



昇腾全栈AI软硬件平台

--Mindspore实践

昇腾全栈 AI 软硬件平台，构筑智能世界的基石

能源、金融、交通、电信、制造、医疗等行业应用



ModelArts



HiAI Service



第三方平台

应用使能

Mind X



MindX DL



MindX Edge



ModelZoo



MindX SDK

全流程开发工具链



MindStudio

AI框架



昇思 MindSpore

TensorFlow/PyTorch 等第三方框架

异构计算架构

CANN

Atlas 系列硬件

实验介绍

CNN（卷积神经网络） 用于图像和语音识别等任务。它包含**卷积层和池化层**，通过学习图像的特征进行分类。CNN适用于处理具有空间结构的数据，并在计算机视觉领域取得重大突破。

本次实验主要专注于理解卷积神经网络(Convolutional Neural Network) 原理，基于**Mindspore框架**，利用网络实现Fashion-Mnist**图像分类**实验任务。

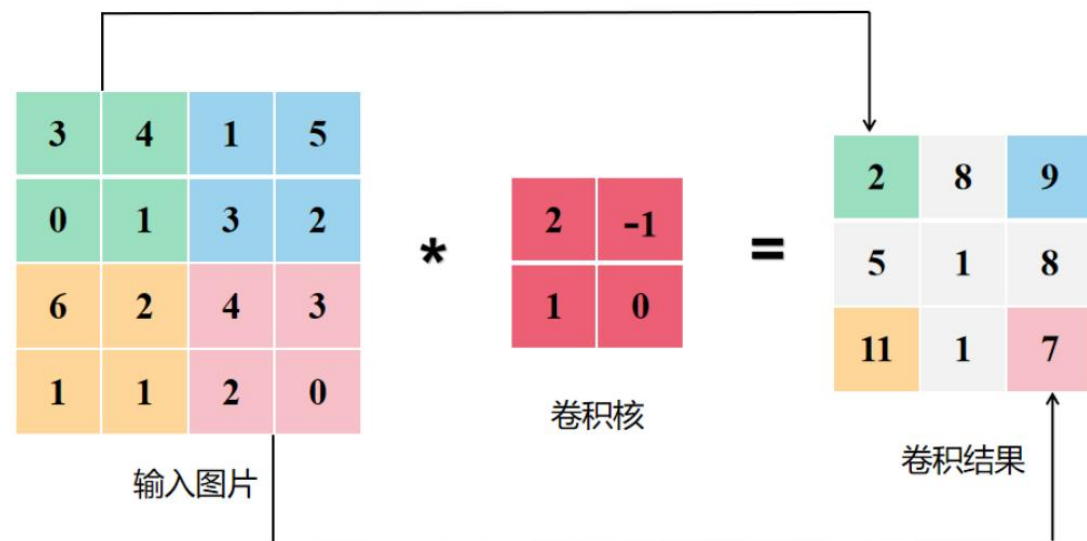
目的：

- 1.掌握如何使用MindSpore进行简单卷积神经网络的开发。
- 2.了解如何使用MindSpore进行简单图片分类任务的训练。
- 3.了解如何使用MindSpore进行简单图片分类任务的测试。

卷积的基本原理

卷积的计算过程：二维卷积操作相当于将卷积核作为一个滑动窗口，将其在输入张量上从左至右、从上至下地滑动，每滑动一下便将卷积核与其相应位置计算出一个加权系数，作为输出张量对应位置上的像素值。其计算过程为：将输入的张量与卷积核进行卷积操作，得到输出矩阵并将其返回。

