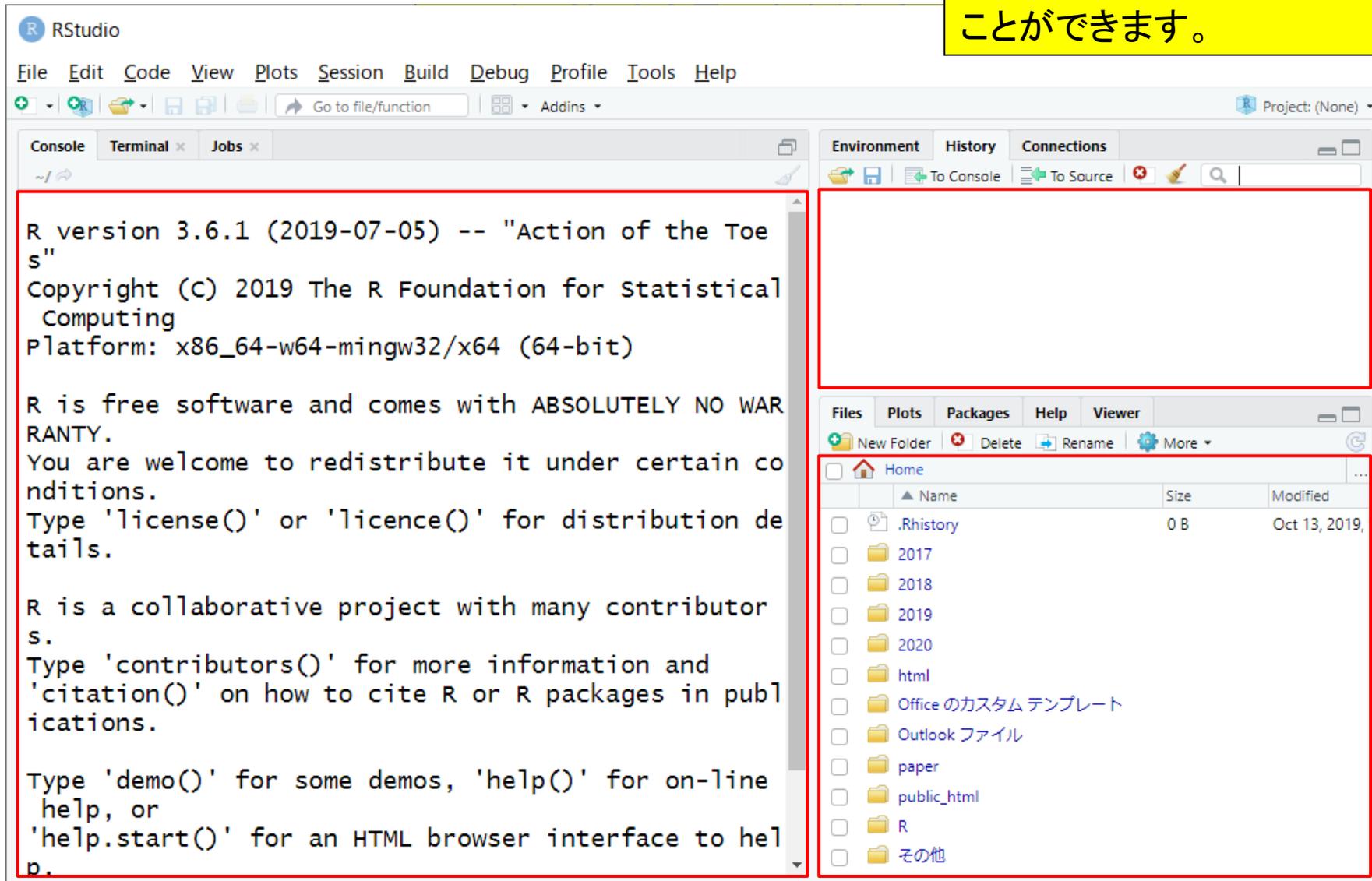
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
```

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

枠の幅1

RStudioは、ここで見えているだけでも画面が3分割されていることがわかります。それぞれの画面の高さや幅を変更することができます。



枠の幅2

例えば、①や②のところにカーソルを持っていき、右に移動させると…

The screenshot shows the RStudio interface. The console pane on the left contains the following text:

```
R version 3.6.1 (2019-07-05) -- "Action of the Toes"  
Copyright (C) 2019 The R Foundation for Statistical Computing  
Platform: x86_64-w64-mingw32/x64 (64-bit)  
  
R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.  
  
R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.  
  
Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.
```

Two red arrows with circled numbers point to the vertical scrollbar of the console pane:

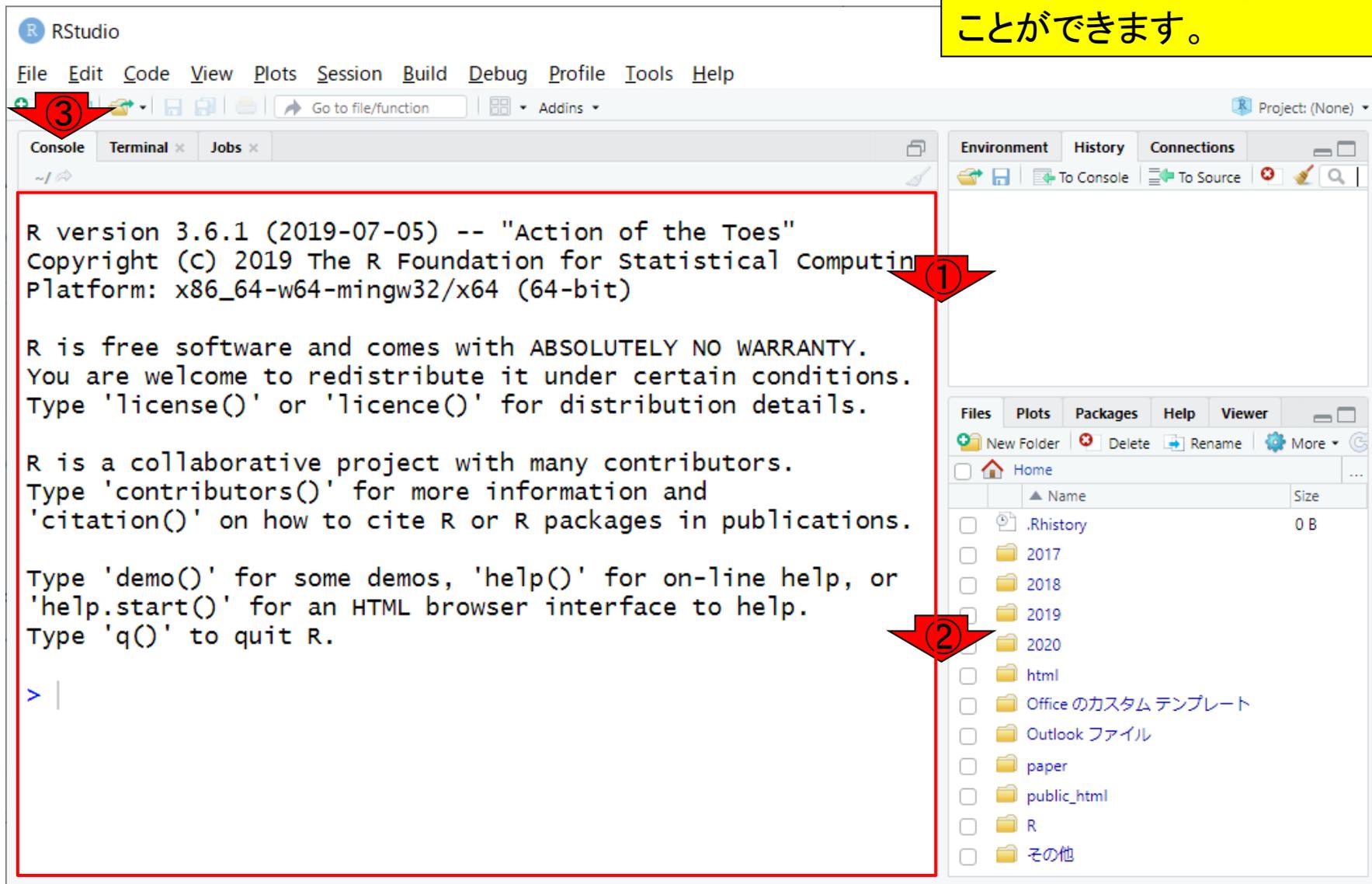
- Arrow ① points to the top part of the scrollbar.
- Arrow ② points to the bottom part of the scrollbar.

The environment pane on the right shows a file explorer view of the Home directory with the following table:

	Name	Size	Modified
<input type="checkbox"/>	.Rhistory	0 B	Oct 13, 2019,
<input type="checkbox"/>	2017		
<input type="checkbox"/>	2018		
<input type="checkbox"/>	2019		
<input type="checkbox"/>	2020		
<input type="checkbox"/>	html		
<input type="checkbox"/>	Officeのカスタム テンプレート		
<input type="checkbox"/>	Outlook ファイル		
<input type="checkbox"/>	paper		
<input type="checkbox"/>	public_html		
<input type="checkbox"/>	R		
<input type="checkbox"/>	その他		

枠の幅3

例えば、①や②のところにカーソルを持っていき、右に移動させると…③
Console(こんそーる)画面の幅を広くすることができます。



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins Project: (None)

Console Terminal x Jobs x

```
R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

Environment History Connections

Files Plots Packages Help Viewer

Name	Size
.Rhistory	0 B
2017	
2018	
2019	
2020	
html	
Officeのカスタム テンプレート	
Outlook ファイル	
paper	
public_html	
R	
その他	

枠の幅4

①の部分も同様にして上下させることで
…。

The screenshot shows the RStudio interface. The console window on the left displays the R startup message and instructions. The file explorer on the right shows the 'Home' directory with a list of folders and files. A red arrow with the number '1' points to the 'Packages' tab in the file explorer, indicating the area to be adjusted.

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins Project: (None)

Console Terminal x Jobs x

```
R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

Environment History Connections

To Console To Source

Files Plots Packages Viewer

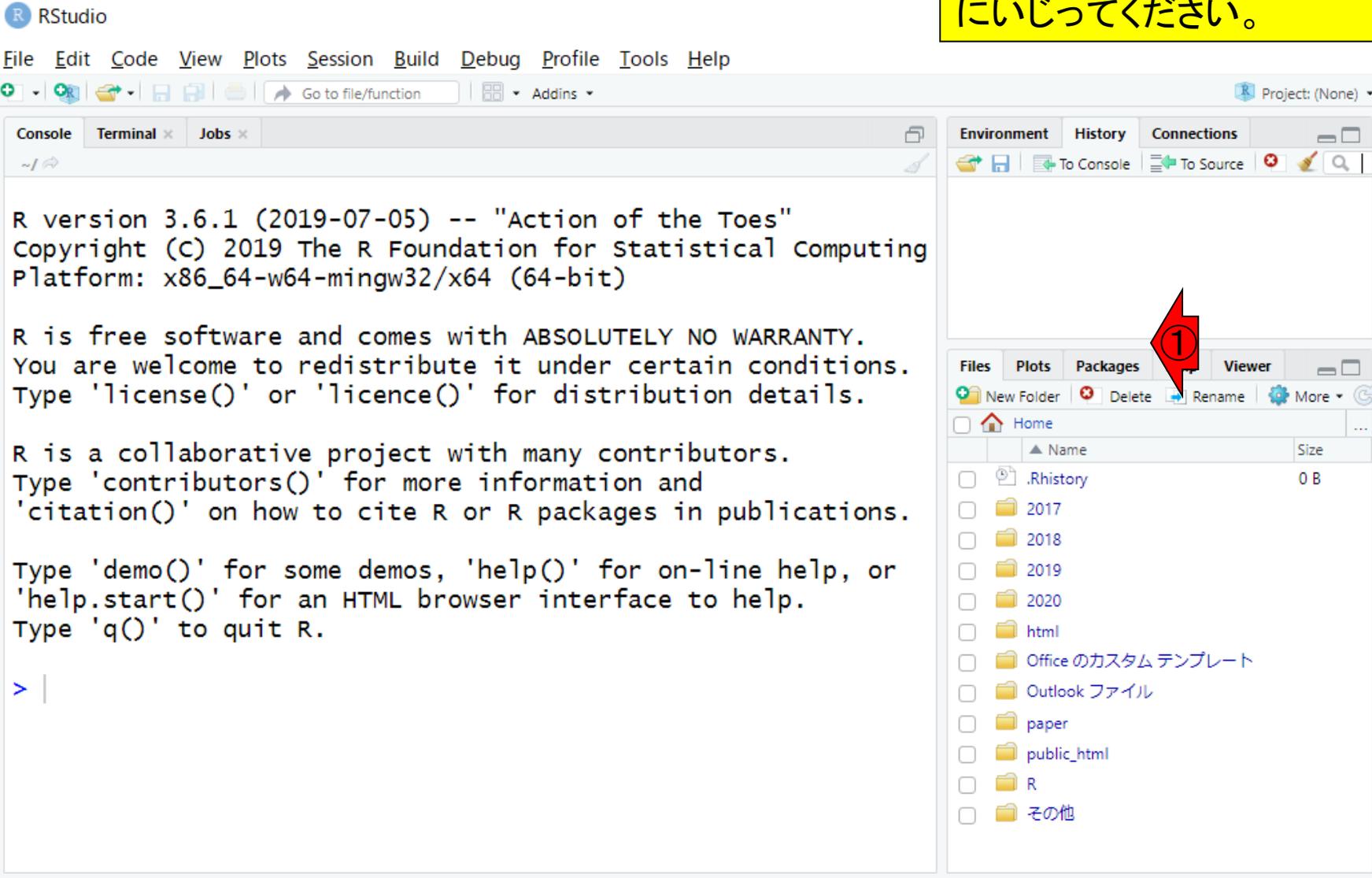
New Folder Delete Rename More

Home

	Name	Size
<input type="checkbox"/>	.Rhistory	0 B
<input type="checkbox"/>	2017	
<input type="checkbox"/>	2018	
<input type="checkbox"/>	2019	
<input type="checkbox"/>	2020	
<input type="checkbox"/>	html	
<input type="checkbox"/>	Officeのカスタム テンプレート	
<input type="checkbox"/>	Outlook ファイル	
<input type="checkbox"/>	paper	
<input type="checkbox"/>	public_html	
<input type="checkbox"/>	R	
<input type="checkbox"/>	その他	

枠の幅5

①の部分も同様にして上下させることで…右側の画面の上下幅を変更することができます。状況に応じて見やすいようにいじってください。



The screenshot shows the RStudio interface. The console window on the left displays the R startup message and instructions. The file explorer on the right shows the 'Home' directory with various folders. A red arrow with the number 1 points to the 'Plots' tab in the file explorer, indicating that the width of the right-hand pane can be adjusted by dragging the tab.

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins Project: (None)

Console Terminal x Jobs x

```
R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

Environment History Connections

Files Plots Packages Viewer

Home

	Name	Size
<input type="checkbox"/>	.Rhistory	0 B
<input type="checkbox"/>	2017	
<input type="checkbox"/>	2018	
<input type="checkbox"/>	2019	
<input type="checkbox"/>	2020	
<input type="checkbox"/>	html	
<input type="checkbox"/>	Officeのカスタム テンプレート	
<input type="checkbox"/>	Outlook ファイル	
<input type="checkbox"/>	paper	
<input type="checkbox"/>	public_html	
<input type="checkbox"/>	R	
<input type="checkbox"/>	その他	

電卓的な利用1

①Console画面上で、②「1+1」と打ち込んでからリターンキーを押したところ。2という結果が表示されていることがわかります。

The screenshot shows the RStudio interface. The console window displays the following text:

```
R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> 1+1
[1] 2
> |
```

A red arrow labeled '1' points to the console window, and another red arrow labeled '2' points to the command '1+1' and its output '[1] 2'.

電卓的な利用2

①Console画面上で、②「1+1」と打ち込んでからリターンキーを押したところ。2という結果が表示されていることがわかります。今度は、③「10*5」で、50という結果が表示されていることがわかります。

RStudio interface showing the console output. A red box highlights the console content. A red arrow labeled ① points to the console area. A red arrow labeled ② points to the input '1+1' and its output '[1] 2'. A red arrow labeled ③ points to the input '10*5' and its output '[1] 50'.

```
R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> 1+1
[1] 2
> 10*5
[1] 50
>
```

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

プロンプト (>)

ここまでの作業で、コマンドを実行すると、その結果が表示されるとともに、①「>」が表示されます。これを「プロンプト」と言います。これが見えていれば、「直前のコマンド実行が完了している」と解釈してよいです。

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Console Terminal x Jobs x
~/

R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

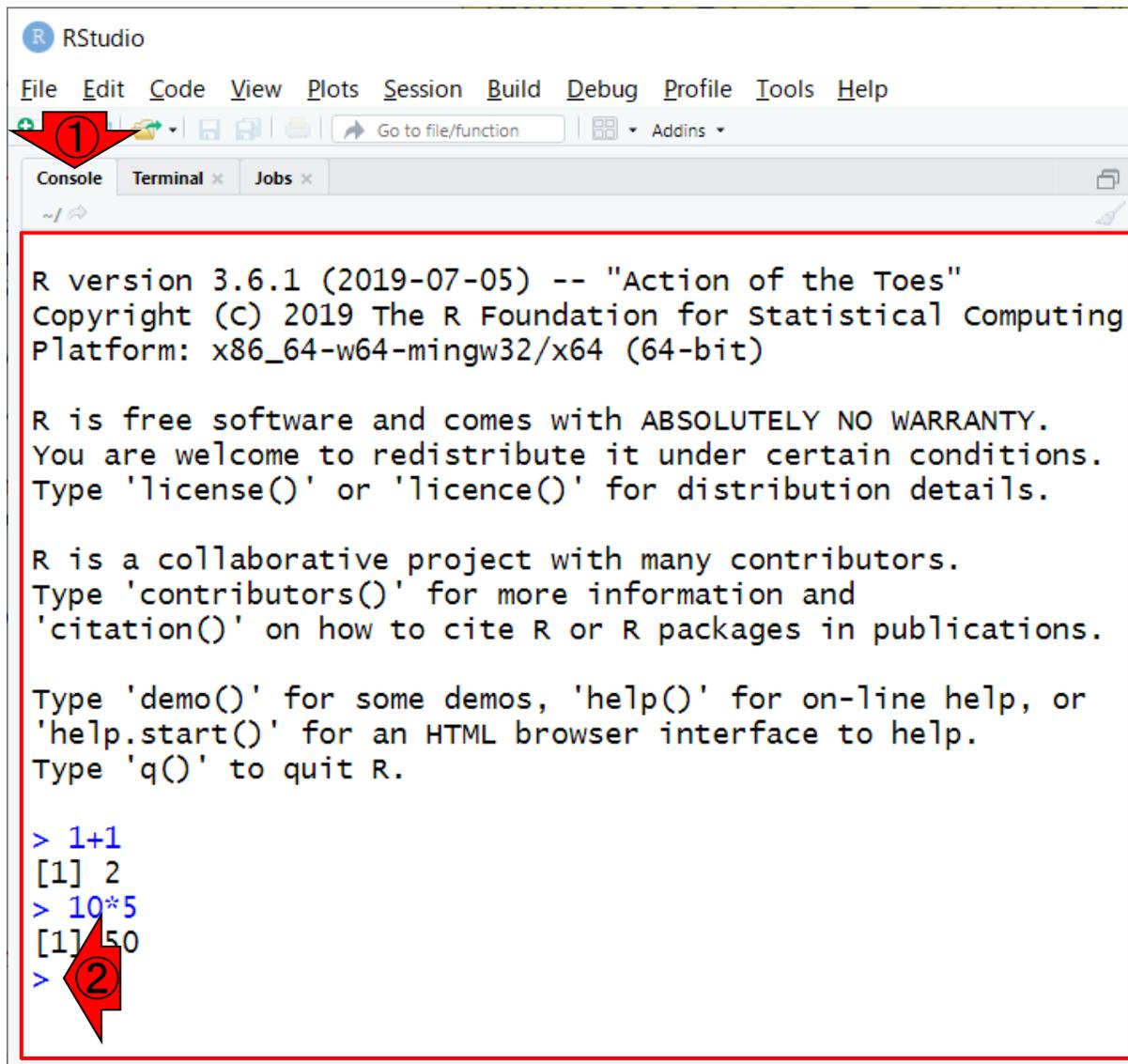
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
```

```
> 1+1
[1] 2
> 10*5
[1] 50
> ①
```

The screenshot shows the RStudio interface. The Environment pane on the right displays the results of the commands: `1+1` resulting in `[1] 2` and `10*5` resulting in `[1] 50`. Below the Environment pane is a File Explorer window showing a directory listing with folders like `.Rhistory`, `2017`, `2018`, `2019`, `2020`, `html`, `Officeのカスタム テンプレート`, `Outlook ファイル`, `paper`, `public_html`, `R`, and `その他`.

Console画面クリア1

このあと、1行1行コマンドを打ち込んでいくと、常に①Console画面上の下部である、②のあたりに入力をするようになります。必要に応じて①Console画面をクリア (Clear Console)して、Console画面上部でコマンド入力できるようにするやり方を示します。



```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Console Terminal x Jobs x
~ / ↗

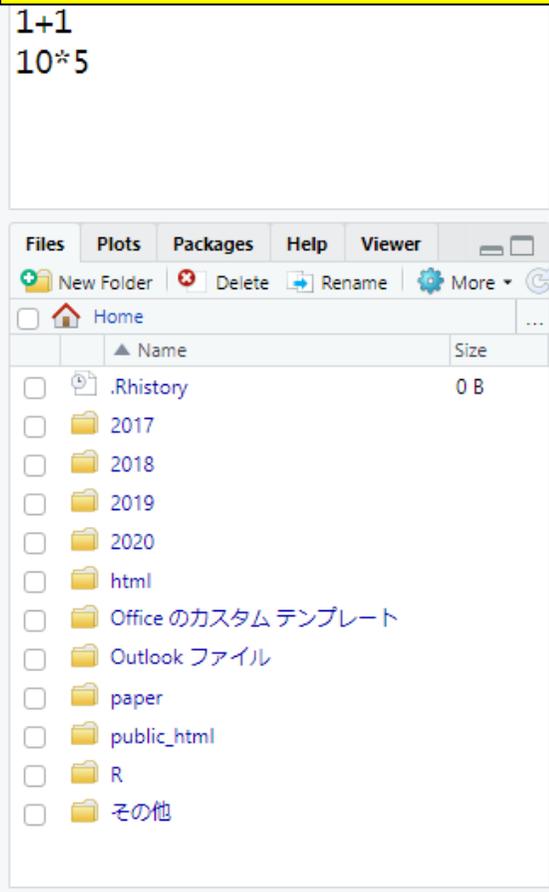
R version 3.6.1 (2019-07-05) -- "Action of the Toes"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> 1+1
[1] 2
> 10*5
[1] 50
>
```



1+1
10*5

Files Plots Packages Help Viewer
New Folder Delete Rename More
Home
Name Size
[] .Rhistory 0 B
[] 2017
[] 2018
[] 2019
[] 2020
[] html
[] Officeのカスタム テンプレート
[] Outlook ファイル
[] paper
[] public_html
[] R
[] その他

Console画面クリア2

基本的には①を押すだけです。②Editメニューから、③Clear Console。④キーボード上で「Ctrl」キーを押しながら「L」キーを押すのでもかまいません。

The screenshot shows the RStudio interface with the console window open. The console contains the following text:

```
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> 1+1
[1] 2
> 10*5
[1] 50
>
```

Annotations on the screenshot:

- ②: Points to the Edit menu.
- ③: Points to the 'Clear Console' option in the Edit menu.
- ④: Points to the 'Clear Console' option in the Edit menu.
- ①: Points to the 'Clear Console' button in the Environment pane.

The Environment pane shows the following variables:

Environment	History	Connections
1+1		
10*5		

The Files pane shows the following folders:

Name	Size
.Rhistory	0 B
2017	
2018	
2019	
2020	
html	
Officeのカスタム テンプレート	
Outlook ファイル	
paper	
public_html	
R	
その他	

Console画面クリア3

基本的には①を押すだけです。②Editメニューから、③Clear Console。④キーボード上で「Ctrl」キーを押しながら「L」キーを押すのでもかまいません。無事、⑤Console画面がクリアされました。

The screenshot shows the RStudio interface. The console window on the left is empty, with a blue prompt character '>' at the top. A red arrow with the number '5' points to the console window. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The right-hand pane is divided into three sections: Environment, History, and Connections; Files, Plots, Packages, Help, and Viewer. The Environment section shows the results of the last command: '1+1' and '10*5'. The Files section shows a file explorer view of the user's home directory, listing folders such as .Rhistory, 2017, 2018, 2019, 2020, html, Office のカスタム テンプレート, Outlook ファイル, paper, public_html, R, and その他.

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

計算の中断1

さきほどの①「Ctrl + L」は、「Ctrlキーを押しながらLキーを押す」という意味で使いました。

The screenshot shows the RStudio interface. The 'Edit' menu is open, displaying various editing actions and their keyboard shortcuts. A red arrow with the number '1' points to the 'Clear Console' option, which has the shortcut 'Ctrl+L'. The console window on the right shows the results of calculations: '1+1' and '10*5'. The file explorer on the bottom right shows a directory structure with folders like '2017', '2018', '2019', '2020', 'html', 'Office のカスタム テンプレート', 'Outlook ファイル', 'paper', 'public_html', 'R', and 'その他'.

Action	Shortcut
Back	Ctrl+F9
Forward	Ctrl+F10
Undo	Ctrl+Z
Redo	Ctrl+Y
Cut	Ctrl+X
Copy	Ctrl+C
Paste	Ctrl+V
Folding	
Find in Files...	Ctrl+Shift+F
Clear Console	Ctrl+L

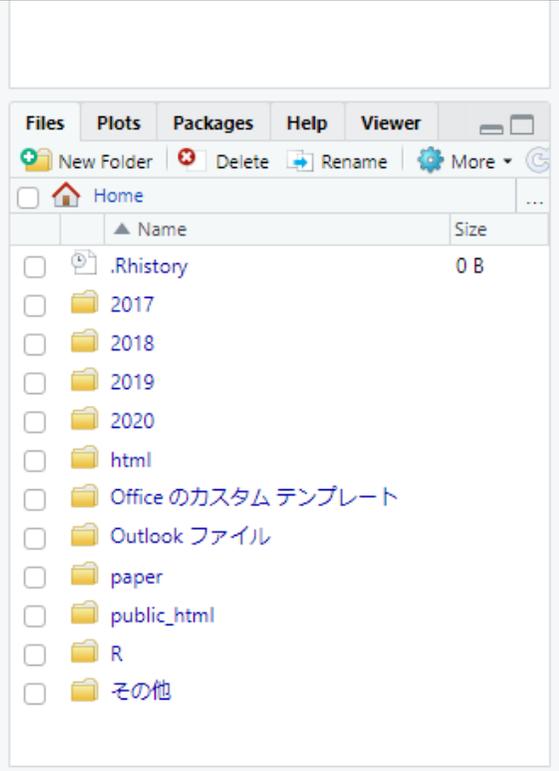
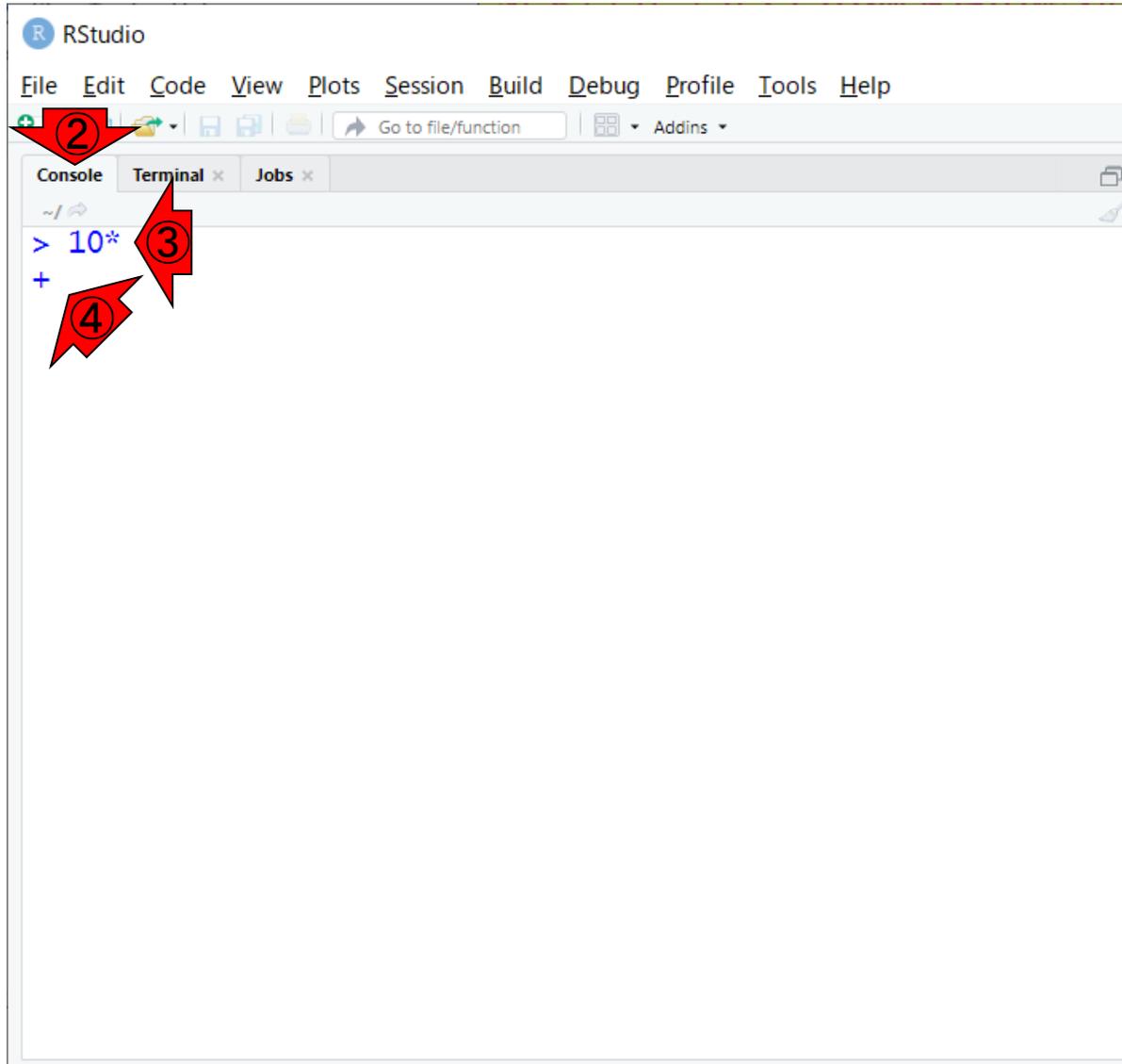
計算の中断2

さきほどの①「Ctrl + L」は、「Ctrlキーを押しながらLキーを押す」という意味で使いました。次に②Console画面上で、コマンド入力が不完全な状態のときに表示される「+」とその解決策を示します。

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. Below the menu bar is a toolbar with icons for saving, opening, and navigating. The main console area on the left shows a prompt `>` with a red arrow pointing to it. The right-hand pane is divided into several sections: Environment, History, Connections, Files, Plots, Packages, Help, and Viewer. The History section shows the commands `1+1` and `10*5`. The Files section shows a file explorer view of the home directory with a list of folders including `.Rhistory`, `2017`, `2018`, `2019`, `2020`, `html`, `Officeのカスタム テンプレート`, `Outlook ファイル`, `paper`, `public_html`, `R`, and `その他`.

計算の中断3

さきほどの①「Ctrl + L」は、「Ctrlキーを押しながらLキーを押す」という意味で使いました。次に②Console画面上で、コマンド入力が不完全な状態のときに表示される「+」とその解決策を示します。例として、「10*5」と入力するつもりだったが、意図的に間違えて③「10*」まででリターンキーを押してみましょう。④のようになればOKです。



計算の中断4

非推奨ですが、どうにかしてこのままの状態
で10*5を計算したい場合は、⑤の
状態から「5」と打ち込んでリターンキーを
押します。

The screenshot shows the RStudio interface. The console window on the left contains the text `> 10*` followed by a cursor `+`. A red arrow with the number 5 inside points to the cursor. The Environment pane on the right shows the following objects:

Object	Value
1+1	2
10*5	50
10*	10

The Files pane on the right shows a file explorer view of the user's home directory, listing folders such as `.Rhistory`, `2017`, `2018`, `2019`, `2020`, `html`, `Officeのカスタム テンプレート`, `Outlook ファイル`, `paper`, `public_html`, `R`, and `その他`.

計算の中断5

非推奨ですが、どうにかしてこのままの状態
で10*5を計算したい場合は、⑤の状態から「5」と打ち込んでリターンキーを押します。
⑥無事「10*5」の計算結果である50が表示され、⑦プロンプトが表示されていることがわかります。

RStudio interface showing the console output:

```
> 10*  
+ 5  
[1] 50  
>
```

The console output shows the result of the calculation: `[1] 50`. Red arrows labeled 6 and 7 point to the prompt and the output respectively.

The Environment pane shows the following variables:

- 1+1
- 10*5
- 10*
- 5

The Files pane shows the following files and folders:

- .Rhistory
- 2017
- 2018
- 2019
- 2020
- html
- Office のカスタム テンプレート
- Outlook ファイル
- paper
- public_html
- R
- その他

計算の中断6

①もう1回同じ状況にします。「プロンプト (>)」ではなく、②「+」になってしまったときの推奨手順は、③「Esc」キー。または「Ctrl + C」または「Ctrl + Z」。

The screenshot shows the RStudio interface. The console window displays the following text:

```
> 10*  
+ 5  
[1] 50  
> 10*  
+
```

Red arrows with numbers 1, 2, and 3 point to the prompt, the plus sign, and the Esc key respectively. The keyboard close-up shows the Esc key highlighted with a red arrow and the number 3. The RStudio interface also shows the Environment, History, and Connections panels, and a Viewer panel showing a file size of 0 B.

計算の中断7

①もう1回同じ状況にします。「プロンプト(>)」ではなく、②「+」になってしまったときの推奨手順は、③「Esc」キー。または「Ctrl + C」または「Ctrl + Z」。④「Esc」キーを押した結果。確かにプロンプト(>)が出ていますね。

The screenshot shows the RStudio interface. The console window on the left contains the following text:

```
> 10*  
+ 5  
[1] 50  
> 10*  
+  
> |
```

A red arrow with the number 4 inside points to the prompt character (>) on the line following the plus sign (+).

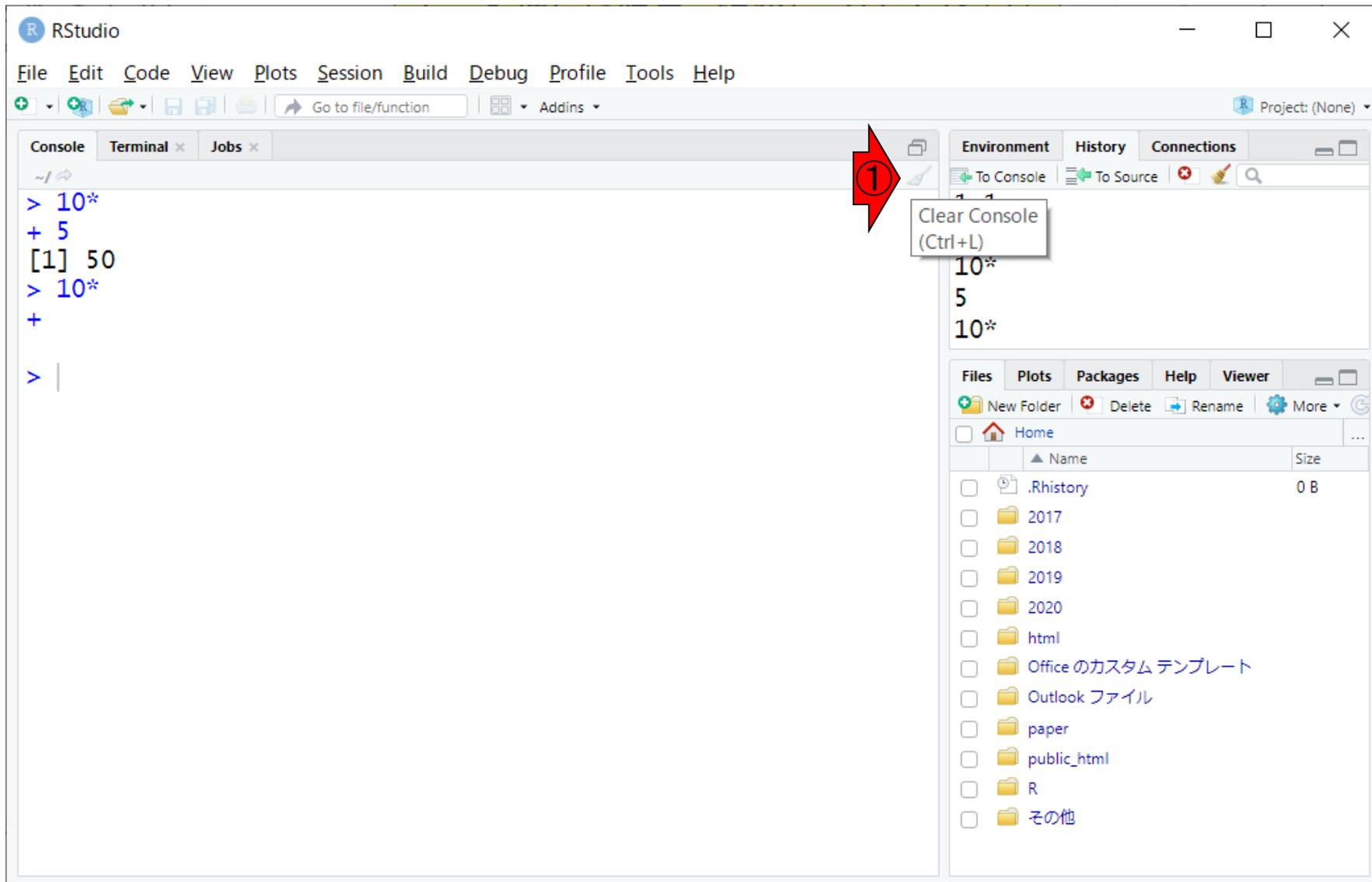
The Environment pane on the right shows the following objects:

- 1+1
- 10*5
- 10*
- 5
- 10*

The Files pane on the right shows a file explorer view with the following folders:

- .Rhistory (0 B)
- 2017
- 2018
- 2019
- 2020
- html
- Office のカスタム テンプレート
- Outlook ファイル
- paper
- public_html
- R
- その他

計算の中断8



The screenshot shows the RStudio interface. The console pane on the left contains the following text:

```
> 10*  
+ 5  
[1] 50  
> 10*  
+  
> |
```

The Environment pane on the right shows a tooltip for the 'Clear Console (Ctrl+L)' button. The tooltip text is:

Clear Console
(Ctrl+L)

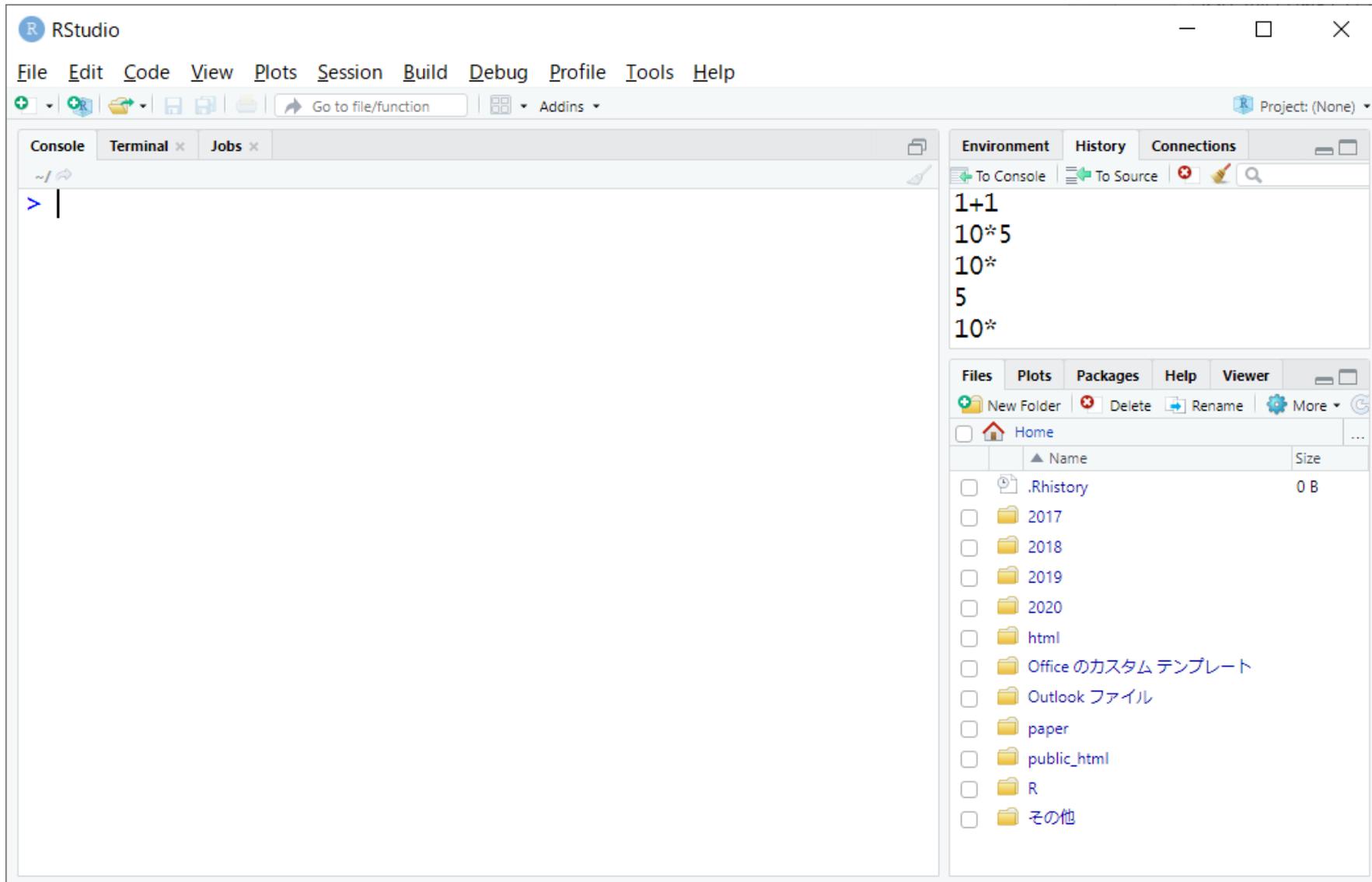
The Environment pane also displays the following text:

```
10*  
5  
10*
```

The Files pane on the right shows a file explorer view with the following items:

Name	Size
.Rhistory	0 B
2017	
2018	
2019	
2020	
html	
Office のカスタム テンプレート	
Outlook ファイル	
paper	
public_html	
R	
その他	

計算の中断9



The screenshot shows the RStudio interface. The console window on the left contains a prompt `>` followed by a vertical bar. The Environment window on the right displays the results of calculations: `1+1`, `10*5`, `10*`, `5`, and `10*`. The Files window at the bottom right shows a file explorer view of the home directory, listing folders such as `.Rhistory`, `2017`, `2018`, `2019`, `2020`, `html`, `Officeのカスタム テンプレート`, `Outlook ファイル`, `paper`, `public_html`, `R`, and `その他`.

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

項目と例題1

① (Rで)塩基配列解析のウェブページは、サンプルデータと解析例を徹底的に充実させています。項目数が非常に多いですが、慣れです。項目だけのものや古いものも多数あります。

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#

(Rで)塩基配列解析

(last modified 2022/03/31, since 2010)

このウェブページのR関連部分は、[インストール | についての推奨手順](#) (Windows2019.10.09版とMacintosh2021.04.01版)に従ってフリーソフトRと必要なパッケージをインストール済みであるという前提で記述しています。初心者の方は[基本的な利用法](#) (Windowsの2020.03.06版の[PPTX](#)と[PDF](#); Macintosh2019.03.12版の[PPTX](#)と[PDF](#))で自習してください。2018年7月に(Rで)塩基配列解析の一部(講習会・書籍・学会誌など)を切り分けて[サブページ](#)に移行しました。(2018/07/18)

What's new? (過去のお知らせはこちら)

- 遺伝子クラスタリングに基づく発現変動遺伝子検出法の論文(Osabe et al., BMC Bioinformatics, 2021)のプログラムをおいている[MBCdegのGitHubサイト](#)に中のコードに存在していたミスを修正しました(RNASeq.Data関数実行部分でNormalizerオプションに与える情報がlog2(size.factors)だったのをlog(size.factors)に変更)。(2022/01/22)

- [はじめに](#) (last modified 2022/03/28) **NEW**
- [過去のお知らせ](#) (last modified 2022/03/28) **NEW**
- [インストール | について](#) (last modified 2021/04/02)
- インストール | R本体とRStudio | 最新版 | [Win用](#) (last modified 2022/03/28) 推奨 **NEW**
- インストール | R本体とRStudio | 最新版 | [Mac用](#) (last modified 2022/03/28) 推奨 **NEW**
- 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 | について](#) (last modified 2018/11/13)
- インストール | Rパッケージ | [必要最小限プラスアルファ](#) (last modified 2022/03/30) 推奨 **NEW**
- インストール | Rパッケージ | [個別\(2018年11月以降\)](#) (last modified 2022/03/31) **NEW**

[トップページへ](#)

項目と例題2

①(Rで)塩基配列解析のウェブページは、サンプルデータと解析例を徹底的に充実させています。項目数が非常に多いですが、慣れです。項目だけのものや古いものも多数あります。②をクリック。

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#

- 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 | について](#) (last modified 2018/11/13)
- インストール | Rパッケージ | [必要最小限プラスアルファ](#) (last modified 2022/03/30) 推奨 NEW
- インストール | Rパッケージ | [個別\(2018年11月以降\)](#) (last modified 2022/03/31) NEW
- インストール | Rパッケージ | [個別\(2018年11月以前\)](#) (last modified 2018/11/12)
- [基本的な利用法](#) (last modified 2020/03/16)
- [サンプルデータ](#) (last modified 2019/09/06)
- 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 2019/04/24)
- イントロ | 一般 | [ランダムな塩基配列を生成](#) (last modified 2014/06/16)
- イントロ | 一般 | [任意の長さの可能な全ての塩基配列を作成](#) (last modified 2015/02/19)
- 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 | について](#) (last modified 2019/09/06)
- イントロ | 一般 | [指定した範囲の配列を取得 | Biostring](#) (last modified 2019/09/06)
- 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 | について](#) (last modified 2019/09/06)
- イントロ | 一般 | [翻訳配列\(translate\)を取得 | Biostrings](#) (last modified 2015/09/12)
- イントロ | 一般 | [翻訳配列\(translate\)を取得 | seqinr\(Charif_2005\)](#) (last modified 2015/03/09)
- イントロ | 一般 | [相補鎖\(complement\)を取得](#) (last modified 2019/03/10)
- イントロ | 一般 | [逆相補鎖\(reverse complement\)を取得](#) (last modified 2019/03/10)
- イントロ | 一般 | [逆鎖\(reverse\)を取得](#) (last modified 2019/03/10)
- イントロ | 一般 | k-mer解析 | k=1(塩基ごとの出現頻度解析) | [Biostrings](#) (last modified 2016/04/27)
- イントロ | 一般 | k-mer解析 | k=2(2連続塩基の出現頻度解析) | [Biostrings](#) (last modified 2016/01/28)
- イントロ | 一般 | k-mer解析 | k=3(3連続塩基の出現頻度解析) | [Biostrings](#) (last modified 2016/01/28)
- イントロ | 一般 | k-mer解析 | k=n(n連続塩基の出現頻度解析) | [Biostrings](#) (last modified 2016/05/01)

[トップページへ](#)

項目と例題3

こんな感じになります。①この項目は、塩基配列のファイルを入力として読み込んで、指定した範囲の配列を切り出して、別のファイルに出力するスクリプトです。

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostrings

ゲスト

更新

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

①

[Biostrings](#)パッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルから様々な部分配列を取得するやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。例えば入力ファイルがヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。したがって、chr4とchr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロード時に、*.fastaという拡張子が*.txtに勝手に変更されることがありますのでご注意ください。ここでは、

「ファイル」 - 「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し以下をコピー。

1. (single-)FASTA形式ファイル(sample1.fasta)の場合:

任意の範囲 (始点が3, 終点が9)の配列を抽出するやり方です。

```
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
param <- c(3, 9)            #抽出したい範囲の始点と終点を指定

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

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

#本番
fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
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
```

2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル(h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta"        #入力ファイル名を指定してin_fに格納
out_f <- "hoge2.fasta"       #出力ファイル名を指定してout_fに格納
```

[トップページへ](#)

項目と例題4

こんな感じになります。①この項目は、塩基配列のファイルを入力として読み込んで、指定した範囲の配列を切り出して、別のファイルに出力するスクリプトです。大抵の項目は、いくつかの例題を提供しています。例えば、②が例題1で、③が例題2だと解釈します。④クリックする項目を間違えて困った人は、「トップページへ」で一番上からやり直してください。

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostrings

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

Biostringsパッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルからやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。ヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。chr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロードや拡張子が*.txtに勝手に変更されることがありますのでご注意ください。ここでは、「ファイル」 - 「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し以下をコピー。

1. (single-)FASTA形式ファイル(sample1.fasta)の場合:

任意の範囲 (始点が3, 終点が9)の配列を抽出するやり方です。

```
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
param <- c(3, 9)           #抽出したい範囲の始点と終点を指定

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

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

#本番
fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
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
```

2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル(h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge2.fasta"    #出力ファイル名を指定してout_fに格納
```

[トップページへ](#)

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

入力ファイルの保存1

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostrings

ゲスト

更新

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

[Biostrings](#)パッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルから様々な部分配列を取得するやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。例えば入力ファイルがヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。したがって、chr4とchr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロード時に、*.fastaとい

て拡張子が*.txtに勝手に変更されることがありますのでご注意ください。ここでは、

①「ファイル」 - 「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し以下をコピー。

1. (single-)FASTA形式ファイル(sample1.fasta)の場合:

任意の範囲 (始点が3, 終点が9)の配列を抽出するやり方です。

```
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
param <- c(3, 9)           #抽出したい範囲の始点と終点を指定

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

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

#本番
fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
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
```

2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル(h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge2.fasta"    #出力ファイル名を指定してout_fに格納
```

[トップページへ](#)

入力ファイルの保存2

①例題1をやります。②デスクトップ上に、(名前は何でもよいですが)hogeというフォルダを作成し、そこに入力として用いる③sample1.fastaを保存すべく...

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostings

ゲスト

更新

イントロ | 一般 | 指定した範囲の配列を取得 | Biostings

[Biostings](#)パッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルから様々な部分配列を取得するやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。例えば入力ファイルがヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。したがって、chr4とchr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロード時に、*.fastaとい

拡張子が*.txtに勝手に変更されることがありますのでご注意ください。また、「ディレクトリの変更」...

1. (single-)FASTA形式ファイル(sample1.fasta)

任意の範囲 (始点が3, 終点が9)の配列を抽出するやり方

```
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
param <- c(3, 9) #抽出する範囲を指定してparamに格納
```

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

```
#入力ファイルの読み込み
fasta <- readDNASTringSet(in_f, format="fasta")
```

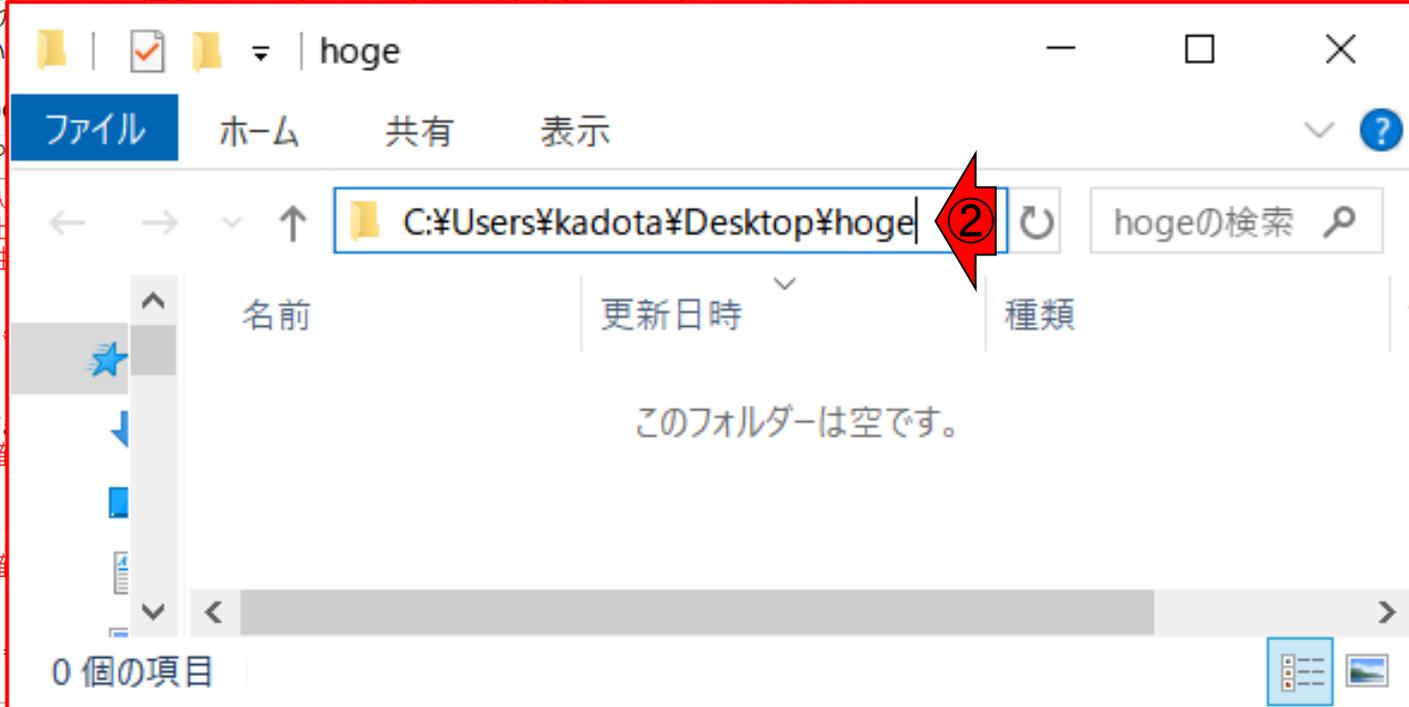
```
#本番
fasta <- subseq(fasta, param[1], param[2])
```

```
#ファイルに保存
writeXStringSet(fasta, file=out_f, format="fasta")
```

2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル(h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta" #入力ファイル名を指定してin_fに格納
out_f <- "hoge2.fasta" #出力ファイル名を指定してout_fに格納
```



[トップページへ](#)

入力ファイルの保存3

①例題1をやります。②デスクトップ上に、(名前は何でもよいですが)hogeというフォルダを作成し、そこに入力として用いる③sample1.fastaを保存すべく、右クリックで④保存。

(Rで)塩基配列解析 × +
iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostings

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

Biostringsパッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルから様々な部分配列を取得するやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。例えば入力ファイルがヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。したがって、chr4とchr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロード時に、*.fastaという拡張子が*.txtに勝手に変更されることがありますのでご注意ください。ここでは、

①「ファイル」 - 「ディレクトリの変更」 - ③「任意のファイル」を置いてあるディレクトリに移動し以下をコピー。

1. (single-)FASTA形式ファイル(sample1.fasta)の場合:

任意の範囲 (始点が3, 終点が9)の配列を抽出す

```
in_f <- "sample1.fasta"
out_f <- "hoge1.fasta"
param <- c(3, 9)
```

#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(in_f, format="fasta")
```

#本番

```
fasta <- subseq(fasta, param[1], param[2])
```

#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
```

新しいタブで開く(T)

新しいウィンドウで開く(W)

シークレット ウィンドウで開く(G)

名前を付けてリンク先を保存(K)...

リンクのアドレスをコピー(E)

検証(I)

Ctrl+Shift+I



2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル(h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta"
out_f <- "hoge2.fasta"
```

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

[トップページへ](#)

入力ファイルの保存4

①例題1をやります。②デスクトップ上に、(名前は何でもよいですが)hogeというフォルダを作成し、そこに入力として用いる③sample1.fastaを保存すべく、右クリックで④保存。⑤デスクトップ、⑥hogeを選択して、⑦保存。⑧ときどき勝手に.txtが追加されることがあるので注意してください。sample1.fastaが正解です。

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostings

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

Biostringsパッケージ中のsubseq関数を用いるやり方を示します。この項目は、「この染ヒトゲノムだった場合に、chr3の20000からchr8の配列のみ抽出といったやり方には対応する拡張子が*.txtに勝手に変更されることがあります。「ファイル」-「ディレクトリの変更」で解決

1. (single-)FASTA形式ファイル(sample1.fasta)

任意の範囲 (始点が3, 終点が9)の配列を抽出

```
in_f <- "sample1.fasta"
out_f <- "hoge1.fasta"
param <- c(3, 9)
```

```
#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(in_f, from=param[1], to=param[2])
```

```
#本番
```

```
fasta <- subseq(fasta, param[1], param[2])
```

```
#ファイルに保存
```

```
writeXStringSet(fasta, file=out_f, format="fasta")
```

2. RefSeqのhuman mRNAのmulti-FASTA形式ファイル

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta"
out_f <- "hoge2.fasta"
```

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

名前を付けて保存

PC > デスクトップ > hoge

整理 ▾ 新しいフォルダー

名前

hoge

更新日時

2020/03/04 16:28

ファイル名(N): sample1.fasta

ファイルの種類(I): FASTA ファイル (*.fasta)

保存(S) キャンセル

[トップページへ](#)

入力ファイルの保存5

①例題1をやります。②デスクトップ上に、(名前は何でもよいですが)hogeというフォルダを作成し、そこに入力として用いる③sample1.fastaを保存すべく、右クリックで④保存。⑤デスクトップ、⑥hogeを選択して、⑦保存。⑧ときどき勝手に.txtが追加されることがあるので注意してください。sample1.fastaが正解です。⑨hogeフォルダの中身が、⑩のようになればOK。

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostrings

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

Biostringsパッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルからやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。ヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。chr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロードが*.txtに勝手に変更されることがありますので、「ファイル」-「ディレクトリの変更」で解析したい

1. (single-)FASTA形式ファイル(sample1.fasta)

任意の範囲 (始点が3, 終点が9)の配列を抽出するやり方

```
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
param <- c(3, 9) #抽出範囲を指定してparamに格納
```

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

```
#入力ファイルの読み込み
fasta <- readDNASTringSet(in_f, format="fasta") #入力ファイルを読み込み
```

```
#本番
fasta <- subseq(fasta, param[1], param[2]) #本番
```

```
#ファイルに保存
writeXStringSet(fasta, file=out_f, format="fasta") #ファイルに保存
```

ファイル ホーム 共有 表示

C:\Users\kadota\Desktop\hoge

名前	更新日時	種類
sample1.fasta	2020/03/04 16:43	FASTA ファイル

1 個の項目

2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル (h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta" #入力ファイル名を指定してin_fに格納
out_f <- "hoge2.fasta" #出力ファイル名を指定してout_fに格納
```

[トップページへ](#)

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

作業ディレクトリの変更1

①hogeフォルダ(ディレクトリ)内にある、
②sample1.fastaが目的のファイルですが、現実には②と同じファイル名のものが別のフォルダにあるかもしれません。
作業ディレクトリの変更という作業は、RStudio上で作業する場所(ディレクトリ or フォルダ)を宣言すること。これによって、以降の作業を③「ファイル名のみ」を与えることで済ませられるので便利です。

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostings

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

Biostringsパッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルからやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。ヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。chr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロードが*.txtに勝手に変更されることがありますので、「ファイル」-「ディレクトリの変更」で解析したい

1. (single-)FASTA形式ファイル(sample1.fasta)の場合:

任意の範囲(始点が3, 終点が9)の配列を抽出するやり方です。

```
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
param <- c(3, 9) #抽出範囲を指定してparamに格納
```

```
#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(in_f, format="fasta")
```

```
#本番
fasta <- subseq(fasta, param[1], param[2])
```

```
#ファイルに保存
writeXStringSet(fasta, file=out_f, format="fasta")
```

hoge

ファイル ホーム 共有 表示

C:\Users\kadota\Desktop\hoge hogeの検索

名前	更新日時	種類
sample1.fasta	2020/03/04 16:43	FASTA ファイル

1 個の項目

2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル(h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲(例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta" #入力ファイル名を指定してin_fに格納
out_f <- "hoge2.fasta" #出力ファイル名を指定してout_fに格納
```

[トップページへ](#)

作業ディレクトリの変更2

この環境では、①の場所を作業ディレクトリに設定することになります。正確には、「C:¥Users...」という①の全ての情報(フルパス、といいます)を指定することで、作業ディレクトリがどこであっても②を読み込むことはできます。そういうやり方を教えるヒトもいると思います。

(Rで)塩基配列解析

x +

← → ↻ iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostrings

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

[Biostrings](#)パッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルから任意の部分配列を取得するやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。例えば入力ファイルがヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。したがって、chr4とchr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロード時に、*.fastaという拡張子が*.txtに勝手に変更されることがありますので「ファイル」-「ディレクトリの変更」で解析したい

1. (single-)FASTA形式ファイル(sample1.fasta)

任意の範囲 (始点が3, 終点が9)の配列を抽出するや

```
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
param <- c(3, 9) #抽出範囲を指定してparamに格納
```

```
#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(in_f, format="fasta")
```

```
#本番
fasta <- subseq(fasta, param[1], param[2])
```

```
#ファイルに保存
writeXStringSet(fasta, file=out_f, format="fasta")
```

hoge

ファイル ホーム 共有 表示

C:¥Users¥kadota¥Desktop¥hoge

名前	更新日時	種類
sample1.fasta	2020/03/04 16:43	FASTA ファイル

1 個の項目

2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル (h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta" #入力ファイル名を指定してin_fに格納
out_f <- "hoge2.fasta" #出力ファイル名を指定してout_fに格納
```

[トップページへ](#)

①をクリック。

作業ディレクトリの変更3

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. Below the menu bar is a toolbar with icons for file operations and a search bar. The main workspace is divided into several panes: Console (left), Environment (top right), History (top right), Connections (top right), Tutorial (top right), Files (bottom right), Plots (bottom right), Packages (bottom right), Help (bottom right), and Viewer (bottom right). The Files pane is currently active and shows the Home directory. A red arrow with the number 1 points to the Home icon in the Files pane.

