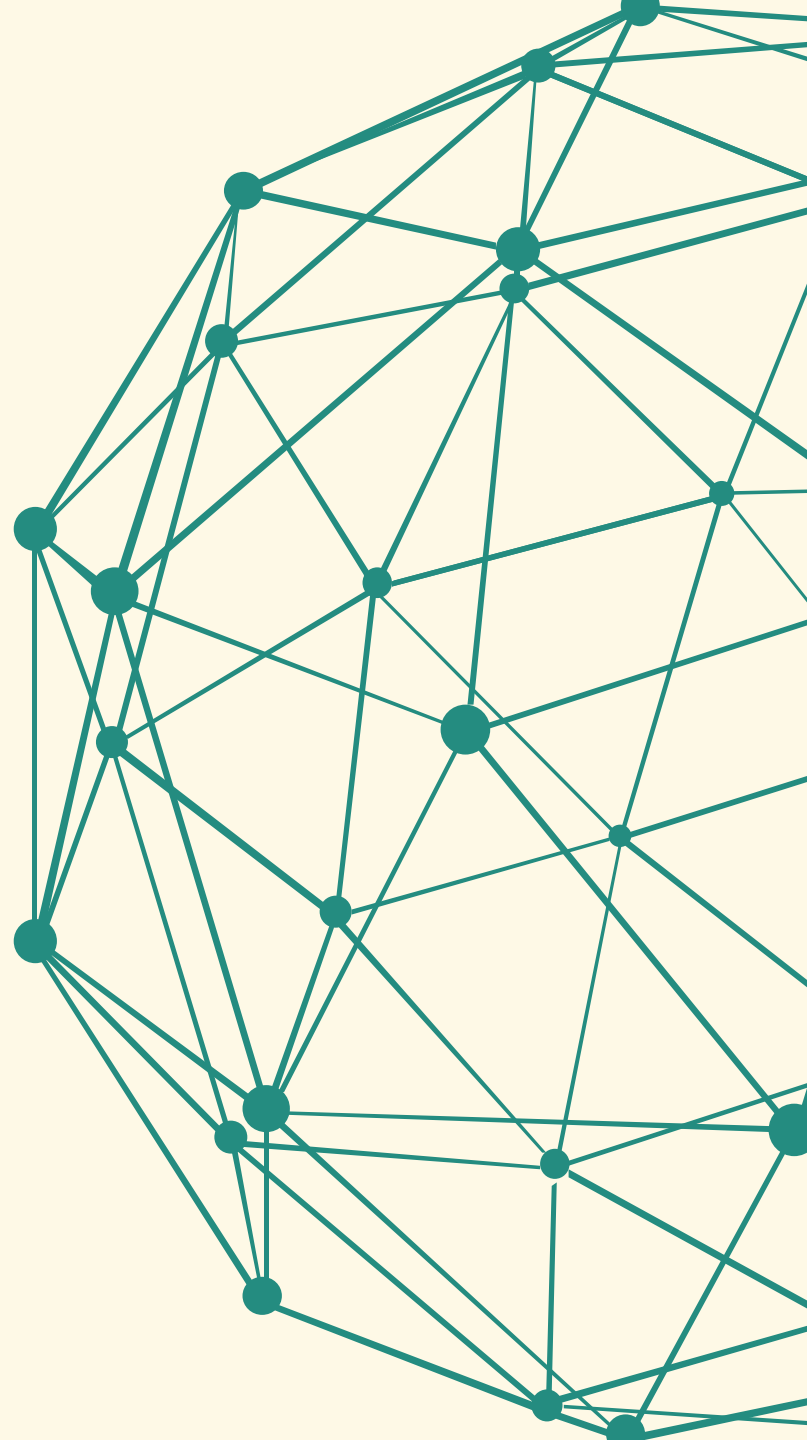


人工智能的数学基础

AI工程师讲座

基础 常用库 机器学习库





目录



01

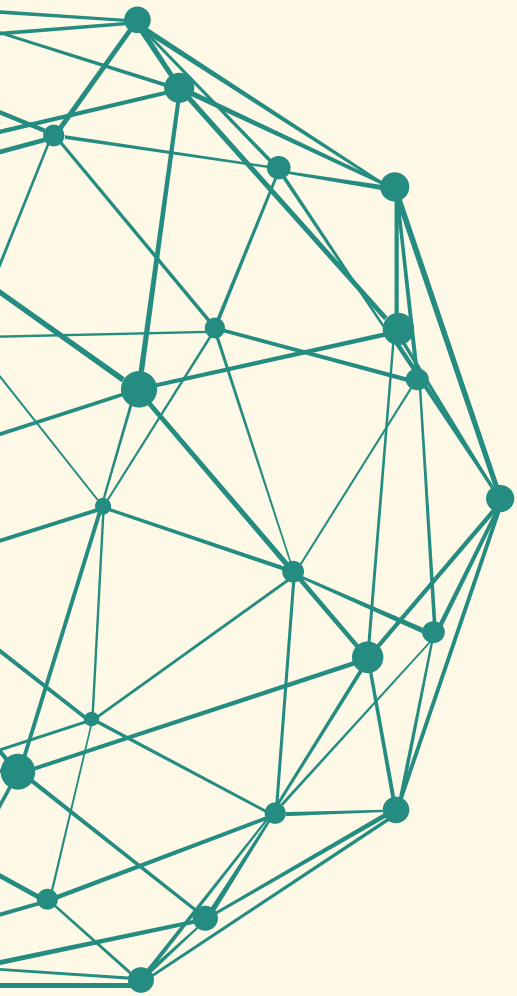
Python基础

02

Python常用库

03

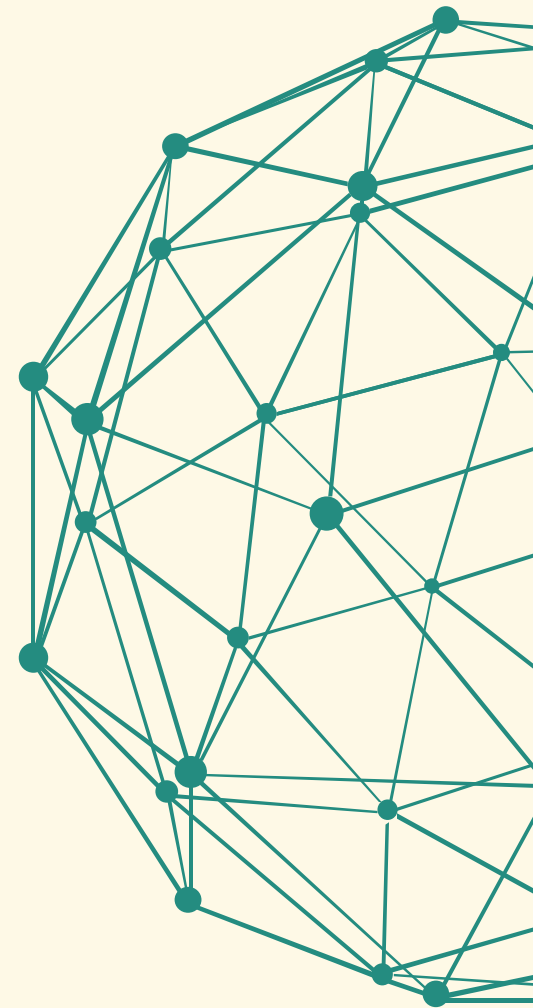
Python机器学习库

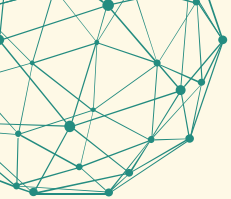


Part / 01

Python基础

BASE





Python基础

PROBABLITY TEHORY

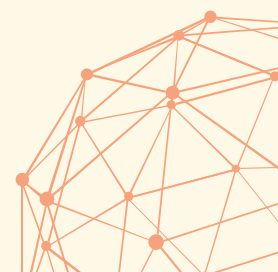
Python环境

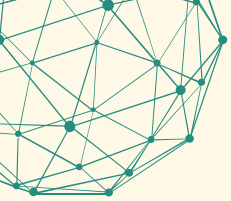
Python3.x

在几乎所有库函数都支持Python3.x的情况下，实在没有理由去推荐Python2.7。



集成了很多基础的库，使用起来较为方便。集成conda包管理工具。





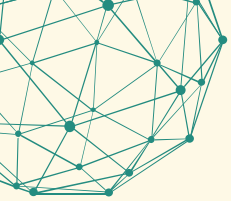
Python基础

PROBABLITY TEHORY

Python使用

Python与常用的面向对象的语言并无不同
唯一可见的不同在于区分域的方式在于文本所在的缩进位置。





元组

```
my_trup = 1, 2, 3