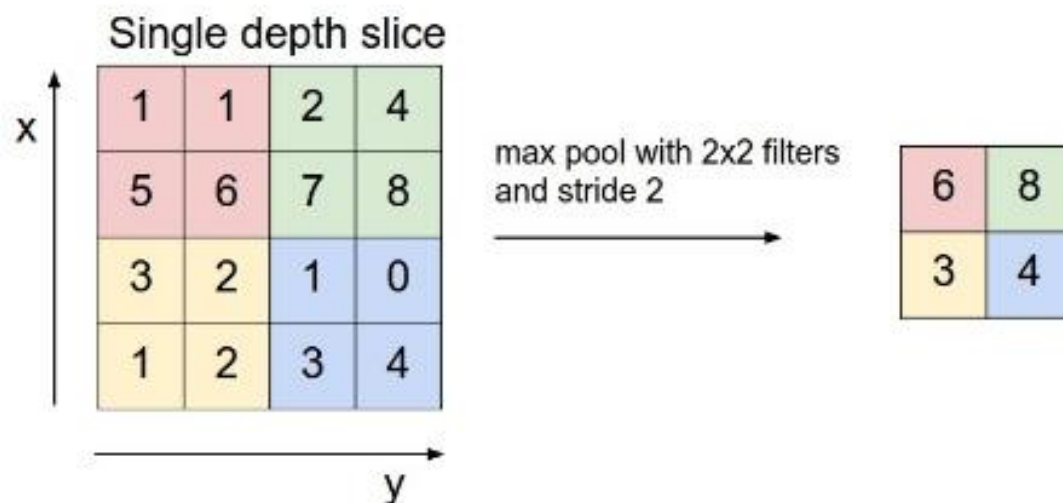
步长为1的卷积过程如下图所示：



池化的基本原理

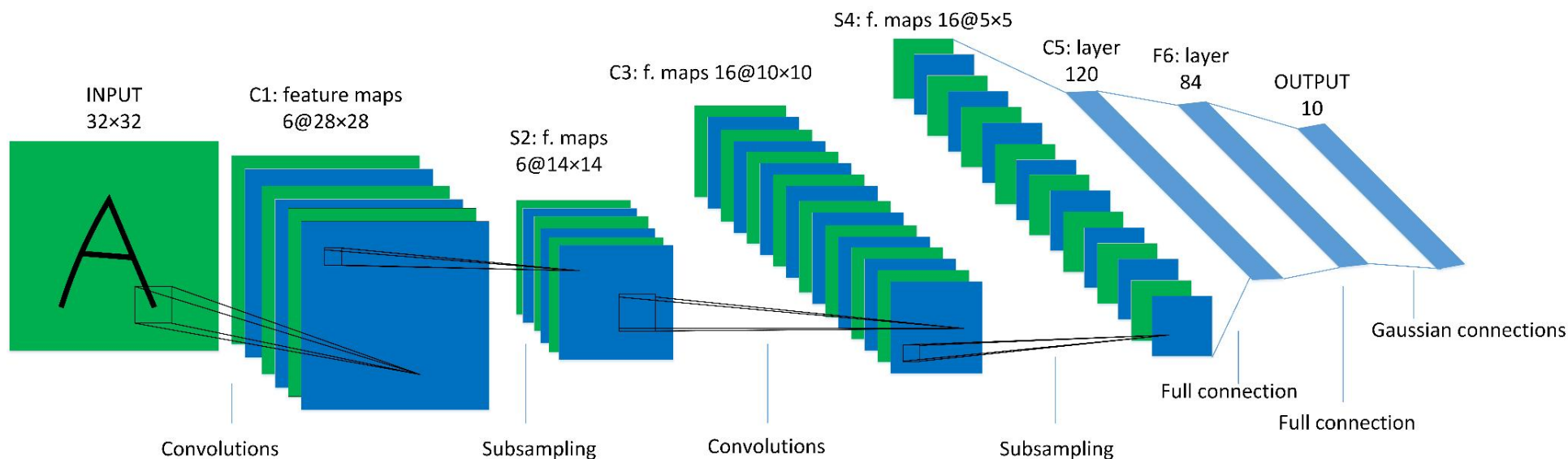
池化的计算过程：最大池化（Max Pooling）是一种常用的下采样操作，通常用于减少神经网络中的参数数量和计算复杂度。最大池化操作的过程类似于卷积操作，但是它并非使用卷积核对输入特征图进行滑动窗口运算，而是将输入特征图划分为若干大小相等的矩形块，每个矩形块内取最大值作为该块的输出，形成输出特征图。

最大池化过程如下图所示：



LeNet网络结构

本次实验我们使用简单的LeNet网络。LeNet 网络不包括输入层的情况下，共有7层: 2个卷积层、2个下采样层(池化层)、3个全连接层。每层都包含不同数量的训练参数，如下图所示:



- **输入层:**数据输入层
- **卷积层:**通过卷积进行局部特征提取
- **池化层:**通过下采样降低特征图的分辨率,降低输出对位置和形变的敏感度,保留关键特性,减少参数和计算
- **全连接层:**将局部特征通过权值矩阵组装成完整图像,完成特征空间到真实类别空间的映射,类似分类器

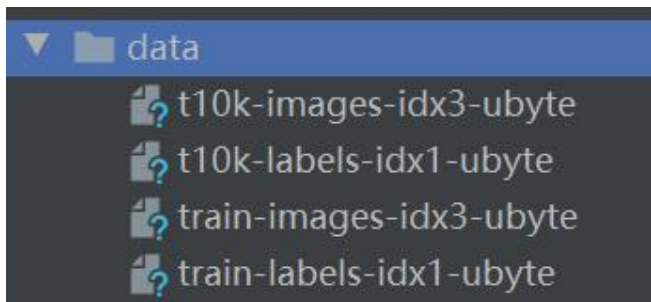
数据集介绍

Fashion-MNIST是一个替代MNIST手写数字集的图像数据集。其涵盖了来自10种类别的共7万个不同商品的正面图片。包括60000/10000的训练测试数据划分，28x28x1的灰度图片。

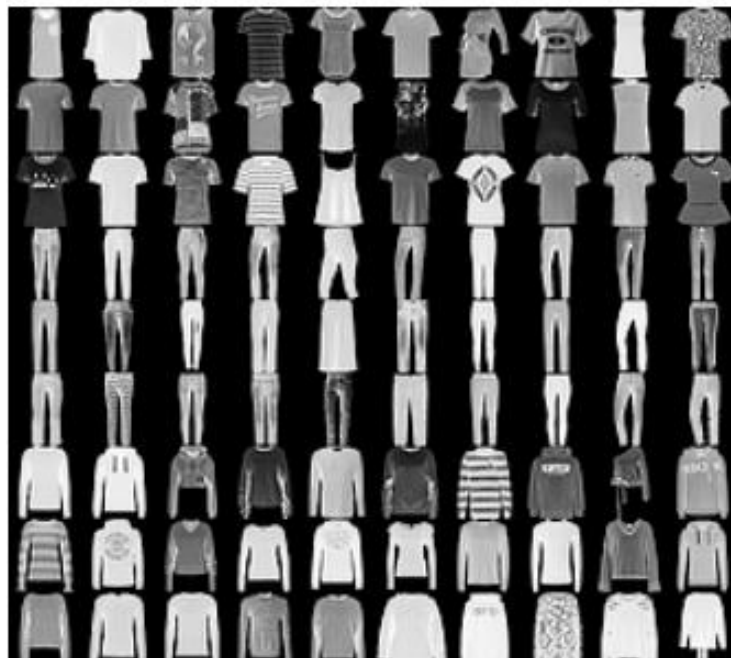
经典的MNIST数据集包含了大量的手写数字。MNIST数据集非常简单，很多深度学习算法在测试集上的准确率已经达到99.6%。

将脚本和数据集组织为如下形式：

```
data/  
├── t10k-images-idx3-ubyte  
├── t10k-labels-idx1-ubyte  
├── train-images-idx3-ubyte  
└── train-labels-idx1-ubyte
```



