

# 初心者セッション： R & RStudio入門

BeginnerR Session: R & RStudio 101

22<sup>nd</sup> October 2022, Tokyo.R #102

Yuta Kanzawa @yutakanzawa



Data Scientist at Zurich Insurance Company Limited, Japan Branch



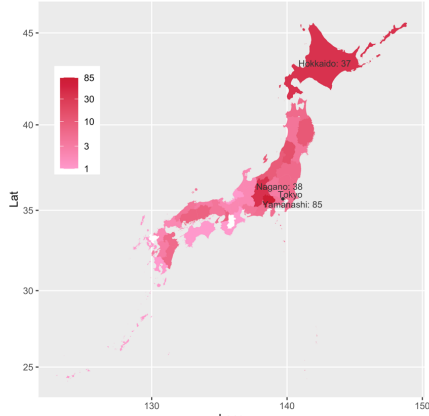
# 神沢雄大 Yuta Kanzawa

- データサイエンティスト@チューリッヒ保険会社
  - 日本支店
- Twitter: [@yutakanzawa](https://twitter.com/yutakanzawa)
- 好きなもの：オペラとワイン
  - ワーグナー
  - ブルゴーニュ (WSET Lv 3→?)
- 使用可能言語：7
  - 人間：日本語、英語、ドイツ語
  - コンピューター：R, Python, SAS, SQL



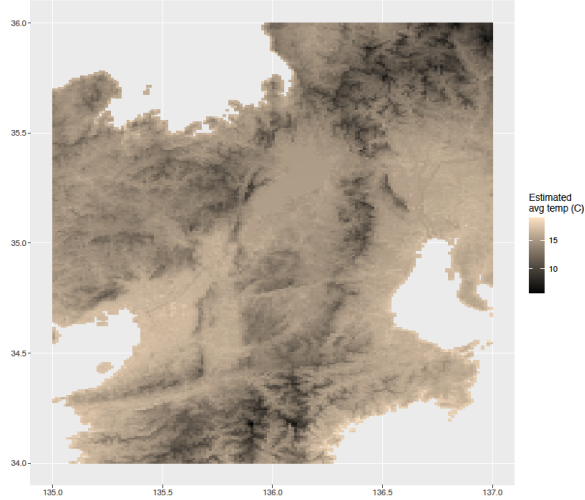
# ポートフォリオ

Number of Wineries in Japan in 2019, by Prefecture

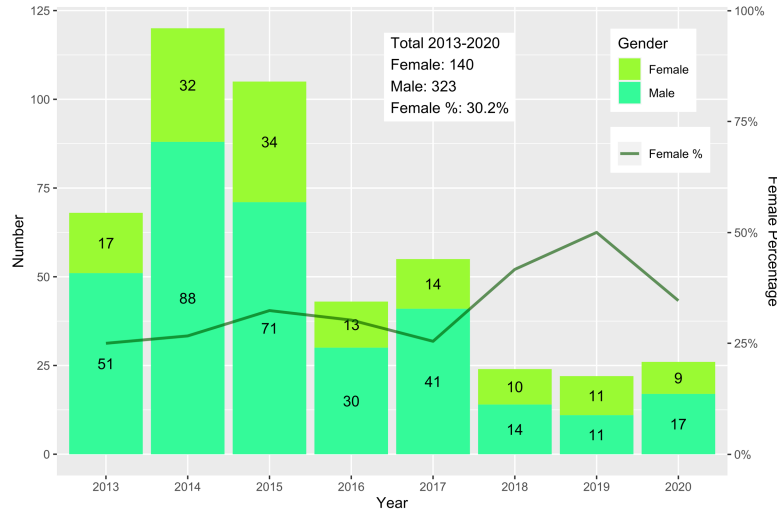


Source: <https://www.nta.go.jp/taxes/sake/shiori-gaikyo/selzogaikyo/kajitsu/pdf/30/30wine01.pdf>

