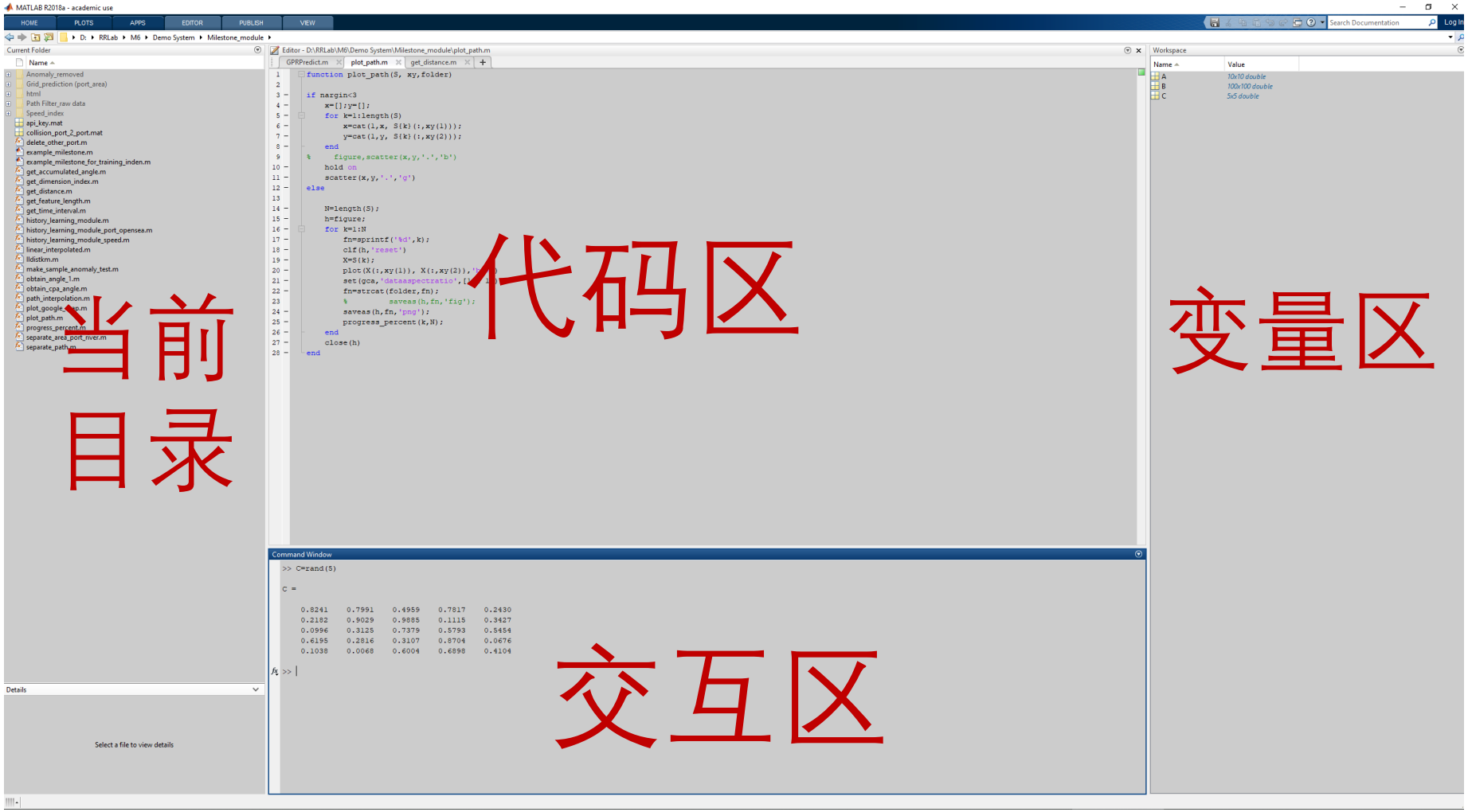
案例实践一：编程与数据分类

主讲人：屠恩美

2018.10

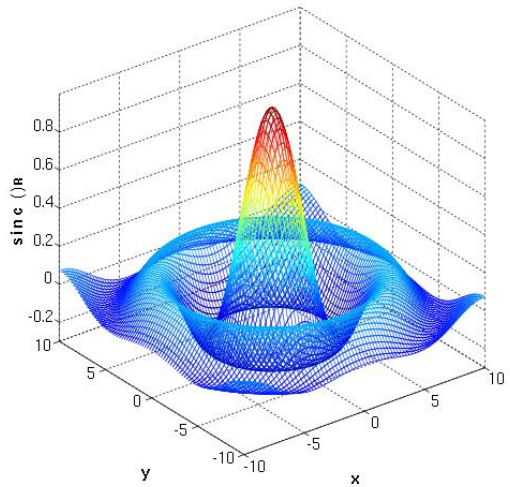
上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

