

# パターン認識特論 1

## 環境構築マニュアル

GPUを用いない環境構築  
Windowsを中心に説明しますがMacでも可  
**資料に表示されるバージョンと  
実際は異なる場合があります**

anaconda  
python 3.6/3.7  
tensorflow1.15.x + Keras2.x  
Jupyter

2020年1月15日版 / 2020年10月25日修正

作成：原 武史（岐阜大学）

## インストールの対象, 条件

Windows10がインストール済みのPC,  
もしくは, MacOS 10.15以上,  
メモリは8GB以上, 16GB以上を推奨,  
インターネット接続, 1GB以上のファイルのダウンロードが必要,  
管理者権限が必要になる場合がある, GPU利用の場合は別途相談.

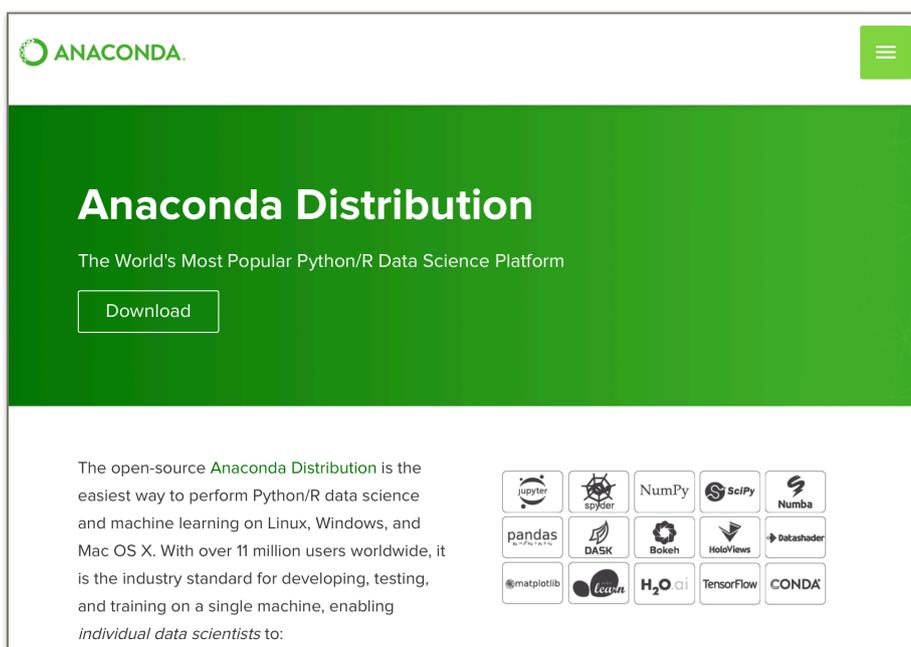
## 手順

インターネットへの接続  
WIFI環境でもよいので, ネットへ接続,  
大容量のファイルをダウンロードするので, 注意が必要,  
Anacondaは, 仮想的な実行環境を構築できます,  
そこで構築した環境に, 必要なライブラリをインストールします,  
Anacondaインストール (Step 1-13)/仮想環境構築 (Step 14-21)  
ライブラリインストール (Step 22-30)  
Jupyter Notebook (Step 31-35)/動作環境の確認 (Step 36-40)

## Step 1 Anacondaのダウンロード, ブラウザーからダウンロードします.

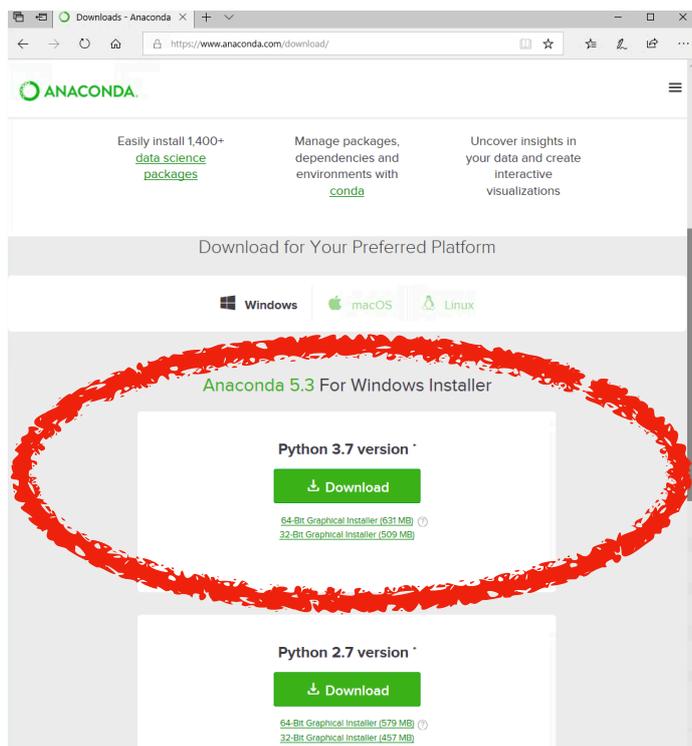
<https://www.anaconda.com/download>

へアクセスします.



## Step 2 ウェブページの下の方に移動します.

Windows Installer/Macが見えます.



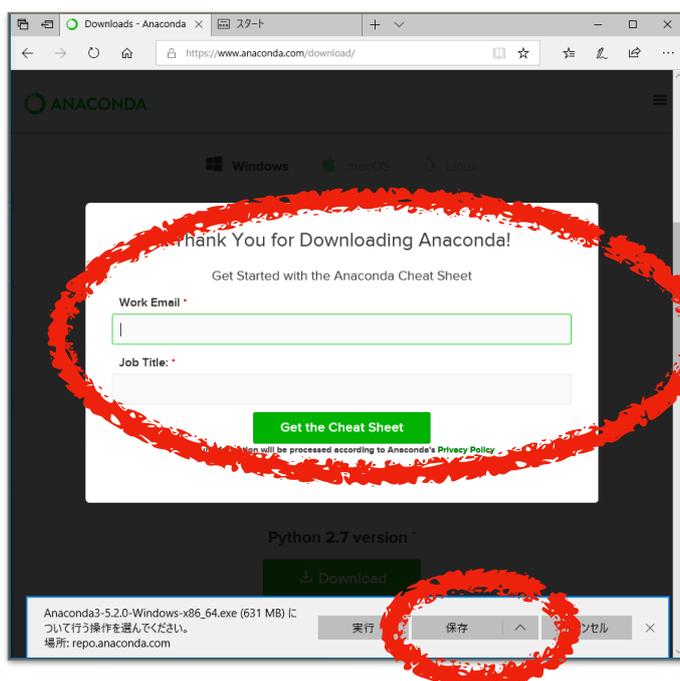
3. 7  
64ビット

**Step 3** Python 3.7 versionの中をみます。

64-Bit Graphical Installerをクリックします。

**Step 4** ダウンロードが始まります。

メールアドレスや所属などの登録を求められる場合があります。しかし、ダウンロードとは無関係です。興味ある人は登録を、始めると以下のような画面がでます。保存をクリックします。通信速度によっては結構時間がかかります。