对数据集可视化效果如下：



实验环境



MindSpore

Miniconda/Anaconda

- Conda 是一个开源的软件包管理系统和环境管理系统，用于安装多个版本的软件包及其依赖关系，并在它们之间轻松切换。
- Conda 是为 Python 程序创建的，适用于 Linux, OSX 和 Windows, 也可以打包和分发其他软件。
- 最流行的 Python 环境管理工具。

总之，Anaconda是一个非常强大的Python发行版，它能够大大简化Python的安装和管理，提高开发者的工作效率。对于数据科学工作者来说，Anaconda更是必不可少的工具之一。

安装mindspore

- mindspore: <https://www.mindspore.cn/install>

一、获取安装命令

查看所有版本和接口变更 >

版本	2.0.0-rc1	1.10.1 ✓	Nightly		
硬件平台	Ascend 910	Ascend 310	GPU CUDA 10.1	GPU CUDA 11.1	CPU ✓
操作系统	Linux-aarch64	Linux-x86_64	Windows-x64 ✓	MacOS-aarch64	MacOS-x86_64
编程语言	Python 3.7	Python 3.8	Python 3.9 ✓		
安装方式	Pip ✓	Conda	Source	Docker	Binary
安装命令	<pre>pip install https://ms-release.obs.cn-north-4.myhuaweicloud.com/1.10.1/MindSpore/cpu/x86_64/mindspore-1.10.1-cp39-cp39-win_amd64.whl --trusted-host ms-release.obs.cn-north-4.myhuaweicloud.com -i https://pypi.tuna.tsinghua.edu.cn/simple</pre> <p># 注意参考下方安装指南，添加运行所需的环境变量配置</p>				

安装mindspore

- 安装命令: `pip install https://ms-release.obs.cn-north-4.myhuaweicloud.com/1.10.1/MindSpore/cpu/x86_64/mindspore-1.10.1-cp39-cp39-win_amd64.whl --trusted-host ms-release.obs.cn-north-4.myhuaweicloud.com -i https://pypi.tuna.tsinghua.edu.cn/simple`
- 验证是否安装成功: `python -c "import mindspore;mindspore.run_check()"`

```
(mindspore) C:\Users\16906>python -c "import mindspore;mindspore.run_check()"
MindSpore version: 1.10.1
The result of multiplication calculation is correct, MindSpore has been installed successfully!
```

实验要求

1. 补全代码，实现LeNet网络。包括网络的初始化和前向过程。
2. 训练LeNet网络，调整网络参数，输出分类准确率。
4. 6月19日晚上12:00之前将代码（.py或者.ipynb文件）、实验报告（doc或pdf文件）

一并打包上传至邮箱。邮箱：mlspring2023@163.com

压缩包和实验报告命名方式：

实验序号_学号_姓名，例如：实验1_111702xxxxxx_王xx

```
class LeNet5(nn.Cell):  
    """  
    Lenet网络结构  
    """  
    def __init__(self, num_class=10, num_channel=1):  
        super(LeNet5, self).__init__()  
        # Todo: 定义Lenet网络结构所需要的操作  
  
    def construct(self, x):  
        # Todo: 使用定义好的操作构建前向网络
```