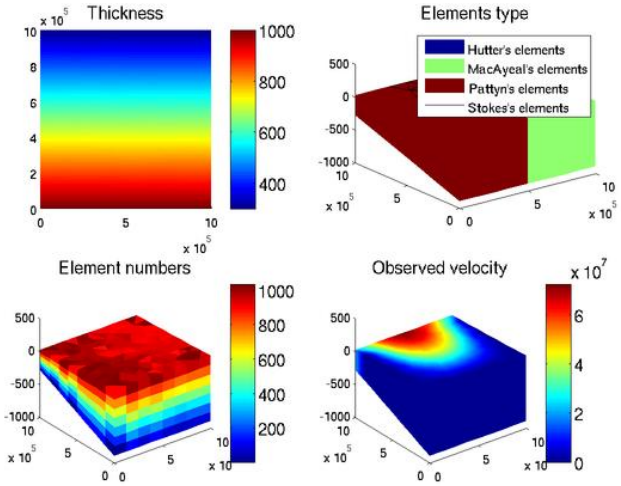
MATLAB


MATLAB R2018a - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Current Folder
Editor - D:\RRLab\M6\Demo System\Milestone_module\plot_path.m
GPRPredict.m
plot_path.m
get_distance.m
function plot_path(S, xy,folder)
if nargin<3
x=[];y=[];
for k=1:length(S)
x=cat(1,x, S{k}(:,xy(1)));
y=cat(1,y, S{k}(:,xy(2)));
end
hold on
scatter(x,y,'.','g')
else
N=length(S);
h=figure;
for k=1:N
fn=sprintf('%d',k);
clf(h,'reset')
X=S{k};
set(gca,'dataaspectratio',[1 1 1])
fn=strcat(folder,fn);
saveas(h,fn,'fig');
progress_percent(k,N);
end
close(h)
end
Workspace
Name
Value
A
10x10 double
B
100x100 double
C
5x5 double
Command Window
>> C=rand(5)
C =
0.8241 0.7991 0.4959 0.7817 0.2430
0.2182 0.9029 0.9885 0.1115 0.3427
0.0996 0.3125 0.7379 0.5793 0.5454
0.6195 0.2816 0.3107 0.8704 0.0676
0.1038 0.0068 0.6004 0.6898 0.4104
Details
Select a file to view details
当前目录
代码区
变量区
交互区

MATLAB

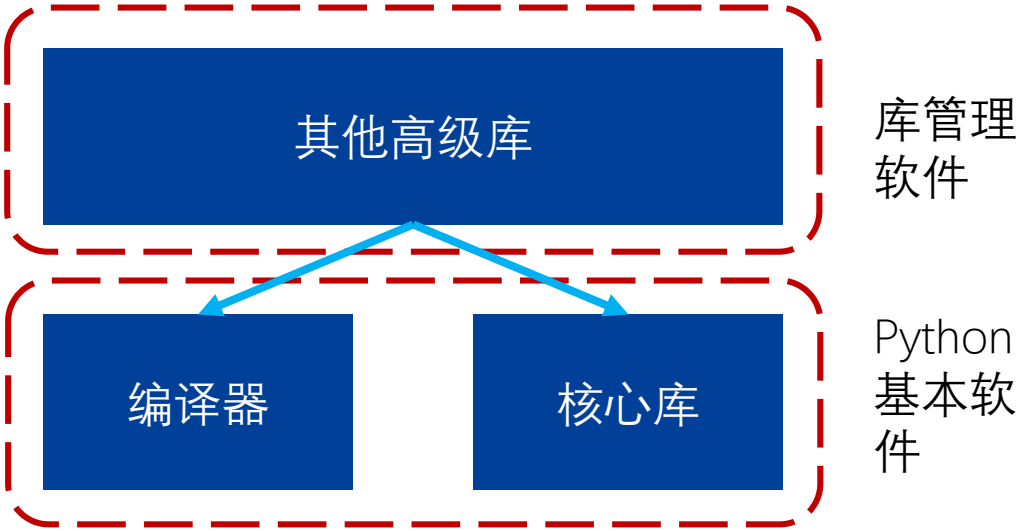
- 编程语言简洁，尤其适合科学计算
- 庞大的计算工具箱集成在同一个环境下，使用便捷
- 丰富的数据可视化(绘图)功能
- 专业的维护，算法实现可靠，更新速度快
- 集成帮助文档全面、细致、专业，教科书级别





Python简介

- Python是一种高级编程语言，1991年发布
- 简单易用，更接近自然语言书写
- 下载网站：https://www.python.org/downloads/rele
- 版本：3.6.x

