

JAEIS2018 日本情報科教育学会 第11回全国大会
企画セッション「プログラミング・ワークショップ」

「Pythonによる手書き文字認識の基礎」 のための事前準備

武蔵野大学 工学部 数理工学科 准教授
中西 崇文

世界の幸せをカタチにする。
Creating Peace & Happiness for the World



このワークショップで得られること

- 今話題の機械学習手法ディープラーニングの基本
(ニューラルネットワーク)を簡単に
プログラミングができる
- 手書きの数字を認識するプログラミングができる
- データサイエンティストの
デファクトスタンダードプログラミング言語
Pythonを触れることができる

準備すること(2つだけ)

1. Anacondaのインストール

– Anacondaをインストールすることで
インストールされるもの

- Python
- Numpyなどのデータ分析に必要な各種パッケージ
- Jupyter Notebook

2. Tensorflow、Kerasのインストール

Python



- 1990年初頭にオランダのガイド・ヴァンロッサムによって開発されたプログラミング言語
- シンプルに設計されており、少ないコード数で書ける
- AI、機械学習、Web開発など広い分野で使われている
- データサイエンティストのデファクトスタンダード言語

Tensorflow



TensorFlow

<https://www.tensorflow.org>

- Googleが開発して、2015年から世界にオープンソースで公開しているライブラリ
- 主に機械学習を構築するのに便利なライブラリ

Keras



<https://keras.io/ja/>

- Pythonで書かれた, TensorFlow上で実行可能な高水準のニューラルネットワークライブラリ
- ニューラルネットワークを簡単にプログラミングできるライブラリ

Numpy



- Pythonで数値計算を効率的に行うための拡張モジュール
- ベクトルや行列を表現できる型付きの多次元配列をサポートすると共にたくさんの数学関数ライブラリを提供する

Jupyter Notebook jupyter

<http://jupyter.org/index.html>

- ブラウザ上で動作するプログラムの対話型実行環境
- プログラムの記述と実行、メモの作成、保存と共有などをブラウザ上で行うことができる

動作環境概要

開発環境
実行環境



パッケージ
ライブラリ



.....

プログラミング
言語



世界の幸せをカタチにする。
Creating Peace & Happiness for the World!

① Anacondaをインストール

インストール手順

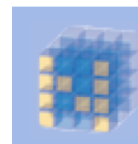
開発環境
実行環境



パッケージ
ライブラリ



TensorFlow



NumPy

.....

プログラミング
言語



② TensorflowとKerasをインストール

世界を平和と幸せにカタチにする。
Creating Peace & Happiness for the World!

Anacondaとは

- Continuum Analytics社が提供するPythonとcondaというパッケージマネージャとデータサイエンスに関する一連のパッケージを含むPythonディストリビューション
- Anacondaをインストールするだけで、Pythonとデータサイエンティストがよく使うパッケージ、ライブラリが一度に導入できる

Anacondaのインストール[1/9]

※WindowsもMacもだいたい手順は同じです

- <https://www.anaconda.com/download/> にブラウザでアクセス
- Python 3.6 version * Downloadをクリックし、インストーラーをダウンロード

Documentation Blog Contact Anaconda Cloud

What is Anaconda? Products Support Resources About [Download](#)

Anaconda 5.2 For macOS Installer

Python 3.6 version *

[↓ Download](#)

[64-Bit Graphical Installer \(613 MB\)](#) ⓘ
[64-Bit Command-Line Installer \(523 MB\)](#) ⓘ

Python 2.7 version *

[↓ Download](#)

[64-Bit Graphical Installer \(617 MB\)](#) ⓘ
[64-Bit Command-Line Installer \(527 MB\)](#) ⓘ

[How to get Python 3.5 or other Python versions](#)
[How to Install ANACONDA](#)

Get Started

世界の幸せをカタチにする。
Creating Peace & Happiness for the World!



Musashino University

Anacondaのインストール[2/9]

- ダウンロードしたインストーラーをダブルクリック



Anacondaのインストール[3/9]

- 基本「Next」、「続ける」「インストール」をクリックしていけばOK



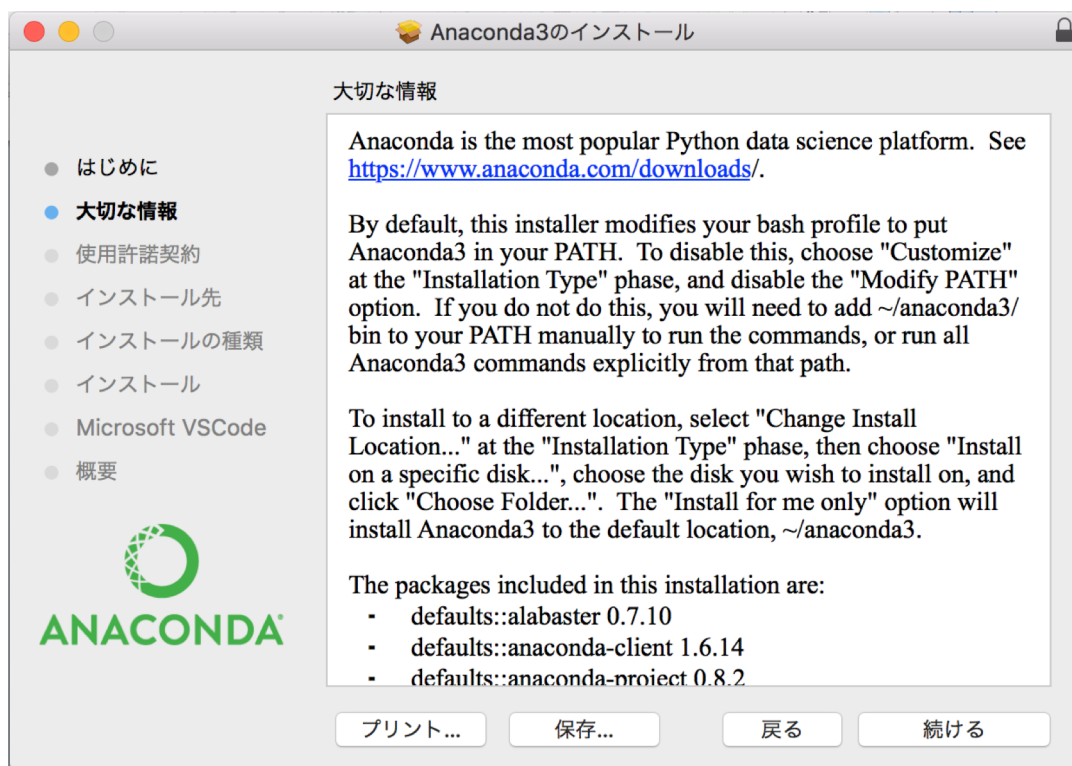
Anacondaのインストール[4/9]