Estimated Avg Temperature around Kyoto

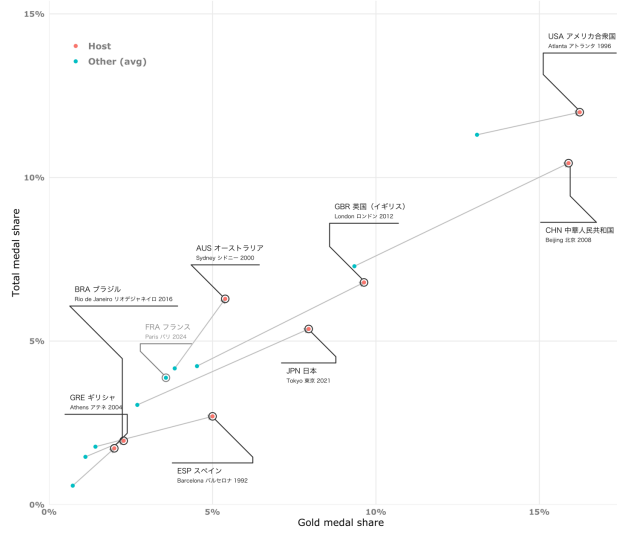


Number of Qualified JSA Sommelier Excellence and Equivalents\* by Year and Gender, 2013-2020



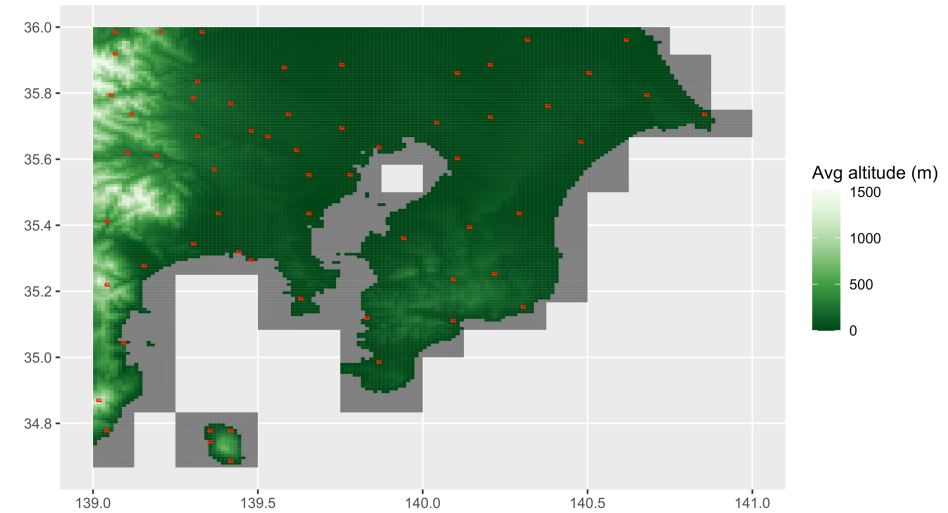
Source: Japan Sommelier Association <https://www.sommelier.jp/exam/pdf/qualifiedholders.pdf>  
 \*Sommelier Excellence (2019-2020), Senior Sommelier (2013-2018), Senior Wine Adviser (2013-2015)