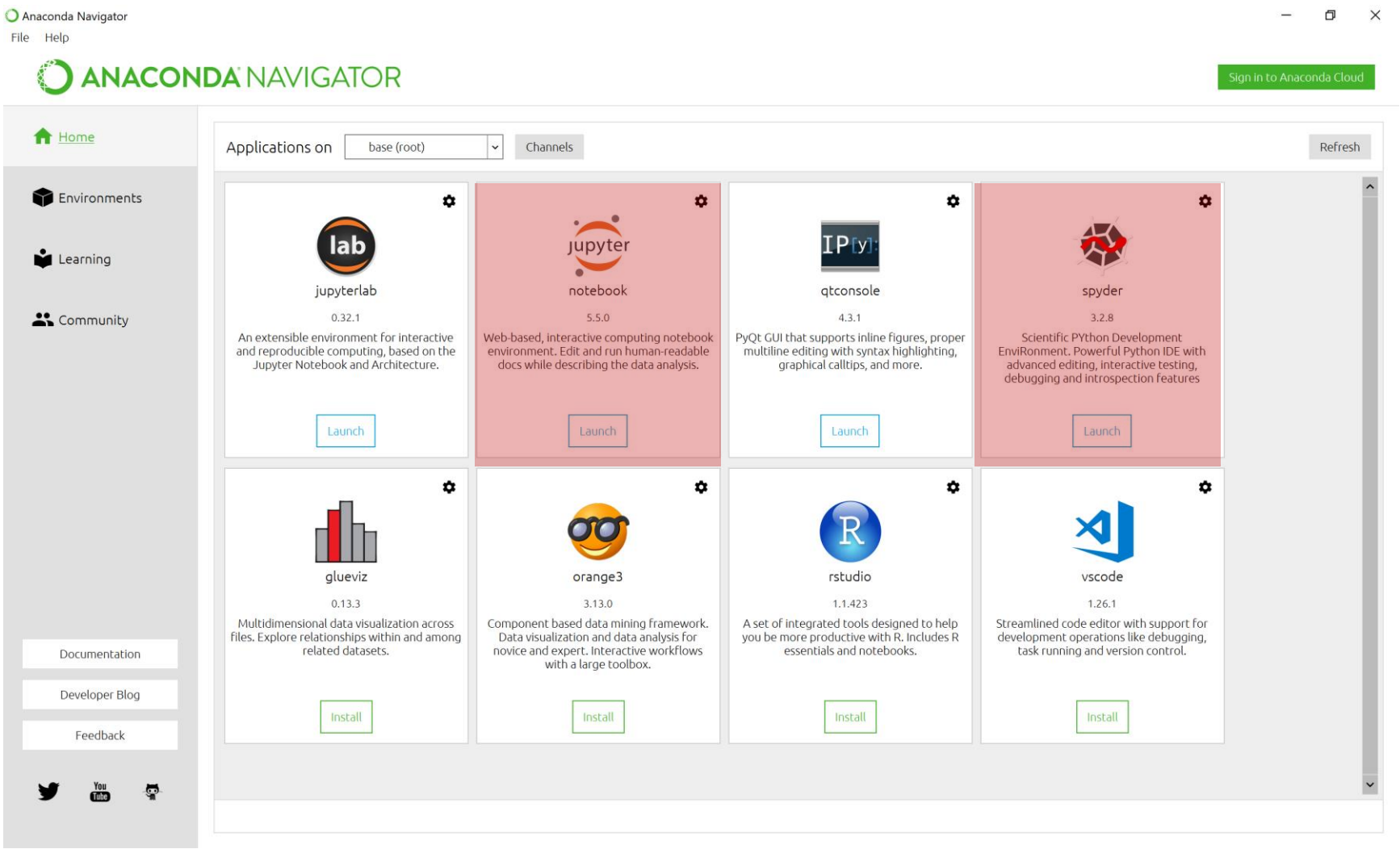


Guido van Rossum

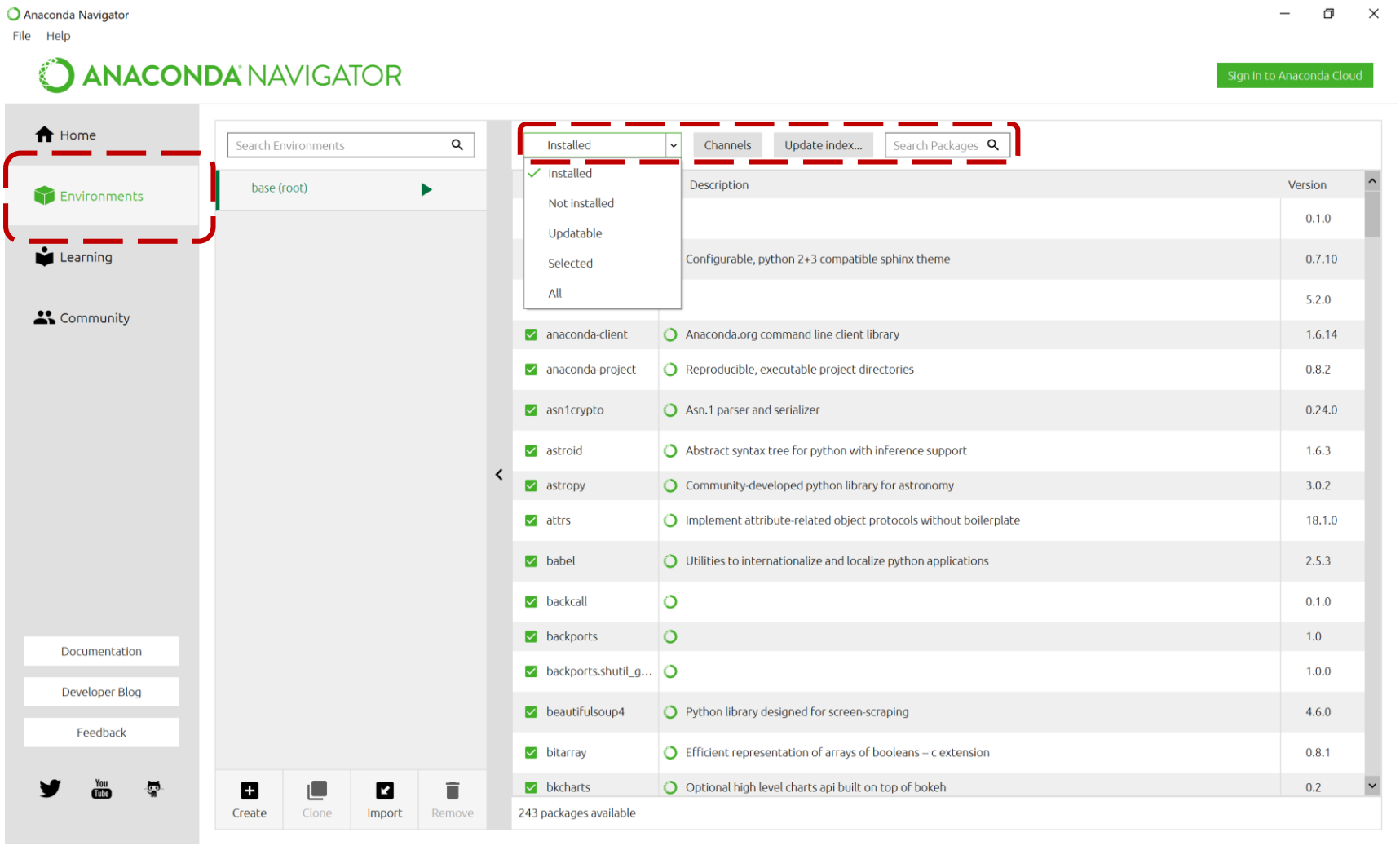
集成安装 - Anaconda

- Anaconda是一个集成的Python软件包/环境管理软件
- 下载地址：https://www.anaconda.com/download/
- 下载版本：3.6 （Windows, Linux, Mac都可以）
