

公開研究会

AIの開発環境とその構築手法 — TensorFlow と NN Console を中心に —

筑波大学大学院システム情報工学研究科
安永守利

もちろん、理論的、学術的に進んだ上に. . .

① 衍違いな学習データ (Big Data)

Google (1997) はもちろん、デジカメも無い！

インターネットはメールがやっと普及し始めた頃。

顔画像は20人×10枚/人のデータベース頼り。

ワトソンもインターネットデータがなければ、学習できない。

② 演算速度の向上 (FPGA や GPU の出現)



③ 開発環境 (ex. Google Tensor Flow) の出現



Cloudではないので、
データ流出は無し。

開発環境

	emaincontri ビューター	対応言語	Github Star*1	公開時期 *2
Tensorflow	Google	<u>Python</u> , C++, Java, Go	68.9k	2015.11
Caffe	カリフォルニア大 学 バークレー校	C, C++	19.6k	2014.11
Caffe2	Facebook	<u>Python</u> , C++	5.4k	2015.12
CNTK	Microsoft	<u>Python</u> , C++, C#, .net, BrainScript	12k	2016.1
MXNet	ワシントン大学	<u>Python</u> , Scala, R, C++, Julia, Perl	11.0k	2015.10
torch7	Facebook*3	Lua	7.2k	2014.8
PyTorch	Facebook	<u>Python</u>	6.6k	2017.1
DL4J	Skymind	Java	7.0k	2014.8
theano	MILA*4	<u>Python</u>	6.8k	2013.12
Chainer	Preferred Networks	<u>Python</u>	2.8k	2015.6



[Image Source: IEEE Spectrum]

We use cookies to ensure you get the best experience on our website. If you continue using our website, we'll assume that you are happy to receive all cookies on the Interesting Engineering website. Please [review our Privacy Policy](#) for more information and/or to change your cookie settings at any time.

IEEE Spectrum Names the Top 10 Programming Languages Of 2017

現状から考えると

開発プログラミング言語: Python

開発環境: Tensor Flow

が主流.

なお、開発プログラミング言語としては、

MATLAB (マスワークス)

もあり。さらに、開発環境としては、

プログラミング不要で
DLやNNを試せます！

Neural Network Console

<https://dl.sony.com/ja/>

(ソニー)

GCP(Google Cloud Platform) ⇒有料

もあり。

何故、Python か

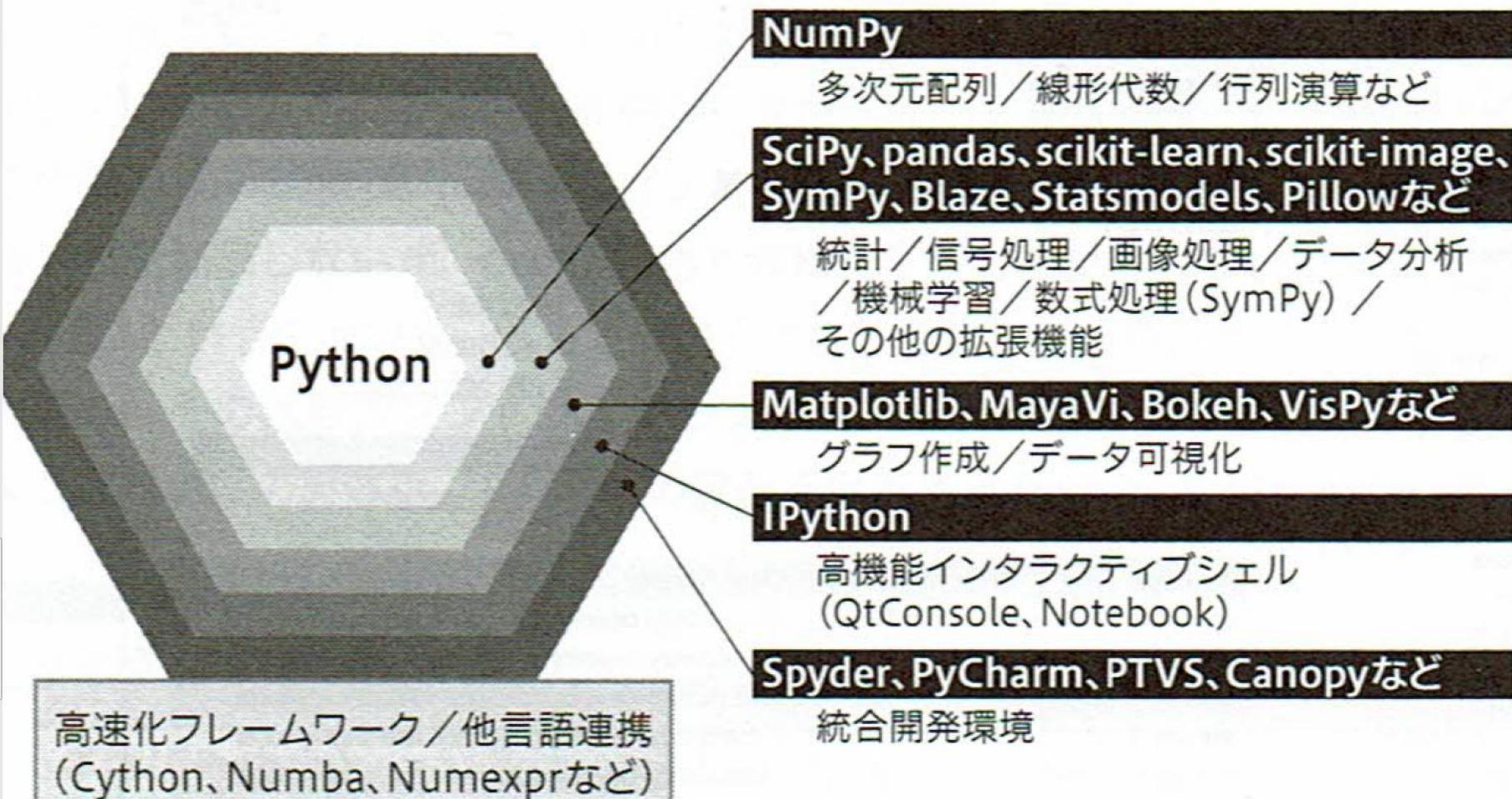
- ・充実したパッケージ
(特にデータサイエンスや科学技術計算)
- ・わかりやすい⇒教育用言語としても良い

Python の特徴

- ・オブジェクト指向言語
- ・インタプリタ言語
- ・(スクリプト言語)

Python の注意事項(1点だけ)

- ・Python 2.x と Python 3.x で互換性に若干の問題あり(ライブラリの対応が異なる).
- ・今からやるなら, python 3.x がよい.
⇒ python 2.x で作られたプログラムをpython 3.x で動かすには
ちょっと手直しが必要.



代表的な無償の
ディストリビューションパッケージ

Anaconda
Enthought Canopy

WinPython
Python(x,y)

Tensorflowについて

<https://www.tensorflow.org/>

The screenshot shows the official TensorFlow website. At the top, there's a navigation bar with links like 'Install', 'Develop', 'API v1.8', 'Deploy', 'Extend', 'Community', 'Versions', 'Ecosystem', 'GitHub', and a search bar. Below the header, a large orange banner features the text 'An open source machine learning framework for everyone' and a 'GET STARTED' button. The main content area has three sections: one about TensorFlow 1.8, one about the Dev Summit 2018, and one about TensorFlow.js.

About TensorFlow

TensorFlow™ is an open source software library for high performance numerical computation. Its flexible architecture allows easy deployment of computation across a variety of platforms (CPUs, GPUs, TPUs), and from desktops to clusters of servers to mobile and edge devices. Originally developed by researchers and engineers from the Google Brain team within Google's AI organization, it comes with strong support for machine learning and deep learning and the flexible numerical computation core is used across many other scientific domains.



Installing TensorFlow



We've built and tested TensorFlow on the following 64-bit laptop/desktop operating systems:

- macOS 10.12.6 (Sierra) or later.
- Ubuntu 16.04 or later
- Windows 7 or later.

Although you might be able to install TensorFlow on other laptop or desktop systems, we only support (and only fix issues in) the preceding configurations.

The following guides explain how to install a version of TensorFlow that enables you to write applications in Python:

- [Installing TensorFlow on Ubuntu](#)
- [Installing TensorFlow on macOS](#)
- [Installing TensorFlow on Windows](#)
- [Installing TensorFlow from Sources](#)

Many aspects of the Python TensorFlow API changed from version 0.n to 1.0. The following guide explains how to migrate older TensorFlow applications to Version 1.0:

- [Transitioning to TensorFlow 1.0](#)

The following guides explain how to install TensorFlow libraries for use in other programming languages. These APIs are aimed at deploying TensorFlow models in applications and are not as extensive as the Python APIs.

- [Installing TensorFlow for Java](#)
- [Installing TensorFlow for C](#)
- [Installing TensorFlow for Go](#)

[Installing TensorFlow](#)[Python](#)[Ubuntu](#)[MacOS](#)[Windows](#)[From source](#)[Transitioning to TensorFlow 1.0](#)[Other Languages](#)[Java](#)[Go](#)[C](#)

Installing TensorFlow on Windows



This guide explains how to install TensorFlow on Windows. Although these instructions might also work on other Windows variants, we have only tested (and we only support) these instructions on machines meeting the following requirements:

- 64-bit, x86 desktops or laptops
- Windows 7 or later

Determine which TensorFlow to install

You must choose one of the following types of TensorFlow to install:

- **TensorFlow with CPU support only.** If your system does not have a NVIDIA® GPU, you must install this version. Note that this version of TensorFlow is typically much easier to install (typically, in 5 or 10 minutes), so even if you have an NVIDIA GPU, we recommend installing this version first. Prebuilt binaries will use AVX instructions.
- **TensorFlow with GPU support.** TensorFlow programs typically run significantly faster on a GPU than on a CPU. Therefore, if your system has a NVIDIA® GPU meeting the prerequisites shown below and you need to run performance-critical applications, you should ultimately install this version.

Anaconda とは

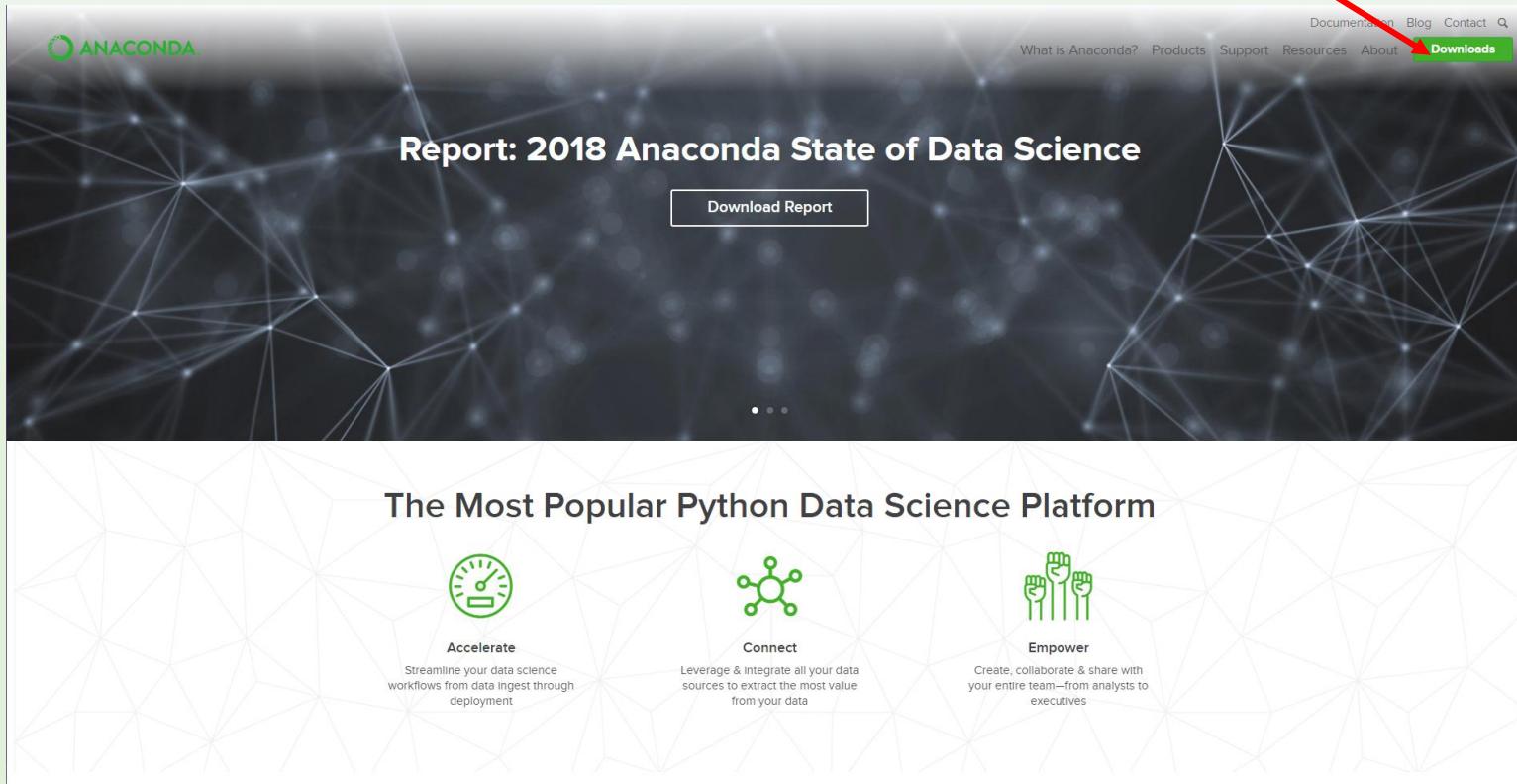
代表的な Python の

無償ディストリビューションパッケージ

TensorFlow 他を簡単・分かりやすくインストールできます。
まずは、Anaconda をインストールします。

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

ダウンロードをクリック



Anaconda で Python 現 × Downloads - Anaconda x

保護された通信 | <https://www.anaconda.com/download/>

Documentation Blog Contact Anaconda Cloud

What is Anaconda? Products Support Resources About Downloads

Download Anaconda Distribution

Version 5.2 | Release Date: May 30, 2018

Download For:   

High-Performance Distribution

Easily install 1,000+ [data science packages](#)

Package Management

Manage packages, dependencies and environments with [conda](#)

Portal to Data Science

Uncover insights in your data and create interactive visualizations

 Windows  macOS  Linux

Anaconda 5.2 For Windows Installer

12:50 2018/06/27

Python 3.6 (or Python 3.7) をDown Load してください.

The screenshot shows the Anaconda download page for Windows. A red arrow points from the top right towards the 'Python 3.6 version' section. The 'Python 3.6 version' section is highlighted with a red box. It contains the text 'Anaconda 5.2 For Windows Installer' and a green 'Download' button. Below the button are links for '64-Bit Graphical Installer (631 MB)' and '32-Bit Graphical Installer (506 MB)'. To the right, there is a section for 'Python 2.7 version' with its own 'Download' button and file links. At the bottom, there are links for 'Behind a firewall?', 'How to get Python 3.5 or other Python versions', and 'How to Install ANACONDA'.

Documentation Blog Contact Anaconda Cloud

What is Anaconda? Products Support Resources About Downloads

Windows macOS Linux

Anaconda 5.2 For Windows Installer

Python 3.6 version *

Download

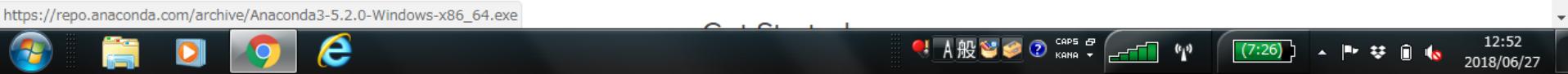
64-Bit Graphical Installer (631 MB) ②
32-Bit Graphical Installer (506 MB)

Python 2.7 version *

Download

64-Bit Graphical Installer (564 MB) ②
32-Bit Graphical Installer (443 MB)

Behind a firewall?
[How to get Python 3.5 or other Python versions](#)
[How to Install ANACONDA](#)



メールアドと職業の登録(個人情報の登録はそれだけです).

Anaconda で Python 現 × Downloads - Anaconda ×

保護された通信 | <https://www.anaconda.com/download/#windows>

Documentation Blog Contact Anaconda Cloud

What is Anaconda? Products Support Resources About Downloads

Thank You for Downloading Anaconda!

Get Started with the Anaconda Cheat Sheet

Work Email *

ac.jp

Job Title: *

Designer

Get the Cheat Sheet

Your information will be processed according to Anaconda's [Privacy Policy](#)

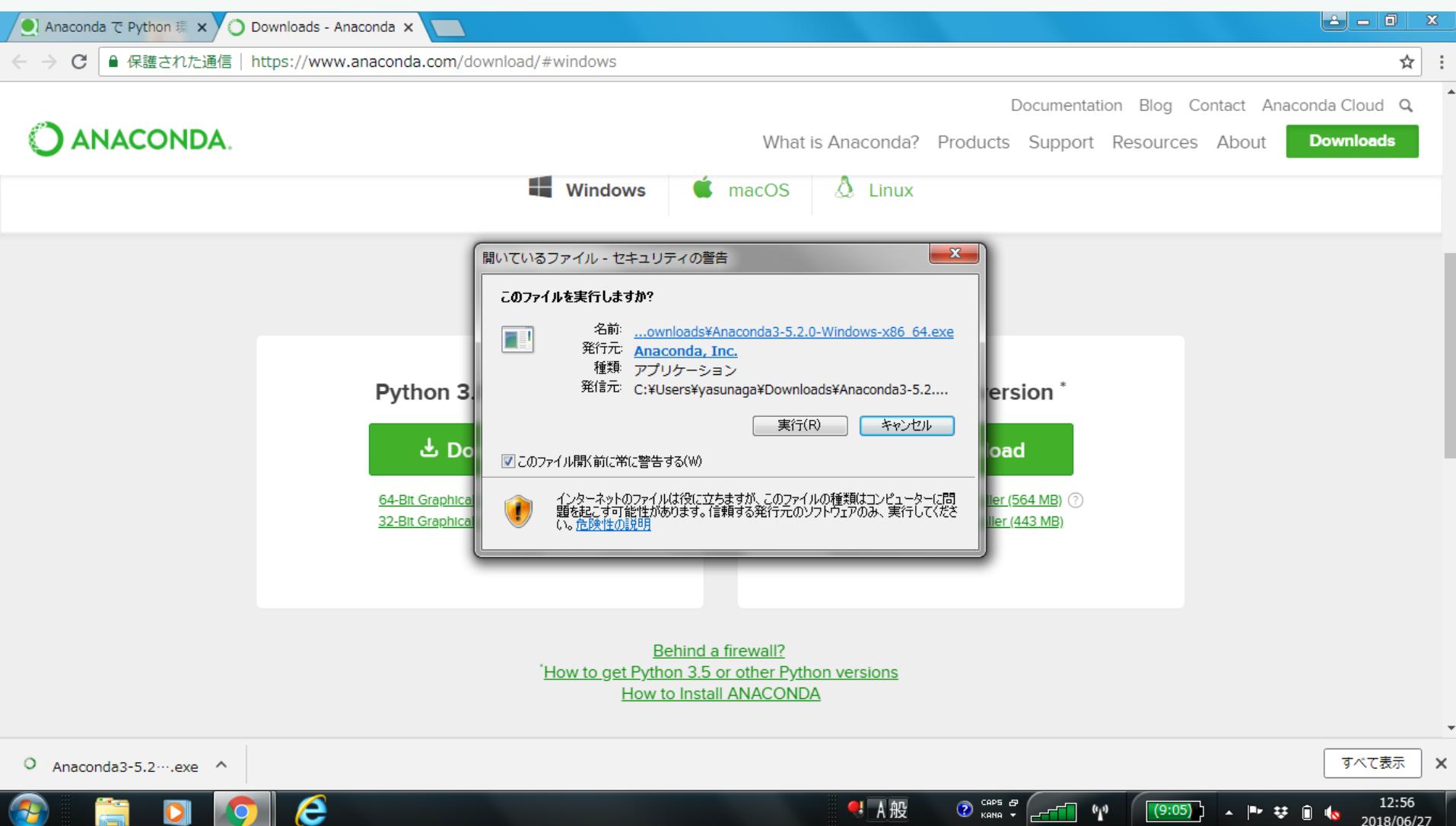
Behind a firewall?
[How to get Python 3.5 or other Python versions](#)
[How to Install ANACONDA](#)

