Julia をインストール：改訂版

–定量的マクロ経済学の数値計算手法と応用–

山田 知明

明治大学
tyamada@meiji.ac.jp

2021 年 10 月 13 日改訂版
オリジナル：2021 年 5 月 15 日@日本経済学会

このスライドについて

- このスライドは 2021 年 5 月 15 日に日本経済学会で報告した「定量的マクロ経済学の数値計算手法と応用」というチュートリアルセッション用に作成した Julia インストールマニュアルの改訂版です
- 以下の点を書き換えています
  - JuliaPro が使えなくなったので代わりに Visual Studio Code から Julia を使用する方法を説明しています
- スライド内の下線がある箇所はハイパーリンクになっています
  - PDF 閲覧ソフトによっては機能しない可能性があります
- コードはGitHubに置いてあります
- 情報は 2021 年 10 月時点
- Mac 版を使っているため、Windows 版については若干の翻訳が必要かも

1. インストール方法

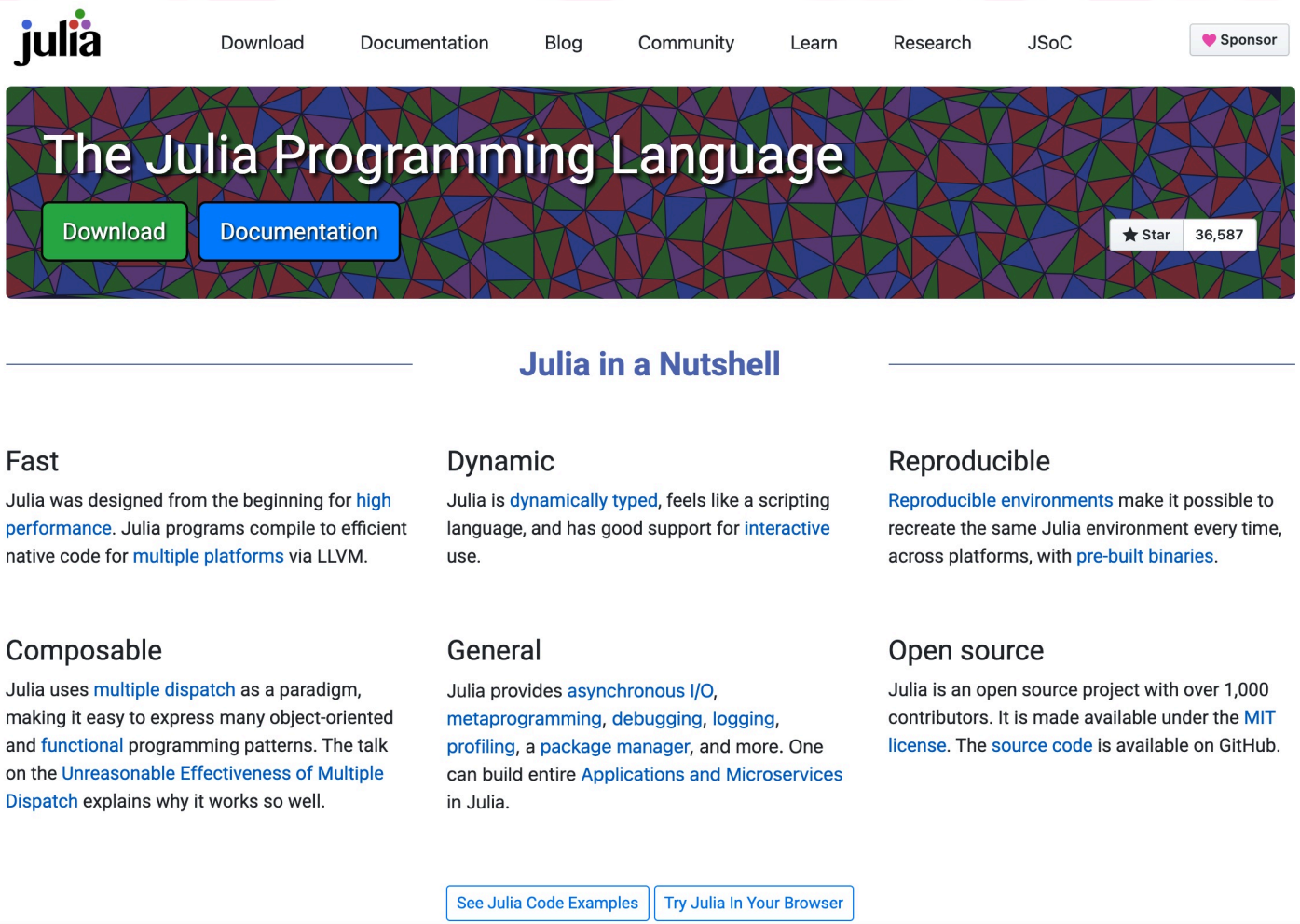
Julia を使う

- なぜ Julia?
  1. フリーで使える
  2. 学習しやすく、書きやすい言語
  3. 高速：公式のベンチマーク
     - ただし、高速化させる方法 (コードの書き方) を理解する必要あり
     - ⇑ 初心者は最初は細かいことを気にする必要なし：後述
  4. 近年、開発が盛ん
     - パッケージも増えてきた
     - チュートリアル等も充実：Youtube チャンネル
  5. VS Code Extension や Jupyter 等、開発環境周りも充実してきた
- 注意：チュートリアルセッションでは Julia(と Fortran) を使っていますが、既に使い慣れた環境があるのであればそちらで OK
  - Matlab、Python、R など

Julia をインストール

- Julia をインストールする方法
  - 公式 HPから Download を選択して自分の環境 (Windows、Mac、Linux) にあったバージョンを選ぶ
    - 色々と書いてあるけど基本的に Current stable release で OK
    - スライド作成時の最新版は v.1.6.3 (Sep 23, 2021)
  - インストーラの指示通りにインストールすれば終了
    - フォルダ名やパスが長すぎるという警告が出るかも
    - その場合は適宜、Julia 用フォルダを別の場所に作る
- Matlab と比較した Julia の問題点
  - 実際に数値計算を始めるまでの敷居がやや高い
    - Matlab はインストールすればすぐに動くけど、Julia はひと手間必要
    - 加えてコマンドラインが怖い...
  - 似たようなことは Python や LaTeX にもいえる
    - R は RStudio のおかげで比較的ラクそう

Julia 公式 HP



- 左上の Download を選択

バージョンを選択

Download Julia

If you like Julia, please consider starring us on GitHub and spreading the word!

Star 36,587

We provide several ways for you to run Julia:

- In the terminal using the built-in Julia command line using the binaries provided below.
- Using Docker images from Docker Hub maintained by the Docker Community.

Please see platform specific instructions for further installation instructions and if you have trouble installing Julia. If the provided download files do not work for you, please file an issue in the Julia project. Different OSes and architectures have varying tiers of support, and are listed at the bottom of this page.

Almost everyone should be downloading and using the latest stable release of Julia. Great care is taken not to break compatibility with older Julia versions, so older code should continue to work with the latest stable Julia release. You should *only* be using the long-term support (LTS) version of Julia if you work at an organization where implementing or certifying upgrades is prohibitively expensive and there is no need for new language features or packages. See this description of "Risk Personas" for more detail on who should be using what versions of Julia based on their risk tolerance.

Note: Julia comes with a built-in package manager which downloads and installs packages from the Internet. In doing so, it necessarily reveals your public IP address to any server you connect to, and service providers may log your IP address. In Julia versions 1.5 and higher, by default the package manager connects to https://pkg.julialang.org, a free public service operated by the Julia project to serve open source package resources to Julia users. This service retains IP address logs for up to 30 days.

Current stable release: v1.6.3 (Sep 23, 2021)

Checksums for this release are available in both MD5 and SHA256 formats.