```

列表

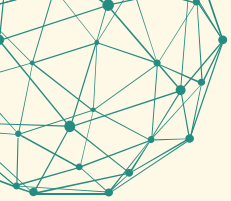
```
my_list = [1, 2, 3]
```

字典

```
my_dict = {"a":1, "b":2, "c":3}
```

元组是不可改变的





Python基础

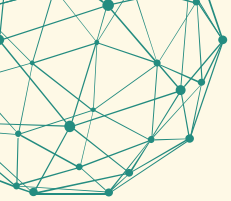
PROBABLITY TEHORY

Python使用

循环：

```
1 my_list=["a","b","c"]
2 #example1
3 for itr_list in my_list:
4     print(itr_list)
5 #example2
6 for itr_idx in range(len(my_list)):
7     print(itr_list[itr_idx])
```





Python基础

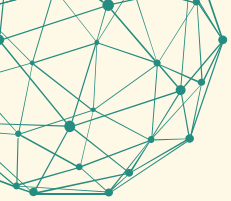
PROBABLITY TEHORY

Python使用

循环：

```
1 my_list=["a","b","c"]
2 #example1
3 for itr_list in my_list:
4     print(itr_list)
5 #example2
6 for itr_idx in range(len(my_list)):
7     print(itr_list[itr_idx])
```