- 基本「Next」、「続ける」「インストール」をクリックしていけばOK



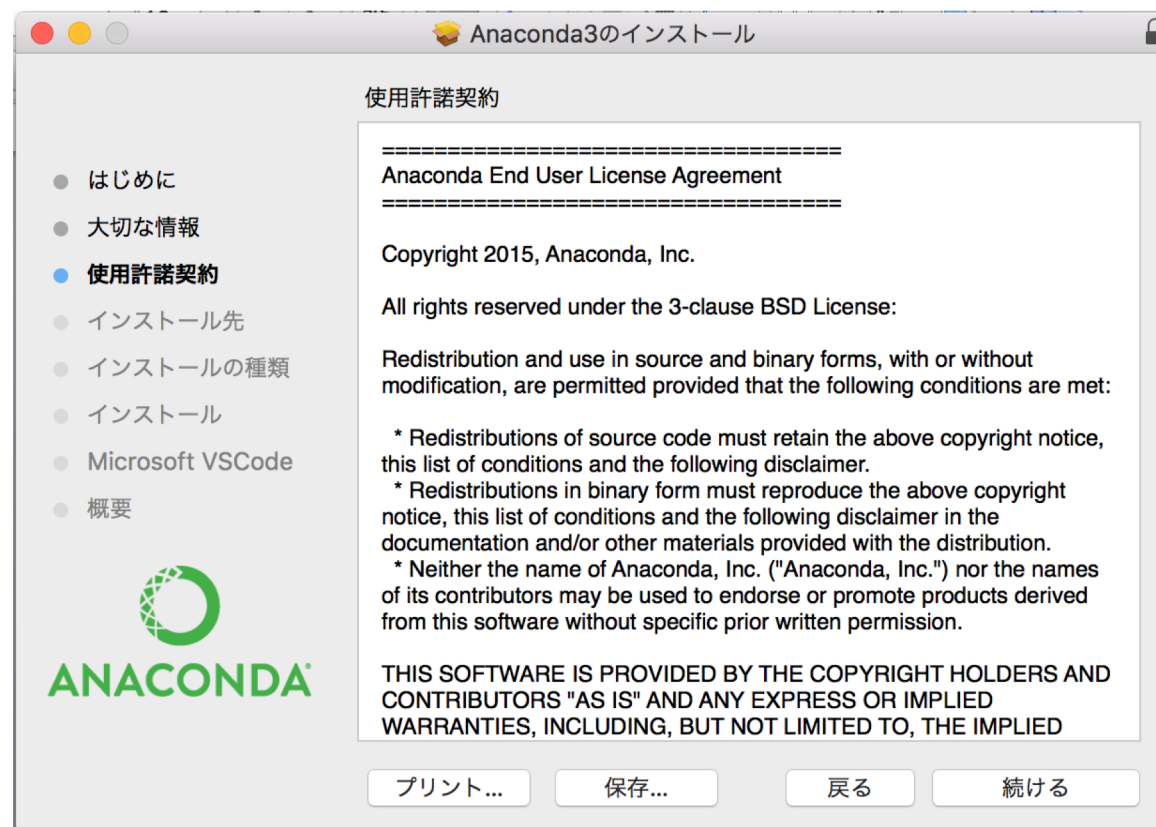
Anacondaのインストール[5/9]

- 基本「Next」、「続ける」「インストール」をクリックしていけばOK



Anacondaのインストール[6/9]

- 基本「Next」、「続ける」「インストール」をクリックしていけばOK



Anacondaのインストール[7/9]