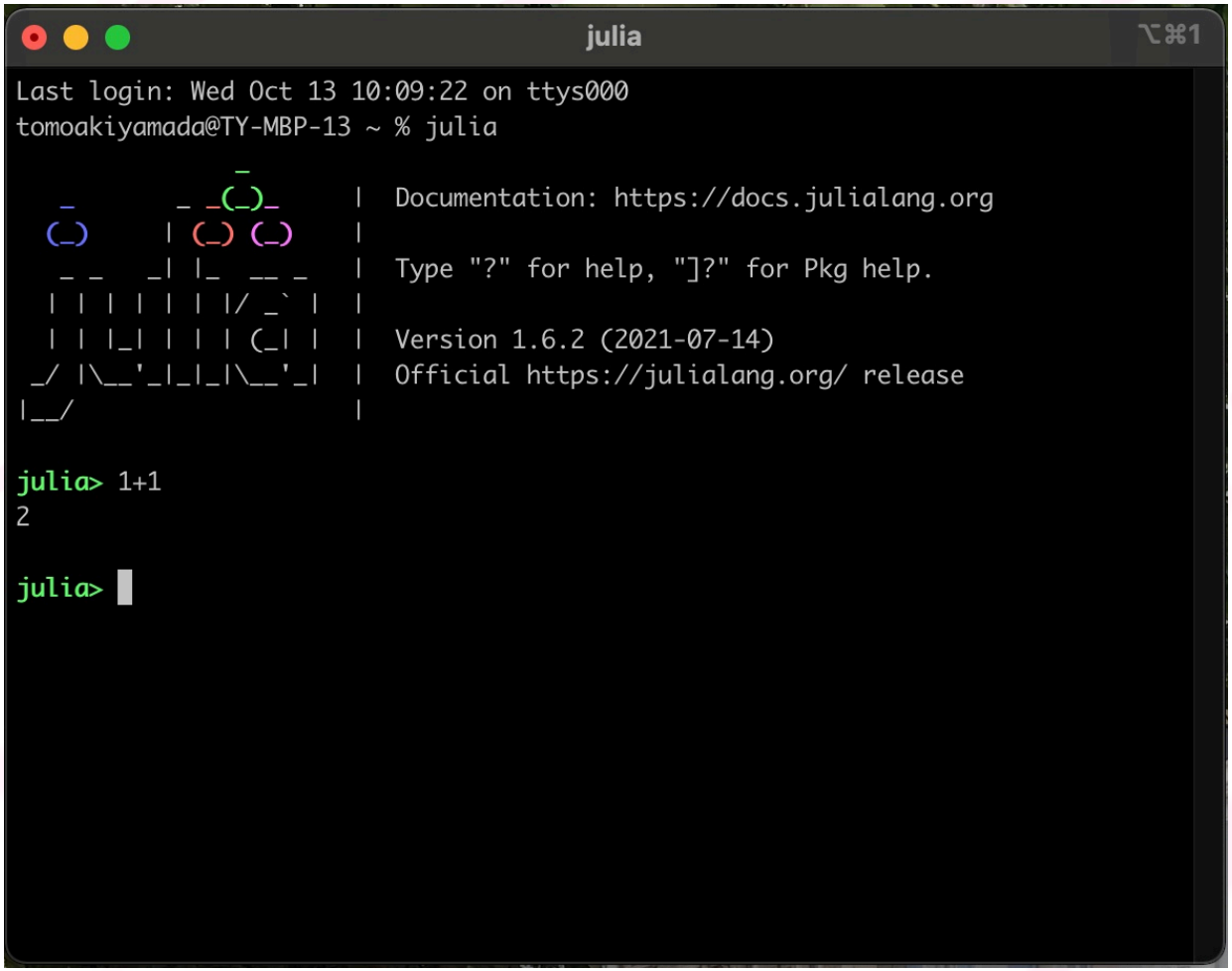
Windows [help]	64-bit (installer), 64-bit (portable)	32-bit (installer), 32-bit (portable)
macOS [help]	64-bit	
Generic Linux on x86 [help]	64-bit (GPG), 64-bit (musl)[1] (GPG)	32-bit (GPG)
Generic Linux on ARM [help]	64-bit (AArch64) (GPG)	32-bit (ARMv7-a hard float) (GPG)

- 自分の環境にあったインストーラをダウンロード

Julia を実用的に使う (続き)

- Julia 本体をインストールしただけでは実用的ではない
  - REPL(黒い窓) だけで数値計算をするのは不可能：次スライド
- 開発環境の整備が必要
  1. コードを書く際にシンタックスをハイライトしてくれるエディタ
     - Visual Studio Code、Sublime Text、Atom 等
  2. 統合環境 (Integrated Development Environment: IDE)
     - JuliaPro：更新を停止したみたい...
     - VS Code を IDE 化する：後述
  3. パッケージの追加も必要：後述
     - 極値 (maximum、minimum) や根 (root) を探すためのパッケージ
     - データを扱うのであれば DataFrame や統計パッケージ
     - 図を描くためのパッケージ
     - ect.

