

## 一、实验环境配置：

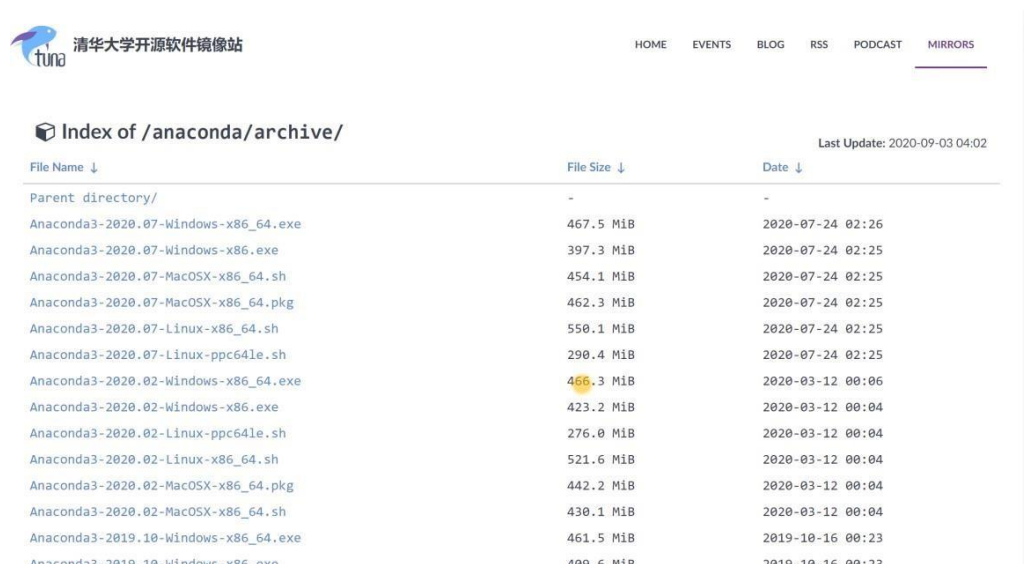
- Anaconda3
- Keras/Tensorflow/Pytorch
- OpenCV,
- Sklearn
- Jupyter Notebook, <https://www.jianshu.com/p/91365f343585>

## 二、64 位 Windows 下进行软件环境安装

本节介绍 64 位 windows 下如何快速安装 conda, python, numpy, sklearn, pytorch 等。

### 1、安装相应版本的 Anaconda（安装完成后, python, numpy, sklearn 都可用）

- (1) 访问<https://mirrors.tuna.tsinghua.edu.cn/anaconda/archive/>



The screenshot shows the 'Index of /anaconda/archive/' page on the Tsinghua University Open Source Software Mirror. It lists various Anaconda installers for different operating systems and architectures. The table has three columns: File Name, File Size, and Date. The file 'Anaconda3-2019.10-Windows-x86\_64.exe' is highlighted in yellow.

File Name ↓	File Size ↓	Date ↓
Parent directory/	-	-
Anaconda3-2020.07-Windows-x86_64.exe	467.5 MiB	2020-07-24 02:26
Anaconda3-2020.07-Windows-x86.exe	397.3 MiB	2020-07-24 02:25
Anaconda3-2020.07-MacOSX-x86_64.sh	454.1 MiB	2020-07-24 02:25
Anaconda3-2020.07-MacOSX-x86_64.pkg	462.3 MiB	2020-07-24 02:25
Anaconda3-2020.07-Linux-x86_64.sh	550.1 MiB	2020-07-24 02:25
Anaconda3-2020.07-Linux-ppc64le.sh	290.4 MiB	2020-07-24 02:25
Anaconda3-2020.02-Windows-x86_64.exe	466.3 MiB	2020-03-12 00:06
Anaconda3-2020.02-Windows-x86.exe	423.2 MiB	2020-03-12 00:04
Anaconda3-2020.02-Linux-ppc64le.sh	276.0 MiB	2020-03-12 00:04
Anaconda3-2020.02-Linux-x86_64.sh	521.6 MiB	2020-03-12 00:04
Anaconda3-2020.02-MacOSX-x86_64.pkg	442.2 MiB	2020-03-12 00:04
Anaconda3-2020.02-MacOSX-x86_64.sh	430.1 MiB	2020-03-12 00:04
Anaconda3-2019.10-Windows-x86_64.exe	461.5 MiB	2019-10-16 00:23
Anaconda3-2019.10-Windows-x86.exe	409.6 MiB	2019-10-16 00:23

