Anacondaの使用方法

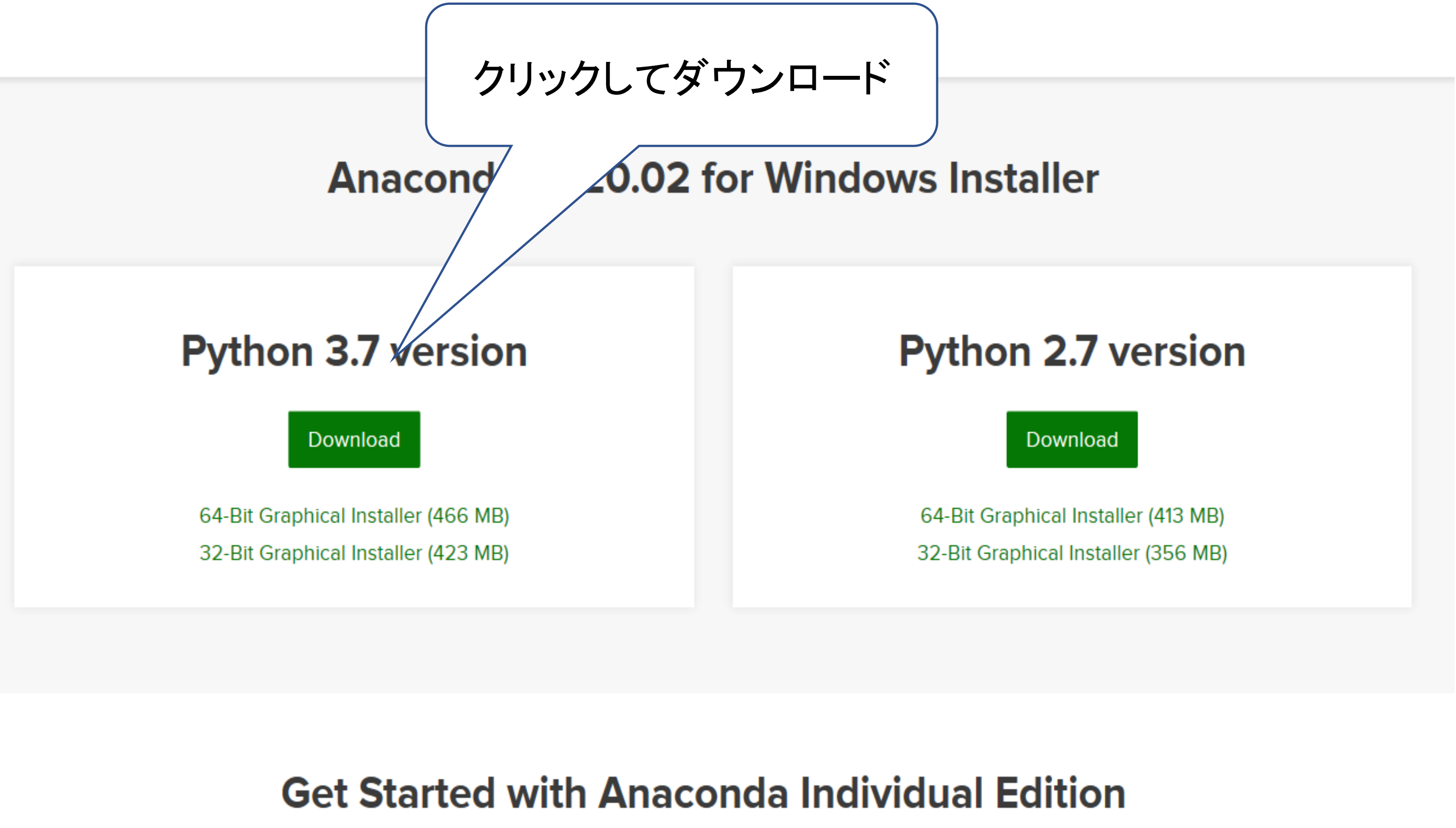
知的高性能計算研究室

2020.4.2

- Anaconda(アナコンダ)とは、「データサイエンス」や「機械学習関連アプリケーション」のためのPythonおよびR言語用のディストリビューションです。必要とするライブラリを1つずつインストールする手間を省くことができ、データサイエンスや機械学習を行うための環境を即座に構築できます。

Anacondaのダウンロード

- https://www.anaconda.com/distribution/



クリックしてダウンロード

Anacond 20.02 for Windows Installer

Python 3.7 version

64-Bit Graphical Installer (466 MB)

32-Bit Graphical Installer (423 MB)

Python 2.7 version

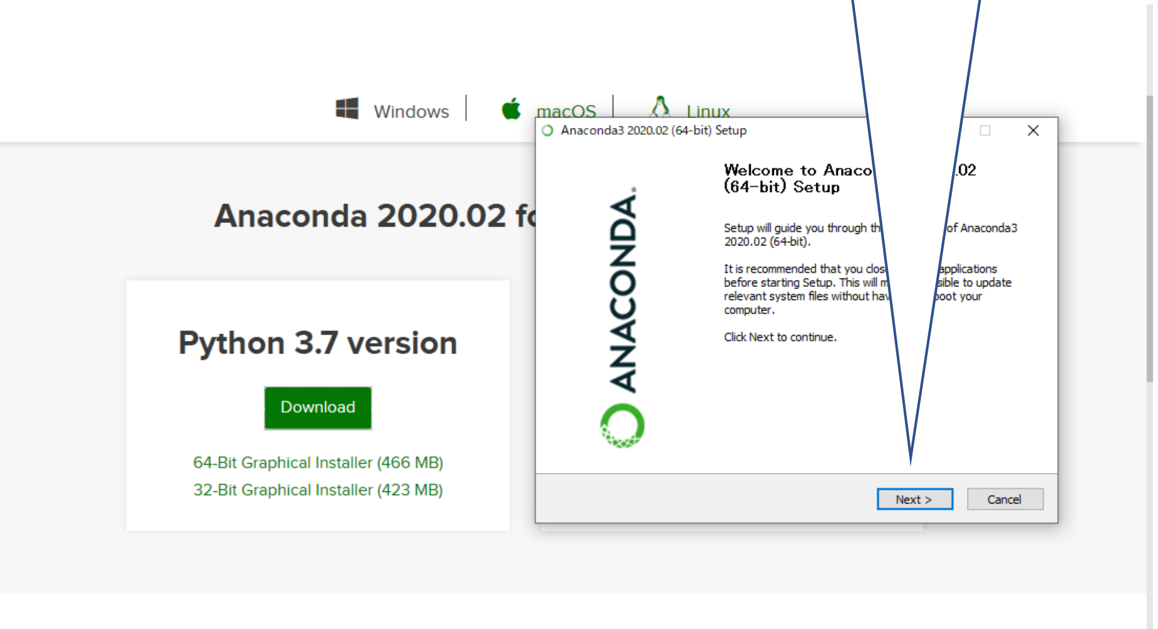
64-Bit Graphical Installer (413 MB)

32-Bit Graphical Installer (356 MB)

Get Started with Anaconda Individual Edition

Anacondaのインストール

Nextを押して、インストール



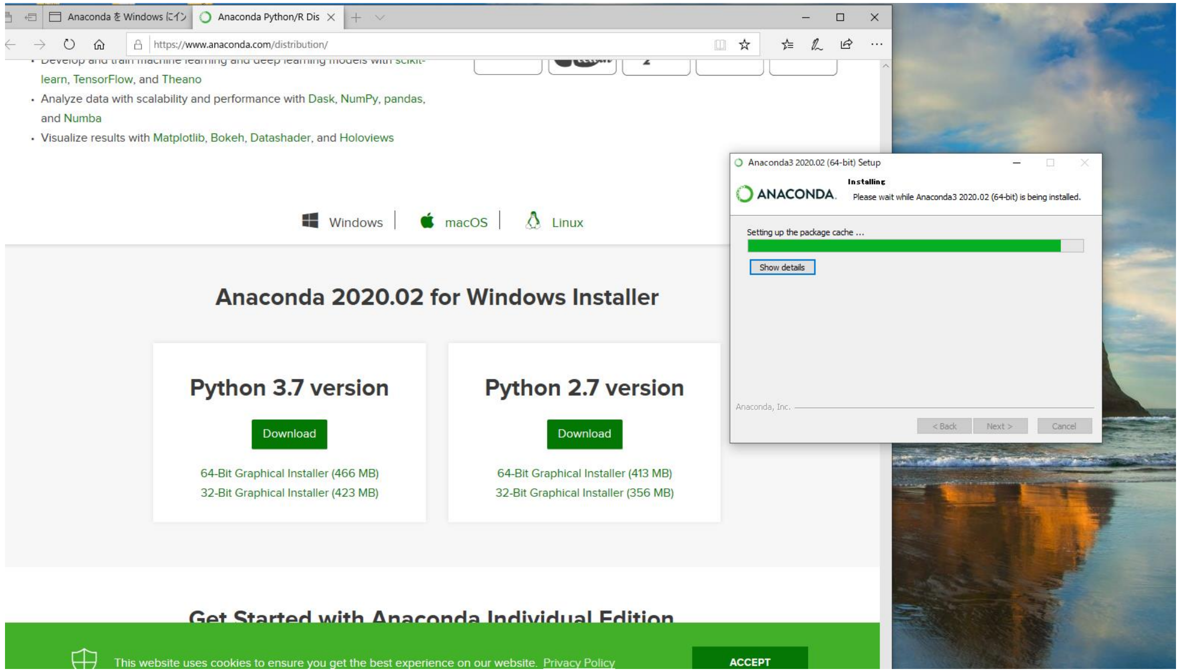
Anaconda3 2020.02 (64-bit) Setup

Welcome to Anaco .02
(64-bit) Setup

Setup will guide you through th of Anaconda3 2020.02 (64-bit).

It is recommended that you clos applications before starting Setup. This will m sible to update relevant system files without hav boot your computer.

Click Next to continue.



- learn, TensorFlow, and Theano
- Analyze data with scalability and performance with Dask, NumPy, pandas, and Numba
- Visualize results with Matplotlib, Bokeh, Datashader, and Holoviews

Anaconda3 2020.02 (64-bit) Setup

Installing

Please wait while Anaconda3 2020.02 (64-bit) is being installed.

