

ANN实验环境配置

1. 实验环境/设备

2. 实验环境配置

2.1. 安装Python（可跳过）

2.2. 安装Anaconda

2.2.1. Conda下载与安装

2.2.2. Conda环境变量安装

2.2.2.1. Windows系统

2.2.2.2. Linux/Unix系统

2.2.3. Anconda使用

2.2.3.1. Anaconda GUI

2.2.3.2. 命令行

2.2.4. GPU版本的Pytorch安装

3. 安装验证

4. FAQ

语雀链接：<https://www.yuque.com/u953085/kg22kp/psy5om8g3ut9awsq>

如果遇到困难，可以**联系助教**

1. 实验环境/设备

- Windows/MacOS/Ubuntu操作系统电脑一台，以带独立Nvidia显卡为优，无显卡也可以运行。（可能提供计算服务器）。
- Python语言
- Anaconda: Python的扩展包管理器和环境管理器
- Pytorch： 基于Python的深度学习编程框架。
- Vscode/Pycharm等Python的开发环境。

2. 实验环境配置

2.1. 安装Python（可跳过）

参照此[Python安装文档](#)完成Python环境的安装。

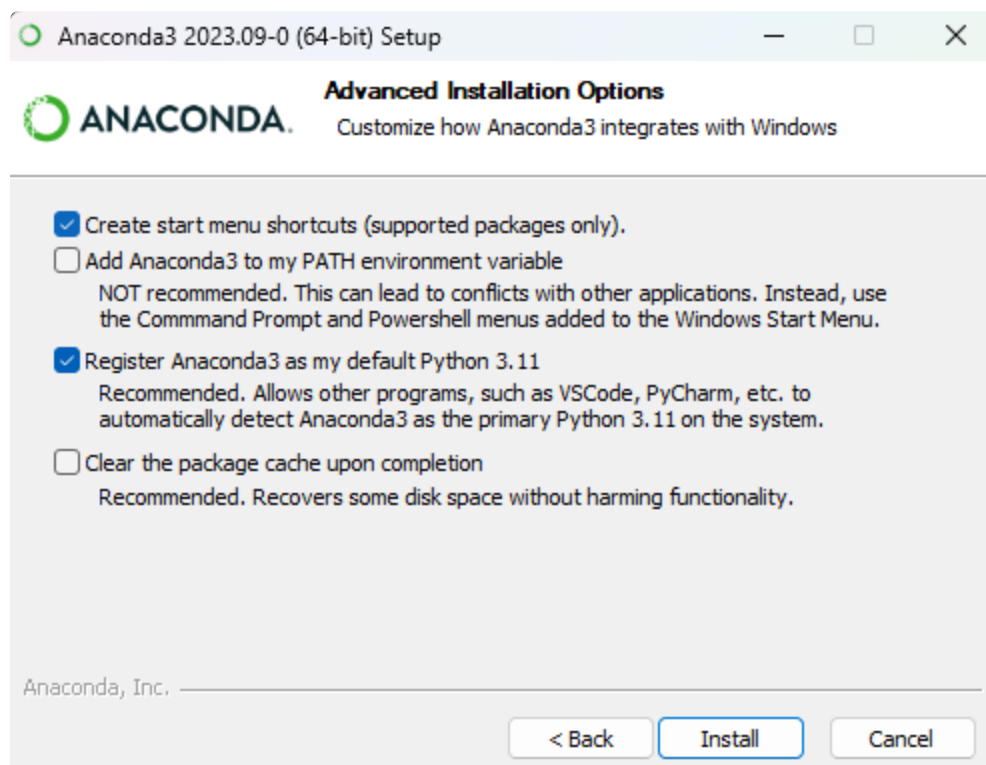
- 此部分可以跳过，直接使用Anaconda中的Python环境

2.2. 安装Anaconda

2.2.1. Conda下载与安装

首先在[Anaconda官网](#)或者[清华源镜像](#)中下载对应系统的Anaconda。下载完成后，运行下载好的Anaconda安装程序。基本是一路点Next。

在这里可以把“Add Anaconda3 to my PATH environment variable”勾选上，也可以之后参照[Conda环境变量](#)一节配置PATH。

