第2讲 Python机器学习环境安装和配置

主讲教师：欧新宇

January 16, 2020

Outlines

- **Python环境的安装和配置**
  - 基本环境的安装 *(仅适用于Python基础学习)*
  - 标准环境的安装 *(适合于包括机器学习、数据分析、可视化等多个领域的开发和学习)*
- **机器学习必需库的安装和测试**
  - Numpy基础科学计算库
  - Scipy科学计算工具库
  - Pandas数据分析工具
  - Matplotlib绘图库
  - Scikit-learn机器学习库

一、 Python环境的安装和配置 – 极简版

- 优点：

  简单、原生

- 缺点：

  处理其他领域的应用需要手动安装大量第三方库

- 适用于：

  《程序设计基础（Python）》课程学习

一、 Python环境的安装和配置 – 极简版

- **安装Python环境**
  - 访问Python官网并下载最新版(https://www.python.org)
  - 双击并运行安装，勾选【Add Python 3.8 to PATH】

- **安装《程序设计基础（Python）》课程的基本库**
  - **安装jieba库**

```
>> pip install jieba
```

  - **安装wordcloud词云库**

```
>> pip install wordcloud
```

注意：因各种未知问题，有时可能需要多次安装才能连接上服务器。

一、 Python环境的安装和配置 – 标准版

- **Python集成环境Anaconda**
- **Visual Studio Code（VSCode)**
- **JupyterLab （Jupyter Notebook)**

- **操作方法：参考本课程Notebook**

=> http://ouxinyu.cn/Teaching/ml.html

=> 第02章 安装和配置 [Notebook]

二、 Python环境的测试

- **测试Python环境**
  - 方法一：打开IDLE交互环境，并执行下列指令进行测试
  - 方法二：打开IDLE文件编辑器，并输入下列代码并运行

```
[1]: print("Hello World!")

Hello World!
```

二、 Python环境的测试

更复杂一些的测试代码（请各位同学尝试）

```
names = input("请输入各个同学行业名称，行业名称之间用空格间隔（回车结束输入）：")
t = names.split()
d = {}
for c in range(len(t)):
    d[t[c]] = d.get(t[c], 0)+1
    ls = list(d.items())
ls.sort(key=lambda x: x[1], reverse=True)
for k in range(len(ls)):
    zy, num = ls[k]
    print("{}:{}".format(zy, num))
```

```
请输入各个同学行业名称，行业名称之间用空格间隔（回车结束输入）：英语 政治 语文 计算机 计算机 计算机
计算机:3
英语:1
政治:1
语文:1
```

0201SimpleExamples.py

三、机器学习必需库的安装和测试

- **Numpy基础科学计算库**

Numpy是Python中最基础的科学计算库，它的功能主要包括高位数组（Array）计算、线性代数计算、傅里叶变换以及产生伪随机数等。

Numpy是**机器学习库scikit-learn**的重要组成部分，因为机器学习库scikit-learn主要依赖于数组形式的数据来进行处理。

- 更多信息请参考：RUNOOB站的Numpy栏目：

https://www.runoob.com/numpy/numpy-tutorial.html

三、机器学习必需库的安装和测试

Numpy基础科学计算库

- **安装：** Numpy是Anaconda的内置库，无需额外安装
- **使用：** **import numpy as np.**
- 使用 import 关键字导入numpy 库，并使用 as 关键字将其简化为

[1]:
```
# 使用import关键字引入numpy库，为了简便使用缩写“np”来表示numpy库。
import numpy as np
# 定义一个变量 i，用于保存数组
i = np.array([[12,34,56],[78,90,11]])
```

[3]:
```
# 输出变量 i
print("i = \n{}".format(i))
```

```
i =
[[12 34 56]
 [78 90 11]]
```

访问课程[Notebook]查看更多实例

三、机器学习必需库的安装和测试

- **Scipy 科学计算工具集**

- **安装：Scipy**是Anaconda的内置库，无需额外安装
- **使用：from scipy import sparse**
- **对scipy的使用需要利用from关键字来引用其内部的子库**np。