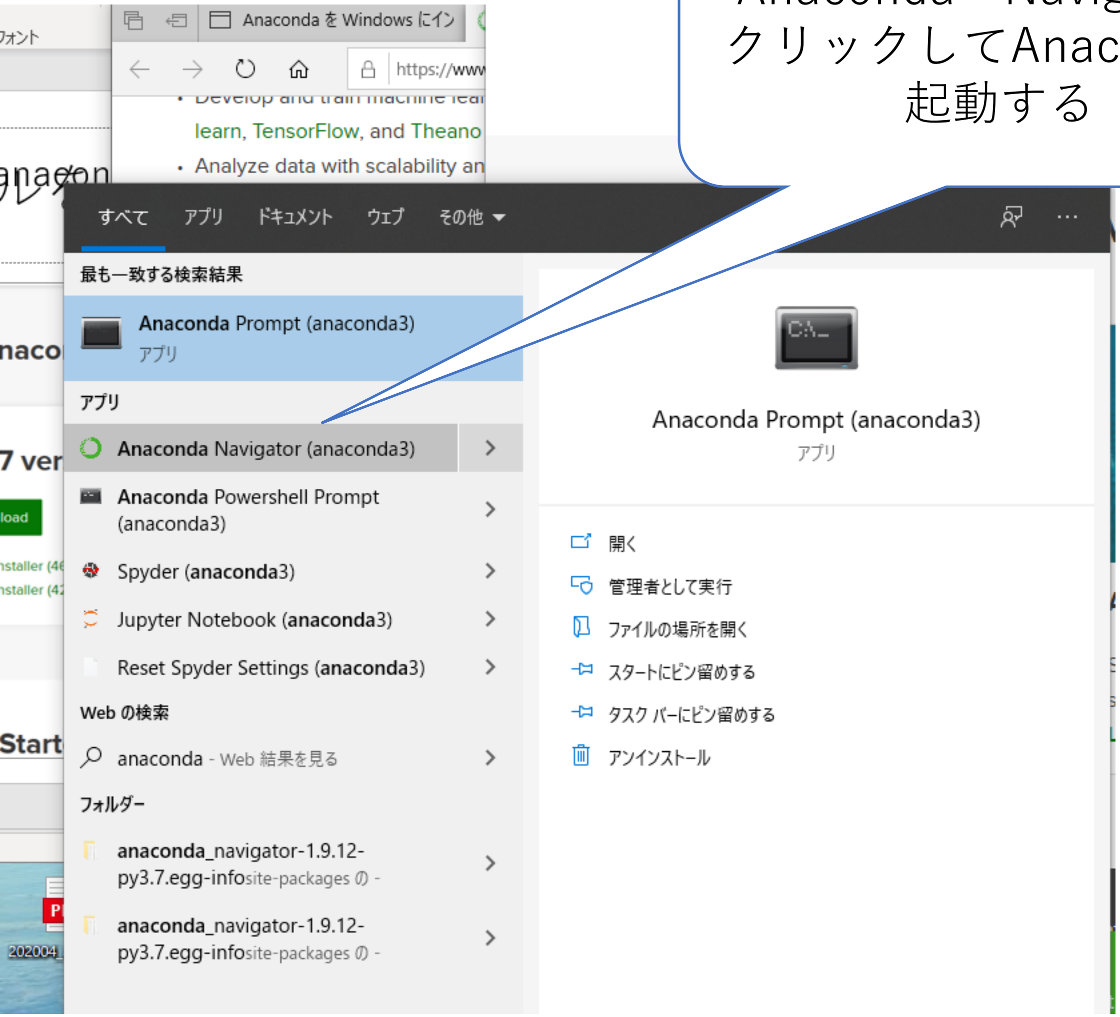
Setting up the package cache ...

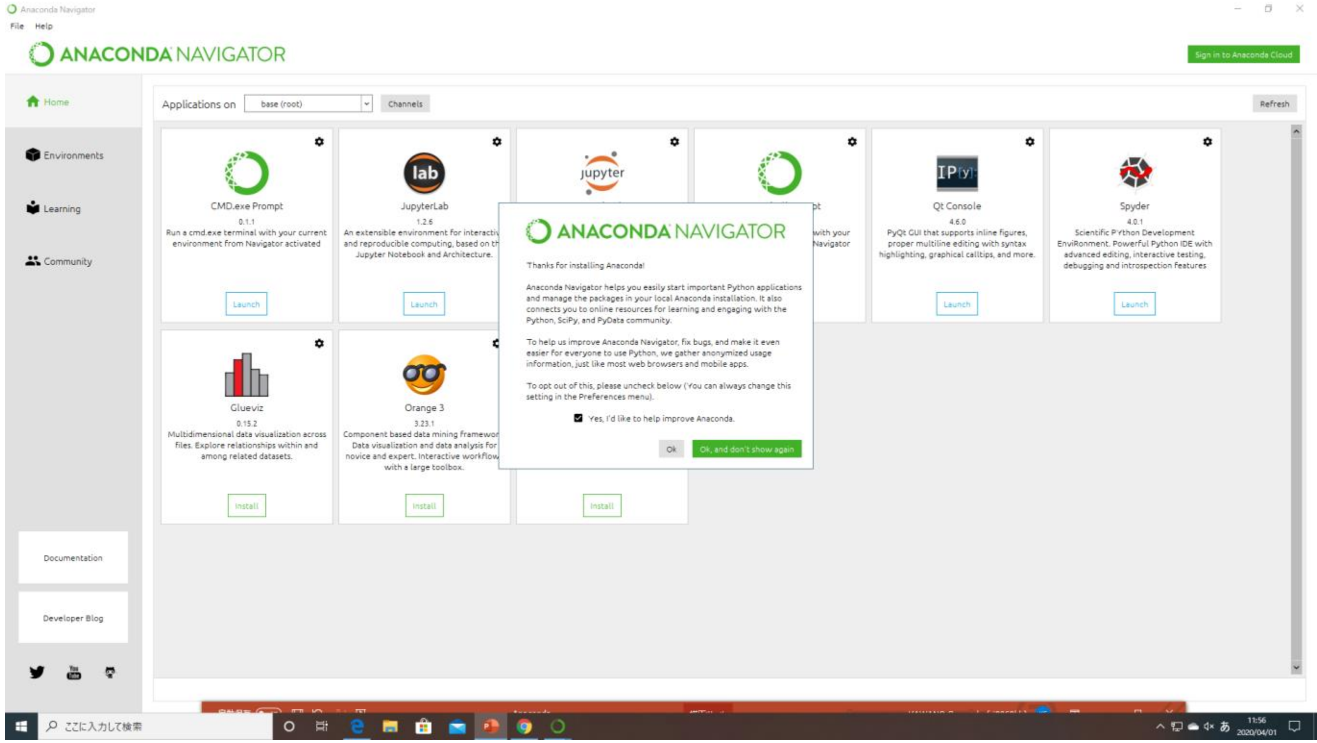
This website uses cookies to ensure you get the best experience on our website. Privacy Policy

Anaconda を Windows にイン
Anaconda Python/R Dis
https://www.anaconda.com/distribution/
Windows
macOS
Linux
Anaconda 2020.02 for Windows Installer
Python 3.7 version
Download
64-Bit Graphical Installer (466 MB)
32-Bit Graphical Installer (423 MB)
Python 2.7 version
Download
64-Bit Graphical Installer (413 MB)
32-Bit Graphical Installer (356 MB)
Get Started with Anaconda Individual Edition
ACCEPT
ANACONDA.
Show details
Anaconda, Inc.
< Back
Next >
Cancel

Anacondaの起動

Anaconda　Navigatorをクリックして Anacondaを起動する


すべて
アプリ
ドキュメント
ウェブ
その他
最も一致する検索結果
Anaconda Prompt (anaconda3)
アプリ
アプリ
Anaconda Navigator (anaconda3)
Anaconda Powershell Prompt (anaconda3)
Spyder (anaconda3)
Jupyter Notebook (anaconda3)
Reset Spyder Settings (anaconda3)
Web の検索
anaconda - Web 結果を見る
フォルダー
anaconda_navigator-1.9.12-py3.7.egg-infosite-packages の -
anaconda_navigator-1.9.12-py3.7.egg-infosite-packages の -
Anaconda Prompt (anaconda3)
アプリ
開く
管理者として実行
ファイルの場所を開く
スタートにピン留めする
タスク バーにピン留めする
アンインストール



ANACONDA NAVIGATOR

Thanks for installing Anaconda!

Anaconda Navigator helps you easily start important Python applications and manage the packages in your local Anaconda installation. It also connects you to online resources for learning and engaging with the Python, SciPy, and PyData community.

To help us improve Anaconda Navigator, fix bugs, and make it even easier for everyone to use Python, we gather anonymized usage information, just like most web browsers and mobile apps.