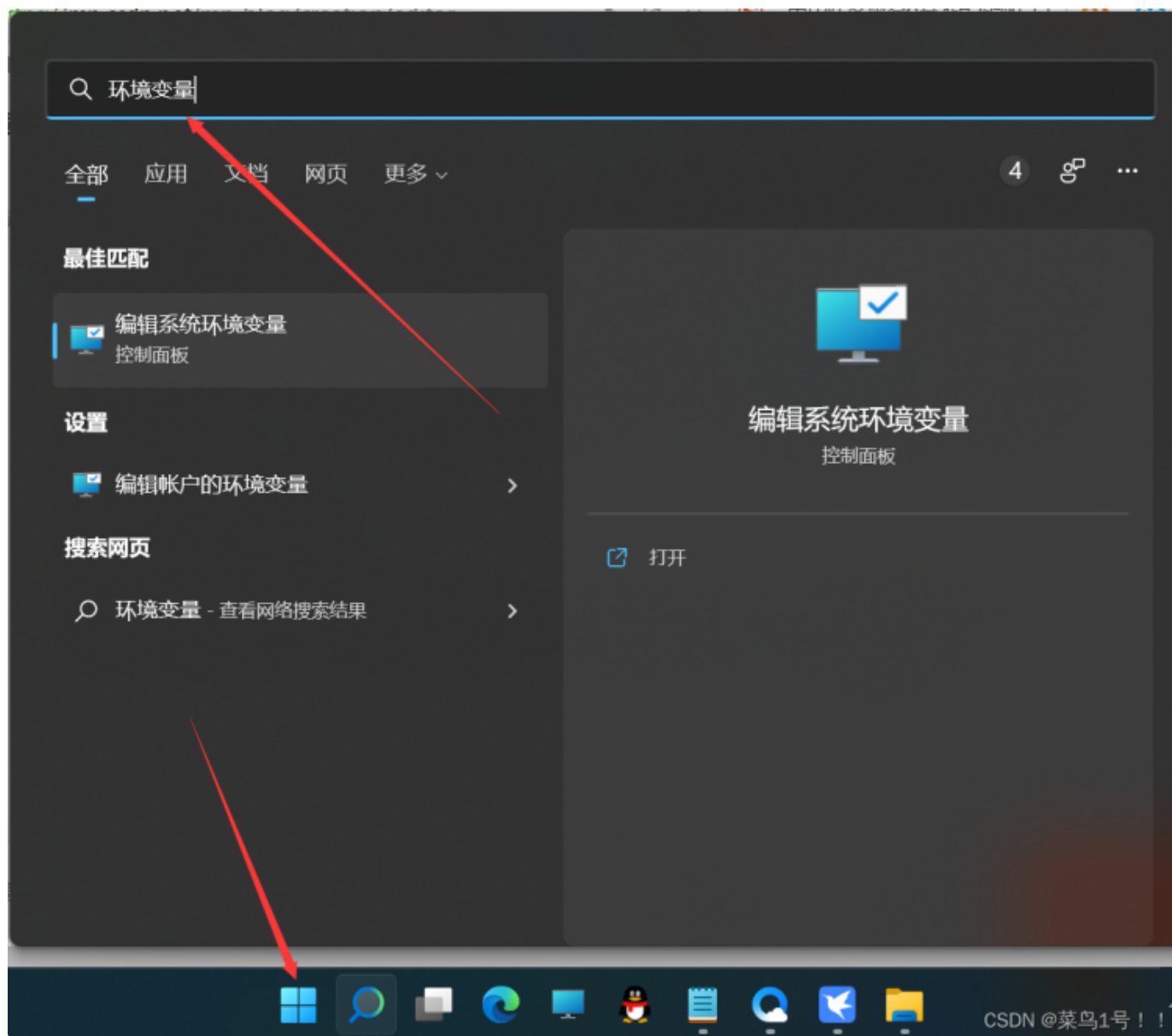


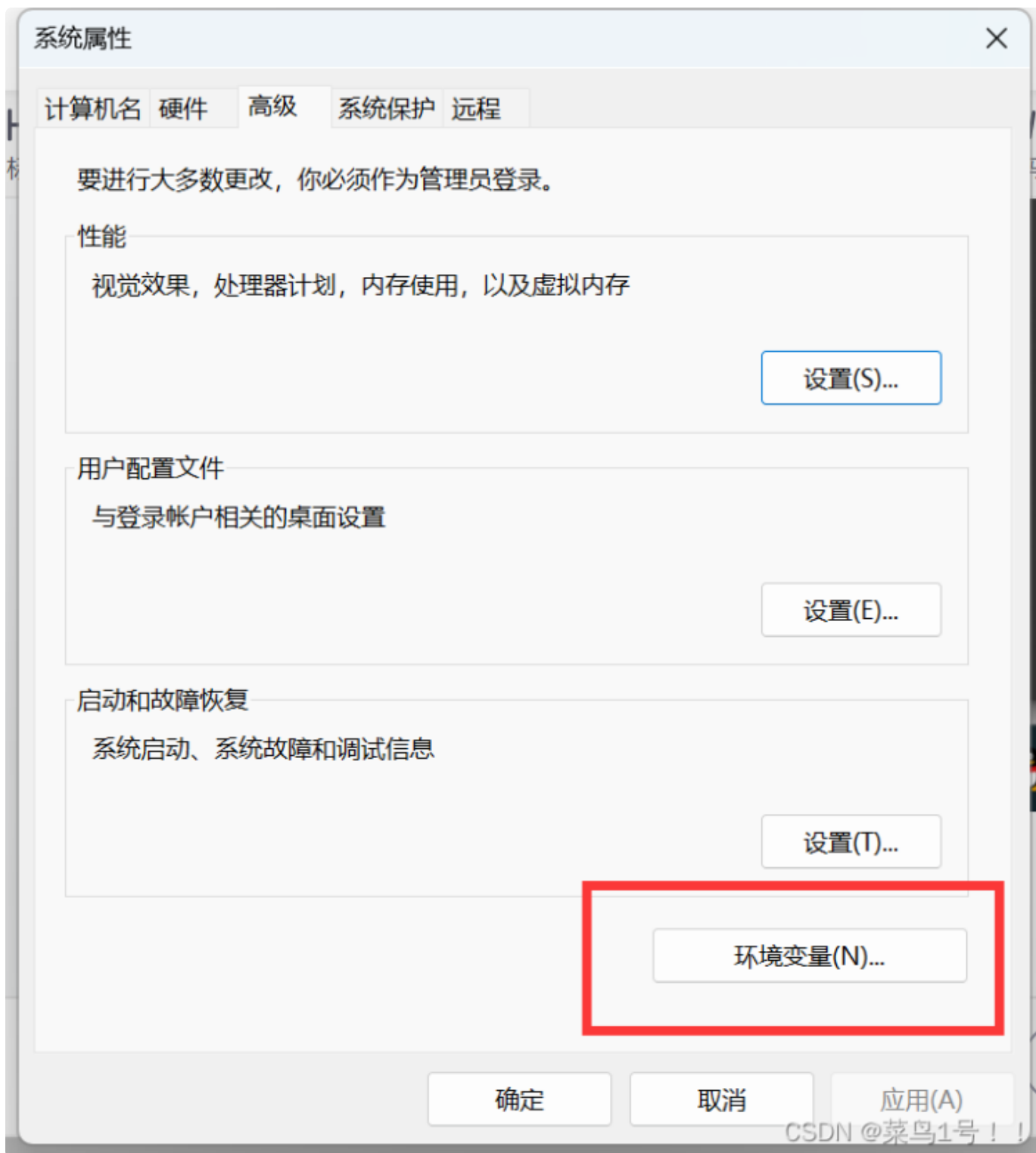
2.2.2. Conda环境变量安装

如果使用Anaconda GUI则不需要配置环境路径

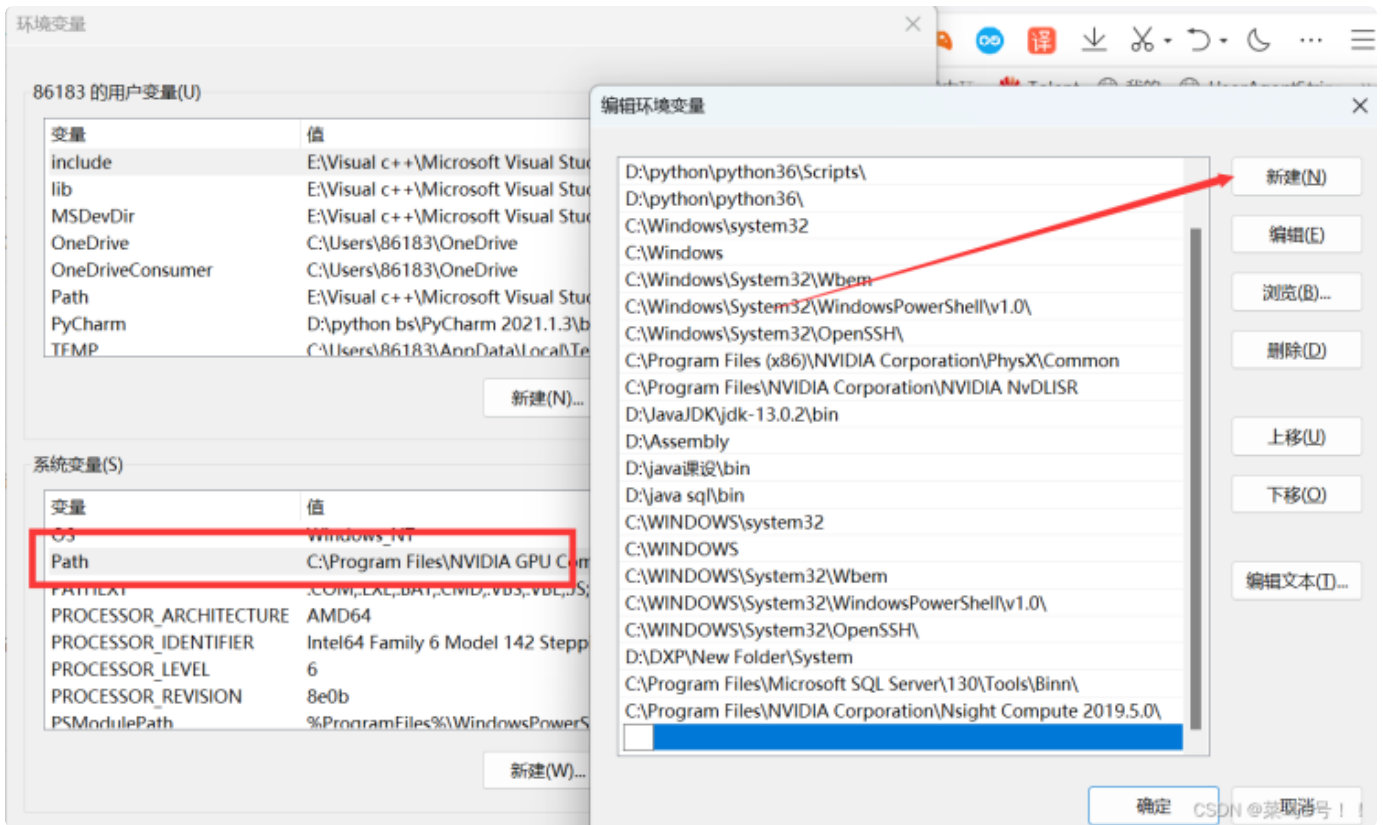
2.2.2.1. Windows系统

对于Windows电脑，按win键，搜索“环境变量”

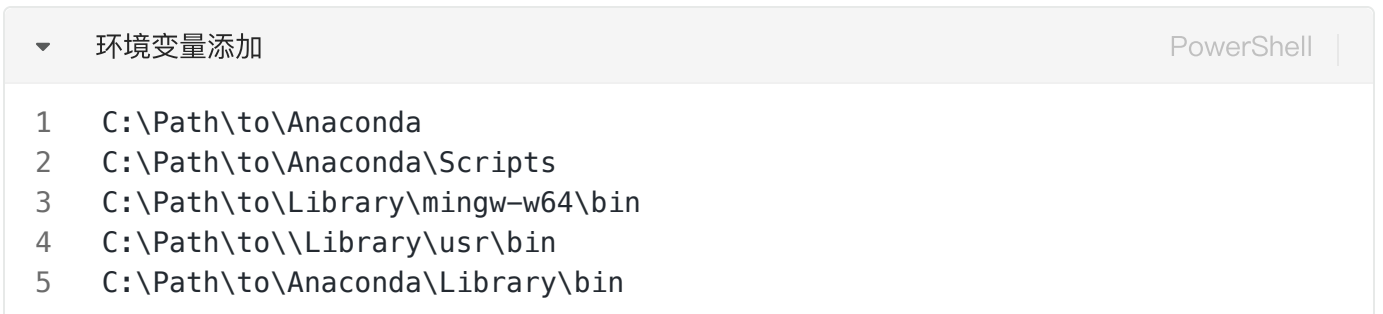