附件1：pycharm安装和使用

下载pycharm，下载社区版，也可下载专业版，经过学生认证后也可免费使用。下载链接：

<https://www.jetbrains.com/pycharm/download/#section=windows>

下载 PyCharm

Windows

macOS

Linux

Professional

适用于科学和 Web Python 开发。支持 HTML、JS 和 SQL。

下载

免费试用

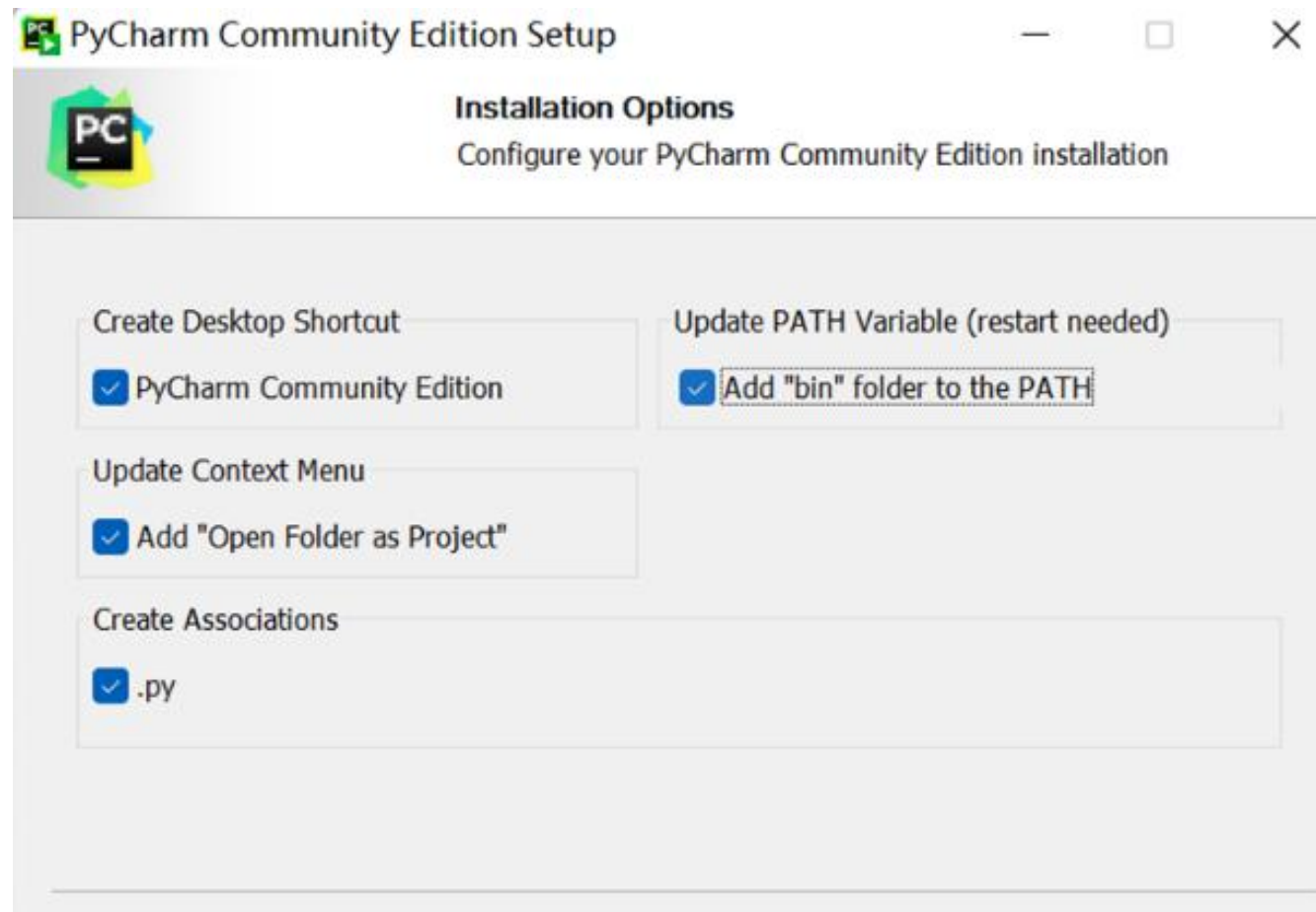
Community

适用于纯 Python 开发

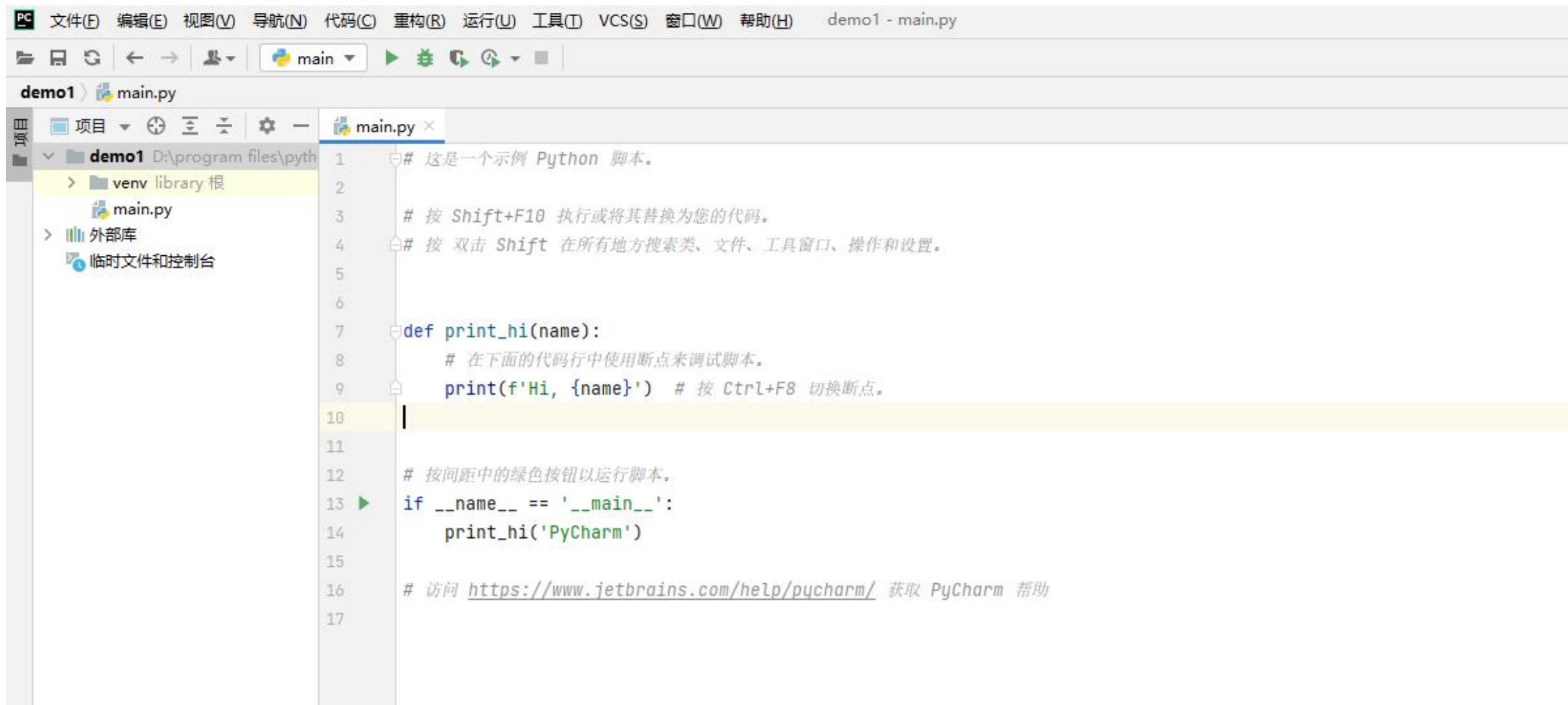
下载