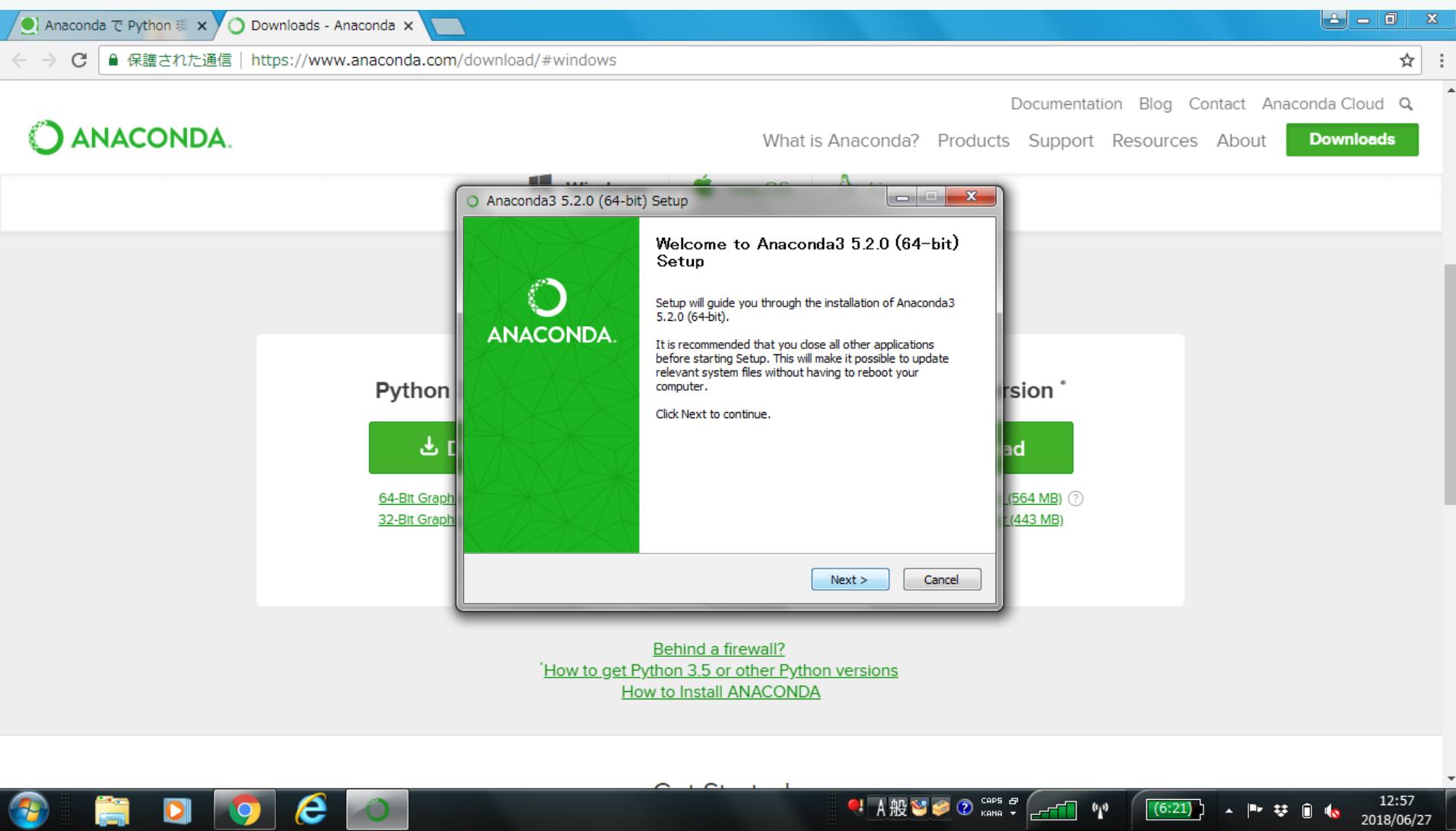
Anaconda3-5.2...exe 232/631 MB、残り 1 分

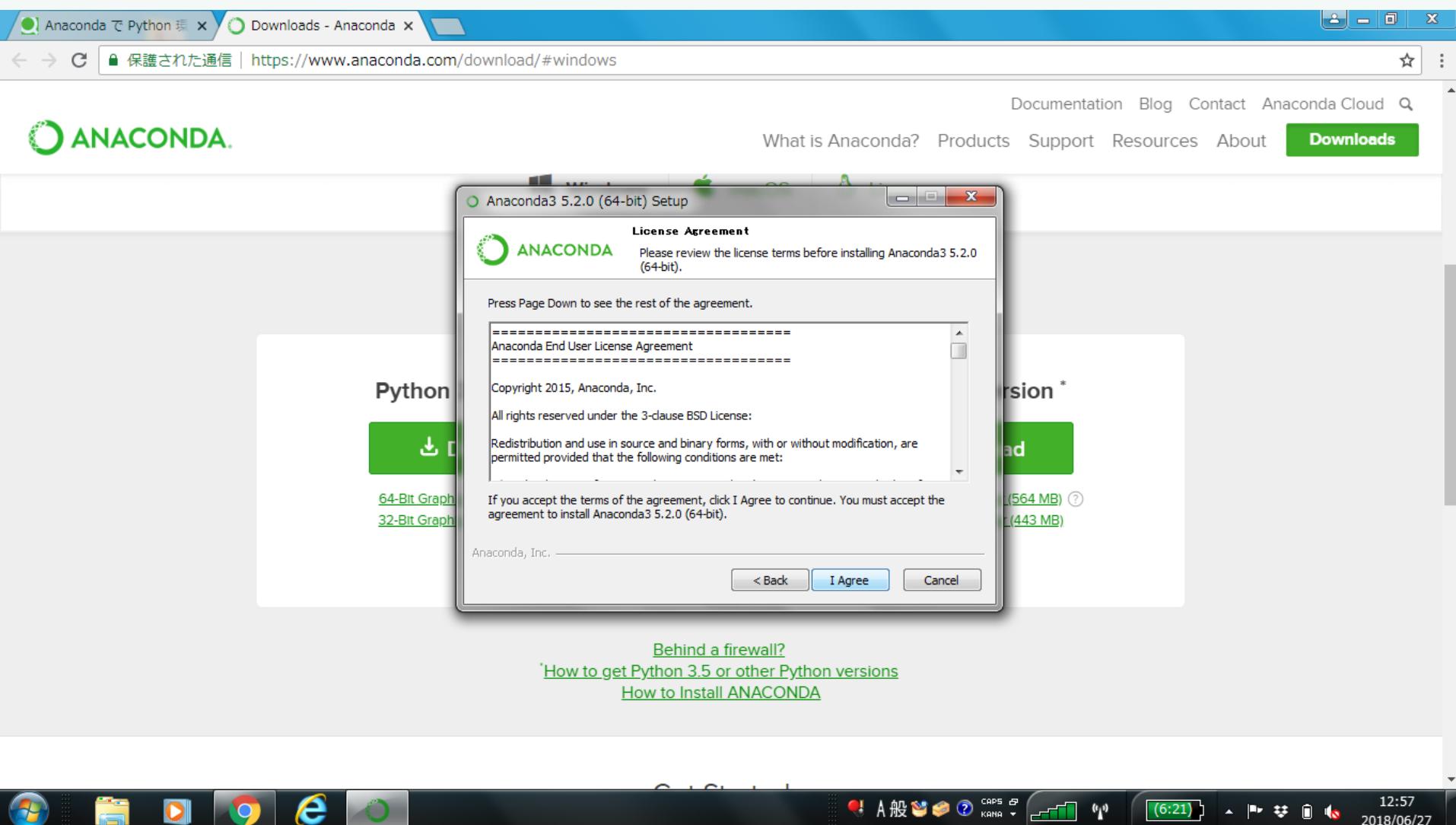
すべて表示

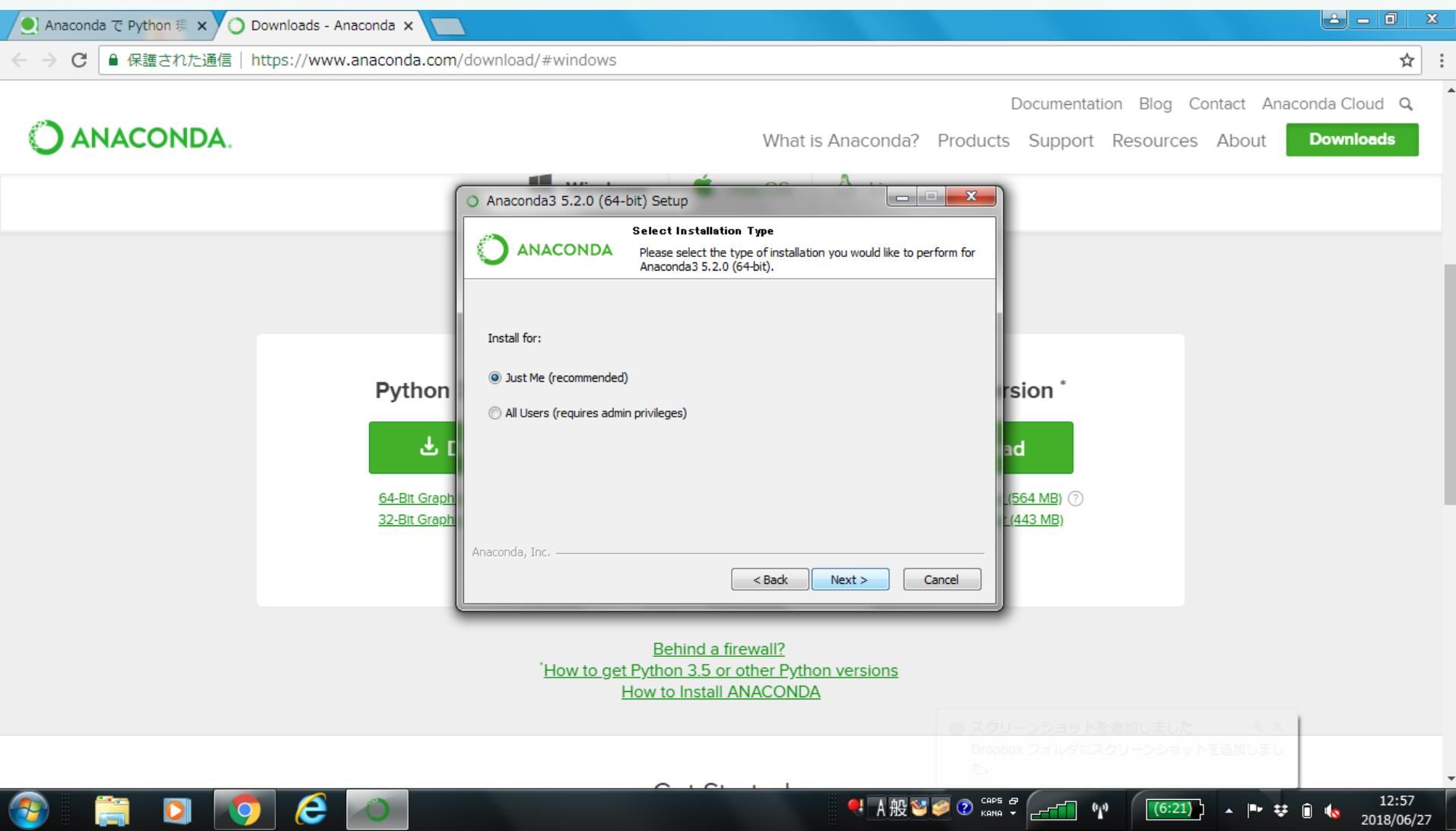
12:53 2018/06/27

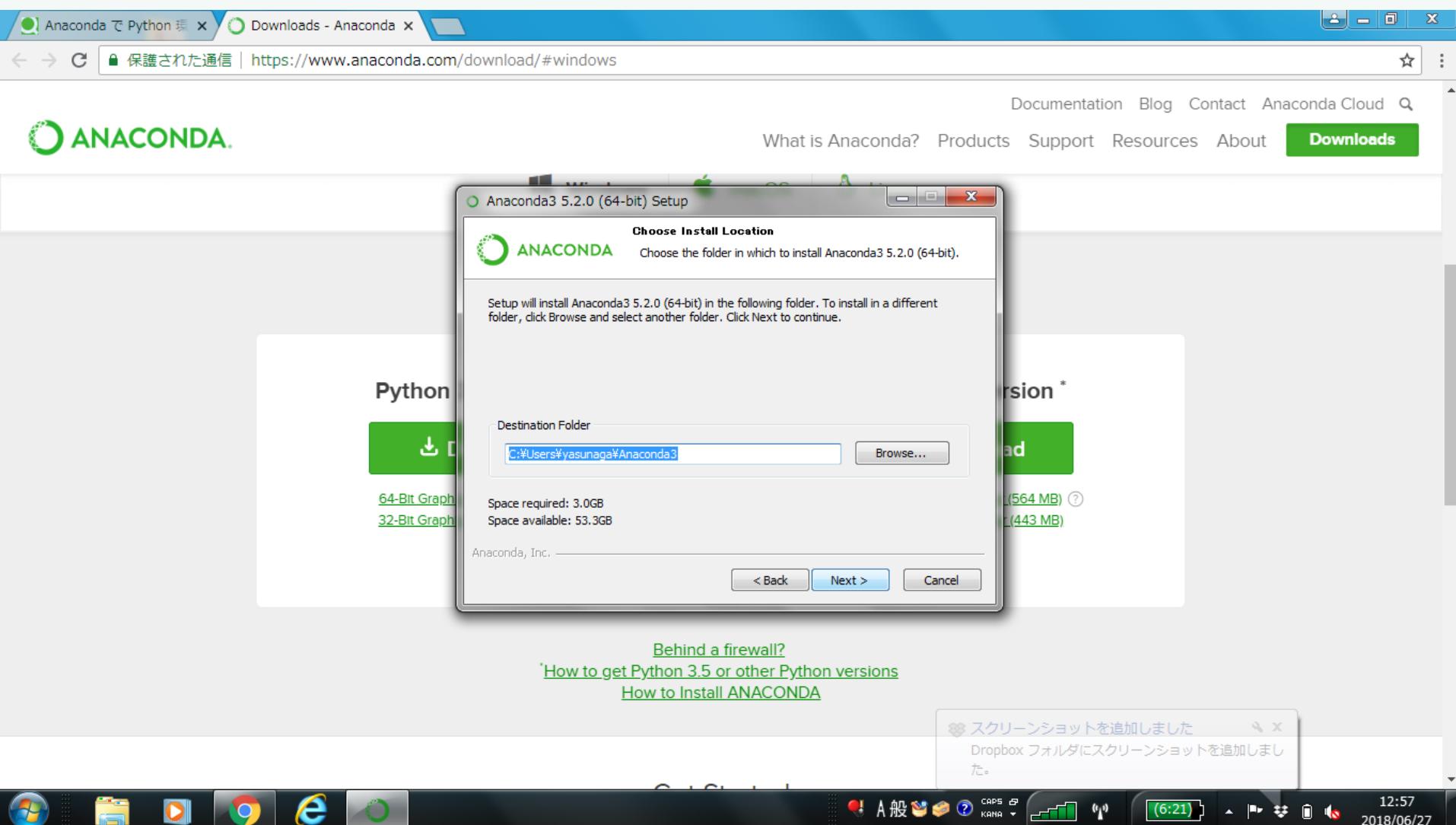
指示に従ってインストール.

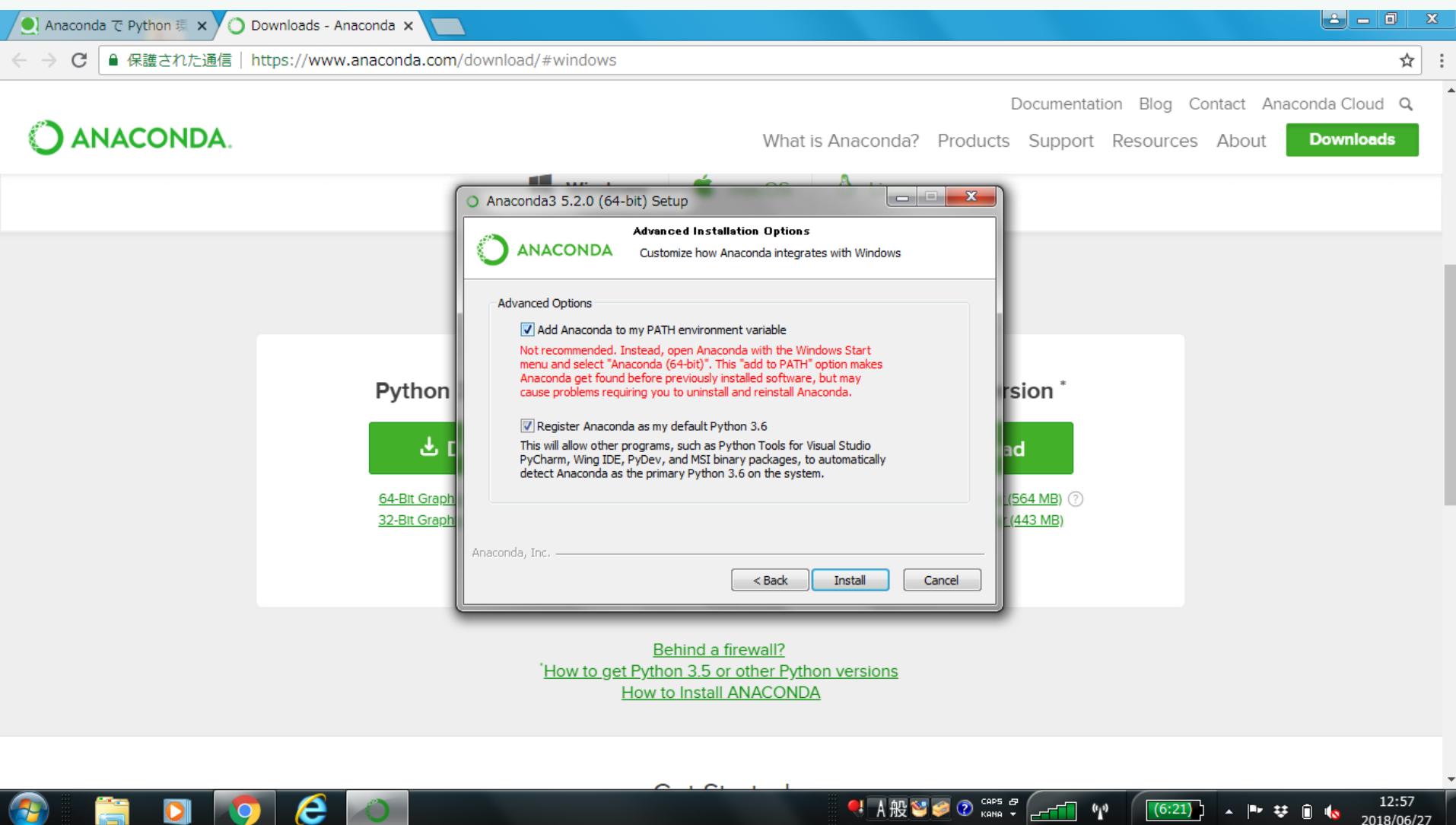


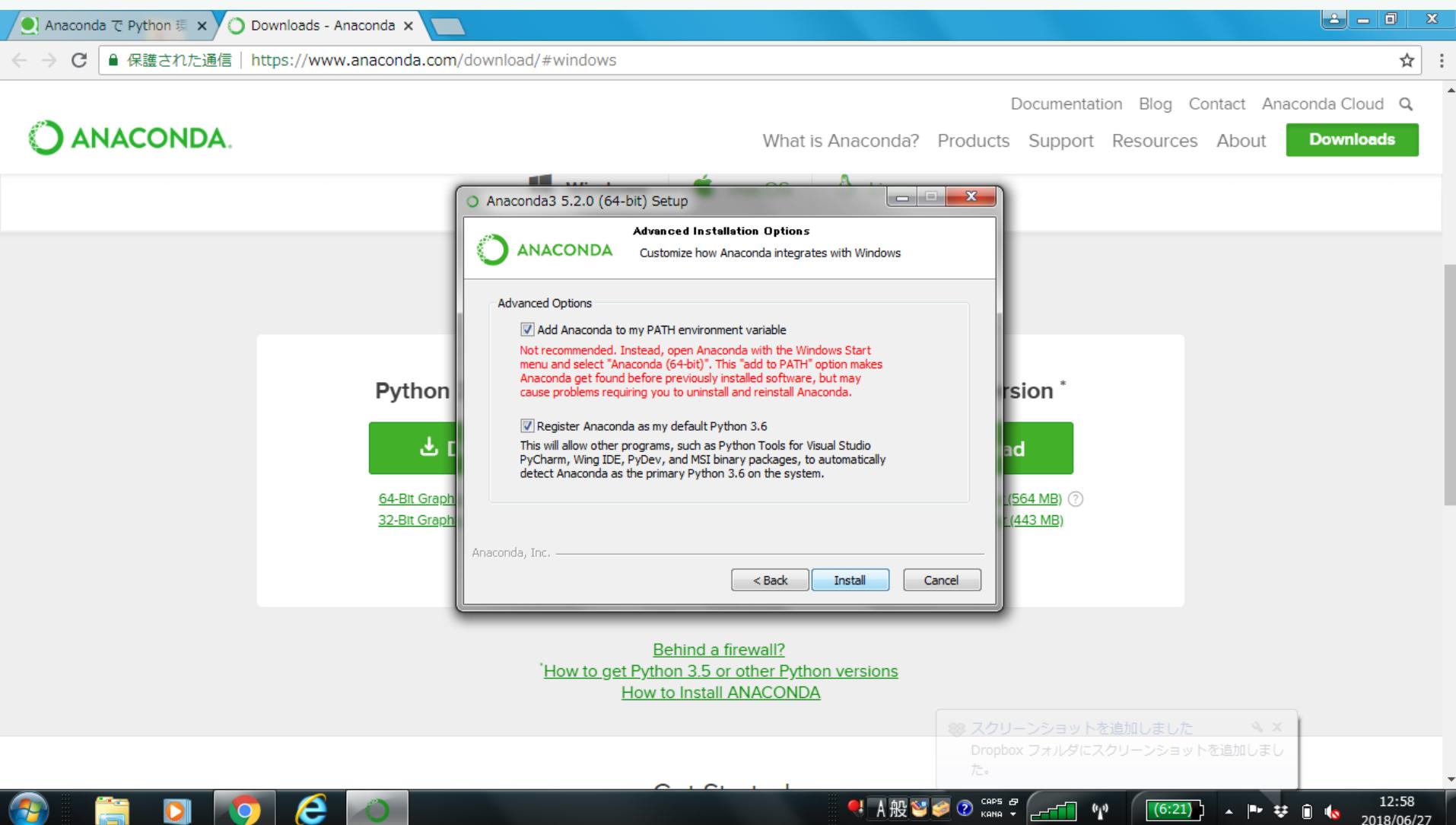




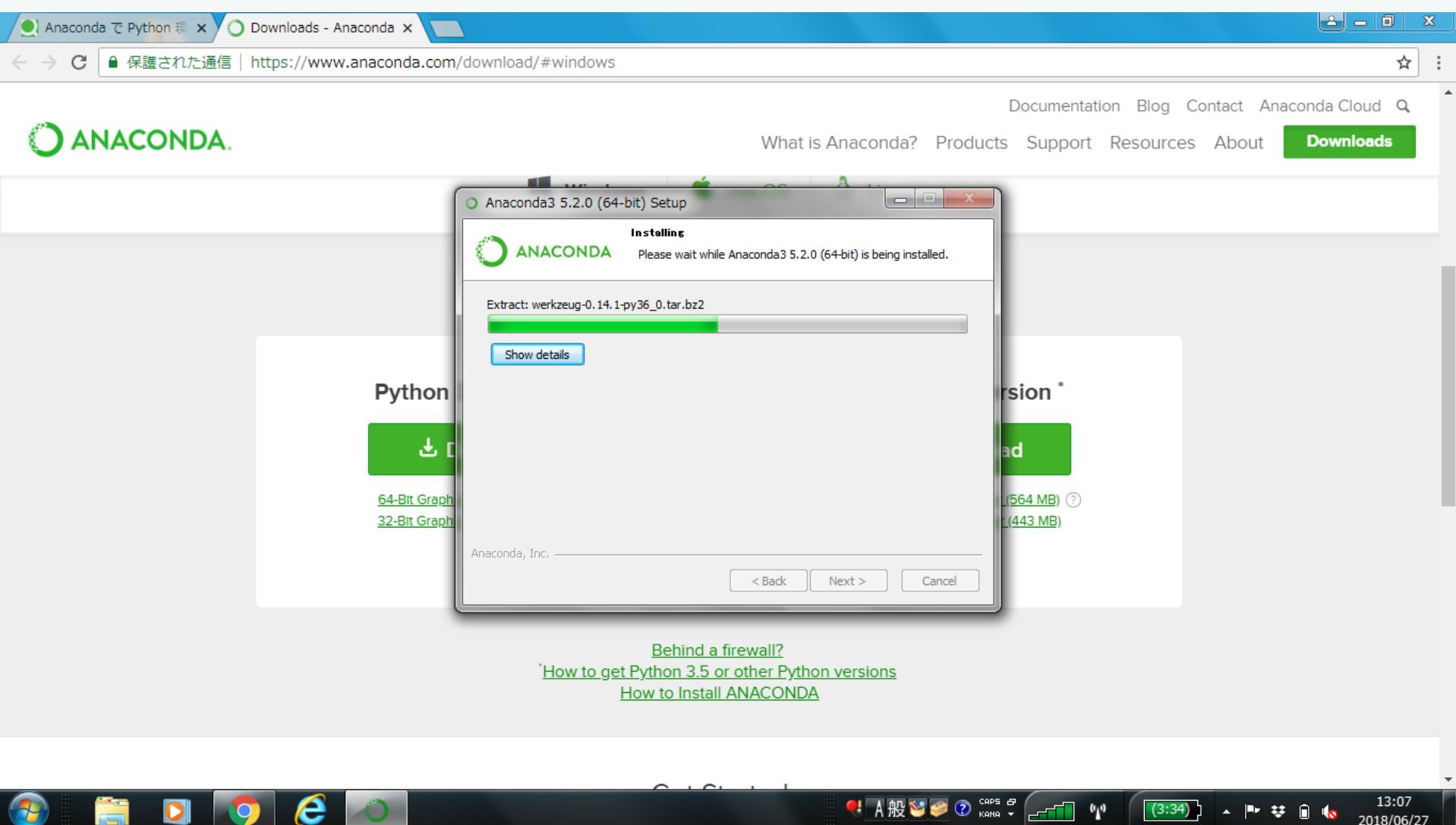


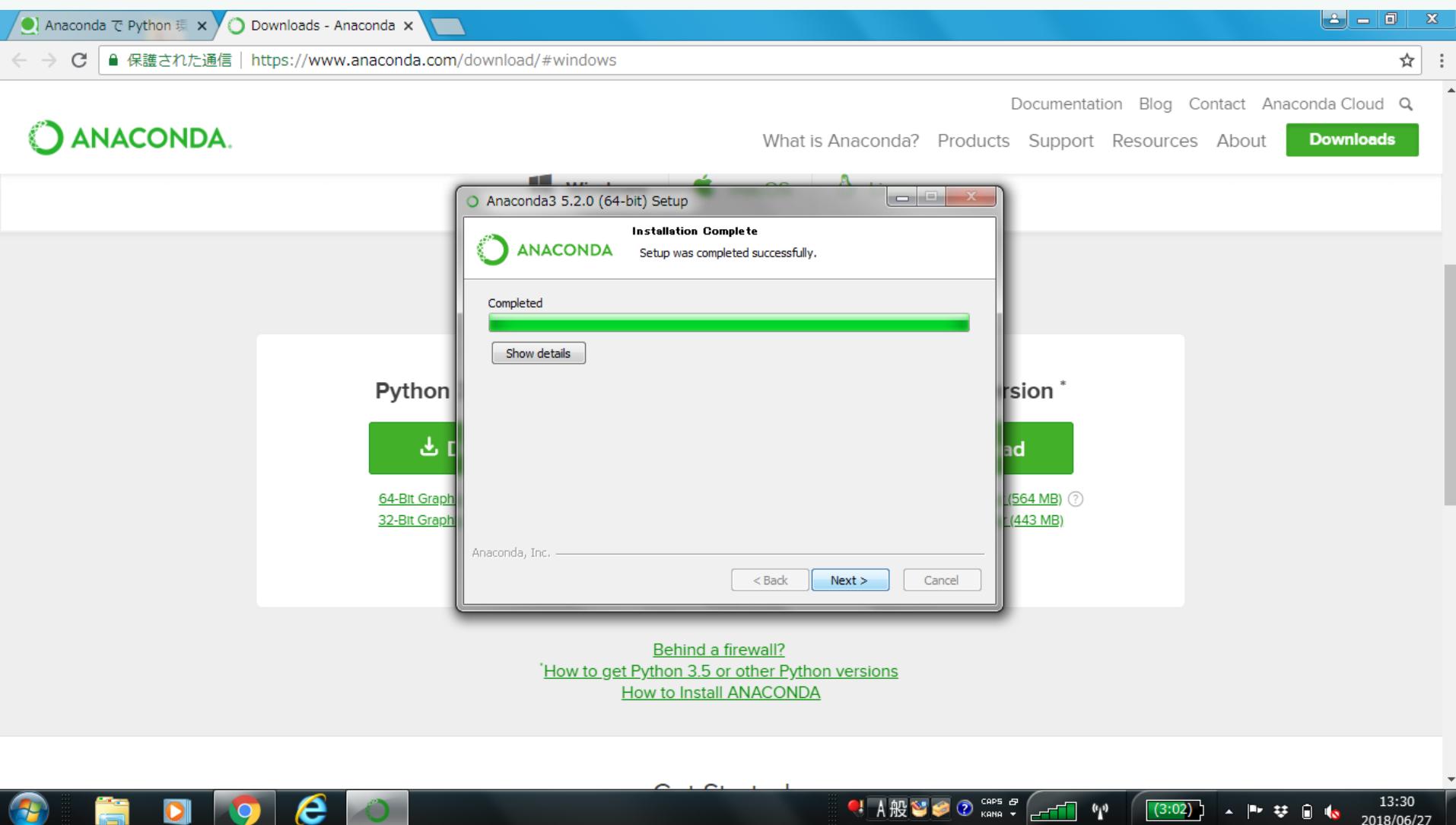


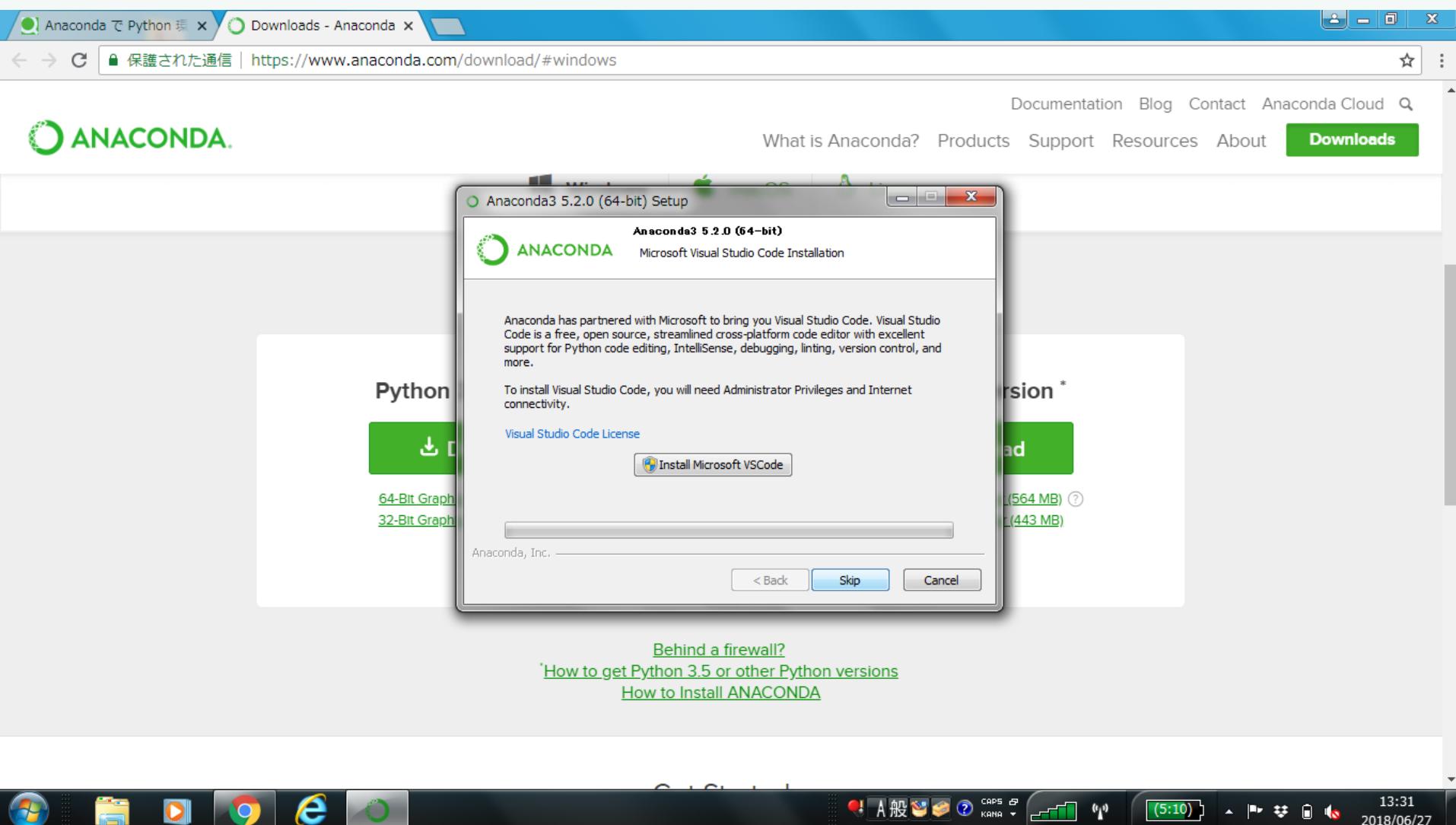




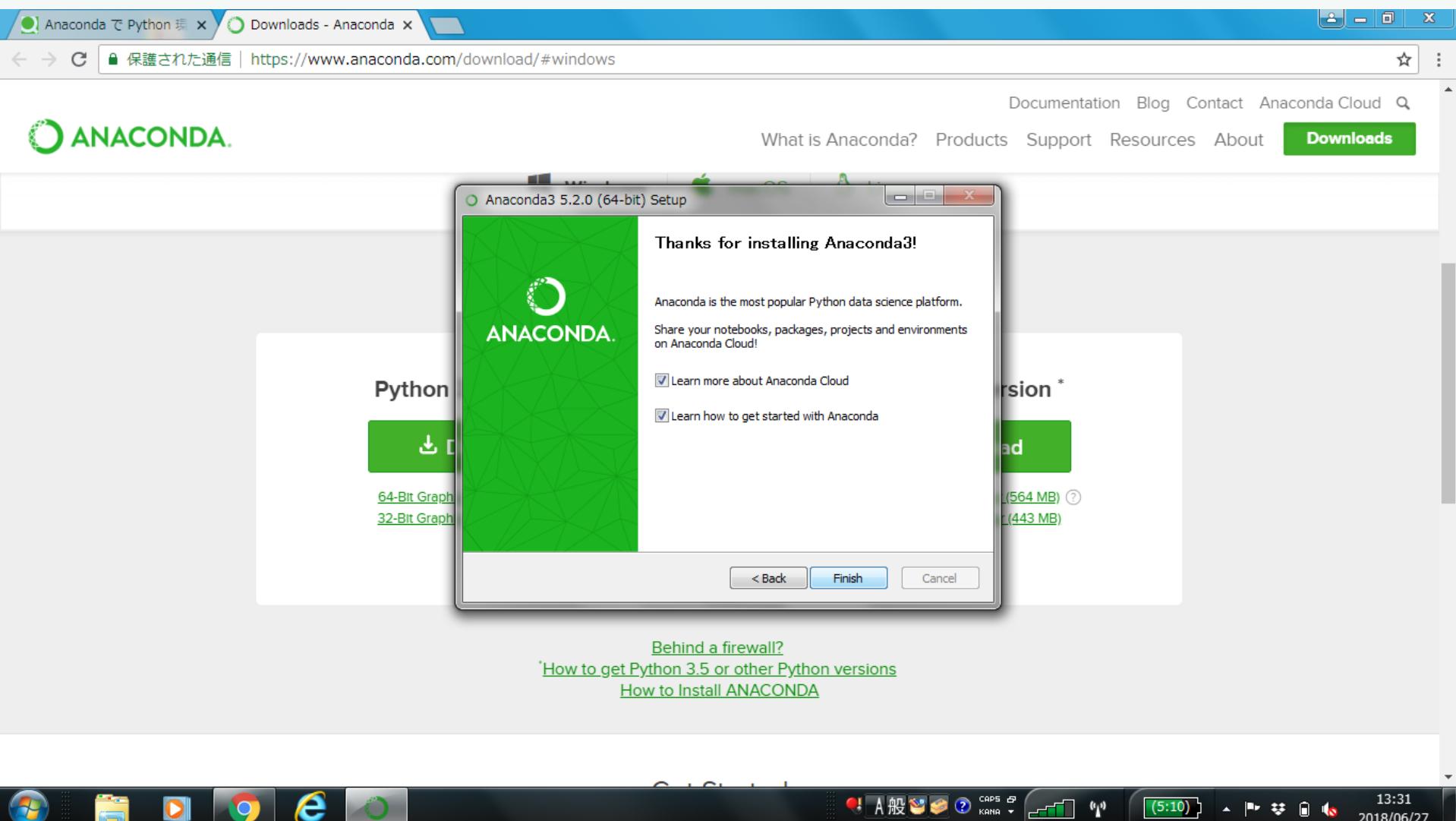
インストール中.