- 基本「I Agree」もしくは「同意する」をクリックしていけばOK



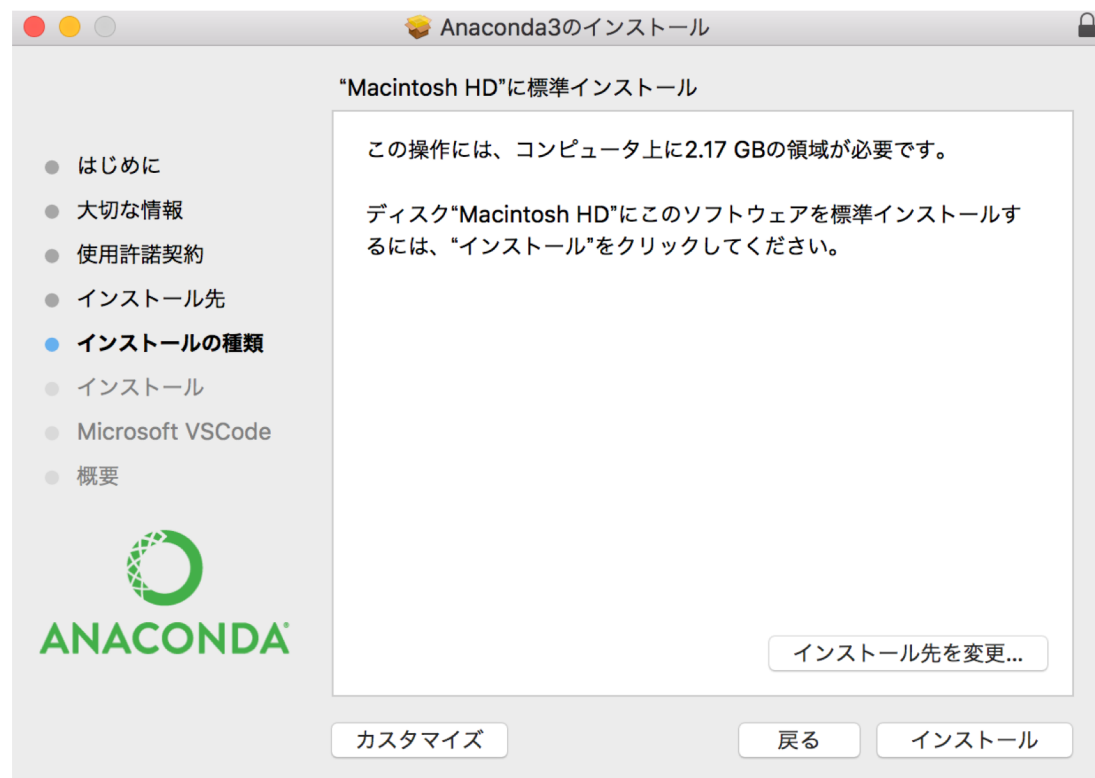
Anacondaのインストール[8/9]

- 基本「Next」、「続ける」「インストール」をクリックしていけばOK



Anacondaのインストール[9/9]

- 「インストール」をクリックすればインストールされる



Microsoft VSCodeはインストールしてもしなくても良い

世界の幸せをカタチにする。
Creating Peace & Happiness for the World



Musashino University

Tensorflow, Kerasのインストール

- 「コマンドプロンプト」(Windows)、
「ターミナル」(Mac)を開く
- 「pip install tensorflow keras」と入力
 - エラーが出なければインストール完了

Jupyter Notebookを立ち上げる

- 「コマンドプロンプト」(Windows)、
「ターミナル」(Mac)を開く
- 「jupyter notebook」と入力
- ブラウザ上に立ち上がる

Jupyter Notebookの画面

Files **Running** Clusters

Select items to perform actions on them. Upload New ▾ ↻

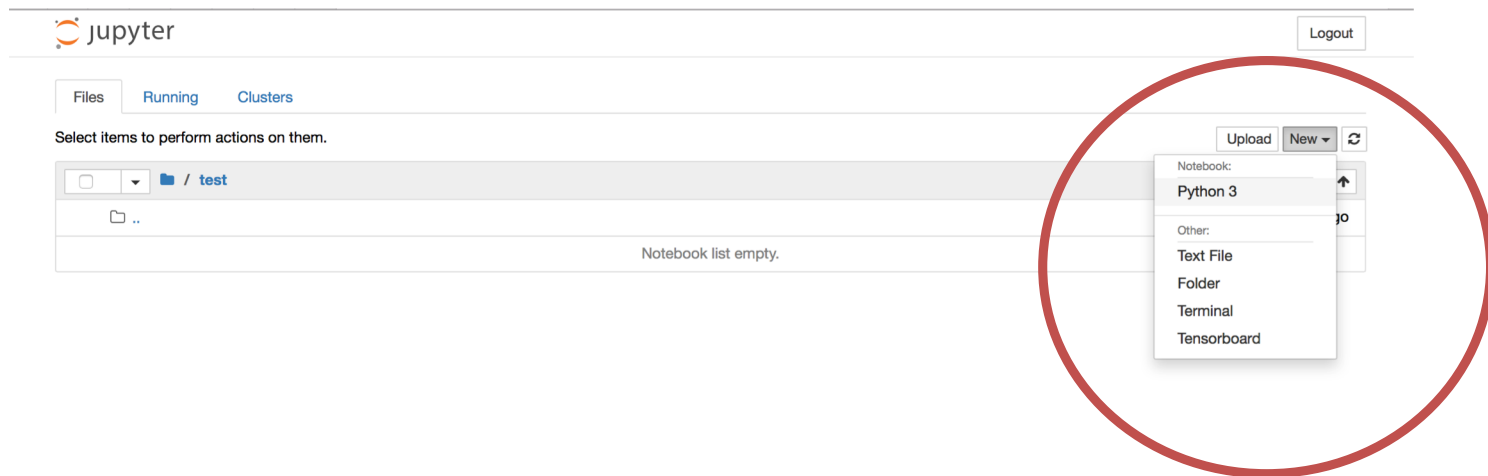
▾ / test Name ↑ Last Modified ↑

<input type="checkbox"/> ..	seconds ago
-----------------------------	-------------

Notebook list empty.