[4]:
```
# 对scipy的使用需要利用from关键字来引用其内部的子库
import numpy as np
from scipy import sparse

# 使用numpy的eye()函数生成一个6行6列的对角矩阵
# 矩阵中对角线上的元素值为 1，其余元素为 0
matrix = np.eye(6)

# 将np数组转化为 CSR格式的Scipy稀疏矩阵 (sparse matrix)
sparse_matrix = sparse.csr_matrix(matrix)
```

三、机器学习必需库的安装和测试

- **Pandas 数据分析工具**

- **安装：Pandas**是Anaconda的内置库，无需额外安装
- **使用：import pandas as pd**
- 使用 import 关键字导入pandas 库，并使用 as 关键字将其简化。

Pandas是Python中进行数据分析的库，它具有以下功能

- 生成类似Excel表格式的数据表，并对数据进行修改操作；
- 从不同的数据源中获取数据，例如：SQL Server, Excel表格, CSV文件, Oracle等；
- 在不同的列中使用不同的数据类型，例如：整型，浮点型，字符串型等。
- 更多信息请参考“Pandas中文网”，URL：https://www.pypandas.cn/

三、机器学习必需库的安装和测试

Pandas 数据分析工具

```
# 使用import关键字引入pandas库，为了简便使用缩写“pd”来表示pandas库。
import pandas as pd

# 使用字典数据类型创建一个数据表，并用pandas库的DataFrame数据结构进行显示
data = {"姓名":["张飞","赵云","夏侯惇","太史慈"],
        "归属国":["蜀国","蜀国","魏国","吴国"],
        "年龄":["33","28","32","30"],
        "武力值":["98","97","94","92"]
}
data_frame = pd.DataFrame(data)
display(data_frame)
```

	姓名	归属国	年龄	武力值
0	张飞	蜀国	33	98
1	赵云	蜀国	28	97
2	夏侯惇	魏国	32	94
3	太史慈	吴国	30	92

三、机器学习必需库的安装和测试

Pandas 数据分析工具

如果想要把一些数据段进行排除，可以使用查询语句来实现。例如，不显示“魏国”的武将信息。

[8]:
```
# 使用“不等于 !=”操作符排除字段中包含特定值的数据
display(data_frame[data_frame.归属国 != "魏国"])
```

	姓名	归属国	年龄	武力值
0	张飞	蜀国	33	98
1	赵云	蜀国	28	97
3	太史慈	吴国	30	92

三、机器学习必需库的安装和测试

Matplotlib绘图库

matplotlib是Python中最重要的绘图库，它可以生成出版质量级别的图形，包括折线图、散点图、直方图等。

- **安装：Matplotlib**是Anaconda的内置库，无需额外安装
- **使用：import matplotlib as plt** *# 载入matplotlib库并简化命名*

  %matplotlib inline *# 实现在 jupyter 中实时绘图*

- 具体信息可以参考RUNOOB的matplotlib板块：
  https://www.runoob.com/w3cnote/matplotlib-tutorial.html
- 英语不错的同学，可以直接访问matplotlib项目页：http://matplotlib.org

三、机器学习必需库的安装和测试

Matplotlib绘图库

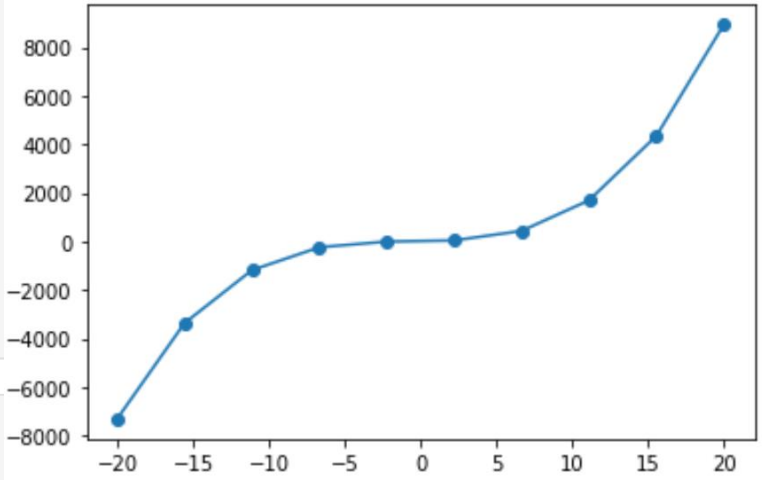
以下代码用于生成一个表达式为：$y = x^3 + 2x^2 + 6x + 5$ 的曲线图。