- 安装：全部默认，直接点“Next”
- 安装后(可能需要重启电脑），启动Anaconda Navigator

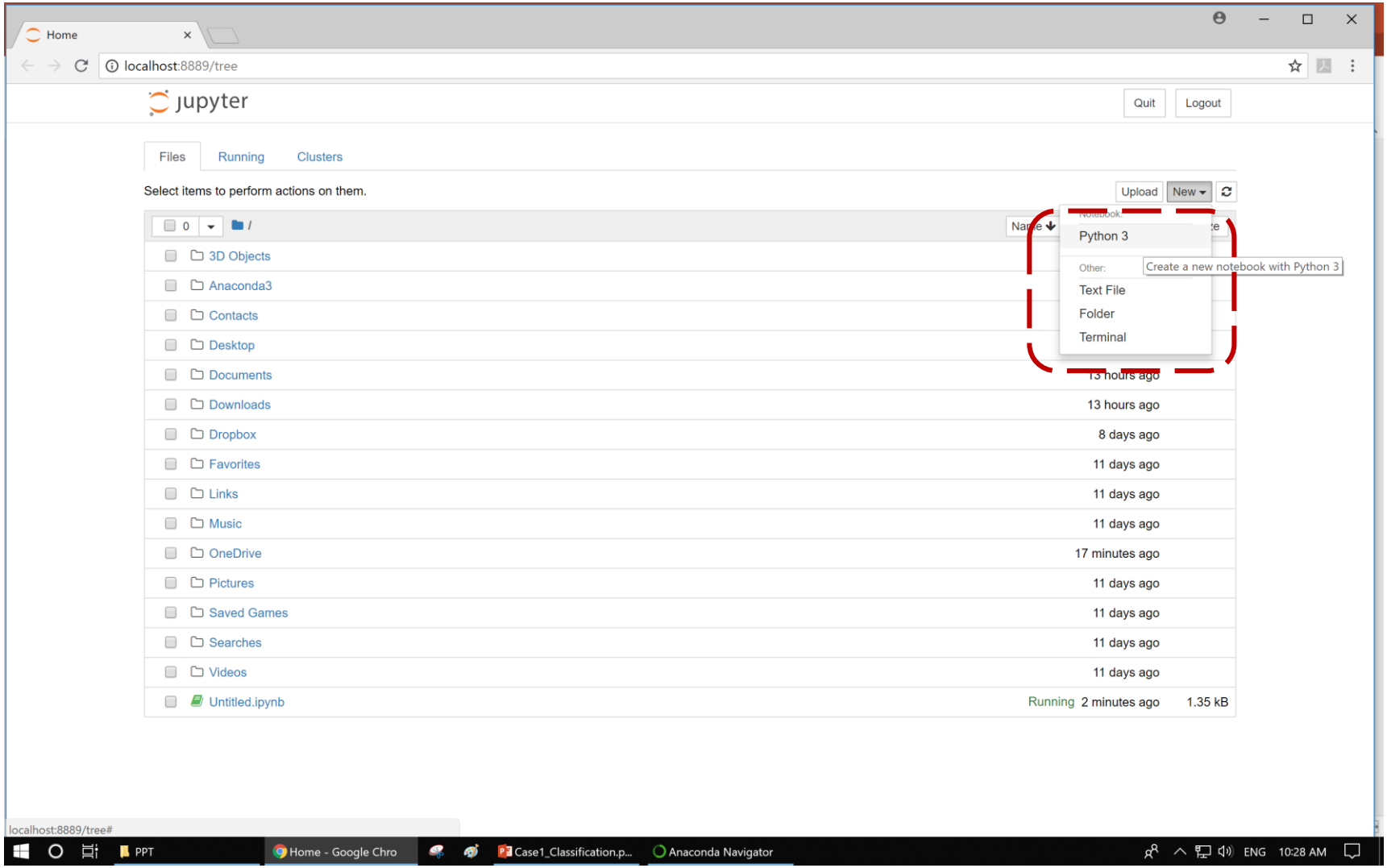
集成安装 - Anaconda



集成安装 - Anaconda



集成安装 - Jupyter



集成安装 - Jupyter

jupyter Untitled1 Last Checkpoint: 3 minutes ago (unsaved changes)