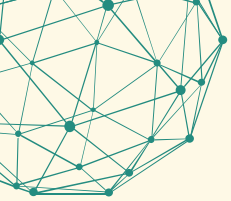


Python使用

循环：

```
1 my_list=["a","b","c"]
2 #example1
3 for itr_list in my_list:
4     print(itr_list)
5 #example2
6 for itr_idx in range(len(my_list)):
7     print(itr_list[itr_idx])
```





Python基础

PROBABLITY TEHORY

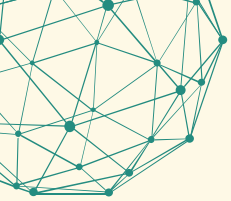
Python使用

判断

```
my_list = [1, 2, 3]
my_dict = {"a":1, "b":2, "c":3}

if 1 in my_list and 2 in my_list:
    print("yes")
print("a" in my_dict)
```





Python基础

PROBABLITY TEHORY

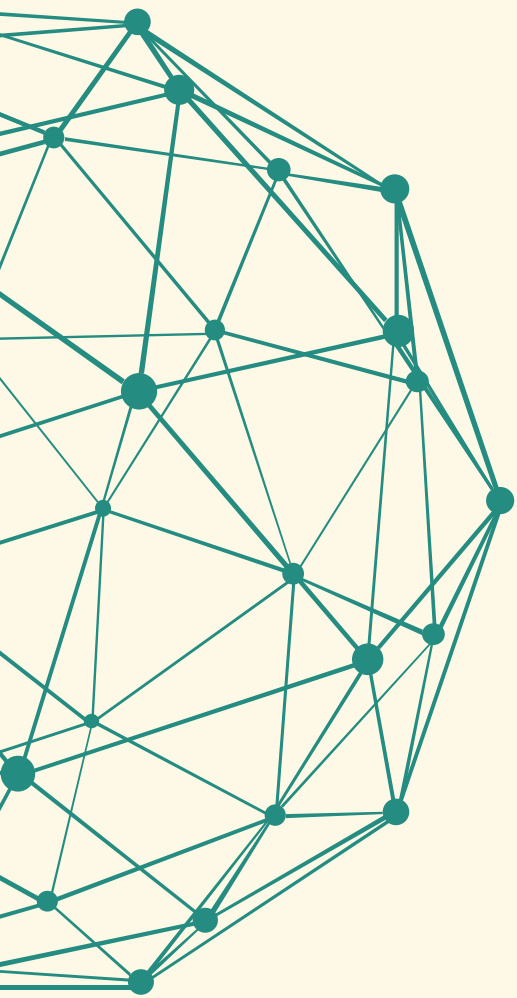
Python使用

import

```
import os
import numpy as np
from numpy import fft
import numpy.fft as nfft
```

脑 > 本地磁盘 (C:) > Anaconda3 > Lib > site-packages > numpy			
名称	修改日期	类型	大小
__pycache__	2017/7/3 8:32	文件夹	
compat	2017/7/3 8:28	文件夹	
core	2017/7/3 8:28	文件夹	
distutils	2017/7/3 8:28	文件夹	
doc	2017/7/3 8:28	文件夹	
f2py	2017/7/3 8:28	文件夹	
fft	2017/7/3 8:28	文件夹	
lib	2017/7/3 8:28	文件夹	
linalg	2017/7/3 8:28	文件夹	
ma	2017/7/3 8:28	文件夹	
matrixlib	2017/7/3 8:28	文件夹	
polynomial	2017/7/3 8:28	文件夹	
random	2017/7/3 8:28	文件夹	
testing	2017/7/3 8:28	文件夹	
tests	2017/7/3 8:28	文件夹	
config.py	2017/7/3 8:27	PY 文件	2 KB
init.py	2017/1/11 3:04	PY 文件	7 KB
_globals.py	2017/1/11 3:04	PY 文件	2 KB
_import_tools.py	2017/1/11 3:04	PY 文件	13 KB
add_newdocs.py	2017/1/11 3:04	PY 文件	219 KB
ctypeslib.py	2017/1/11 3:04	PY 文件	15 KB
dual.py	2017/1/11 3:04	PY 文件	2 KB
matlib.py	2017/1/11 3:04	PY 文件	10 KB
setup.py	2017/1/11 3:04	PY 文件	1 KB
version.py	2017/1/11 3:04	PY 文件	1 KB