[2]:
```
# 通过inline指令，实现在Jupyter中的实时绘图功能
%matplotlib inline

# 使用import关键字引入matplotlib库，为了简便使用缩写“plt”来表示matplotlib库。
import matplotlib.pyplot as plt
import numpy as np

# 使用linspace()函数生成一个-20到20，元素个数为10的等差数列。
# 令数列中的值为 x，并根据表达式计算对应的 y值。
x = np.linspace(-20, 20, 10)
y = x**3 + 2*x**2 + 6*x + 5

#使用plot()函数绘制出曲线图
plt.plot(x, y, marker = "o")
```



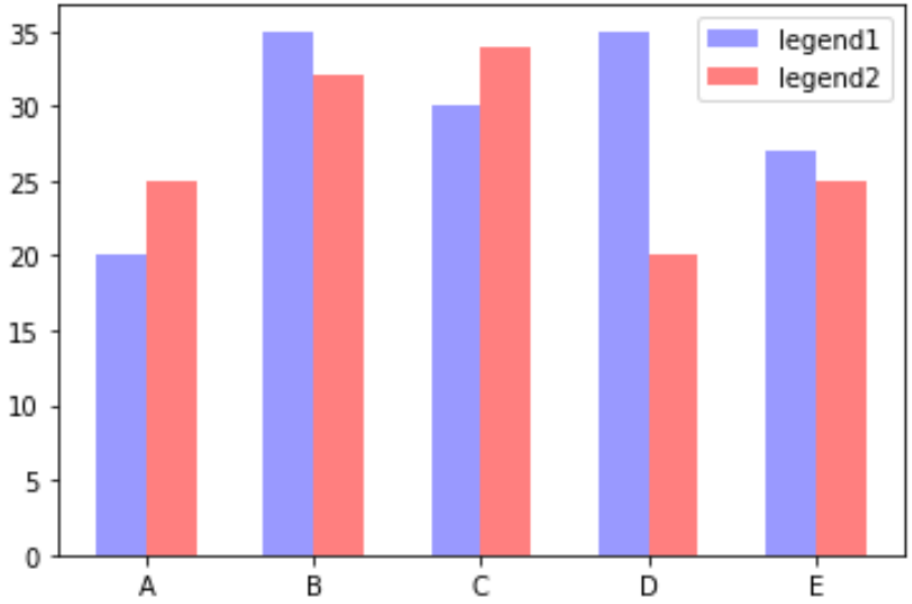
[5]:
```
print("x={}".format(x))
print("y={}".format(y))
```

```
x=[-20.         -15.55555556 -11.11111111  -6.66666667  -2.22222222
   2.22222222   6.66666667  11.11111111  15.55555556  20.        ]
y=[-7315.         -3368.4430727  -1186.4951989   -242.40740741
    -9.43072702    39.18381344   430.18518519  1690.3223594
  4346.34430727  8925.        ]
```

三、机器学习必需库的安装和测试

Matplotlib绘图库

以下代码为使用Matplotlib函数生成直方图



[10]:
```
# 通过inline指令，实现在Jupyter中的实时绘图功能
%matplotlib inline
import matplotlib.pyplot as plt

plt.figure(1)
x_index = np.arange(5)   #柱的索引
x_data = ('A', 'B', 'C', 'D', 'E')
y1_data = (20, 35, 30, 35, 27)
y2_data = (25, 32, 34, 20, 25)
bar_width = 0.3   #定义一个数字代表每个独立柱的宽度

# 使用 bar()函数定义柱状图的各个参数，依次包括：左偏移、高度、柱宽、透明度、颜色、图例
# 关于左偏移，不用关心每根柱的中心不中心，因为只要把刻度线设置在柱的中间就可以了
rects1 = plt.bar(x_index, y1_data, width=bar_width,alpha=0.4, color='b',label='legend1')
rects2 = plt.bar(x_index + bar_width, y2_data, width=bar_width,alpha=0.5,color='r',label='legend2')

# 使用 xticks() 函数设置x轴的刻度线
plt.xticks(x_index + bar_width/2, x_data)
plt.legend() #显示图例
plt.show()
```