免费，开源构建

2.安装时选择自己安装路径，勾选如下选项，其他的全部默认即可。安装完后启动pycharm



3.新建项目后即可开始写python代码

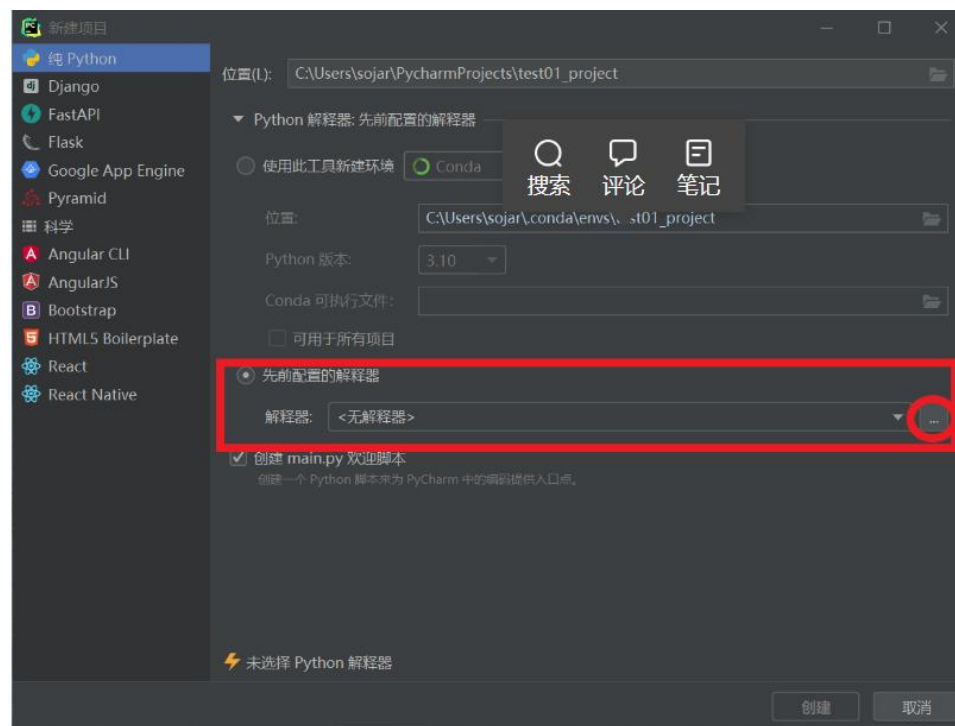


附件2：在Pycharm中配置conda的环境

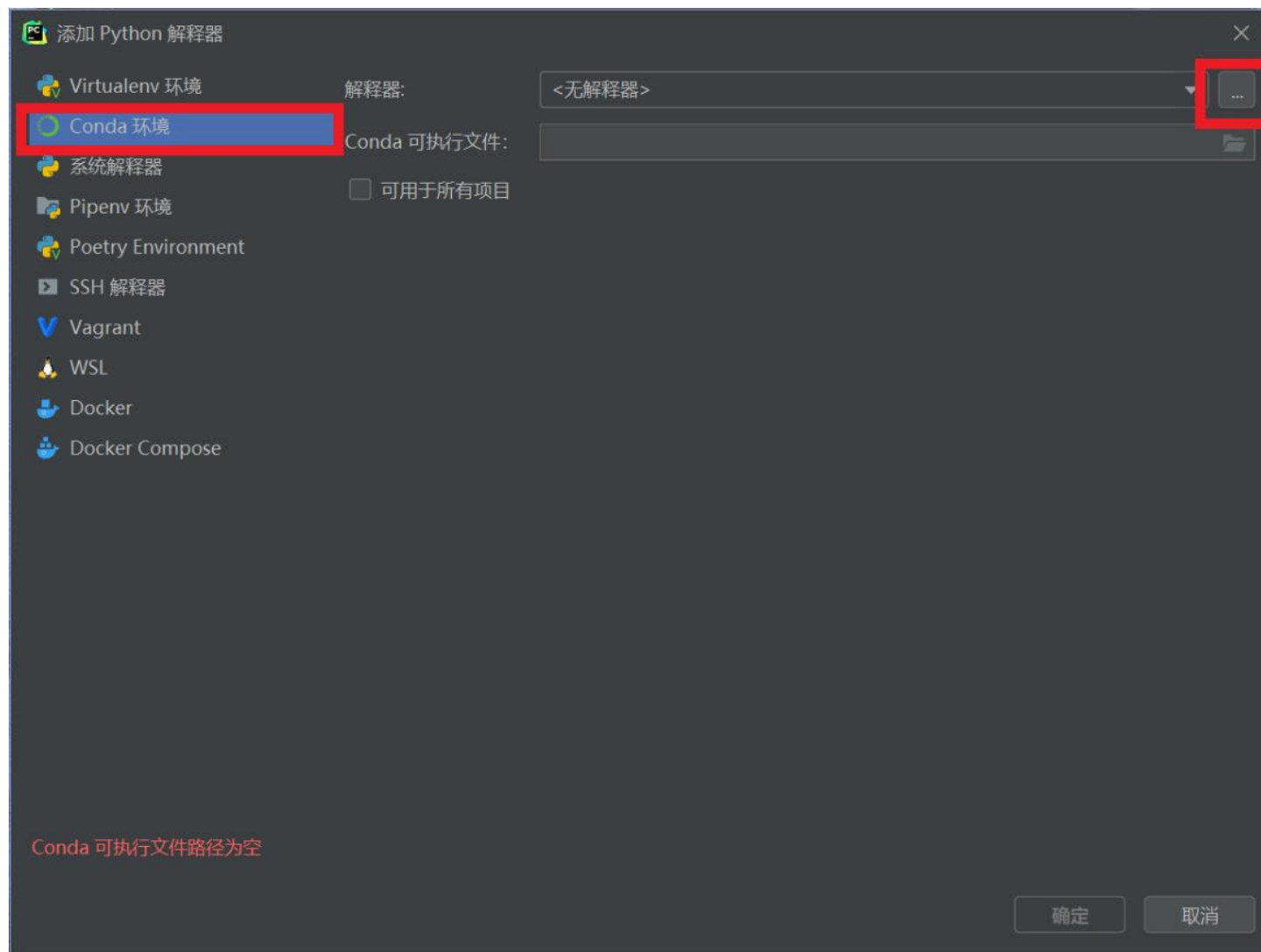
首先你要提前安装好Pycharm和Anaconda在电脑上

首先你要提前安装好Pycharm和Anaconda在电脑上，Anaconda在安装时会配置好一个python环境和带上很多python的第三方库，可以利用conda自带的python环境而不用另外给pycharm配置python环境