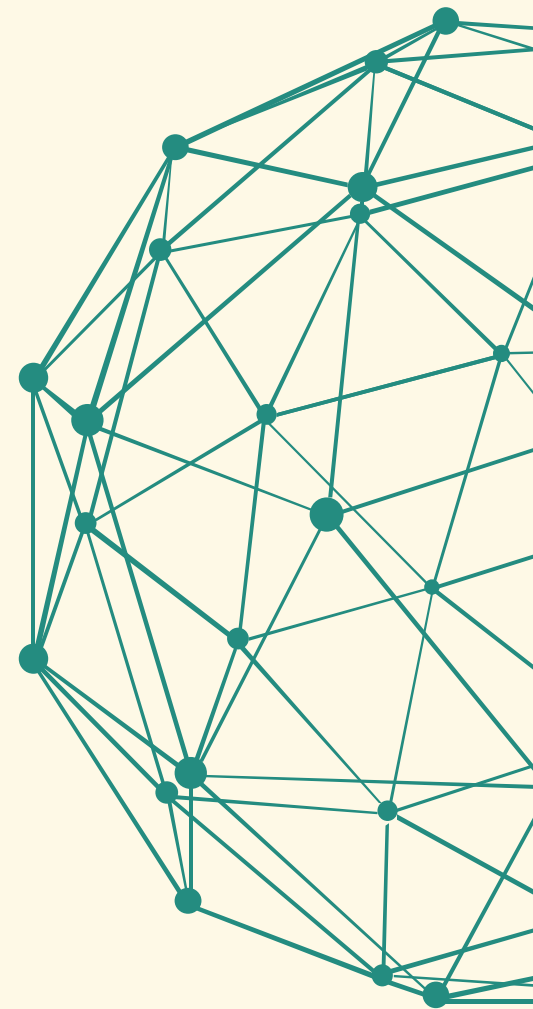


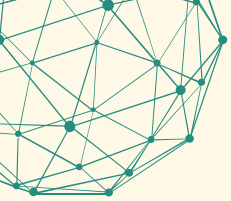


Part / 02

Python常用库

LIBRARY





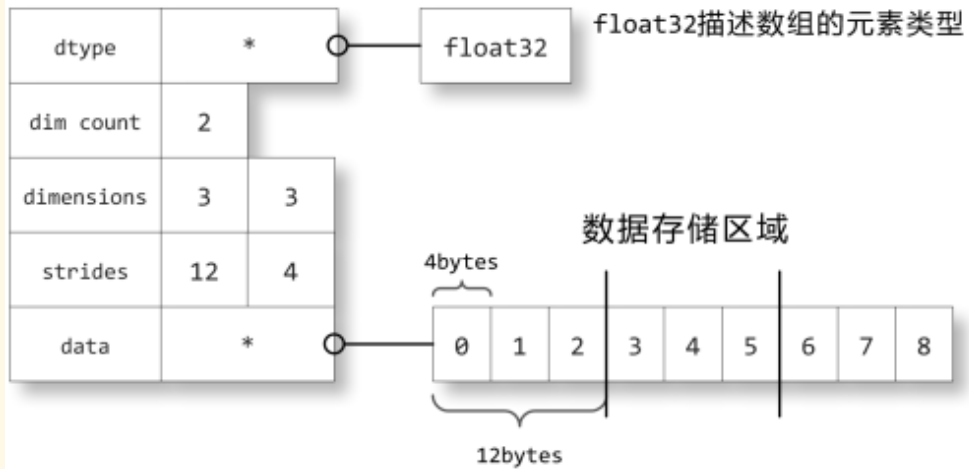
Python常用库

LIBRARY

numpy

numpy是Python的基础数学计算库，其思想与matlab相似。
numpy中矩阵常用array表示

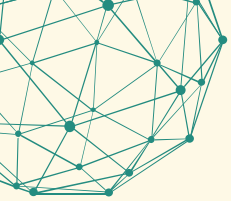
ndarray数据结构



```
np_array = np.ones([4,4])
print(np_array)
print(np_array[1:3,1:3])
```

```
[[ 1.  1.  1.  1.] [ 1.  1.  1.  1.] [ 1.  1.  1.  1.] [ 1.  1.  1.  1.]]
[[ 1.  1.] [ 1.  1.]]
```





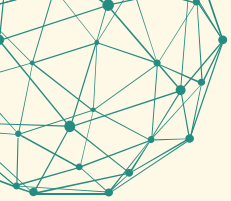
Python常用库

LIBRARY

scipy

scipy是Python的数学计算库，其在numpy的基础上加入了许多科学计算函数。包括常微分方程求解，信号处理，图像处理等。



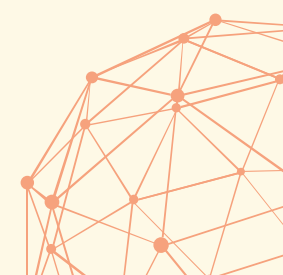
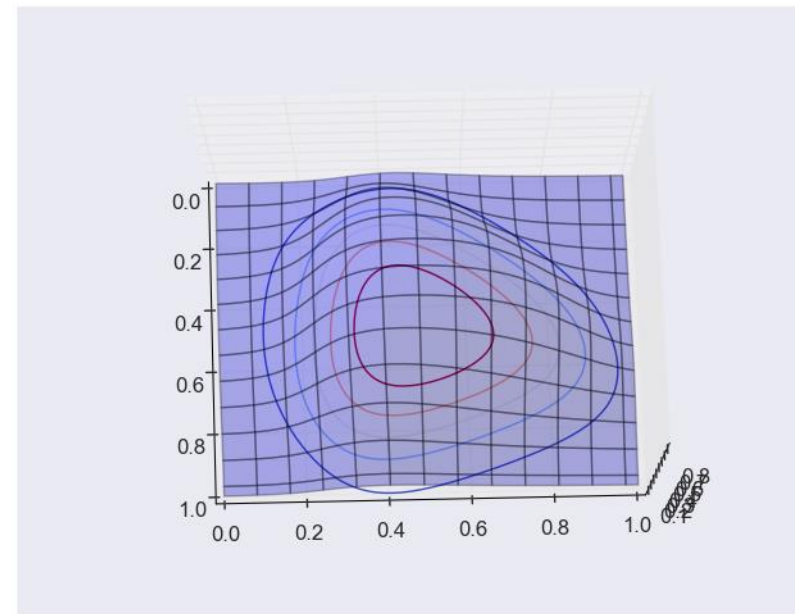