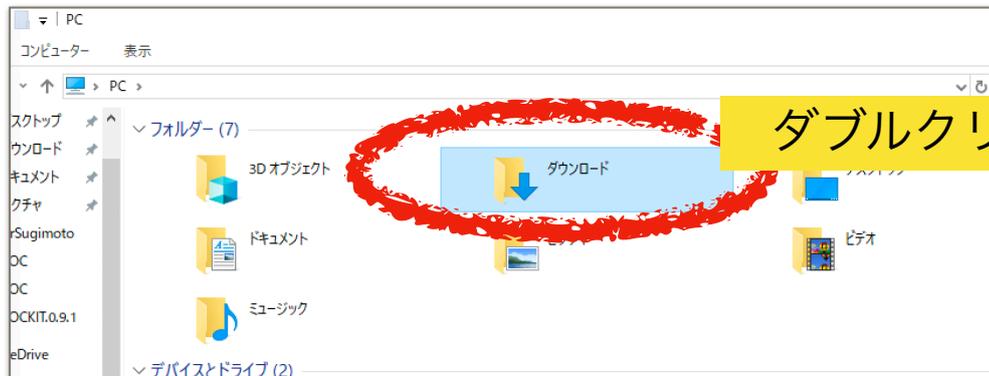


入力不要

保存をクリック

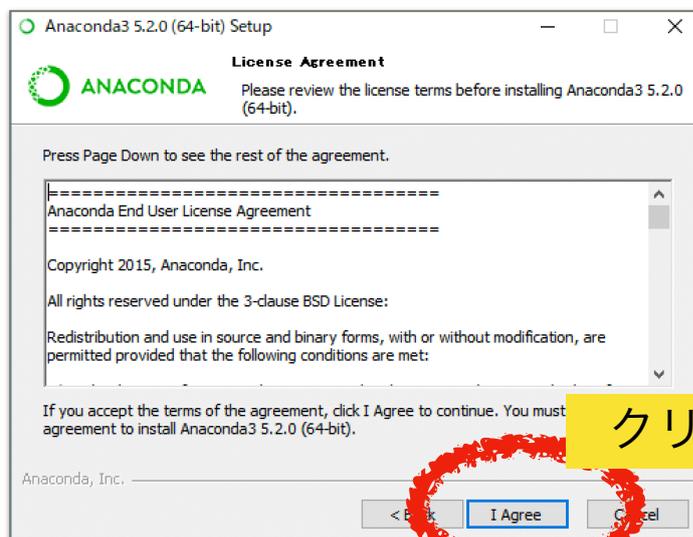
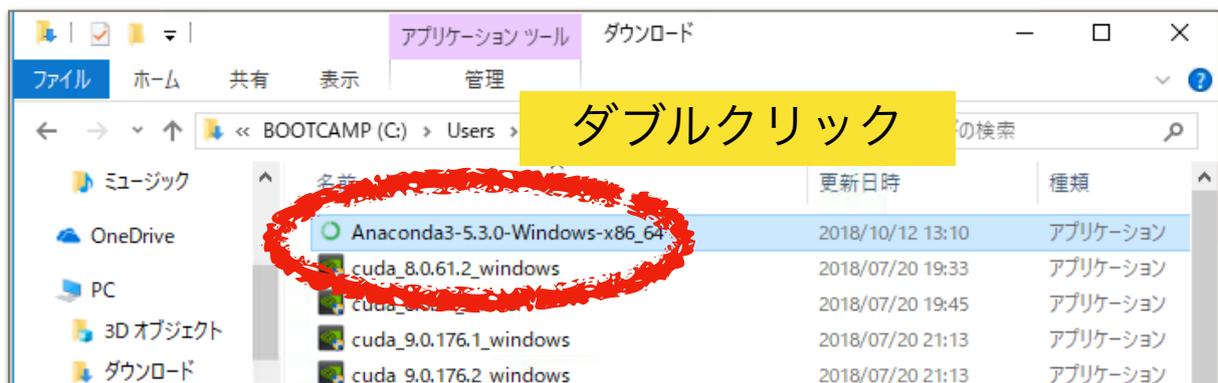
### Step 5 ダウンロードが終了したらインストール

PCの「ダウンロードフォルダー」に保存されます。  
ダウンロードフォルダーをダブルクリックします。



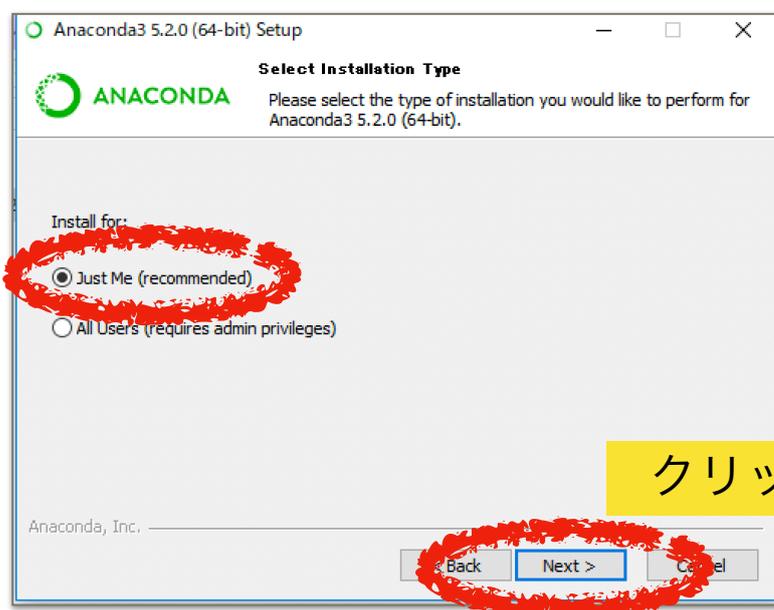
### Step 6 中が表示されて、ダウンロードされたファイルが見えます。

そのファイルがインストーラーです。  
そのファイルをさらにダブルクリックします。



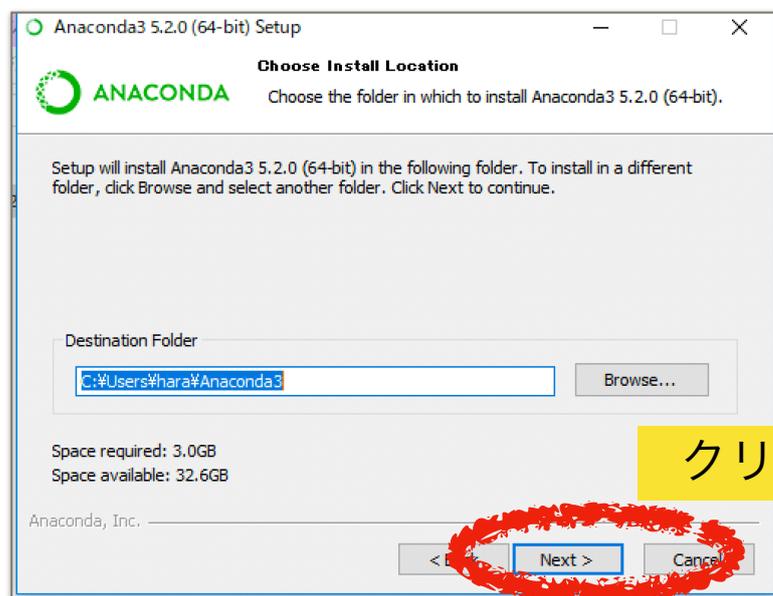
## Step 10 動作環境のパスの指定. これも言われるがまま.

これでファイルのコピー／展開によるインストールが始まります.



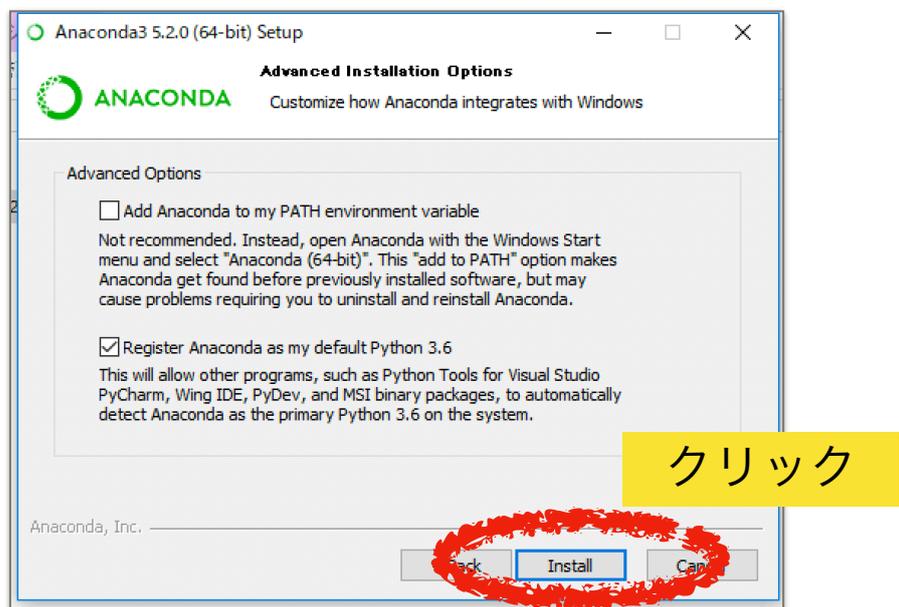
環境によっては20分くらいかかります.