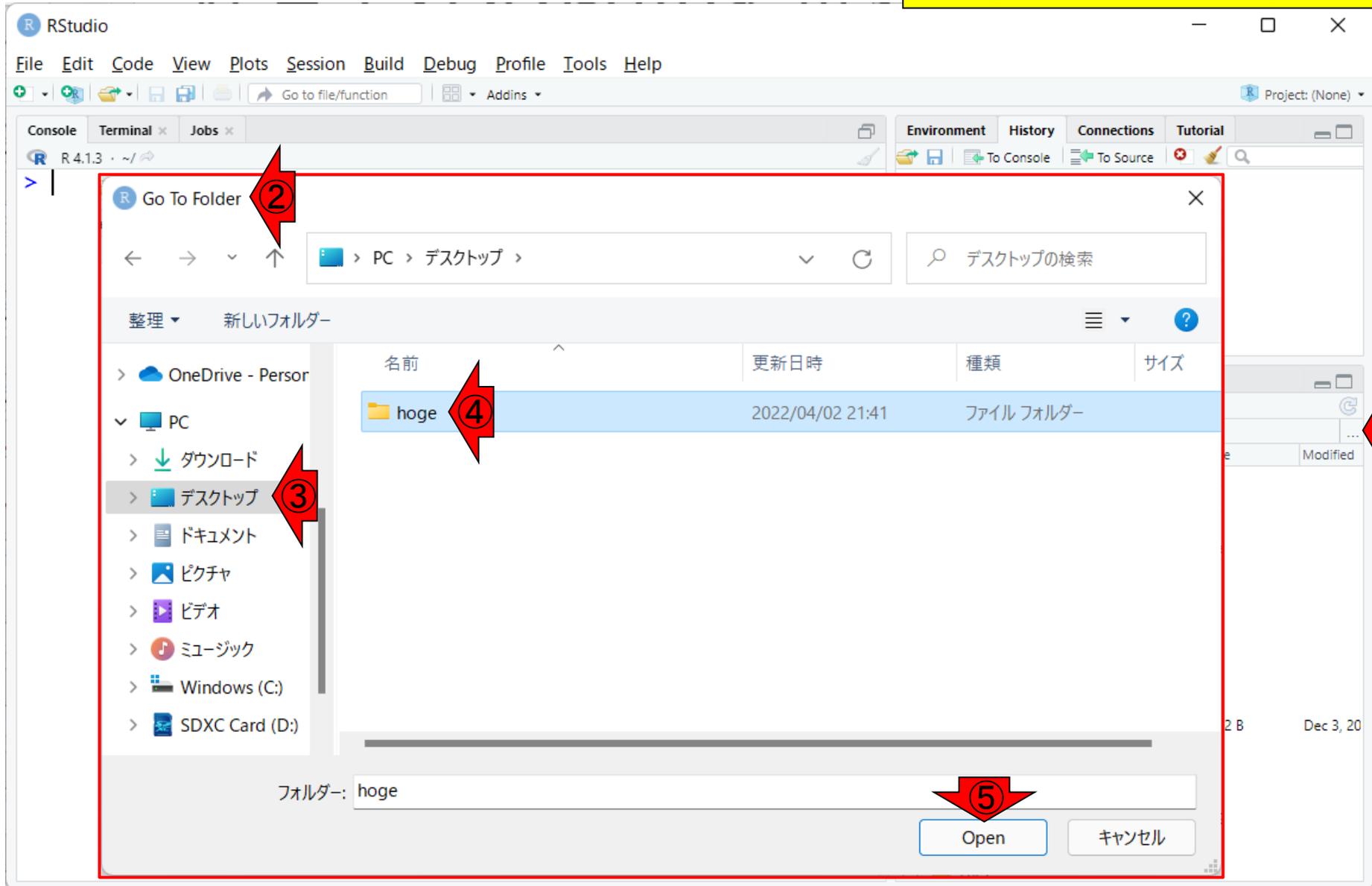
Environment History Connections Tutorial

Files Plots Packages Help Viewer

	Name	Size	Modified
<input type="checkbox"/>	その他		
<input type="checkbox"/>	Zoom		
<input type="checkbox"/>	R		
<input type="checkbox"/>	public_html		
<input type="checkbox"/>	Officeのカスタム テンプレート		
<input type="checkbox"/>	My Videos		
<input type="checkbox"/>	My Pictures		
<input type="checkbox"/>	My Music		
<input type="checkbox"/>	html		
<input type="checkbox"/>	desktop.ini	402 B	Dec 3, 20
<input type="checkbox"/>	2022		
<input type="checkbox"/>	2021		
<input type="checkbox"/>	2020		

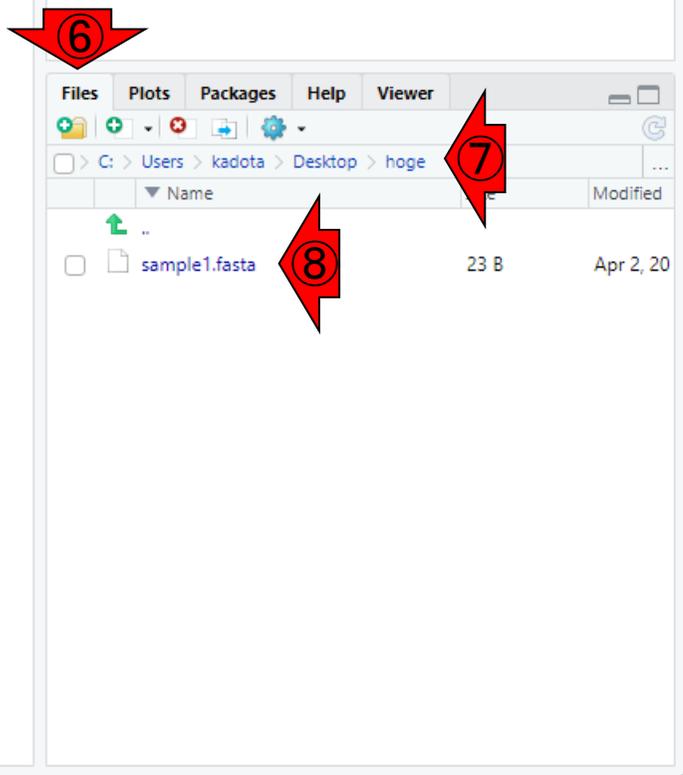
作業ディレクトリの変更4

①をクリック。②Go To Folderが出ますので、先ほど作成した③デスクトップの、④hogeを選択して、⑤Open。



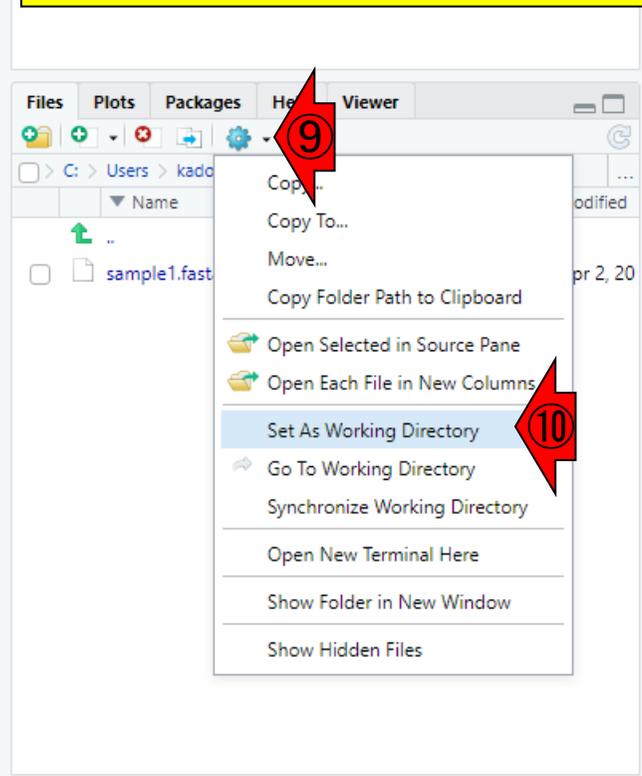
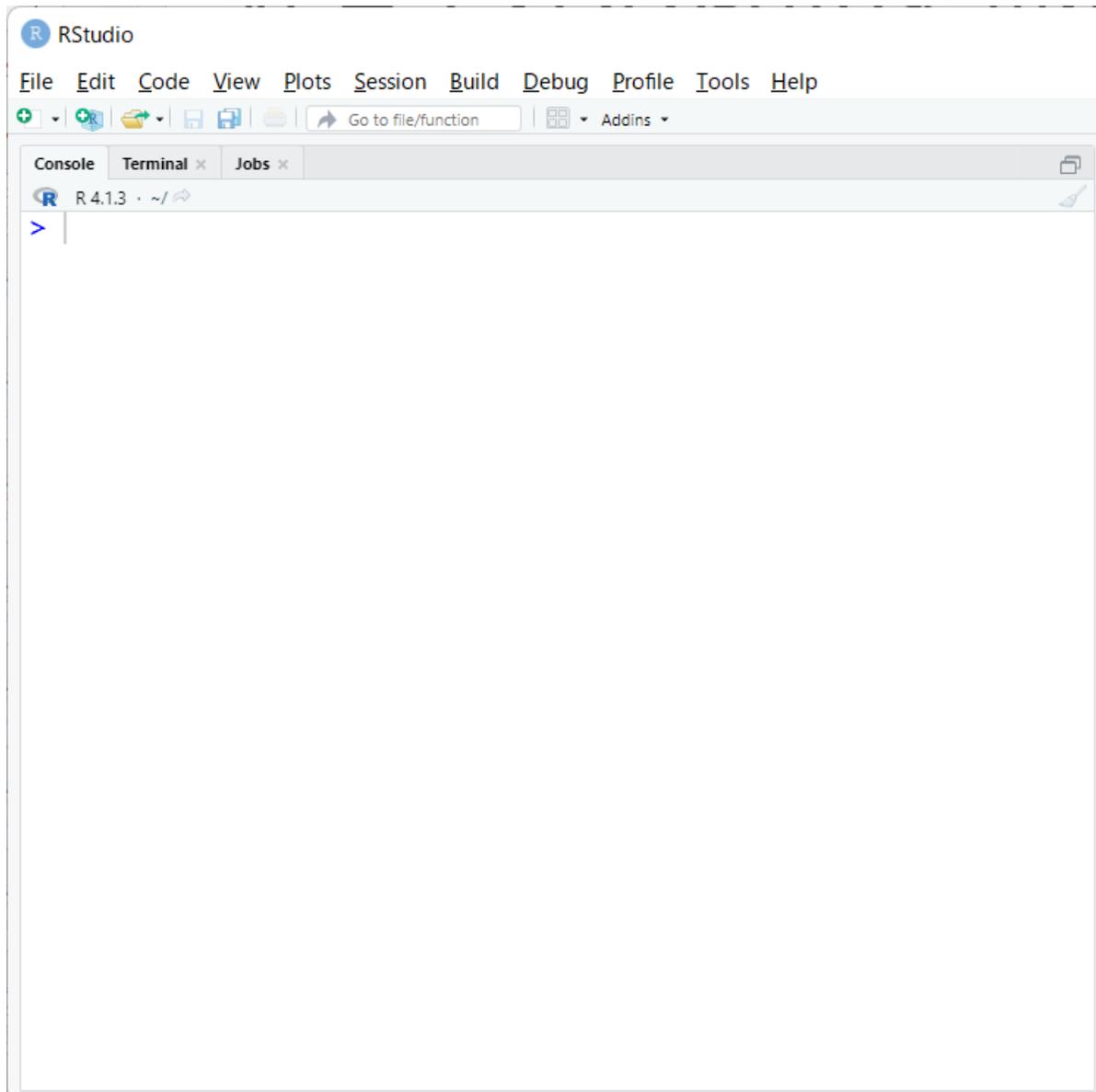
作業ディレクトリの変更5

①をクリック。②Go To Folderが出ますので、先ほど作成した③デスクトップの、④hogeを選択して、⑤Open。こんな感じになって、⑥Filesタブの部分が、⑦先ほど指定したフォルダになり、⑧ダウンロードしたファイル(sample1.fasta)が見えていことがわかります。この段階ではまだ作業ディレクトリの変更が完了していませんのでご注意ください。



作業ディレクトリの変更6

①をクリック。②Go To Folderが出ますので、先ほど作成した③デスクトップの、④hogeを選択して、⑤Open。こんな感じになって、⑥Filesタブの部分が、⑦先ほど指定したフォルダになり、⑧ダウンロードしたファイル(sample1.fasta)が見えていことがわかります。この段階ではまだ作業ディレクトリの変更が完了していませんのでご注意ください。⑨の歯車マーク、⑩Set As Working Directory。



作業ディレクトリの変更7

実行後の状態。⑪のコマンドが自動的に実行されます。ここまで行ってきたGUI上での作業が、⑪に相当することが理解できます。

The screenshot shows the RStudio interface. The console window displays the command `setwd("C:/Users/kadota/Desktop/hoge")` and a red arrow with the number 11 in a circle points to it. The Environment pane shows the command `setwd("C:/Users/kadota/Desktop/hoge...")`. The Files pane shows the current directory `C:/Users/kadota/Desktop/hoge` and a file `sample1.fasta` with a size of 23 B and a modification date of Apr 2, 20.

作業ディレクトリの変更8

実行後の状態。⑪のコマンドが自動的に実行されます。ここまで行ってきたGUI上での作業が、⑪に相当することが理解できます。逆にいえば、⑫の場所で作業すると決めておけば、⑪のコマンドで、GUIのマウスクリック作業を代用できるということです。

The screenshot shows the RStudio interface. The terminal window displays the command `setwd("C:/Users/kadota/Desktop/hoge")` with a red arrow labeled ⑪ pointing to it. Below the command, a red arrow labeled ⑫ points to the prompt `>`. The file explorer window shows the directory `C:/Users/kadota/Desktop/hoge` containing a file named `sample1.fasta`.

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

入力ファイルの確認1

①hogeフォルダ内の、②sample1.fastaをクリックしてみましょう。

The screenshot shows the RStudio interface. The console window on the left contains the command `setwd("C:/Users/kadota/Desktop/hoge")` and a cursor. The file explorer on the right shows the directory `C:/Users/kadota/Desktop/hoge` with a file named `sample1.fasta` (23 B) listed. Red arrows with circled numbers 1 and 2 point to the folder path and the file name, respectively.

入力ファイルの確認2

①hogeフォルダ内の、②sample1.fastaをクリックしてみましょう。こんな感じになります。

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The toolbar contains icons for file operations and a search bar. The main editor window displays the contents of 'sample1.fasta':

```
1 |>kadota
2 | AGTGACGGTCTT
3 |
```

The console window at the bottom shows the command `setwd("C:/Users/kadota/Desktop/hoge")` being executed. The file browser on the right shows the directory `C:\Users\kadota\Desktop\hoge` with a file named `sample1.fasta` (23 B). Red arrows with circled numbers 1 and 2 point to the 'hoge' folder and the 'sample1.fasta' file, respectively.

入力ファイルの確認3

①hogeフォルダ内の、②sample1.fastaをクリックしてみましょう。こんな感じになります。赤枠が、③sample1.fastaの中身に相当します。

The screenshot shows the RStudio interface with the following components:

- Top Panel:** File menu, toolbar with a red arrow labeled ③ pointing to the file icon, and a search bar.
- Editor:** A text editor window titled 'sample1.fasta' containing the following content:

```
1 |>kadota
2 | AGTGACGGTCTT
3 |
```

A red rectangle highlights this content, labeled ③.
- Environment/Console:** The console shows the command `setwd("C:/Users/kadota/Desktop/hoge")`.
- Files Panel:** A file explorer showing the directory `C:\Users\kadota\Desktop\hoge`. It contains a file named `sample1.fasta` (23 B), which is highlighted with a red arrow labeled ②. The parent directory `hoge` is highlighted with a red arrow labeled ①.
- Terminal:** The terminal window shows the command `> setwd("C:/Users/kadota/Desktop/hoge")`.

入力ファイルの確認4

①hogeフォルダ内の、②sample1.fastaをクリックしてみましょう。こんな感じになります。赤枠が、③sample1.fastaの中身に相当します。④行番号情報です。1行が長いデータの場合に、④を眺めることで行の区切りを把握できます。

The screenshot shows the RStudio interface with the following components:

- Top Panel:** RStudio logo and menu items (File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help).
- Toolbar:** Includes a search bar with "Go to file/function" and "Addins" dropdown.
- Source Editor:** Displays the content of "sample1.fasta" with line numbers 1, 2, and 3. A red box highlights the content: `1 |>kadota`, `2 |AGTGACGGTCTT`, and `3 |`. Red arrows point to the line numbers (labeled ④) and the content area (labeled ③).
- Environment Panel:** Shows the current working directory: `setwd("C:/Users/kadota/Desktop/hoge")`.
- Files Panel:** Shows the file explorer for the "hoge" directory. The file "sample1.fasta" is selected. Red arrows point to the directory path (labeled ①) and the file "sample1.fasta" (labeled ②).
- Console:** Shows the command `> setwd("C:/Users/kadota/Desktop/hoge")` being executed.

入力ファイルの確認5

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The toolbar contains icons for file operations and a search bar. The main editor window displays a FASTA file named 'sample1.fasta' with the following content:

```
1 >kadota
2 AGTGACGGTCTT
3
```

A red arrow with the number 1 points to the first line of the FASTA file. The right-hand pane shows the Environment, History, and Connections tabs. The History tab displays the command: `setwd("C:/Users/kadota/Desktop/hoge")`. The bottom pane shows the Console with the same command entered: `> setwd("C:/Users/kadota/Desktop/hoge")`. The Files pane on the right shows the current directory structure: `C:/Users/kadota/Desktop/hoge`, listing a folder `..` and a file `sample1.fasta` with a size of 23 B.

入力ファイルの確認6

今はカーソルが①の位置にあるので、②の位置に変更しただけ。

The screenshot shows the RStudio interface with the following components:

- Source Editor:** A text file named 'sample1.fasta' is open. The content is:

```
1 >kadota
2 AGTGACGGTCTT|
3
```

Red arrows point to the file name 'sample1.fasta' (labeled ①) and the cursor position at the end of line 2 (labeled ②).
- Terminal:** Shows the command `setwd("C:/Users/kadota/Desktop/hoge")` being executed in the directory `C:/Users/kadota/Desktop/hoge/`.
- Files Panel:** Shows the current directory `C:/Users/kadota/Desktop/hoge` containing a file named `sample1.fasta` with a size of 23 B.

入力ファイルの確認7

今はカーソルが①の位置にあるので、②の位置に変更しただけ。③の拡大図が示すように、①のカーソル位置「1:1」から②のカーソル位置「2:13」に変更されていることも分かります。そこそこのエディタには普通についている機能ではありませんが、ゲノム情報を解析する場合は、例えば②の位置が何番目の塩基に相当するのを知りたい場合があるので便利。

RStudio interface showing a text editor with a FASTA file. The editor displays the following content:

```
1 >kadota
2 AGTGACGGTCTT|
3
```

The cursor is positioned at the end of line 2. A red arrow labeled ① points to the cursor position. A red arrow labeled ② points to the position above the cursor. A red arrow labeled ③ points to a zoomed-in view of the text '2:13' in the status bar, indicating the current cursor position.

Console output:

```
> setwd("C:/Users/kadota/Desktop/hoge")
> |
```

File explorer showing the directory structure of the current working directory:

- Files
- Plots
- Packages
- Help
- Viewer

Actions: New Folder, Delete, Rename, More

Path: C: > Users > kadota > Desktop > hoge

Name	Size
..	
sample1.fasta	23 B

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

FASTA形式1

①赤枠で見えているのが「FASTA形式」です。

The screenshot shows the RStudio interface with a text editor window open to a file named 'sample1.fasta'. The editor contains the following text:

```
1 >kadota
2 AGTGACGGTCTT
3
```

A red box highlights the first two lines, and a red arrow labeled '1' points to it. The terminal window at the bottom shows the command:

```
> setwd("C:/Users/kadota/Desktop/hoge")
> |
```

The file browser on the right shows the current directory is 'C:/Users/kadota/Desktop/hoge' and contains a file named 'sample1.fasta' with a size of 23 B.

FASTA形式2

①赤枠で見えているのが「FASTA形式」です。最初に②「大なり記号(>)」、その後③任意の記述内容(この場合はkadota)からなる1行の「description行」があり、その次の行に④塩基 or アミノ酸配列がある(複数行になってもよい)ような形式です。

The screenshot shows the RStudio interface. The main editor window displays a file named 'sample1.fasta' with the following content:

```
1 >kadota
2 AGTGACGGTCTT
3
```

Red arrows with circled numbers point to specific parts of the FASTA format: arrow 2 points to the greater-than sign (>), arrow 3 points to the sequence identifier 'kadota', and arrow 4 points to the sequence 'AGTGACGGTCTT'. The 'Console' window at the bottom shows the command `setwd("C:/Users/kadota/Desktop/hoge")` being executed. The 'Files' pane on the right shows the directory structure: `C:\Users\kadota\Desktop\hoge` containing a file named `sample1.fasta` with a size of 23 B.

FASTA形式3

①赤枠で見えているのが「FASTA形式」です。最初に②「大なり記号(>)」、その後③任意の記述内容(この場合はkadota)からなる1行の「description行」があり、その次の行に④塩基 or アミノ酸配列がある(複数行になってもよい)ような形式です。⑤の部分は任意に変更可能なので、例えばdescription1にしてみると、⑥何か変更したことを意味するアスタリスク(*)が付加されます。

The screenshot displays the RStudio interface with a text editor window open to 'sample1.fasta'. The editor shows the following content:

```
1 >description1
2 AGTGACGGTCTT
3
```

Red arrows point to the asterisk in the description line (labeled 6) and the description text (labeled 5). The terminal window shows the command:

```
> setwd("C:/Users/kadota/Desktop/hoge")
> |
```

The file explorer window shows the directory 'C:/Users/kadota/Desktop/hoge/' containing the file 'sample1.fasta' with a size of 23 B.

FASTA形式4

①赤枠で見えているのが「FASTA形式」です。最初に②「大なり記号(>)」、その後③任意の記述内容(この場合はkadota)からなる1行の「description行」があり、その次の行に④塩基 or アミノ酸配列がある(複数行になってもよい)ような形式です。⑤の部分は任意に変更可能なので、例えばdescription1にしてみると、⑥何か変更したことを意味するアスタリスク(*)が付加されます。⑦を押して保存すると…

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. Below the menu is a toolbar with icons for file operations and a search bar. The main editor window displays a file named 'sample1.fasta*' with the following content:

```
1 >description1
2 AGTGACGGTCTT
3
```

A red arrow points to the asterisk at the end of line 1, labeled with a circled 7. The status bar at the bottom of the editor shows '1:14' and 'Text File'. Below the editor is a console window with the following text:

```
> setwd("C:/Users/kadota/Desktop/hoge")
> |
```

To the right of the console is a file explorer window showing the directory 'C:/Users/kadota/Desktop/hoge'. It contains a file named 'sample1.fasta' with a size of 23 B.

FASTA形式5

①赤枠で見えているのが「FASTA形式」です。最初に②「大なり記号(>)」、その後③任意の記述内容(この場合はkadota)からなる1行の「description行」があり、その次の行に④塩基 or アミノ酸配列がある(複数行になってもよい)ような形式です。⑤の部分は任意に変更可能なので、例えばdescription1にしてみると、⑥何か変更したことを意味するアスタリスク(*)が付加されます。⑦を押して保存すると、⑧*が消え、⑨kadotaの6文字からdescription1の12文字になったので増えた6文字分だけ、⑩ファイルサイズが大きくなっている(23 → 29 bytes)ことが分かります。

RStudio interface showing a FASTA file being edited. The editor displays the following content:

```
1 >description1
2 AGTGACGGTCTT
3
```

The console shows the command:

```
> setwd("C:/Users/kadota/Desktop/hoge")
> |
```

The file explorer at the bottom right shows the file 'sample1.fasta' with a size of 29 B.

FASTA形式6

①description行の文字数は13、②塩基配列の文字数は12。1文字あたり1 byteと計算します。Windowsの場合は1行あたりの改行コードに2 bytes分使いますので、③ $(13 + 12 + 2 + 2) = 29$ bytesという風になります。ゲノム情報の場合は、②の塩基配列情報のほうが圧倒的に①description行の情報よりも多いので、③「ファイルサイズ ≒ ゲノムサイズ」と解釈する場合があります。実際、ヒトゲノムは30億塩基対と言われますが、そのファイルサイズは約3GBです。

The screenshot shows RStudio with a text editor open to 'sample1.fasta'. The content is:

```
>description1
AGTGACGGTCTT
```

Red arrows point to the description line (labeled ①) and the sequence line (labeled ②). A '3' is shown below the description line. Below the editor is a terminal window with the command:

```
> setwd("C:/Users/kadota/Desktop/hoge")
```

To the right, a Windows Explorer window shows the file 'sample1.fasta' with a size of 29 B, indicated by a red arrow labeled ③.

FASTA形式7

赤枠がRのエディタ画面。①×を押して編集を終了させると…

The screenshot shows the RStudio interface. The main editor window displays a FASTA file named 'sample1.fasta' with the following content:

```
1 >description1
2 AGTGACGGTCTT
3
```

A red box highlights the editor window, and a red arrow points to the close button (marked with a circled '1') in the top-left corner of the editor's toolbar.

The terminal window at the bottom shows the command:

```
> setwd("C:/Users/kadota/Desktop/hoge")
> |
```

The file browser on the right shows the current directory 'C:/Users/kadota/Desktop/hoge' containing the file 'sample1.fasta' (29 B).

FASTA形式8

赤枠がRのエディタ画面。①×を押して編集を終了させると、こんな感じになります。

The screenshot shows the RStudio interface. The console window on the left displays the command `setwd("C:/Users/kadota/Desktop/hoge")` and a cursor on the next line. The file explorer on the right shows the directory `C:/Users/kadota/Desktop/hoge` containing a file named `sample1.fasta` with a size of 29 B.

Environment History Connections
setwd("C:/Users/kadota/Desk...