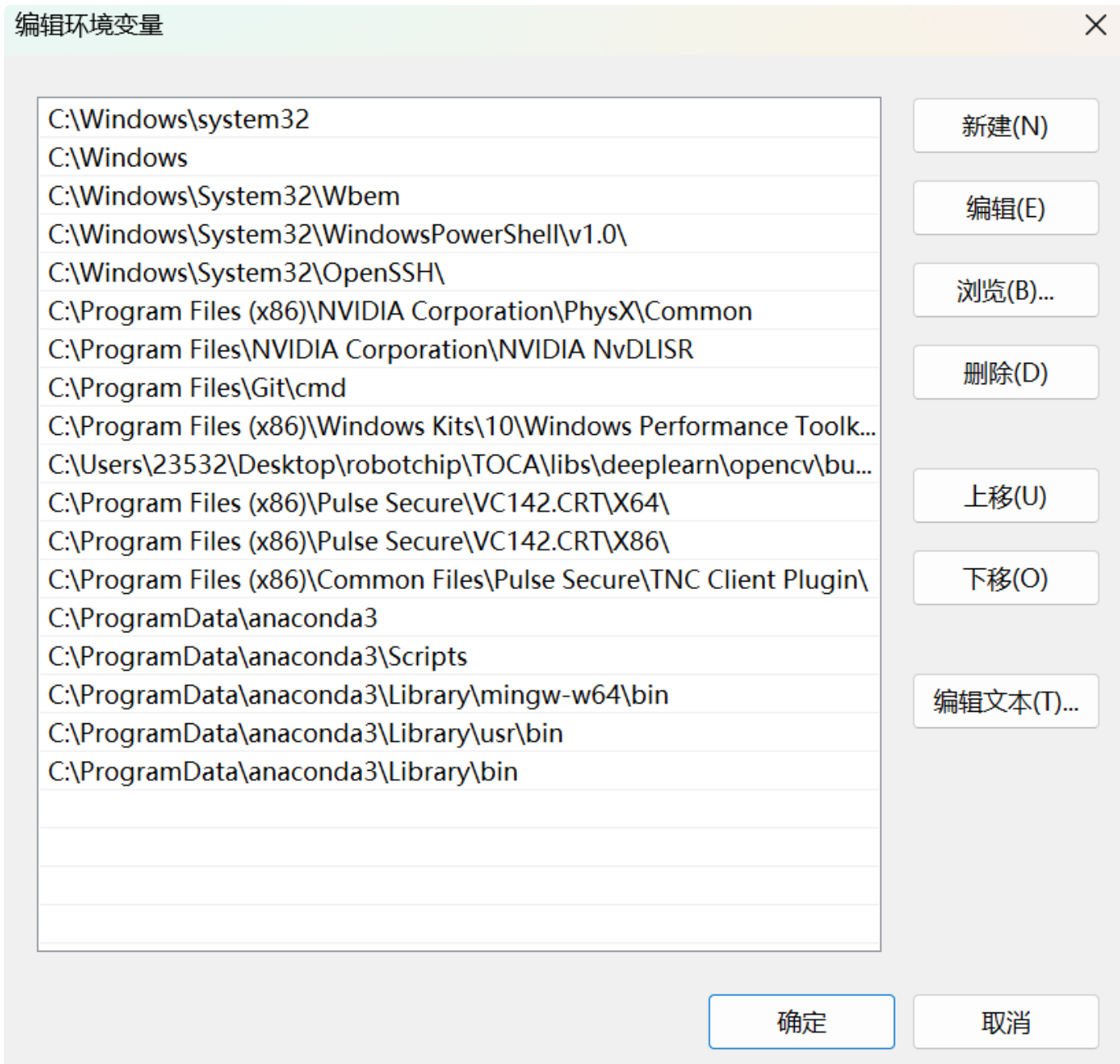
打开环境变量后，双击Path，点击新建



把以下路径添加到Path中（实际路径请根据自己安装的Anaconda路径调整）



安装结果：



打开命令行，输入`conda --version`，可查看Conda是否安装成功以及其版本

```
▼ Conda版本检测 PowerShell |
1  $ conda --version
2  conda 23.7.4
```