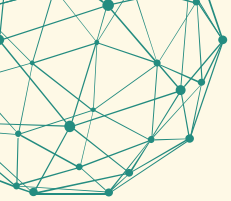
Python常用库

LIBRARY

matplotlib

matplotlib是
Python的最常用的
绘图库，其文档完备，
适合绘制简单的二维、
三维图形。



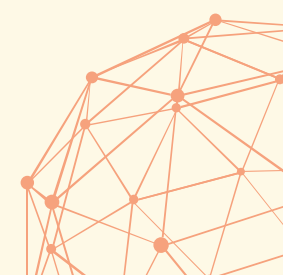
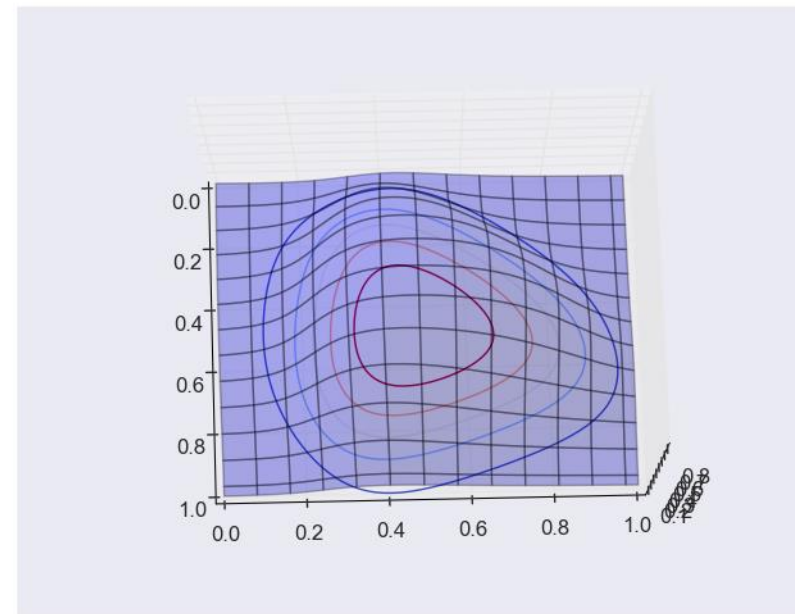