REPL



- REPL(Read-Eval-Print Loop)

Julia を実用的に使う (続き)

- 2021 年 10 月時点で “一つインストールをするだけで全てが揃う” 便利なモノはない
  $\Rightarrow$ テキストエディタに拡張機能 (Extension) を追加して使う
- 以下では VS Code に Julia Extension を追加して使う方法を説明
- Julia Extension が利用できるエディタは VS Code 以外にもある
  - Sublime Text 向け Extension
  - JetBrains 向け Extension
    - Python だと PyCharm を使っている人用
  - Emacs 向け Extension
  - Vim 向け Extension
- 自分で使っていないので設定方法や使い勝手はわかりません
  - Emacs や Vim 使いの人なら自力でなんとか出来るはず!

VS Code + Julia

VS Code + Julia

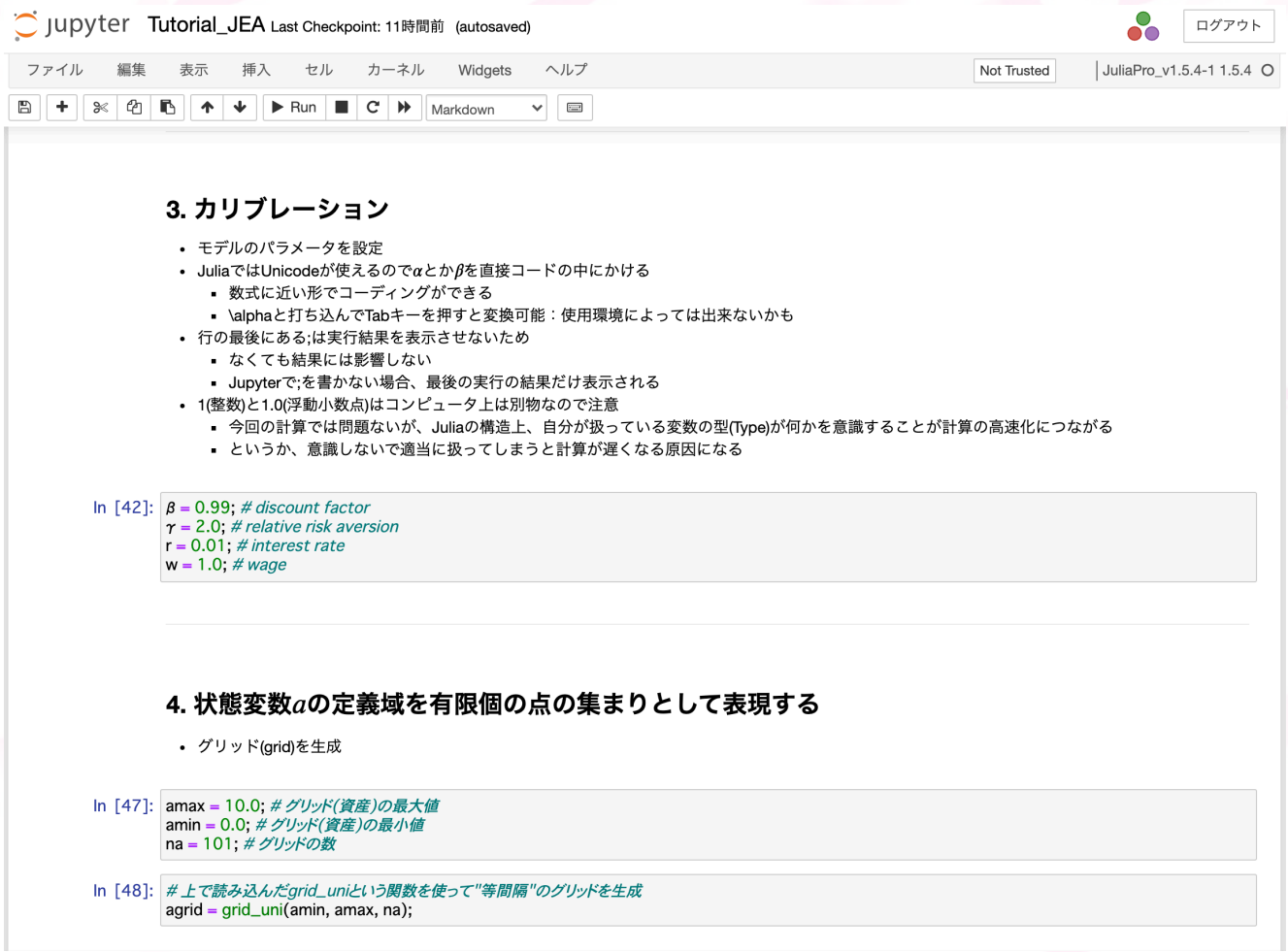
- Julia を Matlab ライクに使いたい
  - コーディングするスペース (もちろんシンタックスをハイライト)
  - ターミナルから実行
  - 変数を表示してくれるペイン、図の表示、デバッグ etc.
  - R だと RStudio みたいな感じ
- Visual Studio Code に Julia 拡張を追加して使う
  - VS Code 自体、とても優れたテキストエディタなのでお勧め!
    - LaTeX もこれで OK
    - Git との連携も良好
  - インストール方法
    - Julia と VS Code をインストール後、VS Code の拡張機能から Julia 拡張をインストール
    - Path を設定：Executable Path に自分の Julia をインストールした場所を設定