当然也可以使用`conda info`命令查询。

2.2.2.2. Linux/Unix系统

只需要将Conda的环境路径添加到PATH即可，在bash中输入：

```

1 # linux系统
2 export PATH = /path/to/anaconda3/bin:$PATH >> ~/.bashrc
3 source ~/.bashrc
4
5 # MacOS
6 export PATH = /path/to/anaconda3/bin:$PATH >> ~/.zshrc
7 source ~/.zshrc

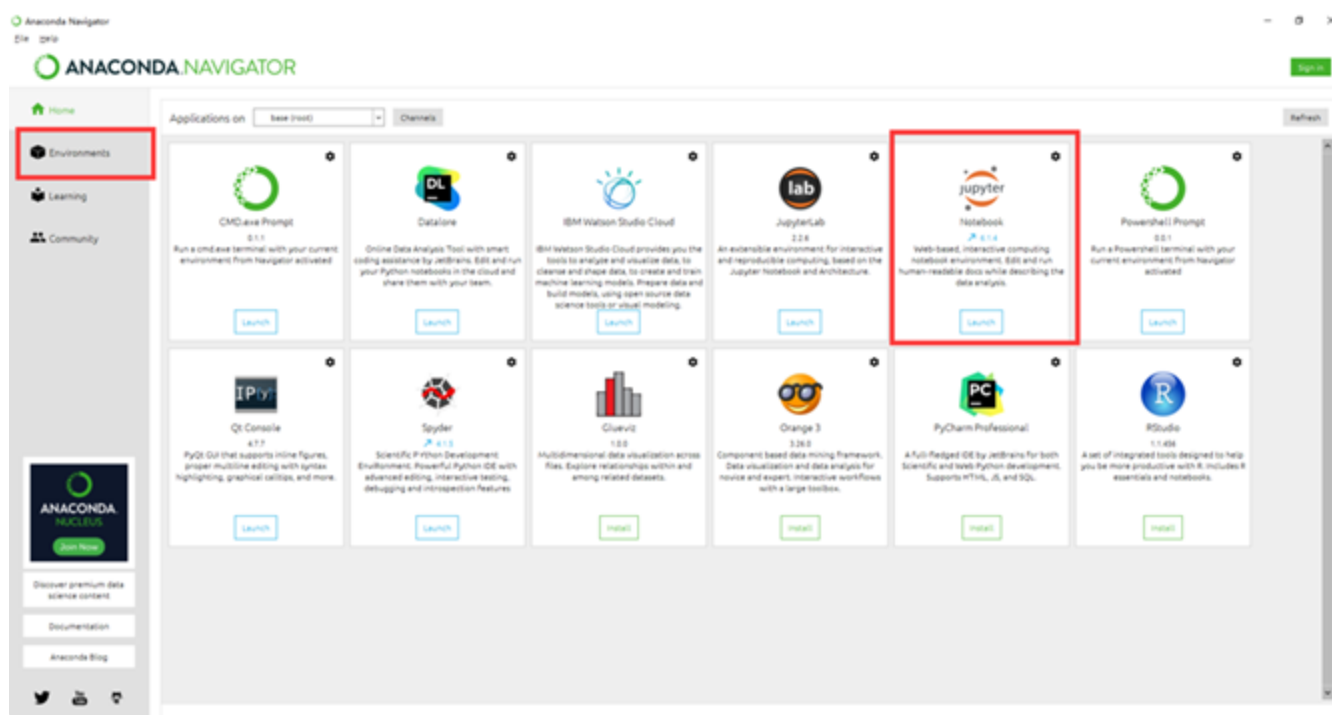
```