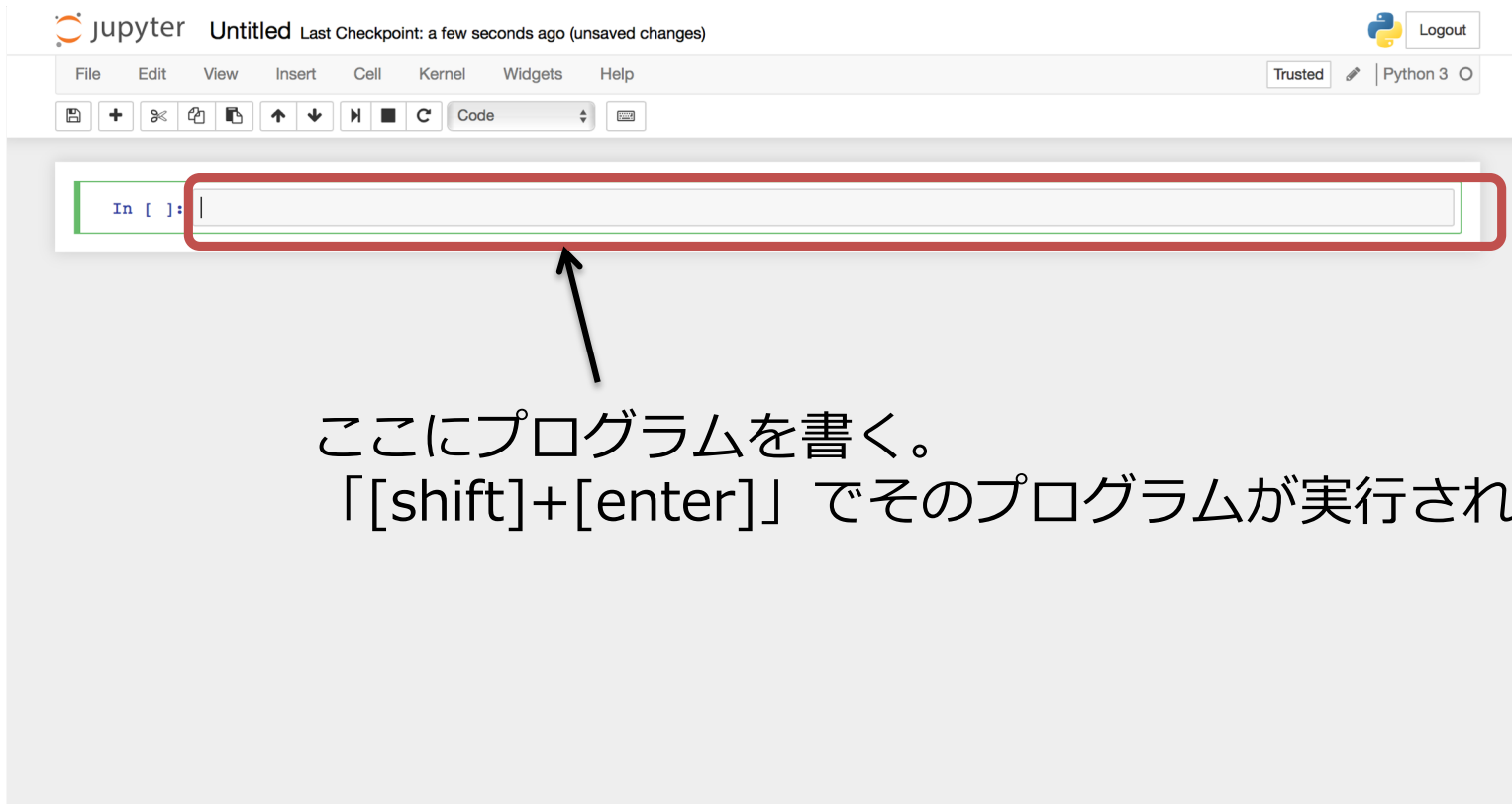
Jupyter Notebookを始める

- [New]をクリックし、[Python3]を選択



このページ内の#localhost:8888/tree/test#を新規タブで開く

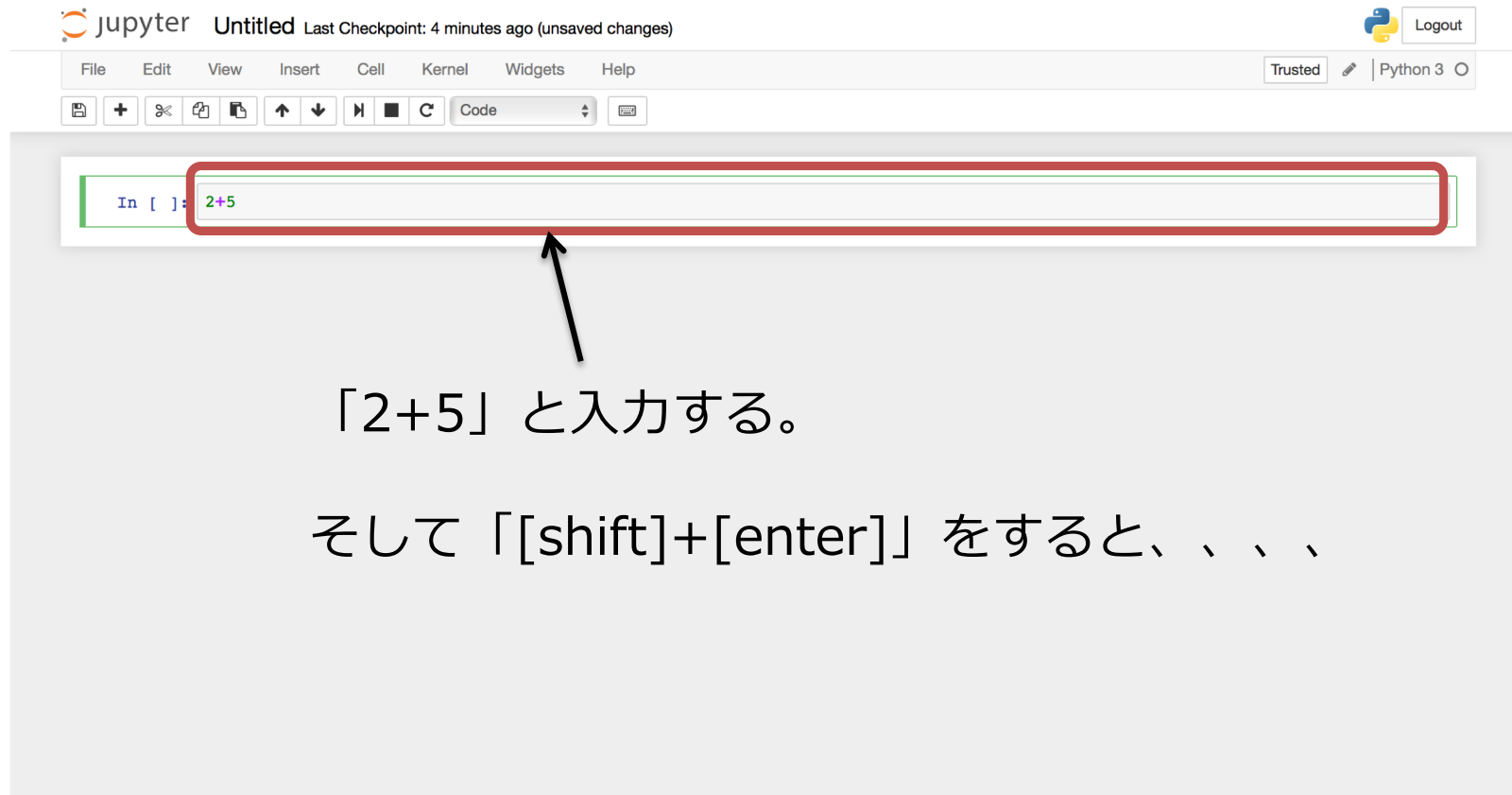
Jupyter Notebookの基本操作



The screenshot shows the Jupyter Notebook interface. At the top, there is a header with the Jupyter logo, the text "jupyter Untitled", and "Last Checkpoint: a few seconds ago (unsaved changes)". On the right side of the header, there is a "Logout" button. Below the header is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", "Widgets", and "Help". To the right of the menu bar, there is a "Trusted" indicator, a pencil icon, and "Python 3". Below the menu bar is a toolbar with various icons for file operations and execution. The main area of the notebook is a code cell, which is highlighted with a red border. The code cell contains the text "In []:" followed by a cursor. An arrow points from the text below to the code cell.