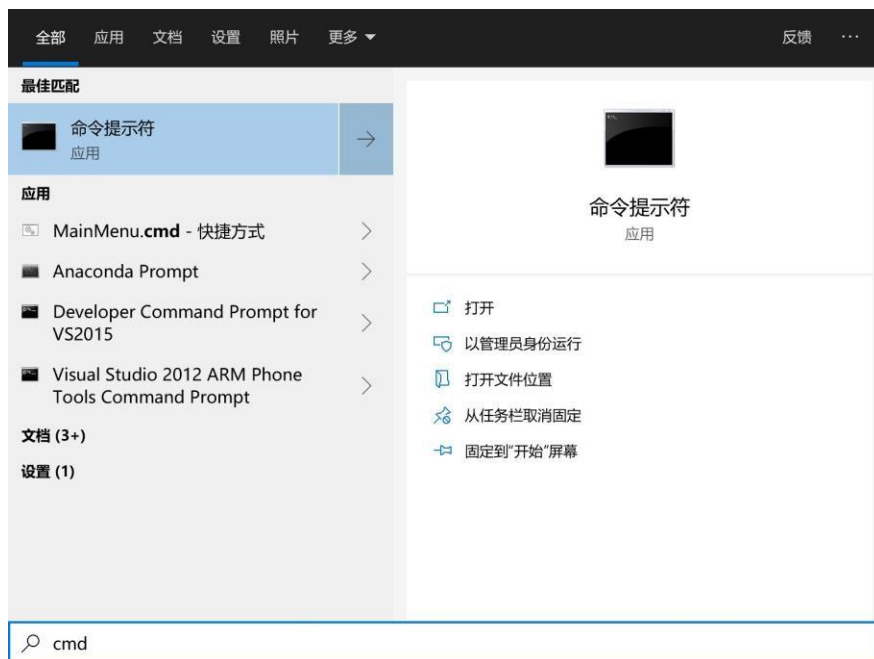
- (2) 点击下载相应安装程序。

选择下载 64 位安装程序，如：Anaconda3-2019.07-Windows-x86\_64.exe。

- (3) 下载完成后，直接双击 exe 文件安装即可。

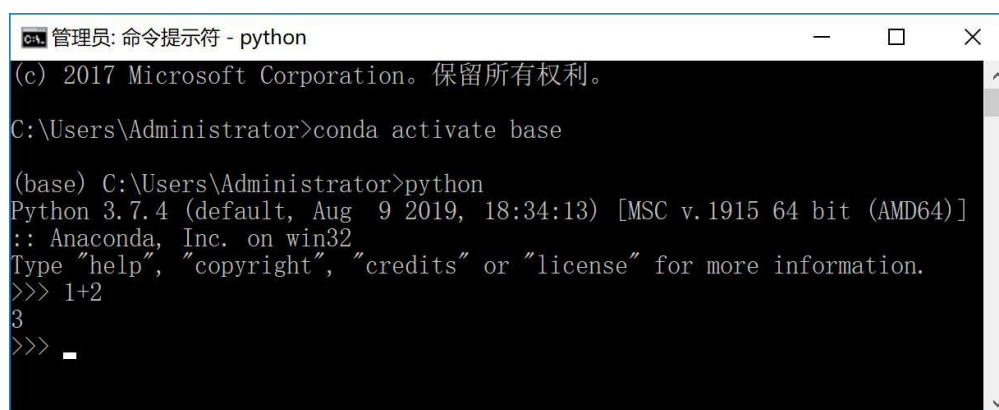
安装时，注意把 conda 路径放到 Windows 的 PATH 环境变量中。（此电脑-属性-高级系统设置-环境变量）