Medal shares of Olympic host countries in the past 30 years

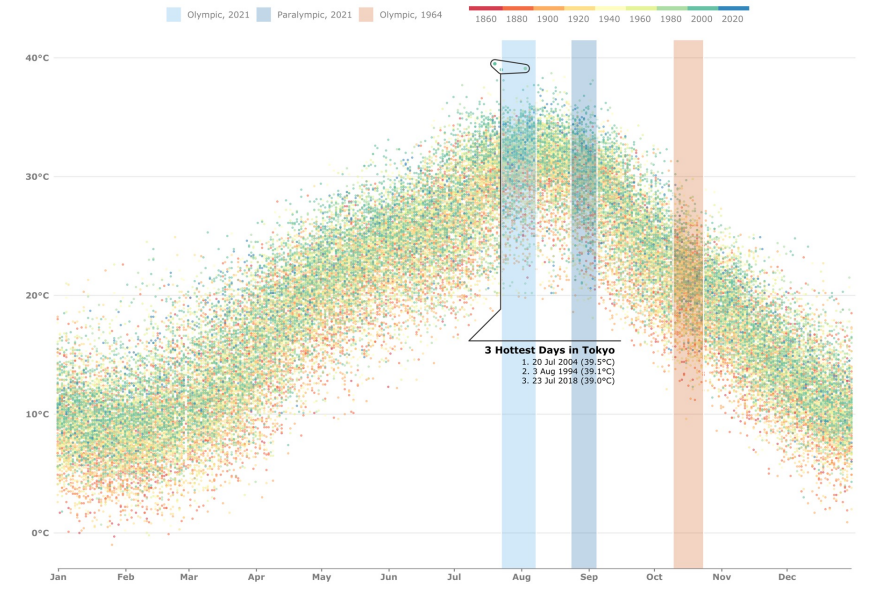


Data: International Olympic Committee via <https://olympics.com> & <https://www.wikipedia.org> - Graphic: Yuta Kanzawa

Avg Altitudes and Weather Observation Stations in Tokyo, Kanagawa, Chiba



Daily maximum temperature in Tokyo, 1875-2021



Data: Japan Meteorological Agency via <https://www.jma.go.jp> - Graphic: Yuta Kanzawa (inspired by Cédric Scherer)

# アジェンダ

- 今日話すこと
  - RとRStudioのインストールと設定
  - RStudioのUI
  - ライブラリーのインストールと管理
- 対象（以下のいずれか）
  - RまたはRStudioを初めて触る人
  - 普段Rを単体で使っている人
  - RStudioをなんとなく使っている人

- 今日話さないこと
  - Jupyter
  - Rの文法

→ R環境の包括的理解

# TL;DR

- R言語をRStudioの環境で使うと◎。
  - 豊富なライブラリー+作業効率
- RStudioの**プロジェクト**機能←オススメ！
- RStudioの4つの窓
  - オブジェクト名と内容、実行したコマンドを確認できる。
  - 便利なショートカットキー
  - ヘルプ参照
- ライブラリー
  - Rのメジャーバージョンアップ時は要注意。

# RとRStudioを選ぶ理由

Why R & RStudio?



# R = ソフトウェア

- アドホックな分析から機械学習パイプラインまで、**幅広いデータサイエンスのタスク**に使える。
  - 豊富なライブラリー群
    - 統計解析
    - 機械学習、深層学習
    - データ可視化
- 長所でもあり短所：**全データをメモリーに乗せる\***。
  - ベクトル（行方向）の演算が早い。
  - 容量の上限：32ビットマシンでは4Gバイト。
  - オブジェクトの長さの上限： $2^{31}-1$ 
    - エラー：cannot allocate vector of length

\* <https://stat.ethz.ch/R-manual/R-devel/library/base/html/Memory-limits.html>

# RStudio = IDE (統合開発環境)



- R専用
  - 他の言語もサポート。
- 作業効率が上がる。
- ~~YouTubeを見たり、部屋の照明を操作できる。~~
- 「RStudio」から「Posit」へ名称変更予定\*。
  - 社名、有償製品
  - (少なくともフリー版の) IDEとしての「RStudio」はそのまま。
- 読み方：「アールステューディオ」、「アールスタジオ」などお好みで。
- 短所：マルチバイト文字が苦手な場合がある。

\* <https://posit.co/>



# インストールと設定

Installs & settings

# ダウンロード元

- 公式サイト

- R本体 : **CRAN** (The Comprehensive R Archive Network)\*
  - <https://cran.r-project.org/>
  - Windowsは「base」と「Rtools」(UNIXの機能を移植) をダウンロード。
- RStudio (今のところ) : **RStudio社のwebサイト**
  - <https://www.rstudio.com/>