ここにプログラムを書く。
「[shift]+[enter]」でそのプログラムが実行される。

Jupyter Notebookで簡単なプログラムを実行する-1[1/2]

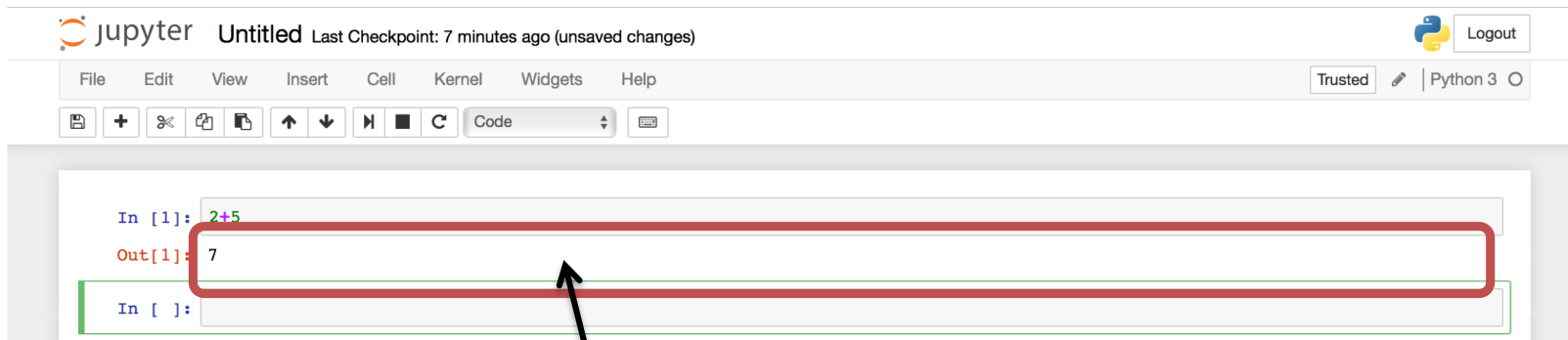


The screenshot shows the Jupyter Notebook interface. At the top, it says "jupyter Untitled Last Checkpoint: 4 minutes ago (unsaved changes)". Below that is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", "Widgets", and "Help". To the right of the menu bar, there is a "Trusted" status indicator, a Python logo, and "Python 3". Below the menu bar is a toolbar with various icons for file operations and execution. The main area shows a code cell with the text "In []: 2+5". A red rectangular box highlights the input area of the code cell. An arrow points from the text below to this red box.

「2+5」と入力する。

そして「[shift]+[enter]」をすると、、、

Jupyter Notebookで簡単なプログラムを実行する-1[2/2]