## Step 11 インストールが一旦終了します.

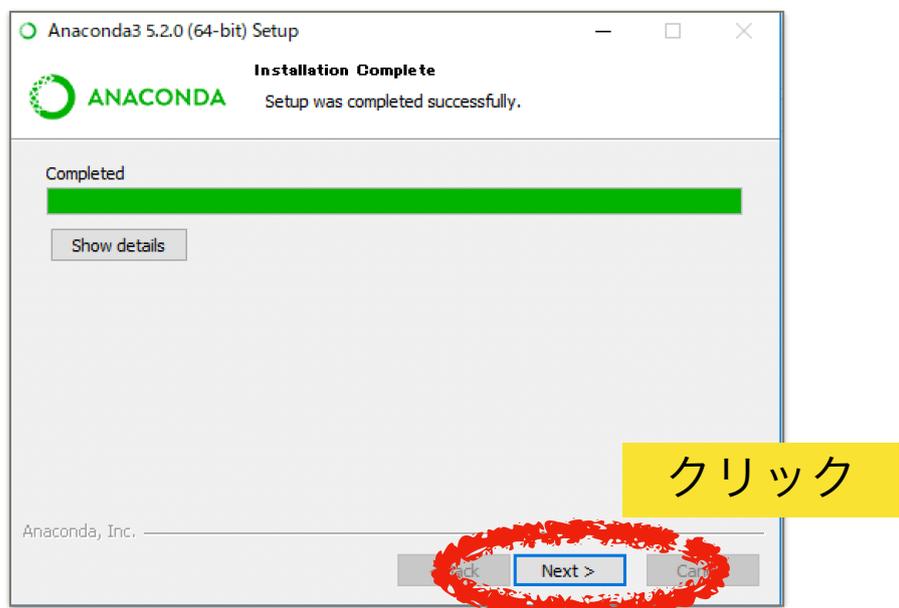


もう少し作業が続きます.

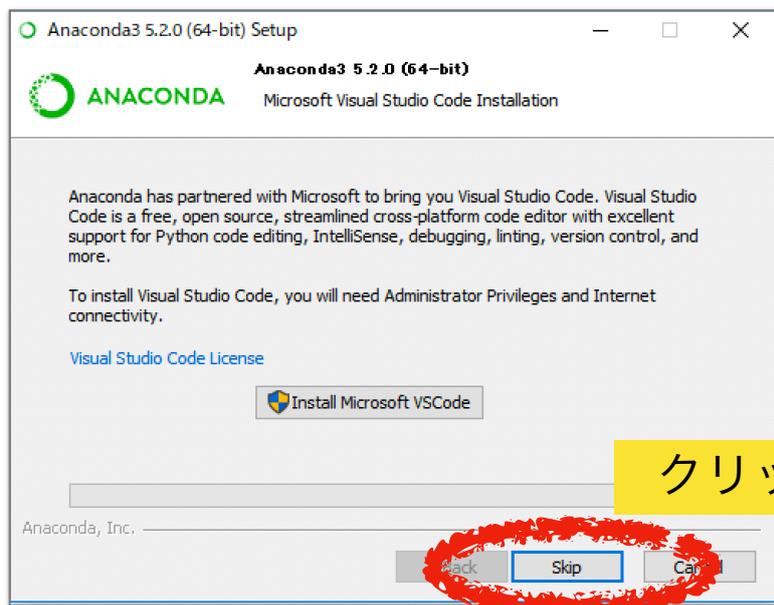
**Step 12** Visual Studio関連のファイルのインストールについて質問されます  
今のところは不要なので、「Skip」します。