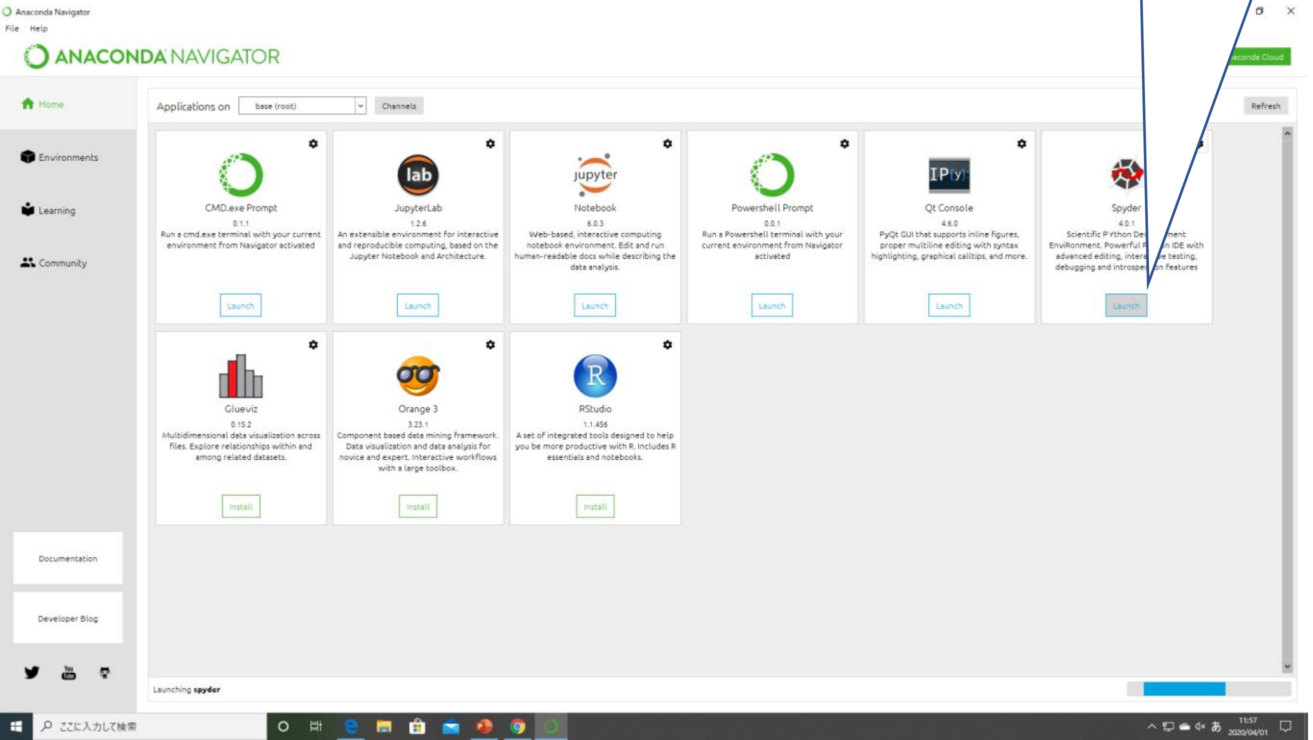
To opt out of this, please uncheck below (You can always change this setting in the Preferences menu).

☑ Yes, I'd like to help improve Anaconda.

Ok　Ok, and don't show again

Editor “spyder”を起動する

Spyderをクリックして、
Editorを起動する



CMD.exe Prompt: Run a cmd.exe terminal with your current environment from Navigator activated

JupyterLab: An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Notebook: Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Powershell Prompt: Run a Powershell terminal with your current environment from Navigator activated

Qt Console: PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

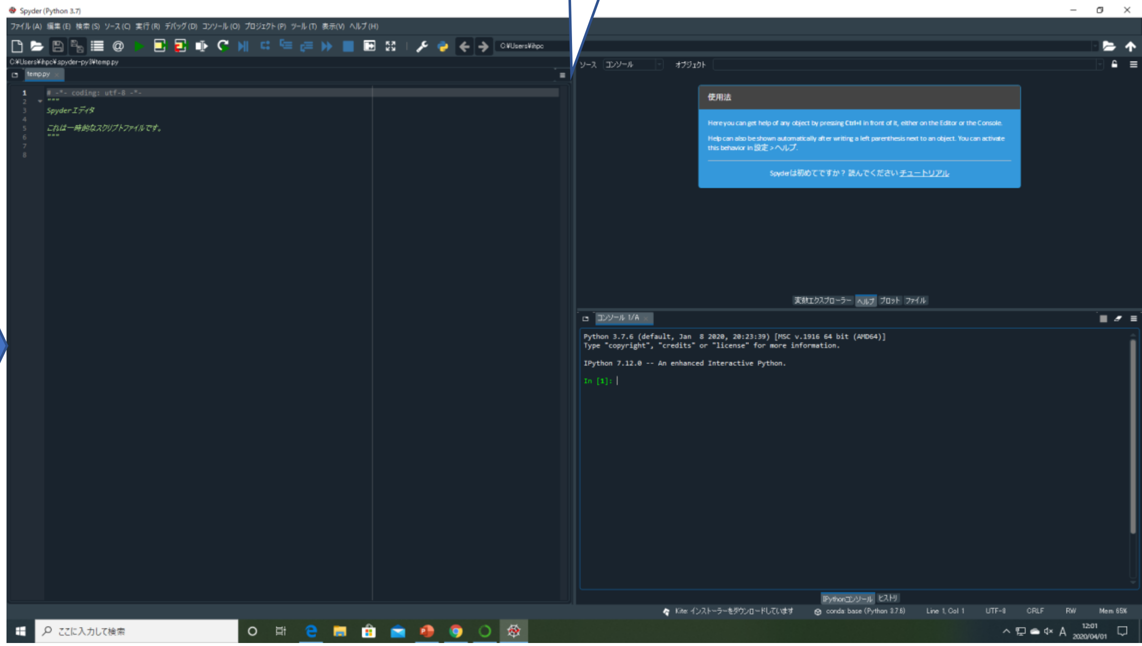
Spyder: Scientific PYthon De[illegible]ment EnviRonment. Powerful P[illegible]n IDE with advanced editing, intera[illegible]e testing, debugging and introspe[illegible]on features

Glueviz: Multidimensional data visualization across files. Explore relationships within and among related datasets.

Orange 3: Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

RStudio: A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

Spyder起動後の様子


Spyder (Python 3.7)
ファイル (A) 編集 (E) 検索 (S) ソース (C) 実行 (R) デバッグ (D) コンソール (O) プロジェクト (P) ツール (T) 表示(V) ヘルプ (H)
C:¥Users¥hpc
C:¥Users¥hpc¥.spyder-py3¥temp.py
temp.py
ソース コンソール オブジェクト
使用法
変数エクスプローラー ヘルプ プロット ファイル
コンソール 1/A
IPythonコンソール ヒストリ
ここに入力して検索

```
# -*- coding: utf-8 -*-
"""
Spyderエディタ

これは一時的なスクリプトファイルです。
"""
```

Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.

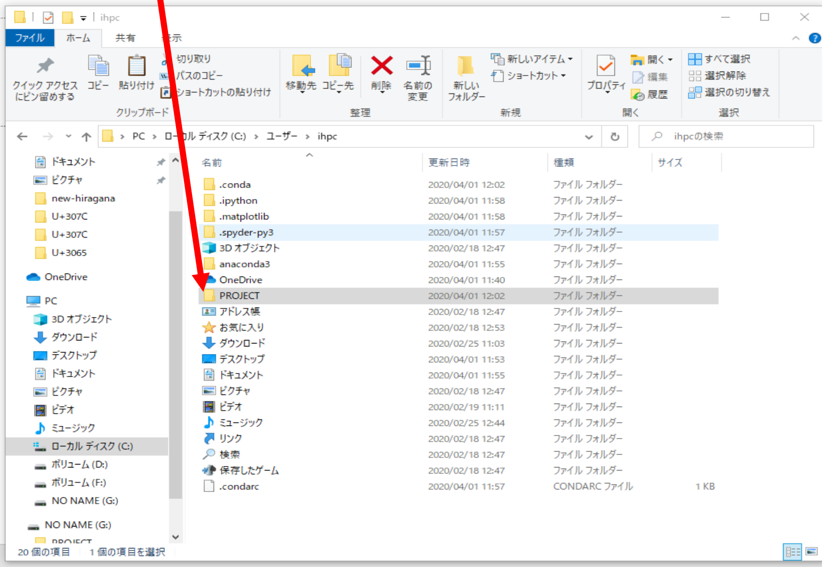
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in 設定 > ヘルプ.

Spyderは初めてですか？ 読んでください チュートリアル

```
Python 3.7.6 (default, Jan  8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: 
```

Spyderによりプロジェクトを起動する

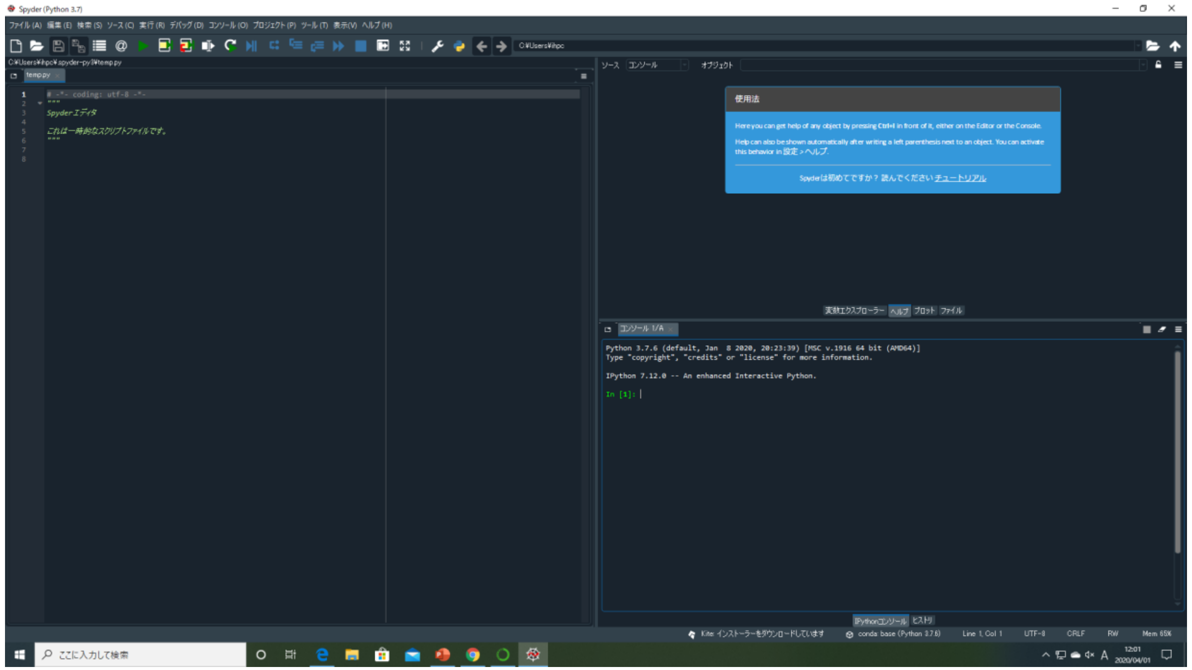
- PC>ローカルディスク＞ユーザー＞ihpc>PROJECT
  - 仮に、上記のディレクトリにプロジェクトが保存されている。



PC > ローカル ディスク (C:) > ユーザー > ihpc

名前	更新日時	種類	サイズ
.conda	2020/04/01 12:02	ファイル フォルダー	
.ipython	2020/04/01 11:58	ファイル フォルダー	
.matplotlib	2020/04/01 11:58	ファイル フォルダー	
.spyder-py3	2020/04/01 11:57	ファイル フォルダー	
3D オブジェクト	2020/02/18 12:47	ファイル フォルダー	
anaconda3	2020/04/01 11:55	ファイル フォルダー	
OneDrive	2020/04/01 11:40	ファイル フォルダー	
PROJECT	2020/04/01 12:02	ファイル フォルダー	
アドレス帳	2020/02/18 12:47	ファイル フォルダー	
お気に入り	2020/02/18 12:53	ファイル フォルダー	
ダウンロード	2020/02/25 11:03	ファイル フォルダー	
デスクトップ	2020/04/01 11:53	ファイル フォルダー	
ドキュメント	2020/04/01 11:55	ファイル フォルダー	
ピクチャ	2020/02/18 12:47	ファイル フォルダー	
ビデオ	2020/02/19 11:11	ファイル フォルダー	
ミュージック	2020/02/25 12:44	ファイル フォルダー	
リンク	2020/02/18 12:47	ファイル フォルダー	
検索	2020/02/18 12:47	ファイル フォルダー	
保存したゲーム	2020/02/18 12:47	ファイル フォルダー	
.condarc	2020/04/01 11:57	CONDARC ファイル	1 KB

20 個の項目　1 個の項目を選択



```
# -*- coding: utf-8 -*-
"""
Spyderエディタ

これは一時的なスクリプトファイルです。
"""
```

使用法

Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in 設定 > ヘルプ.