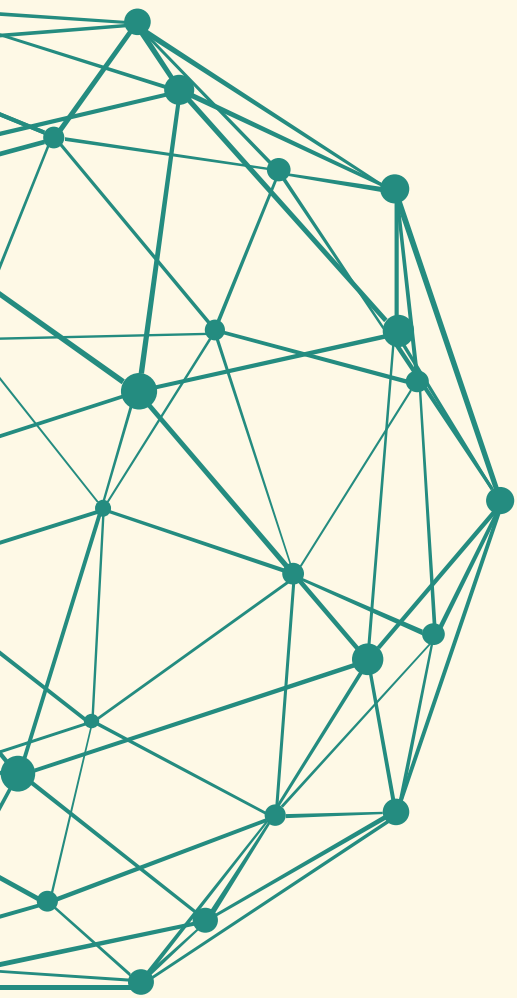
Python常用库

LIBRARY

matplotlib

matplotlib是
Python的最常用的
绘图库，其文档完备，
适合绘制简单的二维、
三维图形。

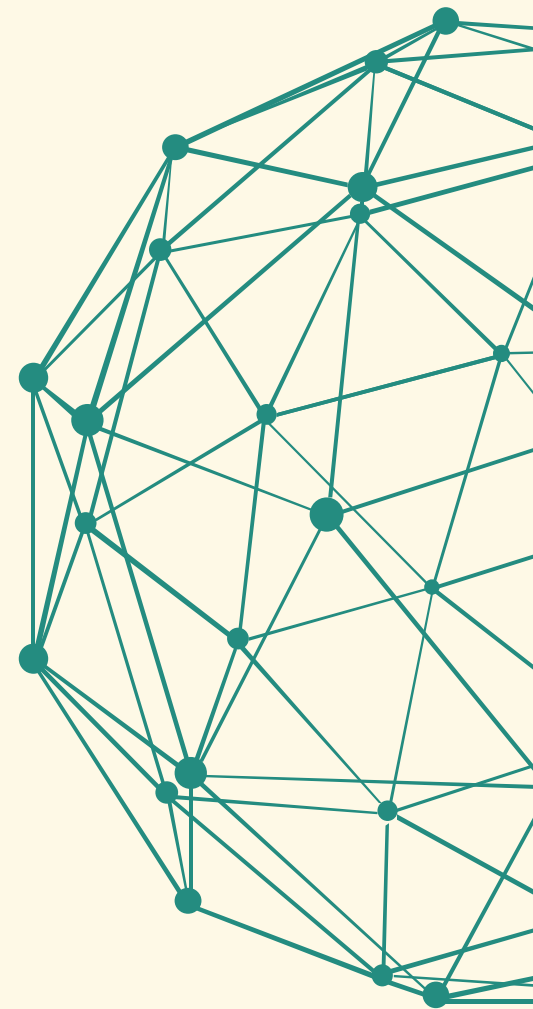


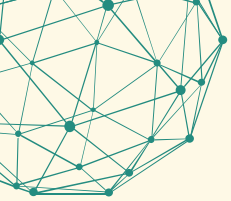


Part / 03

Python机器学习库

LIBRARY



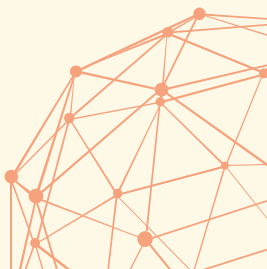


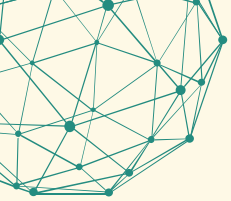
Python机器学习库

LIBRARY

pandas

pandas主要用于数据的清洗
和可视化工作。