**Step 13** 以下の画面が出たら、Anacondaのインストール完了です。

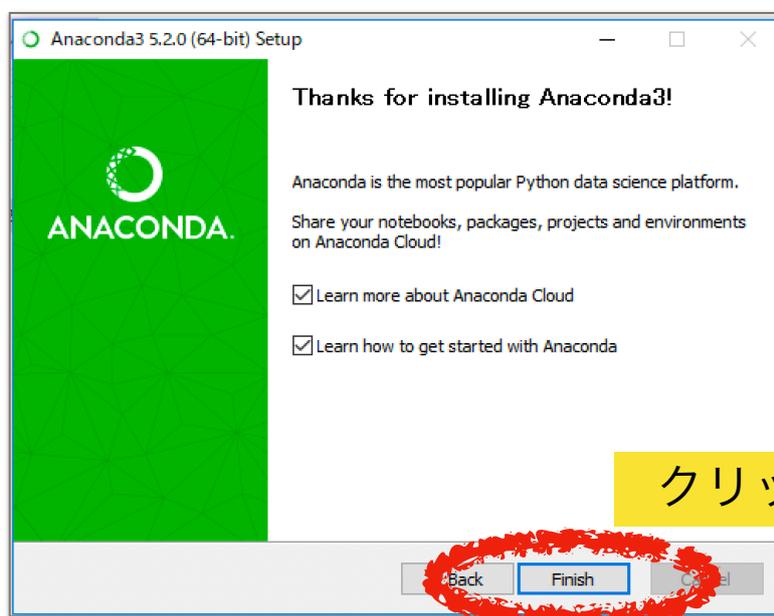


**Step 14** メニューに「Anaconda」が現れて使えるようになります。

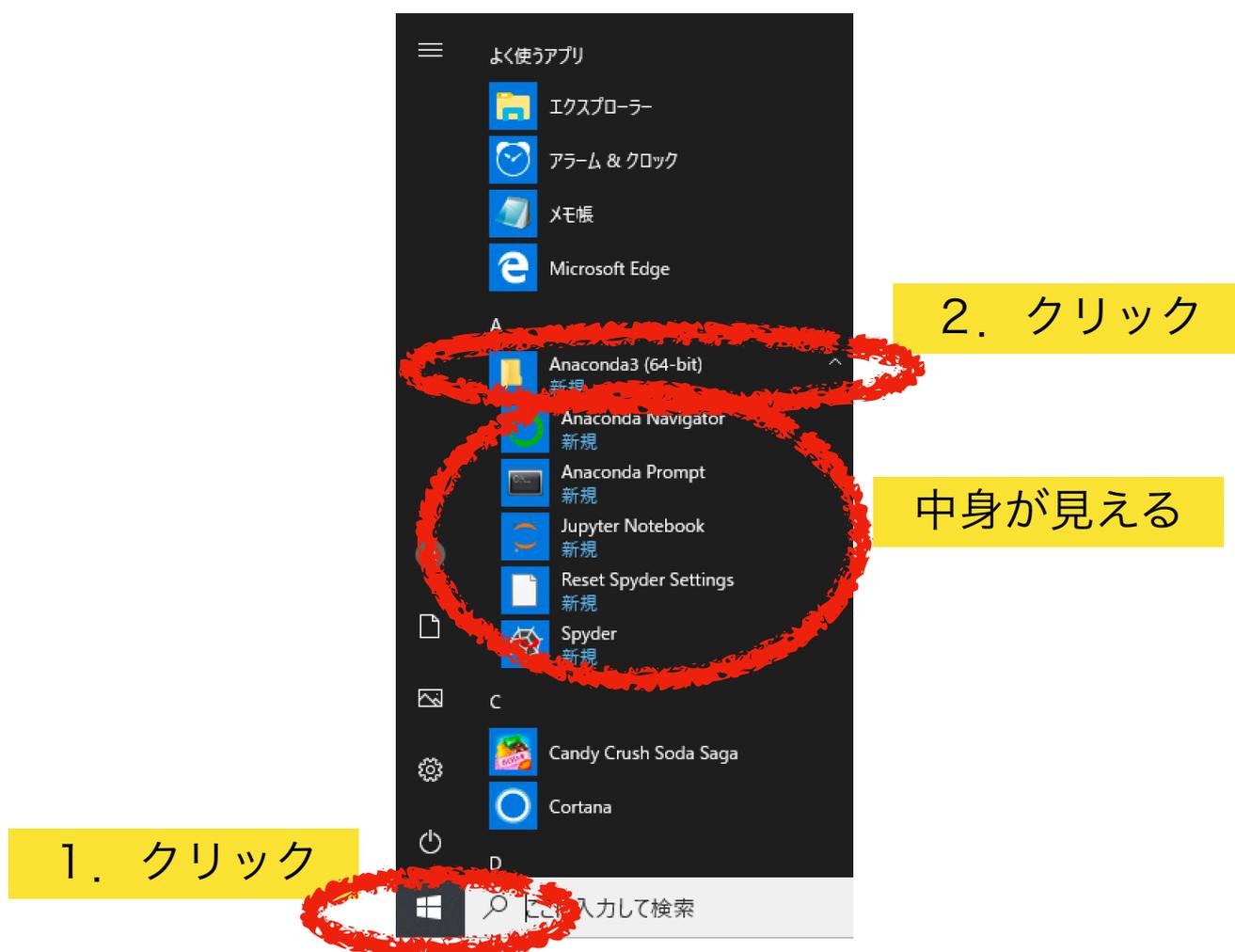


**Step 15** これでAnacondaのインストールが完了です。

仮想環境を作る準備ができました。



これからAnacondaを立ち上げて、

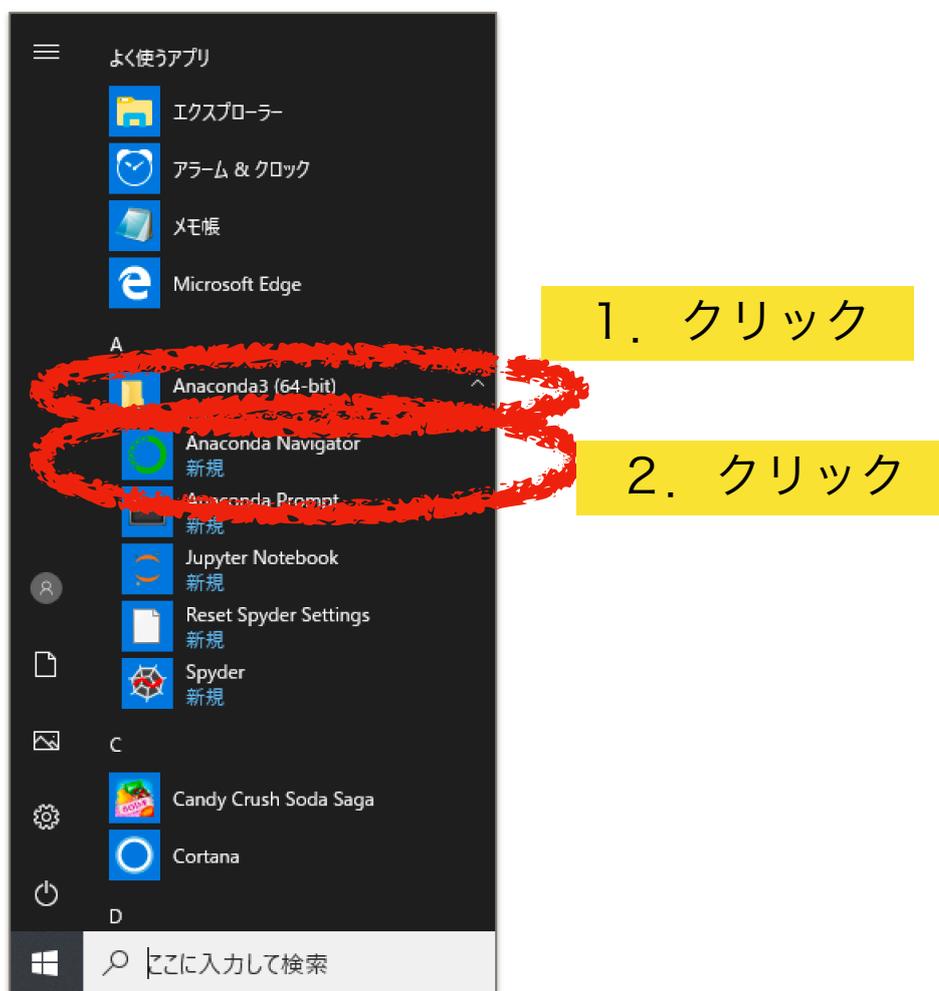


実行する仮想環境（ようは計算するPC）を  
PC上に構築します。  
案外、時間がかかるので、  
ちょっと一休み！

## Step 16 「Anaconda Navigator」を起動します.

似たファイルがあるので注意が必要.

「Anaconda Navigator」を使って環境をつくります.  
Anaconda3フォルダにアプリがまとまっています.  
フォルダーを開くと中にアプリが見えます.