- (4) 安装完成后，运行 cmd，进入 Windows 下的命令行窗口。

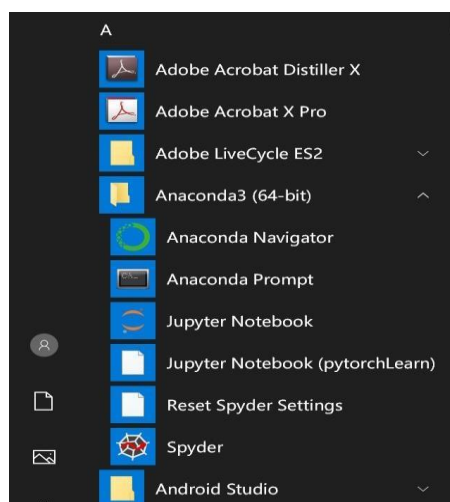


(5) 测试 `conda` 是否安装成功。

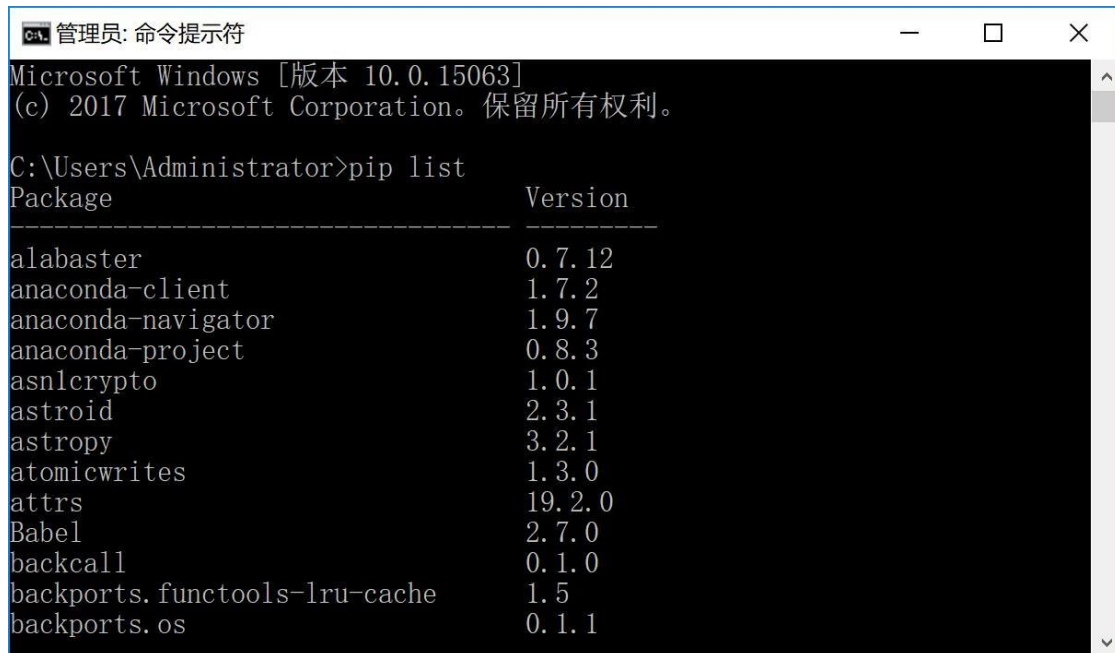
在命令行下运行 `conda activate base` 即进入 `base` 环境，然后运行 `python`，进入Python环境：