Files Plots Packages Help Viewer
New Folder Delete Rename More
C: > Users > kadota > Desktop > hoge
Name Size
..
sample1.fasta 29 B

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

コピペ実行1

おさらい。①この項目は、塩基配列のファイルを入力として読み込んで、指定した範囲の配列を切り出して、別のファイルに出力するスクリプトです。②例題1は、さきほどdescription行部分を変更した③sample1.fastaを入力として…

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostrings

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

Biostringsパッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルから様々な部分配列を取得するやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。例えば入力ファイルがヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。したがって、chr4とchr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロード時に、*.fastaという拡張子が*.txtに勝手に変更されることがありますのでご注意ください。ここでは、

「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. (single-)FASTA形式ファイル(sample1.fasta)の場合:

任意の範囲 (始点が3, 終点が9)の配列を抽出するやり方です。

```
in_f <- "sample1.fasta"
out_f <- "hoge1.fasta"
param <- c(3, 9)

#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(in_f, format="fasta")#in_fで指定したファイルの読み込み
fasta

#本番
fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
fasta

#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
```

2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル(h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta"
out_f <- "hoge2.fasta"
```

[トップページへ](#)

コピペ実行2

おさらい。①この項目は、塩基配列のファイルを入力として読み込んで、指定した範囲の配列を切り出して、別のファイルに出力するスクリプトです。②例題1は、さきほどdescription行部分を変更した③sample1.fastaを入力として、④3-9番目の塩基を切り出して、⑤hoge1.fastaというファイル名で保存するスクリプトです。

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostrings

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

Biostringsパッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルからやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。ヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。chr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロード拡張子が*.txtに勝手に変更されることがありますのでご注意ください。ここでは、「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. (single-)FASTA形式ファイル(sample1.fasta)の場合:

任意の範囲 (始点が3, 終点が9)の配列を抽出するやり方です。

```
in_f <- "sample1.fasta"
out_f <- "hoge1.fasta"
param <- c(3, 9)

#必要なパッケージのロード
library(Biostrings)

#入力ファイルの読み込み
fasta <- readDNASTringSet(in_f, format="fasta")#in_fで指定したファイルの読み込み
fasta

#本番
fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
fasta

#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
```

④

⑤

#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納
#抽出したい範囲の始点と終点を指定

#パッケージの読み込み

#確認してるだけです

#確認してるだけです

2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル(h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta"
out_f <- "hoge2.fasta"
```

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

[トップページへ](#)

コピペ実行3

おさらい。①この項目は、塩基配列のファイルを入力として読み込んで、指定した範囲の配列を切り出して、別のファイルに出力するスクリプトです。②例題1は、さきほどdescription行部分を変更した③sample1.fastaを入力として、④3-9番目の塩基を切り出して、⑤hoge1.fastaというファイル名で保存するスクリプトです。赤枠内を全選択して(反転させて)、⑥コピー。

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostings

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

Biostringsパッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルからやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。ヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。chr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロード拡張子が*.txtに勝手に変更されることがありますのでご注意ください。ここでは、「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

1. (single-)FASTA形式ファイル(sample1.fasta)の場合:

任意の範囲 (始点が3, 終点が9)の配列を抽出するやり方です。

```
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
param <- c(3, 9) #抽出したい範囲の始点と終点を指定

#必要なパッケージをロード
library(Biostrings)

#入力ファイル
fasta <- readFASTA(in_f) #入力ファイル名を...を検索(S)
fasta

#本番
fasta <- subseq(fasta, param) #出力ファイル名を指定してout_fに格納
fasta

#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
```

2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル(h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta" #入力ファイル名を指定してin_fに格納
out_f <- "hoge2.fasta" #出力ファイル名を指定してout_fに格納
```

[トップページへ](#)

コピペ実行4

The screenshot shows the RStudio interface. The console window is active and contains the following text:

```
> setwd("C:/Users/kadota/Desktop/hoge")  
> |
```

A red arrow with the number 7 points to the console window. A context menu is open over the console, with the 'Paste' option highlighted. A red arrow with the number 8 points to the 'Paste' option.

Cut
Copy
Paste
Select all
Reload
Inspect element

おさらい。①この項目は、塩基配列のファイルを入力として読み込んで、指定した範囲の配列を切り出して、別のファイルに出力するスクリプトです。②例題1は、さきほどdescription行部分を変更した③sample1.fastaを入力として、④3-9番目の塩基を切り出して、⑤hoge1.fastaというファイル名で保存するスクリプトです。赤枠内を全選択して(反転させて)、⑥コピー。⑦Consoleがアクティブな状態であることを確認して、⑧右クリックでペースト。(うまくいかないヒトは「Ctrl + V」もトライしてください。)

	Name	Size	Modified
	..		
<input type="checkbox"/>	sample1.fasta	29 B	Apr 3, 20

コピペ実行5

おさらい。①この項目は、塩基配列のファイルを入力として読み込んで、指定した範囲の配列を切り出して、別のファイルに出力するスクリプトです。②例題1は、さきほどdescription行部分を変更した③sample1.fastaを入力として、④3-9番目の塩基を切り出して、⑤hoge1.fastaというファイル名で保存するスクリプトです。赤枠内を全選択して(反転させて)、⑥コピー。⑦Consoleがアクティブな状態であることを確認して、⑧右クリックでペースト。(うまくいかないヒトは「Ctrl + V」もトライしてください。)こんな感じになればOK。⑨リターンキーを押す。

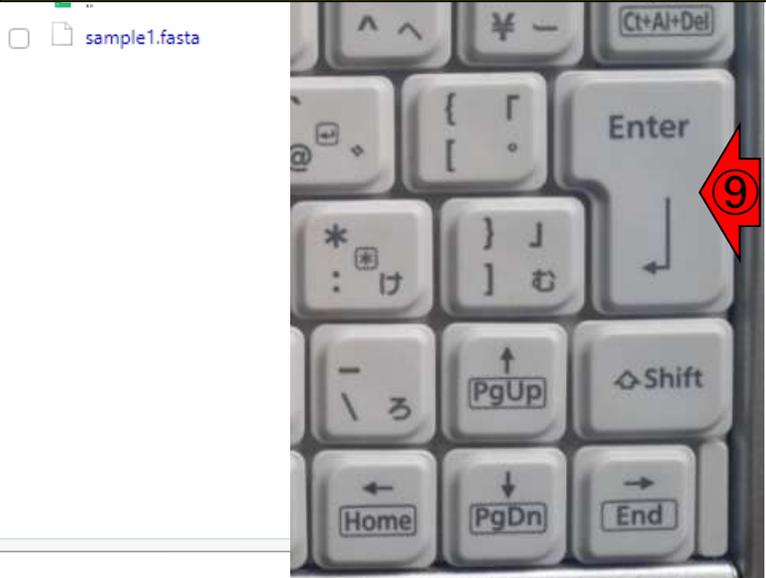
```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Console Terminal Jobs
R 4.1.3 · C:/Users/kadota/Desktop/hoge/
> setwd("C:/Users/kadota/Desktop/hoge")
> in_f <- "sample1.fasta"
#入力ファイル名を指定してin_fに格納
out_f <- "hoge1.fasta"
#出力ファイル名を指定してout_fに格納
param <- c(3, 9)
#抽出したい範囲の始点と終点を指定

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

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

#本番
fasta <- subseq(fasta, param[1], param[2])
)#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
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
```



コピペ実行6

実行完了後の状態。①赤枠内がこんな感じになればOK。出力ファイルである、②hoge1.fastaが作成されていることも分かります。もし①の画面になっていて、②が見られない場合は、③リロードしてみてください。

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Console Terminal Jobs
R 4.1.3 · C:/Users/kadota/Desktop/hoge/
ut_fに格納
> param <- c(3, 9) #抽出したい範囲の始点と終点を指定
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
> #入力ファイルの読み込み
> fasta <- readDNAStringSet(in_f, format="fasta") #in_fで指定したファイルの読み込み
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]   12 AGTGACGGTCTT description1
> #本番
> fasta <- subseq(fasta, param[1], param[2]) #paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]    7 TGACGGT      description1
> #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
>
```

Files Plots Packages Help Viewer
C: > Users > kadota > Desktop > hoge

Name	Size	Modified
..		
sample1.fasta	29 B	Apr 3, 20
hoge1.fasta	24 B	Apr 3, 20

コピペ実行7

実行完了後の状態。①赤枠内がこんな感じになればOK。出力ファイルである、②hoge1.fastaが作成されていることも分かります。もし①の画面になっていて、②が見られない場合は、③リロードしてみてください。④のような感じでエラーが出たヒトは、前提条件を満たしていない(この場合はBiostringsパッケージのインストールができてない)ことを意味します。しかるべき作業を行ってください。

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

Console Terminal x Jobs x

C:/Users/kadota/Desktop/hoge/ ↗

```
> setwd("C:/Users/kadota/Desktop/hoge")
```

```
> in_f <- "sample1.fasta"
```

を指定してin_fに格納

#入力ファイル

```
> out_f <- "hoge1.fasta"
```

を指定してout_fに格納

#出力ファイル名

>

```
> #必要なパッケージをロード
```

```
> library(Biostrings)
```

読み込み

#パッケージの読

library(Biostrings) でエラー:

'Biostrings' という名前のパッケージはありません

④

>

```
> #入力ファイルの読み込み
```

```
> fasta <- readDNAStringSet(in_f, format="fasta")#in_f
```

で指定したファイルの読み込み

readDNAStringSet(in_f, format = "fasta") でエラー:

関数 "readDNAStringSet" を見つけることができませんでした

```
> fasta
```

です

#確認してるだけ

BiocManager::install("Bios...")

Files Plots Packages Help Viewer

New Folder Delete Rename More

C: > Users > kadota > Desktop > hoge

	Name	Size
	..	
	sample1.fasta	29 B

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

出力ファイルの確認1

①が入力ファイルの情報、②が出力ファイルの情報に対応します。③が抽出したい領域の範囲に相当します。①が抽出元で、②が抽出結果に相当します。

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Project: (None)
Environment History Connections Tutorial
fasta <- readDNASTringSet(in_f, format="fasta", ...
fasta #確認してるだけです
#本番
fasta <- subseq(fasta, param[1], ...
fasta #確認してるだけです
#ファイルに保存
writeXStringSet(fasta, file=out_...
Files Plots Packages Help Viewer
C: > Users > kadota > Desktop > hoge
Name Size Modified
..
sample1.fasta 29 B Apr 3, 20
hoge1.fasta 24 B Apr 3, 20

R 4.1.3 · C:/Users/kadota/Desktop/hoge/
ut_fに格納
> param <- c(3, 9) #抽出したい範囲の始点と終点を指定
>
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
>
> #入力ファイルの読み込み
> fasta <- readDNASTringSet(in_f, format="fasta")#in_fで指定したファイルの読み込み
> fasta #確認してるだけです
DNASTringSet object of length 1:
  width seq names
[1] 12 AGTGACGGTCTT description1
>
> #本番
> fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
DNASTringSet object of length 1:
  width seq names
[1] 7 TGACGGT description1
>
> #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
>
```

出力ファイルの確認2

①が入力ファイルの情報、②が出力ファイルの情報に対応します。③が抽出したい領域の範囲に相当します。①が抽出元で、②が抽出結果に相当します。④が抽出領域に相当し、⑤と同じです。

The screenshot shows the RStudio interface. The terminal window displays the following R code and output:

```
R 4.1.3 · C:/Users/kadota/Desktop/hoge/
ut_fに格納
> param <- c(3, 9) #抽出したい範囲の始点と終点を指定
>
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
>
> #入力ファイルの読み込み
> fasta <- readDNAStringSet(in_f, format="fasta")#in_fで指定したファイルの読み込み
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]   12 AGTGACGGTCTT description1
>
> #本番
> fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]    7 TGACGGT      description1
>
> #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
>
```

The file explorer window shows the directory structure:

Name	Size	Modified
..		
sample1.fasta	29 B	Apr 3, 20
hoge1.fasta	24 B	Apr 3, 20

Red arrows and circles highlight key elements: ③ points to the `param` assignment, ④ points to the `subseq` function call, and ⑤ points to the `writeXStringSet` function call. A red bracket underlines the sequence `TGACGGT` in the output, which is also highlighted in the file explorer.

出力ファイルの確認3

①が入力ファイルの情報、②が出力ファイルの情報に対応します。③が抽出したい領域の範囲に相当します。①が抽出元で、②が抽出結果に相当します。④が抽出領域に相当し、⑤と同じです。⑥をクリックして、エディタを開いた状態です。⑦切り出された配列は、確かに④と同じ。

The screenshot shows the RStudio interface. The top pane displays a text editor for 'hoge1.fasta' with the following content:

```
1 >description1
2 TGACGGT
3
```

A red bracket underlines the sequence 'TGACGGT' on line 2, and a red arrow labeled '7' points to it.

The bottom pane shows the R console with the following code and output:

```
ut_1に格納
> param <- c(3, 9) #抽出したい範囲の始点と終点を指定
>
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
>
> #入力ファイルの読み込み
> fasta <- readDNAStringSet(in_f, format="fasta")#in_fで指定したファイルの読み込み
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]   12 AGTGACGGTCTT description1
>
> #本番
> fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
```

A red bracket underlines the sequence 'AGTGACGGTCTT' in the output, and a red arrow labeled '4' points to it.

```
fasta #確認してるだけです
#本番
fasta <- subseq(fasta, param[1], ...
fasta #確認してるだけです
#ファイルに保存
writeXStringSet(fasta, file=out_...
```

The screenshot shows a file explorer window for the directory 'C:\Users\kadota\Desktop\hoge'. It contains two files:

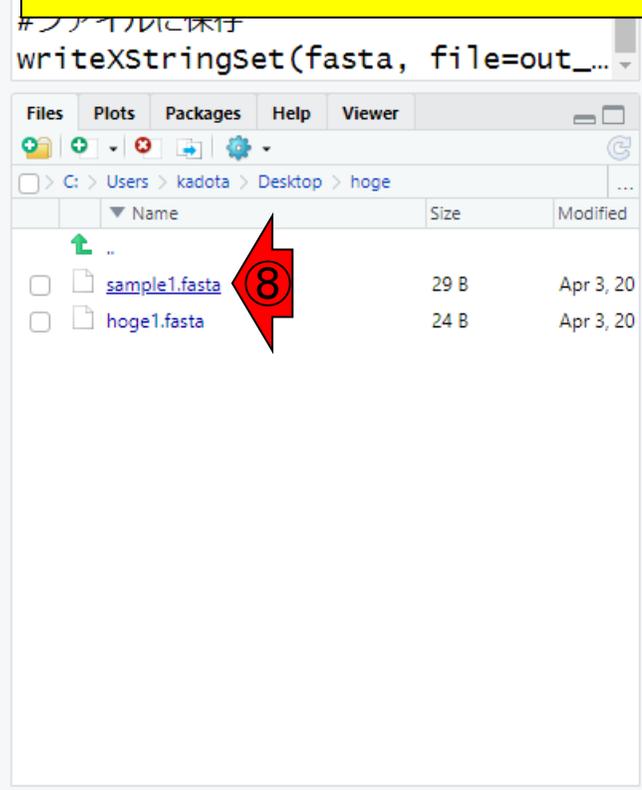
Name	Size	Modified
sample1.fasta	29 B	Apr 3, 20
hoge1.fasta	24 B	Apr 3, 20

A red arrow labeled '6' points to the 'hoge1.fasta' file.

出力ファイルの確認4

①が入力ファイルの情報、②が出力ファイルの情報に対応します。③が抽出したい領域の範囲に相当します。①が抽出元で、②が抽出結果に相当します。④が抽出領域に相当し、⑤と同じです。⑥をクリックして、エディタを開いた状態です。⑦切り出された配列は、確かに④と同じ。⑧入力ファイルのほうもクリックした状態。⑨のような感じでタブが増えていくだけです。

```
RStudio
File Edit View Plots Session Build Debug Profile Tools Help
hoge1.fasta x sample1.fasta x
1 >description1
2 AGTGACGGTCTT
3
1:14 Text file
Console Terminal x Jobs x
R 4.1.3 · C:/Users/kadota/Desktop/hoge/
> param <- c(3, 9) #抽出したい範囲の始点と終点を指定
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
> #入力ファイルの読み込み
> fasta <- readDNAStringSet(in_f, format="fasta")#in_fで指定したファイルの読み込み
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]   12 AGTGACGGTCTT description1
> #本番
> fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
```



Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

Environmentタブ1

①Environmentタブをアクティブにした状態です。現時点では、こういうタブもあるのだという程度でよいですが、慣れてくると非常に便利です。

The screenshot shows the RStudio interface. The Environment tab is active, displaying the 'fasta' object with its structure. The terminal window shows the R code used to read the FASTA file and extract a specific sequence.

```
R 4.1.3 · C:/Users/kadota/Desktop/hoge/
> param <- c(3, 9) #抽出したい範囲の始点と終点を指定
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
> #入力ファイルの読み込み
> fasta <- readDNAStringSet(in_f, format="fasta")#in_fで指定したファイルの読み込み
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]   12 AGTGACGGTCTT description1
> #本番
> fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
```

Environment	History	Connections	Tutorial
fasta	Formal class DNAST...		
Values			
in_f	"sample1.fasta"		
out_f	"hoge1.fasta"		
param	num [1:2] 3 9		

Name	Size	Modified
..		
sample1.fasta	29 B	Apr 3, 20
hoge1.fasta	24 B	Apr 3, 20

Environmentタブ2

①Environmentタブをアクティブにした状態です。現時点では、こういうタブもあるのだという程度でよいですが、慣れてくると非常に便利です。ここは、シンプルに言えば、「現在利用可能なオブジェクト」がリストアップされている部分です。

The screenshot shows the RStudio interface. The script editor contains a FASTA file with a description and a sequence. The console shows the execution of R code to read the FASTA file and extract a specific sequence. The Environment pane on the right shows the 'fasta' object and its values.

```
1 >description1
2 AGTGACGGTCTT
3
```

```
> param <- c(3, 9) #抽出したい範囲の始点と終点を指定
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
> #入力ファイルの読み込み
> fasta <- readDNAStringSet(in_f, format="fasta")#in_fで指定したファイルの読み込み
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]   12 AGTGACGGTCTT description1
> #本番
> fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
```

Data	
fasta	Formal class DNAST...
Values	
in_f	"sample1.fasta"
out_f	"hoge1.fasta"
param	num [1:2] 3 9

Files | Plots | Packages | Help | Viewer

Name	Size	Modified
..		
sample1.fasta	29 B	Apr 3, 20
hoge1.fasta	24 B	Apr 3, 20

Environmentタブ3

例えば、②は切り出したい領域の始点と終点を、2つの要素からなる数値ベクトルとして表現し、それをparamという名前で利用できるようにするコマンドです。それぞれの独立した整数を連結(concatenate)すべく、③cという関数で連結してスカラーをベクトルにしています。

The screenshot shows the RStudio interface. The script editor contains the following code:

```
1 >description1
2 AGTGACGGTCTT
3
```

The console shows the execution of the script with Japanese comments:

```
R 4.1.3 · C:/.../Desktop/hoge
> param <- c(3, 9) #抽出したい範囲の始点と終点を指定
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
> #入力ファイルの読み込み
> fasta <- readDNAStringSet(in_f, format="fasta")#in_fで指定したファイルの読み込み
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]   12 AGTGACGGTCTT description1
> #本番
> fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
```

The Environment pane on the right shows the following data:

Data	Formal class	DNAST...
fasta	Formal class	DNAST...
Values		
in_f	"sample1.fasta"	
out_f	"hoge1.fasta"	
param	num [1:2] 3 9	

The Files pane at the bottom shows the directory structure:

Name	Size	Modified
..		
sample1.fasta	29 B	Apr 3, 20
hoge1.fasta	24 B	Apr 3, 20

Environmentタブ4

例えば、②は切り出したい領域の始点と終点を、2つの要素からなる数値ベクトルとして表現し、それをparamという名前で見られるようにするコマンドです。それぞれの独立した整数を連結 (concatenate) するべく、③cという関数で連結してスカラーをベクトルにしています。④が左向きの矢印っぽくなっていることからある程度予想できますが、「右側の情報を左側のものに代入せよ」という意味です。プログラミングの世界では、通常この向きで物事を表現しますので、「<-」の代わりに「=」が用いられることもあります。

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
hoge1.fasta x sample1.fasta x
1 >description1
2 AGTGACGGTCTT
3
1:14 Text file
Console Terminal Jobs
R 4.1.3 · C:/... Desktop/hog...
> param <- c(3, 9) #抽出したい範囲の始点と終点を指定
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
> #入力ファイルの読み込み
> fasta <- readDNAStringSet(in_f, format="fasta")#in_fで指定したファイルの読み込み
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]   12 AGTGACGGTCTT description1
> #本番
> fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
```

Name	Size	Modified
..		
sample1.fasta	29 B	Apr 3, 20
hoge1.fasta	24 B	Apr 3, 20

Environmentタブ5

それゆえ、⑤paramというオブジェクトの中身は、2つの要素からなる数値ベクトルです。1番目の要素が3で、2番目の要素が9です。それを踏まえると、⑥の意味がよくわかると思います。

The screenshot shows the RStudio interface. The script editor contains the following code:

```
1 >description1
2 AGTGACGGTCTT
3
```

The console shows the execution of the following R code:

```
> param <- c(3, 9) #抽出したい範囲の始点と終点を指定
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
> #入力ファイルの読み込み
> fasta <- readDNAStringSet(in_f, format="fasta")#in_fで指定したファイルの読み込み
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]   12 AGTGACGGTCTT description1
> #本番
> fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
```

The Environment pane on the right shows the following data:

Data	Formal class	DNAST...
fasta	Formal class	DNAST...
Values		
in_f	"sample1.fasta"	
out_f	"hoge1.fasta"	
param	num [1:2] 3 9	

Red arrows labeled ⑤ and ⑥ point to the `param` variable in the console and the `param` row in the Environment pane, respectively.

Environmentタブ6

それゆえ、⑤paramというオブジェクトの中身は、2つの要素からなる数値ベクトルです。1番目の要素が3で、2番目の要素が9です。それを踏まえると、⑥の意味がよくわかると思います。⑦paramオブジェクトの1番目の要素はparam[1]、⑧2番目の要素はparam[2]として表現できます。⑨は部分配列(subsequence)の抽出を行うsubseq関数を用いて、⑩の情報を、⑪で入力配列、⑦で始点、⑧で終点を与えて目的の部分配列を抽出していると読み解けばよいです。

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
hoge1.fasta x sample1.fasta x
1 >description1
2 AGTGACGGTCTT
3
1:14 Text file
Console Terminal x Jobs x
R 4.1.3 · C:/Users/kadota/Desktop/hoge/
> param <- c(3, 9) #抽出したい範囲の始点と終点を指定
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
> #入力ファイルの読み込み
> fasta <- readDNAStringSet(in_f, format="fasta")#in_fで指定したファイルの読み込み
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]   12 AGTGACGGTCTT description1
> #本番
> fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
```

Files Plots Packages Help Viewer

C:\Users\kadota\Desktop\hoge

Name	Size	Modified
..		
sample1.fasta	29 B	Apr 3, 20
hoge1.fasta	24 B	Apr 3, 20

Environmentタブ7

それゆえ、⑤paramというオブジェクトの中身は、2つの要素からなる数値ベクトルです。1番目の要素が3で、2番目の要素が9です。それを踏まえると、⑥の意味がよくわかると思います。⑦paramオブジェクトの1番目の要素はparam[1]、⑧2番目の要素はparam[2]として表現できます。⑨は部分配列(subsequence)の抽出を行うsubseq関数を用いて、⑩の情報を、⑪で入力配列、⑦で始点、⑧で終点を与えて目的の部分配列を抽出していると読み解けばよいです。Rのエディタ部分を削除した状態。⑩と⑫は同じオブジェクト名であるにもかかわらず、中身が異なるのは、⑨subseq関数の実行結果を同じオブジェクト名(fasta)で保存しているからだと理解すればよいです。

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Console Terminal Jobs
R 4.1.3 · C:/Users/kadota/Desktop/hoge/
点を指定
>
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
>
> #入力ファイルの読み込み
> fasta = readDNAStringSet(in_f, format="fasta")#in_fで指定した
ファイルの読み込み
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]   12 AGTGACGGTCTT description1
>
> #本番
> fasta = subseq(fasta, param[1], param[2])#paramで指定した始点
と終点の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
DNAStringSet object of length 1:
  width seq          names
[1]    7 TGACGGT      description1
>
> #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
> View(fasta)
> View(fasta)
> |
```

<input type="checkbox"/>	sample1.fasta	29 B	Apr 3, 20
<input type="checkbox"/>	hoge1.fasta	24 B	Apr 3, 20

Environmentタブ8

①をクリックするとfastaオブジェクトの中身が見られますが、話がややこしくなるのと、まだ難解な事柄であるため無視してよいです。②は、対応関係が理解しやすいと思います。

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Console Terminal Jobs
R 4.1.3 · C:/Users/kadota/Desktop/hoge/
点を指定
>
> #必要なパッケージをロード
> library(Biostrings) #パッケージの読み込み
>
> #入力ファイルの読み込み
> fasta <- readDNASTringSet(in_f, format="fasta")#in_fで指定した
ファイルの読み込み
> fasta #確認してるだけです
DNASTringSet object of length 1:
  width seq          names
[1]   12 AGTGACGGTCTT description1
>
> #本番
> fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点
と終点の範囲の配列を抽出した結果をfastaに格納
> fasta #確認してるだけです
DNASTringSet object of length 1:
  width seq          names
[1]    7 TGACGGT      description1
>
> #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
> view(fasta)
> view(fasta)
> |
```

Environment History Connections Tutorial
R Global Environment
Data
fasta Formal class DNAST...
Values
in_f "sample1.fasta"
out_f "hoge1.fasta"
param num [1:2] 3 9

Files Plots Packages Help Viewer
C: > Users > kadota > Desktop > hoge
Name Size Modified
..
sample1.fasta 29 B Apr 3, 20
hoge1.fasta 24 B Apr 3, 20

Environmentタブ9

①をクリックするとfastaオブジェクトの中身が見られますが、話がややこしくなるのと、まだ難解な事柄であるため無視してよいです。②は、対応関係が理解しやすいと思います。理由は、③で自分が指定している内容だからです。



(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostrings

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

[Biostrings](#)パッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルから様々な部分配列を取得するやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。例えば入力ファイルがヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。したがって、chr4とchr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロード時に、*.fastaという拡張子が*.txtに勝手に変更されることがありますのでご注意ください。ここでは、

「ファイル」 - 「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し以下をコピー。

1. (single-)FASTA形式ファイル(sample1.fasta)の場合:

任意の範囲 (始点が3, 終点が9)の配列を抽出するやり方です。

```
in_f <- "sample1.fasta"
out_f <- "hoge1.fasta"
param <- c(3, 9)
```



#入力ファイル名を指定してin_fに格納
#出力ファイル名を指定してout_fに格納
#抽出したい範囲の始点と終点を指定

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

#パッケージの読み込み

```
#入力ファイルの読み込み
```

```
fasta <- readDNASTringSet(in_f, format="fasta")#in_fで指定したファイルの読み込み
fasta
```

#確認してるだけです

```
#本番
```

```
fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
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
```

2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル(h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta"
out_f <- "hoge2.fasta"
```

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

[トップページへ](#)

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- **フィルタリング(任意のキーワードを含む行を抽出)**
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

①で行う内容自体は非常に原始的なものであり、実際には同じ目的を果たすより美しい手段が存在します。しかし、**最初のうちは原始的な方法のほうが理解しやすい**と思います。②のあたりもしっかり理解し、Rを使う**他の先生の講義の中身に集中**できるようにしておきましょう。



全体像の把握1

アノテーションファイル(annotation.txt)中の第1列目に対して、リストファイル(genelist1.txt)中の文字列と一致する行を抜き出して、hoge1.txtというファイル名で出力したい。

入力:アノテーションファイル(annotation.txt)

	A	B	C	D
1	gene name	accession	description	subcellular_location
2	gene1	hoge01	plasma_mem	nuclear
3	gene2	hoge02	hohinu	membrane
4	gene3	hoge03	agribio	endoplasmic
5	gene4	hoge04	genesis	endoplasmic
6	gene5	hoge05	kamo	membrane
7	gene6	hoge06	netteba	humei
8	gene7	hoge07	tebasaki	nuclear
9	gene8	hoge08	biiru	nuclear
10	gene9	hoge09	nihonshu	nuclear
11	gene10	hoge10	agene1	membrane
12	gene11	hoge11	iyaaaa	endoplasmic

出力:hoge1.txt



	A	B	C	D
1	gene name	accession	description	subcellular_location
2	gene1	hoge01	plasma_mem	nuclear
3	gene7	hoge07	tebasaki	nuclear
4	gene9	hoge09	nihonshu	nuclear

入力:リストファイル(genelist1.txt)

	A
1	gene1
2	gene7
3	gene9

全体像の把握2

アノテーションファイル(annotation.txt)中の第1列目に対して、リストファイル(genelist1.txt)中の文字列と一致する行を抜き出して、hoge1.txtというファイル名で出力したい。①を押して一番上に移動。

(Rで)塩基配列解析

x +

iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_subseq_biostrings

イントロ | 一般 | 指定した範囲の配列を取得 | Biostrings

[Biostrings](#)パッケージ中のsubseq関数を用いて、single-FASTA形式やmulti-FASTA形式ファイルから様々な部分配列を取得するやり方を示します。この項目は、「この染色体の、ここから、ここまで」という指定の仕方になります。例えば入力ファイルがヒトゲノムだった場合に、chr3の20000から500000の座標の配列取得を行いたい場合などに利用します。したがって、chr4とchr8の配列のみ抽出といったやり方には対応していませんのでご注意ください。また、ファイルダウンロード時に、*.fastaという拡張子が*.txtに勝手に変更されることがありますのでご注意ください。ここでは、

「ファイル」 - 「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し以下をコピー。

1. (single-)FASTA形式ファイル(sample1.fasta)の場合:

任意の範囲 (始点が3, 終点が9)の配列を抽出するやり方です。

```
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
param <- c(3, 9)           #抽出したい範囲の始点と終点を指定

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

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

#本番
fasta <- subseq(fasta, param[1], param[2])#paramで指定した始点と終点の範囲の配列を抽出した結果をfastaに格納
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
```

2. RefSeqのhuman mRNAのmulti-FASTA形式のファイル(h_rna.fasta)の場合:

任意のRefSeq ID (例: NM_203348.1)の任意の範囲 (例: 始点が2, 終点が5)の配列の抽出を行うやり方です。

```
in_f <- "h_rna.fasta"      #入力ファイル名を指定してin_fに格納
out_f <- "hoge2.fasta"    #出力ファイル名を指定してout_fに格納
```