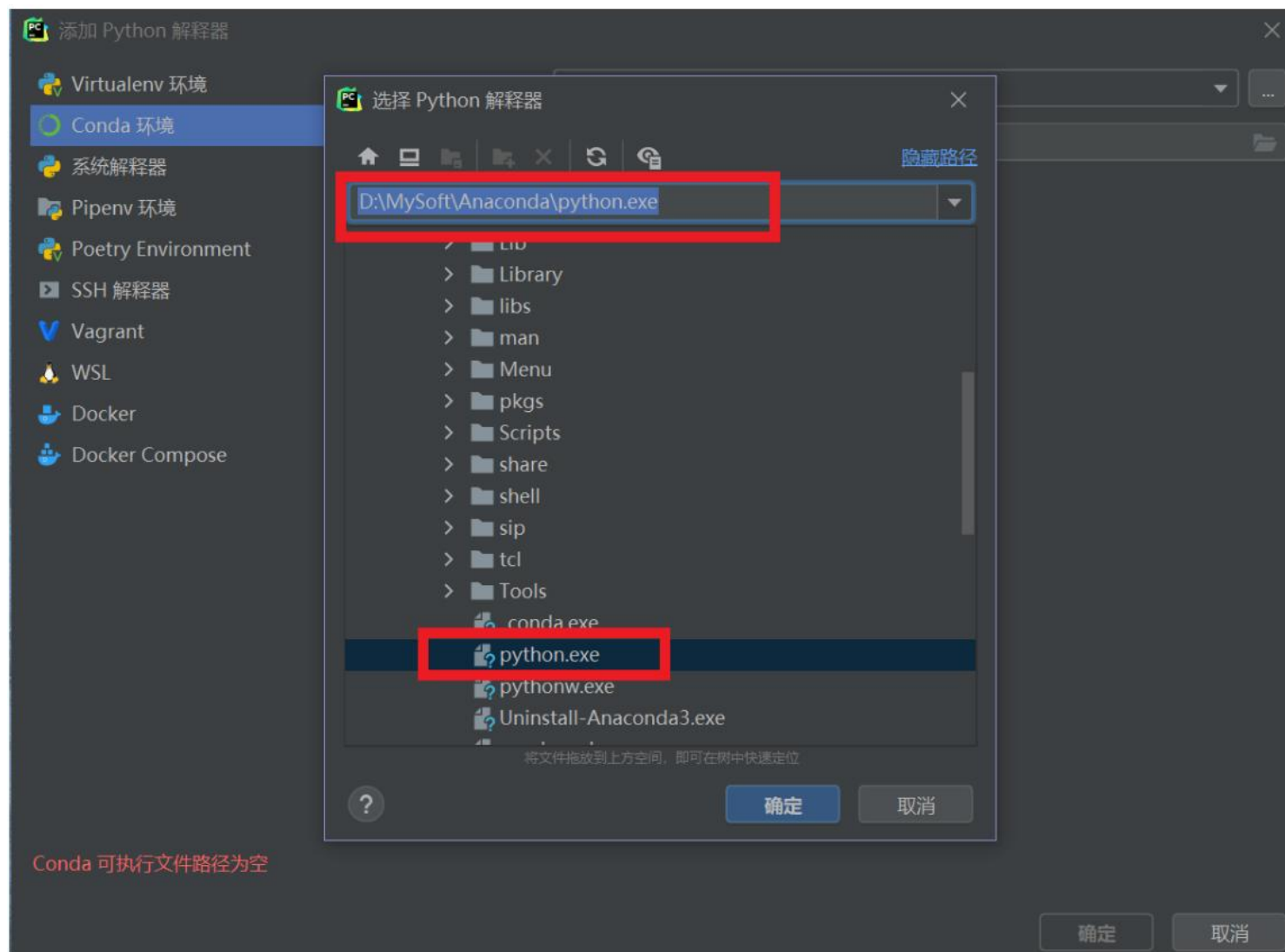
1、先在pycharm新建一个项目，选中如图所示的 先前配置的解器，再点击右边的那三个小点：