2.2.3. Anaconda使用

使用以下两个方法任意一个都可以

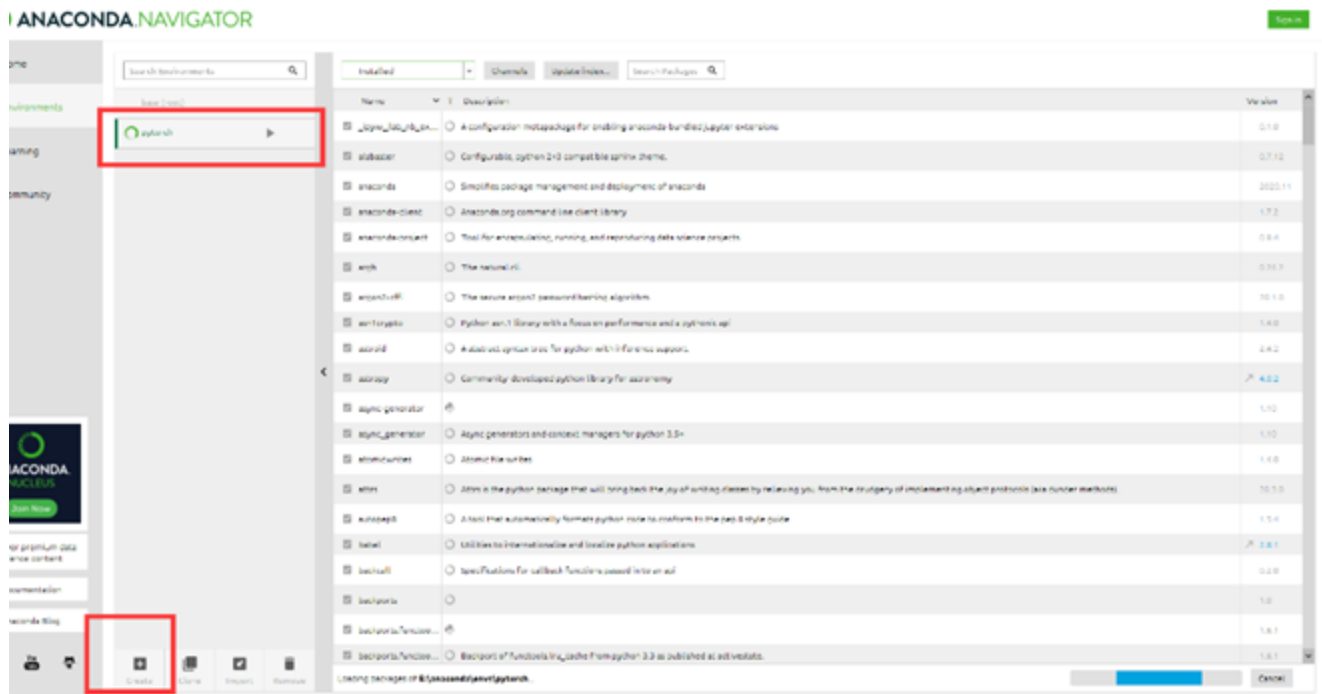
2.2.3.1. Anaconda GUI

安装后可以在应用程序（mac） / 开始（windows）中找到Anaconda的GUI启动图标：Anaconda Navigator。开启后界面如下图所示，其中红框的部分为我们需要用到的。



这里如果Jupyter NoteBook 没有被安装，请自己安装一下。

在上图左侧点击Environment，进入环境配置，如下：

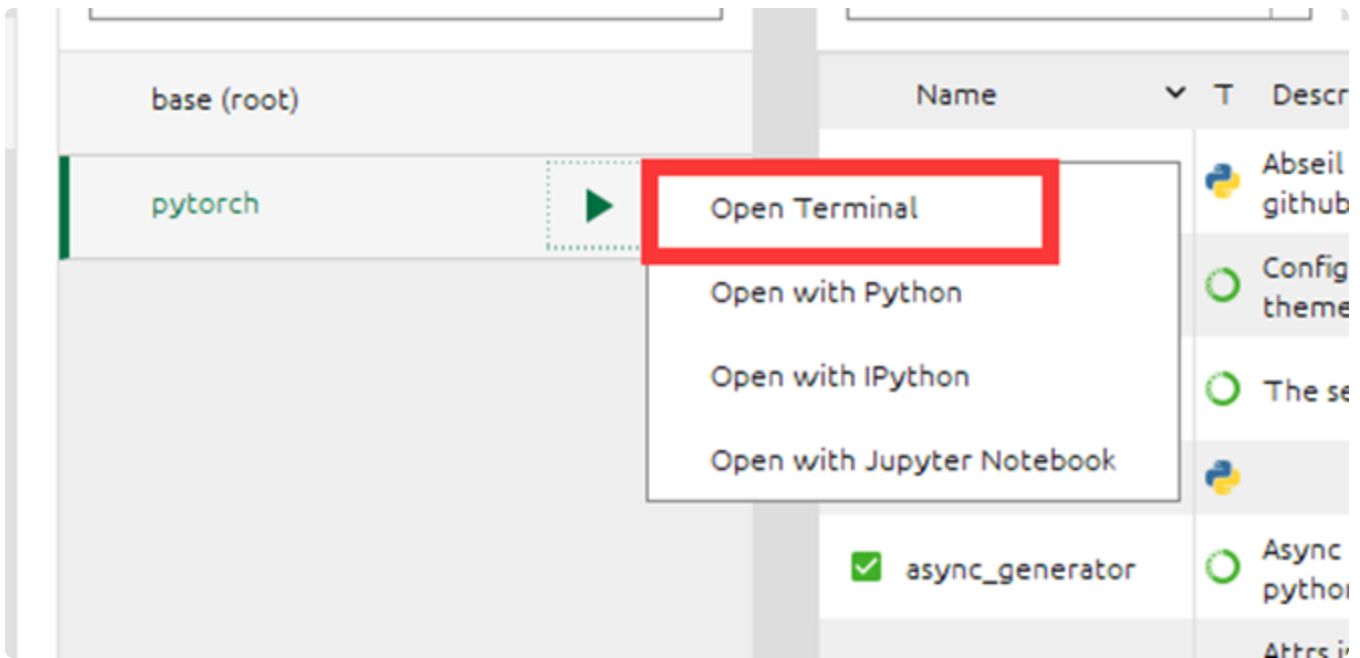


其中每一个环境对应一个独立的Python的虚拟环境。由于Python的包常常会有很多个版本，某些版本的包和另一个包的特定版本会产生不兼容，但我们有时候又常需要升级一些包或者安装新的包，这样会导致环境管理的困难。因此Anaconda提供了一个解决思路：对一个任务建立一个虚拟的环境，和别的环境独立，相当于一个独立沙盒，在这个沙盒中配置对于这个任务所需要的各种包的特定版本。我们在这里就可以新建一个名为Pytorch的环境，然后安装所需的Pytorch版本。

如果要使用GPU版本的Pytorch，请参照[GPU版本的Pytorch安装](#)进行安装。（如果选择作业二的话，建议安装GPU版本的Pytorch）

PyTorch Build	Stable (1.9.0)		Preview (Nightly)		LTS (1.8.2)	
Your OS	Linux		Mac		Windows	
Package	Conda	Pip		LibTorch		Source
Language	Python			C++ / Java		
Compute Platform	CUDA 10.2	CUDA 11.1		ROCm 4.2 (beta)		CPU
Run this Command:	conda install pytorch torchvision torchaudio cpuonly -c pytorch					

但是Pytorch不是一个简单的包，它往往需要很多复杂的依赖，例如Numpy包，一个个手动安装比较麻烦，因此我们可以对新建好的环境使用命令行一键安装所有的包，Pytorch官网提供了安装命令如上图所示。从我们刚刚新建好的环境中打开Terminal（终端）