[トップページへ](#)

①

全体像の把握3

アノテーションファイル(annotation.txt)中の第1列目に対して、リストファイル(genelist.txt)中の文字列と一致する行を抜き出して、hoge1.txtというファイル名で出力したい。①を押して一番上に移動。②さきほど行った項目の上部にある、③をクリック。

- (Rで)塩基配列解析
- iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#
- 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 | について](#) (last modified 2018/11/13)
 - インストール | Rパッケージ | [必要最小限プラスアルファ](#) (last modified 2022/03/30) 推奨 NEW
 - インストール | Rパッケージ | [個別\(2018年11月以降\)](#) (last modified 2022/03/31) NEW
 - インストール | Rパッケージ | [個別\(2018年11月以前\)](#) (last modified 2018/11/12)
 - [基本的な利用法](#) (last modified 2020/03/16)
 - [サンプルデータ](#) (last modified 2019/09/06)
 - 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\(基礎\)](#) ③ (last modified 2019/04/24)
 - イントロ | 一般 | [ランダムな塩基配列を生成](#) (last modified 2014/06/16)
 - イントロ | 一般 | [任意の長さの可能な全ての塩基配列を作成](#) (last modified 2015/02/19)
 - 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 | について](#) (last modified 2019/09/06)
 - イントロ | 一般 | [指定した範囲の配列を取得 | Biostring](#) ② (last modified 2019/09/06)
 - 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 | について](#) (last modified 2019/09/06)
 - イントロ | 一般 | [翻訳配列\(translate\)を取得 | Biostrings](#) (last modified 2015/09/12)
 - イントロ | 一般 | [翻訳配列\(translate\)を取得 | seqinr\(Charif_2005\)](#) (last modified 2015/03/09)
 - イントロ | 一般 | [相補鎖\(complement\)を取得](#) (last modified 2019/03/10)
 - イントロ | 一般 | [逆相補鎖\(reverse complement\)を取得](#) (last modified 2019/03/10)
 - イントロ | 一般 | [逆鎖\(reverse\)を取得](#) (last modified 2019/03/10)
 - イントロ | 一般 | k-mer解析 | k=1(塩基ごとの出現頻度解析) | [Biostrings](#) (last modified 2016/04/27)
 - イントロ | 一般 | k-mer解析 | k=2(2連続塩基の出現頻度解析) | [Biostrings](#) (last modified 2016/01/28)
 - イントロ | 一般 | k-mer解析 | k=3(3連続塩基の出現頻度解析) | [Biostrings](#) (last modified 2016/01/28)
 - イントロ | 一般 | k-mer解析 | k=n(n連続塩基の出現頻度解析) | [Biostrings](#) (last modified 2016/05/01)

[トップページへ](#)

こんな感じになります。①例題1が一番上になるように、5行分ほど下部に移動。

全体像の把握4

(Rで)塩基配列解析

保護されていない通信 | iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_grep_kiso

イントロ | 一般 | 任意のキーワードを含む行を抽出(基礎)

例えばタブ区切りテキストファイルが手元があり、この中からリストファイル中の文字列を含む行を抽出するやり方を示します。Linux (UNIX)のgrepコマンドのようなものであり、perlのハッシュのようなものです。

「①ファイル」 - 「ディレクトリの変更」で解析したいファイルを置いてあるディレクトリに移動し以下をコピー。

1. 目的のタブ区切りテキストファイル([annotation.txt](#))中の第1列目をキーとして、リストファイル([genelist1.txt](#))中のものが含まれる行全体を出力したい場合：

```

in_f1 <- "annotation.txt" #入力ファイル名を指定してin_f1に格納(アノテーションファイル)
in_f2 <- "genelist1.txt" #入力ファイル名を指定してin_f2に格納(リストファイル)
out_f <- "hoge1.txt" #出力ファイル名を指定してout_fに格納
param <- 1 #アノテーションファイル中の検索したい列番号を指定

#入力ファイルの読み込み
data <- read.table(in_f1, header=TRUE, sep="\t", quote="")#in_f1で指定したファイルの読み込み
keywords <- readLines(in_f2) #in_f2で指定したファイルの読み込み
dim(data) #オブジェクトdataの行数と列数を表示

#本番
obj <- is.element(as.character(data[,param]), keywords)#条件を満たすかどうかを判定した結果をobjに格納
out <- data[obj,] #objがTRUEとなる行のみ抽出した結果をoutに格納
dim(out) #オブジェクトoutの行数と列数を表示

#ファイルに保存
write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)#outの中身を指定したファイル名で保

```

[トップページへ](#)

全体像の把握5

こんな感じになります。①例題1が一番上になるように、5行分ほど下部に移動。こんな感じ。

1. 目的のタブ区切りテキストファイル([annotation.txt](#))中の第1列目をキーとして、リストファイル([genelist1.txt](#))中のものが含まれる行全体を出力したい場合：

```
in_f1 <- "annotation.txt" #入力ファイル名を指定してin_f1に格納(アノテーションファイル)
in_f2 <- "genelist1.txt" #入力ファイル名を指定してin_f2に格納(リストファイル)
out_f <- "hoge1.txt" #出力ファイル名を指定してout_fに格納
param <- 1 #アノテーションファイル中の検索したい列番号を指定

#入力ファイルの読み込み
data <- read.table(in_f1, header=TRUE, sep="\t", quote="") #in_f1で指定したファイルの読み込み
keywords <- readLines(in_f2) #in_f2で指定したファイルの読み込み
dim(data) #オブジェクトdataの行数と列数を表示

#本番
obj <- is.element(as.character(data[,param]), keywords) #条件を満たすかどうかを判定した結果をobjに格納
out <- data[obj,] #objがTRUEとなる行のみ抽出した結果をoutに格納
dim(out) #オブジェクトoutの行数と列数を表示

#ファイルに保存
write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F) #outの中身を指定したファイル名で保
```

[トップページへ](#)

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

入力ファイルの保存1

こんな感じになります。①例題1が一番上になるように、5行分ほど下部に移動。こんな感じ。②が入力ファイルであり、③と④からダウンロード可能なので、右クリックで「デスクトップ上にあるhogeフォルダ」に保存。

(Rで)塩基配列解析

保護されていない通信 | iu.asia.jp/~kadota/r_seq.html#intro_g

1. 目的のタブ区切りテキストファイル([annotation.txt](#))中の第1列目をキーとして、リストファイル([genelist1.txt](#))中のものが含まれる行全体を出力したい場合：

```
in_f1 <- "annotation.txt"
in_f2 <- "genelist1.txt"
out_f <- "hoge1.txt"
param <- 1
```

#入力ファイルの読み込み

```
data <- read.table(in_f1, header=TRUE, sep="\t", quote="")
keywords <- readLines(in_f2)
dim(data)
```

#本番

```
obj <- is.element(as.character(data[,param]), keywords)
out <- data[obj,]
dim(out)
```

#ファイルに保存

```
write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)
```

#入力ファイル名を指定してin_f1に格納(アノテーションファイル)
#入力ファイル名を指定してin_f2に格納(リストファイル)
#出力ファイル名を指定してout_fに格納
#アノテーションファイル中の検索したい列番号を指定

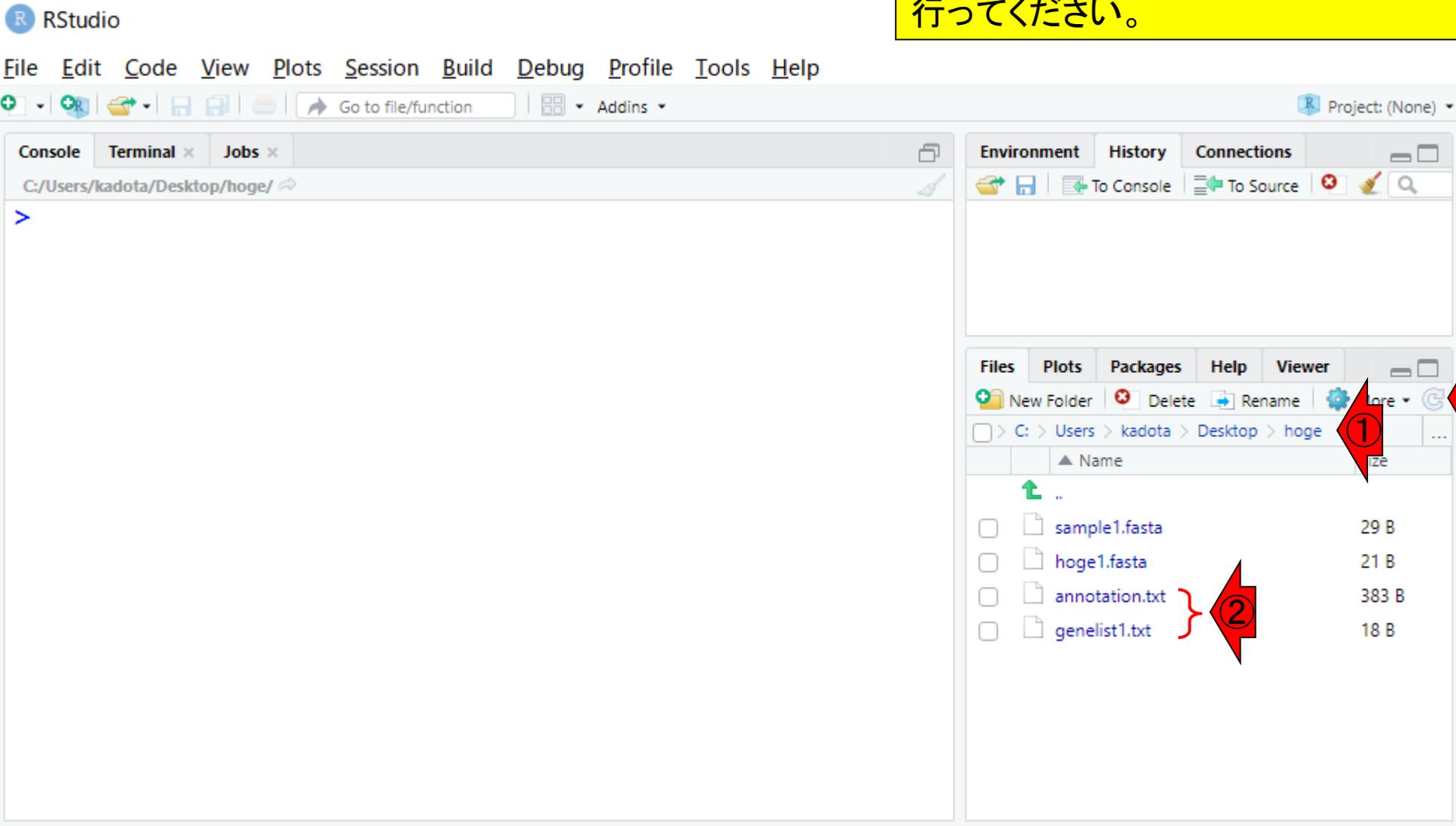
#in_f1で指定したファイルの読み込み
#in_f2で指定したファイルの読み込み
#オブジェクトdataの行数と列数を表示

#条件を満たすかどうかを判定した結果をobjに格納
#objがTRUEとなる行のみ抽出した結果をoutに格納
#オブジェクトoutの行数と列数を表示

[トップページへ](#)

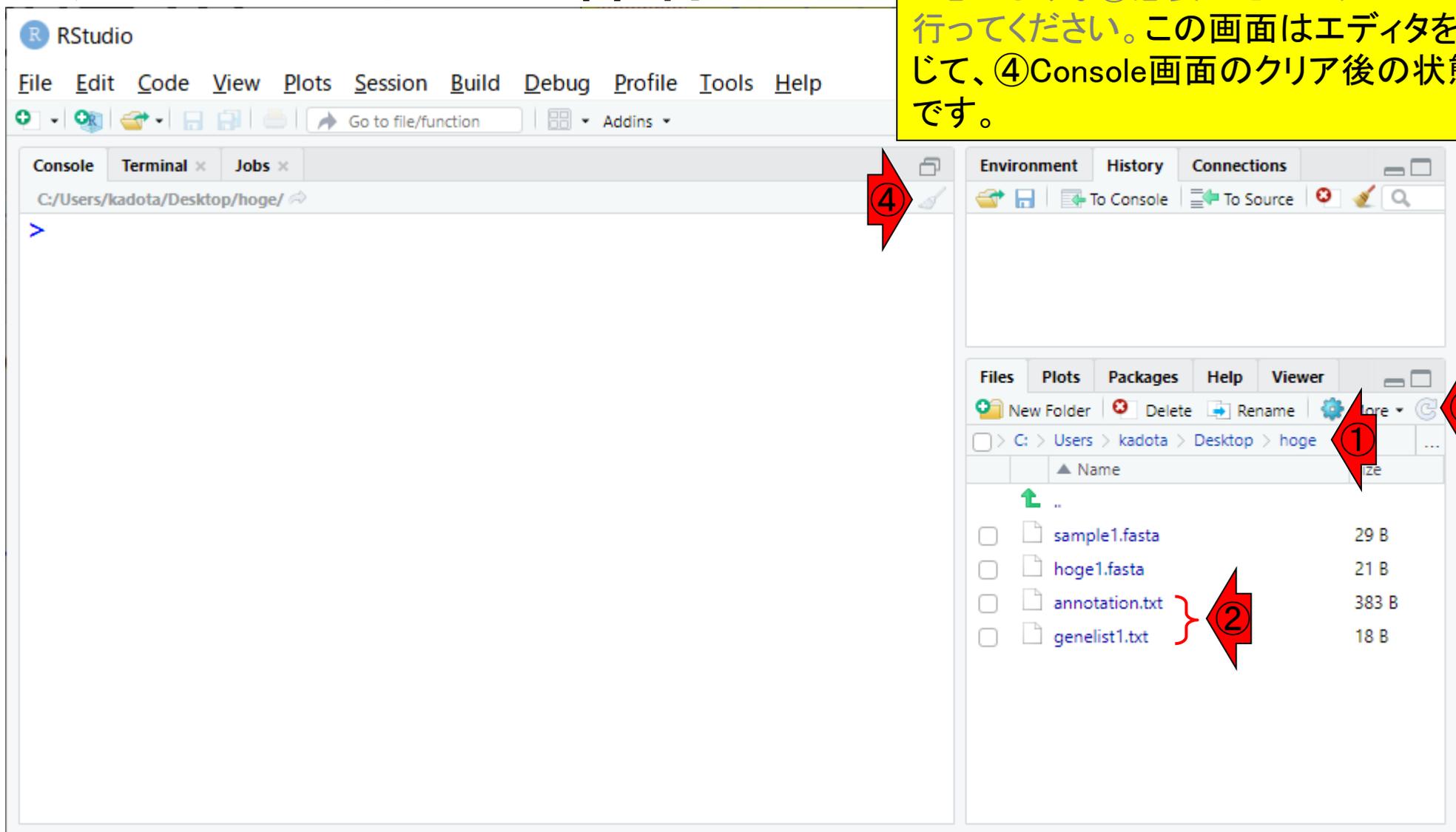
入力ファイルの保存2

正しく保存できれば、RStudio上の① hogeフォルダは、②のように見えていると思います。③必要に応じてリロードも行ってください。



入力ファイルの保存3

正しく保存できれば、RStudio上の① hogeフォルダは、②のように見えていると思います。③必要に応じてリロードも行ってください。この画面はエディタを閉じて、④Console画面のクリア後の状態です。



入力ファイルの保存4

アノテーションファイル(annotation.txt)中の第1列目に対して、リストファイル(genelist1.txt)中の文字列と一致する行を抜き出して、hoge1.txtというファイル名で出力したい、という目的を達成する上で必要な①2つの入力ファイルが作業ディレクトリ上に存在するのでready-to-analyze。

入力:アノテーションファイル(annotation.txt)

	A	B	C	D
1	gene name	accession	description	subcellular_location
2	gene1	hoge01	plasma_mem	nuclear
3	gene2	hoge02	hohinu	membrane
4	gene3	hoge03	agribio	endoplasmic
5	gene4	hoge04	genesis	endoplasmic
6	gene5	hoge05	kamo	membrane
7	gene6	hoge06	netteba	humei
8	gene7	hoge07	tebasaki	nuclear
9	gene8	hoge08	biiru	nuclear
10	gene9	hoge09	nihonshu	nuclear
11	gene10	hoge10	agene1	membrane
12	gene11	hoge11	iyaaaa	endoplasmic

入力:リストファイル(genelist1.txt)

	A
1	gene1
2	gene7
3	gene9

出力:hoge1.txt

	A	B	C	D
1	gene name	accession	description	subcellular_location
2	gene1	hoge01	plasma_mem	nuclear
3	gene7	hoge07	tebasaki	nuclear
4	gene9	hoge09	nihonshu	nuclear

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

コピー実行1

(Rで)塩基配列解析

保護されていない通信 | iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_grep_kiso

ゲスト

1. 目的のタブ区切りテキストファイル([annotation.txt](#))中の第1列目をキーとして、リストファイル([genelist1.txt](#))中のものが含まれる行全体を出力したい場合：

```

in_f1 <- "annotation.txt"      #入力ファイル名を指定してin_f1に格納(アノテーションファイル)
in_f2 <- "genelist1.txt"      #入力ファイル名を指定してin_f2に格納(リストファイル)
out_f <- "hoge1.txt"          #出力ファイル名を指定してout_fに格納
param <- 1                    #アノテーションファイル中の検索したい列番号を指定

#入力ファイルの読み込み
data <- read.table(in_f1, header=TRUE, sep="\t", quote="")#in_f1で指定したファイルの読み込み
keywords <- readLines(in_f2)  #in_f2で指定したファイルの読み込み
dim(data)                     #オブジェクトdataの行数と列数を表示

#本番
obj <- is.element(as.character(data[,param]), keywords)#条件を満たすかどうかを判定した結果をobjに格納
out <- data[obj,]              #objがTRUEとなる行のみ抽出した結果をoutに格納
dim(out)                       #オブジェクトoutの行数と列数を表示

#ファイルに保存
write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)#outの中身を指定したファイル名で保

```

[トップページへ](#)

コピペ実行2

赤枠内をコピー実行。①RStudioのConsole画面上で、②Paste直後の状態。③リターンキーを押す。

The screenshot shows the RStudio interface with the following elements:

- Console:** Contains R code for reading a file and filtering rows. A red box highlights the code. A context menu is open over the code with 'Paste' selected. A red arrow labeled '2' points to the 'Paste' option.
- Environment/History/Connections:** Panels on the right side of the interface.
- Files:** A file explorer showing a directory structure with files like 'sample1.fasta', 'hoge1.fasta', 'annotation', and 'genelist1.txt'. A red arrow labeled '3' points to the Enter key on a keyboard image overlaid on the right side of the screenshot.
- Keyboard:** An image of a keyboard with the Enter key highlighted by a red arrow labeled '3'.

```
quote = ... #in_f2で指定
keywords <- readLines(in_f2) #in_f2で指定
#条件を満たすかどうか
dim(data) #オブジェクト
dataの行数と列数を表示

#本番
obj <- is.element(as.character(ram]),
keywords)#条件を満たすかどうかの結果をobjに格納
out <- data[obj,] #objがTRUEとなる行のみ抽出した結果をoutに格納
dim(out) #オブジェクト
outの行数と列数を表示

#ファイルに保存
write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)#outの中身を指定したファイル名で保存
```

コピー実行3

赤枠内をコピー実行。①RStudioのConsole画面上で、②Paste直後の状態。③リターンキーを押す。こんな感じになって、④指定した出力ファイル(hoge1.txt)が作成できれば基本成功です。④をクリック。

The screenshot shows the RStudio interface with the following components:

- Console:** Contains R code and its output. The code includes comments in Japanese explaining each step. The output shows the dimensions of the data object and the resulting subset.
- Environment:** Shows the current environment with variables like 'out' and 'obj'.
- Files:** A file explorer showing the directory structure and the newly created file 'hoge1.txt'.

```
> dim(data) #オブジェクトdataの行数と列数を表示
[1] 11 4
>
> #本番
> obj <- is.element(as.character(data[,param]), keywords) #条件を満たすかどうかを判定した結果をobjに格納
> out <- data[obj,] #objがTRUEとなる行のみ抽出した結果をoutに格納
> dim(out) #オブジェクトoutの行数と列数を表示
[1] 3 4
>
> #ファイルに保存
> write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F) #outの中身を指定したファイル名で保存
>
>
```

Name	Size
..	
sample1.fasta	29 B
hoge1.fasta	21 B
annotation.txt	383 B
genelist1.txt	18 B
hoge1.txt	148 B

出力ファイルの確認1

赤枠内をコピー実行。①RStudioのConsole画面上で、②Paste直後の状態。③リターンキーを押す。こんな感じになって、④指定した出力ファイル(hoge1.txt)が作成されれば基本成功です。④をクリック。⑤のような感じで出力ファイルの中身が見られます。

```
RStudio  
File Edit Code View Plots Session Build Debug Profile Tools Help  
Go to file/function Addins  
hoge1.txt ×  
1 gene_name accession description subcellular_location  
2 gene1 hoge01 plasma_mem nuclear  
3 gene7 hoge07 tebasaki nuclear  
4 gene9 hoge09 nihonshu nuclear  
5
```

```
Console Terminal × Jobs ×  
C:/Users/kadota/Desktop/hoge/ ↗  
> dim(out) #オブジェクトoutの行数と列数を表示  
[1] 3 4  
>  
> #ファイルに保存  
> write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)#outの中身を指定したファイル名で保存  
>  
> |
```

```
out <- data[obj,] #objがTR...  
dim(out) #オブジェクトoutの行..  
#ファイルに保存  
write.table(out, out_f, se...
```

Files	Plots	Packages	Help	Viewer
New Folder	Delete	Rename	More	
C: > Users > kadota > Desktop > hoge				
Name	Size			
..				
sample1.fasta	29 B			
hoge1.fasta	21 B			
annotation.txt	383 B			
genelist1.txt	18 B			
hoge1.txt	148 B			

出力ファイルの確認2

入力: アノテーションファイル(annotation.txt)

	A	B	C	D
1	gene1	hoge01	plasma_mem	nuclear
2	gene2	hoge02	hohinu	membrane
3	gene3	hoge03	agribio	endoplasmic
4	gene4	hoge04	genesis	endoplasmic
5	gene5	hoge05	kamo	membrane
6	gene6	hoge06	netteba	humei
7	gene7	hoge07	tebasaki	nuclear
8	gene8	hoge08	biiru	nuclear
9	gene9	hoge09	nihonshu	nuclear
10	gene10	hoge10	agene1	membrane
11	gene11	hoge11	iyaaaa	endoplasmic

入力: リストファイル(genelist1.txt)

	A
1	gene1
2	gene7
3	gene9

赤枠内をコピー実行。①RStudioのConsole画面上で、②Paste直後の状態。③リターンキーを押す。こんな感じになって、④指定した出力ファイル(hoge1.txt)が作成できれば基本成功です。④をクリック。⑤のような感じで出力ファイルの中身が見られます。最初に見せたものと同じ結果が得られていますね。

出力: hoge1.txt

	A	B	C	D
1	gene1	hoge01	plasma_mem	nuclear
2	gene7	hoge07	tebasaki	nuclear
3	gene9	hoge09	nihonshu	nuclear

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

Winユーザ向け注意点1

先にExcelで開いているものと同じファイルをRエディタで開こうとすると意味不明なエラーが出るという注意点です。

The screenshot shows the RStudio interface. The editor window displays a text file named 'hoge1.txt' containing a table with 5 rows and 4 columns. A red arrow labeled '2' points to the top-left corner of the editor window. The console window shows the following R code and output:

```
> dim(out) #オブジェクトoutの行数と列数を表示
[1] 3 4
> #ファイルに保存
> write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)#outの中身を指定したファイル名で保存
>
> |
```

The file explorer on the right shows the directory 'C:\Users\kadota\Desktop\hoge' containing several files. A red arrow labeled '1' points to the file 'hoge1.txt' in the list.

gene	accession	description	subcellular_location
gene1	hoge01	plasma_mem	nuclear
gene7	hoge07	tebasaki	nuclear
gene9	hoge09	nihonshu	nuclear

Winユーザー向け注意点2

先にExcelで開いているものと同じファイルをRエディタで開こうとすると意味不明なエラーが出るという注意点です。先に①hoge1.txtを、②Rエディタ上で開いた状態では、問題なくExcelで開くことができます。

The screenshot shows the RStudio environment. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. Below the menu is a toolbar with icons for file operations and a search bar. The main editor window displays a text file named 'hoge1.txt' with the following content:

```
1 gene_name accession description subcellular_location
2 gene1 hoge01 plasma_mem nuclear
3 gene7 hoge07 tebasaki nuclear
4 gene9 hoge09 nihonshu nuclear
5
```

The console window at the bottom shows the following R commands and their output:

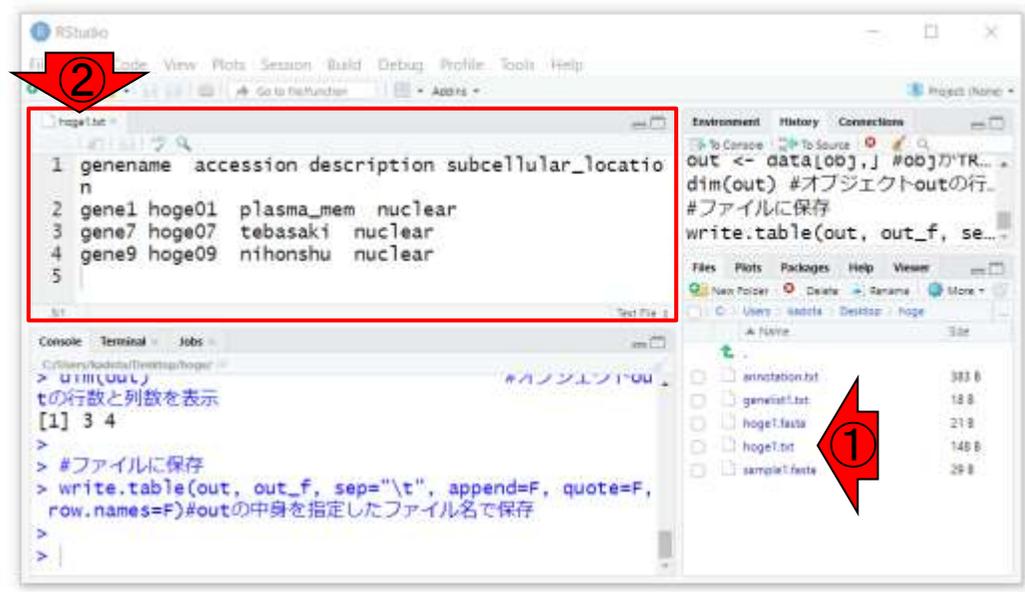
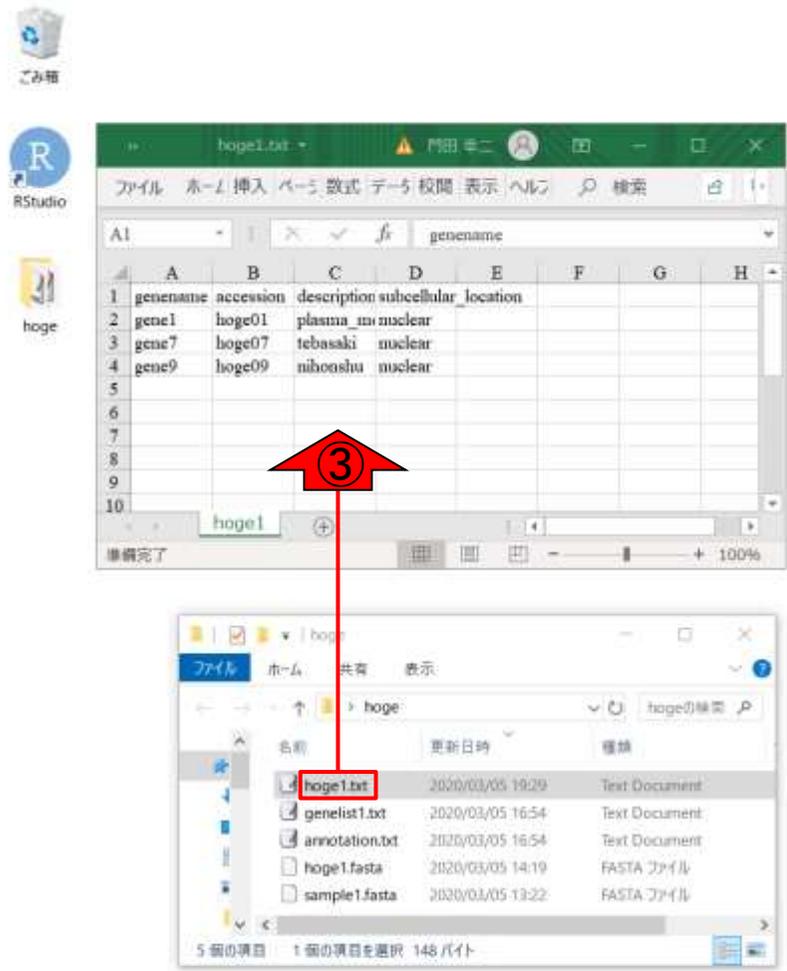
```
> dim(out) #オブジェクトoutの行数と列数を表示
[1] 3 4
> #ファイルに保存
> write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)#outの中身を指定したファイル名で保存
>
>
```

On the right side, the 'Files' pane shows a directory listing for 'C:\Users\kadota\Desktop\hoge'. The files listed are:

Name	Size
..	
sample1.fasta	29 B
hoge1.fasta	21 B
annotation.txt	383 B
genelist1.txt	18 B
hoge1.txt	148 B