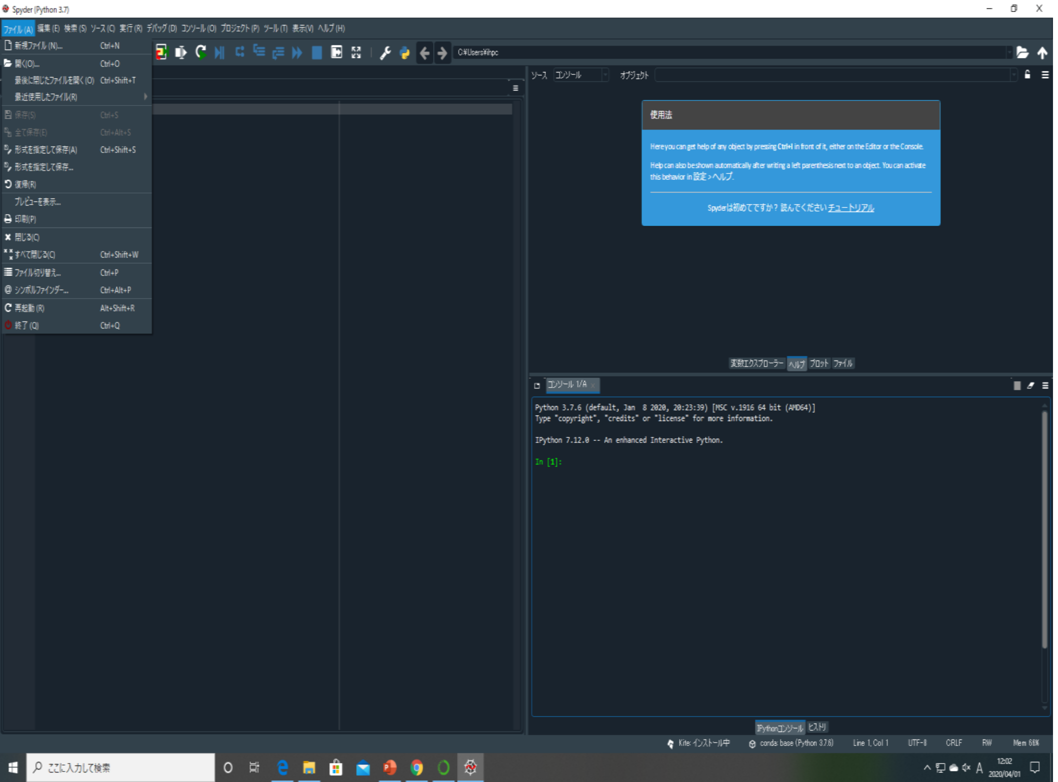
Spyderは初めてですか？ 読んでください チュートリアル

```
Python 3.7.6 (default, Jan  8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.12.0 -- An enhanced Interactive Python.

In [1]: 
```

Spyderによるプロジェクトの起動

- File－＞open

プロセッサのpython ファイルを
クリックする。
ここでは、main1.pyとする

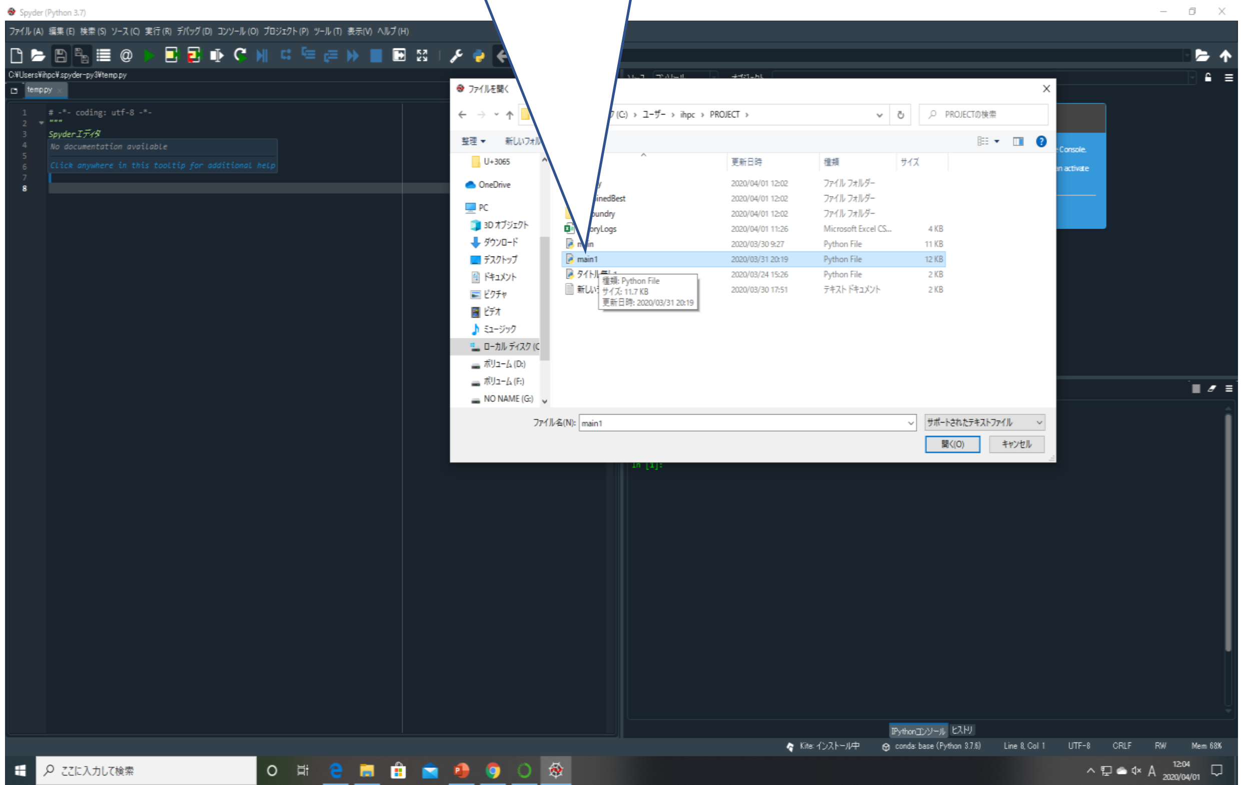

Spyder (Python 3.7)
ファイル (A) 編集 (E) 検索 (S) ソース (C) 実行 (R) デバッグ (D) コンソール (O) プロジェクト (P) ツール (T) 表示(V) ヘルプ (H)
新規ファイル (N)... Ctrl+N
開く(O)... Ctrl+O
最後に閉じたファイルを開く (O) Ctrl+Shift+T
最近使用したファイル(R)
保存(S) Ctrl+S
全て保存(E) Ctrl+Alt+S
形式を指定して保存(A) Ctrl+Shift+S
形式を指定して保存...
復帰(R)
プレビューを表示...
印刷(P)
閉じる(C)
すべて閉じる(C) Ctrl+Shift+W
ファイル切り替え... Ctrl+P
シンボルファインダー... Ctrl+Alt+P
再起動 (R) Alt+Shift+R
終了 (Q) Ctrl+Q
C:¥Users¥ihpc
ソース コンソール オブジェクト
使用法
変数エクスプローラー ヘルプ プロット ファイル
コンソール 1/A
IPythonコンソール ヒストリ

Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in 設定 > ヘルプ.

Spyderは初めてですか？ 読んでください チュートリアル

```
Python 3.7.6 (default, Jan  8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.12.0 -- An enhanced Interactive Python.

In [1]:
```



```
# -*- coding: utf-8 -*-
"""
Spyderエディタ
No documentation available

Click anywhere in this tooltip for additional help
```