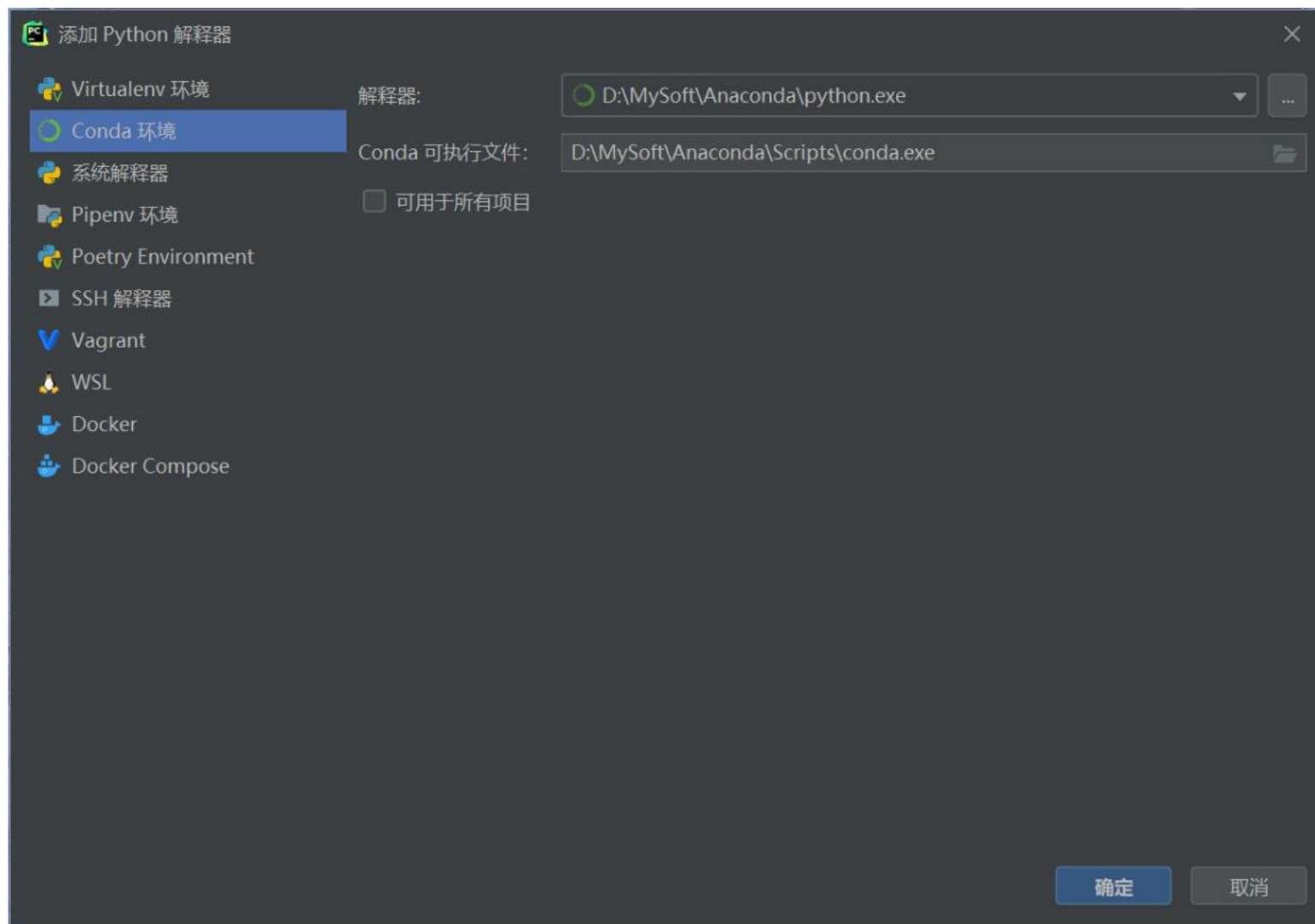
2、再点击左边的 Conda环境，点击右边的三个小点在Anaconda文件中寻找解释器：



3、找到你在电脑上的Anaconda的安装路径，并且在这个目录下找到python.exe(在Tools下面)，然后点击确认：



4、进入这个界面后继续点击确认：



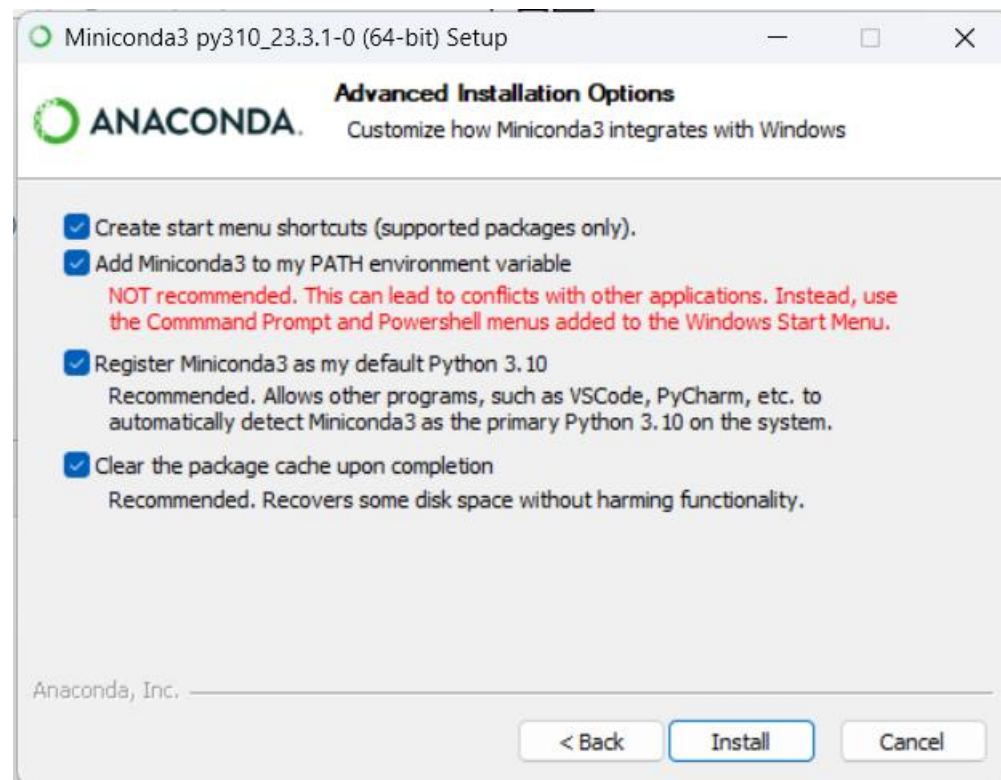
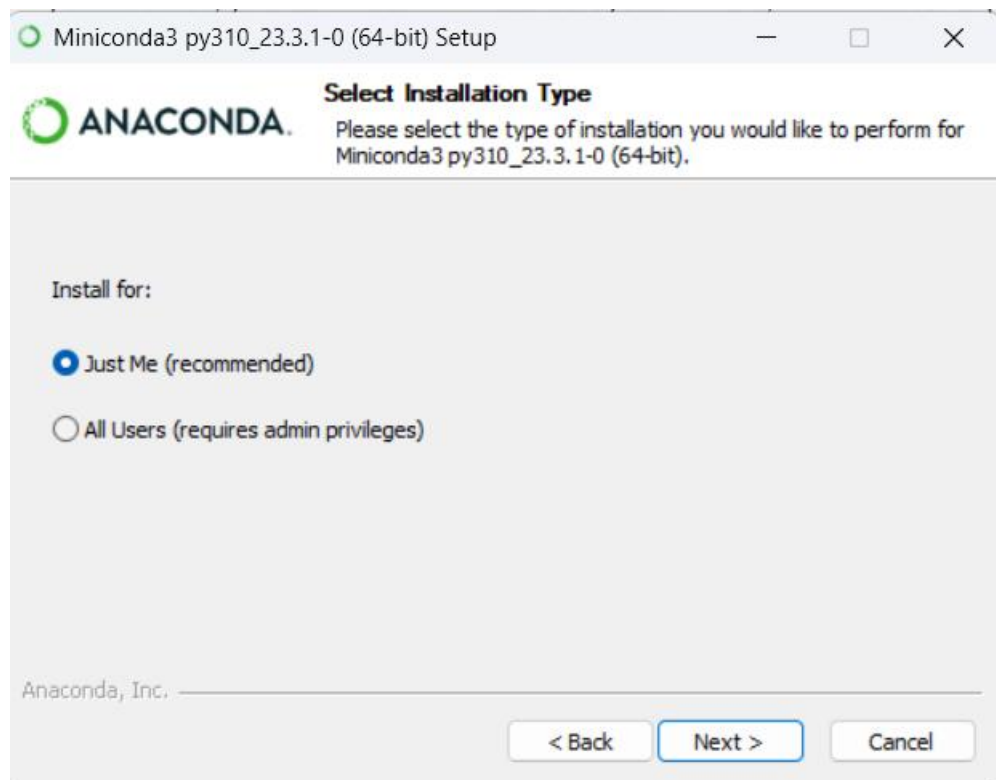
附件三：安装Anaconda/miniconda