- パッケージ管理システム

- Mac: homebrew
- Linux: yum
- 説明は省略。

CRANのトップページのダウンロードリンク

## Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux \(Debian, Fedora/Redhat, Ubuntu\)](#)
- [Download R for macOS](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

\* 「CRAN」の読み方は「クラン」、「シーラン」などお好みで。

# インストール先

- インストーラーのデフォルト
  - 基本的にこれでOK。
- (環境に依っては) ユーザーフォルダー
  - 例：Windowsで「C:¥Program Files」への書き込み制限あり。
  - 自分でPATHを通す必要がある。
- または、IT部門に頼んで管理者権限でインストールしてもらおう。
  - 業務や研究で使うソフトウェア環境を提供するのはIT部門の仕事。
  - ただし、バージョンアップがやや面倒。

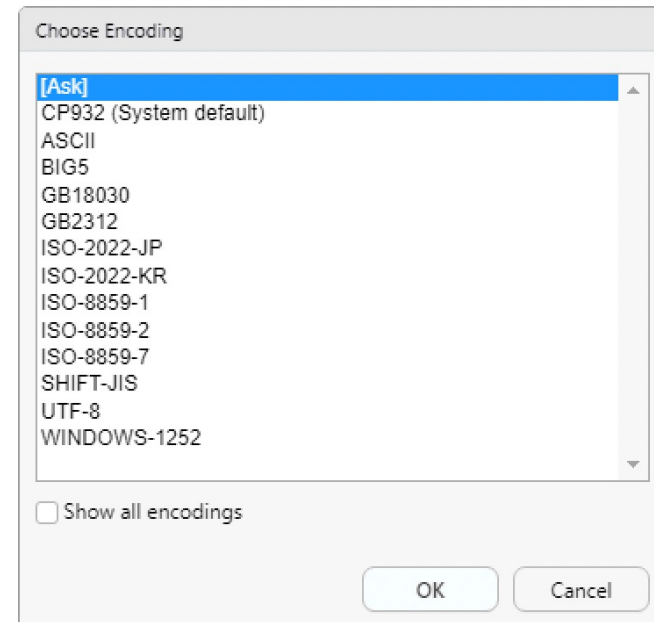
# インストール方法

- ダウンロードした**インストーラー**を実行する。
  - インストール先を変えない場合、**デフォルトの設定**のままでOK。
  - Rtoolsも同様。
- 安全なインストール順：
  - R本体 → (Windowsの場合Rtools) → RStudio
- LinuxについてはCRANのドキュメントを参照。

# RStudioの文字コード設定

- システムデフォルトの文字コードが**UTF-8**の環境では初期値でOK。
  - Windows (**Shift-JIS**や**CP932**) では**文字化け**の恐れ。
    - 場所：Tools→Global Options→Code→Saving→Default text encoding
      - バージョンによる違いあり。

「文字化け」→「譁?1怜喧纏」



# アップデート

- 基本的にインストールと同じ手順。
- ただし、R本体は予めアンインストールしておく。
  - 複数のバージョンが併存してしまう。
  - RStudioとRtoolsは自動で上書きされる。
- R本体のメジャーバージョンアップ（例：4.1→4.2）は要注意。
  - ライブラリーの移動、再インストールが必要（後述）。

# 作業環境

Workspace

# パス (Path)

- 絶対パス (フルパス)
  - **最も上のレベルのフォルダーから** 目的のファイルやフォルダーへの道順
    - 住所のようなもの。
    - 例 : C:\Program Files\R\R-4.2.1\bin\R.exe
- 相対パス
  - **現在のフォルダーから** 目的のファイルやフォルダーへの道順
  - 例 : ..\R-4.2.1\bin\R.exe
    - 現在地 : C:\Program Files\R\R-4.3.0
    - 目的地 : C:\Program Files\R\R-4.2.1\bin\R.exe
- ルート (root) : **基準となるフォルダー** (厳密にはシステムの最上位)
  - 例 : ライブラリーのトップレベルのフォルダー



# 問題：Rをデータやコードのある場所に導くには？

Me: `setwd("C:¥Users¥me¥R¥analysis")`

Jenny\*:

If the first line of your R script is

```
setwd("C:\Users\jenny\path\that\only\I\have")
```

I\* will come into your office and  
SET YOUR COMPUTER ON FIRE 🔥.