Jupyter

Jupyter をインストール

- Jupyter Notebook とは?
  - ブラウザ上で動く統合環境：Julia、Python、R など
  - もともと Python 用だったけど汎用的に拡張
- Julia で Jupyter Notebook を使うには?
  - IJulia というパッケージをインストール
  - パッケージのインストール方法は後述

Jupyter Notebook

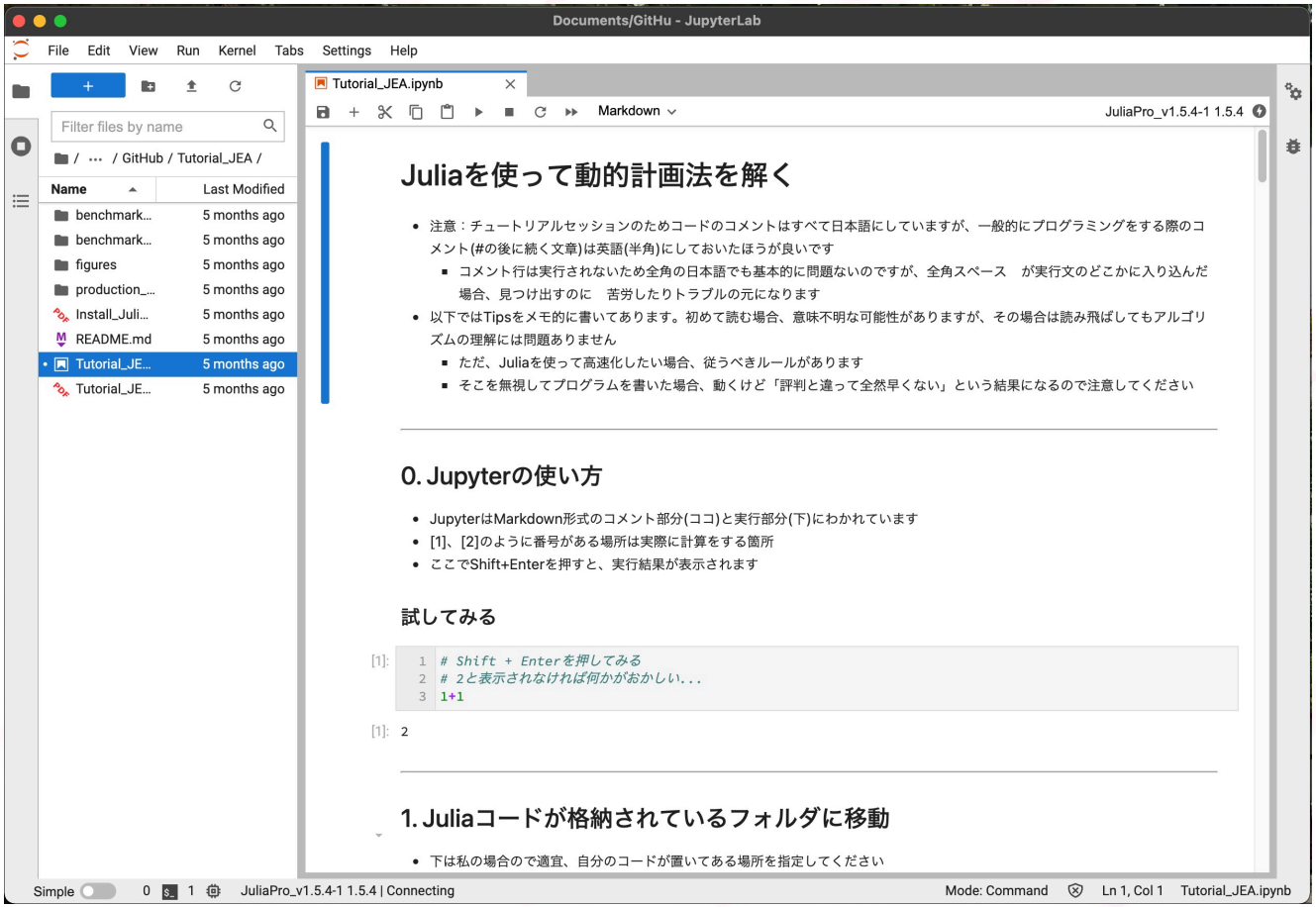


- こんな感じ

JupyterLab

- Jupyter Notebook はターミナルから呼び出してウェブブラウザ上で使うもの
- JupyterLab のアプリ版がリリース
  - ウェブブラウザ経由ではなく直接呼び出して使える
  - JupyterLab App
- JupyterLab
  - Jupyter Notebook の後継
  - App 版ではなく Notebook と同様にウェブブラウザから呼び出すことも可能