インストール終了(Finishをクリック).



初めて使うための説明

Anaconda で Python × Downloads - Anaconda × :: Anaconda Cloud × Getting started with An ×

保護された通信 | <https://docs.anaconda.com/anaconda/user-guide/getting-started>

ANACONDA DOCUMENTATION

Search

Home
Anaconda Enterprise 5
Anaconda Enterprise 4
Anaconda and Top Packages
Anaconda Distribution
Installation
Packages
User guide
Getting started with Anaconda
Tasks
Cheatsheet
Troubleshooting
Anaconda Navigator
Frequently asked questions
Was this page helpful?

≡ [Anaconda Distribution](#) > [Getting started](#)

Getting started with Anaconda

Anaconda Distribution contains **conda** and **Anaconda Navigator**, as well as Python and hundreds of scientific **packages**. When you installed Anaconda, you installed all these too. You can try both conda and Navigator to see which is right for you to manage your packages and environments. You can even switch between them, and the work you do with one can be viewed in the other.

Now, try this simple programming exercise two ways, with Navigator and a terminal, to help you decide which approach is right for you.

Your first Python program: Hello, Anaconda!

Write and run a Python program using Anaconda Navigator.

1. Open Navigator

Choose the instructions for your operating system.

- Windows
- macOS
- Linux

Windows

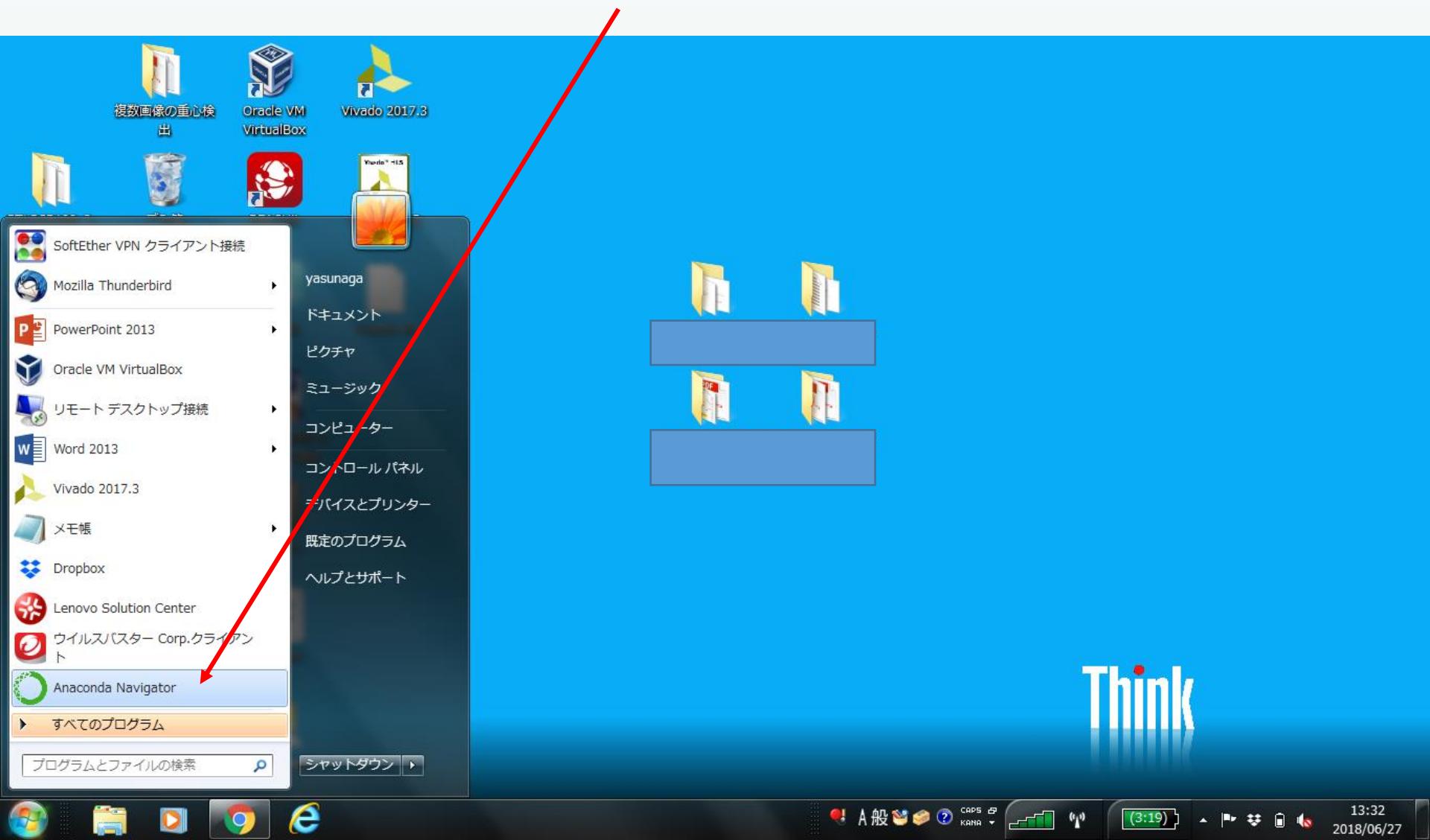
From the Start menu, click the Anaconda Navigator desktop app.

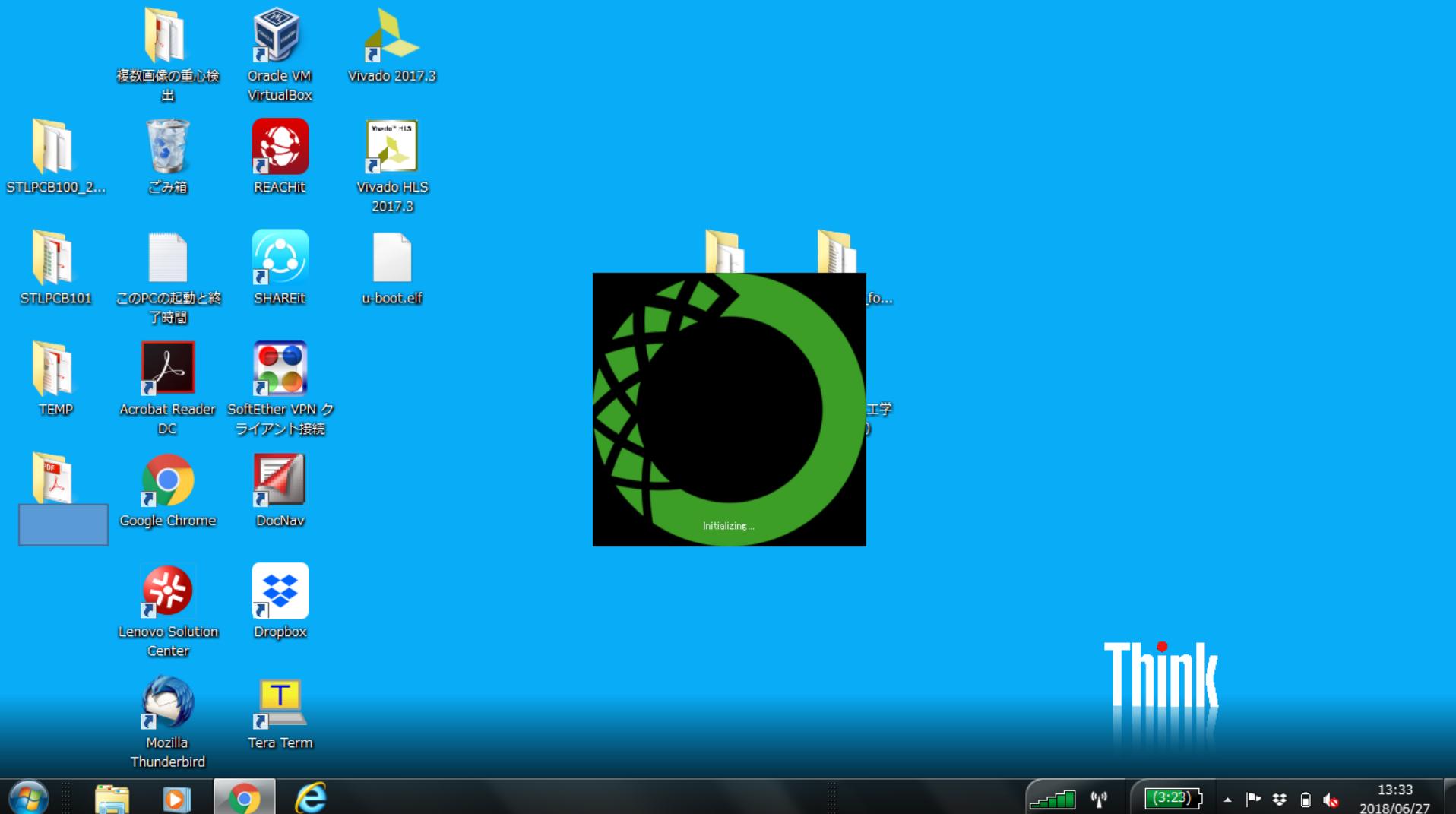


Default Programs
Desktop Gadget Gallery
Internet Explorer

13:32 2018/06/27

Anaconda Navigator から起動





Anaconda (Navigator) の起動です.

Anaconda Navigator

File Help

ANACONDA NAVIGATOR

Sign in to Anaconda Cloud

Refresh

Home Environments Learning Community Documentation Developer Blog Feedback

Twitter YouTube GitHub

Applications on base (root)

lab

jupyterlab 0.32.1

An extensible environment for interactive and reproducible computing, based on Jupyter Notebook and Architecture

Launch

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

glueviz 0.13.3

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

orange3 3.13.0

Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows

rstudio 1.1.423

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

spyder 3.2.8

Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

Launch

vscode 1.24.1

Streamlined code editor with support for development operations like debugging, task running and version control.

13:35 2018/06/27

Anaconda (Navigator)のホーム画面.

Anaconda Navigator

File Help

ANACONDA® NAVIGATOR

Sign in to Anaconda Cloud

Home Environments Learning Community Documentation Developer Blog Feedback

Twitter YouTube GitHub

Applications on base (root) Channels Refresh

 jupyterlab 0.32.1 An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture. Launch	 notebook 5.5.0 Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis. Launch	 qtconsole 4.3.1 PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more. Launch	 spyder 3.2.8 Scientific PYthon Development Environment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features Launch
 glueviz 0.13.3 Multidimensional data visualization across files. Explore relationships within and among related datasets. Launch	 orange3 3.13.0 Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows Launch	 rstudio 1.1.423 A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks. Launch	 vscode 1.24.1 Streamlined code editor with support for development operations like debugging, task running and version control. Launch

13:35 2018/06/27

Environments を見ると、Base という環境ができます。

Installed を見ると、既にインストールされているパッケージが一覧で表示されています。

The screenshot shows the Anaconda Navigator interface. On the left, there's a sidebar with links to Home, Environments (which is selected and highlighted in green), Learning, Community, Documentation, Developer Blog, and Feedback. Below the sidebar are icons for Twitter, YouTube, and GitHub. The main area has two tabs: 'Environments' and 'Installed'. The 'Environments' tab shows a list with one item: 'base (root)'. The 'Installed' tab shows a table of 243 packages. The table has columns for Name, Description, and Version. Some package names are partially visible, such as '_jupyter_lab_nb_ex...', 'alabaster', 'anaconda', 'anaconda-client', 'anaconda-project', 'asn1crypto', 'astroid', 'astropy', 'attrs', 'babel', 'backcall', and 'backports'. The version column shows values like 0.1.0, 0.7.10, 5.2.0, etc.

Name	Description	Version
_jupyter_lab_nb_ex...		0.1.0
alabaster	Configurable, python 2+3 compatible sphinx theme	0.7.10
anaconda		5.2.0
anaconda-client	Anaconda.org command line client library	1.6.14
anaconda-project	Reproducible, executable project directories	0.8.2
asn1crypto	Asn.1 parser and serializer	0.24.0
astroid	Abstract syntax tree for python with inference support	1.6.3
astropy	Community-developed python library for astronomy	3.0.2
attrs	Implement attribute-related object protocols without boilerplate	18.1.0
babel	Utilities to internationalize and localize python applications	2.5.3
backcall		0.1.0
backports		1.0

Not installed を見ると、まだ、インストールされていない
パッケージの一覧が表示されます。

The screenshot shows the Anaconda Navigator interface. On the left is a sidebar with icons for Home, Environments, Learning, Community, Documentation, Developer Blog, and Feedback. Below the sidebar are social media links for Twitter, YouTube, and GitHub. The main area is titled "ANACONDA NAVIGATOR" and shows a list of packages under the "base (root)" environment. A red arrow points from the Japanese text above to the "Not installed" dropdown menu in the top right of the main window. Another red arrow points down to the list of packages. The list includes packages like 7za, nb_ext_conf, r-mutex, windows, absl-py, accelerate, accelerate_cudalib, aenum, affine, agate, agate-dbf, and agate-excel, all marked as not installed. At the bottom of the list, it says "1500 packages available".

Name	Description	Version
7za		920
_nb_ext_conf		0.4.0
_r-mutex		1.0.0
_windows		1.0
absl-py		0.2.2
accelerate		2.3.1
accelerate_cudalib		2.0
aenum		2.1.2
affine	Matrices describing affine transformation of the plane	2.2.0
agate		1.6.1
agate-dbf		0.2.0
agate-excel		0.2.2

新たな開発環境を作ります。
Create をクリックします。

Anaconda Navigator

File Help

Sign in to Anaconda Cloud

Home Environments Learning Community Documentation Developer Blog Feedback

Search Environments base (root)

Installed Channels Update index... Search Packages

Name	Description	Version
_jupyter_nb_ex...		0.1.0
alabaster	Configurable, python 2+3 compatible sphinx theme	0.7.10
anaconda		5.2.0
anaconda-client	Anaconda.org command line client library	1.6.14
anaconda-project	Reproducible, executable project directories	0.8.2
asn1crypto	Asn.1 parser and serializer	0.24.0
astroid	Abstract syntax tree for python with inference support	1.6.3
astropy	Community-developed python library for astronomy	3.0.2
attrs	Implement attribute-related object protocols without boilerplate	18.1.0
babel	Utilities to internationalize and localize python applications	2.5.3
backcall		0.1.0
backports		1.0

243 packages available

Create Clone Import Remove



13:37
2018/06/27

Pythonにチェックします。

The screenshot shows the Anaconda Navigator interface. A red arrow points from the text "Pythonにチェックします。" to the "Create new environment" dialog box. The dialog box has the following fields:

- Name: New environment name
- Location: base (root)
- Packages:
 - Python 3.6
 - R mro

Below the dialog, the main Navigator window displays a list of installed packages:

Name	Description	Version
astropy	Community-developed python library for astronomy	3.0.2
attrs	Implement attribute-related object protocols without boilerplate	18.1.0
babel	Utilities to internationalize and localize python applications	2.5.3
backcall		0.1.0
backports		1.0

At the bottom of the Navigator window, there are buttons for Create, Clone, Import, and Remove, and a message indicating 243 packages available.

Python 3.5 を選びます.

Anaconda Navigator

File Help

ANACONDA NAVIGATOR

Sign in to Anaconda Cloud

Home Environments Learning Community Documentation Developer Blog Feedback

Search Environments Installed Channels Update index... Search Packages

base (root) ▶ Name Description Version

Create new environment

Name: New environment name

Location:

Packages: Python R

3.5
3.6
3.5
2.7

Cancel Create

astropy attrs babel backcall backports

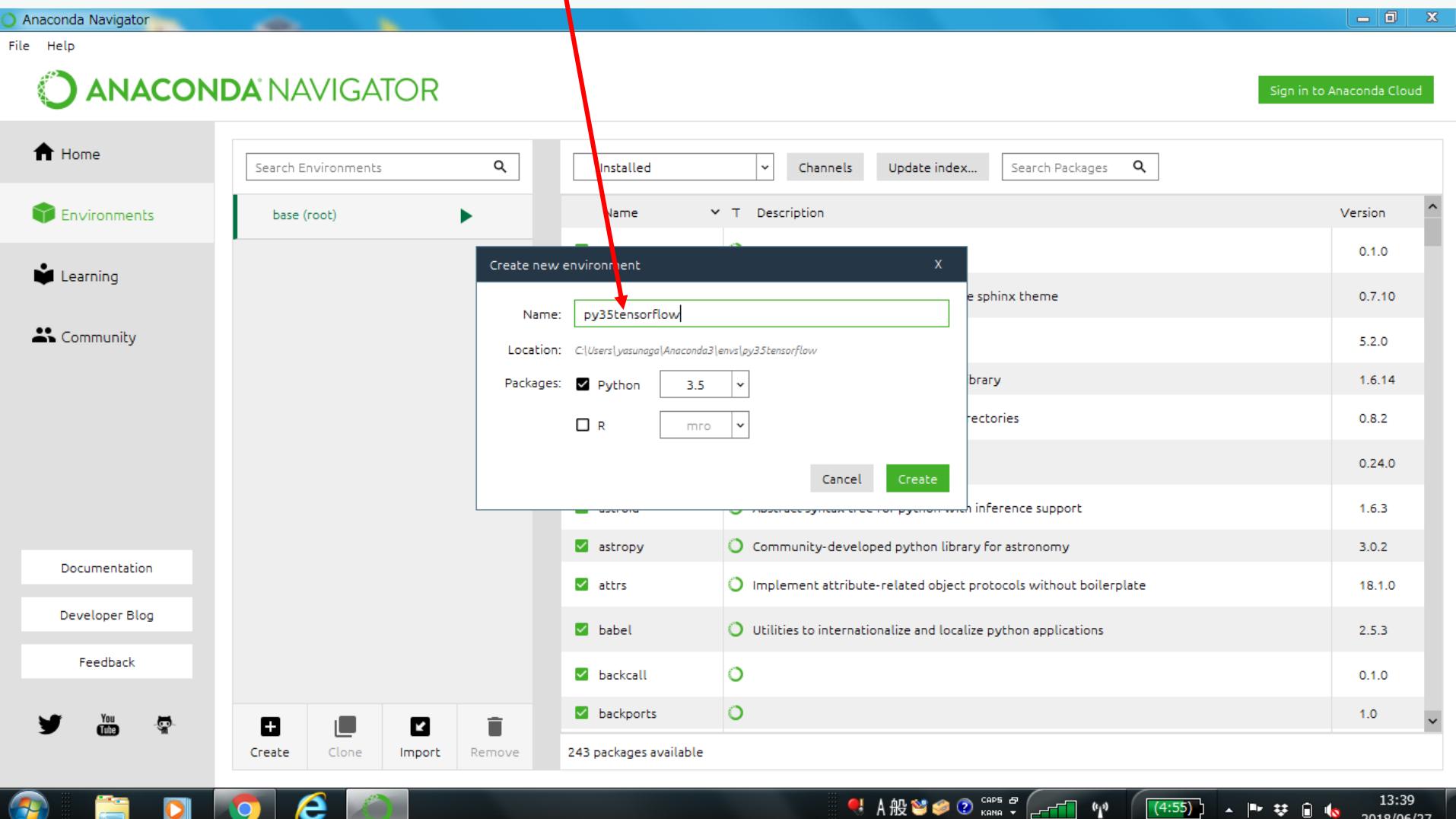
Community-developed python library for astronomy Implement attribute-related object protocols without boilerplate Utilities to internationalize and localize python applications

13:38 2018/06/27

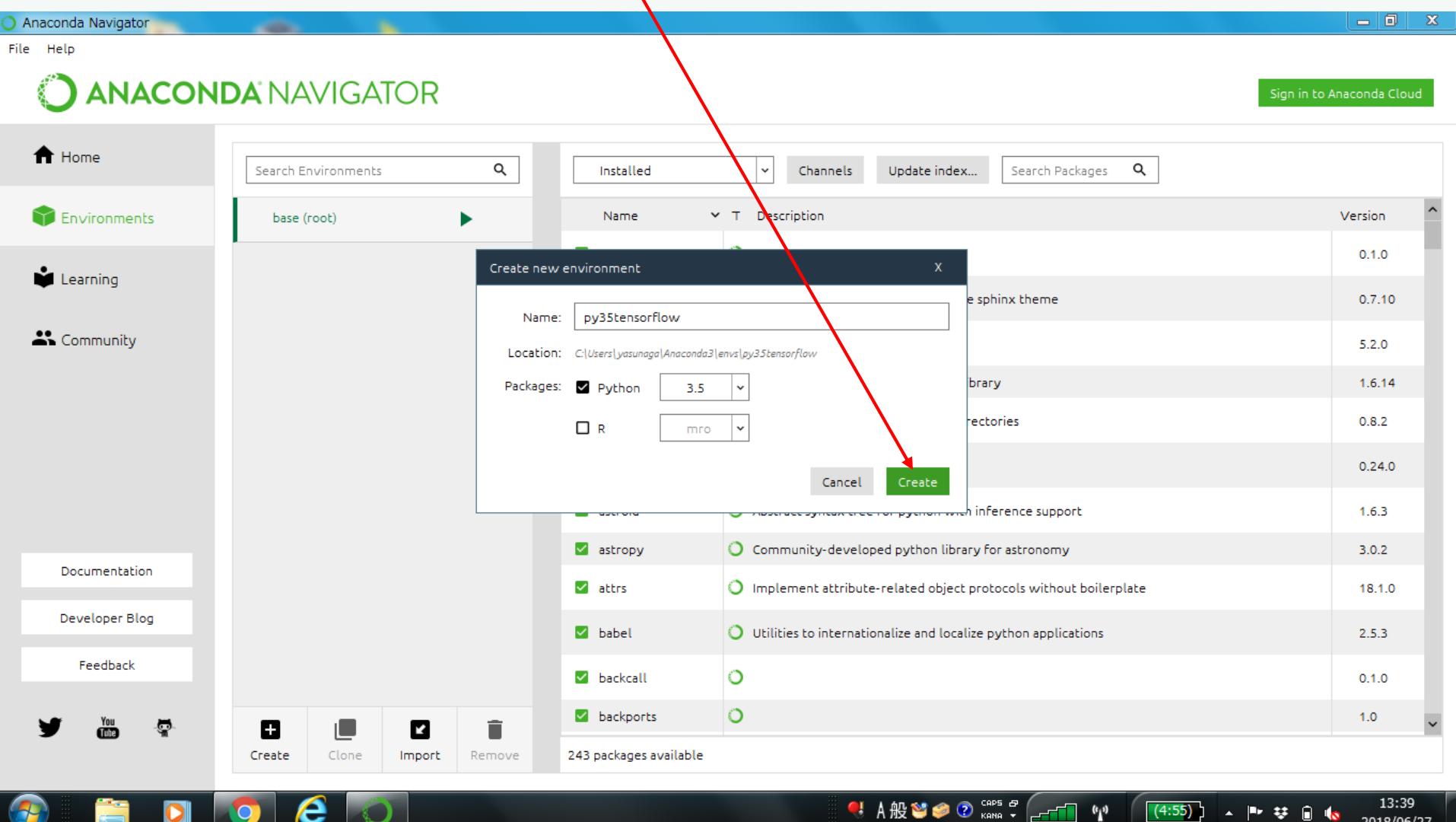
37

The screenshot shows the Anaconda Navigator interface. On the left, there's a sidebar with links for Home, Environments, Learning, Community, Documentation, Developer Blog, and Feedback. Below the sidebar are icons for Twitter, YouTube, and GitHub. The main area has tabs for 'Installed' and 'Channels', and buttons for 'Update index...' and 'Search Packages'. A search bar at the top says 'Search Environments'. In the center, there's a table of installed packages like astropy, attrs, babel, backcall, and backports. At the bottom, there are buttons for 'Create', 'Clone', 'Import', and 'Remove', and a message '243 packages available'. A modal window titled 'Create new environment' is open, prompting for a 'Name' (set to 'New environment name') and a 'Location' (set to 'base (root)'). Under 'Packages', there's a checkbox for 'Python' which is checked, and another for 'R' which is unchecked. A dropdown menu for Python version shows options 3.5, 3.6, and 2.7, with 3.5 selected. A red arrow points from the Japanese text above to the '3.5' option in the dropdown. The status bar at the bottom right shows the time as 13:38 and the date as 2018/06/27.

