
Machine Learning Course 2024 Spring: Prerequisites

1 实验内容

- Python 环境配置，包括开源 Python 发行版本 Anaconda 的安装以及基于网页的交互计算应用程序 Jupyter Notebook 的使用；
- 调用 sklearn 机器学习库中的机器学习模型以及数据集实现线性回归、逻辑回归、决策树、支持向量机、朴素贝叶斯、K 近邻 6 个经典分类器。

1.1 Python 环境配置

1.1.1 安装 Anaconda

进入 Anaconda 官网 (<https://www.anaconda.com/>)，选择适配自己电脑操作系统环境的 Anaconda 最新版本进行下载并一路默认安装在适当位置。

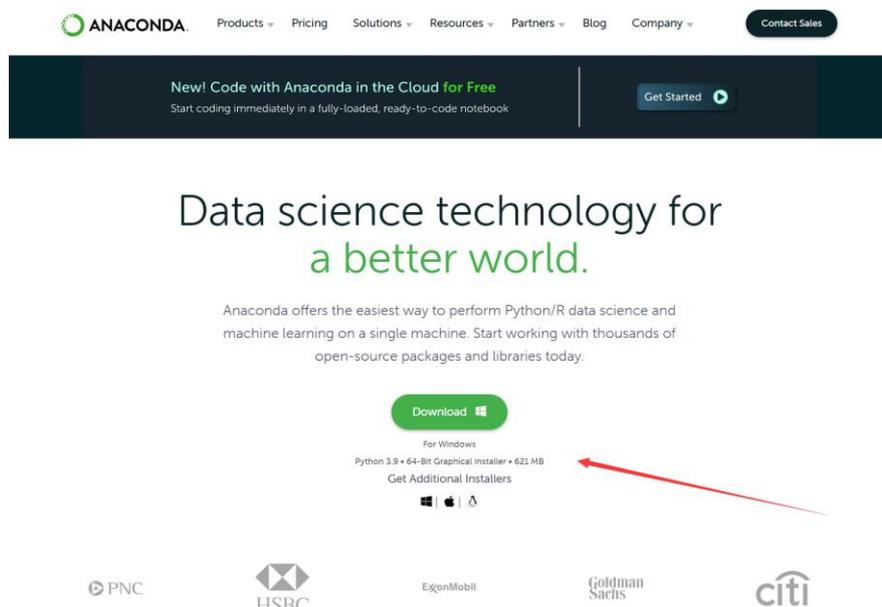


图 1 Anaconda 安装示意图

若要验证安装结果，可选以下任意方法：

- (1) “开始→Anaconda3→Anaconda Navigator”，若成功启动 Anaconda Navigator，显示如图 2 所示界面，则说明安装成功。
- (2) “开始→Anaconda3→右键点击 Anaconda Prompt→以管理员身份运行”，在 Anaconda Prompt 中输入 `conda list`，可以查看已安装的包名和版本号。若结果正常显示（如图 3 所示），则说明安装成功。