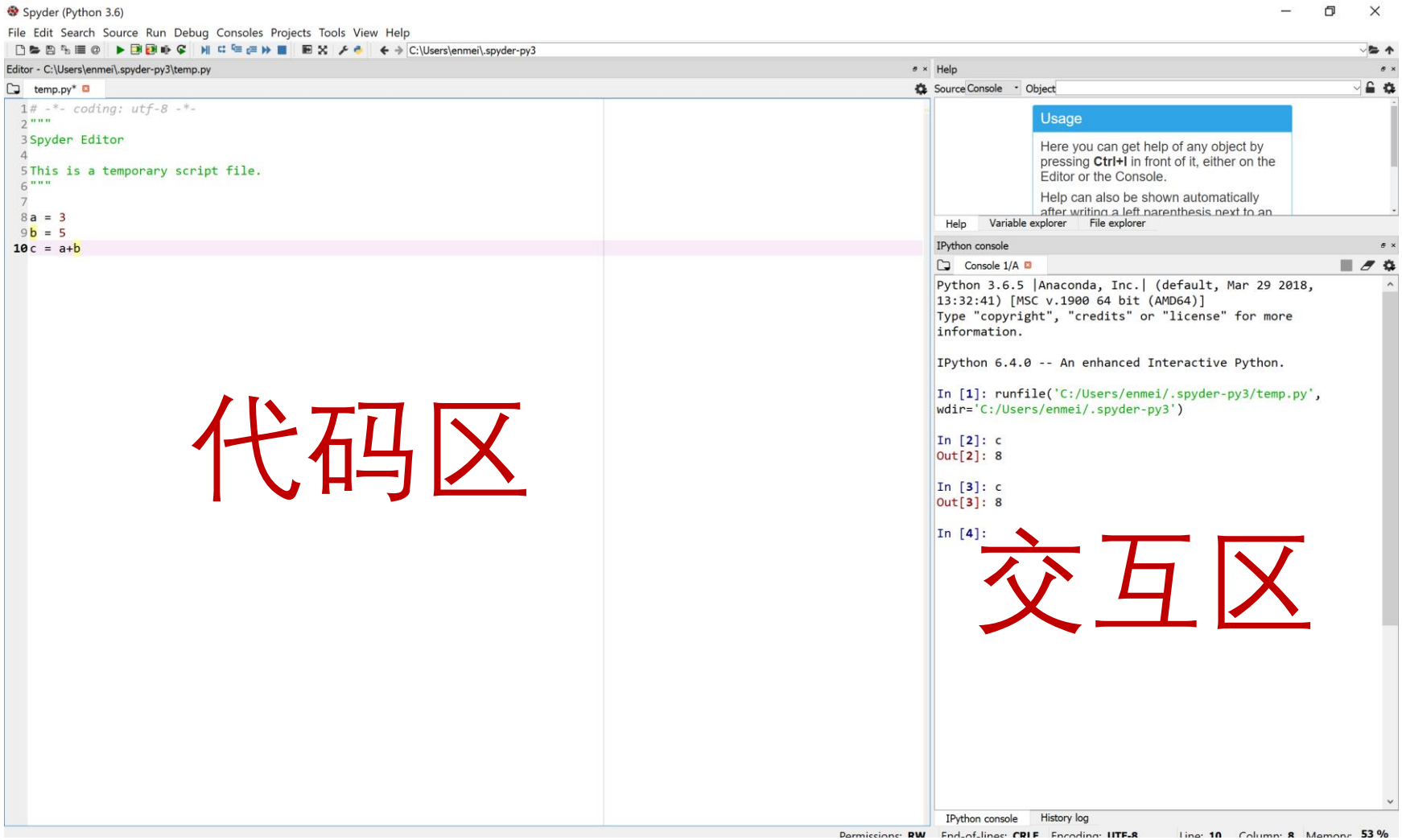
File Edit View Insert Cell Kernel Widgets Help

Run Code

```
In [5]: a = 3
        b = 5

In [6]: c = a+b

In [7]: c
Out[7]: 8

In [ ]:
```

运行某个Cell：“Run” 按钮或者 “Shift”+“ Enter”