\* or maybe Timothée Poisot will



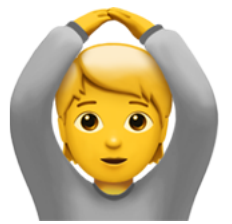
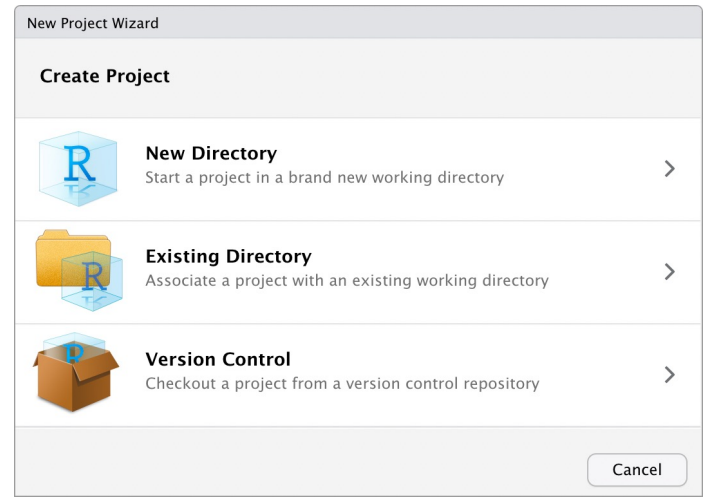
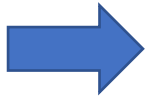
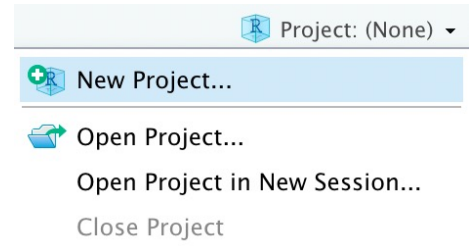
\* <https://twitter.com/hadleywickham/status/940021008764846080>

おすすめ!

# RStudioのプロジェクト機能



- プロジェクト（一連のコードやデータ）のルートフォルダーを設定。
  - 作業フォルダーの設定が不要！
  - 他の端末でも同じように動かせる。
- 画面右上の「Project」ボタンから。