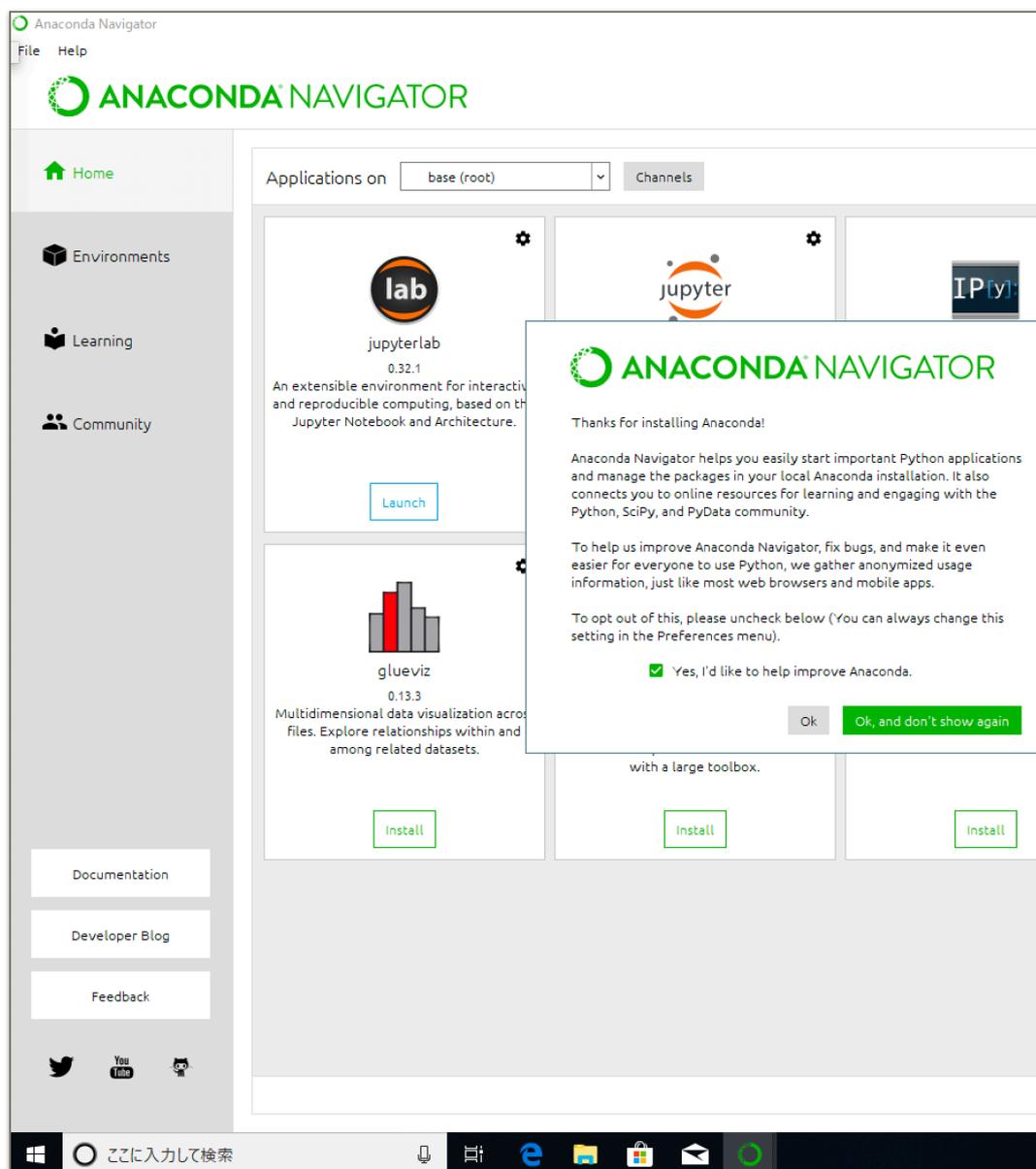


## Step 17 Anaconda Navigatorを起動すると、少し時間がかかります.

初めて起動する場合には、けっこう（5分くらい？）かかります.  
ロゴがふたたび表示される場合があります.  
その中に、設定ファイルを作るメッセージなどが表示されます.  
少し様子を見てください.

**Step 18** 起動して落ち着くと、以下のような画面が表示されます。

ソフトの解析に協力する旨のメッセージがあるので、お好みで、



**Step 19** 左側に「Home, Environments, Learning, Community

と表示されます。確認してください。

これらが表示されない場合には、インストールが失敗です。

Step 1に戻りましょう。

## Step 20 深層学習用の環境を作ります。

左側の「Environments」をクリックします。  
下に「Create」のボタンが現れます。  
その「Create」をクリックします。  
そうすると、右側のようなサブウィンドウが開きます。  
ここで、利用する環境、その名前をつけます。  
環境名は「dls」にしています。自分で好きな名前でもいいです。  
Pythonのみで、バージョンは「3.6」を指定します。  
ただし、新しいPCの場合には3.7にする必要があるかもしれません。  
**TensorFlowとのバージョンの兼ね合いもあるので、まずは3.6で!**  
「Create」をクリックします。ふたたび、しばらく時間がかかります



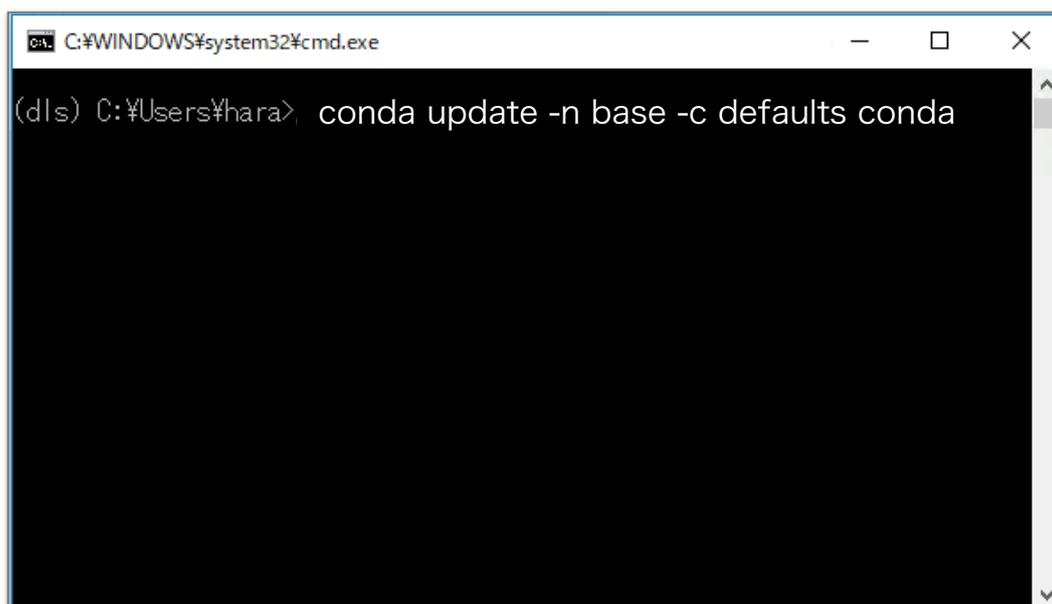
**Step 21** これで深層学習用の仮想環境のプラットフォームができました。  
計算用のPCができた、と考えるとよいです。  
環境名：dls が現れました。

**Step 22** 環境にライブラリーをインストールします。  
非常に重要です。ネットの接続が必須です。  
環境名：dls の右の矢印をクリックします。



サブメニューが開きます。  
Open Terminalをクリックします。

**Step 23** コマンドラインのターミナルが開きます。  
先頭に環境名：dls があります。括弧の中に読めます。  
ここで、conda update -n base -c defaults conda  
をまず投入してください。ちょっと時間がかかります。

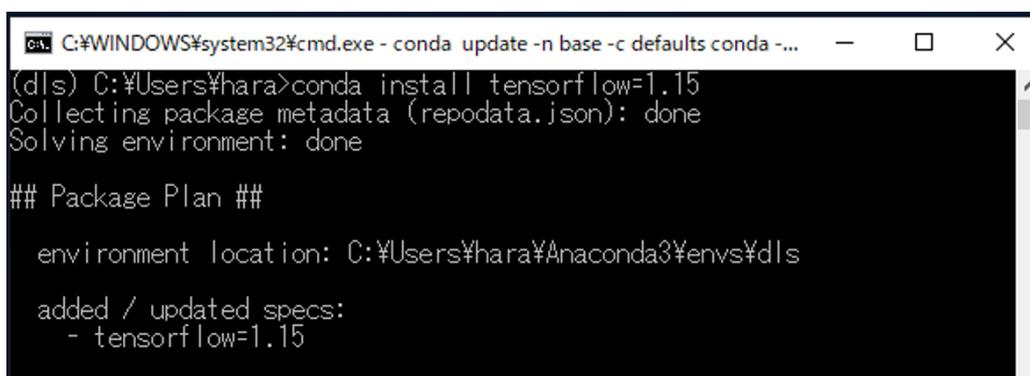


## (お願い！)

Step 23 のコマンドを何度か投入してください。  
矢印キーで上向きを押すと前に投入したコマンドが表示されます。  
最新になった旨の表示ができるまで、「しつこく」やってください。  
時間がかかる場合がありますが、気長にお願いします。

Step 24 ウィンドウをクリックするとコマンドが入力できます。  
次のコマンドを入力します。ちょっと時間がかかります。

```
conda install tensorflow=1.15
```



```
C:\WINDOWS\system32\cmd.exe - conda update -n base -c defaults conda -...
(dls) C:\Users\hara>conda install tensorflow=1.15
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

environment location: C:\Users\hara\Anaconda3\envs\dls

added / updated specs:
- tensorflow=1.15
```

「y」を押したり、「リターンキー」を押して進みます。  
すると、以下のように画面が進みます。  
なぜか途中で止まる場合もある。  
そのときは画面内で「リターン」を連打。