集成安装 - Spyder

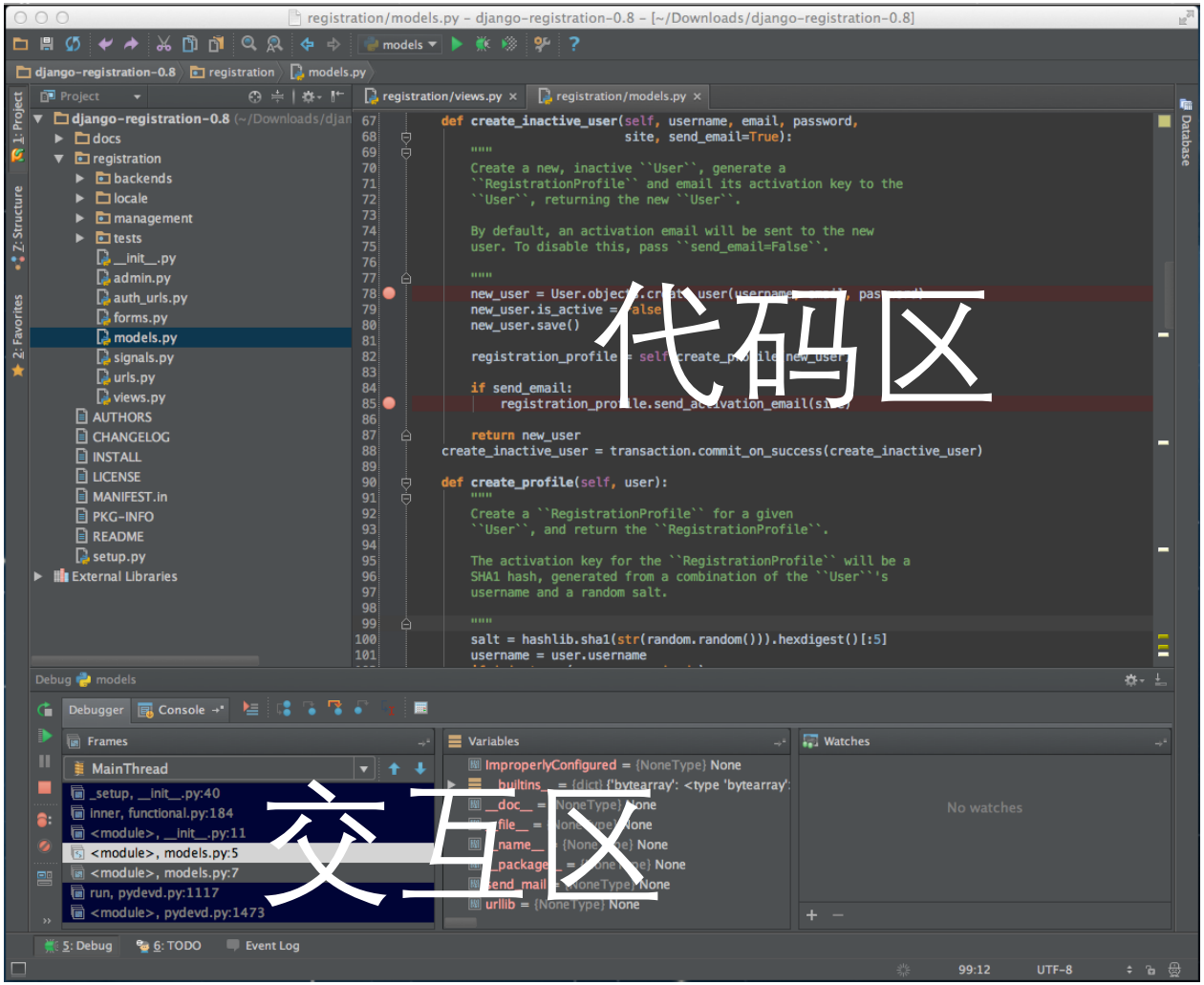

Spyder (Python 3.6)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\enmei\.spyder-py3
Editor - C:\Users\enmei\.spyder-py3\temp.py
temp.py*
-*- coding: utf-8 -*-
"""
Spyder Editor
This is a temporary script file.
"""
a = 3
b = 5
c = a+b
代码区
Help
Source Console
Object
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically
Help
Variable explorer
File explorer
IPython console
Console 1/A
Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:32:41) [MSC v.1900 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 6.4.0 -- An enhanced Interactive Python.
In [1]: runfile('C:/Users/enmei/.spyder-py3/temp.py', wdir='C:/Users/enmei/.spyder-py3')
In [2]: c
Out[2]: 8
In [3]: c
Out[3]: 8
In [4]:
交互区
IPython console
History log

按需安装

- 下载并安装python 3.65:
  https://www.python.org/downloads/release/python-365/
- 安装所需的软件包: pip install packages (numpy, scipy, matplotlib, etc)
- 安装Jupyter:
  - python3 -m pip install --upgrade pip
  - python3 -m pip install jupyter
- 或者安装Pycharm

  https://www.jetbrains.com/pycharm-edu/download/#section=windows

PyCharm



环境比较

Jupyter 还是 PyCharm 还是 Spyder?

- Jupyter 使用方便，交互性好，容易上手，适合初学和小项目
- PyCharm使用稍微繁琐，但功能强大：调试方便，代码提示，版本分离
- Spyder介于二者之间，针对科学计算而开发，对标Matlab部分功能
- 另外PyCharm和Spyder界面可自定义(字体，颜色，主题等)，按个人习惯设置

Python编程

```
x = 34 - 23 # A comment.
y = "Hello" # Another one.
z = 3.45
if z == 3.45 or y == "Hello":
    x = x + 1
    y = y + "World" # String concat.
print(x)
print(y)
```

```
12
HelloWorld
```

Python编程 – if, for, while

- if, for, while等语句块以缩进控制，冒号：起始

```
a = 5
if a < 3:
    a +=1

for i in range(a):
    print(i)
```

```
0
1
2
3
4
```

Python编程 – 函数

- 函数使用def定义，**缩进控制函数范围**
- 接收两种参数：位置参数和关键字参数

```
def add(x, y=1):
    z = x+y
    return z
a = 1
b = 2
c = add(a)
d = add(a, b)
print(c,d)
```

```
2 3
```

Python编程 – 类

- 类使用class定义，**缩进控制类成员**

```
class AutoCounter(object):
    def __init__ (self, base):
        if base > 0:
            self.base = base
        else:
            base = 10
        print(self.base)
    def increase(self, a=1):
        self.base += a
        print(self.base)



c1 = AutoCounter(5)
c1.increase()
```

```
5
6
```

Python编程 – 基本数据类型

- 四种基本数据类型：int, float, complex, string

Numbers: Integers and floats work as you would expect from other languages:

```
x = 3
print(type(x)) # Prints "<class 'int'>"
print(x)       # Prints "3"
print(x + 1)   # Addition; prints "4"
print(x - 1)   # Subtraction; prints "2"
print(x * 2)   # Multiplication; prints "6"
print(x ** 2)  # Exponentiation; prints "9"
x += 1
print(x)  # Prints "4"
x *= 2
print(x)  # Prints "8"
y = 2.5
print(type(y)) # Prints "<class 'float'>"
print(y, y + 1, y * 2, y ** 2) # Prints "2.5 3.5 5.0 6.25"
```