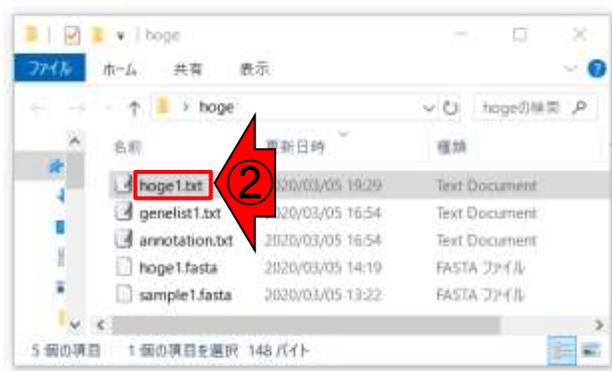
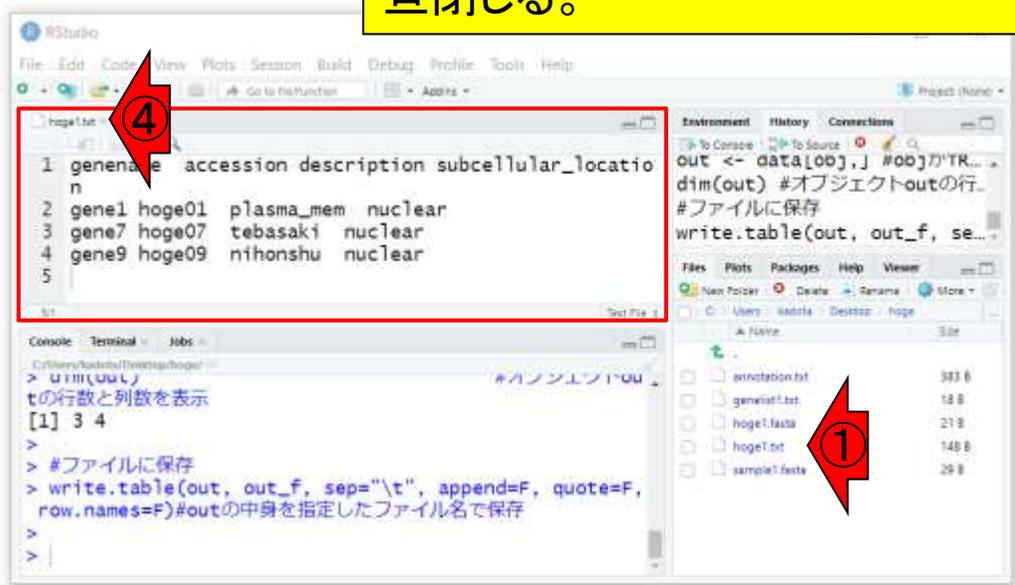
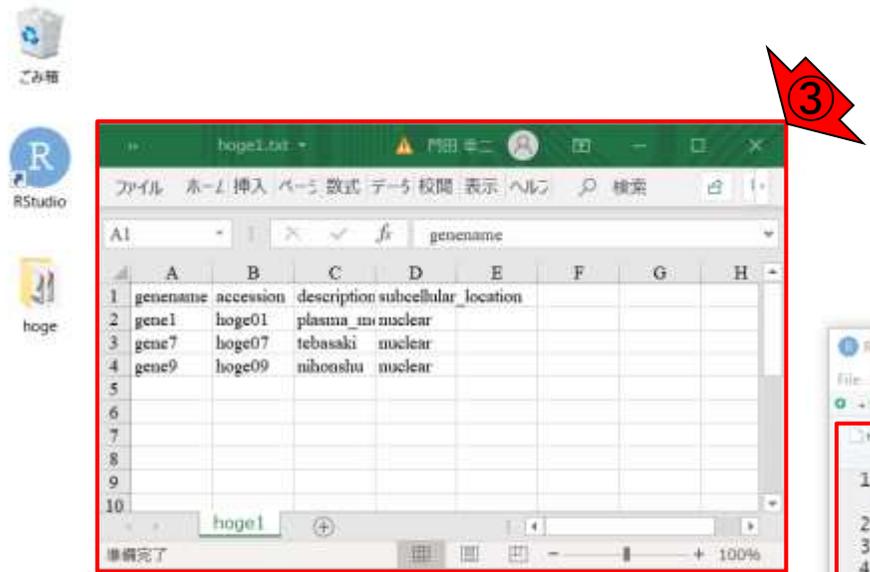
Winユーザー向け注意点3

先にExcelで開いているものと同じファイルをRエディタで開こうとすると意味不明なエラーが出るという注意点です。先に①hoge1.txtを、②Rエディタ上で開いた状態では、③問題なくExcelで開くことができます。つまりこういうこと。



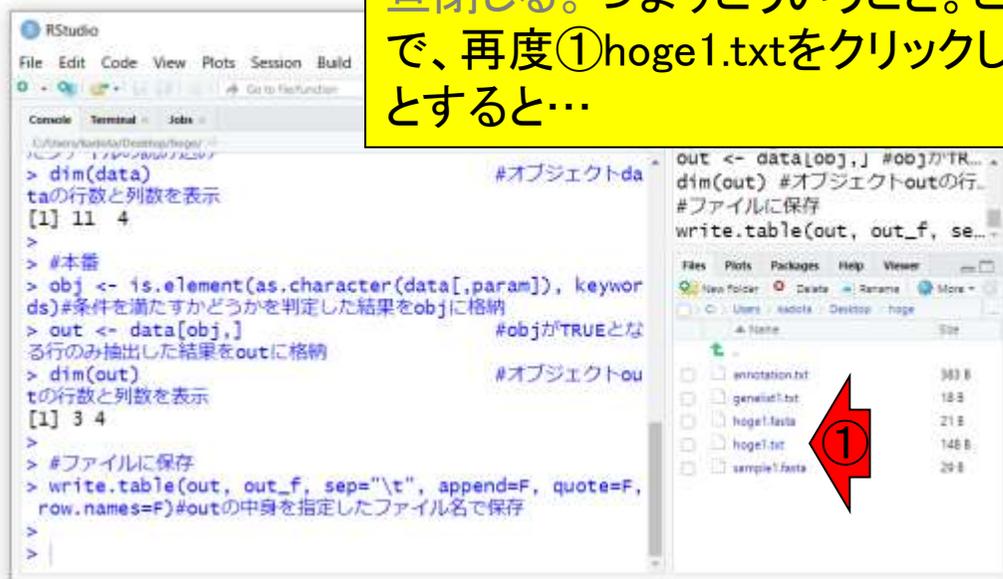
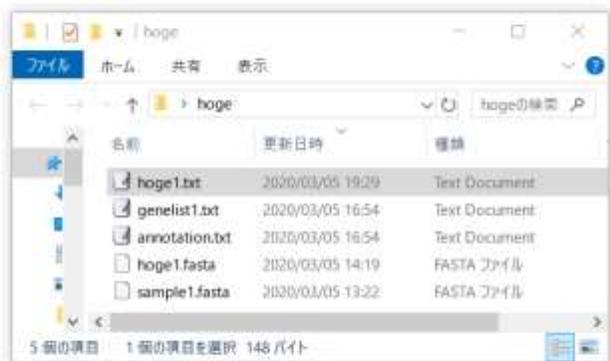
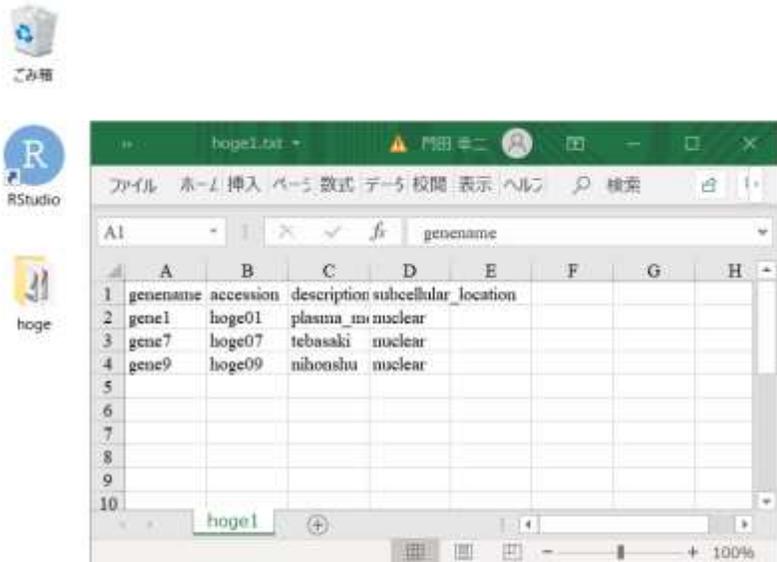
Winユーザ向け注意点4

先にExcelで開いているものと同じファイルをRエディタで開こうとすると意味不明なエラーが出るという注意点です。先に①hoge1.txtを、②Rエディタ上で開いた状態では、③問題なくExcelで開くことができます。つまりこういうこと。次に、① or ②を③Excelで開いている状態で、④Rエディタ上で開いているhoge1.txtを一旦閉じる。



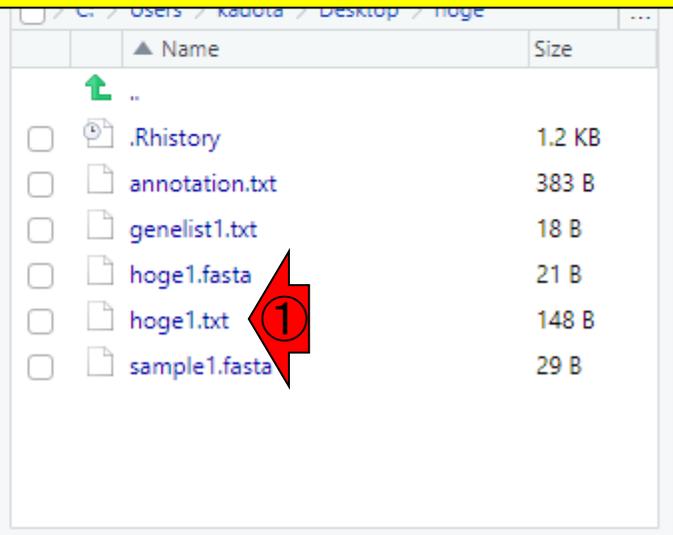
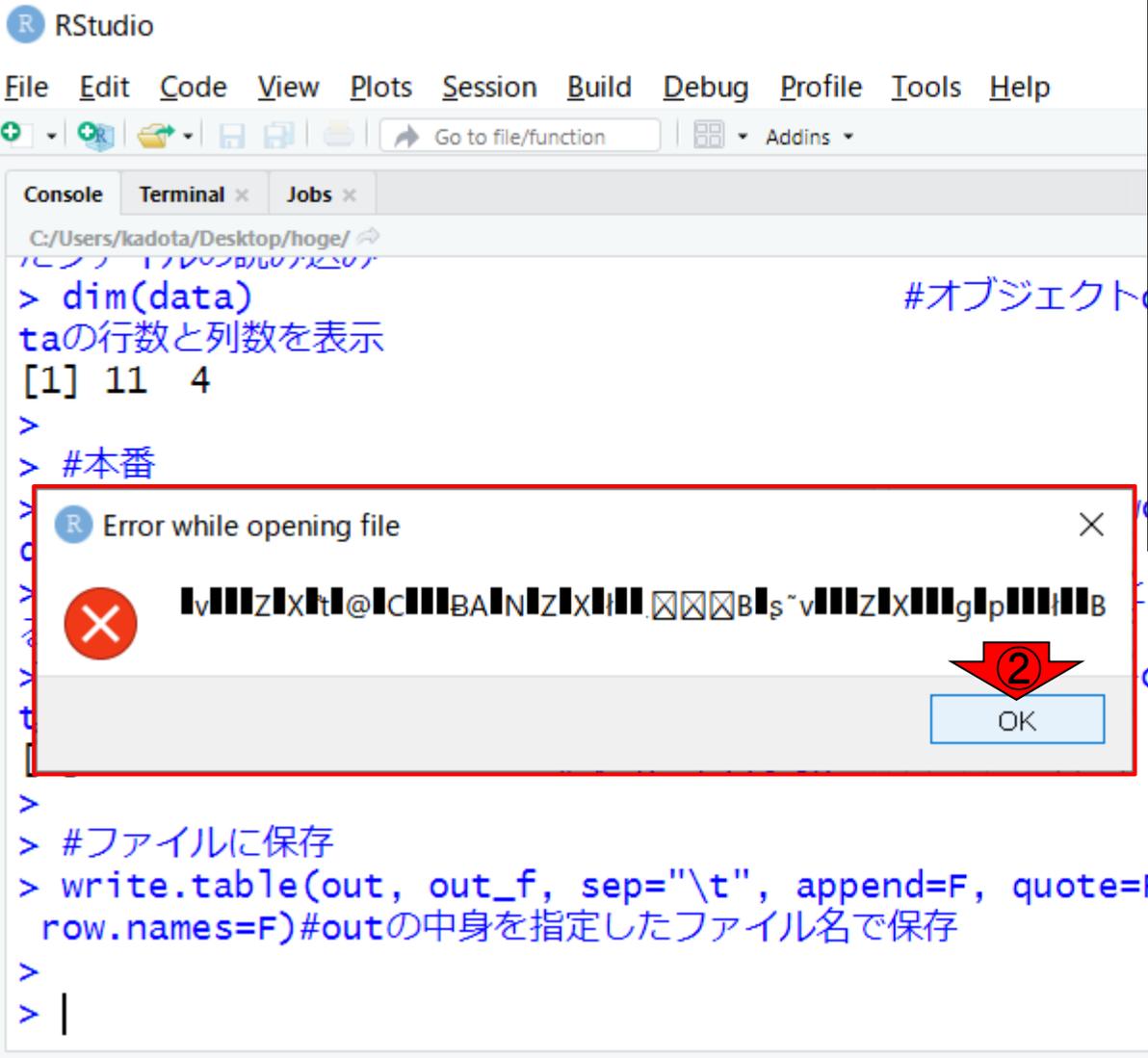
Winユーザ向け注意点5

先にExcelで開いているものと同じファイルをRエディタで開こうとすると意味不明なエラーが出るという注意点です。先に①hoge1.txtを、②Rエディタ上で開いた状態では、③問題なくExcelで開くことができます。つまりこういうこと。次に、① or ②を③Excelで開いている状態で、④Rエディタ上で開いているhoge1.txtを一旦閉じる。つまりこういうこと。この状態で、再度①hoge1.txtをクリックして開こうとすると…



Winユーザー向け注意点6

先にExcelで開いているものと同じファイルをRエディタで開こうとすると意味不明なエラーが出るという注意点です。先に①hoge1.txtを、②Rエディタ上で開いた状態では、③問題なくExcelで開くことができます。つまりこういうこと。次に、①or ②を③Excelで開いている状態で、④Rエディタ上で開いているhoge1.txtを一旦閉じる。つまりこういうこと。この状態で、再度①hoge1.txtをクリックして開こうとすると、こんな感じの意味不明なエラーメッセージが出ますのでお気をつけください。この状態で②OKを押すと…



Winユーザ向け注意点7

こんな風に意味不明な状態になりますのでお気をつけください。こうなったら、RStudioを再起動するのが一番手っ取り早いです。出力予定ファイルと同一のものをExcelで開いているような場合にも同様の不具合が起こりますのでご注意ください。

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. Below the menu is a toolbar with icons for file operations and a search bar. The main editor area is empty. At the bottom, there are three panels: Console, Terminal, and Jobs. The Console panel shows the following R code and output:

```
C:/Users/kadota/Desktop/hoge/
> dim(out) #オブジェクトoutの行数と列数を表示
[1] 3 4
>
> #ファイルに保存
> write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F) #outの中身を指定したファイル名で保存
>
> |
```

The Jobs panel shows a file explorer window for the directory C:\Users\kadota\Desktop\hoge. The file listing is as follows:

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge1.fasta	21 B
hoge1.txt	148 B
sample1.fasta	29 B

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

応用1

①例題1は、②の中身をもつ、③リストファイル(hoge1.txt)を読み込んで、④1列目で一致する行を抽出するスクリプトでした。

(Rで)塩基配列解析 × +

← ① iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_grep_kiso ゲスト

1. 目的のタブ区切りテキストファイル(annotation.txt)中の第1列目をキーとして、リストファイル(genelist1.txt)中のものが含まれる行全体を出力したい場合:

```
in_f1 <- "annotation.txt"
in_f2 <- "genelist1.txt"
out_f <- "hoge1.txt"
param <- 1
```

gene1 ↓

gene7 ↓

gene9 ↓

↓

```
#入力ファイルの読み込み
data <- read.table(in_f1, header=T, as.is=T, quote="")
keywords <- readLines(in_f2)
dim(data)
```

```
#本番
obj <- is.element(as.character(data[,param]), keywords)
```

```
out <- data[obj,]
dim(out)
```

```
#ファイルに書き出す
write.table(out, out_f, row.names=F)
```

	A	B	C	D
1	gene name	accession	description	subcellular_location
2	gene1	hoge01	plasma_mem	nuclear
3	gene2	hoge02	hohinu	membrane
4	gene3	hoge03	agribio	endoplasmic
5	gene4	hoge04	genesis	endoplasmic
6	gene5	hoge05	kamo	membrane
7	gene6	hoge06	netteba	humei
8	gene7	hoge07	tebasaki	nuclear
9	gene8	hoge08	biiru	nuclear
10	gene9	hoge09	nihonshu	nuclear
11	gene10	hoge10	agene1	membrane
12	gene11	hoge11	iyaaaa	endoplasmic

② ファイル名を指定してin_f1に格納(アノテーションファイル)
 ファイル名を指定してin_f2に格納(リストファイル)
 ファイル名を指定してout_fに格納
 テーションファイル中の検索したい列番号を指定

③ read.tableで指定したファイルの読み込み
 readLinesで指定したファイルの読み込み
 dimでオブジェクトdataの行数と列数を表示

④ is.elementで条件を満たすかどうかを判定した結果をobjに格納
 data[obj,]で抽出した結果をoutに格納
 dimでoutの行数と列数を表示

write.tableでoutの中身を指定したファイル名で保存
[トップページ](#)

応用2

①例題1は、②の中身をもつ、③リストファイル(hoge1.txt)を読み込んで、④1列目で一致する行を抽出するスクリプトでした。応用として、⑤新たに作成したリストファイル(list.txt)を読み込んで、⑥4列目で一致する行の抽出を行います。

(Rで)塩基配列解析

保護されていない通信 | iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_g

1. 目的のタブ区切りテキストファイル(annotation.txt)中の第1列目をキーとして、リストファイル(genelist1.txt)中のものが含まれる行全体を出力したい場合:

```
in_f1 <- "annotation.txt"
in_f2 <- "genelist1.txt"
out_f <- "hoge1.txt"
param <- 1
```

#入力ファイルの読み込み

```
data <- read.table(in_f1, header=T, as.is=T, quote="")
keywords <- readLines(in_f2)
dim(data)
```

#本番

```
obj <- is.element(as.character(data[,param]), keywords)
out <- data[obj,]
dim(out)
```

#ファイルに書き出す

```
write.table(out, out_f, as.is=T, quote="")
```

gene1
gene7
gene9

nuclear
membrane

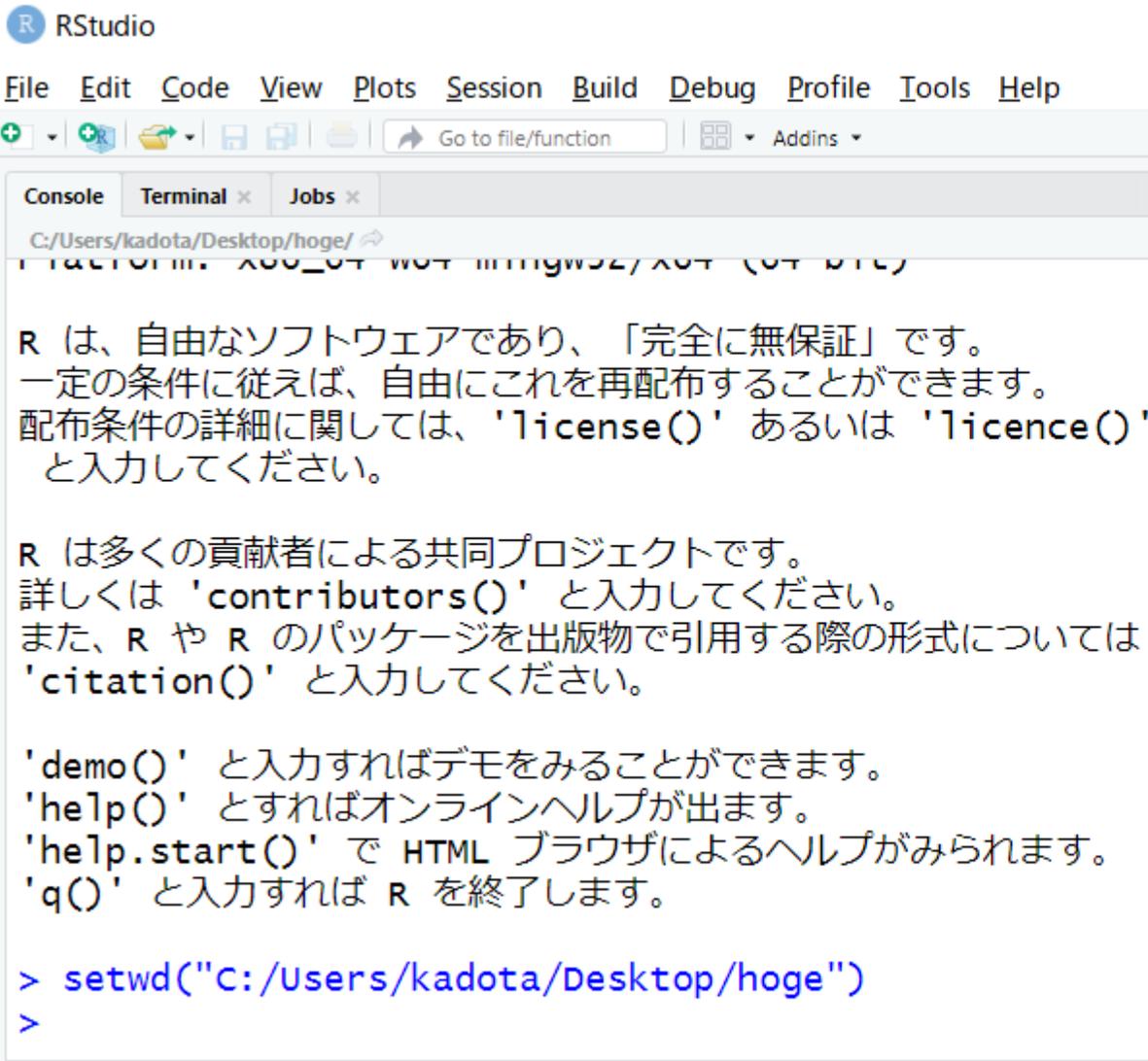
	A	B	C	D
1	gene name	accession	description	subcellular_location
2	gene1	hoge01	plasma_mem	nuclear
3	gene2	hoge02	hohinu	membrane
4	gene3	hoge03	agribio	endoplasmic
5	gene4	hoge04	genesis	endoplasmic
6	gene5	hoge05	kamo	membrane
7	gene6	hoge06	netteba	humei
8	gene7	hoge07	tebasaki	nuclear
9	gene8	hoge08	biiru	nuclear
10	gene9	hoge09	nihonshu	nuclear
11	gene10	hoge10	agene1	membrane
12	gene11	hoge11	iyaaaa	endoplasmic

	A	B	C	D
1	gene name	accession	description	subcellular_location
2	gene1	hoge01	plasma_mem	nuclear
3	gene2	hoge02	hohinu	membrane
4	gene3	hoge03	agribio	endoplasmic
5	gene4	hoge04	genesis	endoplasmic
6	gene5	hoge05	kamo	membrane
7	gene6	hoge06	netteba	humei
8	gene7	hoge07	tebasaki	nuclear
9	gene8	hoge08	biiru	nuclear
10	gene9	hoge09	nihonshu	nuclear
11	gene10	hoge10	agene1	membrane
12	gene11	hoge11	iyaaaa	endoplasmic

row name

応用3

①例題1は、②の中身をもつ、③リストファイル(hoge1.txt)を読み込んで、④1列目で一致する行を抽出するスクリプトでした。応用として、⑤新たに作成したりストファイル(list.txt)を読み込んで、⑥4列目で一致する行の抽出を行います。これはRStudioを再起動して、⑦作業ディレクトリの変更まで終了した状態。

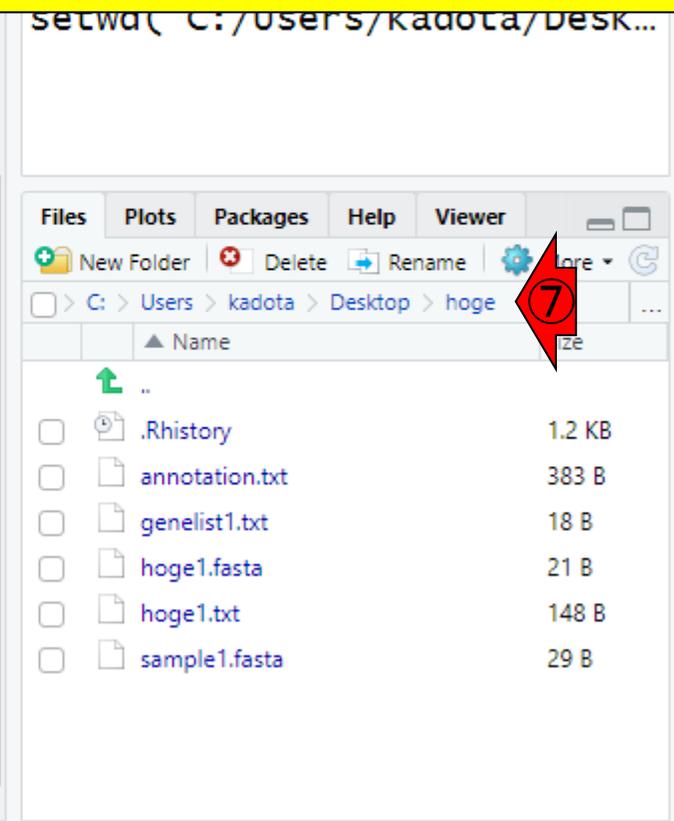


R は、自由なソフトウェアであり、「完全に無保証」です。一定の条件に従えば、自由にこれを再配布することができます。配布条件の詳細に関しては、'license()' あるいは 'licence()' と入力してください。

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ついては '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す。
'q()' と入力すれば R を終了します。

```
> setwd("C:/Users/kadota/Desktop/hoge")  
>
```



Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge1.fasta	21 B
hoge1.txt	148 B
sample1.fasta	29 B

応用4

まずは①リストファイル(list.txt)を作成
すべく、②File、③New File、④Text File。

The screenshot shows the R Studio interface. The 'File' menu is open, and the path 'File > New File > Text File' is highlighted with red arrows and circled numbers 2, 3, and 4. The 'Text File' option is also circled with a red arrow and the number 4. In the background, a file explorer window is open, showing a list of files in a directory. The files listed are: .., .Rhistory (1.2 KB), annotation.txt (383 B), genelist1.txt (18 B), hoge1.fasta (21 B), hoge1.txt (148 B), and sample1.fasta (29 B). A red box highlights the text 'nuclear' and 'membrane' in the file explorer, with a red arrow and the number 1 pointing to it.

応用5

まずは①リストファイル(list.txt)を作成
すべく、②File、③New File、④Test File。
こんな感じになります。

The screenshot shows the RStudio environment. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The toolbar contains various icons for file operations and navigation. The main editor window shows a single line of code: `1`. The Environment pane on the right shows the current working directory: `setwd("C:/Users/kadota/Desktop/hoge")`. Below this, a list of files is displayed:

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge1.fasta	21 B
hoge1.txt	148 B
sample1.fasta	29 B

The Files pane on the right shows the current directory: `C:/Users/kadota/Desktop/hoge`. A red box highlights the text `nuclear` and `membrane` in the Environment pane, with a red arrow pointing to the first character of `nuclear`. The Console window at the bottom shows the following R commands and their output:

```
> setwd("C:/Users/kadota/Desktop/hoge")
> 
```

The console output includes: `C:/Users/kadota/Desktop/hoge/`, `CTCacton() 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す。`, and `'q()' と入力すれば R を終了します。`

応用6

まずは①リストファイル(list.txt)を作成すべく、②File、③New File、④Test File。こんな感じになります。①と同じように抽出したいキーワードを入力。この際、②の部分で最後にリターンキーを「押すか押さないか」でコピー実行時に「警告が出ないか出るか」の違いがでますがそれも含めて楽しんでください。

RStudio interface showing a text editor with the following content:

```
1 nuclear
2 membrane
3
```

The console shows the following text:

```
C:\Users\kadota\Desktop\hoge/
C:\Users\kadota\Desktop\hoge/
> 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す。
> q() と入力すれば R を終了します。
> setwd("C:/Users/kadota/Desktop/hoge")
>
```

File Explorer window showing the contents of the folder 'C:\Users\kadota\Desktop\hoge':

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge1.fasta	21 B
hoge1.txt	148 B
sample1.fasta	29 B

The text 'nuclear' and 'membrane' is highlighted in a red box with blue arrows pointing to the end of each line. A red arrow labeled '1' points to the first line.