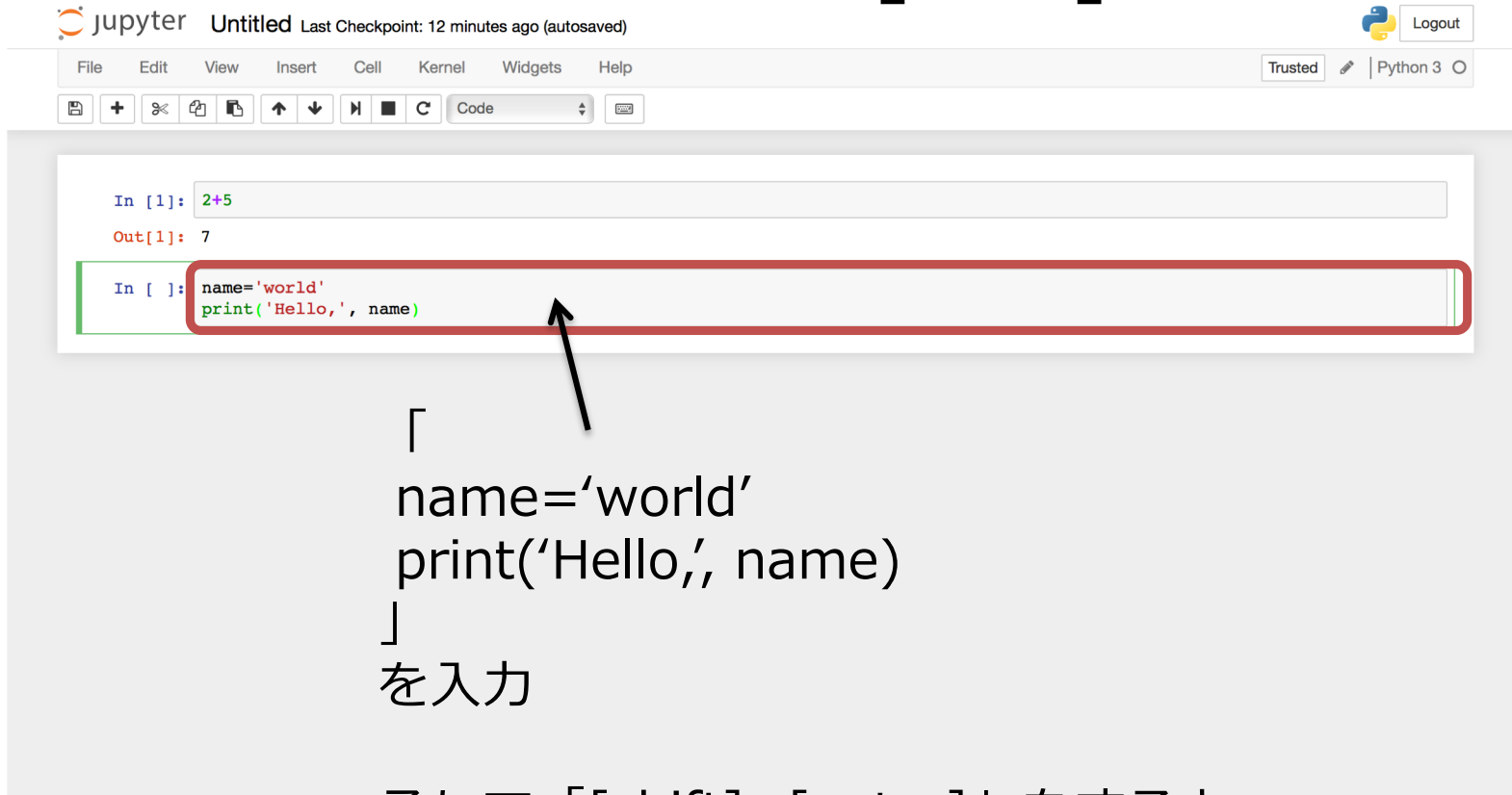


The screenshot shows a Jupyter Notebook interface. At the top, it says "jupyter Untitled Last Checkpoint: 7 minutes ago (unsaved changes)". Below that is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", "Widgets", and "Help". To the right of the menu bar are "Trusted", a pencil icon, and "Python 3". Below the menu bar is a toolbar with icons for saving, adding, undo, redo, and running code. The main area shows a code cell with "In [1]: 2+5" and an output cell with "Out[1]: 7". A red box highlights the output "7", and an arrow points to it.

「7」と出力される

正しくプログラムが実行された！

Jupyter Notebookで簡単なプログラムを実行する-2[1/2]



The screenshot shows a Jupyter Notebook interface. The top bar includes the Jupyter logo, the name 'Untitled', and the text 'Last Checkpoint: 12 minutes ago (autosaved)'. On the right, there is a 'Logout' button and a Python logo. Below the top bar is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. To the right of the menu bar, it says 'Trusted' and 'Python 3'. Below the menu bar is a toolbar with various icons for file operations and execution. The main area contains a code cell with the following content:

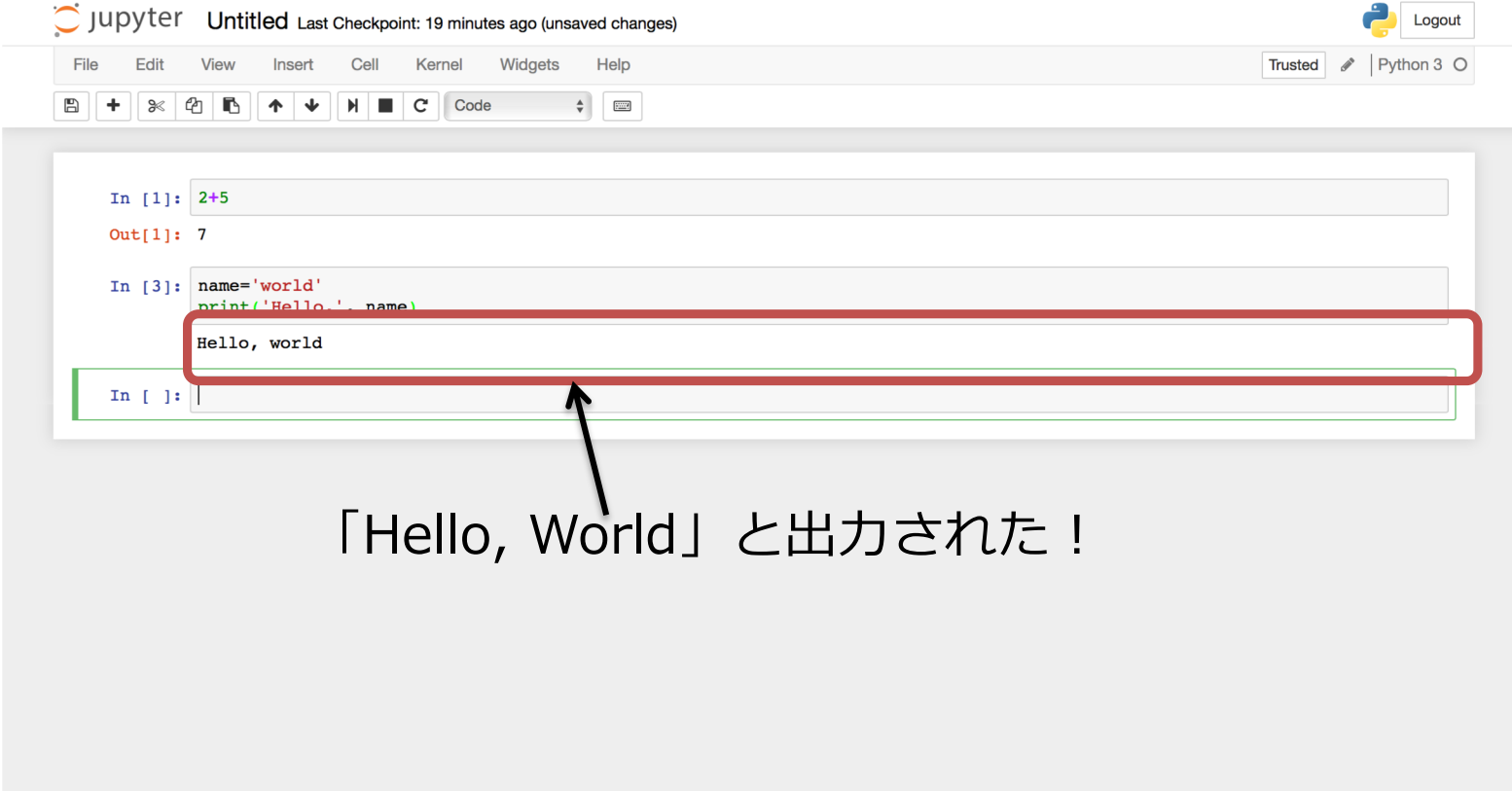
```
In [1]: 2+5
Out[1]: 7
In [ ]: name='world'
        print('Hello,', name)
```

The code cell is highlighted with a red border. An arrow points from the text below to the code cell.

「
name='world'
print('Hello,', name)
」
を入力

そして「[shift]+[enter]」をすると、、、

Jupyter Notebookで簡単なプログラムを実行する-2[2/2]



The screenshot shows a Jupyter Notebook interface. At the top, it says "jupyter Untitled Last Checkpoint: 19 minutes ago (unsaved changes)" and "Logout". Below that is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", "Widgets", and "Help". There are also buttons for "Trusted" and "Python 3". The main area contains three code cells. The first cell has the code `2+5` and its output is `7`. The second cell has the code `name='world'` and `print('Hello.', name)`, and its output is `Hello, world`. The third cell is empty with the code `In []:`. A red box highlights the output of the second cell, and an arrow points to it from the text below.

```
In [1]: 2+5
Out[1]: 7

In [3]: name='world'
        print('Hello.', name)
Hello, world

In [ ]:
```

「Hello, World」と出力された！

Numpyでベクトル計算

Jupyter Untitled Last Checkpoint: 25 minutes ago (unsaved changes) Python 3 Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

In [14]: `import numpy as np
a = np.array([1, 2])
b = np.array([3, 4])`

Numpyを呼び出して、
a=[1,2]、b=[3,4]というベクトルを作る

In [9]: `a+b` ベクトルの足し算
Out[9]: `array([4, 6])`

In [10]: `a-b` ベクトルの引き算
Out[10]: `array([-2, -2])`

In [11]: `a*b` ベクトルの掛け算
Out[11]: `array([3, 8])`

In [12]: `a/b` ベクトルの割り算
Out[12]: `array([0.33333333, 0.5])`

In [13]: `np.dot(a,b)` 内積
Out[13]: `11`

In []:

Numpyで行列計算

jupyter Untitled Last Checkpoint: 33 minutes ago (unsaved changes) Python 3 Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

In [18]: `c=np.array([[1,2],[3,4]])`
`d=np.array([[5,6],[7,8]])` $c = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, d = \begin{bmatrix} 5 & 6 \\ 7 & 7 \end{bmatrix}$ という行列を作る

In [19]: `c+d` 行列の足し算
Out[19]: `array([[6, 8],
[10, 12]])`

In [20]: `c-d` 行列の引き算
Out[20]: `array([[-4, -4],
[-4, -4]])`

In [21]: `c*d` 行列の掛け算
Out[21]: `array([[5, 12],
[21, 32]])`

In [22]: `c/d` 行列の割り算
Out[22]: `array([[0.2, 0.33333333],
[0.42857143, 0.5]])`

In [23]: `np.dot(c,d)` 内積
Out[23]: `array([[19, 22],
[43, 50]])`

Numpyの特別な演算

$c = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ という行列を作る

```
In [25]: c=np.array([[1,2],[3,4]])
```

```
In [26]: c+5
```

```
Out[26]: array([[6, 7],  
               [8, 9]])
```

全ての要素に5が足されている

```
In [27]: c-5
```

```
Out[27]: array([[ -4,  -3],  
               [ -2,  -1]])
```

全ての要素に5が引かれている

```
In [ ]:
```

Tensorflow、Kerasが正しくインストールされているか確かめる

- 「import keras」と入力して「[shift]+[enter]」
- 「Using TensorFlow backend」とのみ表示されたら、正しくインストールされている

```
In [28]: import keras
```

```
Using TensorFlow backend.
```