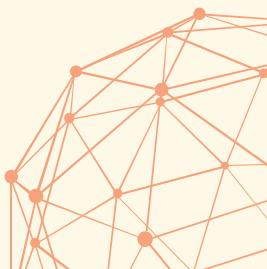


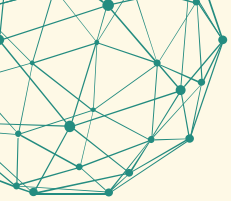
Python机器学习库

LIBRARY

scikit-learn

Python的机器学习库
其中包含了很多机器学习算法

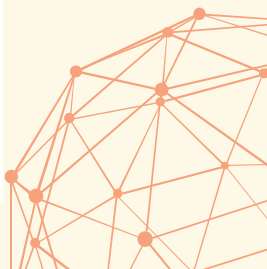
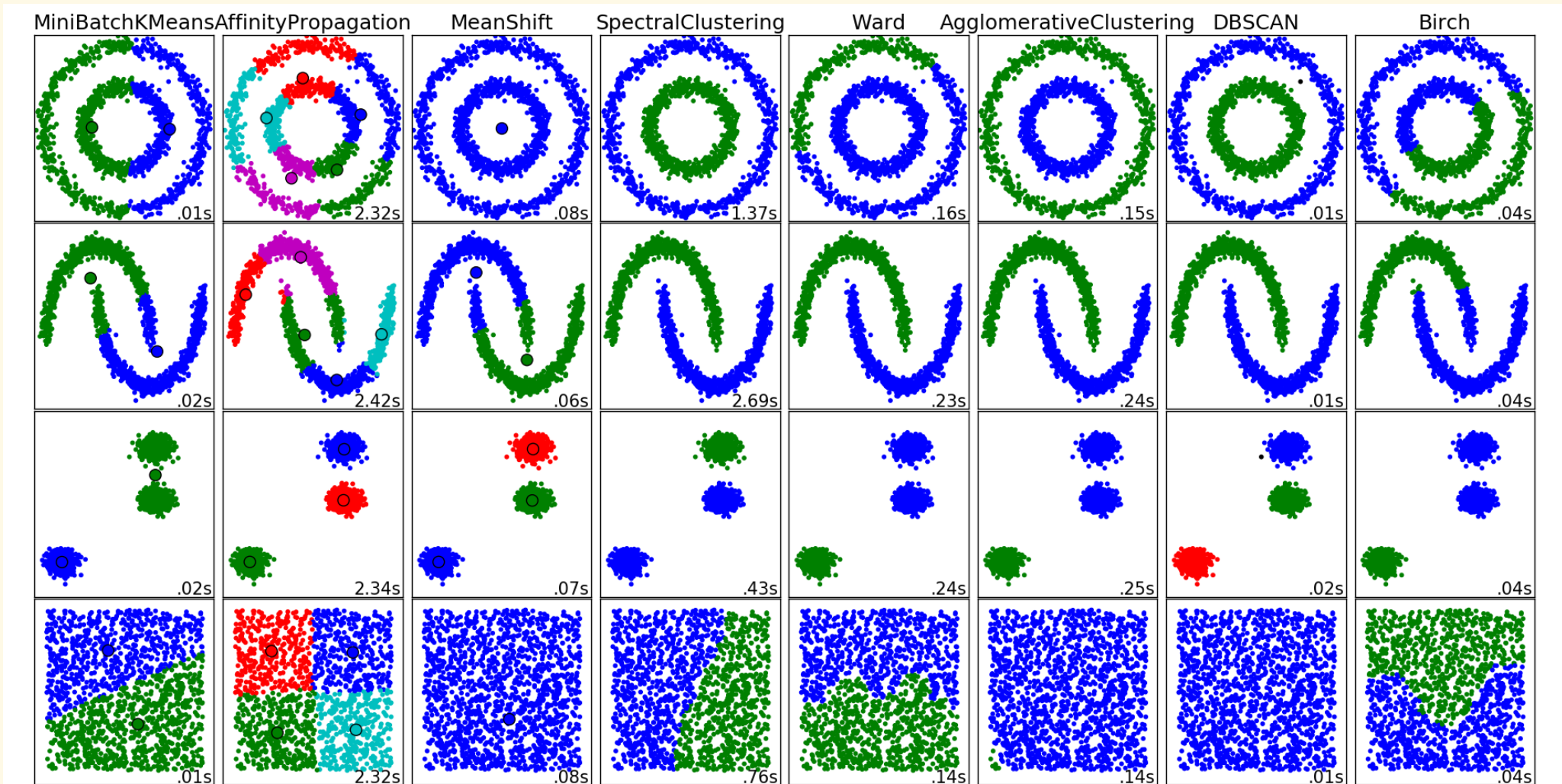


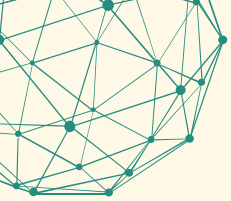


Python机器学习库

LIBRARY

scikit-learn

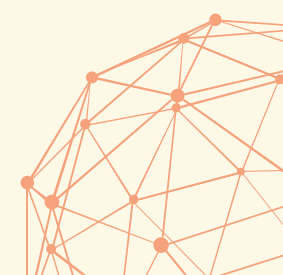
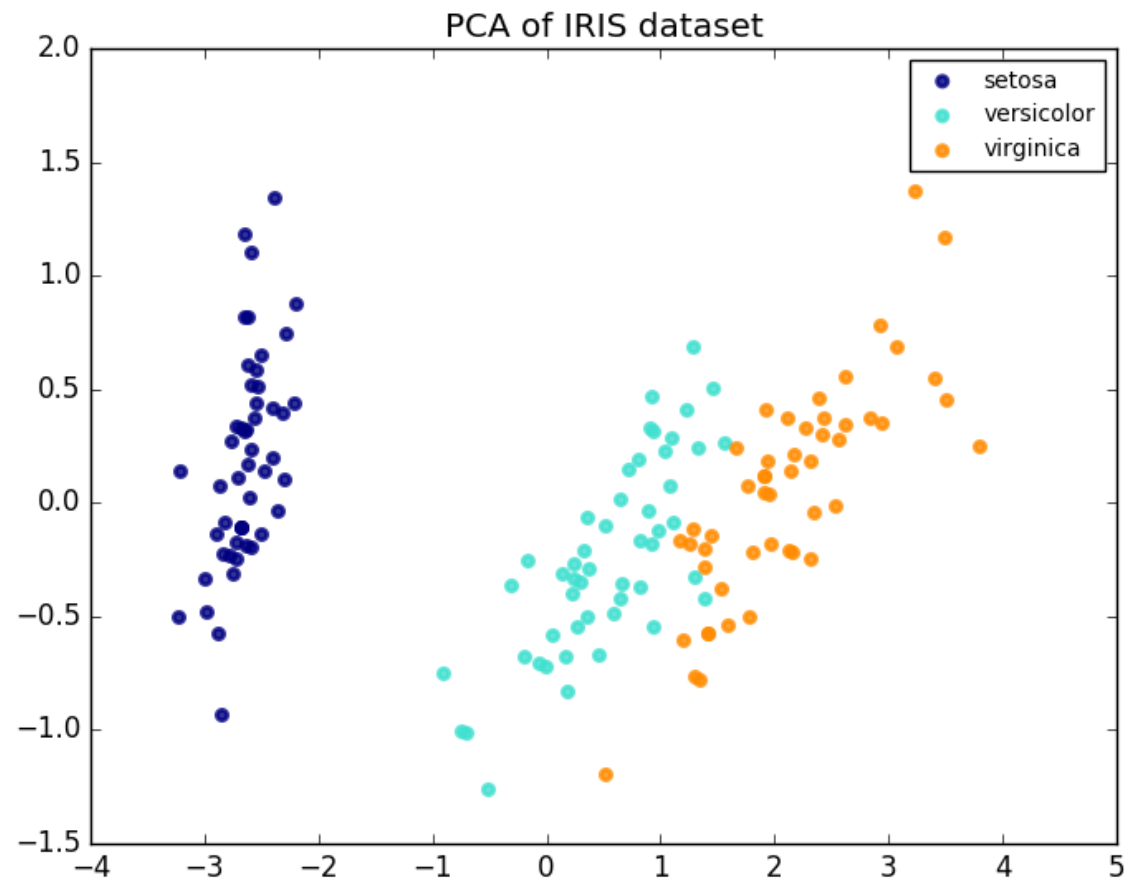