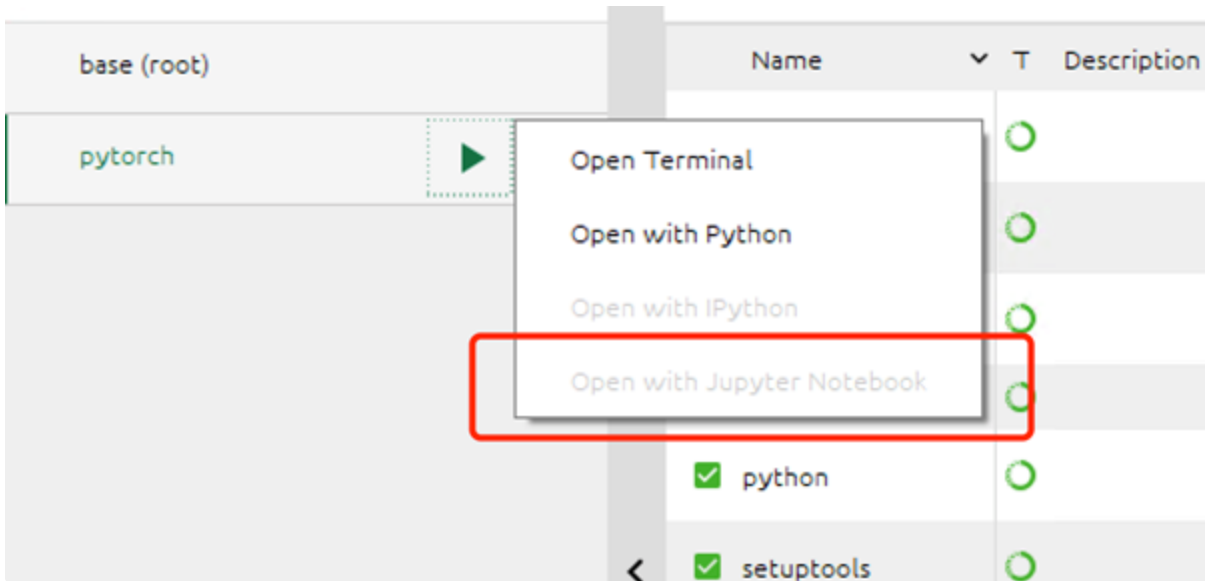
这里我们用Windows演示，MacOS和Ubuntu可以直接在终端中配置conda，安装好Anaconda后具体命令为：

```
Bash |
1  conda create -n Pytorch # 创建一个名字为pytorch的虚拟环境
2  conda activate Pytorch # 激活Pytorch
3  conda install Pytorch torchvision ... # (这条命令需要看Pytorch官网对你的安装需求给出的具体参数是什么样的)。
4  pip install matplotlib # 注意: matplotlib不建议使用conda去安装, 在jupyter notebook下使用会因不明原因导致内核崩溃
```

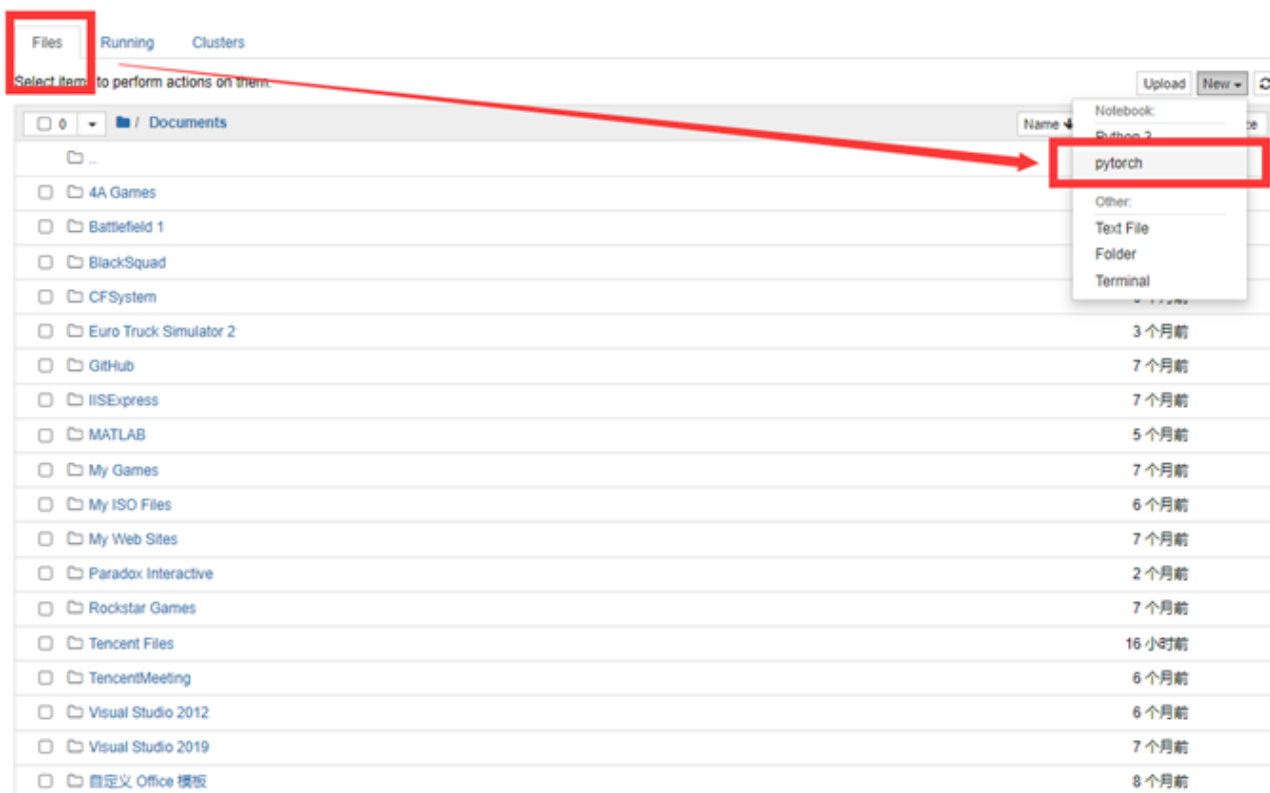
如果发现安装太慢，可以搜索“anaconda改清华源”，按清华tuna给的照流程操作，安装结束以后，在终端输入：

```
Bash |
1  python
2  import torch
```

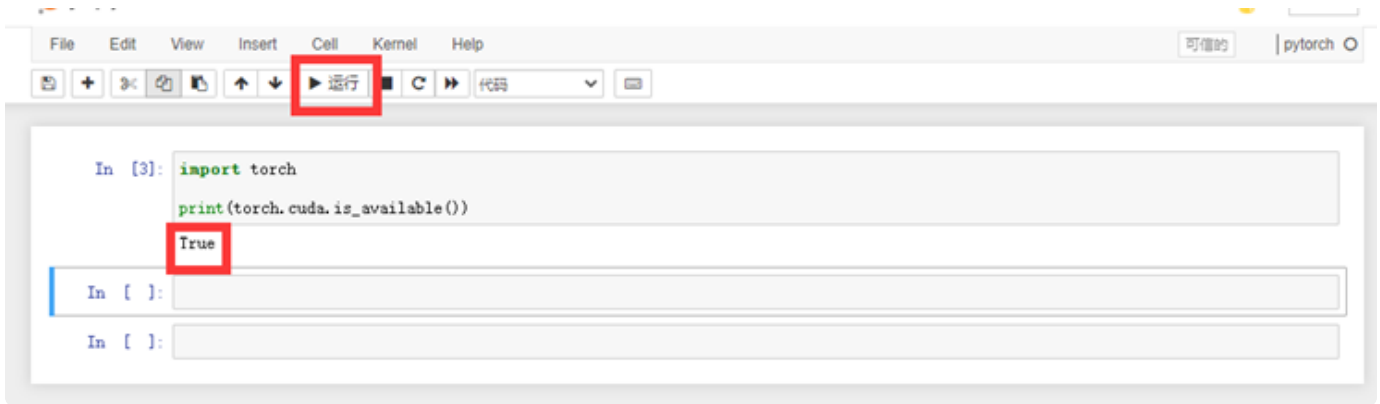
如果没有报错，即安装完毕，如果报错，请检查是否激活环境/成功安装



安装好Pytorch以后，我们点击上图中的 **Open with Jupyter Notebook**（首页安装后会变成黑色），打开Python记事本。Jupyter Notebook是一个带有文件管理、互动脚本和命令终端三大主要功能的python轻量级开发软件，非常适合教学演示和开发、学习。这里我们主要使用的是其互动脚本，如下图所示：