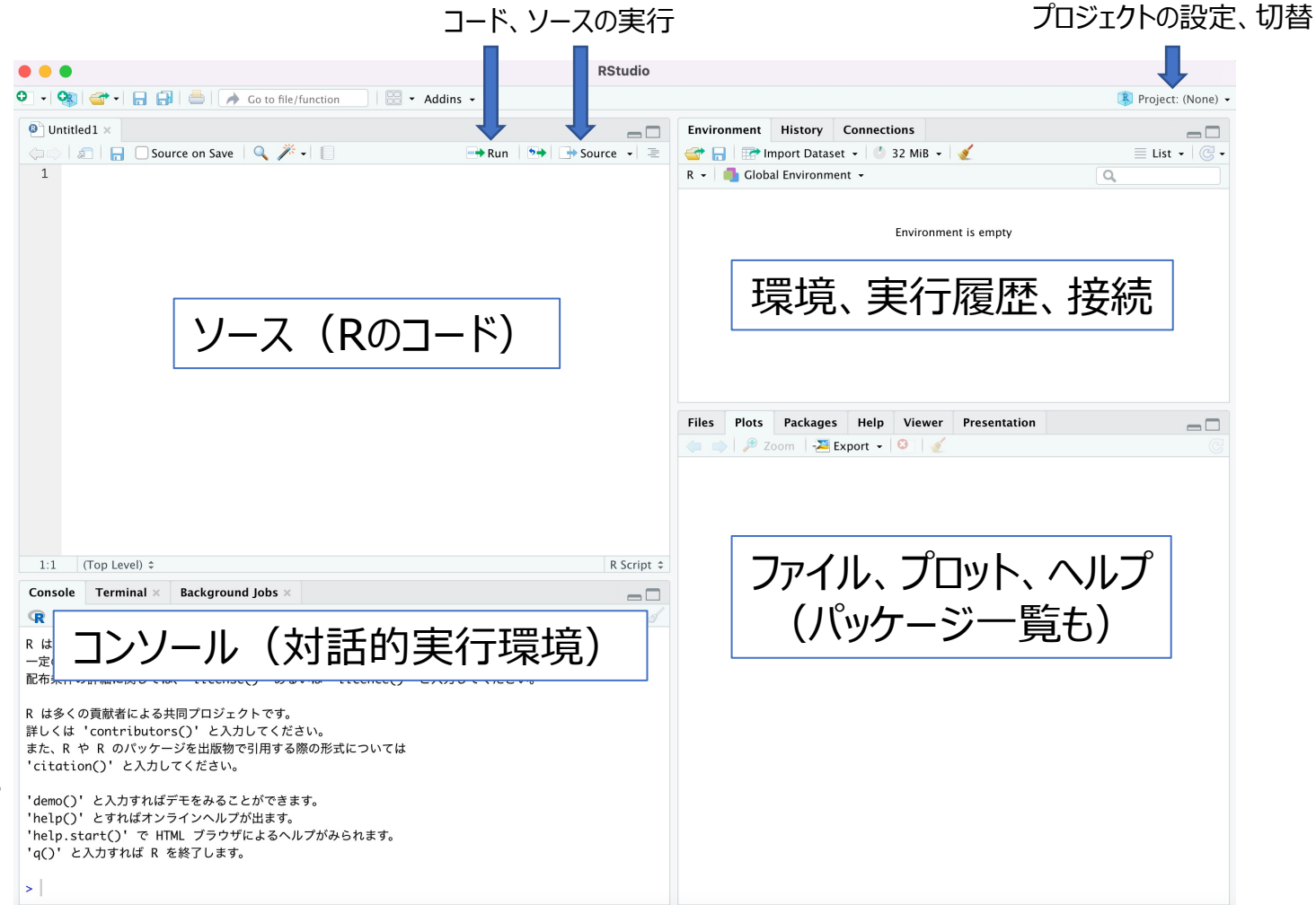


# RStudioのUI

UI of RStudio

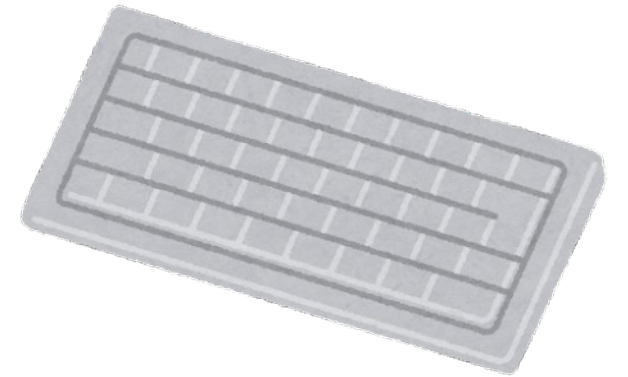
# 4つの窓（ペイン）

- デフォルト（右図）
  - ソース
    - ここで読み書き。
  - コンソール
    - 「>|」のところに入力。
    - ヘルプ入力
    - Rコマンドの実行
  - 環境、実行履歴
    - オブジェクト名と内容
    - Rコマンドを再実行できる。
  - プロット、ヘルプ



# コード編集に便利なショートカットキー



- コメント : 「#」
  - Win: 「Ctrl+Shift+C」、Mac: 「Cmd+Shift+C」
  - アンコメント (コメント解除) も同じ。
- 代入演算子 : 「<-」
  - Win: 「Alt+-」、Mac: 「Option+-」
- パイプ演算子 : 「%>%」
  - Win: 「Ctrl+Shift+M」、Mac: 「Cmd+Shift+M」
- ファイルの保存
  - Win: 「Ctrl+S」、Mac: 「Cmd+S」