Name(環境名)に適当な名前を入れます。



Create をクリックします。



新たな環境を作成中です。

Anaconda Navigator

File Help

Sign in to Anaconda Cloud

Home Environments Learning Community Documentation Developer Blog Feedback

Search Environments

Installed Channels Update Index... Search Packages

Name	Description	Version
<input checked="" type="checkbox"/> _ipyw_jlab_nb_ex...		0.1.0
<input checked="" type="checkbox"/> alabaster	Configurable, python 2+3 compatible sphinx theme	0.7.10
<input checked="" type="checkbox"/> anaconda		5.2.0
<input checked="" type="checkbox"/> anaconda-client	Anaconda.org command line client library	1.6.14
<input checked="" type="checkbox"/> anaconda-project	Reproducible, executable project directories	0.8.2
<input checked="" type="checkbox"/> asn1crypto	Asn.1 parser and serializer	0.24.0
<input checked="" type="checkbox"/> astroid	Abstract syntax tree for python with inference support	1.6.3
<input checked="" type="checkbox"/> astropy	Community-developed python library for astronomy	3.0.2
<input checked="" type="checkbox"/> attrs	Implement attribute-related object protocols without boilerplate	18.1.0
<input checked="" type="checkbox"/> babel	Utilities to internationalize and localize python applications	2.5.3
<input checked="" type="checkbox"/> backcall		0.1.0
<input checked="" type="checkbox"/> backports		1.0

Creating environment C:\Users\yasunaga\Anaconda3\envs\py35tensorflow

Create Clone Import Remove

13:40 2018/06/27

新たな環境(py35tensorflow)ができました。

Anaconda Navigator

File Help

ANACONDA NAVIGATOR

Sign in to Anaconda Cloud

Home Environments Learning Community Documentation Developer Blog Feedback

Search Environments

base (root)

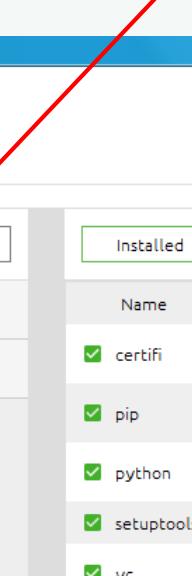
py35tensorflow

Installed Channels Update index... Search Packages

Name	Description	Version
certifi	Python package for providing mozilla's ca bundle.	2018.4.16
pip	Pypa recommended tool for installing python packages	10.0.1
python	General purpose programming language	3.5.5
setuptools	Download, build, install, upgrade, and uninstall python packages	39.2.0
vc	A meta-package to track vc features.	14
vs2015_runtime	Bundles of the msvc 14 (vs 2015) runtime	14.0.25123
wheel	Built-package format for python	0.31.1
wincertstore	Python module to extract ca and crt certs from windows' cert store	0.2

Create Clone Import Remove

8 packages available



13:47 2018/06/27

既にインストールされているパッケージが表示されます。

Anaconda Navigator

File Help

Sign in to Anaconda Cloud

ANACONDA NAVIGATOR

Home Environments Learning Community Documentation Developer Blog Feedback

Search Environments

base (root) py35tensorflow

Installed Not installed Updatable Selected All

setup tools vc vs2015_runtime wheel wincertstore

Description Version

Description	Version
python package for providing mozilla's ca bundle.	2018.4.16
a recommended tool for installing python packages	10.0.1
general purpose programming language	3.5.5
Download, build, install, upgrade, and uninstall python packages	39.2.0
A meta-package to track vc features.	14
Bundles of the msvc 14 (vs 2015) runtime	14.0.25123
Built-package format for python	0.31.1
Python module to extract ca and crt certs from windows' cert store	0.2

Create Clone Import Remove

8 packages available

13:47 2018/06/27

まだ、インストールされていないパッケージが表示されます。

Anaconda Navigator

File Help

Sign in to Anaconda Cloud

ANACONDA NAVIGATOR

Home Environments Learning Community Documentation Developer Blog Feedback

Search Environments

base (root) py35tensorflow

Not installed

Channels Update index... Search Packages

Name	Description	Version
7za		920
_ipyw_jlab_nb_ex...		0.1.0
_nb_ext_conf		0.4.0
_r-mutex		1.0.0
_windows		1.0
absl-py		0.2.2
accelerate		2.3.1
accelerate_cudalib		2.0
aenum		2.1.2
affine	Matrices describing affine transformation of the plane	2.2.1
agate		1.6.1
agate-dbf		0.2.0

Create Clone Import Remove

1776 packages available

13:47 2018/06/27

TensorFlowのパッケージは、まだ入って(インストールされて)いません。
そこで、Not installed で “tenso”まで入力すると自動てきに検索されて
TensorFlow 関係のパッケージの一覧が表示されます。

The screenshot shows the Anaconda Navigator interface. On the left, there's a sidebar with icons for Home, Environments, Learning, and Community, along with links for Documentation, Developer Blog, and Feedback. Below the sidebar are social media sharing icons for Twitter, YouTube, and GitHub. The main area has a search bar at the top with the text "Not installed" and a search term "tenso". A red arrow points from the text "tenso" in the search bar down to the "Name" column of the search results table. The results table lists five packages: r-tensorflow, tensorflow, tensorboard, tensorflow-base, and tensorflow-gpu, all with green circular status indicators. At the bottom of the results table, it says "5 packages available matching 'tenso'".

Name	Description	Version
r-tensorflow		1.4.3
tensorflow		1.8.0
tensorboard		1.8.0
tensorflow-base		1.8.0
tensorflow-gpu		1.1.0

必要な TensorFlow のパッケージにチェックを入れます。

The screenshot shows the Anaconda Navigator interface. On the left, there's a sidebar with icons for Home, Environments, Learning, and Community, along with links for Documentation, Developer Blog, and Feedback. Below the sidebar are social media icons for Twitter, YouTube, and GitHub. The main area has a search bar at the top with the text 'tenso'. A red arrow points from the text '必要な TensorFlow のパッケージにチェックを入れます。' to the 'tensorflow' package in the search results table. The table lists five packages under the heading 'Not installed': r-tensorflow (version 1.4.3), tensorflow (version 1.8.0), tensorflow-base (version 1.8.0), and tensorflow-gpu (version 1.1.0). The 'tensorflow' package is selected, indicated by a green checkmark icon. At the bottom of the main area, it says '5 packages available matching "tenso" 3 packages selected'.

Name	Description	Version
r-tensorflow		1.4.3
<input checked="" type="checkbox"/> tensorflow		1.8.0
<input checked="" type="checkbox"/> tensorflow-base		1.8.0
<input type="checkbox"/> tensorflow-gpu		1.1.0

Apply をクリック.

Anaconda Navigator

File Help

ANACONDA NAVIGATOR

Sign in to Anaconda Cloud

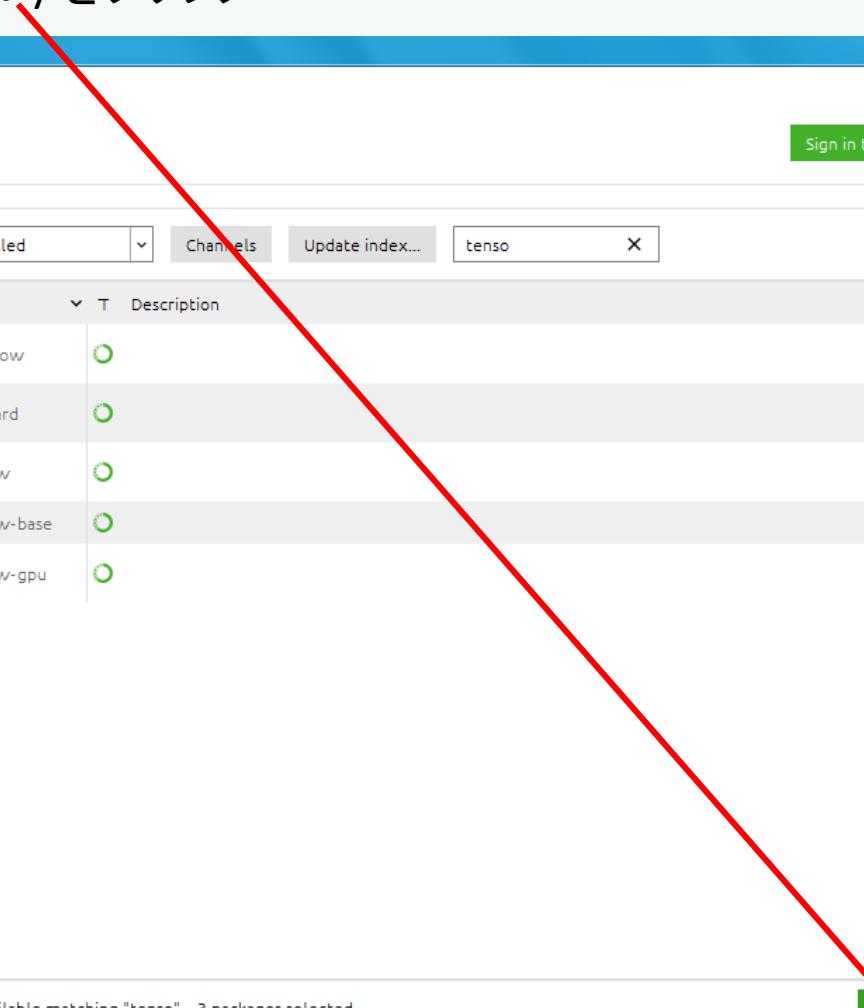
Home Environments Learning Community Documentation Developer Blog Feedback

Search Environments Not installed Channels Update index... tenso

base (root) py35tensorflow

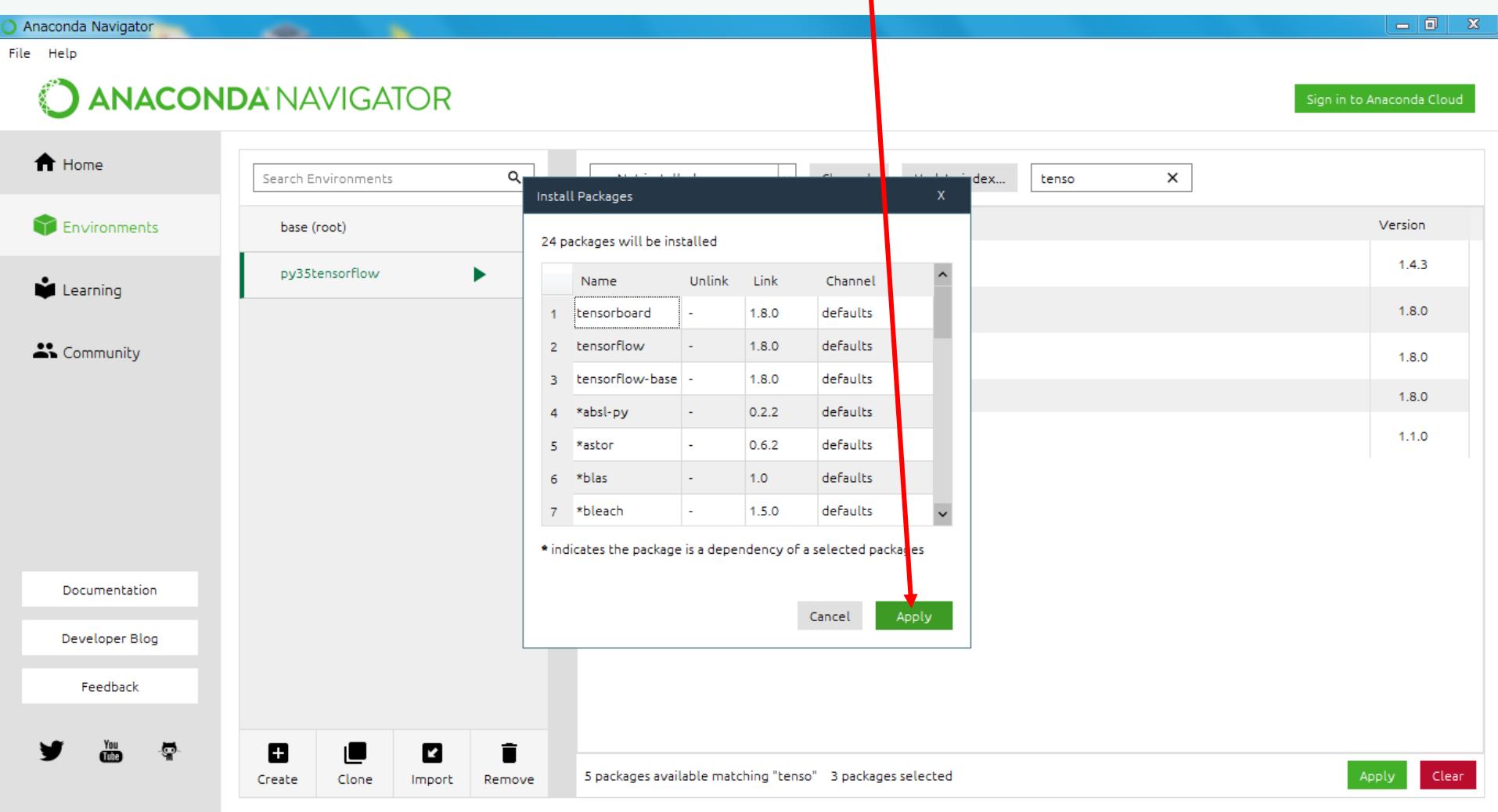
Name	Description	Version
<input type="checkbox"/> r-tensorflow	○	1.4.3
<input checked="" type="checkbox"/> tensorflow	○	1.8.0
<input checked="" type="checkbox"/> tensorflow-base	○	1.8.0
<input type="checkbox"/> tensorflow-gpu	○	1.1.0

5 packages available matching "tenso" 3 packages selected



13:51 2018/06/27

TensorFlow 関係でチェックしたパッケージで必要そうな
パッケージが全て表示されるので、Applyをクリック。



TensorFlow 関係のパッケージが全てインストールされました.

A screenshot of the Anaconda Navigator application interface. The main window displays a list of installed packages in the 'base (root)' environment. A red arrow points from the text above to the 'tensorflow' package in the list. The search bar at the top right shows the query 'tens'. The table below lists three packages:

Name	Description	Version
tensorboard		1.8.0
tensorflow		1.8.0
tensorflow-base		1.8.0

The sidebar on the left includes links for Home, Environments, Learning, and Community, along with Documentation, Developer Blog, and Feedback buttons. The bottom navigation bar contains icons for Create, Clone, Import, Remove, and a status message: '3 packages available matching "tens"'.

その他、必要なパッケージを全てインストールします。
まず、numpy（数値計算用パッケージ）を探して、インストールします。

The screenshot shows the Anaconda Navigator interface. On the left, there's a sidebar with icons for Home, Environments, Learning, Community, Documentation, Developer Blog, and Feedback. Below the sidebar are social media links for Twitter, YouTube, and GitHub. The main area is titled "ANACONDA NAVIGATOR". It has a search bar at the top with the text "Search Environments" and a magnifying glass icon. To the right of the search bar are buttons for "Not installed", "Channels", "Update index...", and a search input field containing "numpy" with a green border and an "X" button. A red arrow points from the text above to this search input field. Below the search bar is a table with columns for Name, Description, and Version. The table lists 10 packages available matching "numpy":

Name	Description	Version
blaze	Numpy and pandas interface to big data	0.9.1
bottlechest	Fast numpy array functions specialized for use in orange	0.7.1
bottleneck	Fast numpy array functions written in cython.	1.2.1
msgpack-numpy		0.4.3
netcdf4	Python/numpy interface to netcdf library	1.4.0
numba	Numpy aware dynamic python compiler using llvm	0.9.0
numexpr	Fast numerical expression evaluator for numpy	2.6.5
numpy-devel		1.14.5
numpydoc	Sphinx extension to support docstrings in numpy format	0.8.0
snuggs	S-expressions for numpy	1.4.2

At the bottom of the main area, it says "10 packages available matching 'numpy'".

The taskbar at the bottom of the screen shows several icons: Start button, File Explorer, File Manager, Google Chrome, Internet Explorer, and the Anaconda Navigator icon. On the far right of the taskbar, there are system status icons, a battery level indicator, a volume icon, and a date/time stamp "14:05 2018/06/27".

インストールするパッケージにチェックを入れます。

Anaconda Navigator

File Help

Sign in to Anaconda Cloud

Home Environments Learning Community Documentation Developer Blog Feedback

Search Environments

base (root) py35tensorflow

Not installed Channels Update index... numpy X

Name	Description	Version
blaze	Numpy and pandas interface to big data	0.9.1
bottlechest	Fast numpy array functions specialized for use in orange	0.7.1
bottleneck	Fast numpy array functions written in cython.	1.2.1
msgpack-numpy		0.4.3
netcdF4	Python/numpy interface to netcdf library	1.4.0
numba	Numpy aware dynamic python compiler using llvm	0.9.0
numexpr	Fast numerical expression evaluator for numpy	2.6.5
numpy-devel		1.14.5
numpydoc	Sphinx extension to support docstrings in numpy format	0.8.0
snuggs	S-expressions for numpy	1.4.2

10 packages available matching "numpy" 10 packages selected

Create Clone Import Remove Apply Clear



14:05
2018/06/27

ANACONDA NAVIGATOR

[Sign in to Anaconda Cloud](#)[Home](#)[Environments](#)[Learning](#)[Community](#)[Documentation](#)[Developer Blog](#)[Feedback](#)[Create](#)[Clone](#)[Import](#)[Remove](#)

Search Environments



base (root)

py35tensorflow ➤

Install Packages

84 packages will be installed

Name	Unlink	Link	Channel
1 blaze	-	0.11.3	defaults
2 bottlechest	-	0.7.1	defaults
3 bottleneck	-	1.2.1	defaults
4 msgpack-numpy	-	0.4.3	defaults
5 netcdf4	-	1.4.0	defaults
6 numba	-	0.38.1	defaults

* indicates the package is a dependency of a selected packages

[Cancel](#)[Apply](#)

numpy X

	Version
use in orange	0.9.1
hon.	0.7.1
hon.	1.2.1
using llvm	0.4.3
numpy	1.4.0
numpy format	0.9.0
numpy	2.6.5
	1.14.5
numpy format	0.8.0
	1.4.2

snuggs

S-expressions for numpy

10 packages available matching "numpy" 10 packages selected

[Apply](#)[Clear](#)14:06
2018/06/27

Anaconda Navigator

File Help

ANACONDA NAVIGATOR

Sign in to Anaconda Cloud

Home Environments Learning Community Documentation Developer Blog Feedback

Search Environments

Installed Channels Update index...

Name	Description	Version
blaze	Numpy and pandas interface to big data	0.11.3
bottlechest	Fast numpy array functions specialized for use in orange	0.7.1
bottleneck	Fast numpy array functions written in cython.	1.2.1
msgpack-numpy		0.4.3
netcdf4	Python/numpy interface to netcdf library	1.4.0
numba	Numpy aware dynamic python compiler using llvm	0.38.1
numexpr	Fast numerical expression evaluator for numpy	2.6.5
numpy	Array processing for numbers, strings, records, and objects	1.14.5
numpy-base		1.14.5
numpy-devel		1.14.5
numpydoc	Sphinx extension to support docstrings in numpy format	0.8.0
snuggs	S-expressions for numpy	1.4.2

12 packages available matching "numpy"

Create Clone Import Remove

14:20 2018/06/27

matplotlib (グラフ等描画用パッケージ)も、インストールします。

The screenshot shows the Anaconda Navigator interface. On the left, there's a sidebar with icons for Home, Environments, Learning, and Community, along with links for Documentation, Developer Blog, and Feedback. Below the sidebar are social media icons for Twitter, YouTube, and GitHub. The main area is titled "ANACONDA NAVIGATOR". It has a search bar at the top with the text "Search Environments" and a magnifying glass icon. To the right of the search bar are dropdown menus for "Not installed", "Channels", and "Update index...", and a search input field containing "mpl" with a clear button. A red arrow points from the text "matplotlib (グラフ等描画用パッケージ)も、インストールします。" to this search input field. Below the search bar is a table with columns for Name, Description, and Version. The table shows three packages: basemap (1.0.7), matplotlib (2.2.2), and matplotlib-tests (1.5.1). The "matplotlib" row is highlighted with a green background. A red arrow also points to the "matplotlib" row. At the bottom of the main area, it says "3 packages available matching \"mpl\" 1 package selected". There are "Apply" and "Clear" buttons on the right. The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray.

Name	Description	Version
basemap	Plot data on map projections with matplotlib	1.0.7
matplotlib	Python 2d plotting library	2.2.2
matplotlib-tests		1.5.1

ANACONDA NAVIGATOR

[Sign in to Anaconda Cloud](#)[Home](#)[Environments](#)[Learning](#)[Community](#)[Documentation](#)[Developer Blog](#)[Feedback](#)[Create](#)[Clone](#)[Import](#)[Remove](#)

3 packages available matching "matpl" 1 package selected

[Apply](#)[Clear](#)Search Environments 

base (root)

py35tensorflow ➤

Install Packages

10 packages will be installed

	Name	Unlink	Link	Channel
1	matplotlib	-	2.2.2	defaults
2	*cycler	-	0.10.0	defaults
3	*freetype	-	2.8	defaults
4	*icu	-	58.2	defaults
5	*kiwisolver	-	1.0.1	defaults
6	*libpng	-	1.6.34	defaults
7	*pyqt	-	5.9.2	defaults

* indicates the package is a dependency of a selected package

[Cancel](#)[Apply](#)14:25
2018/06/27

pandas (データ解析パッケージ)も、インストールします。

Anaconda Navigator

File Help

Sign in to Anaconda Cloud

Home Environments Learning Community Documentation Developer Blog Feedback

Search Environments Not installed Channels Update index... pandas

Name	Description	Version
geopandas		0.3.0
pandas-datareader	Data readers extracted from the pandas codebase	0.6.0
pandas-profiling		1.4.1
pandasql	Sqldf for pandas	0.7.3
qgrid	Pandas dataframe viewer for ipython notebook	1.0.5

5 packages available matching "pandas" 5 packages selected

Create Clone Import Remove Apply Clear

The screenshot shows the Anaconda Navigator interface. On the left, there's a sidebar with links for Home, Environments, Learning, and Community, along with buttons for Documentation, Developer Blog, and Feedback. Below the sidebar are social media icons for Twitter, YouTube, and GitHub. The main area has a search bar at the top with the text 'Not installed' and a search term 'pandas'. To the right of the search bar are buttons for 'Channels' and 'Update index...'. Below the search bar is a table of package results. The table has columns for Name, Description, and Version. The packages listed are geopandas, pandas-datareader, pandas-profiling, pandasql, and qgrid. All packages are marked as 'Not installed'. The 'pandas-datareader' package is described as 'Data readers extracted from the pandas codebase'. The 'pandasql' package is described as 'Sqldf for pandas'. The 'qgrid' package is described as 'Pandas dataframe viewer for ipython notebook'. The versions listed are 0.3.0, 0.6.0, 1.4.1, 0.7.3, and 1.0.5 respectively. At the bottom of the main area, it says '5 packages available matching "pandas" 5 packages selected'. There are 'Apply' and 'Clear' buttons at the bottom right. Red arrows from the Japanese text above point to the search bar and the results table.