Python编程 – 基本数据类型

Strings: Python has great support for strings:

```
hello = 'hello'    # String literals can use single quotes
world = "world"    # or double quotes; it does not matter.
print(hello)       # Prints "hello"
print(len(hello))  # String length; prints "5"
hw = hello + ' ' + world  # String concatenation
print(hw)  # prints "hello world"
hw12 = '%s %s %d' % (hello, world, 12)  # sprintf style string formatting
print(hw12)  # prints "hello world 12"
```

Python编程 – 高级数据类型

- 四种符合数据类型 List, Dictionary, Tuple, Set

```
xs = [3, 1, 2]     # Create a list
print(xs, xs[2])   # Prints "[3, 1, 2] 2"
print(xs[-1])      # Negative indices count from the end of the list; prints "2"
xs[2] = 'foo'      # Lists can contain elements of different types
print(xs)          # Prints "[3, 1, 'foo']"
xs.append('bar')   # Add a new element to the end of the list
print(xs)          # Prints "[3, 1, 'foo', 'bar']"
x = xs.pop()       # Remove and return the last element of the list
print(x, xs)       # Prints "bar [3, 1, 'foo']"
```

```
nums = list(range(5))     # range is a built-in function that creates a list of integers
print(nums)               # Prints "[0, 1, 2, 3, 4]"
print(nums[2:4])          # Get a slice from index 2 to 4 (exclusive); prints "[2, 3]"
print(nums[2:])           # Get a slice from index 2 to the end; prints "[2, 3, 4]"
print(nums[:2])           # Get a slice from the start to index 2 (exclusive); prints "[0, 1]
print(nums[:])            # Get a slice of the whole list; prints "[0, 1, 2, 3, 4]"
print(nums[:-1])          # Slice indices can be negative; prints "[0, 1, 2, 3]"
nums[2:4] = [8, 9]        # Assign a new sublist to a slice
print(nums)               # Prints "[0, 1, 8, 9, 4]"
```

Python编程 – 高级数据类型

- 四种符合数据类型 List, Dictionary, Tuple, Set

```
d = {'cat': 'cute', 'dog': 'furry'}  # Create a new dictionary with some data
print(d['cat'])       # Get an entry from a dictionary; prints "cute"
print('cat' in d)     # Check if a dictionary has a given key; prints "True"
d['fish'] = 'wet'     # Set an entry in a dictionary
print(d['fish'])      # Prints "wet"
# print(d['monkey'])  # KeyError: 'monkey' not a key of d
print(d.get('monkey', 'N/A'))  # Get an element with a default; prints "N/A"
print(d.get('fish', 'N/A'))    # Get an element with a default; prints "wet"
del d['fish']         # Remove an element from a dictionary
print(d.get('fish', 'N/A')) # "fish" is no longer a key; prints "N/A"
```

```
d = {'person': 2, 'cat': 4, 'spider': 8}
for animal in d:
    legs = d[animal]
    print('A %s has %d legs' % (animal, legs))
# Prints "A person has 2 legs", "A cat has 4 legs", "A spider has 8 legs"
```

Python编程 – 高级数据类型

- 四种符合数据类型 List, Dictionary, Tuple, Set

```
d = {(x, x + 1): x for x in range(10)}  # Create a dictionary with tuple keys
t = (5, 6)        # Create a tuple
print(type(t))    # Prints "<class 'tuple'>"
print(d[t])       # Prints "5"
print(d[(1, 2)])  # Prints "1"
```

- 与list的区别:
  - Tuple中的数值一旦创建后，不能被修改

    t[0] = 2 ✖
  - Tuple可以做dictionary的索引值

Python编程 – 高级数据类型

- 四种符合数据类型 List, Dictionary, Tuple, Set

```
animals = {'cat', 'dog'}
print('cat' in animals)   # Check if an element is in a set; prints "True"
print('fish' in animals)  # prints "False"
animals.add('fish')       # Add an element to a set
print('fish' in animals)  # Prints "True"
print(len(animals))       # Number of elements in a set; prints "3"
animals.add('cat')        # Adding an element that is already in the set does nothing
print(len(animals))       # Prints "3"
animals.remove('cat')     # Remove an element from a set
print(len(animals))       # Prints "2"
```

Numpy

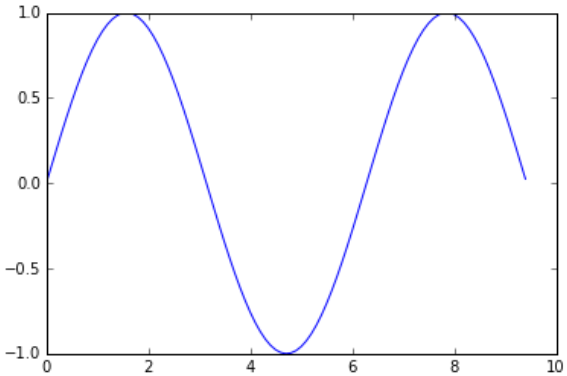
- Numpy is the core library for scientific computing in Python
- 基本数据结构是矩阵(array)，包括向量，矩阵和张量(多维矩阵)

```
import numpy as np

a = np.array([1, 2, 3])   # Create a rank 1 array
print(type(a))            # Prints "<class 'numpy.ndarray'>"
print(a.shape)            # Prints "(3,)"
print(a[0], a[1], a[2])   # Prints "1 2 3"
a[0] = 5                  # Change an element of the array
print(a)                  # Prints "[5, 2, 3]"

b = np.array([[1,2,3],[4,5,6]])    # Create a rank 2 array
print(b.shape)                     # Prints "(2, 3)"
print(b[0, 0], b[0, 1], b[1, 0])   # Prints "1 2 4"
```