点击环境名称即可在当前目录下新建一个Python互动脚本文件，如下图，编写一个简单的Python脚本，点击运行，即可输出对应的结果。下图中是安装了带有CUDA的Pytorch，如果是不带CUDA的，即会输出**False**，是正常结果。



执行到这一步基本完成了环境配备，接下来的工作都将在这个环境中完成。

2.2.3.2. 命令行

由于Python的包常常会有很多个版本，某些版本的包和另一个包的特定版本会产生不兼容，但我们有时候又常需要升级一些包或者安装新的包，这样会导致环境管理的困难。因此Anaconda提供了一个解决思路：对一个任务建立一个虚拟的环境，和别的环境独立，相当于一个独立沙盒，在这个沙盒中配置对于这个任务所需要的各种包的特定版本。我们在这里就可以新建一个名为Pytorch的环境，然后安装所需的Pytorch版本。

登录Pytorch官网，查看对应的安装命令和版本

PyTorch Build	Stable (1.9.0)		Preview (Nightly)		LTS (1.8.2)	
Your OS	Linux		Mac		Windows	
Package	Conda	Pip		LibTorch		Source
Language	Python			C++ / Java		
Compute Platform	CUDA 10.2	CUDA 11.1		ROCm 4.2 (beta)		CPU
Run this Command:	conda install pytorch torchvision torchaudio cpuonly -c pytorch					

如果要使用GPU版本的Pytorch，请参照[GPU版本的Pytorch安装](#)进行安装。（如果选择作业二的话，建议安装GPU版本的Pytorch）

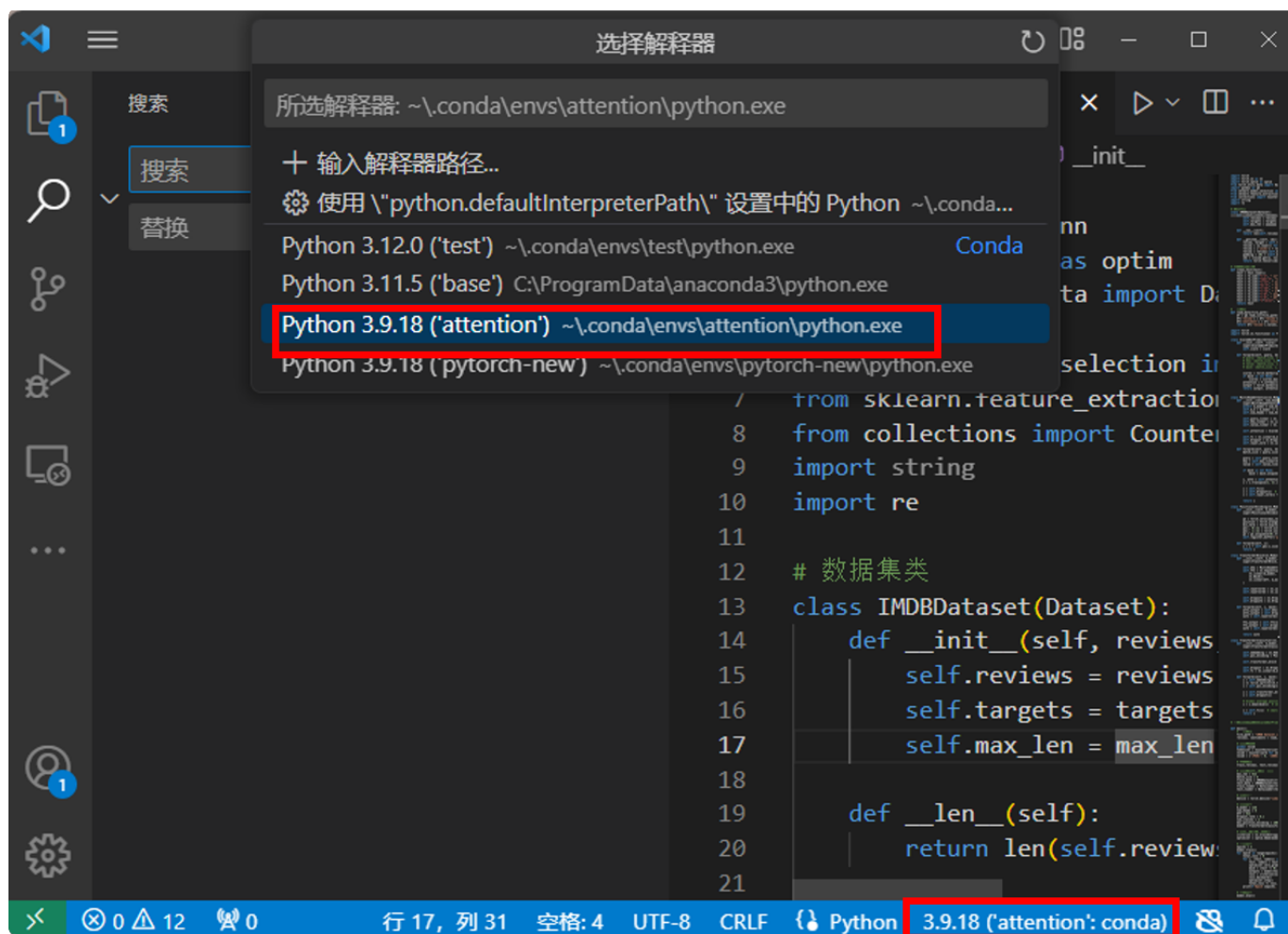
打开命令行

```

1 conda create -n Pytorch # 创建一个名字为pytorch的虚拟环境
2 conda activate Pytorch # 激活Pytorch
3 conda install Pytorch torchvision ... # (这条命令需要看Pytorch官网对你的安装需求给出的具体参数是什么样的)。
4 pip install matplotlib # 注意: matplotlib不建议使用conda去安装, 在jupyter notebook下使用会因不明原因导致内核崩溃