応用7

まずは①リストファイル(list.txt)を作成すべく、②File、③New File、④Test File。こんな感じになります。①と同じように抽出したいキーワードを入力。この際、②の部分で最後にリターンキーを「押すか押さないか」でコピー実行時に「警告が出ないか出るか」の違いがでますがそれも含めて楽しんでください。③を押して保存。

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

```
1 number  
2 membrane  
3
```

3:1 Text File

Console Terminal Jobs

```
C:/Users/kadota/Desktop/hoge/ ↗  
> 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す。  
> q() と入力すれば R を終了します。  
  
> setwd("C:/Users/kadota/Desktop/hoge")  
> |
```

Files Plots Packages Help Viewer

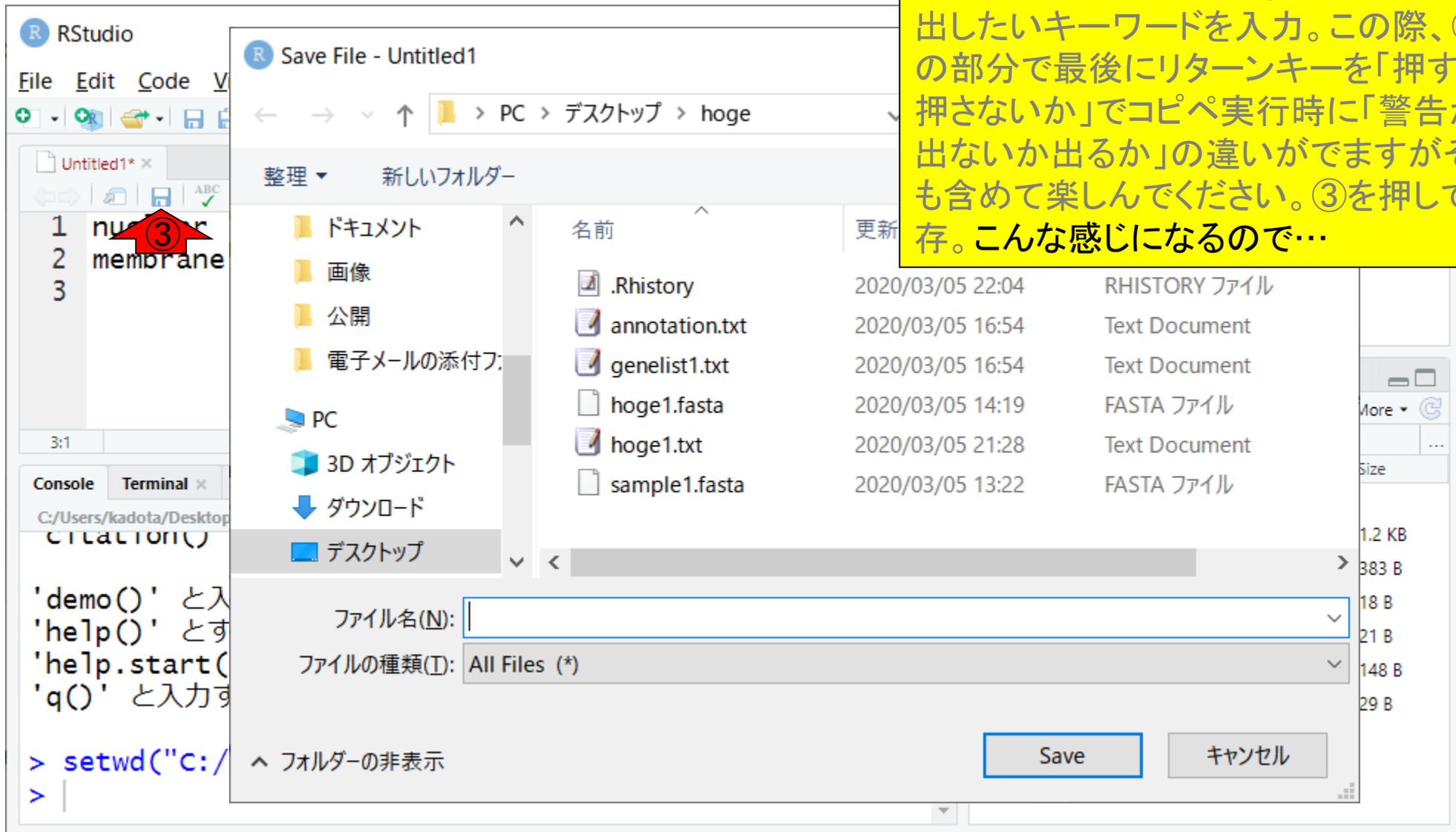
New Folder Delete Rename More

C: > Users > kadota > Desktop > hoge

	Name	Size
	..	
	.Rhistory	1.2 KB
	annotation.txt	383 B
	genelist1.txt	18 B
	hoge1.fasta	21 B
	hoge1.txt	148 B
	sample1.fasta	29 B

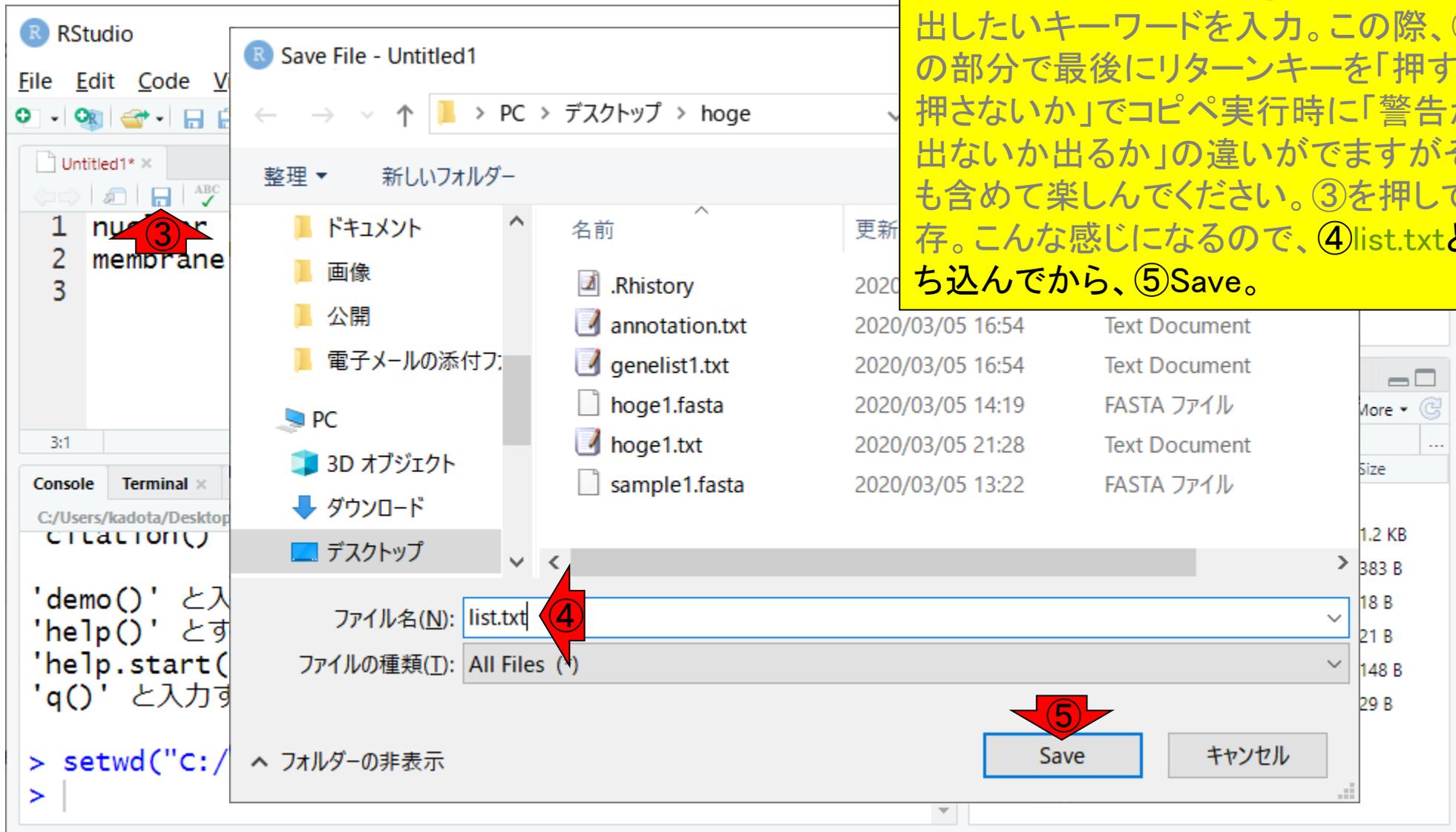
応用8

まずは①リストファイル(list.txt)を作成すべく、②File、③New File、④Test File。こんな感じになります。①と同じように抽出したいキーワードを入力。この際、②の部分で最後にリターンキーを「押すか押さないか」でコピー実行時に「警告が出ないか出るか」の違いがでますがそれも含めて楽しんでください。③を押して保存。こんな感じになるので…



応用9

まずは①リストファイル(list.txt)を作成すべく、②File、③New File、④Test File。こんな感じになります。①と同じように抽出したいキーワードを入力。この際、②の部分で最後にリターンキーを「押すか押さないか」でコピペ実行時に「警告が出ないか出るか」の違いがでますがそれも含めて楽しんでください。③を押して保存。こんな感じになるので、④list.txtと打ち込んでから、⑤Save。



応用10

まずは①リストファイル(list.txt)を作成すべく、②File、③New File、④Test File。こんな感じになります。①と同じように抽出したいキーワードを入力。この際、②の部分で最後にリターンキーを「押すか押さないか」でコピー実行時に「警告が出ないか出るか」の違いがでますがそれも含めて楽しんでください。③を押して保存。こんな感じになるので、④list.txtと打ち込んでから、⑤Save。私の場合は、⑥リロードすると、⑦list.txtが見えました。

RStudio interface showing a text editor with the following content:

```
1 nuclear
2 membrane
3 |
```

The console shows the following output:

```
C:\Users\kadota\Desktop\hoge/
C:\Users\kadota\Desktop\hoge/
'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す。
'q()' と入力すれば R を終了します。
> setwd("C:/Users/kadota/Desktop/hoge")
> |
```

Windows File Explorer showing the contents of the 'hoge' directory:

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge1.fasta	21 B
hoge1.txt	148 B
list.txt	19 B
sample1.fasta	29 B

次は、①例題1のスクリプト(or コード)を
テンプレートとして利用したいので…

応用11

1. 目的のタブ区切りテキストファイル([annotation.txt](#))中の第1列目をキーとして、リストファイル([genelist1.txt](#))中のものが含まれる行全体を出力したい場合：

```
in_f1 <- "annotation.txt" #入力ファイル名を指定してin_f1に格納(アノテーションファイル)
in_f2 <- "genelist1.txt"  #入力ファイル名を指定してin_f2に格納(リストファイル)
out_f  <- "hoge1.txt"     #出力ファイル名を指定してout_fに格納
param <- 1                #アノテーションファイル中の検索したい列番号を指定

#入力ファイルの読み込み
data <- read.table(in_f1, header=TRUE, sep="\t", quote="")#in_f1で指定したファイルの読み込み
keywords <- readLines(in_f2) #in_f2で指定したファイルの読み込み
dim(data) #オブジェクトdataの行数と列数を表示

#本番
obj <- is.element(as.character(data[,param]), keywords)#条件を満たすかどうかを判定した結果をobjに格納
out <- data[obj,] #objがTRUEとなる行のみ抽出した結果をoutに格納
dim(out) #オブジェクトoutの行数と列数を表示

#ファイルに保存
write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)#outの中身を指定したファイル名で保
```

[トップページへ](#)

応用12

次は、①例題1のスクリプト(or コード)を
テンプレートとして利用したいので、②赤
枠内のコード全体をコピーしておく。

(Rで)塩基配列解析

← ① 保護されていない通信 | iu.a.u-tokyo.ac.jp/~kadota/r_seq.html#intro_general_grep_kiso

1. 目的のタブ区切りテキストファイル([annotation.txt](#))中の第1列目をキーとして、リストファイル([genelist1.txt](#))中のものが含まれる行全体を出力したい場合：

```
in f1 <- "annotation.txt" #入力ファイル名を指定してin f1に格納(アノテーションファイル)
in f2 <- "genelist1.txt" #入力ファイル名を指定してin f2に格納(リストファイル)
out f <- "hoge1.txt" #出力ファイル名を指定してout fに格納
param <- 1
#入力ファイルの読み込み
data <- read.table(in f1)
keywords <- readLines(in f2)
dim(data)
#本番
obj <- is.element(as.character(data[,param]), keywords)#条件を満たすかどうかを判定した結果をobjに格納
out <- data[obj,] #objがTRUEとなる行のみ抽出した結果をoutに格納
dim(out) #オブジェクトoutの行数と列数を表示
#ファイルに保存
write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)#outの中身を指定したファイル名で保
```

②

コピー(C)	Ctrl+C
Google で「in_f1 <- "annotation.txt" #入力ファイル名を...」を検索(S)	
印刷(P)...	Ctrl+P
検証(I)	Ctrl+Shift+I

[トップページへ](#)

応用13

次は、①例題1のスクリプト(or コード)をテンプレートとして利用したいので、②赤枠内のコード全体をコピーしておく。③ File、④New File、⑤R Scriptとして、「新規のRスクリプト」を開く。

The screenshot shows the R Studio interface. The 'File' menu is open, and the 'New File' option is selected. The 'R Script' option is highlighted in the submenu. The code editor shows the following code:

```
setwd("C:/Users/kadota/Desktop/hoge")
```

A red box highlights the text `/hoge')` at the end of the line.

応用14

次は、①例題1のスクリプト(or コード)をテンプレートとして利用したいので、②赤枠内のコード全体をコピーしておく。③File、④New File、⑤R Scriptとして、「新規のRスクリプト」を開く。こんな感じになって、⑥Untitled1という中身が空っぽの新規スクリプトファイルが開きます。(Wordで「名称未設定」の新規文書を開くようなものですね。)

RStudio interface showing a new R script file named 'Untitled1' being created. The console shows the following text:

```
C:\Users\kadota\Desktop\hoge/
C:\Users\kadota\Desktop\hoge/

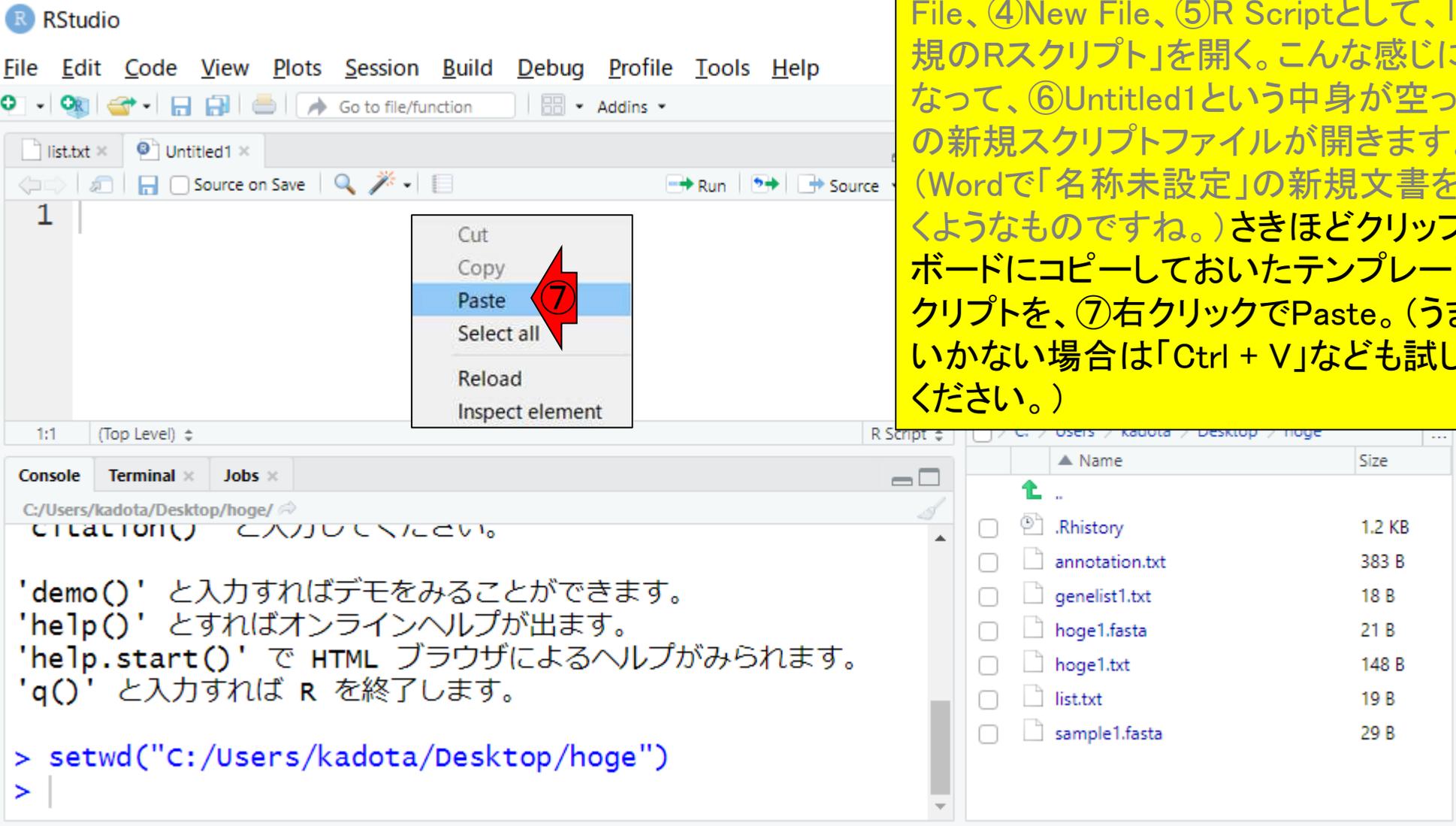
'demo()' と入力すればデモをみることができます。
'help()' とすればオンラインヘルプが出ます。
'help.start()' で HTML ブラウザによるヘルプがみられます。
'q()' と入力すれば R を終了します。

> setwd("C:/Users/kadota/Desktop/hoge")
> |
```

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge1.fasta	21 B
hoge1.txt	148 B
list.txt	19 B
sample1.fasta	29 B

応用15

次は、①例題1のスクリプト(or コード)をテンプレートとして利用したいので、②赤枠内のコード全体をコピーしておく。③File、④New File、⑤R Scriptとして、「新規のRスクリプト」を開く。こんな感じになって、⑥Untitled1という中身が空っぽの新規スクリプトファイルが開きます。(Wordで「名称未設定」の新規文書を開くようなものですね。)さきほどクリップボードにコピーしておいたテンプレートスクリプトを、⑦右クリックでPaste。(うまくいかない場合は「Ctrl + V」なども試してください。)



応用16

The screenshot shows the RStudio interface. The editor window contains the following R code:

```

12 obj <- ts.element(as.character(data[,param]), keyv
13 out <- data[obj,] #objがTRUE
14 dim(out) #オブジェクト
15
16 #ファイルに保存
17 write.table(out, out_f, sep="\t", append=F, quote=
18 '

```

The console window shows the following text:

```

C:\Users\kadota\Desktop\hoge/
C:\Users\kadota\Desktop\hoge/
'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す。
'q()' と入力すれば R を終了します。

> setwd("C:/Users/kadota/Desktop/hoge")
>

```

The file explorer on the right shows the directory structure:

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge1.fasta	21 B
hoge1.txt	148 B
list.txt	19 B
sample1.fasta	29 B

応用17

こんな感じになればOK。今は①エディタ画面と②Console画面の縦幅が非常に狭い。例えば①エディタ画面を広く使いたい場合は、③を押す。

The screenshot shows the RStudio interface with the following elements:

- Menu Bar:** File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help
- Toolbar:** Includes icons for file operations and a search bar.
- Editor:** Contains R code:

```
12 obj <- ts.element(as.character(data[,param]), keyv...
13 out <- data[obj,] #objがTRUE
14 dim(out) #オブジェクト
15
16 #ファイルに保存
17 write.table(out, out_f, sep="\t", append=F, quote=
18 '
```
- Environment/History/Connections:** Shows 'setwd("C:/Users/kadota/Desktop/hoge...")'.
- Files/Plots/Packages/Help/Viewer:** Shows a file explorer view of the 'hoge' directory with files like '.Rhistory', 'annotation.txt', 'genelist1.txt', 'hoge1.fasta', 'hoge1.txt', 'list.txt', and 'sample1.fasta'.
- Console/Terminal/Jobs:** Shows R help text and a command:

```
> setwd("C:/Users/kadota/Desktop/hoge")
```

Annotations:

- ③: Points to the maximize button in the top right of the editor window.
- ①: Points to the editor window.
- ②: Points to the console window.

応用18

こんな感じになればOK。今は①エディタ画面と②Console画面の縦幅が非常に狭い。例えば①エディタ画面を広く使いたい場合は、③を押す。こんな感じになります。もう一度③を押すと元に戻せる。

The screenshot shows the RStudio interface with the following elements:

- Editor:** Contains R code for reading a table and writing it to a file. The code is partially obscured by a red box.
- Console:** Located at the bottom, also highlighted by a red box.
- Environment/History/Connections:** Located on the right side.
- Files:** Located at the bottom right, showing a file explorer view of the 'hoge' directory.

Annotations:

- ①: Points to the maximize button in the editor window's title bar.
- ②: Points to the editor area.
- ③: Points to the maximize button in the console window's title bar.

応用19

元に戻りました。④はエディタ画面の縦幅を最小にする操作であり、⑤Console画面をを最大にする操作と同じです。以降は特に明記しませんが、見やすいように適宜いじってください。

The screenshot shows the RStudio interface with the following components:

- Code Editor:** Contains R code for writing a table to a file. Red arrow ④ points to the editor's vertical scrollbar.
- Environment/History/Connections:** Shows the current working directory as 'C:/Users/kadota/Desktop/hoge'.
- Files Panel:** Shows a file explorer view of the 'hoge' directory with files like '.Rhistory', 'annotation.txt', 'genelist1.txt', 'hoge1.fasta', 'hoge1.txt', 'list.txt', and 'sample1.fasta'. Red arrow ⑤ points to the file explorer's vertical scrollbar.
- Console:** Shows the R prompt and the command `setwd("C:/Users/kadota/Desktop/hoge")`.

応用20

The screenshot shows the RStudio interface. The script editor contains the following R code:

```
1 in_f1 <- "annotation.txt" #入力ファイ
2 in_f2 <- "genelist1.txt" #入力ファ
3 out_f <- "hoge1.txt" #出力ファイ
4 param <- 1 #アノテーシ
5
6 #入力ファイルの読み込み
7 data <- read.table(in_f1, header=TRUE, sep="\t")
8
```

The console shows the following output:

```
> setwd("C:/Users/kadota/Desktop/hoge")
> |
```

The file explorer shows the current directory: C:\Users\kadota\Desktop\hoge. The files listed are:

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge1.fasta	21 B
hoge1.txt	148 B
list.txt	19 B
sample1.fasta	29 B

A red arrow with the number 1 points to the top of the script editor window.

応用21

①エディタ画面の上部に移動したところ。
今は、②の部分のファイル名を、③
list.txtに置き換えたい。

The screenshot shows the RStudio interface. The script editor contains the following code:

```
1 in_f1 <- "annotation.txt" #入力ファイ
2 in_f2 <- "genelist1.txt" #入力ファ
3 out_f <- "hoge1.txt" #出力ファイ
4 param <- 1 #アノテーシ
5
6 #入力ファイルの読み込み
7 data <- read.table(in_f1, header=TRUE, sep="\t")
8
```

The Environment pane on the right shows the following variables:

```
setwd("C:/Users/kadota/...")
nuclear
membrane
```

The Files pane shows the current directory structure:

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge1.fasta	21 B
hoge1.txt	148 B
list.txt	19 B
sample1.fasta	29 B

The Console pane shows the following output:

```
> setwd("C:/Users/kadota/Desktop/hoge")
>
```

応用22

①エディタ画面の上部に移動したところ。今は、②の部分のファイル名を、③ list.txtに置き換えたい。また、④1列目ではなく、⑤4列目で一致する行を抽出したいので…

```
1 in_f1 <- "annotation.txt" #入力ファイ
2 in_f2 <- "genelist1.txt" #入力ファイ
3 out_f <- "hoge1.txt" #出力ファイ
4 param <- 1 #アノテーシ
5
6 #入力ファイルの読み込み
7 data <- read.table(in_f1, header=TRUE, sep="\t")
```

'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す。
'q()' と入力すれば R を終了します。

```
> setwd("C:/Users/kadota/Desktop/hoge")
> |
```

	A	B	C	D
1	gene name	accession	description	subcellular_location
2	gene1	hoge01	plasma_mem	nuclear
3	gene2	hoge02	hohinu	membrane
4	gene3	hoge03	agribio	endoplasmic
5	gene4	hoge04	genesis	endoplasmic
6	gene5	hoge05	kamo	membrane
7	gene6	hoge06	netteba	humei
8	gene7	hoge07	tebasaki	nuclear
9	gene8	hoge08	biiru	nuclear
10	gene9	hoge09	nihonshu	nuclear
11	gene10	hoge10	agene1	membrane
12	gene11	hoge11	iyaaaa	endoplasmic

応用23

①エディタ画面の上部に移動したところ。
今は、②の部分のファイル名を、③
list.txtに置き換えたい。また、④1列目では
なく、⑤4列目で一致する行を抽出したい
ので、⑥こんな感じで書き替えます。

The screenshot shows the RStudio interface. The script editor contains the following code:

```
1 in_f1 <- "annotation.txt" #入力ファイ
2 in_f2 <- "list.txt" #入力ファイル名:
3 out_f <- "hoge1.txt" #出力ファイ
4 param <- 4 #アノテーシ
5
6 #入力ファイルの読み込み
7 data <- read.table(in_f1, header=TRUE, sep="\t")
8
```

Red arrows labeled '6' point to the file names in lines 2 and 4. The console shows the following output:

```
> setwd("C:/Users/kadota/Desktop/hoge")
> |
```

The Environment pane shows the current working directory: `setwd("C:/Users/kadota/Desktop/hoge")`. The Files pane shows the directory structure:

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge1.fasta	21 B
hoge1.txt	148 B
list.txt	19 B
sample1.fasta	29 B

応用24

①エディタ画面の上部に移動したところ。今は、②の部分のファイル名を、③ list.txtに置き換えたい。また、④1列目ではなく、⑤4列目で一致する行を抽出したいので、⑥こんな感じで書き替えます。⑦を押して、(.txtにしても特に問題はありませんが)Rスクリプトであることを明示した拡張子.Rをつけて、⑨Save。

The screenshot shows the RStudio interface with a 'Save File - Untitled1' dialog box open. The dialog is set to save in the 'デスクトップ' (Desktop) folder. The file name is 'hoge.R' and the file type is 'All Files (*)'. A red arrow labeled '8' points to the file name field. The 'Save' button is highlighted with a red arrow labeled '9'. In the background, the RStudio editor shows a script with a red arrow labeled '7' pointing to line 7.

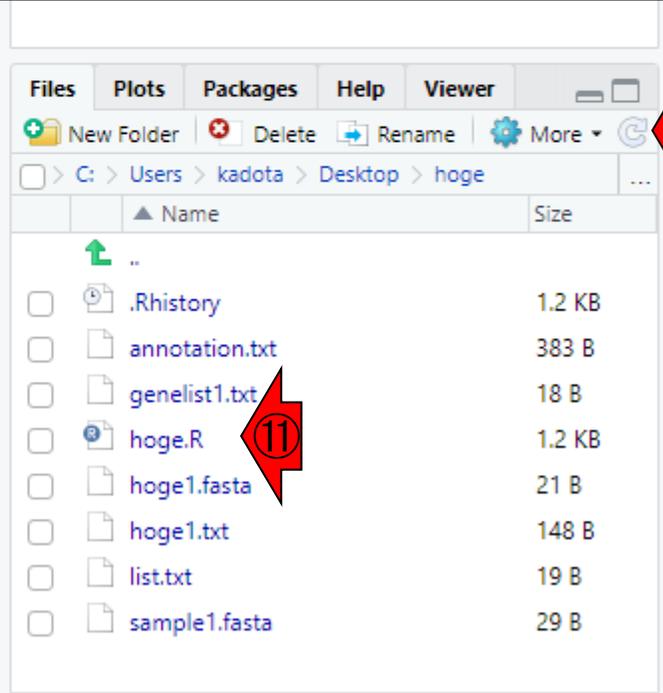
名前	更新日時	種類
.Rhistory	2020/03/05 22:04	RHISTORY ファイル
annotation.txt	2020/03/05 16:54	Text Document
genelist1.txt	2020/03/05 16:54	Text Document
hoge1.fasta	2020/03/05 14:19	FASTA ファイル
hoge1.txt	2020/03/05 21:28	Text Document
list.txt	2020/03/05 22:29	Text Document
sample1.fasta	2020/03/05 13:22	FASTA ファイル

```
1 in_1 <- readLines("list.txt")
2 in_r2 <- readLines("list.txt")
3 out_f <- readLines("list.txt")
4 param <- readLines("list.txt")
5
6 #入力
7 data <- readLines("list.txt")
8
```

応用25

①エディタ画面の上部に移動したところ。今は、②の部分のファイル名を、③ list.txtに置き換えたい。また、④1列目ではなく、⑤4列目で一致する行を抽出したいので、⑥こんな感じで書き替えます。⑦を押して、(.txtにしても特に問題はありませんが)Rスクリプトであることを明示した拡張子.Rをつけて、⑨Save。私の場合は、⑩リロードすると、⑪hoge.Rが見えました。

```
RStudio  
File Edit Code View Plots Session Build Debug Profile Tools Help  
Go to file/function Addins  
list.txt x hoge.R x  
Source on Save Run Source  
1 in_f1 <- "annotation.txt" #入力ファイル  
2 in_f2 <- "list.txt" #入力ファイル  
3 out_f <- "hoge1.txt" #出力ファイル  
4 param <- 4 #アノテーション  
5  
6 #入力ファイルの読み込み  
7 data <- read.table(in_f1, header=TRUE, sep="\t")  
8  
4:11 (Top Level) R Script  
Console Terminal x Jobs x  
C:/Users/kadota/Desktop/hoge/  
'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す。  
'q()' と入力すれば R を終了します。  
> setwd("C:/Users/kadota/Desktop/hoge")  
> |
```



応用26

いよいよ変更後のスクリプトの実行。①
赤枠内のコード全体を選択。

The screenshot shows the RStudio interface with the following components:

- Script Editor:** Contains an R script with the following code:

```
1 in_f1 <- "annotation.txt" #入力ファイル名
2 in_f2 <- "list.txt" #入力ファイル名
3 out_f <- "hoge1.txt" #出力ファイル名
4 param <- 4 #アノテーション
5
6 #入力ファイルの読み込み
7 data <- read.table(in_f1, header=TRUE, sep="\t")
```

Lines 1 through 7 are highlighted with a red rectangular box. A red arrow with the number '1' points to this box.
- Environment:** Shows the current working directory as `setwd("C:/Users/kadota/Desktop/hoge")`.
- Files:** A file explorer view showing the contents of the `hoge` directory:

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge.R	1.2 KB
hoge1.fasta	21 B
hoge1.txt	148 B
list.txt	19 B
sample1.fasta	29 B
- Console:** Shows the execution of `setwd("C:/Users/kadota/Desktop/hoge")` and provides help text for `demo()`, `help()`, `help.start()`, and `q()`.