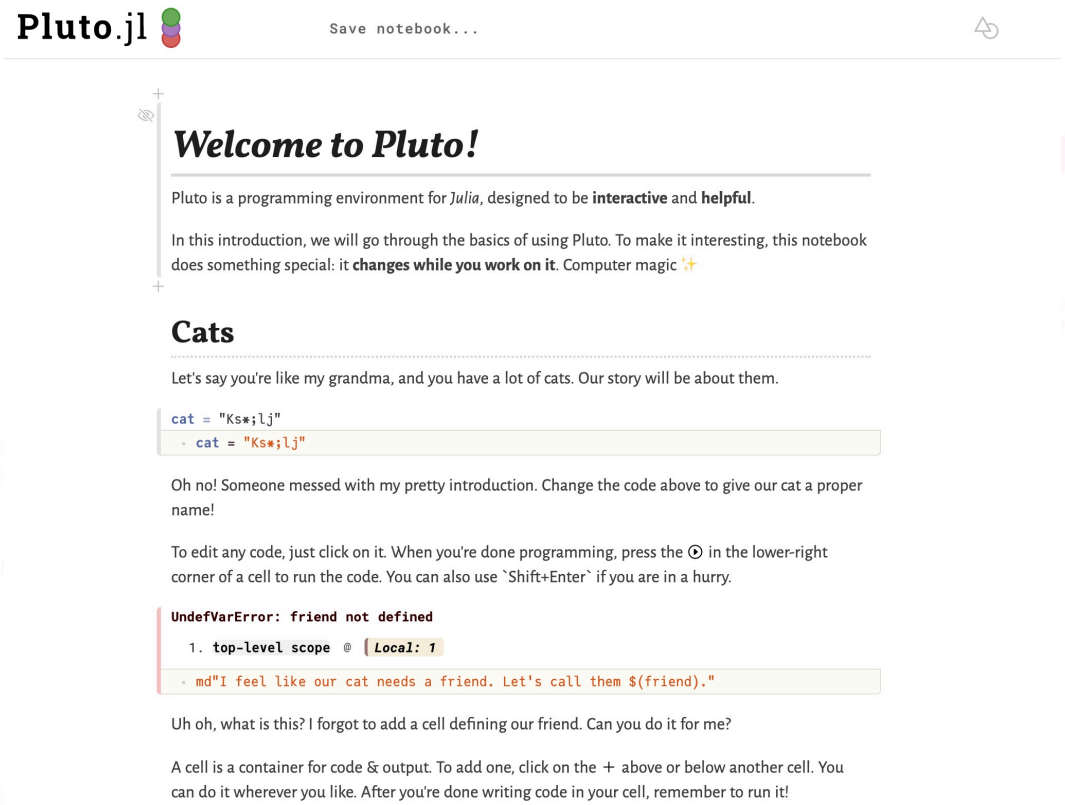
JupyterLab App



- JupyterLab App はこんな感じ

Pluto

- Pluto という Julia オリジナルのパッケージもあり
  - 開発中でまだ実用性は低め



パッケージ

Julia にパッケージをインストール

- パッケージ：Julia に必要な機能を追加
  - 非線形方程式の根を計算したい
  - キレイな図を描きたい
  - 統計データを扱いたい
  - などなど
- チュートリアルセッションで使ったコードを動かすために必要なパッケージ
  - `Dierckx`
  - `Interpolations`
  - `Optim`
  - `Printf`
  - `Plots`

Packages をインストール

```
using Pkg

Pkg.add("Dierckx")
Pkg.add("IJulia")
Pkg.add("Interpolations")
Pkg.add("Optim")
Pkg.add("Plots")
```

- パッケージをインストールする方法
- REPL で] を押してパッケージモードにしてから `add IJulia` でも一緒

Tips

Julia を高速で動かすためのすごく大雑把な知識

- 注意：専門家ではないので不正確だったり間違っている箇所があるかもしれない点、ご了承ください
- 以下は大雑把なイメージです
- それでも、知っているのと知らないのとでは Julia を使っていく上で差が出るので一応、メモ書き程度にまとめました
- 今後、研究で Julia を使っていこうと思ったら公式マニュアルか本などを一通り目を通したほうが良いと思います
  - 「Matlab でこう書いていたのを Julia だとどう書くんだろう」みたいな対応関係の検索だけだと Julia の機能を十分に使いこなせない可能性あり
- 公式の Tips をしっかり読む!

Julia を高速で動かすためのすごく大雑把な知識 (続き)

- 「Julia は Matlab や R より早い」と聞いていたのに体感できない
  - なぜ??
- C や Fortran は静的型付け (static typing) という仕組みで、変数の型 (整数とか浮動小数点) を事前に固定した上でコンパイル (機械語に翻訳) する
  - 事前に自分自身で変数の型を宣言する必要あり
  - コンピュータは型に沿った形で最適化