ANACONDA NAVIGATOR

[Sign in to Anaconda Cloud](#)[Home](#)[Environments](#)[Learning](#)[Community](#)[Documentation](#)[Developer Blog](#)[Feedback](#)[Create](#)[Clone](#)[Import](#)[Remove](#)

5 packages available matching "pandas" 5 packages selected

[Apply](#)[Clear](#)

base (root)

py35tensorflow

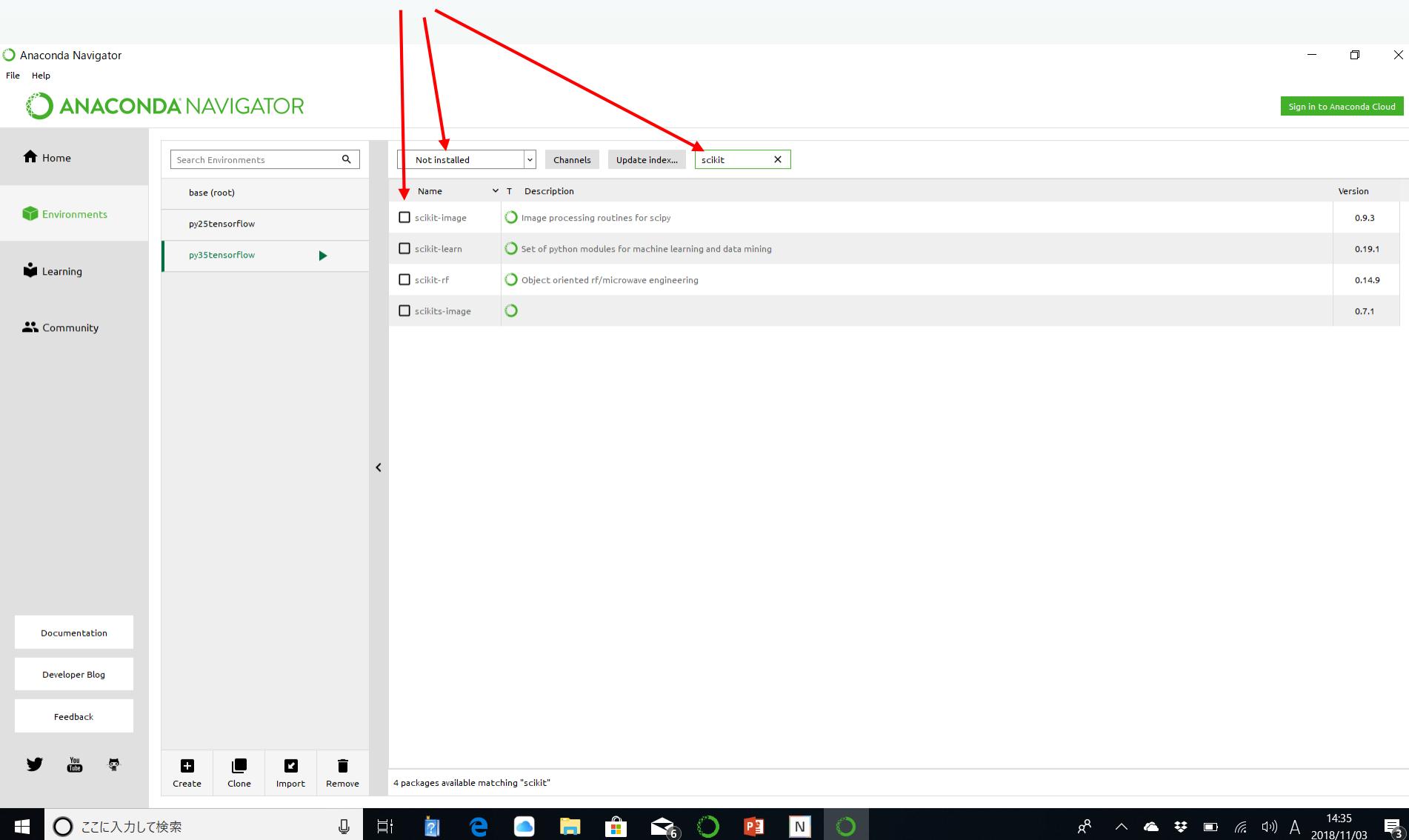
Install Packages
83 packages will be installed

Name	Unlink	Link	Channel	Version
1 geopandas	-	0.3.0	defaults	0.3.0
2 pandas-datareader	-	0.6.0	defaults	0.6.0
3 pandas-profiling	-	1.4.1	defaults	1.4.1
4 pandasql	-	0.7.3	defaults	0.7.3
5 qgrid	-	1.0.5	defaults	1.0.5

* indicates the package is a dependency of a selected packages

[Cancel](#)[Apply](#)14:38
2018/06/27

scikit-learn (機械学習パッケージ)も、インストールします。



Anaconda Navigator

File Help

ANACONDA NAVIGATOR

Sign in to Anaconda Cloud

Home Environments Learning Community Documentation Developer Blog Feedback

Search Environments Not installed Channels Update index... scikit

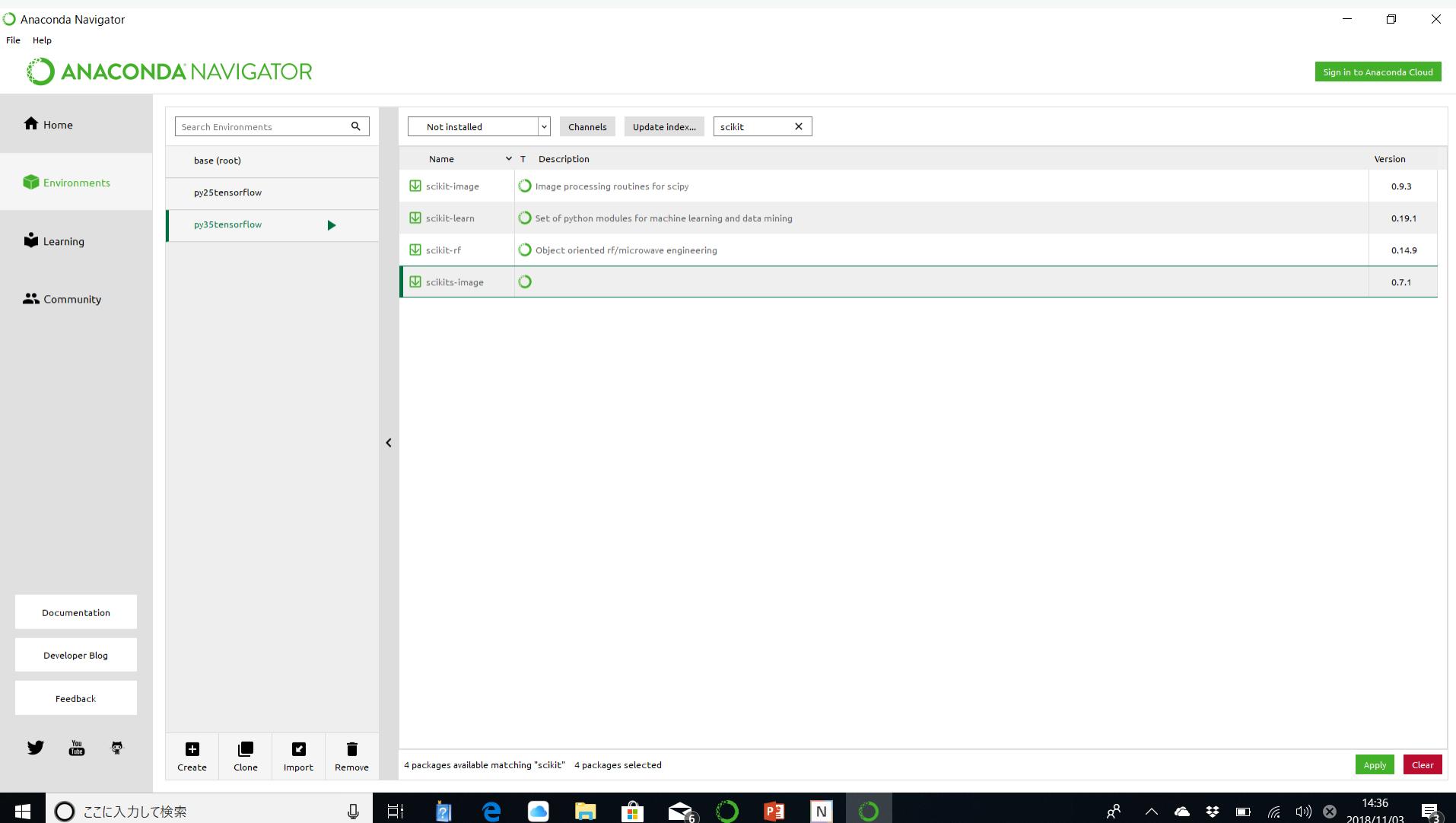
Name	Description	Version
scikit-image	Image processing routines for scipy	0.9.3
scikit-learn	Set of python modules for machine learning and data mining	0.19.1
scikit-rf	Object oriented rf/microwave engineering	0.14.9
scikits-image		0.7.1

Create Clone Import Remove

4 packages available matching "scikit" 4 packages selected Apply Clear

ここに入力して検索

14:36 2018/11/03



Anaconda Navigator

File Help

ANACONDA NAVIGATOR

Sign in to Anaconda Cloud

Home Environments Learning Community Documentation Developer Blog Feedback

Create Clone Import Remove

Search Environments Not installed Channels Update index... scikit

Name	Description	Version
scikit-image	Image processing routines for scipy	0.9.3
scikit-learn	Set of python modules for machine learning and data mining	0.19.1
scikit-rf	Object oriented RF simulation	0.14.9
scikits-image	Image processing routines for scikits	0.7.1

Install Packages

90 packages will be installed

Name	Unlink	Link	Channel
1 scikit-image	-	0.9.3	defaults
2 scikit-learn	-	0.19.1	defaults
3 scikit-rf	-	0.14.9	defaults
4 scikits-image	-	0.7.1	defaults
5 *backports	1.0	1.0	defaults

* indicates the package is a dependency of a selected packages

Cancel Apply

4 packages available matching "scikit" 4 packages selected

Apply Clear

The screenshot shows the Anaconda Navigator interface. On the left is a sidebar with links for Home, Environments, Learning, Community, Documentation, Developer Blog, and Feedback. Below these are social media icons for Twitter, YouTube, and GitHub. At the bottom of the sidebar are buttons for Create, Clone, Import, and Remove. The main area has a search bar at the top with 'Search Environments' and a dropdown set to 'Not installed'. A search term 'scikit' is entered in the search bar. Below the search bar is a table of packages. The table has columns for Name, Description, and Version. It lists four packages: scikit-image, scikit-learn, scikit-rf, and scikits-image. An 'Install Packages' dialog box is open over the main area, showing a list of 90 packages to be installed. The dialog includes a table with columns for Name, Unlink, Link, and Channel, listing the same four packages along with backports. It also includes a note about dependencies and buttons for Cancel and Apply.

pickle (python のオブジェクト保存用パッケージ) も、ついでにインストールします。

The screenshot shows the Anaconda Navigator interface. On the left, there's a sidebar with icons for Home, Environments, Learning, and Community, along with links for Documentation, Developer Blog, and Feedback. Below the sidebar are social media icons for Twitter, YouTube, and GitHub. The main area is titled "ANAconda NAVIGATOR". At the top, there's a search bar with "Search Environments" and a magnifying glass icon, followed by dropdown menus for "Not installed", "Channels", and "Update index...". A search input field contains "pickl" with a clear button. Below the search bar is a table with columns for Name, Description, and Version. One row is highlighted with a green border, showing "picklable-itertools" as the name, "Itertools, but pickleable" as the description, and "0.1.1" as the version. Red arrows point from the Japanese text above to the "Not installed" dropdown and the search input field.

Name	Description	Version
picklable-itertools	Itertools, but pickleable	0.1.1

1 package available matching "pickl" 1 package selected

Anaconda Navigator

File Help

ANACONDA NAVIGATOR

Sign in to Anaconda Cloud

Home Environments Learning Community Documentation Developer Blog Feedback

Search Environments

base (root)

py35tensorflow ➤

Install Packages

1 package will be installed

Name	Unlink	Link	Channel
picklable-itertools	-	0.1.1	defaults

Cancel Apply

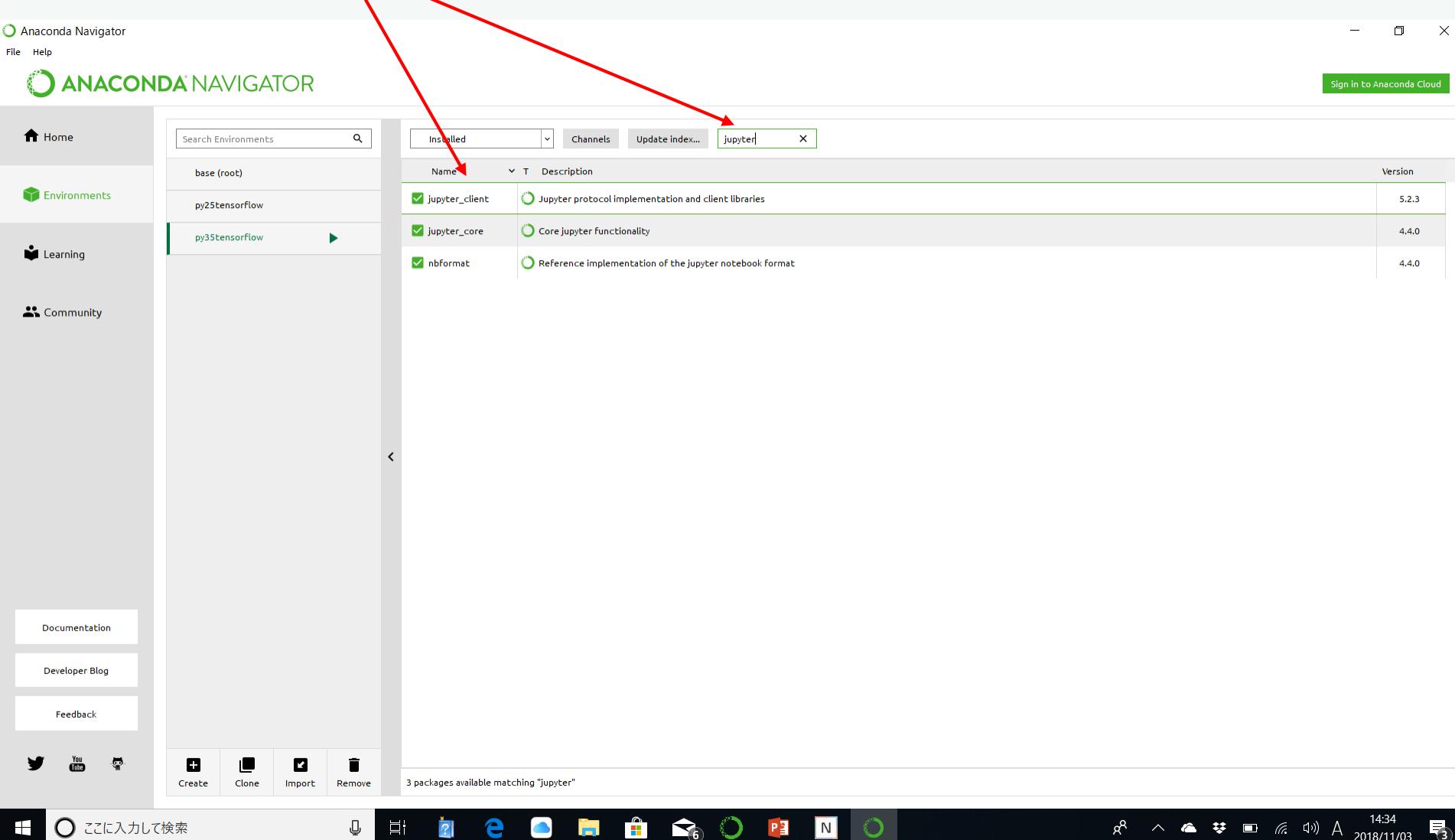
1 package available matching "pickl" 1 package selected

Create Clone Import Remove

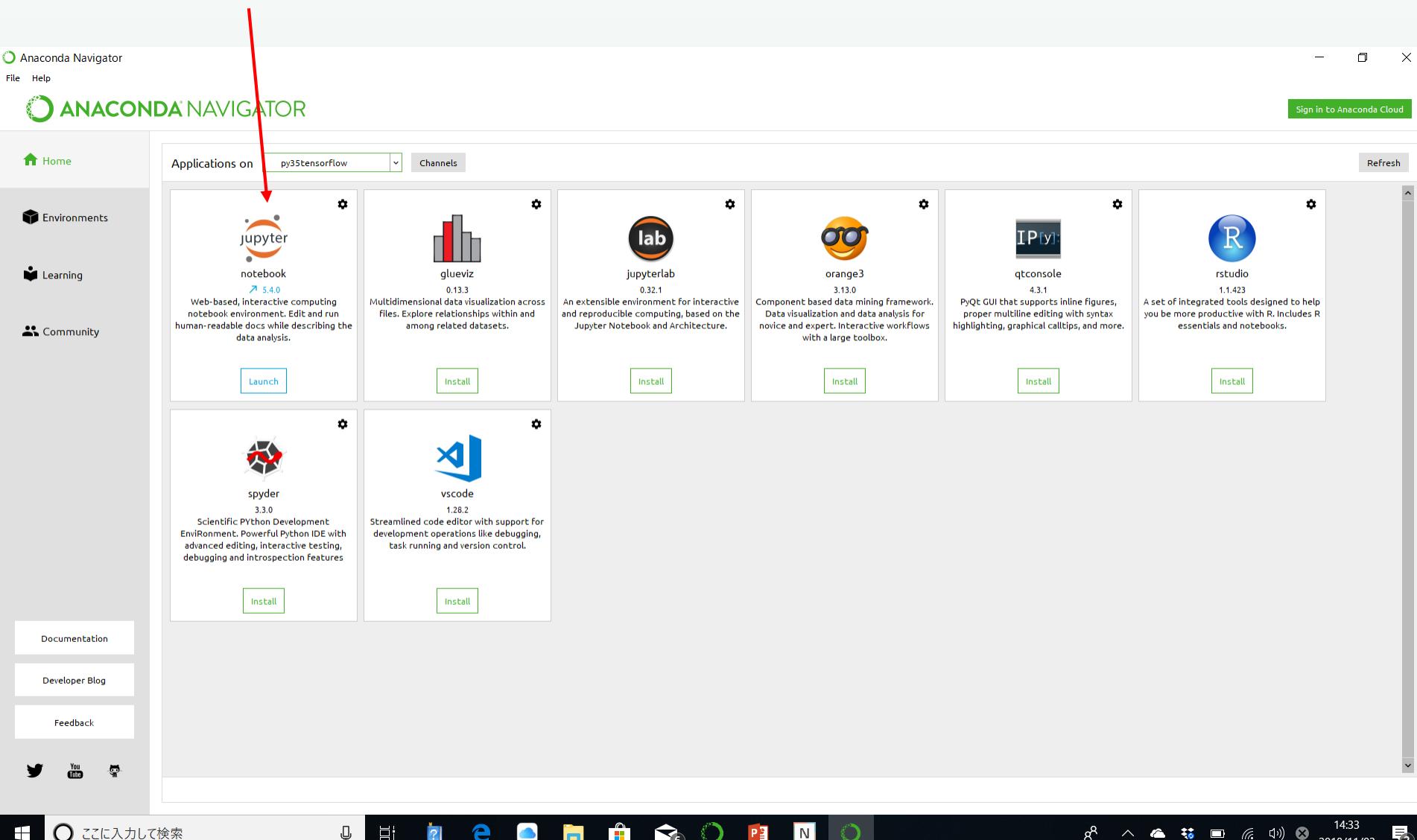
Apply Clear

15:02 2018/06/27

最後に、jupyter (python 用ノートブックのパッケージ) もインストールします。



jupyter は、ホームメニューをクリックすることでもインストールできます。



```

generate_data.py -- C:\Users\yusunaga\Desktop\Python-and-TF -- Atom
ファイル(F) 権限(E) 表示(V) 選択(S) 検索(I) フォルダ(P) ヘルプ(H)
generate_data.py
Welcome

1 validate = int(loop_range * 0.1)

2
3 print('df_stock[終値].shape[0] = ', df_stock['終値'].shape[0])

4 files_name = []
5 labels = []
6 for i in range(loop_range):
7     start = i * divide_value
8     end = (i + 1) * divide_value
9     write_file_name = dir_name + 'data.' + str(i) + '.csv'
10    if ((end+1) < df_stock['終値'].shape[0]):
11        df_stock[[start:end]].to_csv(write_file_name, header=None, index=None)
12        files_name.append(write_file_name)
13        labels.append(int(df_stock['終値'].iloc[[end+1]]) / 1000)
14
15 print(len(labels))
16 print(files_name)
17
18 f_train = open('stock_train.csv', 'w')
19 writer_train = csv.writer(f_train, lineterminator='\n')
20
21 f_validation = open('stock_validation.csv', 'w')
22 writer_validation = csv.writer(f_validation, lineterminator='\n')
23
24 header = ['xdata', 'ylabel']
25 writer_train.writerow(header)
26 writer_validation.writerow(header)
27
28 for i, (file, label) in enumerate(zip(files_name, labels)):
29     if (i < len(files_name) - validate):
30         writer_train.writerow((file, label))
31     if (i >= (len(files_name) - validate)):
32         writer_validation.writerow((file, label))
33
34 f_train.close()
35 f_validation.close()
36
37 print("Complete !")
38
39 def generate_evaluation_data(evaluation_file, divide_value):
40     dir_name = './evaluation_data/'
41
42     if not os.path.exists(dir_name):
43
44

```

エディタによる
プログラムコーディング

```

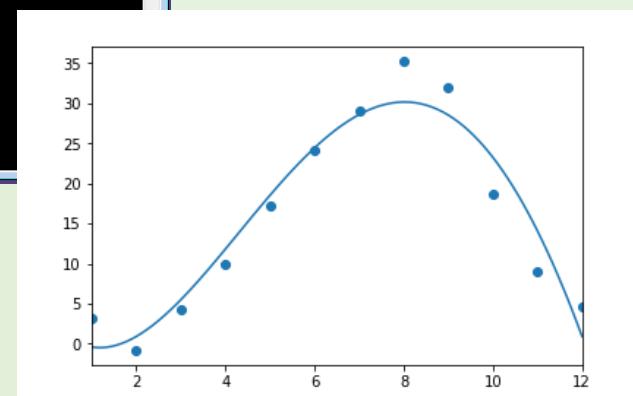
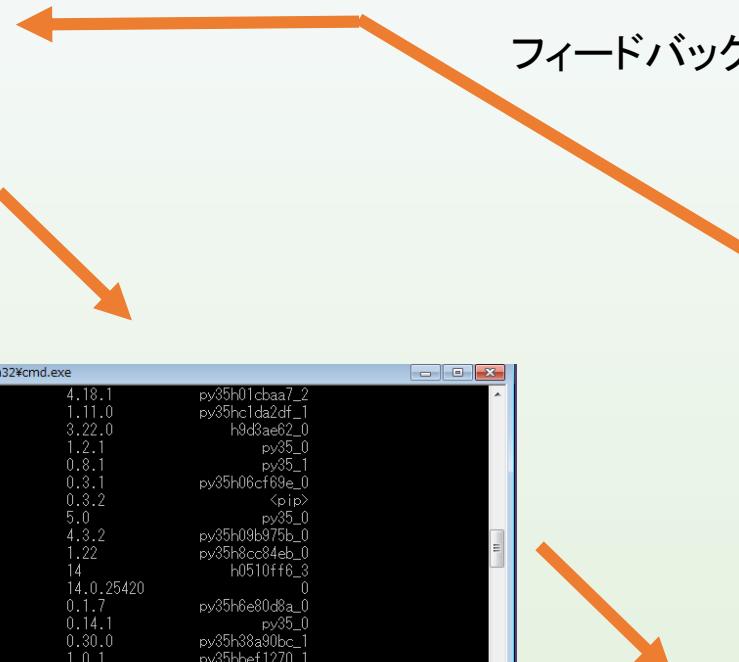
C:\Windows\system32*cmd.exe
sip           4.18.1      py35h01cbaa7_2
six            1.11.0      py35hc1da2df_1
sqlite          3.22.0      h9d3ae62_0
tensorflow       1.2.1       px35_0
terminado        0.8.1       px35_1
testpath         0.3.1      py35h06cf69e_0
tflearn          0.3.2      <pip>
tornado          5.0        px35_0
traitlets        4.3.2      py35hb9b75b_0
urllib3          1.22       py35hbcc84eb_0
vc              14          h0510ff6_3
vs2015_runtime   14.0.25420  0
width           0.1.7      py35h6e80d8a_0
werkzeug         0.14.1      px35_0
wheel            0.30.0      py35h38a90bc_1
win_inet_pton    1.0.1      py35hbef1270_1
win_unicode_console 0.5      py35h56988b5_0
wincertstore    0.2        py35hfbebdb8_0
winpty           0.4.3      py35h4f9c72_4
zlib             1.2.11     h8395fce_2

(py35tensorflow) C:\VSPB_Data>python test.py

```

プログラムの実行

通常のプログラム開発
⇒煩雑で手間がかかる。



結果表示・評価

jupyter notebook について【基本的な使い方】

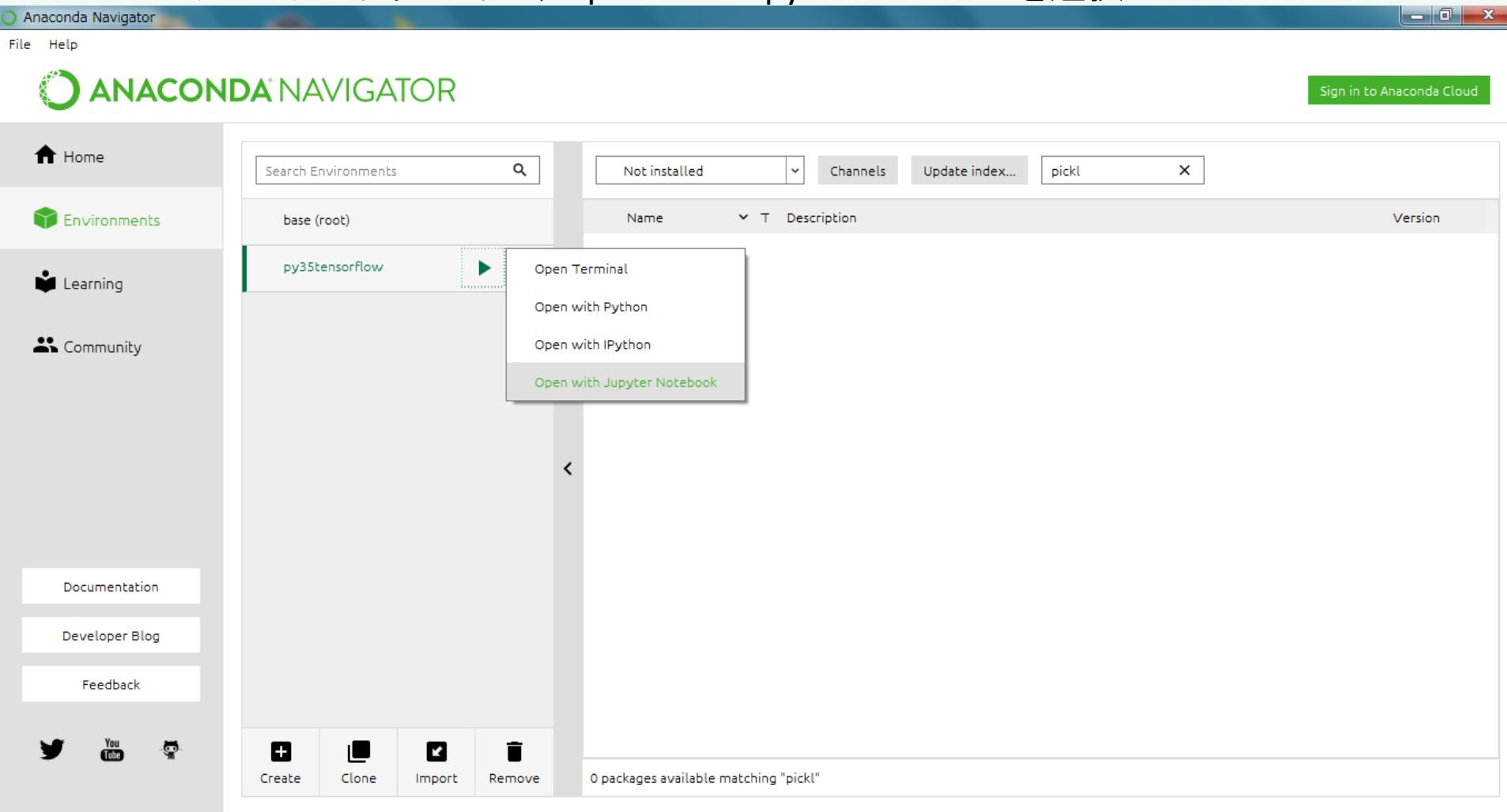
“Webベースの実行可能な文書フォーマット”

notebook : プログラムや説明書の作成と作成したプログラムを実行し、結果を記録できるデータサイエンス用のツール. (データサイエンスとの間で広く使われている.) そのほか, MATLAB, Mathematica, etc.



jupyter notebook を使ってみましょう

Jupyter Notebookを使うには, anaconda の Environments で既にある環境の
ポップアップウィンドウで, Open with Jupyter Notebook を選択.



Jupyter Notebookが立ち上りました(ブラウザ上).

A screenshot of a Windows desktop environment. At the top, there is a taskbar with several open browser tabs. From left to right, the tabs are: 'Anaconda で Python' (with a green icon), 'Downloads - Anaconda' (with a green icon), ':: Anaconda Cloud' (with a green icon), 'Getting started with AI' (with a green icon), 'Home' (with an orange icon), and 'TensorFlow の回帰曲線' (with an orange icon). The title bar of the active window shows the URL 'localhost:8888/tree'. The main content area displays the Jupyter Notebook interface, featuring a sidebar with 'Files', 'Running', and 'Clusters' tabs, and a central list of files and folders. The list includes standard system directories like 'Anaconda3', 'Contacts', 'Desktop', etc., along with a 'project_1' folder. The bottom of the screen shows the Windows taskbar with icons for Start, File Explorer, Google Chrome, Internet Explorer, and Task View, along with system status icons like battery level, signal strength, and volume.