三、机器学习必需库的安装和测试

Scikit-learn 机器学习库

scikit-learn是Python中最重要的机器学习模块之一。它基于Scipy库，在不同的领域中已经发展出大量基于Scipy的工具包，它们被统一称为Scikits，其中最著名的一个分支就是scikit-learn。它包含众多的机器学习算法，主要分为六大类：分类、回归、聚类、数据降维、模型选择和数据预处理。下列给出一个使用scikit-learn进行分类的简单例子。在下例中会随机生成包含300个具有两种属性数据的数据集，然后利用简单的SVM分类器实现分类。

- **安装：scikit-learn**是Anaconda的内置库，无需额外安装
- **使用：scikit-learn**库的使用比较复杂，后续的课程将会逐渐讲解。

三、机器学习必需库的安装和测试

Scikit-learn 机器学习库

[11]:
```
# 载入基础科学计算库 numpy
import numpy as np
# 载入可视化数据的模块 matplotlib
import matplotlib.pyplot as plt

# 从scikit-learn 库中载入预处理模块, 数据生成模块, 数据分割模块(划分为
# 训练集和测试集)和 支持向量机SVM的Support Vector Classifier分类模块
from sklearn.datasets.samples_generator import make_classification
from sklearn import preprocessing
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
```

三、机器学习必需库的安装和测试

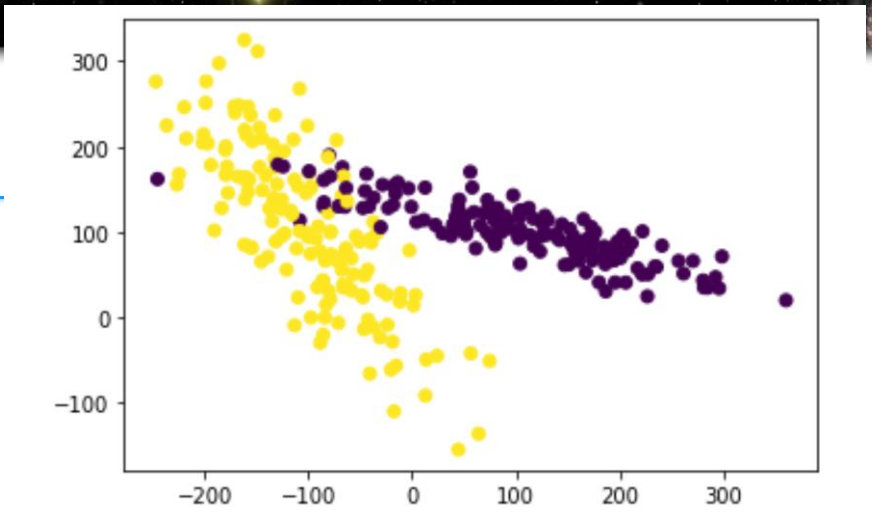
Scikit-learn 机器学习库

2. 生成数据集

[12]:
```
# 生成300个具有2种属性的数据
X, y = make_classification(n_samples=300, n_features=2,
                           n_redundant=0, n_informative=2,
                           random_state=22, n_clusters_per_class=1,
                           scale=100)
```

三、机器学习必需库的安装和测试

Scikit-learn 机器学习库



[13]:
```
#可视化数据
plt.scatter(X[:, 0], X[:, 1], c=y)
plt.show()

# 实现数据的正则化，可以有效提高分类精度
X = preprocessing.scale(X)

# 使用 train_test_split() 函数，将样本分割为 train训练集和 test测试集，
# 其中测试集数量为 30%
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)

# 定义SVC的核函数
clf = SVC(gamma = "auto")
# 使用fit()函数对模型进行训练
clf.fit(X_train, y_train)
# 使用 test测试集输出测试准确率
print(clf.score(X_test, y_test))
```

0.9666666666666667

欧老师的联系方式

读万卷书 行万里路 只为最好的修炼

QQ：14777591 (宇宙骑士)

Email：ouxinyu@alumni.hust.edu.cn

Tel：18687840023