

Anaconda 以及 jupyter notebook常用命令

安装 **Anaconda** 后，常使用jupyter notebook进行一些代码的编写。整理了一些在Anaconda prompt和jupyter notebook中自己比较常用的命令，希望能有所帮助。

Anaconda prompt界面命令

创建虚拟环境以及虚拟环境激活

个人使用Anaconda最主要的原因就是可以更方便的创建一个虚拟环境，在一个单独的环境中设置需要的python版本、需要的包，后续程序编写时也能共享、不需要重新一个一个添加导入。

最初虚拟环境的添加使用的是Anaconda Navigator，也就是图形化的界面。后续发现这个界面加载太慢，更喜欢使用Anaconda prompt，直接在命令提示符中执行需要的操作。

打开Anaconda prompt后：

创建新的虚拟环境对应的命令：

```
1 | conda create -n [env-name] python=[版本号]
```

其中，env-name创建的虚拟环境的名字，版本号用于指明对应的Python版本。

如想要创建一个opencv对应的虚拟环境，希望的Python版本为3.7，环境名字为cv2，则对应的命令为：

```
1 | conda create -n cv2 python=3.7
```

创建虚拟环境后，使用activate命令进行激活：

```
1 | activate [环境名]
```

比如要激活上面我创建的环境，只需要键入命令：

```
1 | activate cv2
```

包的安装

创建一个环境后，就可以安装需要的包了。个人推荐在Anaconda官网，anaconda.org的网站上搜索对应命令。

下面用 **scrapy** 作为例子，具体操作如下：

1. 网站搜索栏搜索你需要的包，在搜索栏输入对应包的名字（这里为scrapy）。



Where packages, notebooks, projects and environments are shared.

SEARCH PACKAGES

 scrapy

Join Today

[Sign Up](#)

[Sign In](#)

Username

Pick a username

Email Address

Your email

Enter Password

SecretPassword

Re-enter Password

SecretPassword

☐ I accept the [Terms & Conditions](#)



进行人机身份验证




reCAPTCHA


[隐私权](#) - [使用条款](#)

[Register For Free](#)


2. 页面中找到需要的包（我一般选择第一个下载量最高的），点击包的名字。比如这里点击conda-forge/scrapy中的scrapy。


 [Gallery](#) [About](#) [Anaconda](#) [Help](#) [Download Anaconda](#)


You must login to search private packages








Search Anaconda.org ... 

Filters

Type: All 

Access: All 

Platform: All 

 Favorites	 Downloads	 Package (owner / package)		Platform
29	494746	 conda-forge / scrapy 2.5.1 A high-level Python Screen Scraping framework	conda	linux osx- win- win-
6	21911	 anaconda / scrapy 2.4.1 A high-level Python Screen Scraping framework	conda	linux linux linux-py osx- win- win-
3	15324	 Scrapinghub / scrapy 1.2.0 A high-level Web Crawling and Web Scraping framework	conda	linux linux osx- win- win-
0	1774	 main / scrapy 2.4.1 A high-level Python Screen Scraping framework	conda	linux linux linux-py osx- win- win-

3. 复制任意命令，粘贴到anaconda prompt中激活的虚拟环境里，直接运行即可。中途会确认一下是否安装以及更新，输入y确认即可。

Documentation: <https://docs.scrapy.org>
494746 total downloads
Last upload: 3 months and 12 days ago

Installers

Info: This package contains files in non-standard labels.

conda install ?

linux-64	v2.5.1
win-32	v1.5.0
osx-64	v2.5.1
win-64	v2.5.1

To install this package with conda run one of the following:

```
conda install -c conda-forge scrapy  
conda install -c conda-forge/label/cf201901 scrapy  
conda install -c conda-forge/label/cf202003 scrapy
```

Description

Scrapy is an open source and collaborative framework for extracting the data you need from websites in a fast, simple, yet extensible way.

CSD

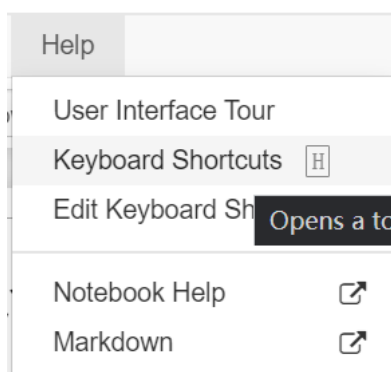
jupyter notebook常用命令

jupyter notebook的激活

激活对应虚拟环境后，在界面输入 `jupyter notebook` 即可激活并运行jupyter notebook。

```
1 | jupyter notebook
```

进入jupyter notebook界面后，可以直接在菜单 `Help -> Keyboard Shortcuts` 中查找到对应的所有键盘快捷命令。



不过我感觉常用的并不多。我比较常用的命令有如下这些：

1 切换输入模式

按 `ESC` 进入命令行模式，对应jupyter notebook中选中的框为蓝色。

比如下面就是命令行模式下，选中的框。

1 模型选择、欠拟合和过拟合

通过多项式拟合来交互探索这些概念

```
In [11]: import math
import numpy as np
import torch
from torch import nn
from d2l import torch as d2l
```

做一个简单的人工数据集

使用三阶多项式生成训练和测试数据的标签，加上了一些噪音

$$y = 5 + 1.2x - 3.4 \frac{x^2}{2!} + 5.6 \frac{x^3}{3!} + \epsilon \text{ where } \epsilon \sim \mathcal{N}(0, 0.1^2).$$

CSDN

按 **Enter** 则会进入输入模式，可以对选中的框进行文本或代码的输入。对应模式下选中的框为绿色。

1 模型选择、欠拟合和过拟合

通过多项式拟合来交互探索这些概念

```
In [11]: import math
import numpy as np
import torch
from torch import nn
from d2l import torch as d2l
```

CSDN

2 运行选中的代码块

在输入模式或者命令行模式下，按下 **Enter + Shift** 即可运行选中的单元格，并选择下面的单元格。

3 单元格的添加或删除

命令行模式下：

- 按下字母 **A** 可以在当前选中的单元格上方插入新的单元格
- 按下字母 **B** 可以在当前选中的单元格下方插入新的单元格。
- 按下字母 **D** 两次可以删除当前选中的单元格。

4 单元格变为 markdown

按下 **Enter + Shift** 后生成的单元格默认是代码块的形式，如果要进行markdown的输入则需要将对应单元格从代码块格式切换为markdown。对应步骤为，按下 **ESC** 进入命令行模式后，按下 **M** 将对应格式转化为markdown，而后继续markdown的输入。

5 编写代码常用的快捷键执行逻辑

我通常的编写逻辑为：

- 按下 **Esc**，而后按下 **M** 进入命令行模式，按下 **Enter** 进行markdown格式的文本输入，相当于一个比较大段的注释。
- 按下 **Enter + Shift** 运行这个单元格，而后输入对应的Python代码。
- 按下 **Enter + Shift** 运行完成的代码块，查看输出。如果不符合预期结果就按下 **Enter** 选中对应单元格进行修改。
- 而后新生成的单元格默认为代码块的形式。如果需要输入markdown则按下 **ESC** 进入命令行模式，而后按下 **M** 转化单元格格式，而后正常进行m输入。
- 如果涉及到单元格的插入、修改或者删除，则按下 **ESC** 进入命令行模式后按下对应的键即可。

写在最后

整体上只代表我个人常用的命令，比较小白，也是目前的经验。后续会随着自己的学习不断更新。如果有问题，请多多指教！