- Matlab や Python、R などは動的型付け (dynamic typing) という仕組みで変数の型にうるさくない
  - プログラミングの際に細かいことを気にしなくても良い
  - コンピュータはその都度変数を機械語に解釈するので遅い
    - 最適化が出来ない (しにくい?)

Julia を高速で動かすためのすごく大雑把な知識 (続き)

- Julia はどっち?
- 基本的に動的型付け
  - Matlab とか R みたいに変数の型を気にしないで書いて動く
  - ただし、その場合は計算速度が遅い可能性あり
- Just-in-time コンパイルと LLVM という仕組みで高速化
  - 細かいことは参考文献を読んでください
  - 重要：Julia は型を意識することで高速化できる
  - 他にも、「一つの長いコードを書かないで小分けに関数にまとめる」などの工夫が高速化のポイント
- Python や Matlab にも独自の高速化の方法がある
  - Cython、MEX など

Julia を高速で動かすためのすごく大雑把な知識 (続き)

- まとめ：Julia を本格的に研究で使うのであれば仕組みを理解する必要があるけど、まずは難しいことを考えずに、手を動かしてみるのが良いと思います!

Matlab から引っ越してきて戸惑うこと

- ローカル変数の範囲
- 関数の使い方：高速化には必須

参考文献

参考文献：Web 検索ではなく本で読みたい場合

- 進藤裕之・佐藤建太『1 から始める Julia プログラミング』コロナ社
- Kaminski and Szufel 著・中田秀基訳『Julia プログラミング クックブック』オライリー・ジャパン
  - 中級者向け
  - 英語版もあるけど短くまとまった日本語版補遺がとても良い
- Lauwens, B. and A.B. Downey (2019): "Think Julia How to Think Like a Computer Scientist," O'Reilly.
  - "Think Python" という有名な本の Julia 版
- Balbaert, I. and A.Salceanu (2019,May): "Learning Path Julia 1.0 Programming Complete Reference Guide," PACT.
- Sengupta, A. (2019,June): "Julia High Performance Second Edition," PACT.
  - 中級者向け