上述界面表示 `conda` 安装成功。Windows 开始菜单中也有相关程序：

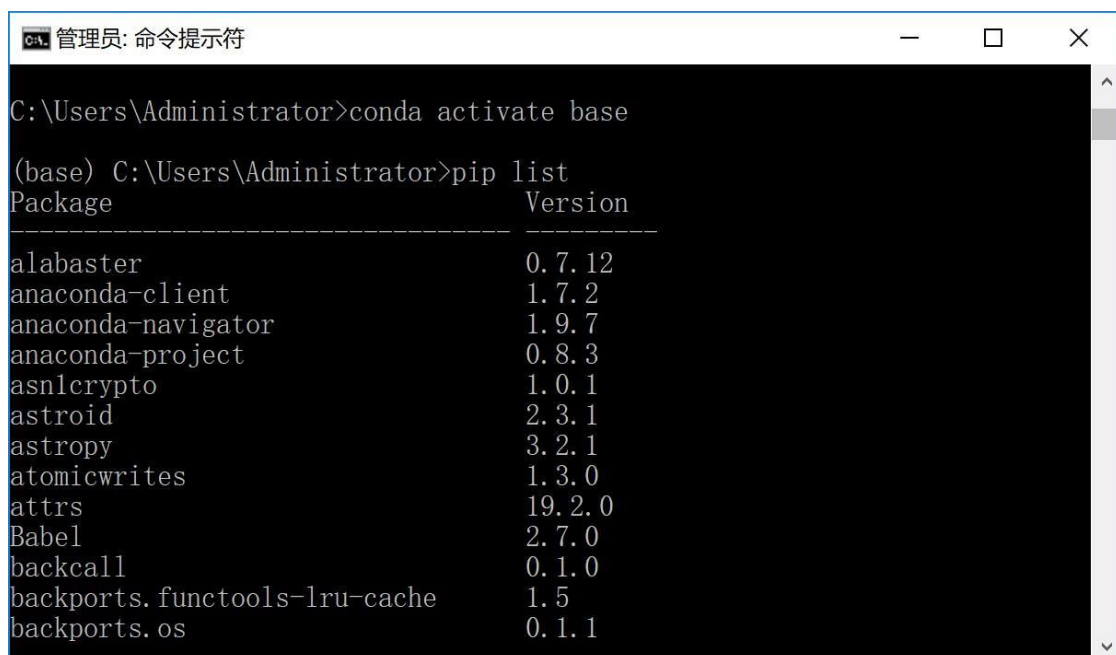


(6) 在 windows 命令行下 conda 的 base 环境中或运行 `pip list` 查看 conda 默认已经安装的包。可以发现 `numpy`, `sklearn` 等都已经安装好。



```
Microsoft Windows [版本 10.0.15063]
(c) 2017 Microsoft Corporation。保留所有权利。

C:\Users\Administrator>pip list
Package                                Version
-----
alabaster                              0.7.12
anaconda-client                        1.7.2
anaconda-navigator                    1.9.7
anaconda-project                      0.8.3
asn1crypto                            1.0.1
astroid                                2.3.1
astropy                                3.2.1
atomicwrites                           1.3.0
attrs                                  19.2.0
Babel                                  2.7.0
backcall                               0.1.0
backports.functools-lru-cache          1.5
backports.os                           0.1.1
```



```
C:\Users\Administrator>conda activate base

(base) C:\Users\Administrator>pip list
Package                                Version
-----
alabaster                              0.7.12
anaconda-client                        1.7.2
anaconda-navigator                    1.9.7
anaconda-project                      0.8.3
asn1crypto                            1.0.1
astroid                                2.3.1
astropy                                3.2.1
atomicwrites                           1.3.0
attrs                                  19.2.0
Babel                                  2.7.0
backcall                               0.1.0
backports.functools-lru-cache          1.5
backports.os                           0.1.1
```

## 2. 在 conda 环境中安装 pytorch

(1) 修改配置文件“.condarc”