これでTensorflowがインストールできました。

また、それに必要なライブラリがインストールできました。  
それらの動作確認をします。

## Step 25 Pythonのバージョンの確認

ターミナルでコマンド：pythonと入力します。  
 これでPythonが起動されます  
 起動メッセージを注目します。

```

C:\WINDOWS\system32\cmd.exe
(dls) C:\Users\%hara>python
Python 3.6.6 [Anaconda, Inc.] (default, Jun 28 2018, 11:27:44) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> quit()
(dls) C:\Users\%hara>_
    
```

先頭にPython 3.6.6 [Anaconda, Inc.]…と表示されます。  
 Python 3.6.6が表示されます。3.6.8になっているかもしれません。  
 いずれにしても、これで、バージョンが3.6.6であるとわかります。  
 >>> と先頭に表示されます。  
 これがPythonの動作画面です。  
 >>>で、Pythonが起動していることがわかります。  
 Pythonを終了するためには、quit() を投入します。  
 最初のプロンプトに戻ります。

## Step 26 TensorFlowのバージョンの確認

pythonを起動します。  
 Step 25と同じ画面が表示され、Pythonが起動します。  
 次のコマンドで、TensorFlowを読み込みます。

```
import tensorflow
```

すこし時間がかかり「>>>」が表示されます。  
 警告が表示される場合があります。  
 そして、もう一度「import tensorflow」を実行してください。  
 すぐに「>>>」が表示されます。すぐに表示されない場合は失敗です。  
 エラーが表示される場合も同様に失敗です。  
 TensorFlowのインストールをやり直し、Step 20に戻ります。

その際に、Python3.6ではなく「3.7」にします。

(Step 20で失敗した環境を消して、新たに作ります)  
 それでも解決しない場合があります。

**Step 27 >>>のあとに、次のコマンドを入力します.**

```
tensorflow.__version__
```

\_\_ の部分は、アンダースコア「\_」を半角で2つです。  
その前にピリオド「.」があります。  
半角1つのアンダースコアはエラーが出ます。  
正しく動作すると、tensorflowのバージョンが1.15.0であると表示されます。  
quit()で終了します。

この1.15.0は、TensorFlow Ver.1系の最後のバージョンです。  
CPU版とGPU版が統合されています。  
GPUを利用する場合には、  
    CUDA ver.10.0 (10.1ではダメだよ！)  
    cuDNN ver.7.4.1以上  
に対応したnVIDIAのGPUが必要です。  
GPUはCUDA Compute Capability 3.5以上である必要があります！

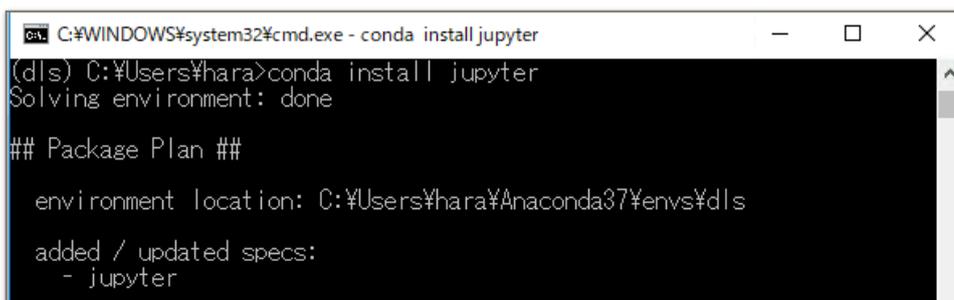
これで、  
Anaconda  
TensorFlow  
のインストールができました。

もう少しだけつづくよ！

## Step 28 Jupyter Notebookのインストール

Jupyter Notebookは、ブラウザで動作する実行環境です。  
後ほど使う時があるのでインストールします。  
仮想環境を実行した画面で、次のコマンドでインストールします。

```
conda install jupyter
```

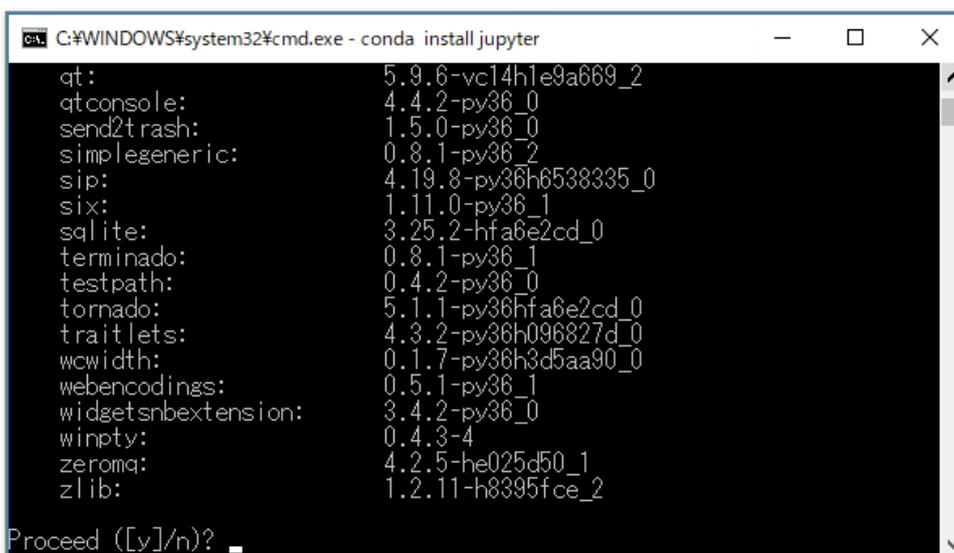


```
C:\WINDOWS\system32\cmd.exe - conda install jupyter
(dls) C:\Users\hara>conda install jupyter
Solving environment: done

## Package Plan ##

environment location: C:\Users\hara\Anaconda37\envs\dls
added / updated specs:
- jupyter
```

途中で、インストールしてよいか、確認があります。



```
C:\WINDOWS\system32\cmd.exe - conda install jupyter
qt: 5.9.6-vc14h1e9a669_2
qtconsole: 4.4.2-py36_0
send2trash: 1.5.0-py36_0
simplegeneric: 0.8.1-py36_2
sip: 4.19.8-py36h6538335_0
six: 1.11.0-py36_1
sqlite: 3.25.2-hfa6e2cd_0
terminado: 0.8.1-py36_1
testpath: 0.4.2-py36_0
tornado: 5.1.1-py36hfa6e2cd_0
traitlets: 4.3.2-py36h096827d_0
wcwidth: 0.1.7-py36h3d5aa90_0
webencodings: 0.5.1-py36_1
widgetsnbextension: 3.4.2-py36_0
winpty: 0.4.3-4
zeromq: 4.2.5-he025d50_1
zlib: 1.2.11-h8395fce_2
Proceed ([y]/n)?
```

「y」を押して続けます。

この後、けっこう時間がかかります。

途中で止まっている場合もあるので、しつこく「Return」をおします。

### Step 29 さらに、必要になるライブラリをインストールします。

Step 28と同じように、コマンドで順次、インストールします。

Step 28と同様に、それぞれで確認のため「y」を押す必要があります

```
conda install matplotlib
conda install pillow
conda install imageio
conda install pandas
conda install spyder
conda install scikit-learn
```