Matplotlib

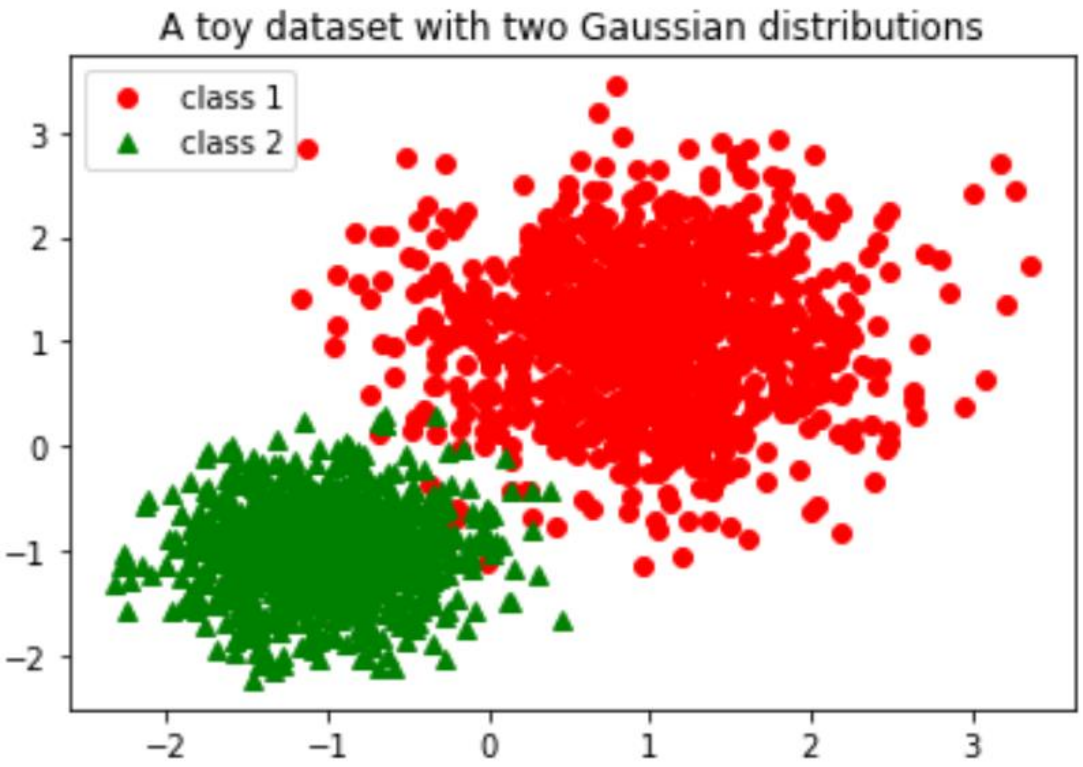
- Matplotlib is a plotting library to generate various types of figures

```
import numpy as np
import matplotlib.pyplot as plt

# Compute the x and y coordinates for points on a sine curve
x = np.arange(0, 3 * np.pi, 0.1)
y = np.sin(x)

# Plot the points using matplotlib
plt.plot(x, y)
plt.show()  # You must call plt.show() to make graphics appear.
```



更多例子：
http://cs231n.github.io/python-numpy-tutorial/

双高斯分布数据集



	feature 1	feature2	label
0	0.541078	0.084685	1.0
1	1.835211	0.414565	1.0
2	0.463693	0.600649	1.0
3	0.215385	0.838101	1.0
4	0.402105	1.605083	1.0
5	0.595065	0.079514	1.0
6	0.604953	0.947740	1.0
7	1.126134	-0.553929	1.0
8	0.860798	1.610948	1.0
9	1.414678	1.456316	1.0

作业

- 理论题：25%

1）写出最小二乘求解如下广义线性模型的 w, b 详细推到过程

$$y = e^{wx+b}$$

2）假设有三家工厂A, B, C共同生产一种台灯，他们产品占比和次品率分别如下：

工厂名（类别名）	产品占比(先验概率)	次品率(条件概率)
A	$0.35 = P(A)$	$0.015 = P(\text{次品} \mid A)$
B	$0.35 = P(B)$	$0.010 = P(\text{次品} \mid B)$
C	$0.30 = P(C)$	$0.020 = P(\text{次品} \mid C)$

某次随机抽检一个样品，该样品是次品概率有多大？如果该样品是次品，则它来自工厂A, B, C 的概率分别有多大？

作业

- 实践题：75%=25% x 3
  1) 实现线性分类器并在西瓜3.0数据集上用前80%训练、后20%测试时的精度
  2) 实现Naïve Bayes分类器并在西瓜3.0数据集上测试k=5重交叉验证精度
  3) 比较SVM使用不同（至少4种）核函数时，西瓜3.0数据集上用前80%训练、后20%测试的精度（可使用任意svm算法实现软件包）
- 附加题：(可不做，做对20%额外分)

  实现对数几率回归并在西瓜3.0上与线性分类器、NB和SVM做性能比较
- **注意：只允许实践题3)中SVM算法使用已有库中的实现，其他均需自己实现（发现作弊或抄袭，本次作业0分处理）。**

作业提交

- 文档：包括理论题的回答和实践题的**结果截图**
- 代码：包括所有运行需要的代码和数据
- 把以上两部分压缩在一个压缩包，发送到邮箱wangzihao33@sjtu.edu.cn
- 邮件标题格式：课程ML作业1_学号_姓名

- DDL：第5周周日（**10月16日**）23:59之前（以邮件收到时间为准）