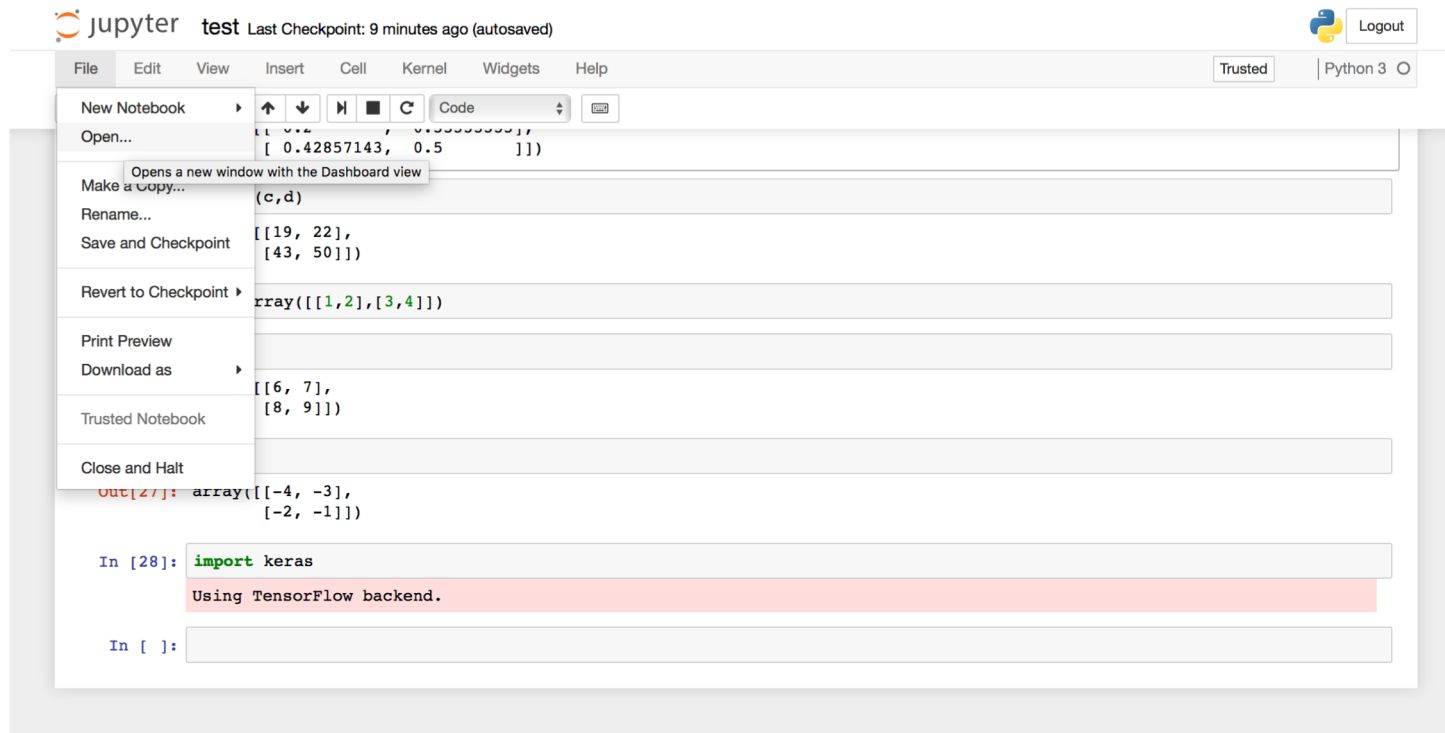
```
In [ ]:
```

ipynb形式ファイルを読み込む[1/3]

- Python Notebookでのプログラム、実行結果は、.ipynb形式ファイルで保持される
- .ipynb形式ファイルを読み込めば、前回の作業から進めることができる


ipynb形式ファイルを読み込む[2/3]

- 「File」 → 「Open」 を選択



ipynb形式ファイルを読み込む[3/3]

ここをクリックすることでディレクトリ(フォルダ)を指定



Select items to perform actions on them. Upload New ▾ ↻

<input type="checkbox"/>	▼	test	Name ↑	Last Modified ↑
<input type="checkbox"/>	📁	..		seconds ago
<input type="checkbox"/>	📄	test.ipynb		Running 10 minutes ago

該当の.ipynbファイルをクリック

講演者プロフィール

- 中西崇文(Takafumi Nakanishi)
- 武蔵野大学 工学部 准教授
- 国際大学グローバル・コミュニケーション・センター(GLOCOM) 主任研究員
- デジタルハリウッド大学大学院 客員教授
- 博士(工学)。
 - 1978年12月28日生まれ、三重県伊勢市出身。
 - 2006年3月、筑波大学大学院システム情報工学研究科にて博士(工学)の学位取得。
 - 2006年4月、情報通信研究機構(NICT) 研究員。
 - 2014年4月、国際大学グローバル・コミュニケーション・センター(GLOCOM) 准教授・主任研究員
 - 2018年4月、武蔵野大学 工学部 数理工学科 准教授
- 専門
 - データマイニング、データ分析システム、統合データベース、感性情報処理、メディアコンテンツ分析。近年は、ビッグデータ分析手法を通じたデータ分析工学分野の創出に興味を持つ。
 - 総務省「AIネットワーク社会推進会議」構成員
経済産業省「流通・物流分野における情報の利活用に関する研究会」委員
総務省「ICTインテリジェント化影響評価検討会議」構成員
等歴任。
- Line ID: piano_lullaby
- Twitter: @piano_lullaby
- Facebook: <http://facebook.com/pianolullaby>