OpenCV関連もインストールします。

```
conda install cython
conda install opencv
```

DICOM関連もインストールします。少しコマンドが違います。

以下は、途中で折り返しています。1行で書きます。

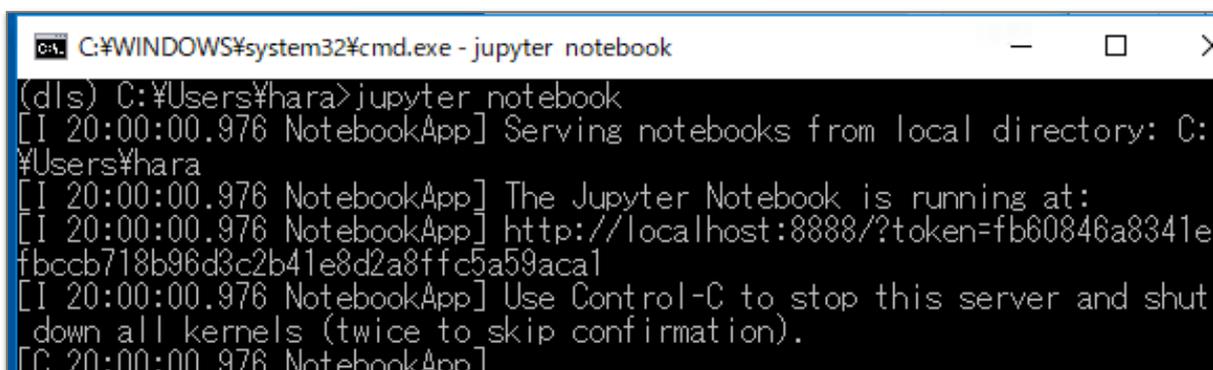
```
conda install --channel
https://conda.anaconda.org/conda-forge pydicom
```

### Step 30 Kerasをインストールします

```
conda install keras
```

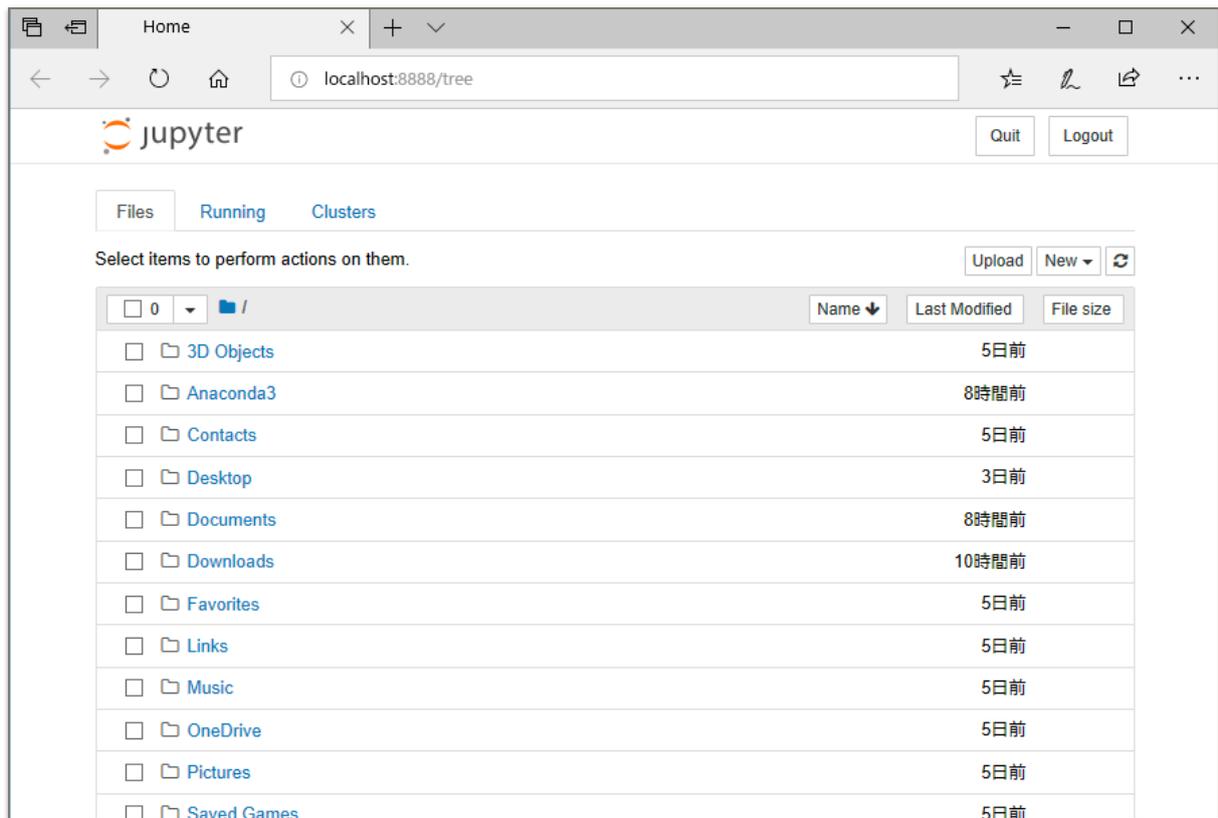
### Step 31 Jupyter Notebookの起動。コマンドで実行します。

```
jupyter notebook
```

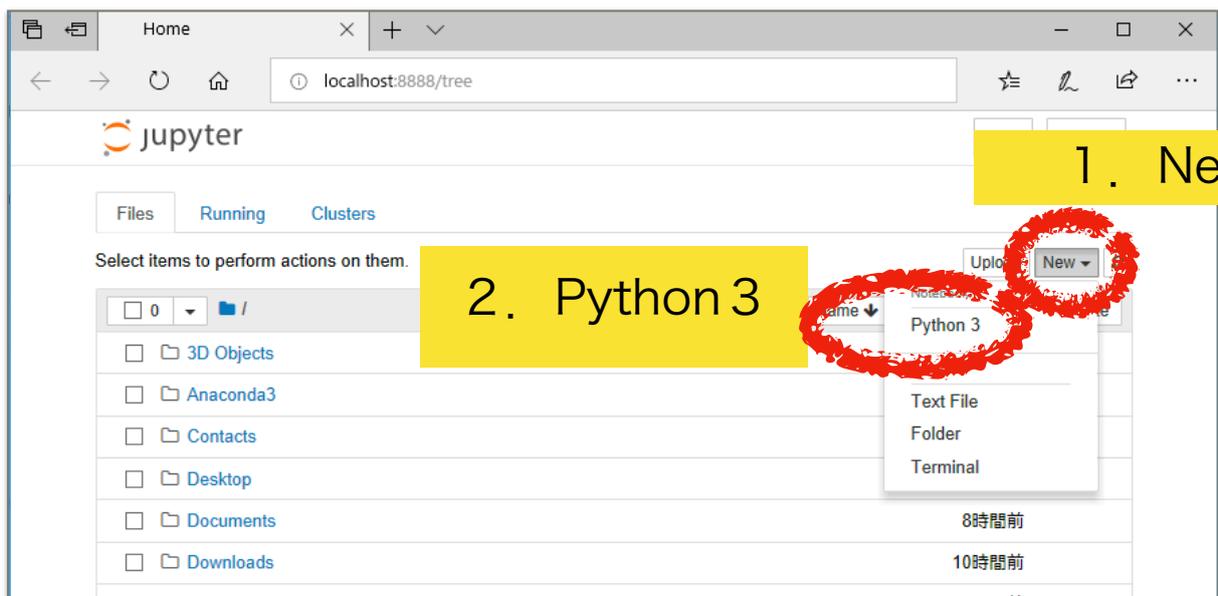


```
C:\WINDOWS\system32\cmd.exe - jupyter notebook
(dls) C:\Users\hara>jupyter notebook
[I 20:00:00.976 NotebookApp] Serving notebooks from local directory: C:\Users\hara
[I 20:00:00.976 NotebookApp] The Jupyter Notebook is running at:
[I 20:00:00.976 NotebookApp] http://localhost:8888/?token=fb60846a8341efbccb718b96d3c2b41e8d2a8ffc5a59aca1
[I 20:00:00.976 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 20:00:00.976 NotebookApp]
```