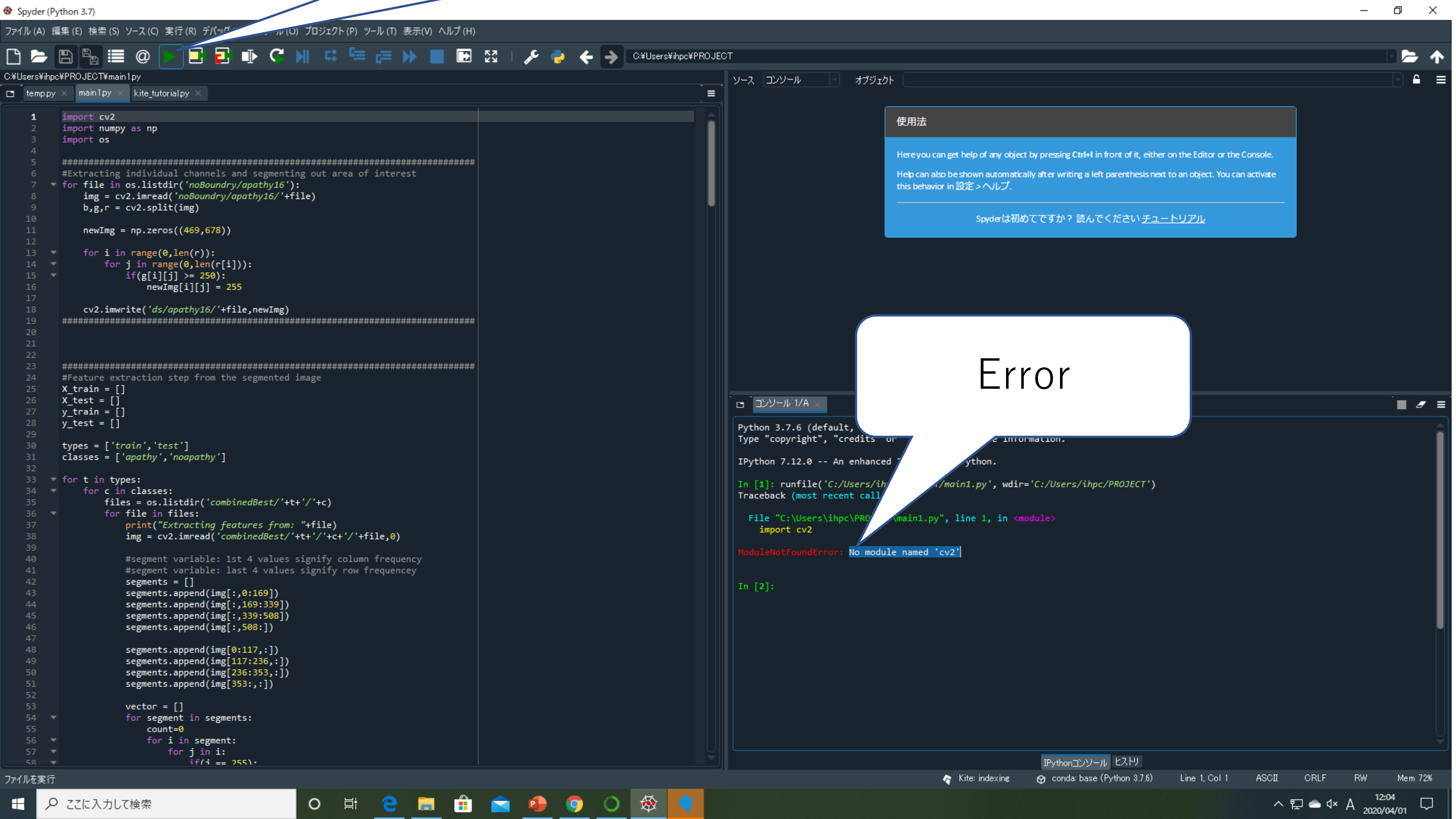
	更新日時	種類	サイズ
y	2020/04/01 12:02	ファイル フォルダー	
inedBest	2020/04/01 12:02	ファイル フォルダー	
oundry	2020/04/01 12:02	ファイル フォルダー	
oryLogs	2020/04/01 11:26	Microsoft Excel CS...	4 KB
m in	2020/03/30 9:27	Python File	11 KB
main1	2020/03/31 20:19	Python File	12 KB
タイトル	2020/03/24 15:26	Python File	2 KB
新しい	2020/03/30 17:51	テキスト ドキュメント	2 KB

種類: Python File
サイズ: 11.7 KB
更新日時: 2020/03/31 20:19

ファイル名(N): main1
サポートされたテキストファイル

ファイルの実行とエラーのエラー対策

三角の ボタンを押して実行するを実行
or “F5”を押す



Error

```python
import cv2
import numpy as np
import os

###################################################################################
#Extracting individual channels and segmenting out area of interest
for file in os.listdir('noBoundry/apathy16'):
    img = cv2.imread('noBoundry/apathy16/'+file)
    b,g,r = cv2.split(img)

    newImg = np.zeros((469,678))

    for i in range(0,len(r)):
        for j in range(0,len(r[i])):
            if(g[i][j] >= 250):
                newImg[i][j] = 255

    cv2.imwrite('ds/apathy16/'+file,newImg)
###################################################################################



###################################################################################
#Feature extraction step from the segmented image
X_train = []
X_test = []
y_train = []
y_test = []

types = ['train','test']
classes = ['apathy','noapathy']

for t in types:
    for c in classes:
        files = os.listdir('combinedBest/'+t+'/'+c)
        for file in files:
            print("Extracting features from: "+file)
            img = cv2.imread('combinedBest/'+t+'/'+c+'/'+file,0)

            #segment variable: 1st 4 values signify column frequency
            #segment variable: last 4 values signify row frequencey
            segments = []
            segments.append(img[:,0:169])
            segments.append(img[:,169:339])
            segments.append(img[:,339:508])
            segments.append(img[:,508:])

            segments.append(img[0:117,:])
            segments.append(img[117:236,:])
            segments.append(img[236:353,:])
            segments.append(img[353:,:])

            vector = []
            for segment in segments:
                count=0
                for i in segment:
                    for j in i:
                        if(j == 255):
```

```
Python 3.7.6 (default,
Type "copyright", "credits" or                          e information.

IPython 7.12.0 -- An enhanced                      ython.

In [1]: runfile('C:/Users/ih          /main1.py', wdir='C:/Users/ihpc/PROJECT')
Traceback (most recent call

  File "C:\Users\ihpc\PRO        \main1.py", line 1, in <module>
    import cv2

ModuleNotFoundError: No module named 'cv2'


In [2]:
```

使用法

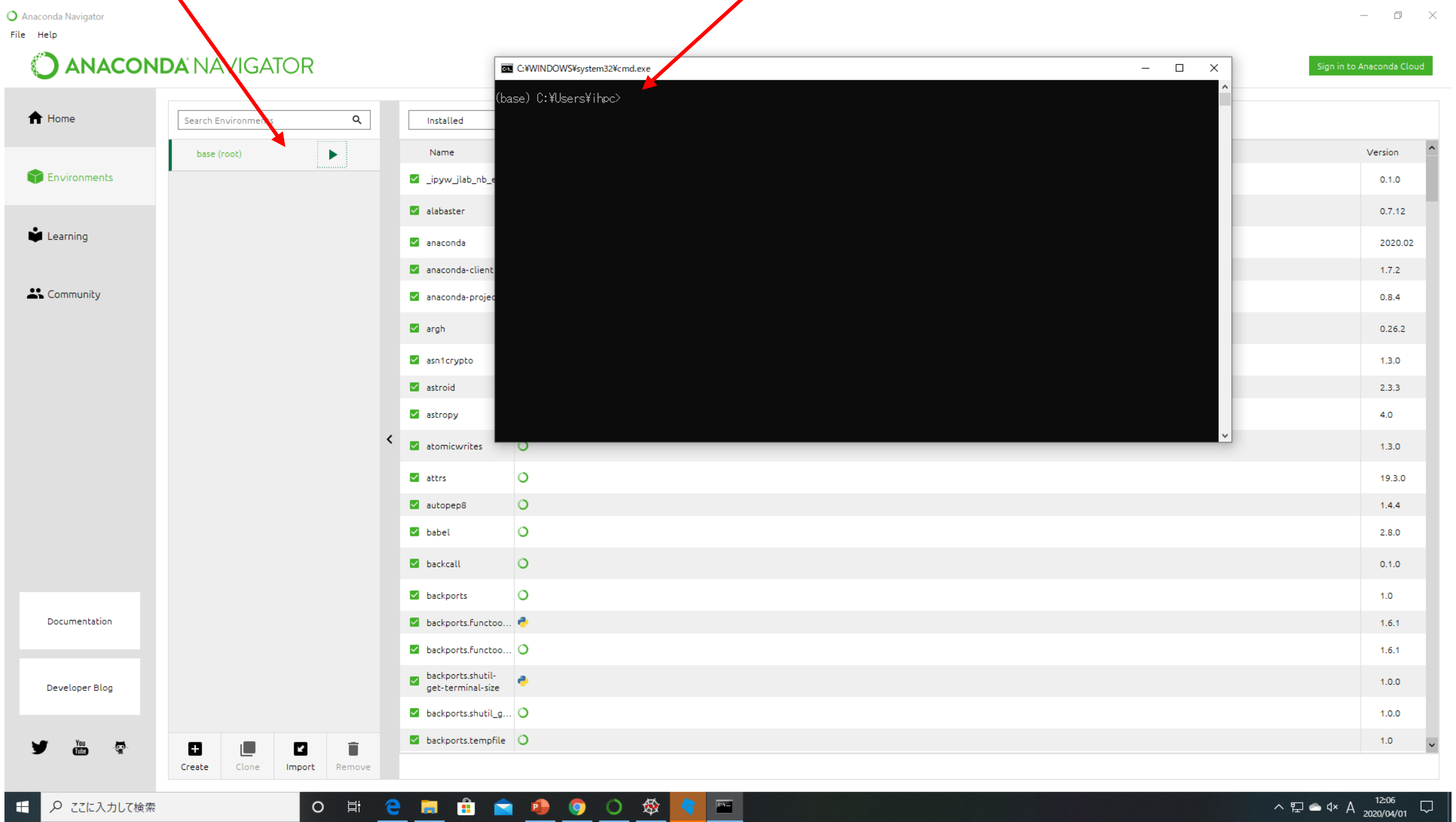
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in 設定 > ヘルプ.