```

打开vscode（需要安装python，jupyter notebook插件），点击右下角的python环境，并选择之前安装的虚拟环境即可。



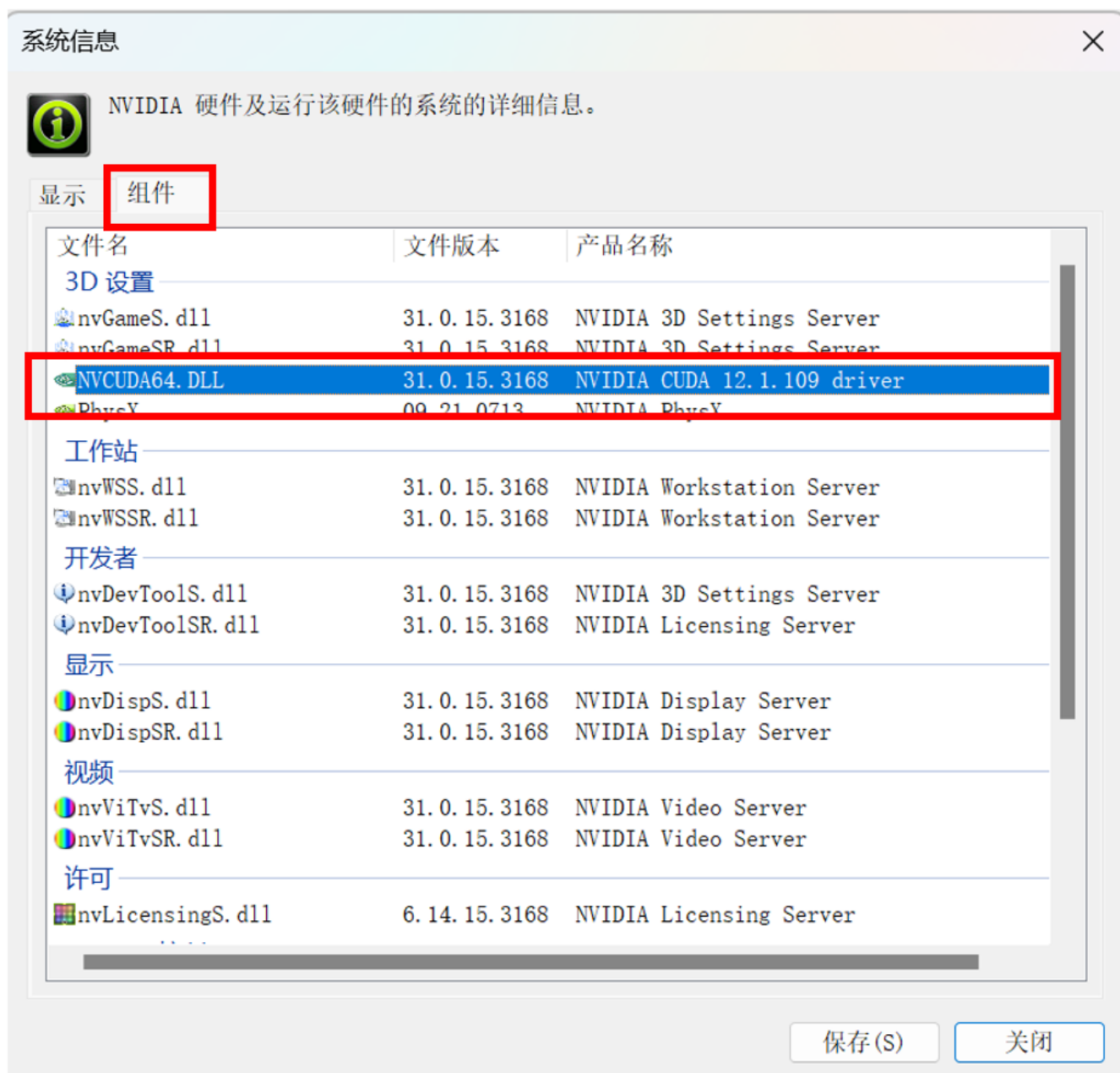
2.2.4. GPU版本的Pytorch安装

如果选择作业二，且笔记本上有NVIDIA显卡的话，可以尝试安装GPU版本的Pytorch。接下来以Windows系统为例，示范如何安装GPU版本的Pytorch。（如果尝试失败，可能会提供服务器）

首先，打开NVIDIA控制面板，点击左下角的系统信息



点击组件，查看CUDA的版本信息，在这里，我的CUDA版本是12.1

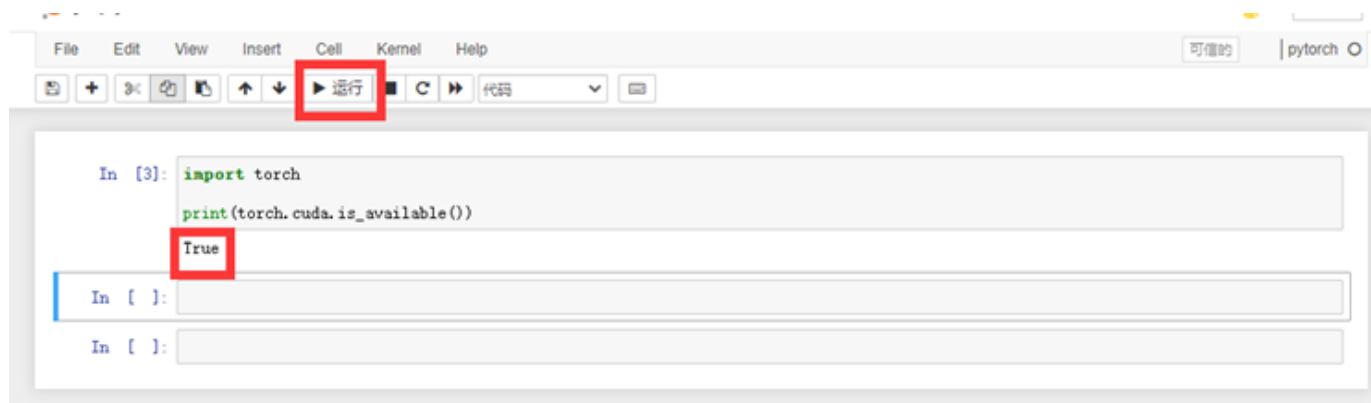


到pytorch上搜索对应CUDA版本的安装命令：

PyTorch Build	Stable (2.1.0)		Preview (Nightly)	
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python		C++ / Java	
Compute Platform	CUDA 11.8	CUDA 12.1	ROCm 5.6	CPU
Run this Command:	<pre>pip3 install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cu121</pre>			

按照命令进行安装即可。

编写一个简单的Python脚本，点击运行，即可输出对应的结果。如果安装gpu版本的pytorch成功，则会输出True。



3. 安装验证

运行以下文件，可全部通过即可。

[neural_networks_tutorial.ipynb](#)

[XOR.ipynb](#)

4. FAQ

1. Q: 运行ipynb显示No module named "matplotlib "

A: 是在新建的Pytorch虚拟环境中的命令行中输入以下命令：

▼

Bash |

```
1  pip install matplotlib
```

2.