応用27

いよいよ変更後のスクリプトの実行。①
赤枠内のコード全体を選択。こんな感じ。

The screenshot shows the RStudio interface. The script editor on the left contains R code, with lines 12 through 18 highlighted in blue and enclosed in a red rectangular selection box. A red circle with the number '1' and an arrow points to this selection box. The code in the editor is as follows:

```
12 obj <- is.element(as.character(data[,param]), keyv)
13 out <- data[obj,] #objがTRUE
14 dim(out) #オブジェクト
15
16 #ファイルに保存
17 write.table(out, out_f, sep="\t", append=F, quote=
18 |
```

The Environment pane on the right shows the current working directory: `setwd("C:/Users/kadota/Desktop/hoge...")`. The Files pane at the bottom right shows a file explorer view of the `C:/Users/kadota/Desktop/hoge` directory, listing files such as `.Rhistory`, `annotation.txt`, `genelist1.txt`, `hoge.R`, `hoge1.fasta`, `hoge1.txt`, `list.txt`, and `sample1.fasta`.

The Console pane at the bottom left shows the following output:

```
> setwd("C:/Users/kadota/Desktop/hoge")
> |
```

The console also contains introductory text for the R environment:

```
'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す。
'q()' と入力すれば R を終了します。
```

応用28

いよいよ変更後のスクリプトの実行。①
赤枠内のコード全体を選択。こんな感じ。
この状態で②Runを押すと…

The screenshot shows the RStudio interface with the following components:

- Code Editor:** Contains R code for selecting elements from a data frame and writing to a file. Lines 12-18 are highlighted with a red box. A red arrow labeled '2' points to the Run button above the editor.
- Environment/History/Connections:** Shows the current working directory set to 'C:/Users/kadota/Desktop/hoge...' with a red arrow labeled '1' pointing to the 'To Console' button.
- Files Panel:** Shows a file explorer view of the 'hoge' directory on the desktop, listing files like .Rhistory, annotation.txt, genelist1.txt, hoge.R, hoge1.fasta, hoge1.txt, list.txt, and sample1.fasta.
- Console:** Shows the output of the 'setwd()' command: `> setwd("C:/Users/kadota/Desktop/hoge")`.

応用29

いよいよ変更後のスクリプトの実行。①赤枠内のコード全体を選択。こんな感じ。この状態で②Runを押すと、出力予定ファイル(hoge1.txt)をExcelで開いている状態で実行したWinユーザは③のような感じでエラーが出ます。

The screenshot shows the RStudio interface. The source editor contains R code for selecting a column and writing it to a file. The console shows a permission error. The file explorer shows the directory contents.

```
12 obj <- rs.element(as.character(data[,param]), keyv...
13 out <- data[obj,] #objがTRUE
14 dim(out) #オブジェクト
15
16 #ファイルに保存
17 write.table(out, out_f, sep="\t", append=F, quote=
18 |
```

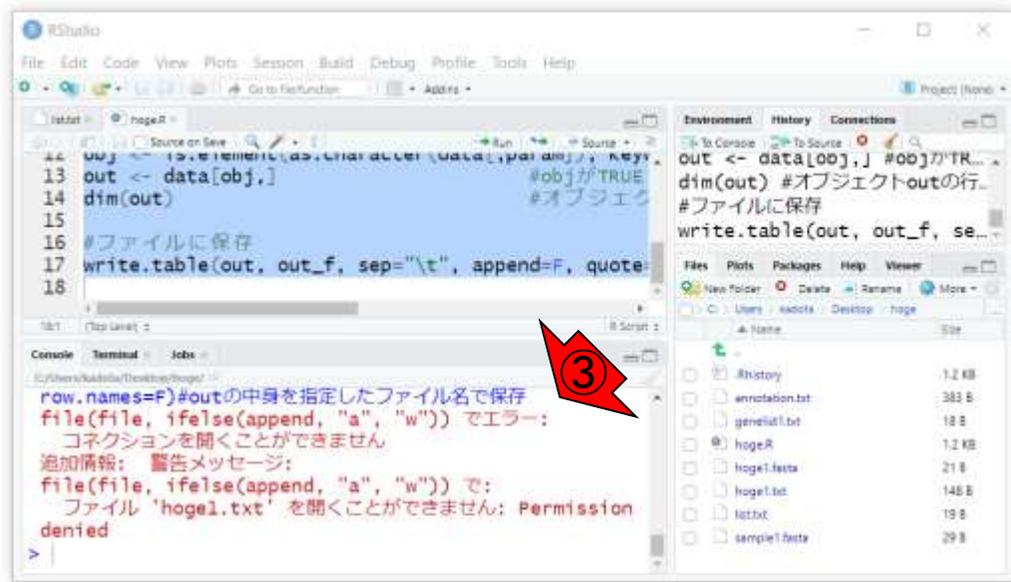
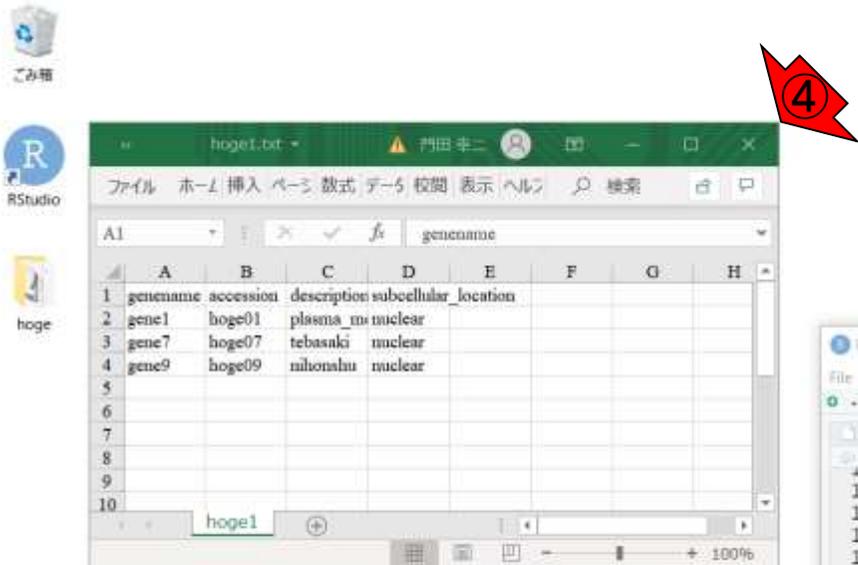
Console output:

```
C:/Users/kadota/Desktop/hoge/
row.names=F)#outの中身を指定したファイル名で保存
file(file, ifelse(append, "a", "w")) でエラー:
  コネクションを開くことができません
追加情報: 警告メッセージ:
file(file, ifelse(append, "a", "w")) で:
  ファイル 'hoge1.txt' を開くことができません: Permission
denied
> |
```

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge.R	1.2 KB
hoge1.fasta	21 B
hoge1.txt	148 B
list.txt	19 B
sample1.fasta	29 B

応用30

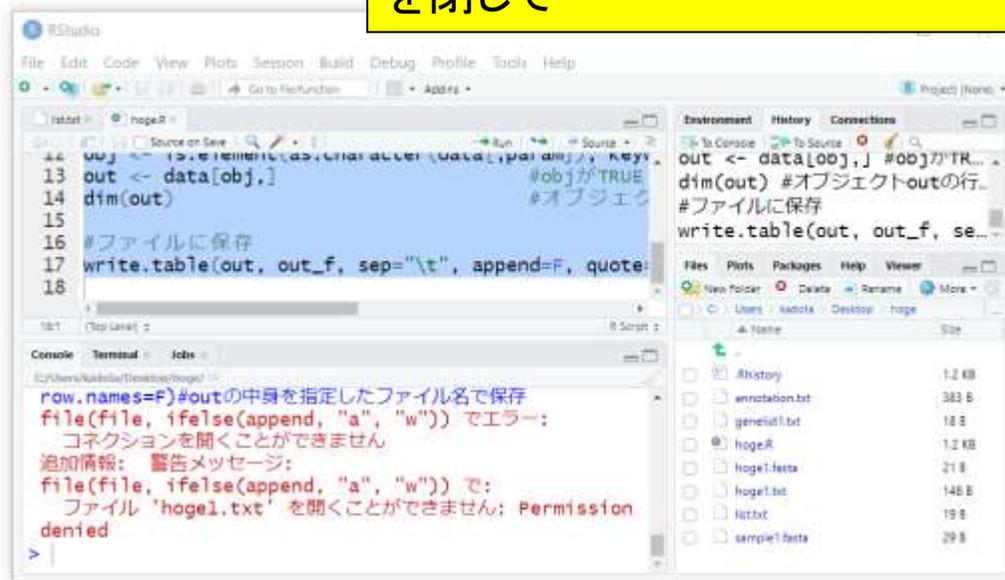
いよいよ変更後のスクリプトの実行。①赤枠内のコード全体を選択。こんな感じ。この状態で②Runを押すと、出力予定ファイル(hoge.1.txt)をExcelで開いている状態で実行したWinユーザは③のような感じでエラーが出ます。つまり、先に④のようにしていると、③のようになるということ。



応用31



いよいよ変更後のスクリプトの実行。①赤枠内のコード全体を選択。こんな感じ。この状態で②Runを押すと、出力予定ファイル(hoge.1.txt)をExcelで開いている状態で実行したWinユーザは③のような感じでエラーが出ます。つまり、先に④のようにしていると、③のようになるということ。④Excelで開いているhoge1.txtを閉じて…



応用32

いよいよ変更後のスクリプトの実行。①赤枠内のコード全体を選択。こんな感じ。この状態で②Runを押すと、出力予定ファイル(hoge1.txt)をExcelで開いている状態で実行したWinユーザは③のような感じでエラーが出ます。つまり、先に④のようにしていると、③のようになるということ。④Excelで開いているhoge1.txtを閉じて、⑤Runを押してリトライ。この作業自体はコピペ実行と同じです。



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

list.txt x hoge.R x

```
12 obj <- rs.element(as.character(data[,param]), K)
13 out <- data[obj,] #objがTRUE
14 dim(out) #オブジェクトの次元
15
16 #ファイルに保存
17 write.table(out, out_f, sep="\t", append=F, quote="")
18
```

18:1 (Top Level) R Script

Console Terminal x Jobs x

C:/Users/kadota/Desktop/hoge/

```
row.names=F)#outの中身を指定したファイル名で保存
file(file, ifelse(append, "a", "w")) でエラー:
  コネクションを開くことができません
追加情報: 警告メッセージ:
file(file, ifelse(append, "a", "w")) で:
  ファイル 'hoge1.txt' を開くことができません: Permission
denied
>
```

write.table(out, out_f, se...

Files Plots Packages Help Viewer

New Folder Delete Rename More

C: > Users > kadota > Desktop > hoge

	Name	Size
	..	
	.Rhistory	1.2 KB
	annotation.txt	383 B
	genelist1.txt	18 B
	hoge.R	1.2 KB
	hoge1.fasta	21 B
	hoge1.txt	148 B
	list.txt	19 B
	sample1.fasta	29 B

応用33

いよいよ変更後のスクリプトの実行。①赤枠内のコード全体を選択。こんな感じ。この状態で②Runを押すと、出力予定ファイル(hoge.1.txt)をExcelで開いている状態で実行したWinユーザは③のような感じでエラーが出ます。つまり、先に④のようにしていると、③のようになるということ。④Excelで開いているhoge1.txtを閉じて、⑤Runを押してリトライ。この作業自体はコピペ実行と同じです。実行結果。エラーは出なくなりましたね。⑥をクリックして中身をエディタで表示させてみましょう。

The screenshot shows the RStudio interface. The top pane displays an R script with the following code:

```
12 obj <- ts.element(as.character(data[,param]), K)
13 out <- data[obj,] #objがTRUE
14 dim(out) #オブジェクトの次元
15
16 #ファイルに保存
17 write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)
18
```

The bottom-left pane shows the console output:

```
> dim(out) #オブジェクトout
tの行数と列数を表示
[1] 7 4
>
> #ファイルに保存
> write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)
#outの中身を指定したファイル名で保存
>
```

The bottom-right pane shows a file explorer window with the following table:

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge.R	1.2 KB
hoge1.fasta	21 B
hoge1.txt	265 B
list.txt	19 B
sample1.fasta	29 B

A red arrow with the number 6 points to the file 'hoge1.txt' in the file explorer.

応用34

いよいよ変更後のスクリプトの実行。①赤枠内のコード全体を選択。こんな感じ。この状態で②Runを押すと、出力予定ファイル(hoge1.txt)をExcelで開いている状態で実行したWinユーザは③のような感じでエラーが出ます。つまり、先に④のようにしていると、③のようになるということ。④Excelで開いているhoge1.txtを閉じて、⑤Runを押してリトライ。この作業自体はコピペ実行と同じです。実行結果。エラーは出なくなりましたね。⑥をクリックして中身をエディタで表示させてみましょう。⑦表示結果。

The screenshot shows the RStudio interface. The script editor contains a table with 7 rows and 4 columns. The console shows the execution of `dim(out)` resulting in `[1] 7 4`, and `write.table` being used to save the output to `hoge1.txt`. A file explorer window is open, showing the contents of the `hoge` directory, with `hoge1.txt` selected and highlighted by a red arrow labeled ⑥.

```
1 gene_name accession description subcellular_location
2 gene1 hoge01 plasma_mem nuclear
3 gene2 hoge02 hohinu membrane
4 gene5 hoge05 kamo membrane
5 gene7 hoge07 tebasaki nuclear
6 gene8 hoge08 biiru nuclear
7 gene9 hoge09 nihonku nuclear
```

```
> dim(out) #オブジェクトout
tの行数と列数を表示
[1] 7 4
>
> #ファイルに保存
> write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F) #outの中身を指定したファイル名で保存
>
```

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge.R	1.2 KB
hoge1.fasta	21 B
hoge1.txt	265 B
list.txt	19 B
sample1.fasta	29 B

応用35

いよいよ変更後のスクリプトの実行。①赤枠内のコード全体を選択。こんな感じ。この状態で②Runを押すと、出力予定ファイル(hoge1.txt)をExcelで開いている状態で実行したWinユーザは③のような感じでエラーが出ます。つまり、先に④のようにしていると、③のようになるということ。④Excelで開いているhoge1.txtを閉じて、⑤Runを押してリトライ。この作業自体はコピペ実行と同じです。実行結果。エラーは出なくなりましたね。⑥をクリックして中身をエディタで表示させてみましょう。⑦表示結果。確かに意図通り⑧4列目がnuclear or membraneとなっている行の抽出ができていますね。

The screenshot shows the RStudio interface. At the top, the menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. Below the menu bar is a toolbar with various icons. The main editor window displays a script with the following content:

```
1 gene_name accession description subcellular_location
2 gene1 hoge01 plasma_mem nuclear
3 gene2 hoge02 hohinu membrane
4 gene5 hoge05 kamo membrane
5 gene7 hoge07 tebasaki nuclear
6 gene8 hoge08 biiru nuclear
7 gene9 hoge09 nihonaki nuclear
```

Red arrows labeled 7 and 8 point to the script content. Below the editor is the Console window, which shows the following output:

```
> dim(out)
#オブジェクトout
tの行数と列数を表示
[1] 7 4
>
> #ファイルに保存
> write.table(out, out_f, sep="\t", append=F, quote=F,
row.names=F)#outの中身を指定したファイル名で保存
> |
```

At the bottom right, a file explorer window shows the contents of the directory C:/Users/kadota/Desktop/hoge/. The files listed are:

File Name	Size
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge.R	1.2 KB
hoge1.fasta	21 B
hoge1.txt	265 B
list.txt	19 B
sample1.fasta	29 B

A red arrow labeled 6 points to the hoge1.txt file in the file explorer.

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

警告メッセージ1

The screenshot shows the RStudio interface. The script editor contains a table of gene data. The console window shows the execution of R code and a warning message. Red arrows point to the warning message in the console and the 'To Console' button in the Environment pane.

```
1 geneName accession description subcellular_location
2 gene1 hoge01 plasma_mem nuclear
3 gene2 hoge02 hohinu membrane
4 gene5 hoge05 kamo membrane
5 gene7 hoge07 tebasaki nuclear
6 gene8 hoge08 biiru nuclear
7 gene9 hoge09 nihonryu nuclear
```

```
> dim(out) #オブジェクトoutの行数と列数を表示
[1] 7 4
> #ファイルに保存
> write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F) #outの中身を指定したファイル名で保存
> |
```

Environment pane: out <- data[obj,] #objがTR...
dim(out) #オブジェクトoutの行数と列数を表示
write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F) #outの中身を指定したファイル名で保存

Files pane: C:\Users\kadota\Desktop\hoge
Name Size
..
.Rhistory 1.2 KB
annotation.txt 383 B
genelist1.txt 18 B
hoge.R 1.2 KB
hoge1.fasta 21 B
hoge1.txt 265 B
list.txt 19 B
sample1.fasta 29 B

警告メッセージ2

①を押して、②Console画面を最大化。こんな感じになります。③ページ上部に移動して、コピペ実行したコードの最初のほうが見えるようにします。

The screenshot shows the RStudio interface. The console on the left contains the following R code and its output:

```
> dim(data) #オブジェクトdataの行数と列数を表示
[1] 11 4
>
> #本番
> obj <- is.element(as.character(data[,param]), keywords) #条件を満たすかどうかを判定した結果をobjに格納
> out <- data[obj,] #objがTRUEとなる行のみ抽出した結果をoutに格納
> dim(out) #オブジェクトoutの行数と列数を表示
[1] 7 4
>
> #ファイルに保存
> write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F) #outの中身を指定したファイル名で保存
>
```

The right-hand pane shows a file explorer for the directory `C:\Users\kadota\Desktop\hoge`. The files listed are:

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge.R	1.2 KB
hoge1.fasta	21 B
hoge1.txt	265 B
list.txt	19 B
sample1.fasta	29 B

A red arrow with the number 3 points to the top of the file explorer pane.

警告メッセージ3

①を押して、②Console画面を最大化。こんな感じになります。③ページ上部に移動して、コピペ実行したコードの最初のほうが見えるようにします。こんな感じ。

The screenshot shows the RStudio interface. The console on the left contains the following R code and its output:

```
> in_f1 <- "annotation.txt" #入力ファイル名
#指定してin_f1に格納(アノテーションファイル)
> in_f2 <- "list.txt" #入力ファイル名を指定
#してin_f2に格納(リストファイル)
> out_f <- "hoge1.txt" #出力ファイル名
#を指定してout_fに格納
> param <- 4 #アノテーション
#ファイル中の検索したい列番号を指定
>
> #入力ファイルの読み込み
> data <- read.table(in_f1, header=TRUE, sep="\t", quote="")#in_f1で指定したファイルの読み込み
> keywords <- readLines(in_f2) #in_f2で指定したファイルの読み込み
> dim(data) #オブジェクトdataの行数と列数を表示
[1] 11 4
```

The right-hand pane shows the 'Files' view of the current directory (C:\Users\kadota\Desktop\hoge). The file list is as follows:

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge.R	1.2 KB
hoge1.fasta	21 B
hoge1.txt	265 B
list.txt	19 B
sample1.fasta	29 B

A red arrow with the number 3 points to the top of the file list, indicating the instruction to scroll to the top of the page.

警告メッセージ4

①を押して、②Console画面を最大化。こんな感じになります。③ページ上部に移動して、コピペ実行したコードの最初のほうが見えるようにします。こんな感じ。これは、④の部分で改行を入れた、⑤ list.txtを、⑥の部分で読み込ませたもの。特に何のメッセージも表示されていない。

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

Source

```
> in_f1 <- "annotation.txt" #入力ファイル名
#入力ファイル名を指定してin_f1に格納(アノテーションファイル)
> in_f2 <- "list.txt" #入力ファイル名を指定
#入力ファイル名を指定してin_f2に格納(リストファイル)
> out_f <- "hoge1.txt" #出力ファイル名
#出力ファイル名を指定してout_fに格納
> param <- 4 #アノテーション
#アノテーションファイル中の検索したい列番号を指定
>
> #入力ファイルの読み込み
> data <- read.table(in_f1, header=TRUE, sep="\t", quote="") #in_f1で指定したファイルの読み込み
> keywords <- readLines(in_f2) #in_f2で指定したファイルの読み込み
> dim(data) #オブジェクトdata
[1] 11 4
```



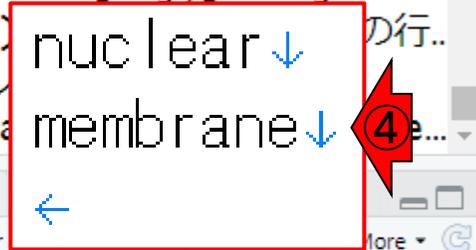
To Console To Source

```
out <- data[ob1,] #ob1がTR...
dim(out)
#ファイル
write.ta
```

nuclear ↓
membrane ↓

Files Plots
New Folder
C: > Users > kadota > Desktop > hoge

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge.R	1.2 KB
hoge1.fasta	21 B
hoge1.txt	265 B
list.txt	19 B
sample1.fasta	29 B



警告メッセージ5

④の部分で改行を入れずに⑤list.txtで実行すると、⑥当該ファイルの読み込み部分で、警告メッセージがでます。この場合は⑧出力ファイルを眺めることで結果には影響しないことがわかる。Rは警告メッセージ後の記述内容が比較的分かりやすいのでよく読むべし。

The screenshot shows the RStudio interface with the following elements:

- Source Editor:** Contains R code for reading files and searching for keywords. Annotations ⑤, ⑥, and ⑦ point to specific lines of code.
- Console:** Shows the execution output, including a warning message from `readLines` and the start of a `dim` function call. Annotation ④ points to the `nuclear` keyword in the output.
- Files Panel:** Shows a file explorer view of the `hoge` directory. Annotation ⑧ points to the `list.txt` file.

```
> in_f1 <- "annotation.txt" #入力ファイル名
#指定してin_f1に格納(アノテーションファイル)
> in_f2 <- "list.txt" #入力ファイル名を指定
#指定してin_f2に格納(リストファイル)
> out_f <- "hoge1.txt" #出力ファイル名
#指定してout_fに格納
> param <- 4 #アノテーション
#ファイル中の検索したい列番号を指定
>
> #入力ファイルの読み込み
> data <- read.table(in_f1, header=TRUE, sep="\t", quo
te="")#in_f1で指定したファイルの読み込み
> keywords <- readLines(in_f2) #in_f2で指定し
たファイルの読み込み
警告メッセージ:
readLines(in_f2) で: 'list.txt' で不完全な最終行が見つ
かりました
dim(data) #オブジェクトの
```

Files Panel:

Name	Size
..	
.Rhistory	1.2 KB
annotation.txt	383 B
genelist1.txt	18 B
hoge.R	1.2 KB
hoge1.fasta	21 B
hoge1.txt	265 B
list.txt	17 B
sample1.fasta	29 B

Contents

- 前提条件、RStudioの起動
- 見栄えの統一、終了
- Console画面の見栄え、設定変更
- 枠の幅、電卓的な利用
- プロンプト(>)、Console画面クリア
- 計算の中断
- 指定した範囲の配列を取得
 - 項目と例題、入力ファイルの保存
 - 作業ディレクトリの変更
 - 入力ファイルの確認、FASTA形式
 - コピペ実行、出力ファイルの確認、Environmentタブ
- フィルタリング(任意のキーワードを含む行を抽出)
 - 全体像の把握、入力ファイルの保存、コピペ実行、出力ファイルの確認
 - Winユーザ向け注意点、応用
- 警告メッセージ、ありがちなミス

ありがちなミス0

この後エラーを意図的に出すので、まずはうまくいく例を必要最小限のコードで示す。

The screenshot shows the RStudio interface. The console pane on the left displays the following R code and its output:

```
> in_f1 <- "annotation.txt" #
> in_f2 <- "genelist1.txt" #
> out_f <- "hoge1.txt" #
> param <- 1 #
>
> #入力ファイルの読み込み
> data <- read.table(in_f1, header=TRUE, sep="\t", quote="")#
> keywords <- readLines(in_f2) #
> dim(data) #
[1] 11 4
>
> #本番
> obj <- is.element(as.character(data[,param]), keywords)#
> out <- data[obj,] #
> dim(out) #
[1] 3 4
>
> #ファイルに保存
> write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)#
> |
```

The environment pane on the right shows the current environment with the following objects:

```
obj <- is.element(as.char...
out <- data[obj,] #
dim(out) #
#ファイルに保存
```

The Files pane at the bottom right shows the current directory structure:

Name	Size
..	
annotation.txt	383 B
genelist1.txt	18 B

ありがちなミス1

①作業ディレクトリの変更を忘れているため、②in_f1で指定した最初のファイルの読み込み段階で、③エラーが出る。作業ディレクトリ内には入力ファイルは存在しないので当然。

The screenshot shows the RStudio interface. The console window displays the following R code and its output:

```
> in_f1 <- "annotation.txt" #
> in_f2 <- "genelist1.txt" #
> out_f <- "hogel.txt" #
> param <- 1 #
>
> #入力ファイルの読み込み
> data <- read.table(in_f1, header=TRUE, sep="\t", quote="")#
file(file, "rt") でエラー: コネクションを開くことができません
追加情報: 警告メッセージ:
file(file, "rt") で:
ファイル 'annotation.txt' を開くことができません: No such file or dire
ctory
> keywords <- readLines(in_f2) #
file(con, "r") でエラー: コネクションを開くことができません
追加情報: 警告メッセージ:
file(con, "r") で:
ファイル 'genelist1.txt' を開くことができません: No such file or direc
tory
> dim(data) #
[1] 11 4
>
```

Red arrows point to the following lines in the console:

- ② points to the line `in_f1 <- "annotation.txt"`.
- ③ points to the line `#入力ファイルの読み込み`.

The Environment pane on the right shows the following R code:

```
out <- data[obj,] #
dim(out) #
#ファイルに保存
write.table(out, out_f, s...
```

The Files pane on the right shows the file explorer view, with a red arrow pointing to the Home directory:

- ① points to the Home directory in the file explorer.

ありがちなミス2

①作業ディレクトリの変更はできているが、②in_f2で指定した最初のファイルが③存在しないので、④その読み込み段階でエラーが出る。

The screenshot shows the RStudio interface. The console window displays the following R code and its output:

```
> in_f1 <- "annotation.txt" #
> in_f2 <- "genelist1.txt" #
> out_f <- "hoge1.txt" #
> param <- 1 #
>
> #入力ファイルの読み込み
> data <- read.table(in_f1, header=TRUE, sep="\t", quote="")#
> keywords <- readLines(in_f2) #
file(con, "r") でエラー: コネクションを開くことができません
追加情報: 警告メッセージ:
file(con, "r") で:
  ファイル 'genelist1.txt' を開くことができません: No such file or direc
tory
> dim(data) #
[1] 11 4
>
> #本番
> obj <- is.element(as.character(data[,param]), keywords)#
> out <- data[obj,] #
> dim(out) #
[1] 3 4
```

The file explorer on the right shows the current directory path: C:\Users\kadota\Desktop\hoge. The files listed are '..' and 'annotation.txt'. A red arrow points to the 'annotation.txt' file.

Red arrows with numbers 1, 2, 3, and 4 point to specific elements in the screenshot:

- ①: Points to the file explorer window.
- ②: Points to the line `in_f2 <- "genelist1.txt"` in the console.
- ③: Points to the file `annotation.txt` in the file explorer.
- ④: Points to the error message `file(con, "r") でエラー: コネクションを開くことができません` in the console.

ありがちなミス3

①出力予定ファイルが既に作業ディレクトリ内に存在し、それを②Excelなど別のプログラムで開いているWinユーザは、③最後のwrite.table関数のところでエラーが出る。

The screenshot shows RStudio with a file explorer on the right displaying a directory named 'hoge' containing files 'annotation.txt', 'genelist1.txt', and 'hoge1.txt'. A red arrow labeled '1' points to 'hoge1.txt'. The central spreadsheet view shows a table with columns 'A', 'B', 'C', 'D', 'E' and rows of gene data. A red arrow labeled '2' points to the 'hoge1.txt' tab. The console on the left shows R code and error messages. A red arrow labeled '3' points to the error message.

```
> #入力ファイルの読み込み
> data <- read.table(in_f
> keywords <- readLines(i
> dim(data)
[1] 11 4
>
> #本番
> obj <- is.element(as.ch
> out <- data[obj,]
> dim(out)
[1] 3 4
>
> #ファイルに保存
> write.table(out, out_f, sep="\t", append=F, quote=F, row.names=F)#
file(file, ifelse(append, "a", "w")) でエラー:
コネクションを開くことができません
追加情報: 警告メッセージ:
file(file, ifelse(append, "a", "w")) で:
ファイル 'hoge1.txt' を開くことができません: Permission denied
>
```

	A	B	C	D	E
1	gene1	hoge01	plasma_m	nuclear	
2	gene2	hoge02	hohinu	membrane	
3	gene5	hoge05	kamo	membrane	
4	gene7	hoge07	tebasaki	nuclear	
5	gene8	hoge08	biiru	nuclear	
6	gene9	hoge09	nihonshu	nuclear	
7	gene10	hoge10	agene1	membrane	
8					
9					