File/Folder	Last Modified
Anaconda3	2時間前
Contacts	2年前
Desktop	9日前
Documents	2時間前
Downloads	2時間前
Dropbox	2時間前
Favorites	2年前
Links	1年前
Music	2年前
Pictures	1年前
project_1	8ヶ月前
REACHit	3年前
Roaming	4年前
Saved Games	2年前
Searches	2年前
Videos	2年前

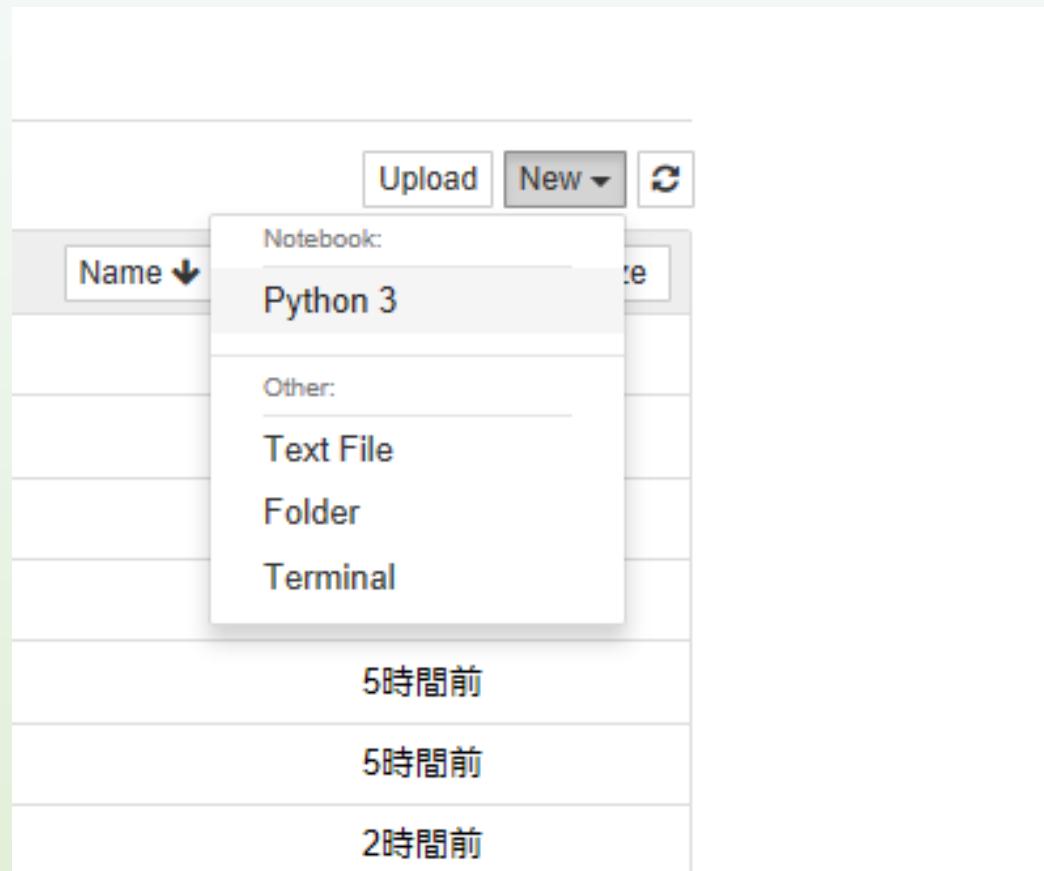
JIEP_jupyter_basic.ipynb

このプレゼンで使用するサンプルプログラムは、全て

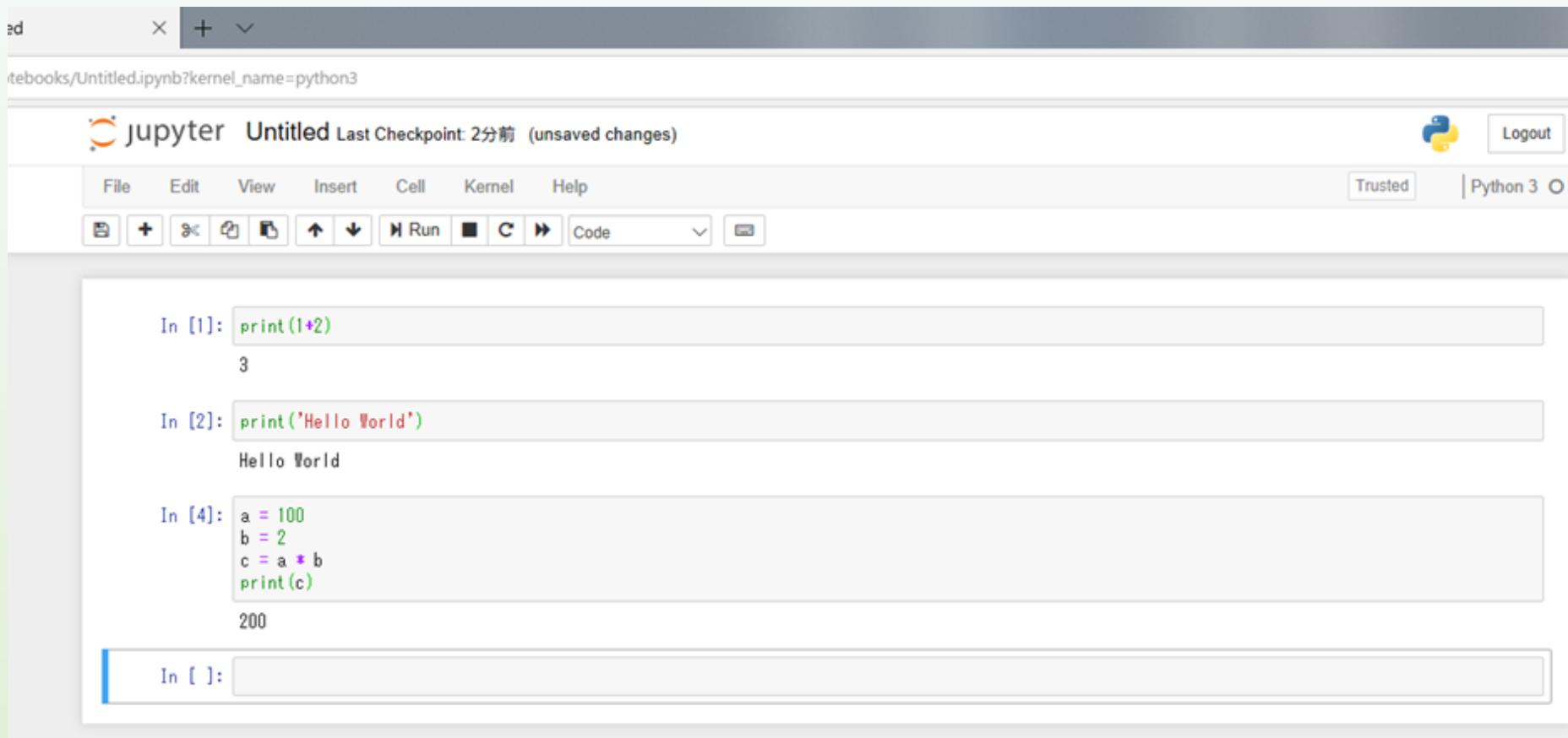
https://github.com/moritoshi/JIEP2018_Nov_seminar

においてあります。

プログラムのコーディング



プログラムのコーディング



The screenshot shows a Jupyter Notebook interface with the following details:

- Header:** "jupyter Untitled Last Checkpoint: 2分前 (unsaved changes)"
- Toolbar:** File, Edit, View, Insert, Cell, Kernel, Help, Trusted, Python 3
- Cells:**
 - In [1]: `print(1+2)`
3
 - In [2]: `print('Hello World')`
Hello World
 - In [4]:

```
a = 100
b = 2
c = a * b
print(c)
```


200
 - In []: (Empty cell)

* 実行は, “Run”をクリック
あるいは, Ctl + Enter か Shift + Enter

説明の記載 (Markdown)

notebooks/Untitled.ipynb:kernel_name=python3

The screenshot shows a Jupyter Notebook interface with three code cells:

- In [1]: `print(1+2)` → Output: 3
- In [2]: `print('Hello Wo` → Output: Hello World
- In [4]: `a = 100
b = 2
c = a * b
print(c)` → Output: 200

The "Cell" menu is open over the fourth cell (In [4]). The "Cell Type" submenu is expanded, showing "Code" (selected), "Markdown", and "Raw NBConvert".

説明の記載 (Markdown)

/notebooks/Untitled.ipynb?kernel_name=python3

The screenshot shows a Jupyter Notebook interface with the following details:

- Header:** jupyter Untitled Last Checkpoint: 10分前 (autosaved), Logout
- Toolbar:** File, Edit, View, Insert, Cell, Kernel, Help, Trusted, Python 3
- Cells:**
 - In [1]: `print(1+2)`
3
 - In [2]: `print('Hello World')`
Hello World
 - In [4]:

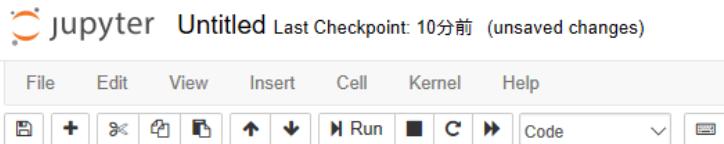
```
a = 100
b = 2
c = a * b
print(c)
```


200
- Note:** A green-bordered box contains the text:

Mark Downによる説明の記載
本日は時間が無いので、*Mark Down*については、別途調べてください。

説明の記載 (Markdown)

3/notebooks/Untitled.ipynb?kernel_name=python3



In [1]: `print(1+2)`

3

In [2]: `print('Hello World')`

Hello World

In [4]: `a = 100
b = 2
c = a * b
print(c)`

200

Mark Down による説明の記載

本日は時間が無いので, *Mark Down*については, 別途調べてください.

In []:

File Edit View Insert Cell Kernel Help

Trusted

Python 3



```
b = 2  
c = a * b  
print(c)
```

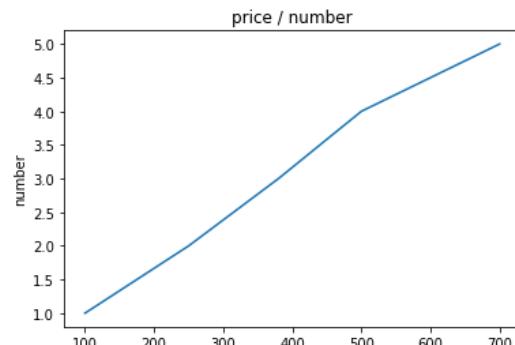
200

Mark Down による説明の記載

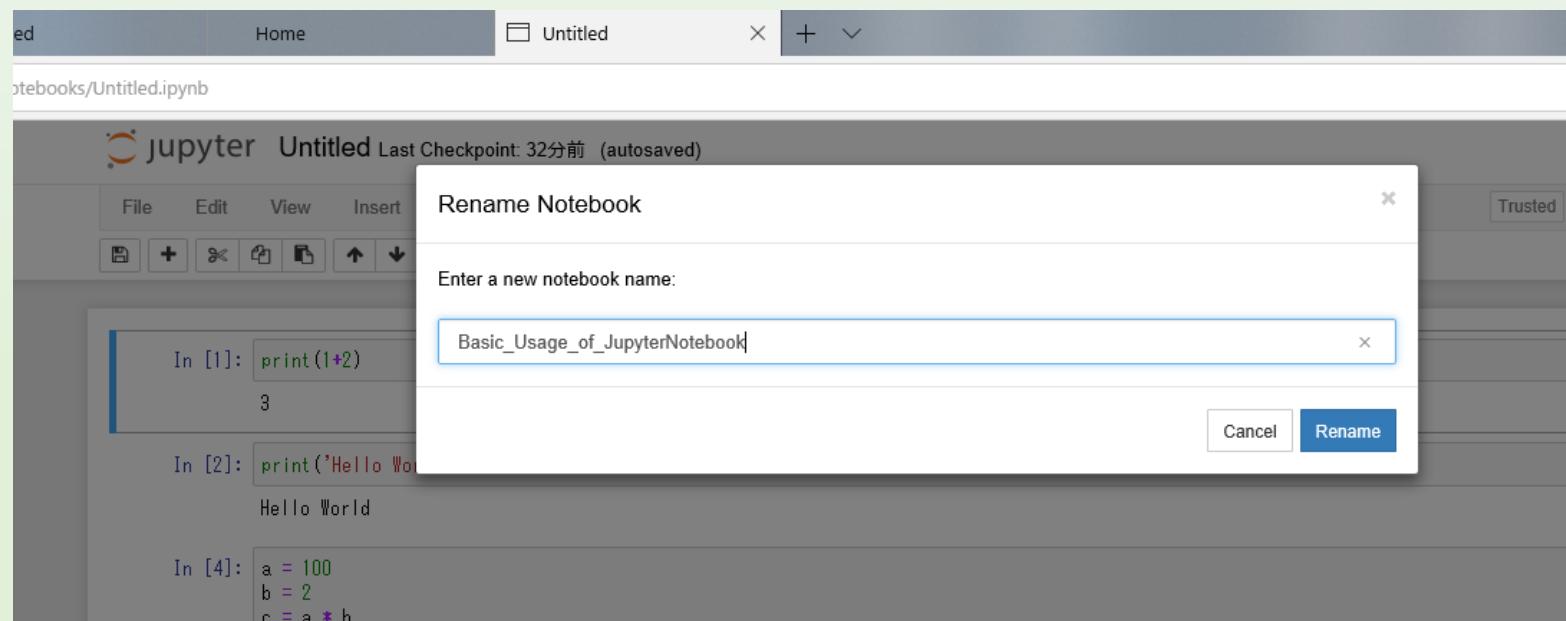
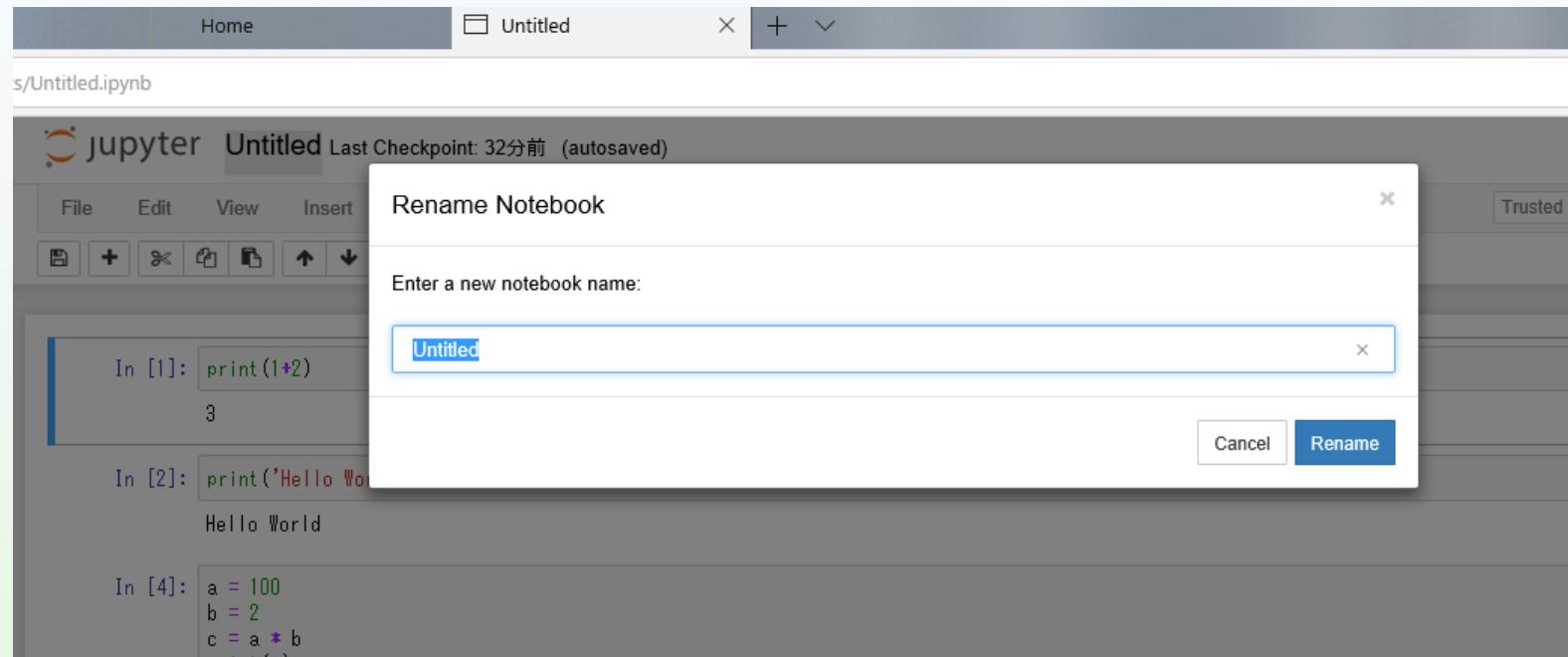
本日は時間が無いので、*Mark Down*については、別途調べてください。

In [10]: %matplotlib inline

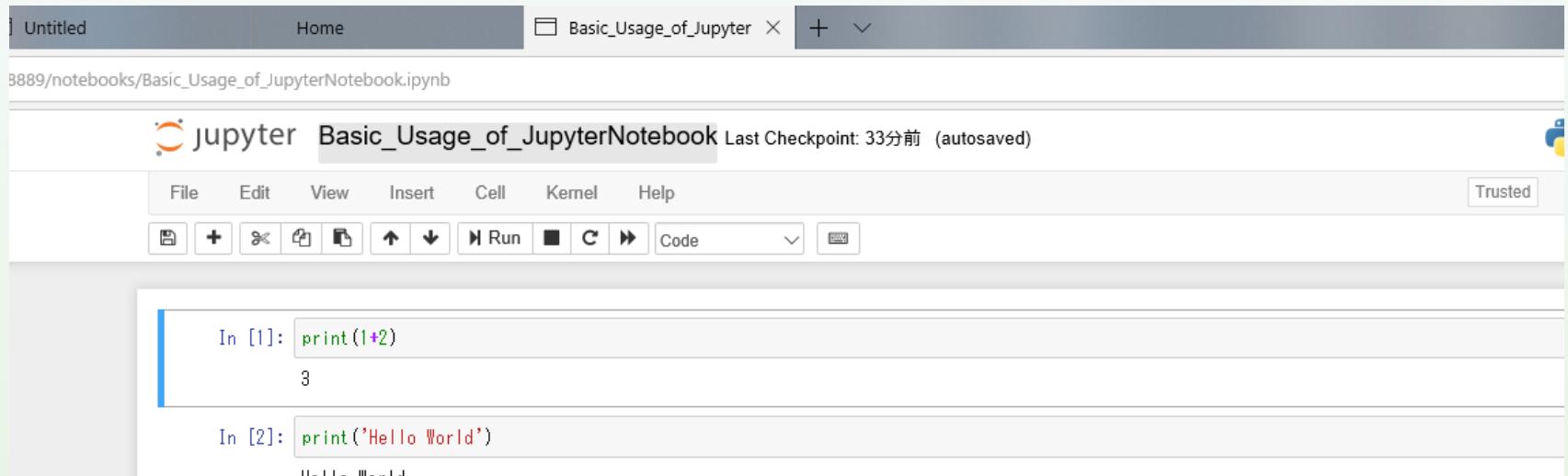
```
import matplotlib.pyplot as plt  
  
price = [100, 250, 380, 500, 700]  
number = [1, 2, 3, 4, 5]  
  
# グラフを書く  
plt.plot(price, number)  
  
# グラフのタイトル  
plt.title("price / number")  
  
# x軸のラベル  
plt.xlabel("price")  
  
# y軸のラベル  
plt.ylabel("number")  
  
# 表示する  
plt.show()
```



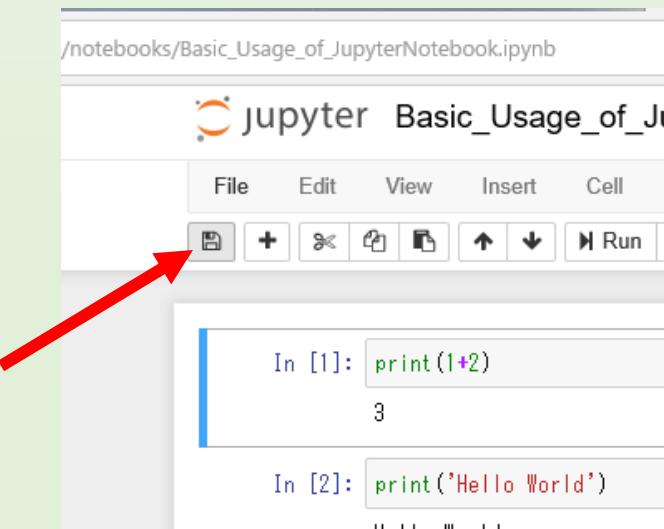
このように, jupyter notebook を使うことで, プログラム(ソースコード)とその説明文(文章), 実行結果をまとめておく(保存しておく)ことができるので, 様々なデータ解析を行い, その試行錯誤の結果を記録し, また, 他人に説明することができます.



Notebook の名前の変更



Notebook の保存



Notebook (jpynbファイル)はデフォルトで、

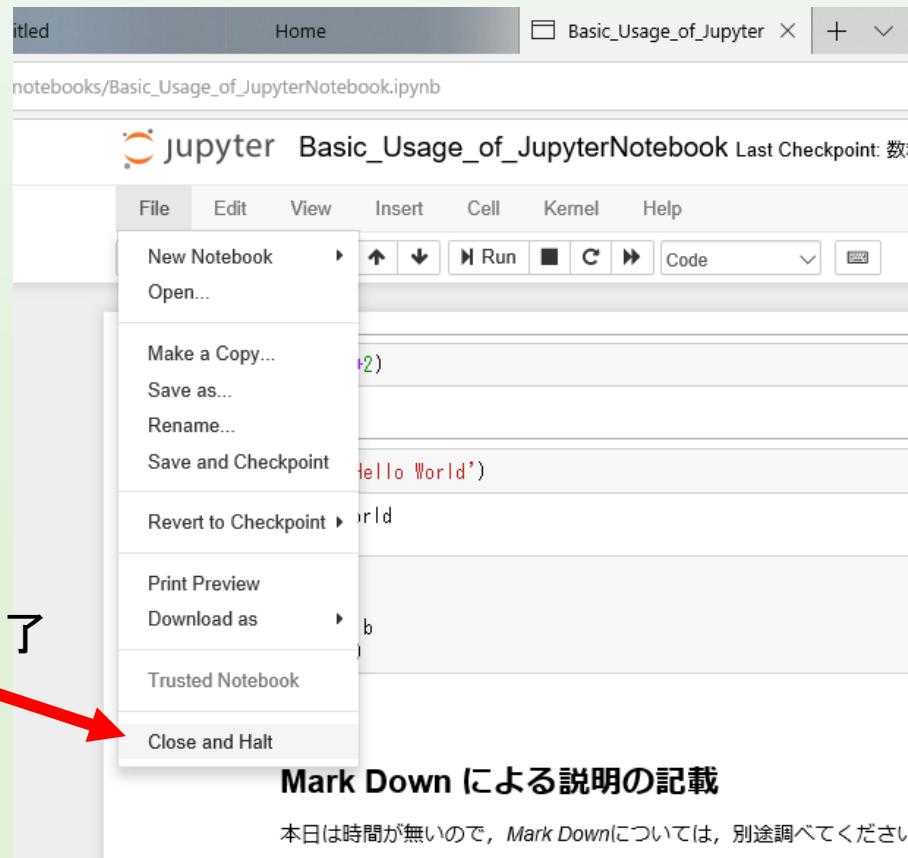
Windows7

C:¥SPB_Data¥

Windows10

C:¥Users(ユーザー)¥"user_name"¥

ここに、jpynb ファイルができるていることを確認してください。

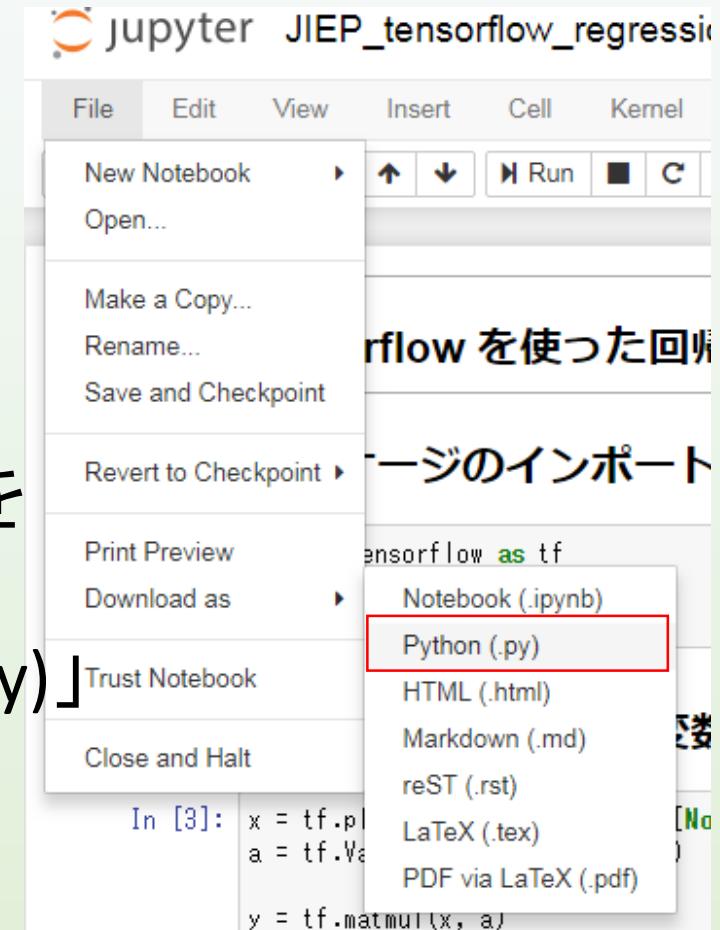


なお,
jupyter notebook の ファイルは,

*****.ipynb

というファイル名で保存されます。
このファイルから, python のコードを
抽出したい(変換したい)場合は,
「File」→「Download as」→「Python(py)」
で

*****.py



ファイルがダウンロードできます.