豆知識 : エディターのデフォルトフォント

- Windows: Lucida Console
- Mac: Monaco

# コードとソースの実行

- コード（ファイルの一部）の実行
  - 実行範囲：カーソルのある行を含む一連のコードもしくは選択範囲
  - ソースペイン右角の「Run」ボタン  Run
  - ショートカットキー
    - Win: 「Ctrl+Enter」、Mac: 「Cmd+Return」
- ソース（ファイル全体）の実行
  - ソースペイン右角の「Source」ボタン  Source
  - ショートカットキー
    - 出力あり → Win: 「Ctrl+Shift+Enter」、Mac: 「Cmd+Shift+Return」
    - 出力なし → Win: 「Ctrl+Shift+S」、Mac: 「Cmd+Shift+S」


# ヘルプ参照

- 関数の使い方（引数と返り値の説明や実行例）
- Rの標準機能
  - 調べたい関数名の**前に「？」**を付けたコマンドをコンソールで実行。
    - 例：「?help」
  - RStudioでは**右下のペイン**に表示される。
    - HTMLドキュメントが表示されるので、リンクを辿って調べることもできる。
  - オフライン環境でも調べられる。
- 短所：**正確な名前**を入力しないと見つからない。
  - 「??」を付けて検索してみる。
  - felpライブラリー\*のfuzzyhelp()（あいまい検索）



\* <https://github.com/atusy/felp>

# おまけ : R Markdown(.Rmd)ファイルのknit実行

- ソースペイン上の「Knit」ボタン  Knit
- ショートカットキー
  - Win: 「Ctrl+Shift+K」、Mac: 「Cmd+Shift+K」





# ライブラリーの インストールとアップデート

Installs & updates of libraries

# 「ライブラリー」? 「パッケージ」?

- ライブラリー
  - 何らかの機能の集合 (**概念**)



- パッケージ
  - その機能を実装したもの (**実体**)



# インストールと利用

- インストール方法
  - `install.packages()`にインストールするライブラリー名を与える。
    - 例：`install.packages("tidyverse")`
- インストール先：デフォルトではローカルにインストールされる。
  - `.libPaths()`で確認できる。
  - 任意の場所に配置できる。→設定ファイル「.Renviron」\*
- 使う時に読み込む（呼び出す）必要がある。
  - `library()`がオススメ。1つずつ読み込む（コードの最初に1回だけ）。
    - 例：`library(tidyverse)` ←引用符なし！
- アップデート：インストールと同じ。

\* <https://stat.ethz.ch/R-manual/R-patched/library/base/html/Startup.html>

# Rのメジャーバージョンアップ時の対応

- ライブラリーはRのバージョンに紐づくフォルダーに置かれる。
  - 例：R/win-library/4.2/
- 問題点
  - 「4.2 → 4.3」や「4.2 → 5.0」というバージョンアップ  
→ ライブラリーが参照できなくなる！
    - 注：「4.2.0 → 4.2.1」というマイナーバージョンアップでは影響なし。
- 対応策
  - 新しいバージョンのフォルダーに移動する。→ 順次インストールし直す。

# 警告「パッケージ〇〇はR x.y.zの下で構築されました」

- 英語 : Package 'hoge' was built under R version x.y.z
- ライブラリーの読み込み時やインストール時にこのような警告が出る。
  - 原因 : Rのバージョン違い
  - 対応策 :
    - 読み込み時
      - 当該ライブラリーをインストールし直す。
      - 自分の使っているRのバージョンがライブラリーのインストール時より新しい。
    - インストール時
      - Rをバージョンアップする。
      - 自分の使っているRのバージョンがCRANの最新版より古い。