Spyderは初めてですか？ 読んでください チュートリアル

エラー１：no module of “cv”

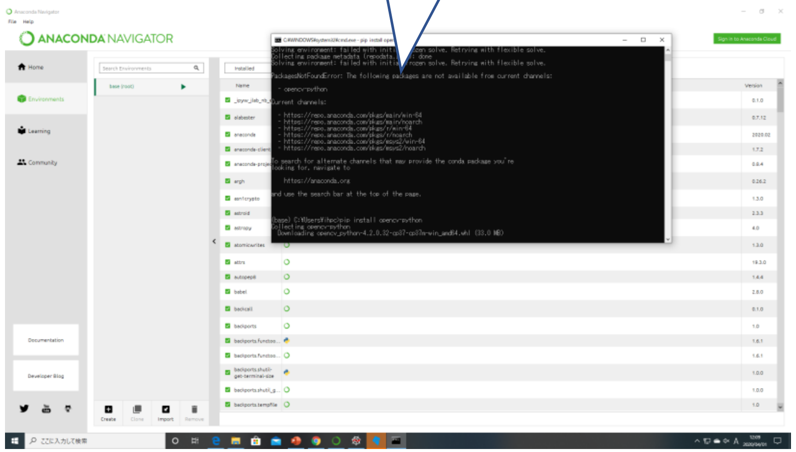
- Environments => base(root)->open terminal
- “Pip install opencv-python “を入力し、実行する
  - エラーの原因は、opencvがインストールされていない



```
(base) C:¥Users¥ihpc>
```

Name	Version
_ipyw_jlab_nb_e	0.1.0
alabaster	0.7.12
anaconda	2020.02
anaconda-client	1.7.2
anaconda-projec	0.8.4
argh	0.26.2
asn1crypto	1.3.0
astroid	2.3.3
astropy	4.0
atomicwrites	1.3.0
attrs	19.3.0
autopep8	1.4.4
babel	2.8.0
backcall	0.1.0
backports	1.0
backports.functoo...	1.6.1
backports.functoo...	1.6.1
backports.shutil-get-terminal-size	1.0.0
backports.shutil_g...	1.0.0
backports.tempfile	1.0

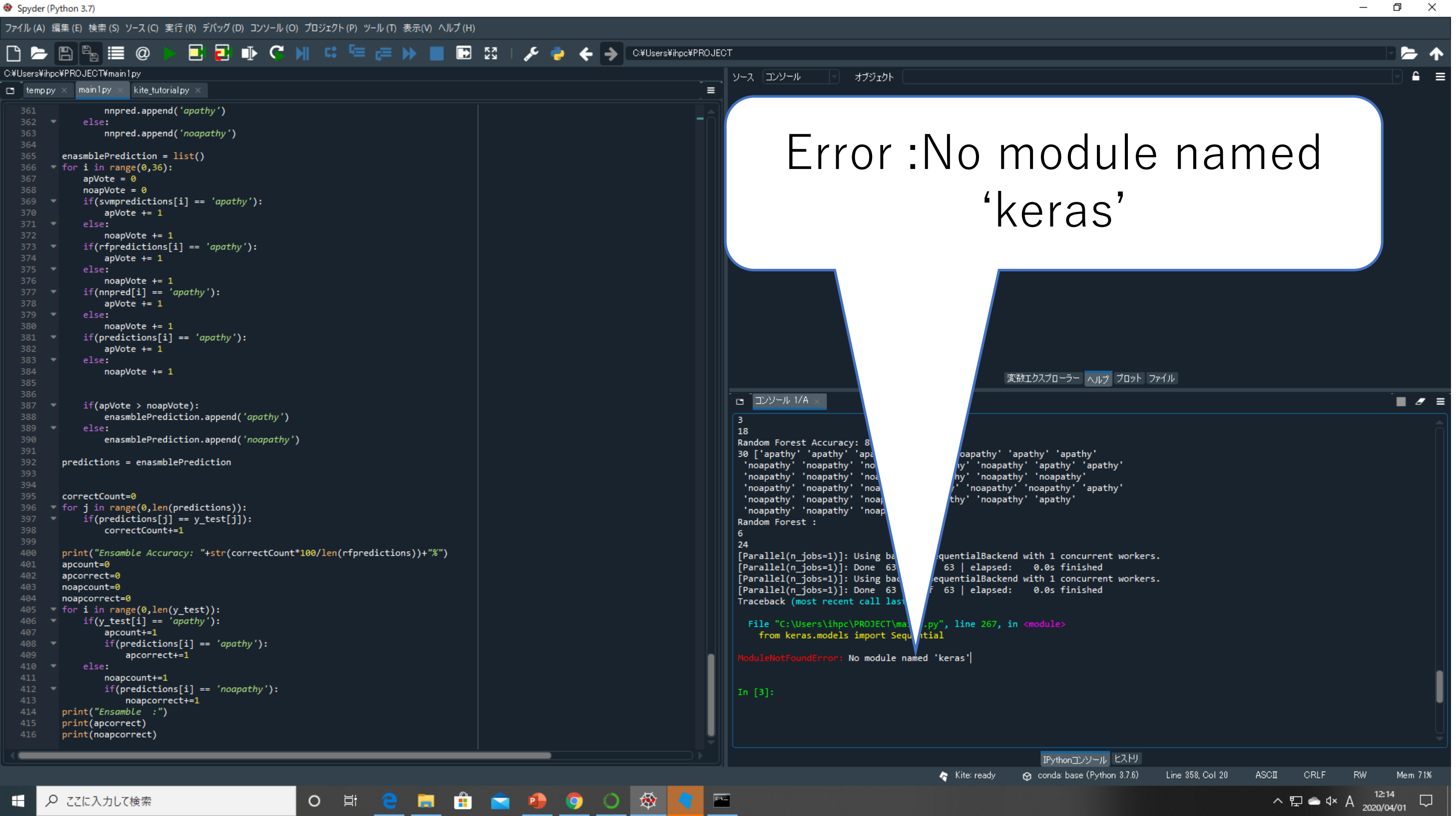
opencv-python を Installの様子



```
Solving environment: failed with initi    ozen solve. Retrying with flexible solve.
Collecting package metadata (repodata.j   ): done
Solving environment: failed with initia   rozen solve. Retrying with flexible solve.

PackagesNotFoundError: The following packages are not available from current channels:

  - opencv-python

Current channels:

  - https://repo.anaconda.com/pkgs/main/win-64
  - https://repo.anaconda.com/pkgs/main/noarch
  - https://repo.anaconda.com/pkgs/r/win-64
  - https://repo.anaconda.com/pkgs/r/noarch
  - https://repo.anaconda.com/pkgs/msys2/win-64
  - https://repo.anaconda.com/pkgs/msys2/noarch

To search for alternate channels that may provide the conda package you're
looking for, navigate to

    https://anaconda.org

and use the search bar at the top of the page.



(base) C:¥Users¥ihpc>pip install opencv-python
Collecting opencv-python
  Downloading opencv_python-4.2.0.32-cp37-cp37m-win_amd64.whl (33.0 MB)
```

エラー2：no module of “keras”

- 再実行すると、”no module of “Keras”のエラーが出ます。
  - 原因：Kerasがインストールされていない


Spyder (Python 3.7)
ファイル (A) 編集 (E) 検索 (S) ソース (C) 実行 (R) デバッグ (D) コンソール (O) プロジェクト (P) ツール (T) 表示(V) ヘルプ (H)
C:¥Users¥ihpc¥PROJECT
C:¥Users¥ihpc¥PROJECT¥main1.py
temp.py
main1.py
kite_tutorial.py
ソース コンソール オブジェクト
変数エクスプローラー ヘルプ プロット ファイル
コンソール 1/A
IPythonコンソール ヒストリ
Kite: ready
conda: base (Python 3.7.6)
Line 358, Col 20
ASCII
CRLF
RW
Mem 71%
ここに入力して検索
12:14
2020/04/01

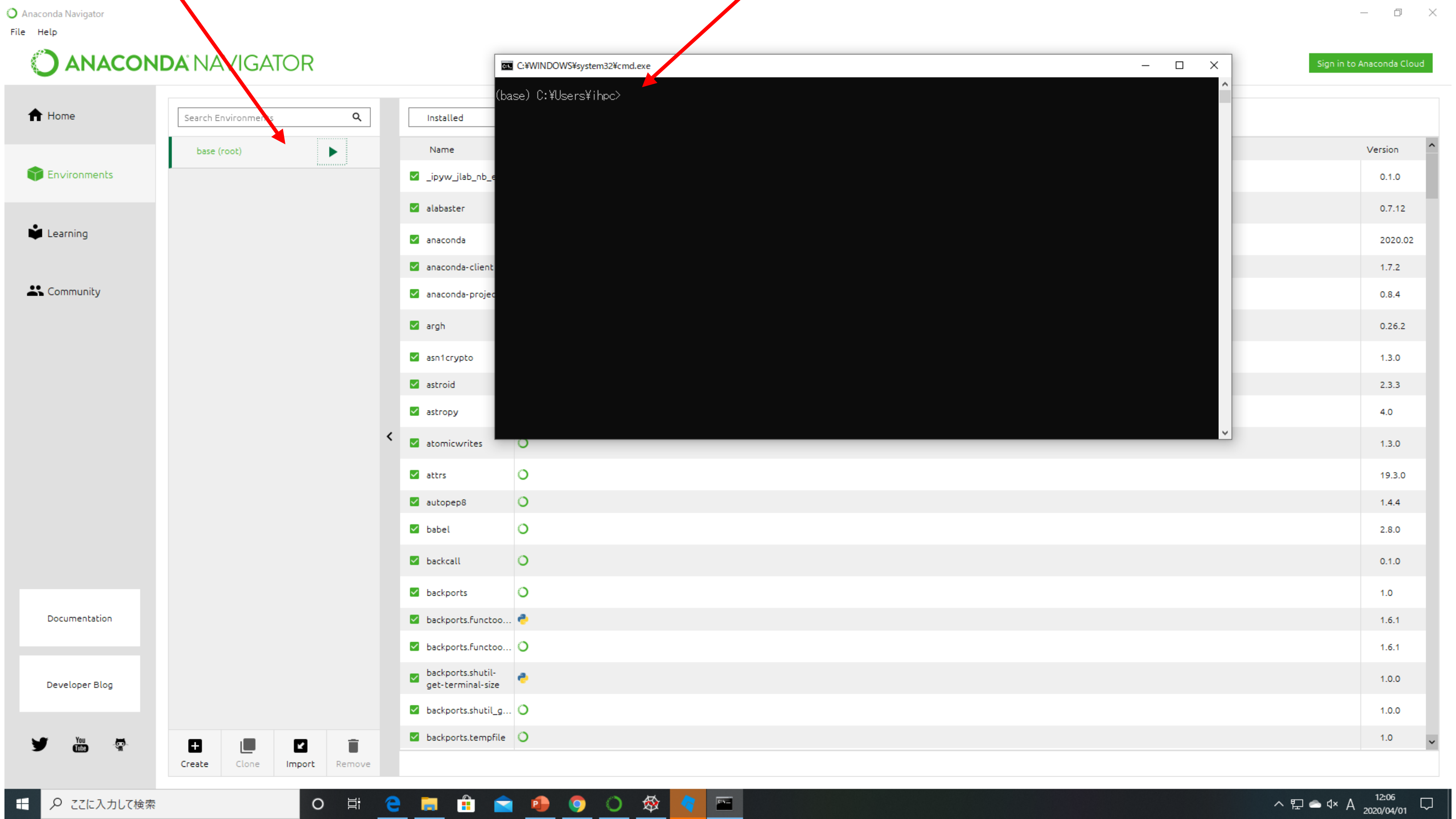
```python
            nnpred.append('apathy')
        else:
            nnpred.append('noapathy')

enasmblePrediction = list()
for i in range(0,36):
    apVote = 0
    noapVote = 0
    if(svmpredictions[i] == 'apathy'):
        apVote += 1
    else:
        noapVote += 1
    if(rfpredictions[i] == 'apathy'):
        apVote += 1
    else:
        noapVote += 1
    if(nnpred[i] == 'apathy'):
        apVote += 1
    else:
        noapVote += 1
    if(predictions[i] == 'apathy'):
        apVote += 1
    else:
        noapVote += 1


    if(apVote > noapVote):
        enasmblePrediction.append('apathy')
    else:
        enasmblePrediction.append('noapathy')

predictions = enasmblePrediction


correctCount=0
for j in range(0,len(predictions)):
    if(predictions[j] == y_test[j]):
        correctCount+=1

print("Ensamble Accuracy: "+str(correctCount*100/len(rfpredictions))+"%")
apcount=0
apcorrect=0
noapcount=0
noapcorrect=0
for i in range(0,len(y_test)):
    if(y_test[i] == 'apathy'):
        apcount+=1
        if(predictions[i] == 'apathy'):
            apcorrect+=1
    else:
        noapcount+=1
        if(predictions[i] == 'noapathy'):
            noapcorrect+=1
print("Ensamble  :")
print(apcorrect)
print(noapcorrect)
```

```
3
18
Random Forest Accuracy: 8
30 ['apathy' 'apathy' 'apa                   oapathy' 'apathy' 'apathy'
 'noapathy' 'noapathy' 'no                 hy' 'noapathy' 'apathy' 'apathy'
 'noapathy' 'noapathy' 'noa                hy' 'noapathy' 'noapathy'
 'noapathy' 'noapathy' 'noa               y' 'noapathy' 'noapathy' 'apathy'
 'noapathy' 'noapathy' 'noap              thy' 'noapathy' 'apathy'
 'noapathy' 'noapathy' 'noap
Random Forest :
6
24
[Parallel(n_jobs=1)]: Using ba        quentialBackend with 1 concurrent workers.
[Parallel(n_jobs=1)]: Done  63         63 | elapsed:    0.0s finished
[Parallel(n_jobs=1)]: Using bac       equentialBackend with 1 concurrent workers.
[Parallel(n_jobs=1)]: Done  63      f  63 | elapsed:    0.0s finished
Traceback (most recent call last

  File "C:\Users\ihpc\PROJECT\ma    .py", line 267, in <module>
    from keras.models import Sequ  ntial

ModuleNotFoundError: No module named 'keras'


In [3]:
```