コピー & ペーストでも良いですが. . .

jupyter notebook を使って python の基礎を学びましょう

JIEP_python_basic.ipynb

1. 文字列と数値(型)
2. 変数
3. 配列(リスト)
4. 条件分岐
5. 繰り返し(for 文)
6. 関数
7. メソッド (関数とメソッドの違い)

まずは、とにかく、
tensorflow を動かしみてみましょう

[JIEP_tensorflwo_regression.jpynb](#)

x	月	1	2	3	4	5	6	7	8	9	10	11	12
y	温度	3.1	-0.8	4.2	9.9	17.3	24.2	29.0	35.2	32.0	18.6	9.1	4.6

回帰曲線

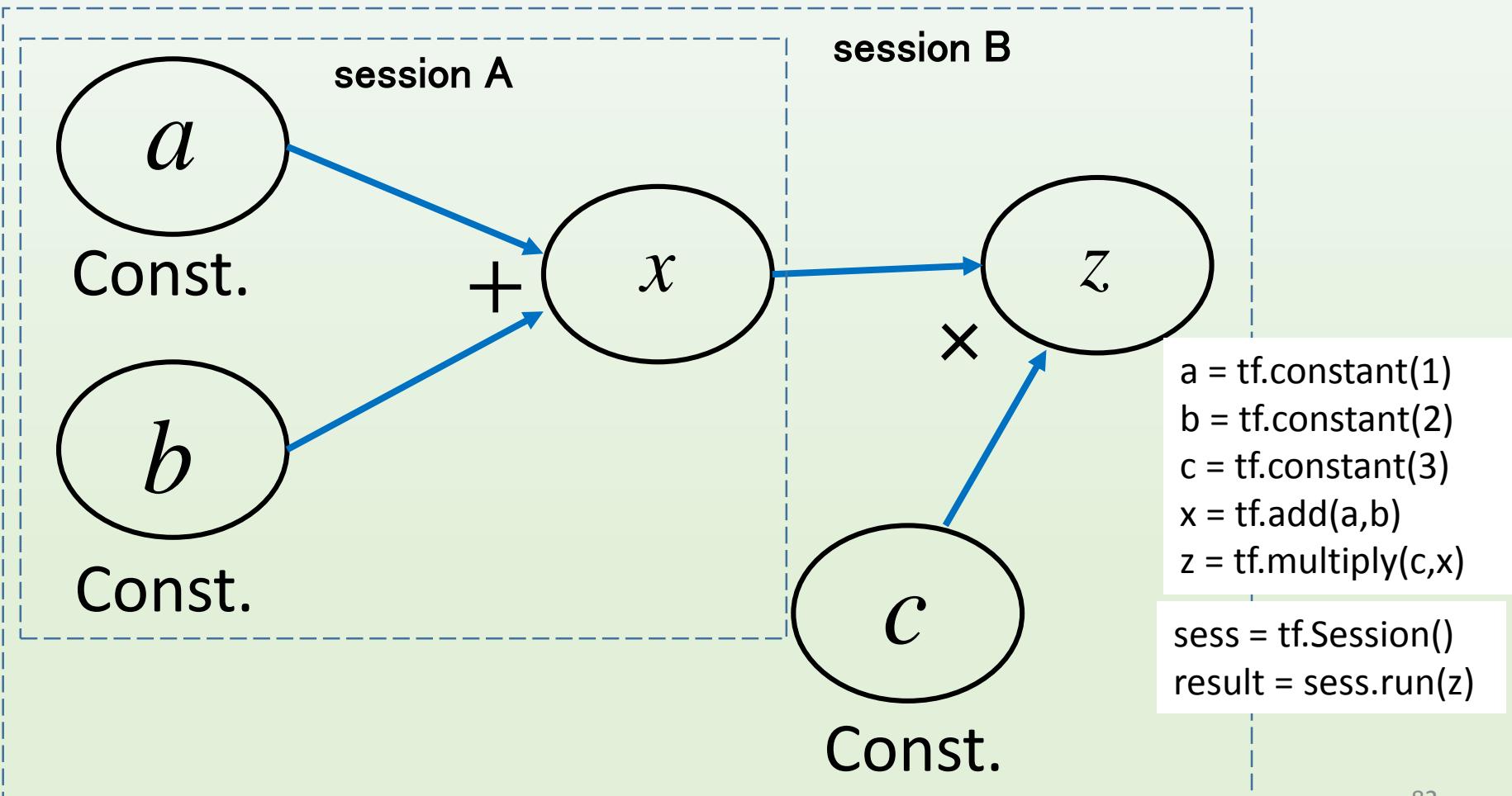
$$y = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4$$

を求めてみる。

「セッション」と「データフロー グラフ」について

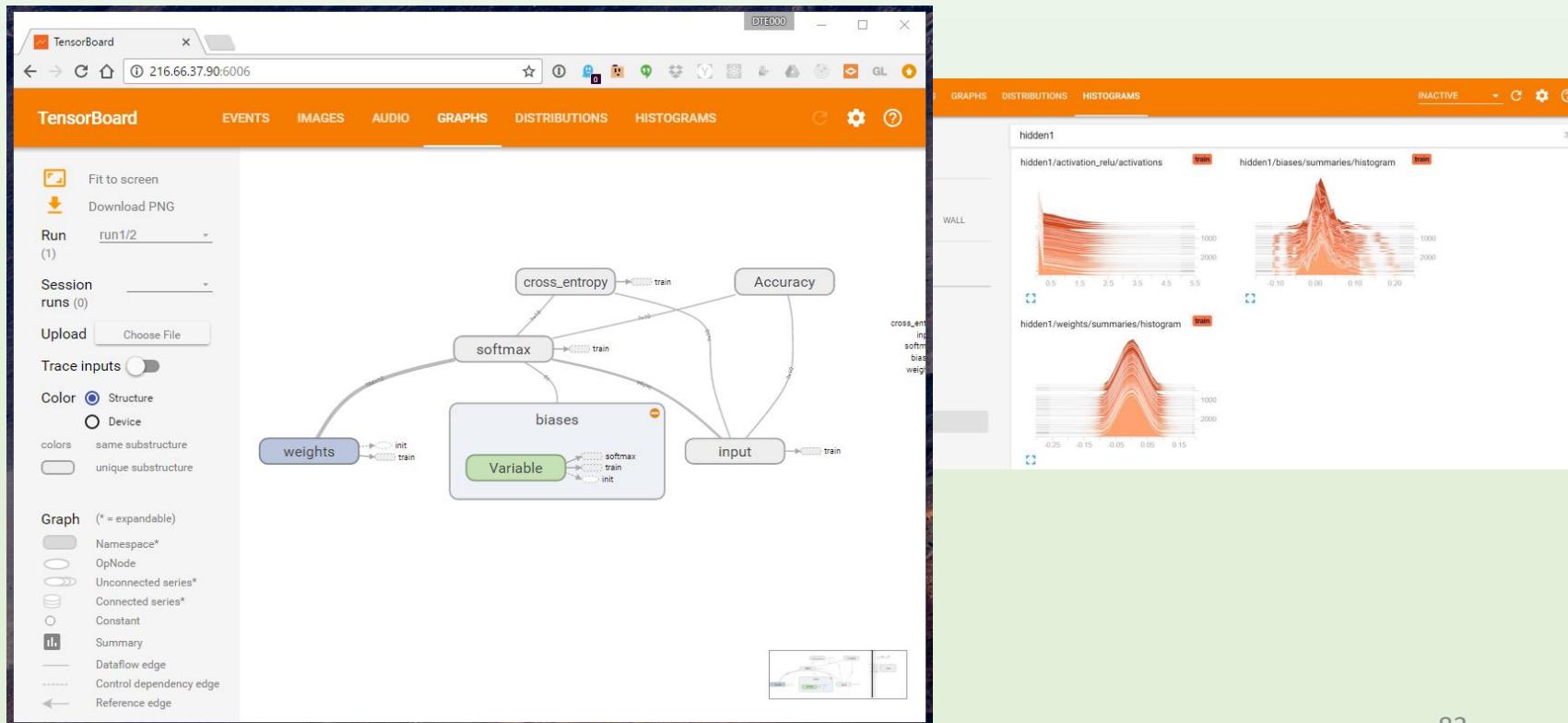
JIEP_tensorboard_basic_SESSION.ipynb

$$z = (a + b) \times c \rightarrow z = (a + b) * c$$



tensorflow の可視化 —TensorBoardを動かしてみましょう—

JIEP_tensorflow_basic_TENSORBOARD.ipynb



Tensor Board の起動方法

Anaconda にて, 「Open Terminal」
ターミナルで

`tensorboard --logdir=*****`

を入力(***** は, Tensorflow の実行結果を保存
したファイル).

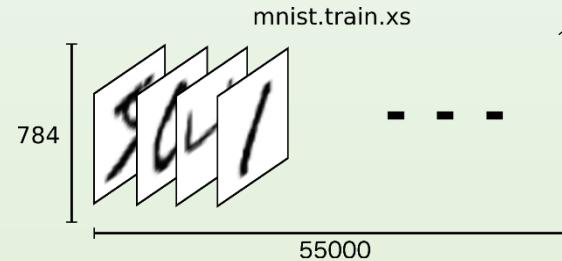
Jupyter Notebookにて, 「New」→「Terminal」でも動くはず.

Wenブラウザで, ターミナルに表示された
http://*****:6006 を開く.

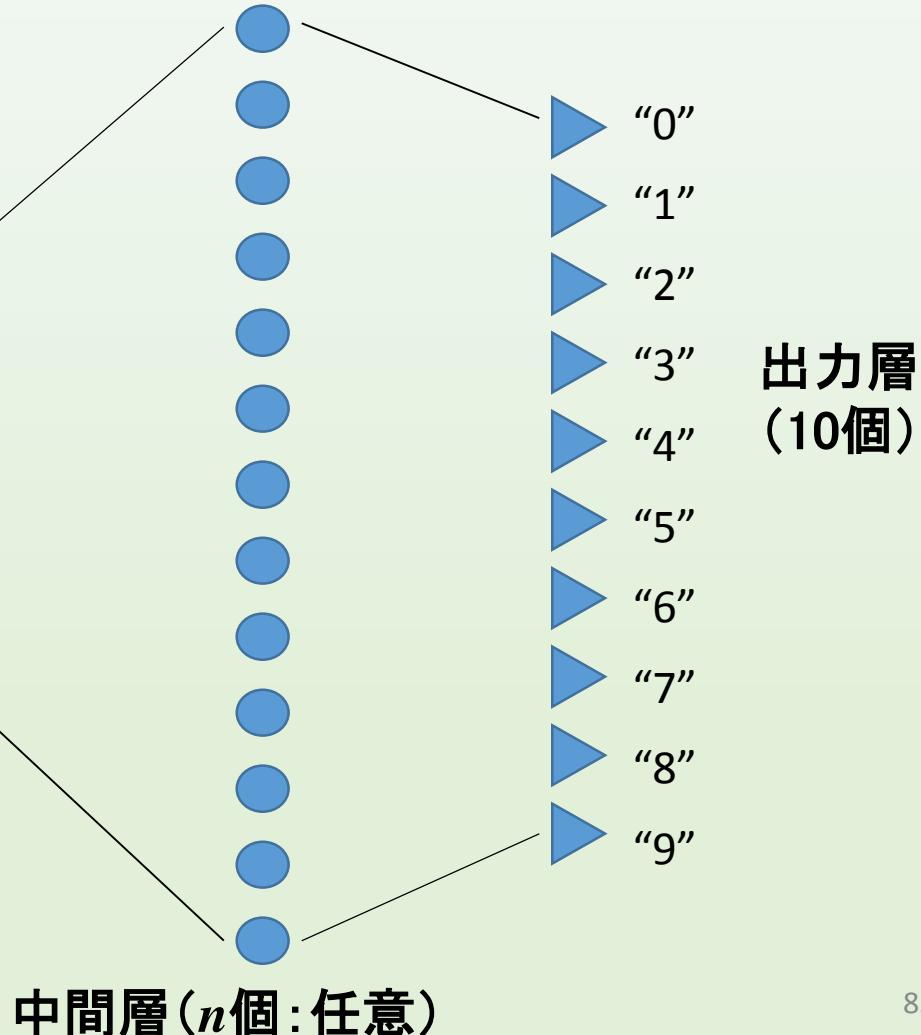
Deep Learning をMNISTに適用してみましょう

JIEP_tensorflow_nn_tanso-MNIST.ipynb

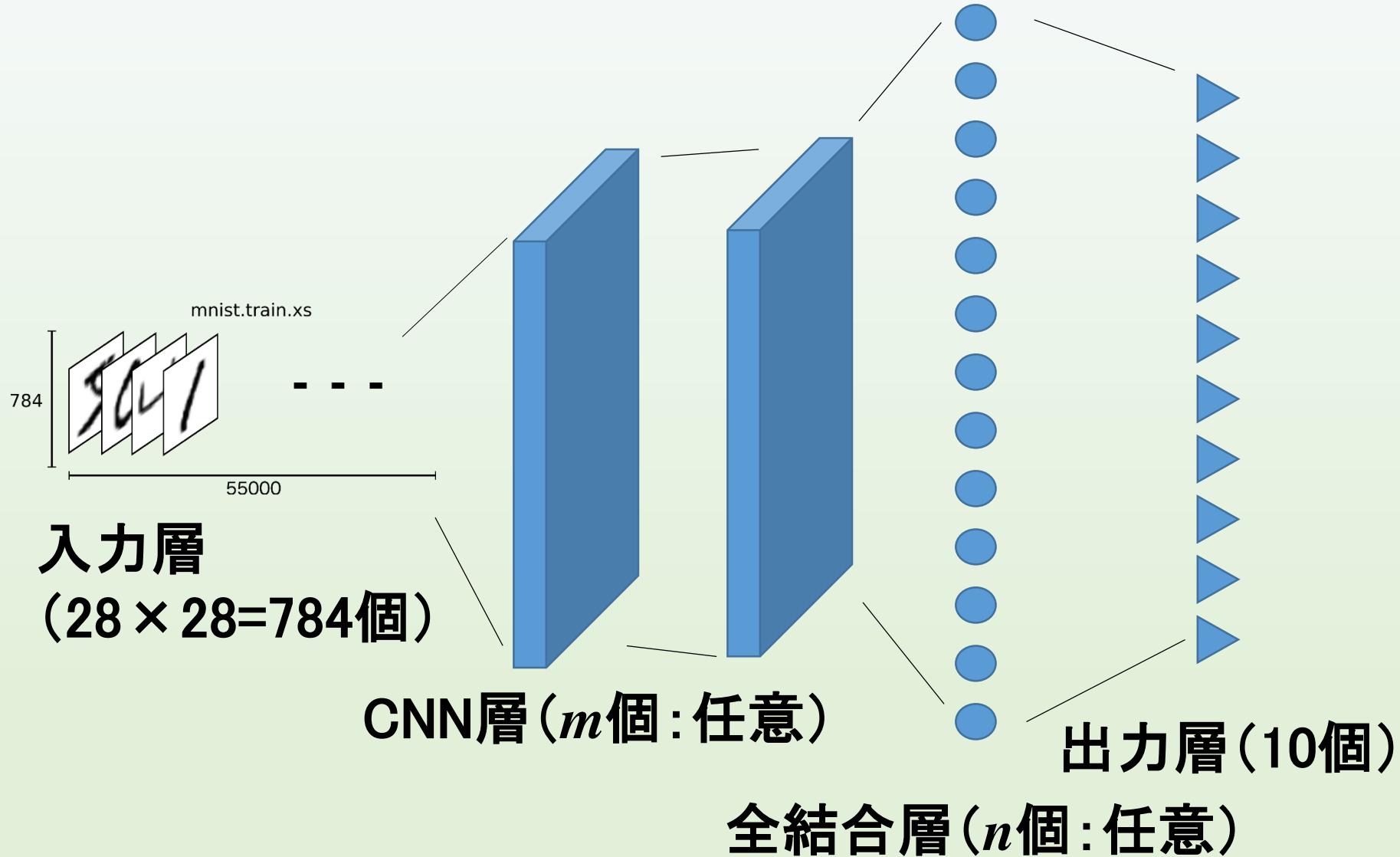
手書き文字認識 (CNN無しのDeep Learning)



入力層
($28 \times 28 = 784$ 個)

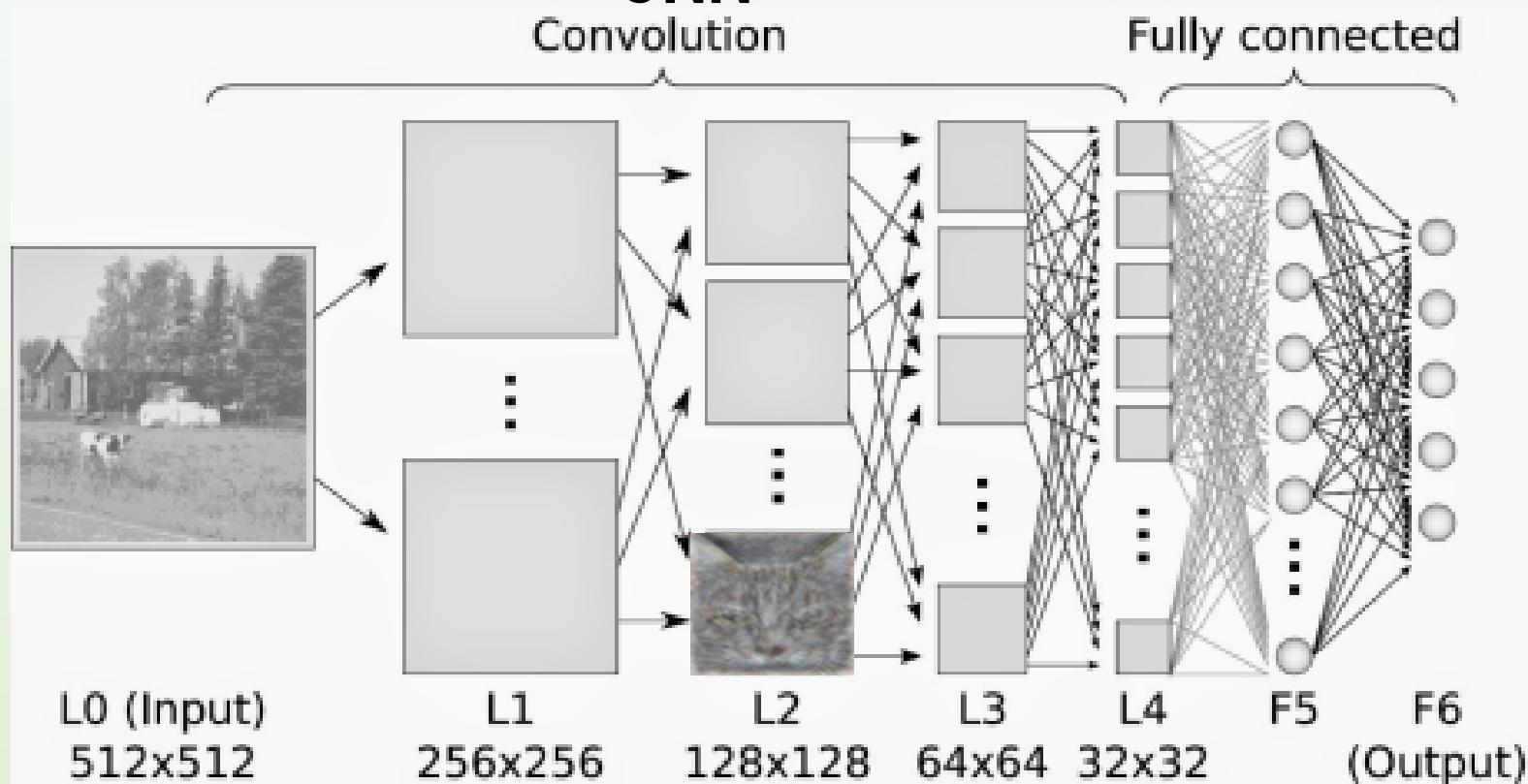


手書き文字認識(CNN無有りのDeep Learning)



自動特徴抽出

CNN



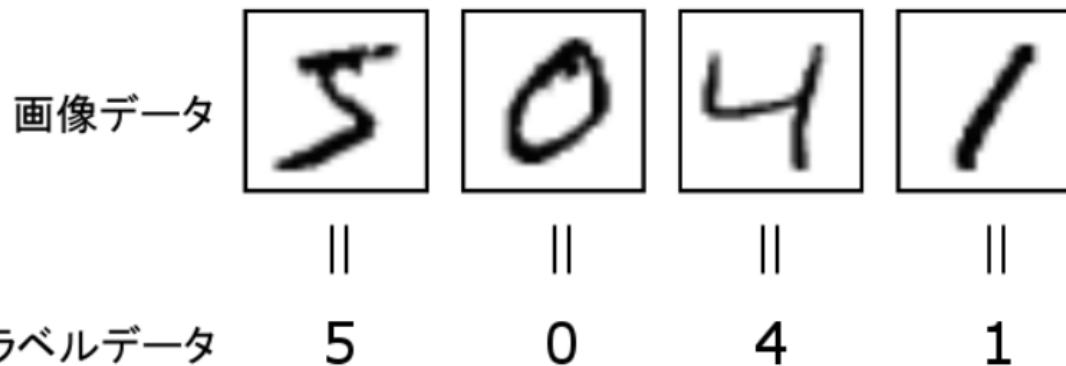
<https://www.quora.com/What-is-a-convolutional-neural-network>

CNN と Deep Learning

MNISTデータ

Mixed National Institute of Standards and Technology database

オリジナル配布元: <http://yann.lecun.com/exdb/mnist/>



それらのペアは、学習用に60,000個、検証用に10,000個の数だけ提供されています。

MNISTデータ
└ 学習用データ (60,000個)
 |
 | └ 画像データ
 | └ ラベルデータ
 └ 検証用データ (10,000個)
 |
 | └ 画像データ
 | └ ラベルデータ

4	4	4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9	9	9	9

MNISTデータ構造

ラベルデータ (train-labels-idx1-ubyte / t10k-labels-idx1-ubyte)

ラベルデータが保存されたファイル(train-labels-idx1-ubyte / t10k-labels-idx1-ubyte)は、次のような仕様になっています。

offset	type	value	description
0000	32 bit integer	0x00000801(2049)	識別子(定数)
0004	32 bit integer	60000 or 10000	ラベルデータの数
0008	unsigned byte	0 ~ 9	1つ目のデータのラベル
0009	unsigned byte	0 ~ 9	2つ目のデータのラベル
....
xxxx	unsigned byte	0 ~ 9	最後のデータのラベル

ラベルは、画像が何の数字を表すかの 0 から 9 までの数値です。

画像データ (train-images-idx3-ubyte / t10k-images-idx3-ubyte)

画像データが保存されたファイル(train-images-idx3-ubyte / t10k-images-idx3-ubyte)は、次のような仕様になっています。

offset	type	value	description
0000	32 bit integer	0x00000803(2051)	識別子(定数)
0004	32 bit integer	60000	画像データの数
0008	32 bit integer	28	1画像あたりのデータ行数
0012	32 bit integer	28	1画像あたりのデータ列数
0016	unsigned byte	0 ~ 255	1つめの画像の1ピクセル目の値
0017	unsigned byte	0 ~ 255	1つめの画像の2ピクセル目の値
....
xxxx	unsigned byte	0 ~ 255	最後の画像の784ピクセル目の値

ピクセルの値は、0 から 255 までの値で、0 が白を、255 が黒を表します。

tensorflow の TensorBoard を表示してみましょう.

JIEP_tensorflow_nn_tanso-MNIST-TENSORBORD.ipynb

/tmp/data の下に mnist_sl_logs ができます.

先ほどと同様に、ターミナルから

```
tensorboard --logdir=/tmp/mnist_sl_logs
```

と入力.

その後表示されたURLをブラウザに入力

* 左側のツールバーの文字が重なるときは、ブラウザの表示サイズを調整する。

tensorflow で, MNIST 以外の応用をやってみる — Titanic 問題(Kaggle のベンチマーク問題) —



<https://www.kaggle.com/>
データサイエンスと機械学習の総合サイト

JIEP_tensorflow_TITANIC.ipynb

scikit-learn がインストールされていない場合,
インストールしてください.

タイタニックのデータ

./input/train.csv

./input/test.csv

を入れてください.

TFLearnについて

TFLearn は, tensorflow の“ラッパー(API)”です.
tensorflow のセッションなどを隠蔽し,
簡単な記述で Deep Learning などが実行できます.

TFLearnについて

<http://tflearn.org>

The screenshot shows a web browser displaying the TFLearn documentation at <http://tflearn.org>. The page title is "TFLearn | TensorFlow Deep Learn". The URL in the address bar is "tflearn.org". The page content includes a sidebar with navigation links for Home, TFLearn, Quick overview, Where to Start?, Model Visualization, Sources, Contributions, License, Index, Installation, Getting Started, Tutorials, Examples, Models, Deep Neural Network, Generative Neural Network, Layers, Core Layers, Convolutional Layers, Recurrent Layers, Normalization Layers, GitHub, and Next ». The main content area features a heading "TFLearn: Deep learning library featuring a higher-level API for TensorFlow." followed by a paragraph about its modular and transparent nature built on top of TensorFlow. It lists features such as an easy-to-use API, fast prototyping, full transparency, powerful helper functions, and easy graph visualization. A note at the bottom states that the high-level API supports most recent deep learning models and is compatible with TensorFlow v1.0 and over. The browser taskbar at the bottom shows various pinned icons.