- 下载地址：<https://docs.conda.io/en/latest/miniconda.html>
<https://www.anaconda.com/>
- 选择适合自己操作系统的版本，这里选择的是Windows64位版本

Latest Miniconda Installer Links

Latest - Conda 23.3.1 Python 3.10.10 released April 24, 2023

Platform	Name	SHA256 hash
Windows	Miniconda3 Windows 64-bit	307194e1f12bbeb52b083634e89cc67db4f7980bd542254b43d3309eaf7cb358
	Miniconda3 Windows 32-bit	4fb64e6c9c28b88beab16994bfba4829110ea3145baa60bda5344174ab65d462
macOS	Miniconda3 macOS Intel x86 64-bit bash	5abc78b664b7da9d14ade330534cc98283bb838c6b10ad9cfd8b9cc4153f8104
	Miniconda3 macOS Intel x86 64-bit pkg	cca31a0f1e5394f2b739726dc22551c2a19afd689c13a25668887ba706cba58
	Miniconda3 macOS Apple M1 64-bit bash	9d1d12573339c49050b0d5a840af0ff6c32d33c3de1b3db478c01878eb003d64
	Miniconda3 macOS Apple M1 64-bit pkg	6997472c5ff90a772eb77e6397f4e3e227736c83a7f7b839da33d6cc7facb75d
Linux	Miniconda3 Linux 64-bit	aef279d6baea7f67940f16aad17ebe5f6aac97487c7c03466ff01f4819e5a651
	Miniconda3 Linux-aarch64 64-bit	6950c7b1f4f65ce9b87ee1a2d684837771ae7b2e6044e0da9e915d1dee6c924c
	Miniconda3 Linux-ppc64le 64-bit	b3de538cd542bc4f5a2f2d2a79386288d6e04f0e1459755f3cefe64763e51d16
	Miniconda3 Linux-s390x 64-bit	ed4f51afc967e921ff5721151f567a4c43c4288ac93ec2393c6238b8c4891de8



为本用户安装，并添加到环境变量

常用的conda命令

- 新建环境: `conda create -n 环境名 python==3.9`

例如: `conda create -n mindspore python==3.9`会创建一个名为mindspore的python环境。

- 激活环境: `conda activate 环境名`

默认的conda环境是base, 如图所示:

```
(base) C:\Users\16906>
```

使用`conda activate mindspore`可以切换到刚刚创建的python3.9环境:

```
(base) C:\Users\16906>conda activate mindspore
```

```
(mindspore) C:\Users\16906>python
Python 3.9.16 (main, Mar  8 2023, 10:39:24) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

常用的conda命令

- 查看环境列表: conda env list

```
(mindspore) D:\python\code\feedforward>conda env list
# conda environments:
#
base                                D:\Anaconda
CSP_modified-master                D:\Anaconda\envs\CSP_modified-master
Facerecg_code                      D:\Anaconda\envs\Facerecg_code
                                   D:\Anaconda\envs\Facerecg_code\keshe
code                               D:\Anaconda\envs\code
huawei                              D:\Anaconda\envs\huawei
mindspore                          * D:\Anaconda\envs\mindspore
                                   D:\Anaconda\envs\papercode\adaptivw-inattention
python数据分析                     D:\Anaconda\envs\python数据分析
tensor                             D:\Anaconda\envs\tensor
tf                                  D:\Anaconda\envs\tf
untitled                           D:\Anaconda\envs\untitled
人工智能                           D:\Anaconda\envs\人工智能
                                   D:\Anaconda\envs\人工智能\env
```

- 删除环境: conda remove -n 环境名 --all