Error :No module named ‘keras’

エラー１：no module of “cv”

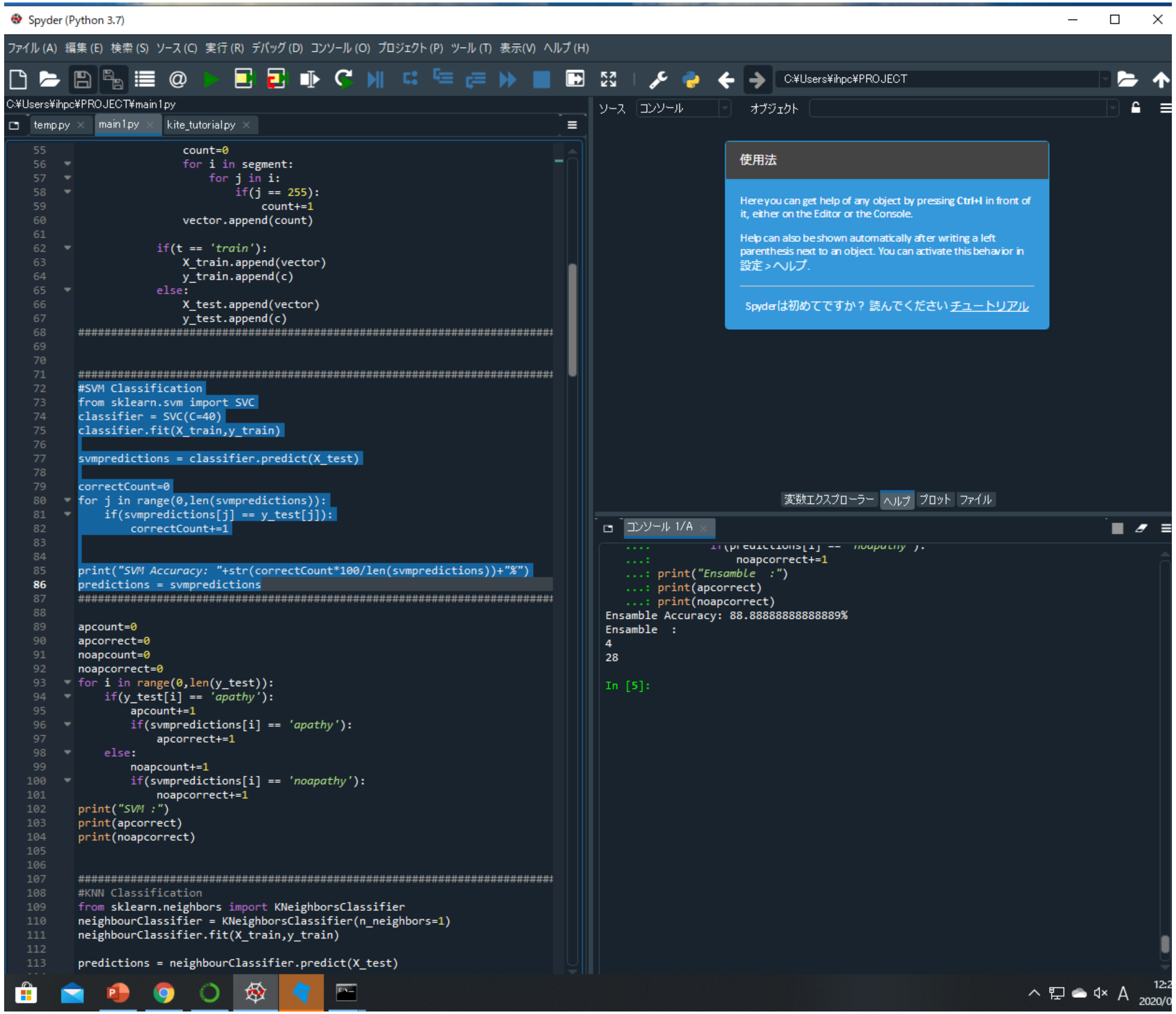
- Environments => base(root)->open terminal
- “conda install teras “を入力し、実行する



```
(base) C:¥Users¥ihpc>
```

Name	Version
☑ _ipyw_jlab_nb_e	0.1.0
☑ alabaster	0.7.12
☑ anaconda	2020.02
☑ anaconda-client	1.7.2
☑ anaconda-projec	0.8.4
☑ argh	0.26.2
☑ asn1crypto	1.3.0
☑ astroid	2.3.3
☑ astropy	4.0
☑ atomicwrites	1.3.0
☑ attrs	19.3.0
☑ autopep8	1.4.4
☑ babel	2.8.0
☑ backcall	0.1.0
☑ backports	1.0
☑ backports.functoo…	1.6.1
☑ backports.functoo…	1.6.1
☑ backports.shutil-get-terminal-size	1.0.0
☑ backports.shutil_g…	1.0.0
☑ backports.tempfile	1.0

実行成功の様子



```
                count=0
                for i in segment:
                    for j in i:
                        if(j == 255):
                            count+=1
                vector.append(count)

            if(t == 'train'):
                X_train.append(vector)
                y_train.append(c)
            else:
                X_test.append(vector)
                y_test.append(c)
########################################################################


########################################################################
#SVM Classification
from sklearn.svm import SVC
classifier = SVC(C=40)
classifier.fit(X_train,y_train)

svmpredictions = classifier.predict(X_test)

correctCount=0
for j in range(0,len(svmpredictions)):
    if(svmpredictions[j] == y_test[j]):
        correctCount+=1


print("SVM Accuracy: "+str(correctCount*100/len(svmpredictions))+"%")
predictions = svmpredictions
########################################################################

apcount=0
apcorrect=0
noapcount=0
noapcorrect=0
for i in range(0,len(y_test)):
    if(y_test[i] == 'apathy'):
        apcount+=1
        if(svmpredictions[i] == 'apathy'):
            apcorrect+=1
    else:
        noapcount+=1
        if(svmpredictions[i] == 'noapathy'):
            noapcorrect+=1
print("SVM :")
print(apcorrect)
print(noapcorrect)


########################################################################
#KNN Classification
from sklearn.neighbors import KNeighborsClassifier
neighbourClassifier = KNeighborsClassifier(n_neighbors=1)
neighbourClassifier.fit(X_train,y_train)

predictions = neighbourClassifier.predict(X_test)
```

使用法

Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in 設定 > ヘルプ.