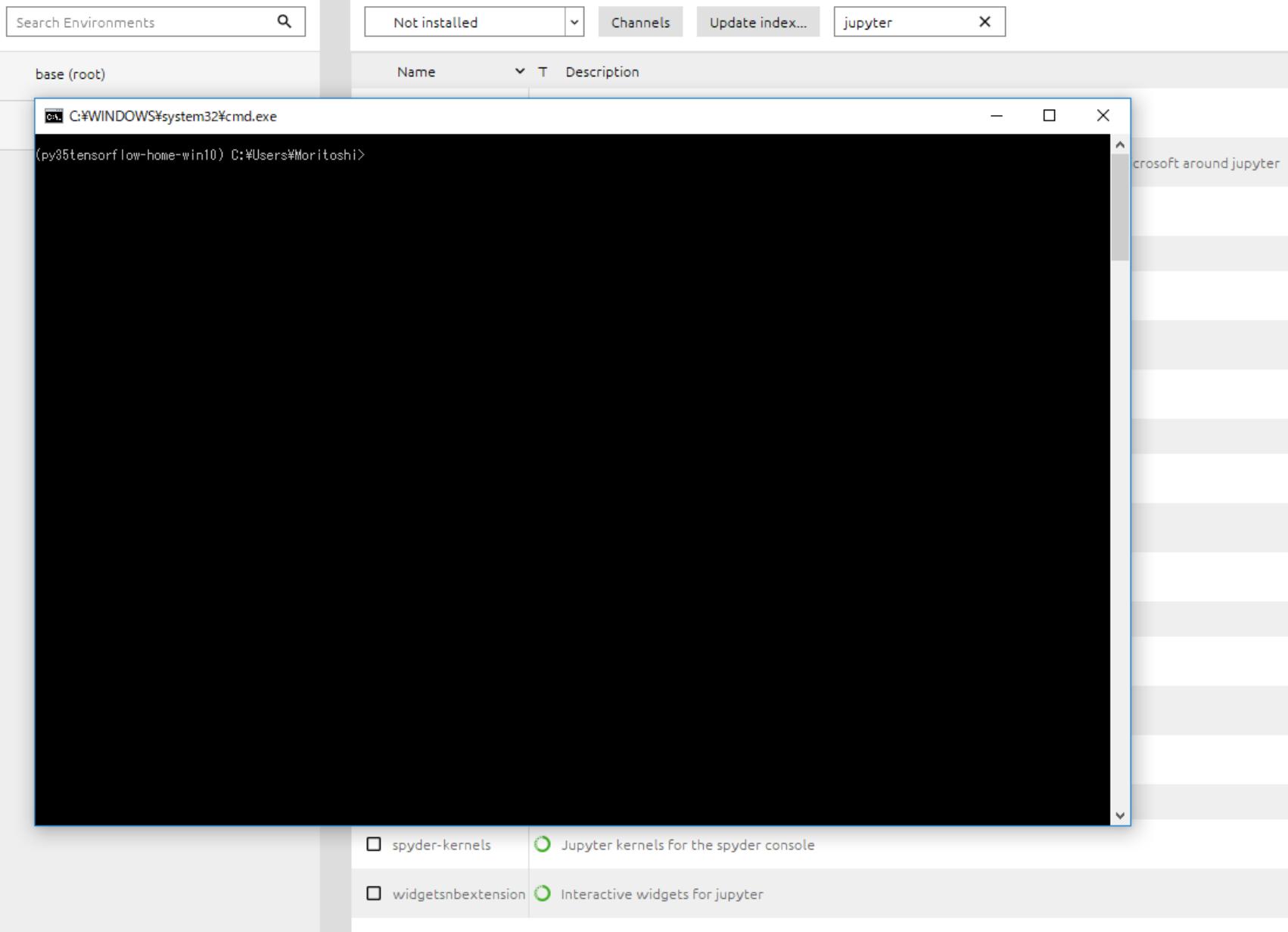
TFLearnのインストール

ANACONDA NAVIGATOR

Search Environments Not installed Channels Update index... jupyter

base (root)

Name	T	Description
jupyter_console	<input type="checkbox"/>	Jupyter terminal console
jupyter_dashboard	<input type="checkbox"/>	
jupyter_kernel_gateway	<input type="checkbox"/>	Jupyter kernel gateway
metakernel	<input type="checkbox"/>	Metakernel for jupyter.



```
C:\WINDOWS\system32\cmd.exe
```

```
(py35tensorflow-home-win10) C:\Users\Moritoshi>pip install tflearn
```

```
C:\WINDOWS\system32\cmd.exe - pip install tflearn
```

```
(py35tensorflow-home-win10) C:\Users\Moritoshi>pip install tflearn
Collecting tflearn
  Downloading https://files.pythonhosted.org/packages/16/ec/e9ce1b52e71f6dff3bd844f020cef7140779e783ab27512ea7c7275ddee5/tflearn-0.3.2.tar.gz (98kB)
    100% |██████████| 102kB 1.3MB/s
```

```
C:\WINDOWS\system32\cmd.exe - pip install tflearn
```

```
(py35tensorflow-home-win10) C:\Users\Moritoshi>pip install tflearn
Collecting tflearn
  Downloading https://files.pythonhosted.org/packages/16/ec/e9ce1b52e71f6dff3bd844f020cef7140779e783ab27512ea7c7275ddee5/tflearn-0.3.2.tar.gz (98kB)
    100% |██████████| 102kB 1.3MB/s
Requirement already satisfied: numpy in c:\users\moritoshi\anaconda3\envs\py35tensorflow-home-win10\lib\site-packages (from tflearn) (1.15.2)
Requirement already satisfied: six in c:\users\moritoshi\anaconda3\envs\py35tensorflow-home-win10\lib\site-packages (from tflearn) (1.11.0)
Collecting Pillow (from tflearn)
  Downloading https://files.pythonhosted.org/packages/bd/39/c76eaf781343162bdb1cf4854cb3bd5947a87ee44383e5acd6c48d69c4a1/Pillow-5.3.0-cp36-cp36m-win_amd64.whl (1.6MB)
    100% |██████████| 1.6MB 2.0MB/s
Building wheels for collected packages: tflearn
```

```
C:\WINDOWS\system32\cmd.exe - pip install tflearn

(py35tensorflow-home-win10) C:\Users\Moritoshi>pip install tflearn
Collecting tflearn
  Downloading https://files.pythonhosted.org/packages/16/ec/e9ce1b52e71f6dff3bd944f020cef7140779e783ab27512ea7c7275ddee5/tflearn-0.3.2.tar.gz (98kB)
    100% |██████████| 102kB 1.3MB/s
Requirement already satisfied: numpy in c:\users\moritoshi\anaconda3\envs\py35tensorflow-home-win10\lib\site-packages (from tflearn) (1.15.2)
Requirement already satisfied: six in c:\users\moritoshi\anaconda3\envs\py35tensorflow-home-win10\lib\site-packages (from tflearn) (1.11.0)
Collecting Pillow (from tflearn)
  Downloading https://files.pythonhosted.org/packages/bd/39/c76eaf781343162bdb1cf4854cb8bd5947a87ee44363e5acd8c48d89c4a1/Pillow-5.3.0-cp36-cp36m-win_amd64.whl (1.6MB)
    100% |██████████| 1.6MB 2.0MB/s
Building wheels for collected packages: tflearn
  Running setup.py bdist_wheel for tflearn ... done
  Stored in directory: C:\Users\Moritoshi\AppData\Local\pip\Cache\wheels\d0\f8\69\0ef3ee395aac2e5d15d89efd29a8a216f3c27767b43b72c008
Successfully built tflearn
twisted 18.7.0 requires PyHamcrest>=1.9.0, which is not installed.
mkl-random 1.0.1 requires cython, which is not installed.
tensorflow 1.10.0 has requirement numpy<=1.14.5,>=1.13.3, but you'll have numpy 1.15.2 which is incompatible.
tensorflow 1.10.0 has requirement setuptools<=39.1.0, but you'll have setuptools 40.4.3 which is incompatible.
Installing collected packages: Pillow, tflearn
Successfully installed Pillow-5.3.0 tflearn-0.3.2
```

```
C:\WINDOWS\system32\cmd.exe

(py35tensorflow-home-win10) C:\Users\Moritoshi>pip install tflearn
Collecting tflearn
  Downloading https://files.pythonhosted.org/packages/16/ec/e9ce1b52e71f6dff3bd944f020cef7140779e783ab27512ea7c7275ddee5/tflearn-0.3.2.tar.gz (98kB)
    100% |██████████| 102kB 1.3MB/s
Requirement already satisfied: numpy in c:\users\moritoshi\anaconda3\envs\py35tensorflow-home-win10\lib\site-packages (from tflearn) (1.15.2)
Requirement already satisfied: six in c:\users\moritoshi\anaconda3\envs\py35tensorflow-home-win10\lib\site-packages (from tflearn) (1.11.0)
Collecting Pillow (from tflearn)
  Downloading https://files.pythonhosted.org/packages/bd/39/c76eaf781343162bdb1cf4854cb8bd5947a87ee44363e5acd8c48d89c4a1/Pillow-5.3.0-cp36-cp36m-win_amd64.whl (1.6MB)
    100% |██████████| 1.6MB 2.0MB/s
Building wheels for collected packages: tflearn
  Running setup.py bdist_wheel for tflearn ... done
  Stored in directory: C:\Users\Moritoshi\AppData\Local\pip\Cache\wheels\d0\f8\69\0ef3ee395aac2e5d15d89efd29a8a216f3c27767b43b72c008
Successfully built tflearn
twisted 18.7.0 requires PyHamcrest>=1.9.0, which is not installed.
mkl-random 1.0.1 requires cython, which is not installed.
tensorflow 1.10.0 has requirement numpy<=1.14.5,>=1.13.3, but you'll have numpy 1.15.2 which is incompatible.
tensorflow 1.10.0 has requirement setuptools<=39.1.0, but you'll have setuptools 40.4.3 which is incompatible.
Installing collected packages: Pillow, tflearn
Successfully installed Pillow-5.3.0 tflearn-0.3.2
You are using pip version 10.0.1, however version 18.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

(py35tensorflow-home-win10) C:\Users\Moritoshi>
```

C:\WINDOWS\system32\cmd.exe

(py35tensorflow-home-win10) C:\Users\Moritoshi>pip list

Package	Version
absl-py	0.5.0
alabaster	0.7.11
appdirs	1.4.3
asn1crypto	0.24.0
astor	0.7.1
attrs	18.2.0
Automat	0.7.0
Babel	2.8.0
certifi	2018.4.16
cffi	1.11.5
chardet	3.0.4
cryptography	2.4.2
docutils	0.14.1
enum34	1.1.6
feedparser	5.2.0
greenlet	0.4.13
idna	2.8
itsdangerous	1.1.0
jedi	0.12.1
MarkupSafe	1.1.1
more-itertools	5.0.0
numpy	1.15.4
packaging	18.3
pbr	5.1.3
pexpect	4.6.0
pip	18.1.1
platformdirs	1.1.0
psycopg2	2.8.4
pyasn1	0.4.7
pyasn1-modules	0.2.7
Pygments	2.3.1
pylint	2.3.1
pytz	2018.7
PyWavelets	1.0.1
PyYAML	3.13
requests	2.22.0
retrying	1.3.3
scipy	1.2.1
setuptools	41.0.1
six	1.12.0
sqlalchemy	1.2.12
statsmodels	0.9.0
tensorboard	1.10.0
tensorflow	1.10.0
termcolor	1.1.0
terminado	0.8.1
testpath	0.4.2
tflearn	0.3.2
toolz	0.9.0
tornado	5.1.1
traitlets	4.3.2
Twisted	18.7.0
urllib3	1.23

```
import tensorflow as tf
import tflearn

import tflearn.datasets.mnist as mnist

from matplotlib import pyplot as plt
from matplotlib import cm
import numpy as np

trainX, trainY, testX, testY =
mnist.load_data('./mnist', one_hot=True)
```

Windows7

C:¥SPB_Data¥

Windows10

C:¥Users(ユーザー)¥"user_name"¥

ここに、mnist (約11MB) というフォルダができていれば成功！

mnist データの説明 <https://weblabo.oscasierra.net/python/ai-mnist-data-detail.html>

TFLearn で, MNISTをやってみる

[JIEP_TFlearn.ipynb](#)

[JIEP_TFlearn_basic.ipynb](#)

[JIEP_TFlearn_basic.ipynb](#)

TFLearn 用にMNISTのデータを読み込む. データのチェック
⇒ここではまだ, TFLearn は使っていない(準備).

[JIEP_TFlearn.ipynb](#)

まずは実行.

重みの取り出しあは,

```
w = model.get_weights(net.W)  
print(w)
```

TFLearn での TensorBoard 表示

```
model = tflearn.DNN(net, tensorboard_verbose=3)
```

`tensorboard_verbose=3` は tensorboard 用ログの情報レベル(0が最低, 3が最高)

```
model.fit(trainX, trainY, n_epoch=2, batch_size=100, validation_set=0.1,  
show_metric=True, run_id='TSBD')
```

`run_id='TSBD'` は tensroboard 用ログの出力フォルダ名

/tmp/tflearn_logs/ の下にこのフォルダができる

tensorboard 出力は, ターミナルで, tensorboard --logdir=/tmp/tflearn_logs/TSBD と入力
⇒ hppt://マシン名:6006 が出る.

SONY Neural Network Console

プログラミング言語を使わずニューラルネットワーク(DLなど)を使えます。

ダウンロードはここから

<https://dl.sony.com/ja/>

The screenshot shows the Sony Neural Network Console homepage. At the top, there's a navigation bar with tabs for 'Neural Network Console' and 'Neural Network Libraries'. On the right side of the header, there are language selection ('EN') and sign-in ('サインイン') buttons. The main content area features a large background image of a neural network grid. In the center, the text 'Neural Network Console' is displayed. Below it, a subtitle reads: 'ニューラルネットワークを直感的に設計。学習・評価を快適に実現するディープラーニング・ツール。'. At the bottom, there are two call-to-action buttons: 'クラウドではじめる' and 'Windowsアプリではじめる'. A red arrow points from the text 'ダウンロードはここから' to the 'Windowsアプリではじめる' button. The footer contains links for 'Cloud Version', 'Windows Version', 'Blog', 'Community', 'Support', and 'Business'. The bottom navigation bar includes icons for search, file operations, and system status, along with the date '2018/11/03' and a page number '103'.

「個人情報の取り扱い」を最後までスクロールし、
メールアドレスを入力して、送信。



Neural Network Console Cloud

ディープラーニングをいつでもどこからでも利用できるクラウド版も用意しています。



メールが届いたら、インストールファイルをダウンロード。
約800MBなので、少々時間がかかります。あとは、ファイルを解等するだけです。

ソニーの無償AIソフト Neural Netwo × | N WinAPP - Neural Network Consol × M Neural Network Console Download × +

https://mail.google.com/mail/u/0/#inbox/ FMfcgxvzLWwMWzJdkCpxQClkvrNvGQzj

Gmail メールを検索

作成

受信トレイ 52,213

スター付き
スヌーズ中
重要

NEURAL NETWORK CONSOLE

【日本語は英語の後に記載しています】

Dear moritoshi.yasunaga@gmail.com

Welcome to Neural Network Console.

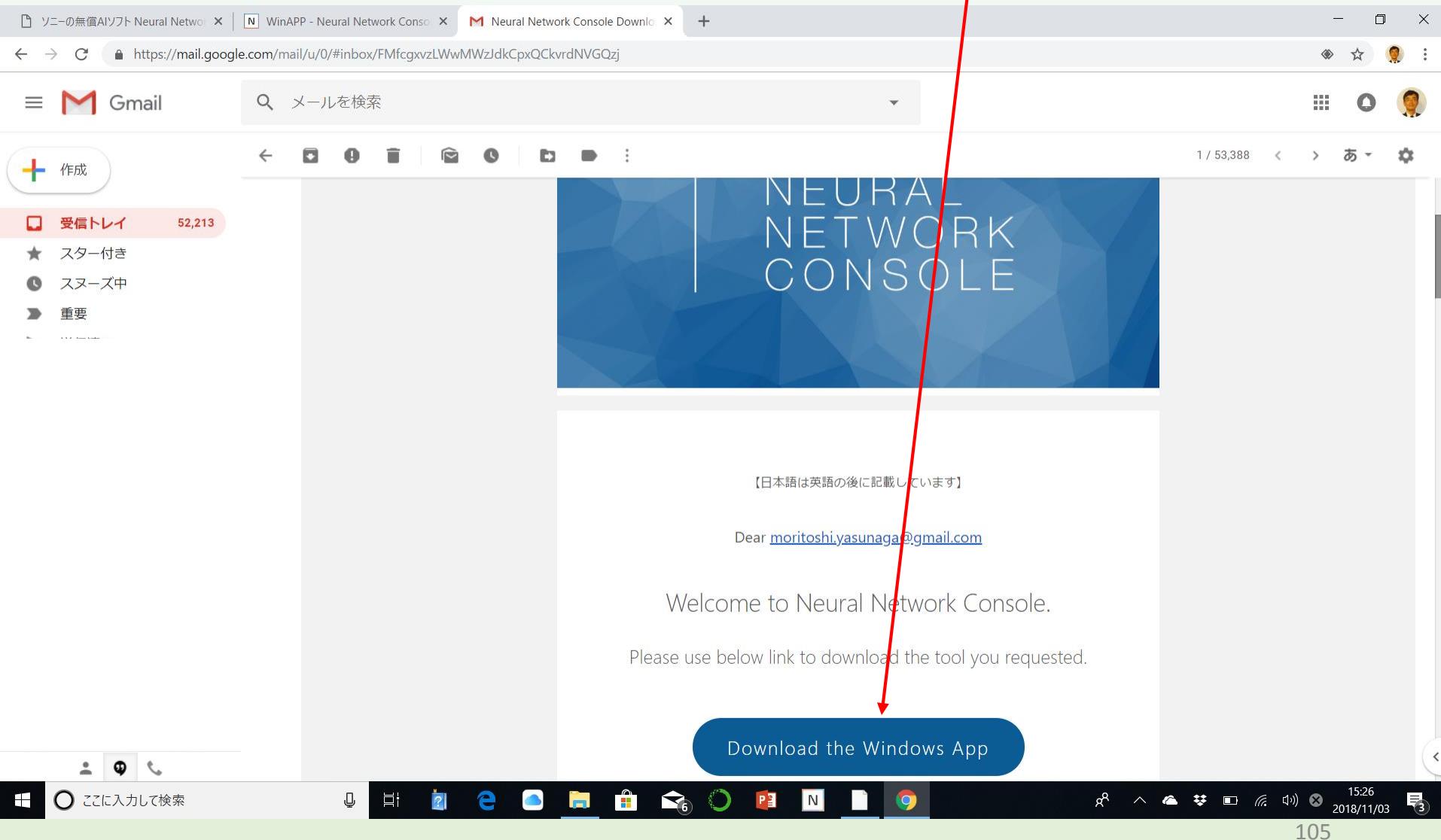
Please use below link to download the tool you requested.

Download the Windows App

ここに入力して検索

15:26 2018/11/03

105



SONY Neural Network Console

プログラミング言語を使わずニューラルネットワーク(DLなど)を使えます。

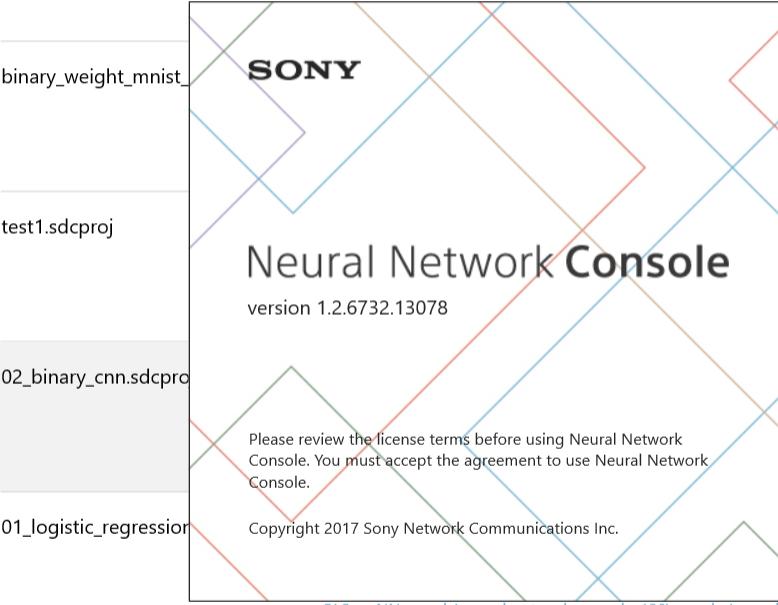
<https://dl.sony.com/ja/>

Neural Network Console

PROJECT

+ New Project Open Project Search Text Here ACTION

DATASET

semi_supervised_learning_VAT.sdc	Dataset "Labeled": mnist_training_100.csv (100 datas, 2 columns) Dataset "Unlabeled": mnist_training_unlabeled.csv (60000 datas, 1 columns) Dataset "Validation": mnist_test.csv (10000 datas, 2 columns)	2018/07/15 16:10:36
binary_weight_mnist.sdcproj		2018/07/15 15:56:10
test1.sdcproj		2018/07/15 15:55:14
02_binary_cnn.sdcproj		2018/07/15 15:54:45
01_logistic_regression.sdcproj		2018/07/15 15:45:34
06_auto_encoder.sdcproj	Inputs : x[1,28,28] Outputs : x[1,28,28] Training Dataset : small_mnist_4or9_training.csv (1500 datas, 2 columns) Validation Dataset : small_mnist_4or9_test.csv (491 datas, 2 columns)	2015/06/18 14:15:40

Training Evaluation

Overview

Statistics

Output	0
CostParameter	0
CostAdd	0
CostMultiply	0
CostMultiplyAdd	0
CostDivision	0
CostExp	0
CostIf	0

Tasks

Training: ----

Evaluation: ----



最初からいくつかのサンプルが入っているのすぐに試せます。

Neural Network Console

PROJECT

+

New Project

Open Project

Search Text Here

ACTION

Training

Evaluation

DATASET

12_residual_learning.sdcproj

Inputs : x[1,28,28] Outputs : y[1]

Training Dataset : mnist_training.csv (60000 datas, 2 columns)

Validation Dataset : mnist_test.csv (10000 datas, 2 columns)

C:\SonyNNconsole\neural_network_console_120\samples\sample_project\tutorial\basics\

LeNet.sdcproj

Dataset "Training": mnist_training.csv (60000 datas, 2 columns)

Dataset "Validation": mnist_test.csv (10000 datas, 2 columns)

C:\SonyNNconsole\neural_network_console_120\samples\sample_project\image_recognition

mnist_vae.sdcproj

Dataset "Training": mnist_training.csv (60000 datas, 2 columns)

Dataset "Validation": mnist_test.csv (10000 datas, 2 columns)

C:\SonyNNconsole\neural_network_console_120\samples\sample_project\image_generation

mnist_dcgan_with_label.sdcproj

Dataset "Training": mnist_training.csv (60000 datas, 2 columns)

Dataset "Validation": mnist_test.csv (10000 datas, 2 columns)

C:\SonyNNconsole\neural_network_console_120\samples\sample_project\image_generation

binary_connect_mnist_MLP.sdcproj

Dataset "Training": mnist_training.csv (60000 datas, 2 columns)

Dataset "Validation": mnist_test.csv (10000 datas, 2 columns)

C:\SonyNNconsole\neural_network_console_120\samples\sample_project\tutorial\binary_ne

Overview

Input
B BinaryWeightBias
R Null
B BinaryWeightBias_J
R Null
B BinaryWeightBias_J
R Null
B BinaryWeightBias_J
S Softmax
C CategoryCrossEntropy

Statistics

Output 13,093

CostParameter 319,114

CostAdd 10,014,730

CostMultiply 0

CostMultiplyAdd 0

CostDivision 10

CostExp 10

CostIf 6,144

Tasks

Training: ----

Evaluation: ----

ここに入力して検索

16:52 2018/07/15

NN(DL)の構成もブロックを積む(接続する)だけです.

N LeNet.sdcproj - Neural Network Console (NNabla)

EDIT TRAINING EVALUATION DATASET CONFIG

Components Search Text Here

IO

- Input
- Loss
- SquaredError
- HuberLoss
- AbsoluteError
- EpsilonInsensitiveLoss
- BinaryCrossEntropy
- SigmoidCrossEntropy
- CategoricalCrossEntropy
- SoftmaxCrossEntropy

Main

Training Evaluation

Overview

Statistics

Output	21,919
CostParameter	78,810
CostAdd	11,304
CostMultiply	0
CostMultiplyAdd	802,876
CostDivision	490
CostExp	490
CostIf	18,068

Tasks

Training: ----

Evaluation: ----

ここに入力して検索

16:53 2018/07/15

学習状態もオンラインで観測できます。この図では、過学習が観測されています。

N LeNet.sdcproj - Neural Network Console (NNabla)

EDIT TRAINING EVALUATION

Results History ACTION ▾ Elapsed: 00:00:01:03 Remaining: 00:00:00:00 Total: 00:00:01:03 0 / 10) Training Evaluation

20170804_140639 PARETO OPTIMAL Learning Curve Trade-off Graph: All Linear Scale Log Scale ▶ ▷ □ ▷ □

Training Validation Best Validation Multiply Add 0.007822 0.050883 0.032139 @ epoch 3 802,876 EVALUATED

Learning Curve COST TRAINING ERROR VALIDATION ERROR

Cost Epoch Error

2017-08-04 14:06:39,474 Training process is started.
python "K:\neural_network_console_170725\libs\nnabla\python\src\nnabla\utils\cli\cli.py" train
-c "K:\neural_network_console_170725\samples\sample_project\image_recognition\MNIST\LeNet.files"
-o "K:\neural_network_console_170725\samples\sample_project\image_recognition\MNIST\LeNet.files"
2017-08-04 14:06:40,488 [nnabla]: Train with contexts {'cpu': ['default'], 'cuda': ['default', 'cudnn']}
2017-08-04 14:06:42,859 [nnabla]: Training epoch 1 of 10 begin
2017-08-04 14:06:54,341 [nnabla]: epoch 1 of 10 cost=0.172375 {train_error=0.053857, valid_error=0.052542}
2017-08-04 14:07:00,701 [nnabla]: epoch 2 of 10 cost=0.050152 {train_error=0.022021, valid_error=0.010254}

Overview

I Input	.
C Convolution	—
R Relu	—
M MatMul	—
C Convolution2d	—
M MatMul2	—
T Tanh	—
A Affine	—
R Relu2	—
A Affine2	—
S Softmax	—
C CrossEntropyLoss	—

Statistics

Output	21,919
CostParameter	78,810
CostAdd	11,304
CostMultiply	0
CostMultiplyAdd	802,876
CostDivision	490
CostExp	490
CostIf	18,068

Tasks

Training:	20170804_140639
Evaluation:	20170804_140639

ここに入力して検索

16:56 2018/07/15

学習や評価用のデータは、CSVファイルやpngなどの別ファイルから入出力します。

N LeNet.sdcproj - Neural Network Console (NNabla)

EDIT TRAINING EVALUATION

Results History ACTION ▾ Elapsed: 00:00:00:11 Remaining: 00:00:00:00 Total: 00:00:00:11 0000 / 10000) Training Evaluation

20170804_140639 PARETO OPTIMAL Output Result Confusion Matrix: y - y' ▾ ▶ ▶ □ □ ▶ □

Index	x:image	y:label	y'_0
1	K:\neural_network_console_170\	7	2.92585264106e-05
2	K:\neural_network_console_170\	2	1.71542794192e-05
3	K:\neural_network_console_170\	1	1.41157315738e-05
4	K:\neural_network_console_170\	0	0.99989211559e-05

Training Validation Best Validation Multiply Add 0.007822 0.050883 0.032139 @ epoch 3 802,876 EVALUATED

Overview

Statistics

Output	21,919
CostParameter	78,810
CostAdd	11,304
CostMultiply	0
CostMultiplyAdd	802,876
CostDivision	490
CostExp	490
CostIf	18,068

Tasks

Training: 20170804_140639

Evaluation: 20170804_140639

2017-08-04 14:07:44,027 Evaluation process is started.
python "K:\neural_network_console_170725\libs\nnabla\python\src\nnabla\utils\cli\cli.py" forward
-c "K:\neural_network_console_170725\samples\sample_project\image_recognition\MNIST\LeNet.files"
-p "K:\neural_network_console_170725\samples\sample_project\image_recognition\MNIST\LeNet.files"
-d "K:\neural_network_console_170725\samples\sample_dataset\mnist\mnist_test.csv"
-o "K:\neural_network_console_170725\samples\sample_project\image_recognition\MNIST\LeNet.files"
2017-08-04 14:07:46,773 [nnabla]: data 64 / 10000
2017-08-04 14:07:46,925 [nnabla]: data 128 / 10000

ここに入力して検索

16:57 2018/07/15

SONY Neural Network Console の実装方法

「ACTION」をプルダウン⇒「EXPORT」⇒「Python Code (Nnabla)」

学習済ニューラルネットワークを利用する方法は3通り

- Neural Network Libraries Pythonコードからの実行 **おすすめ**

1. Neural Network Console上で推論に用いるネットワークを右クリックして、Export、Python Code (NNabla) を選択
2. 学習結果のparameters.h5を、load_parametersコマンドで読み込み

```
import nnabla as nn
nn.load_parameters('./parameters.h5')
```

3. 2によりパラメータが読み込まれた状態で、1でExportされたネットワークを実行 (forward)

- Neural Network LibrariesのCLI (Python利用) からの実行 **簡単**

```
python "(path of Neural Network Console)/libs/nnabla/python/src/nnabla/utils/cli/cli.py" forward
    -c Network definition file included in the training result folder (net.nntxt)
    -p Parameter file included in the training result folder (parameters.h5)
    -d Dataset CSV file of input data
    -o Inference result output folder
```

- Neural Network Libraries C++からの実行 **コンパクトに製品搭載する際に**

- https://github.com/sony/nnabla/tree/master/examples/cpp/mnist_runtime