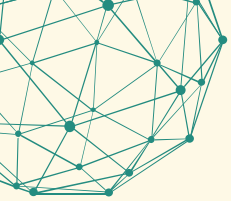


Python机器学习库

LIBRARY

scikit-learn

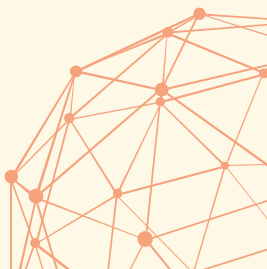
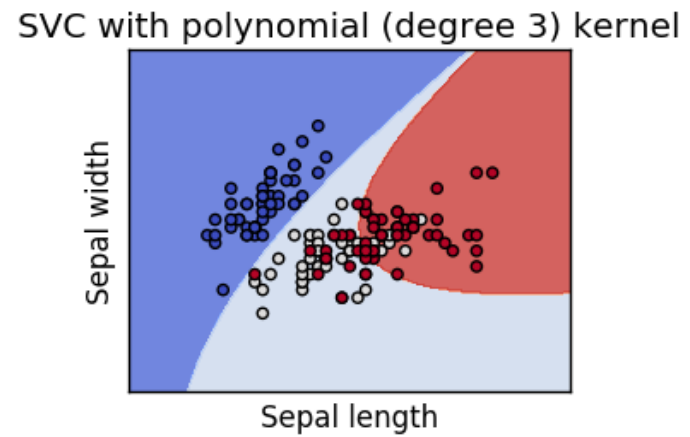
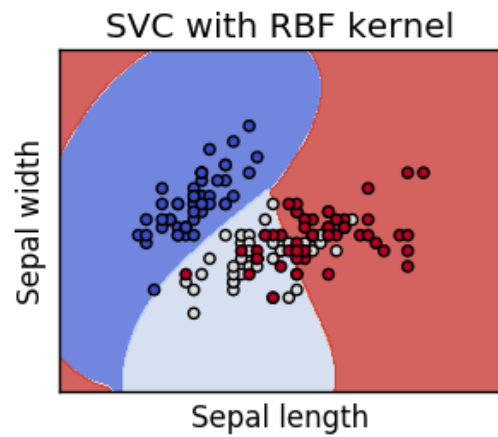
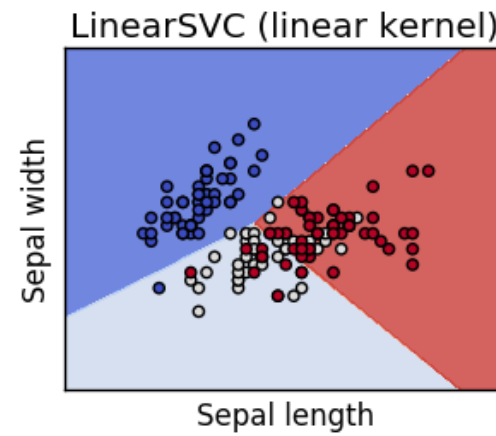
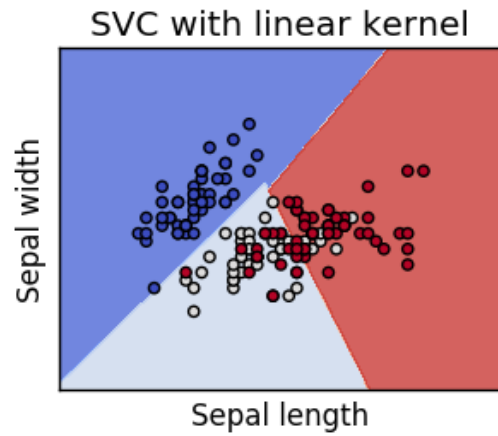




Python机器学习库

LIBRARY

scikit-learn



THANKS
AI工程师讲座