Spyderは初めてですか？ 読んでください チュートリアル

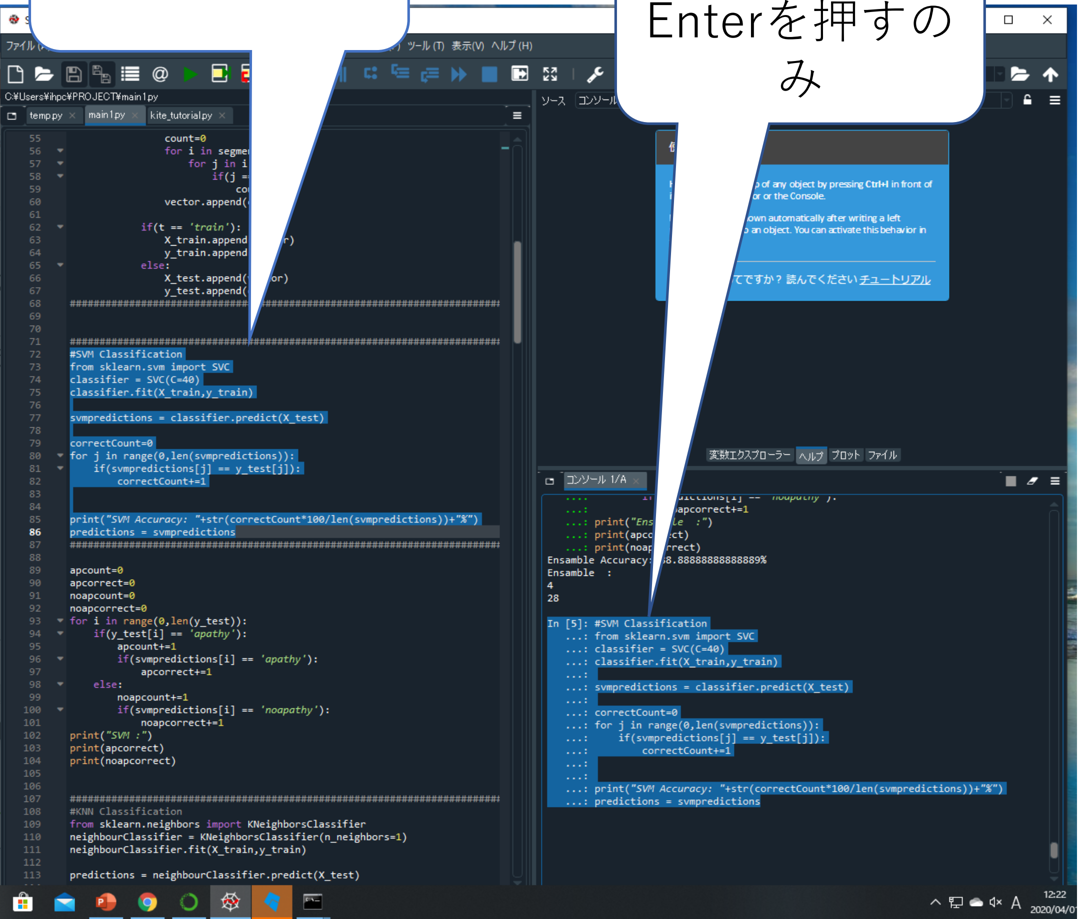
```
   ....:         if(predictions[i] == 'noapathy'):
   ...:             noapcorrect+=1
   ...: print("Ensamble  :")
   ...: print(apcorrect)
   ...: print(noapcorrect)
Ensamble Accuracy: 88.88888888888889%
Ensamble  :
4
28

In [5]:
```

部分の実行と結果

- 一部のコードを切り取って、部分的に実行することができます。

Copy

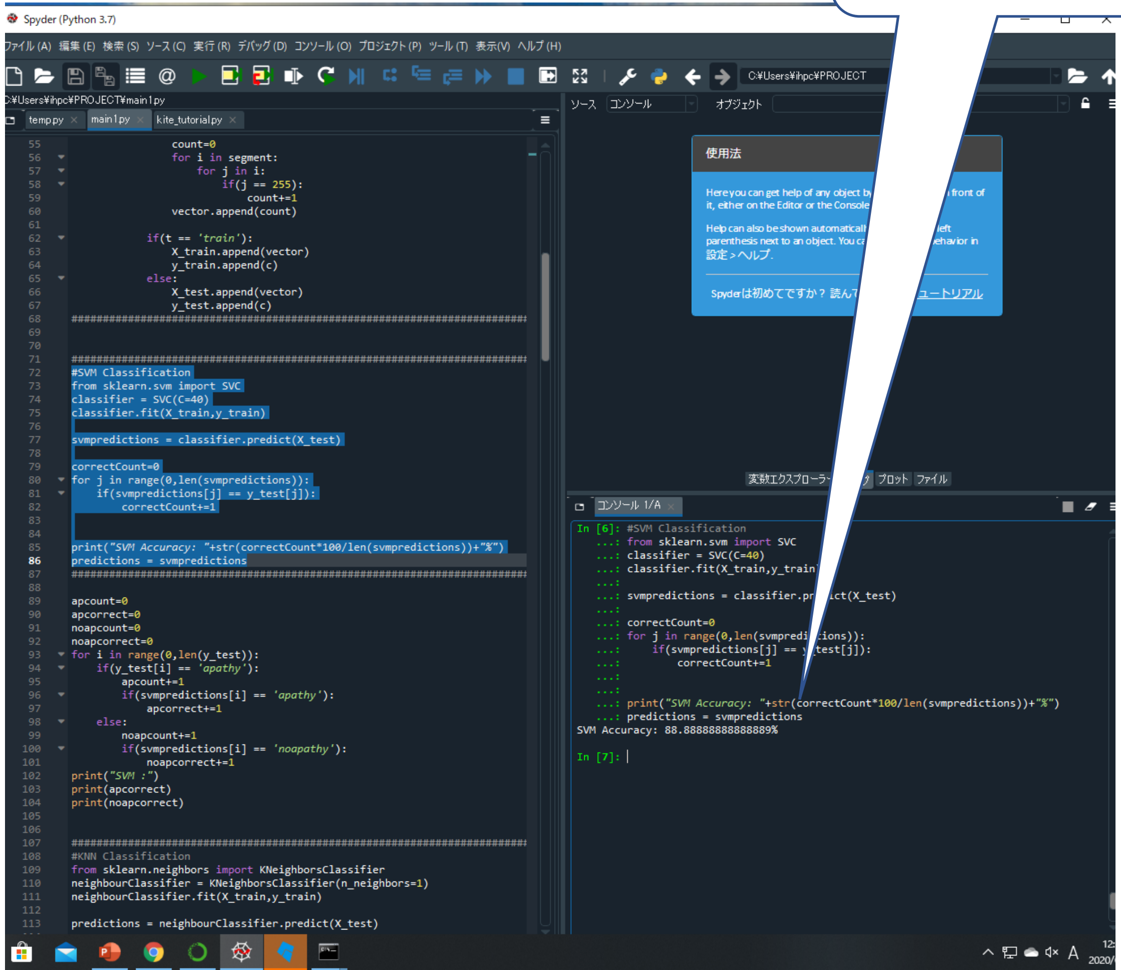
Pasteして、Enterを押すのみ



```
#SVM Classification
from sklearn.svm import SVC
classifier = SVC(C=40)
classifier.fit(X_train,y_train)

svmpredictions = classifier.predict(X_test)

correctCount=0
for j in range(0,len(svmpredictions)):
    if(svmpredictions[j] == y_test[j]):
        correctCount+=1


print("SVM Accuracy: "+str(correctCount*100/len(svmpredictions))+"%")
predictions = svmpredictions
```

```
   ...:                 noapcorrect+=1
   ...: print("Ensamble  :")
   ...: print(apcorrect)
   ...: print(noapcorrect)
Ensamble Accuracy: 88.8888888888889%
Ensamble  :
4
28

In [5]: #SVM Classification
   ...: from sklearn.svm import SVC
   ...: classifier = SVC(C=40)
   ...: classifier.fit(X_train,y_train)
   ...:
   ...: svmpredictions = classifier.predict(X_test)
   ...:
   ...: correctCount=0
   ...: for j in range(0,len(svmpredictions)):
   ...:     if(svmpredictions[j] == y_test[j]):
   ...:         correctCount+=1
   ...:
   ...:
   ...: print("SVM Accuracy: "+str(correctCount*100/len(svmpredictions))+"%")
   ...: predictions = svmpredictions
```

Result



Spyder (Python 3.7)
ファイル (A) 編集 (E) 検索 (S) ソース (C) 実行 (R) デバッグ (D) コンソール (O) プロジェクト (P) ツール (T) 表示 (V) ヘルプ (H)
C:¥Users¥ihpc¥PROJECT
C:¥Users¥ihpc¥PROJECT¥main1.py
temp.py
main1.py
kite_tutorial.py
ソース
コンソール
オブジェクト
使用法
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in 設定 > ヘルプ.
Spyderは初めてですか？ 読んでください チュートリアル
変数エクスプローラー
ヘルプ
プロット
ファイル
コンソール 1/A

```
                count=0
                for i in segment:
                    for j in i:
                        if(j == 255):
                            count+=1
                vector.append(count)

            if(t == 'train'):
                X_train.append(vector)
                y_train.append(c)
            else:
                X_test.append(vector)
                y_test.append(c)
########################################################################


########################################################################
#SVM Classification
from sklearn.svm import SVC
classifier = SVC(C=40)
classifier.fit(X_train,y_train)

svmpredictions = classifier.predict(X_test)

correctCount=0
for j in range(0,len(svmpredictions)):
    if(svmpredictions[j] == y_test[j]):
        correctCount+=1


print("SVM Accuracy: "+str(correctCount*100/len(svmpredictions))+"%")
predictions = svmpredictions
########################################################################

apcount=0
apcorrect=0
noapcount=0
noapcorrect=0
for i in range(0,len(y_test)):
    if(y_test[i] == 'apathy'):
        apcount+=1
        if(svmpredictions[i] == 'apathy'):
            apcorrect+=1
    else:
        noapcount+=1
        if(svmpredictions[i] == 'noapathy'):
            noapcorrect+=1
print("SVM :")
print(apcorrect)
print(noapcorrect)


########################################################################
#KNN Classification
from sklearn.neighbors import KNeighborsClassifier
neighbourClassifier = KNeighborsClassifier(n_neighbors=1)
neighbourClassifier.fit(X_train,y_train)

predictions = neighbourClassifier.predict(X_test)
```

```
In [6]: #SVM Classification
   ...: from sklearn.svm import SVC
   ...: classifier = SVC(C=40)
   ...: classifier.fit(X_train,y_train)
   ...:
   ...: svmpredictions = classifier.predict(X_test)
   ...:
   ...: correctCount=0
   ...: for j in range(0,len(svmpredictions)):
   ...:     if(svmpredictions[j] == y_test[j]):
   ...:         correctCount+=1
   ...:
   ...:
   ...: print("SVM Accuracy: "+str(correctCount*100/len(svmpredictions))+"%")
   ...: predictions = svmpredictions
SVM Accuracy: 88.88888888888889%

In [7]: 
```