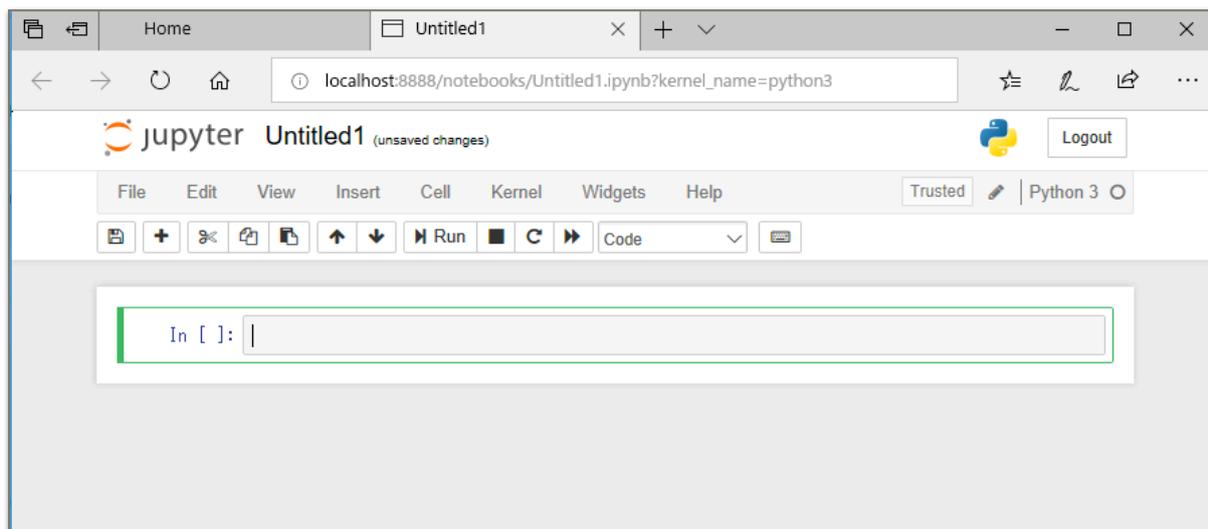
## Step 32 しばらく待つと、ブラウザの画面が開きます



## Step 34 右側の「New」をクリックすると、サブメニューが開きます。 Python3を選びます。



## Step 35 実行画面が表示されます.



## Step 36 Pythonの環境の簡単な確認

In []の枠の中に次のプログラムを入れましょう.

```
a=1
b=2
c=a+b
d=a*b
print('a + b = ', c)
print('a * b = ', d)
```

「Shift」キーを押しながら「リターン」を押すと実行されます.

```
In [5]: a=1
b=2
c=a+b
d=a*b
print('a + b = ', c)
print('a * b = ', d)
```

←ここまで書いて  
Shift+Return

結果が表示される→

```
a + b = 3
a * b = 2
```

計算結果が表示されます.

### Step 37 TensorFlowも使ってみましょう

In []の枠の中に次のプログラムを入れましょう。

```
import tensorflow as tf

a = tf.constant(2, name='a')
b = tf.constant(3, name='b')
c = tf.add(a, b)
d = tf.multiply(a, b)

with tf.Session() as session:
    print('a + b = ', session.run(c))
    print('a * b = ', session.run(d))
```

```
In [12]: import tensorflow as tf

a = tf.constant(2, name='a')
b = tf.constant(3, name='b')
c = tf.add(a, b)
d = tf.multiply(a, b)

with tf.Session() as se:
    print('a + b = ', se.run(c))
    print('a * b = ', se.run(d))
```

結果が表示  
される→

```
a + b = 5
a * b = 6
```

←ここまで書いて  
Shift+Return

### Step 38 一度、Jupyter Notebook, Anacondaをすべて終了しましょう。

そして、もう一度、

Anacondaを立ち上げ、

環境：dls を選び、

ターミナルを立ち上げ、

Jupyter Notebookを立ち上げ、

ましょう

**Step 39 Kerasのデータセットをダウンロードしてみましょう。**

文字認識データ：MNISTはよく知られています。

fashion\_mnistは、MNISTと互換のファッション記事データです。

```
from keras.datasets import fashion_mnist
(x_train, y_train), (x_test, y_test)=fashion_mnist.load_data()
```

```
In [4]: from keras.datasets import fashion_mnist
        (x_train, y_train), (x_test, y_test)=fashion_mnist.load_data()

Downloading data from http://fashion-mnist.s3-website.eu-central-
labels-idx1-ubyte.gz
32768/29515 [=====] - 1s 20us/step
Downloading data from http://fashion-mnist.s3-website.eu-central-
images-idx3-ubyte.gz
26427392/26421880 [=====] - 34s 1us/step
Downloading data from http://fashion-mnist.s3-website.eu-central-
abels-idx1-ubyte.gz
8192/5148 [=====] - 0s
Downloading data from http://fashion-mnist.s3-website.eu-central-
mages-idx3-ubyte.gz
4423680/4422102 [=====] - 9s 2us/step
```

**Step 40 データセットの数, 形式を表示してみましょう**

```
print(x_train.shape, x_test.shape, y_train.shape, y_test.shape)
```

```
In [6]: print(x_train.shape, x_test.shape, y_train.shape, y_test.shape)

(60000, 28, 28) (10000, 28, 28) (60000,) (10000,)
```

x\_trainは、 60000枚の画像で、 1 画像は28x28画素。

x\_testは、 10000枚の画像で、 1 画像は28x28画素。

であることがわかります。

また、 y\_train, y\_testは、 それぞれに対応するラベル番号です。

ラベル番号は、 0 : シャツ, 1 : ズボン, . . . 9 : ブーツの意味。

## Step 41 実際の演習用のデータダウンロード

少し時間がかかるので、あらかじめダウンロードするとよいでしょう。

#文字認識の有名データベース：MNIST

```
from tensorflow.python.keras.datasets import mnist
(image_train, label_train), (image_test, label_test) = mnist.load_data()
```

#自然画像のデータベース・CIFARの練習用データベース：CIFAR10

```
from tensorflow.python.keras.datasets import cifar10
(img_train, label_train), (img_test, label_test) = cifar10.load_data()
```

## まとめ

1. Anacondaがインストールできた
2. Anacondaで実習用の環境が構築できた：dls
3. 仮想環境のコマンドプロンプトが操作できた
4. 仮想環境にTensorFlow, Keras, その他必要なライブラリをインストールできた
5. コマンドプロンプトでpythonが起動できた
6. pythonのバージョン, tensorflowのバージョンが確認できた
7. Jupyter Notebookが起動できた
8. Notebookの中で, Pythonを起動できた
9. Anaconda, Jupyter Notebookの再起動ができた
10. 起動したPythonの画面で, tensorflowが動いた
11. 起動したPythonの画面で, kerasが動いた
12. ネットに接続して, データベースがダウンロードできた

お疲れ様でした！  
環境構築ができました  
(GPUを用いない計算／演習用です)

既知の問題：

1. アカウント名に和文が含まれているとエラーが出る  
・別アカウントを作って対応.
2. 一部5.2の画面が残っている (anaconda install部分)
3. どうしてもダメなときは、環境を削除してから再構築. ちょっと危険かも.  
環境名が「dls」ならば,  
  
deactivate  
conda remove -n dls --all
4. 新しいPCの場合, TensorFlowのインストールに失敗する (dllの非対応) .  
CPUの命令に対応の差異がある.  
今のところ, 事前に判別する対処方法が見つからない.  
Pythonを3.6で試してダメなら3.7にする.  
最初から3.7でもよいかもしれない.  
しかし, 古いPCはTensorFlowの最新版 (1.X系) ではエラーになることも.  
(まとめ)  
Pythonは3.6  
TensorFlowは1.15.0  
Kerasはその依存版
5. TensorFlow1.15でもCPUのマルチコアを最大限利用できない  
TF2.0/2.1ならフルに使える
6. Python3.8のAnacondaしか見えない  
それでもよいのでやってみてください.  
もしくは古いバージョンを探してインストールしてください.
7. Python 3.6が表示されない  
Base (root) のターミナルで  
conda install python=3.6  
を入れる. 時間がかかる.  
その後, anacondaを再起動すると見えるはず.