# ネット接続がないか制限下 : miniCRAN



- オフライン環境にて予め指定したライブラリーの使用を可能にする\*。
  - オンライン環境にてローカルリポジトリを構築。
    - 依存関係を含む必要なライブラリーが1つのフォルダーにダウンロードされる。
  - オフライン環境にコピーする。
    - 媒体に焼くなどして。
- Dockerが使えない場合に特に有用。

\* <https://learn.microsoft.com/ja-jp/sql/machine-learning/package-management/create-a-local-package-repository-using-minicran>

# まとめ

Long story short

# Long story short (1/2)

- R言語をRStudioの環境で使うと◎。
  - 豊富なライブラリー+作業効率
  - インストールは基本的にデフォルト設定のままOK。
  - R本体をアップデートするときは予めアンインストールしておく。
- RStudioの**プロジェクト機能**←オススメ！
  - プロジェクト（一連のコードやデータ）の**ルートフォルダー**を設定。
    - 作業フォルダーの設定が不要！
    - 他の端末でも同じように動かせる。



# Long story short (2/2)

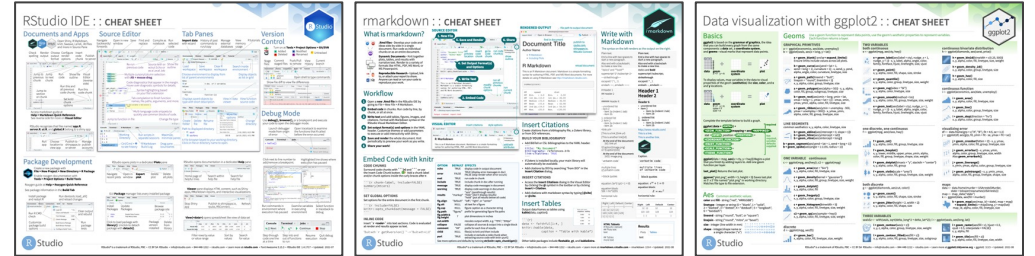
- RStudioの4つの窓
  - 右上：オブジェクト名と内容、実行したコマンドを確認できる。
  - 便利な**ショートカットキー**：コメント、代入演算子、パイプ演算子、実行
  - ヘルプ参照：右下のペインの表示内容からリンクを辿れる。
- ライブラリー
  - インストール：`install.packages()`
  - 読み込み：`library()`
  - Rのメジャーバージョンアップ時は要注意。
  - オフライン環境ではminiCRANが有用（Dockerがダメな時）。

# ネットで公開されている便利なリソース

- RStudio公式チートシート

- RStudioのライブラリーのものもある。

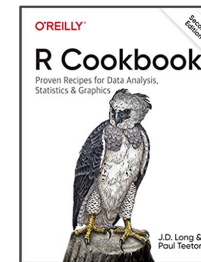
- <https://www.rstudio.com/resources/cheatsheets/>



- R Cookbook

- R Cookbookの著者によるwebサイト

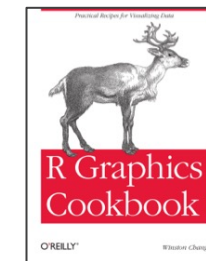
- <https://rc2e.com/graphics>



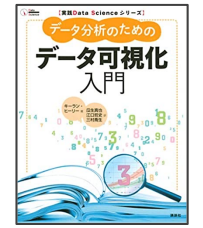
- Cookbook for R

- R Graphics Cookbookの著者によるwebサイト

- <http://www.cookbook-r.com/Graphs/>



# 参考書



- 『RユーザーのためのRStudio入門』（松村、湯谷、紀ノ定、前田 2021年）
  - いわゆる「**宇宙船本**」
- 『Rクックブック』第2版（Long, Teetor、2020年）
- 『Rではじめるデータサイエンス』（Wickham, Grolemund、2017年）
- 『Rグラフィックスクックブック』第2版（Chang、2019年）
- 『データ分析のためのデータ可視化入門』（Healy、2021年）

**Enjoy!**