首先运行: `conda config --set show_channel_urls yes`

该命令在 windows 用户目录下（如 C:\Users\administrator\）生成该文件，打开该文件，将如下内容进行替换文件内容：

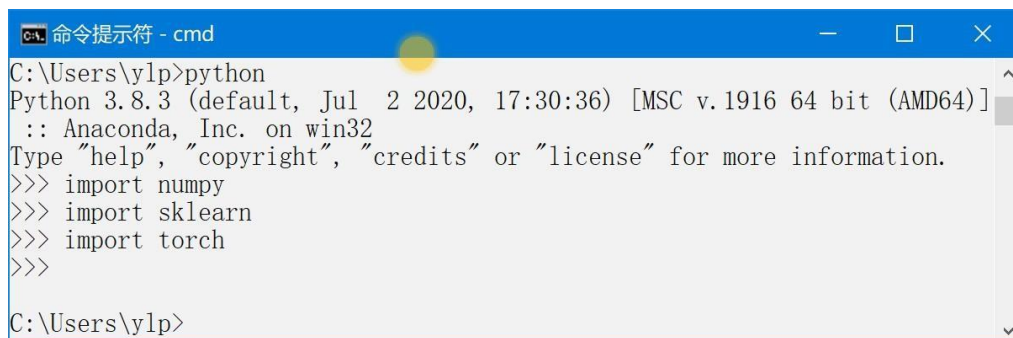
```
channels:
  - defaults
show_channel_urls: true
channel_alias: https://mirrors.tuna.tsinghua.edu.cn/anaconda
default_channels:
  - https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main
  - https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/free
  - https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/r
  - https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/pro
  - https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/msys2
custom_channels:
```

(2) 运行 cmd，进入命令行，运行 `conda install pytorch`

安装 CPU 版本的 pytorch。安装过程中根据提示，输入相关选择，如 yes。

若在自己带 GPU 的电脑上安装 pytorch，请访问 [pytorch.org](https://pytorch.org)，根据说明进行安装。

(3) 检查是否可使用 python 下的 pytorch 框架。



```
命令提示符 - cmd
C:\Users\yly>python
Python 3.8.3 (default, Jul 2 2020, 17:30:36) [MSC v.1916 64 bit (AMD64)]
:: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import numpy
>>> import sklearn
>>> import torch
>>>
C:\Users\yly>
```

(4) 若安装 torchvision，则执行：

```
conda install torchvision -c pytorch
```

## 实验二选一

### 选项 1 CIFAR-10图像分类实验（评分加权系数： 0.85）

#### 1.1 实验目标

通过人工智能导论所学内容，应用机器学习模型如 **SVM** 以及深度学习的方法完成图像分类的建模。

#### 1.2 实验内容

##### 1.2.1 卷积神经网络

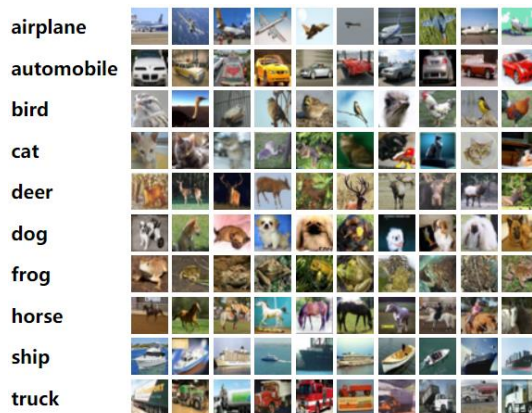
利用Pytorch或Tensorflow等深度学习框架实现（可选在jupyter notebook上完成，有加分）：使用~~卷积神经网络~~在CIFAR-10上完成分类任务，并打印测试集准确率。

##### 1.2.2 SVM

利用sklearn中自带的~~SVM~~库函数实现（可选在jupyter notebook上完成，有加分）：使用~~SVM~~在CIFAR-10上完成分类任务，并打印测试集准确率。

#### 1.3 实验数据

CIFAR-10 是由 Hinton 的学生 Alex Krizhevsky 和 Ilya Sutskever 整理的一个用于识别普适物体的小型数据集。一共包含 10 个类别的 RGB 彩色图 片：飞机（ airplane ）、汽车（ automobile ）、鸟类（ bird ）、猫（ cat ）、鹿（ deer ）、狗（ dog ）、蛙类（ frog ）、马（ horse ）、船（ ship ）和卡车（ truck ）。图片的尺寸为 32×32 ，数据集中一共有 50000 张训练图片和 10000 张测试图片。



#### 1.4 实验验收

现场运行，现场验收。

## 选项 2 文本情感分析实验（评分加权系数：1.0）

### 2.1 实验目标

通过人工智能导论所学内容，应用机器学习以及深度学习的方法完成文本情感分析的建模。

初步掌握经典的机器学习方法 **SVM** 在文本分类的应用

初步掌握循环神经网络及卷积神经网络在文本分类的应用

比较机器学习以及深度学习方法的异同

### 2.2 实验内容

#### 2.2.1 机器学习模型

使用sklearn中自带的SVM库函数完成IMDB数据集分类任务（可选使用jupyter notebook完成，有加分）

#### 2.2.2 深度学习模型

使用Pytorch或Tensorflow等深度学习框架完成（可选使用jupyter notebook，有加分）：

- （1）单独使用循环神经网络如 **GRU** 或者 **LSTM** 在 **IMDB** 数据集上完成分类任务，并且打印测试集准确率
- （2）单独使用一维卷积神经网络在 **IMDB** 数据集上完成分类任务，并且打印测试集准确率
- （3）组合使用一维卷积神经网络以及循环神经网络级联在 **IMDB** 数据集上完成分类任务，并且打印测试集准确率

自行调整库函数或者各网络中的超参数，如句子长度、词嵌入大小、循环神经隐状态大小，卷积核尺寸以及数目等，最终取测试集上准确率最高时的超参数。

### 2.3 实验数据集

IMDB 情绪分类数据集由 IMDB 用户的 50,000 个电影评论组成，标签为二分类：positive（1）和 negative（0）。[请点击数据下载，提取码 8td7。](#)

本链接的评论数据已经过预处理，每一个评论样本都被编码为一个整数形式的单词索引序列，同时为了得到的向量数据不会太大，该链接里的数据集仅保留前 10000 个最常见出现的单词，低频单词已被舍弃。评论中的单词按照它们在数据集中的总体频率进行索引，即整数编码  $i$  为第  $i$  个最频繁的词。例如本链接训练集第一个样本为：

[1,14,22,16,43,530,973,1622,1385,65,458,4468,66,3941,4,173,36,256,5,25,100,43,838,112,50,670,2,9,3  
5,480,284,5,150,4,172,112,167,2,336,385,39,4,172,4536,1111,17,546,38,13,447,4,192,50,16,6,147,202  
5,19,14,22,4,1920,4613,469,4,22,71,87,12,16,43,530,38,76,15,13,1247,4,22,17,515,17,12,16,626,18,2,5  
,62,386,12,8,316,8,106,5,4,2223,5244,16,480,66,3785,33,4,130,12,16,38,619,5,25,124,51,36,135,48,25,  
1415,33,6,22,12,215,28,77,52,5,14,407,16,82,2,8,4,107,117,5952,15,256,4,2,7,3766,5,723,36,71,43,530  
,476,26,400,317,46,7,4,2,1029,13,104,88,4,381,15,297,98,32,2071,56,26,141,6,194,7486,18,4,226,22,2  
1,134,476,26,480,5,144,30,5535,18,51,36,28,224,92,25,104,4,226,65,16,38,1334,88,12,16,283,5,16,447  
2,113,103,32,15,16,5345,19,178,32]

该样本真实评论数据为：

"? this film was just brilliant casting location scenery story direction everyone's really suited the part they played and you could just imagine being there robert ? is an amazing actor and now the same being director ? father came from the same scottish island as myself so i loved the fact there was a real connection with this film the witty remarks throughout the film were great it was just brilliant so much that i bought the film as soon as it was released for ? and would recommend it to everyone to watch and the fly fishing was amazing really cried at the end it was so sad and you know what they say if you cryat a film it must have been good and this definitely was also ? to the two little boy's that played the ? of norman and paul they were just brilliant children are often left out of the ? list i think because the stars that play them all grown up are such a big profile for the whole film but these children are amazing and should be praised for what they have done don't you think the whole story was so lovely because it was true and was someone's life after all that was shared with us all"

在本数据集的 50,000 个评论中，25000 个作为训练集，另 25000 个作为测试集，训练集和测试集的 **positive** 样本和 **negative** 样本数目均衡。

数据加载可参考：

```
import pickle
with open("./data.pkl","rb") as f:
    train_data, train_labels, test_data, test_labels = pickle.load(f)
```

其中 `train_data` 和 `train_labels` 是训练数据的样本以及标签，`test_data` 和 `test_labels` 是测试集的样本及标签。

## 2.4 实验验收

现场运行，现场验收。