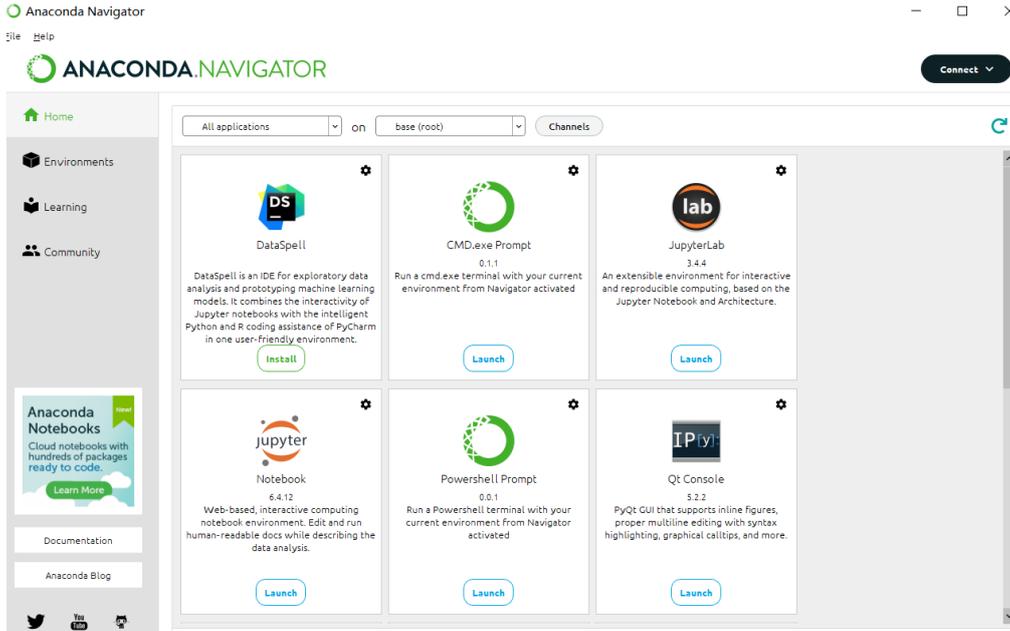


图 2 Anaconda Navigator 界面

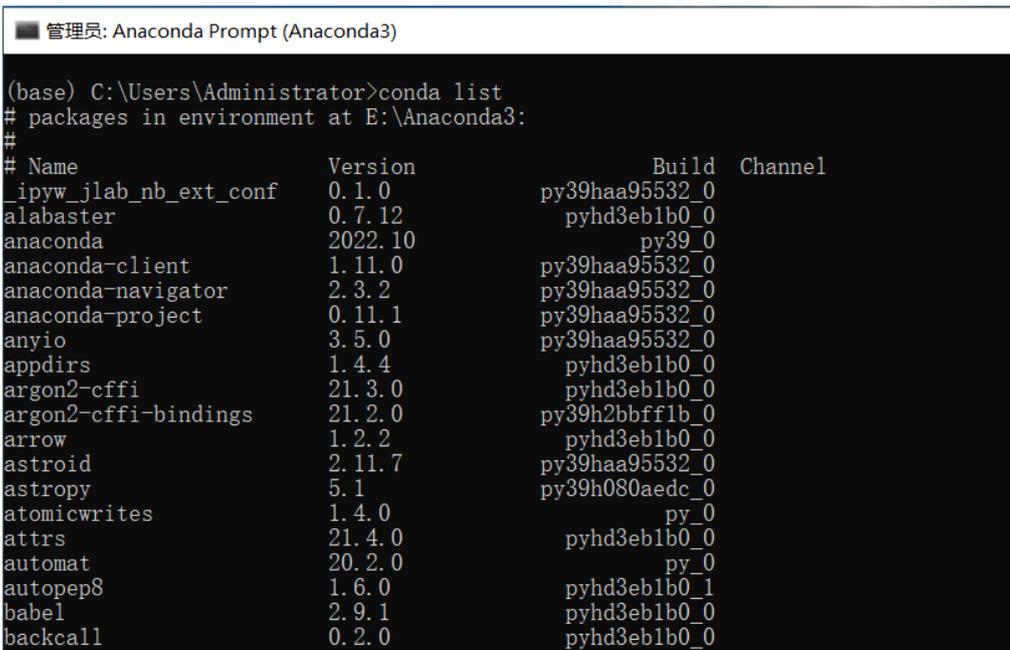


图 3 Anaconda Prompt 中输入 conda list 后的正常结果

1.1.2 Jupyter Notebook 的使用

在 Anaconda Navigator 中选择 Jupyter Notebook 或在 Anaconda Prompt 中输入 jupyter notebook 以启动 Jupyter Notebook。成功启动后，会在终端显示如图 4 所示的一系列 Notebook 的服务器信息，同时浏览器将会自动打开 Jupyter Notebook 页面，如图 5 所示。

（注：之后在 Jupyter Notebook 使用过程中进行的所有操作，请保持终端不要关闭。）

```
管理员: C:\Windows\system32\cmd.exe - jupyter notebook
Microsoft Windows [版本 10.0.19044.2604]
(c) Microsoft Corporation. 保留所有权利。

(base) C:\Users\Administrator>jupyter notebook
[I 2023-02-21 21:10:45.533 LabApp] JupyterLab extension loaded from E:\Anaconda3\lib\site-packages\jupyterlab
[I 2023-02-21 21:10:45.533 LabApp] JupyterLab application directory is E:\Anaconda3\share\jupyter\lab
[I 21:10:45.538 NotebookApp] The port 8888 is already in use, trying another port.
[I 21:10:45.541 NotebookApp] Serving notebooks from local directory: C:\Users\Administrator
[I 21:10:45.541 NotebookApp] Jupyter Notebook 6.4.12 is running at:
[I 21:10:45.541 NotebookApp] http://localhost:8889/?token=82693c0e285ee3c3920b0d97e8865bfa03281d85a00f4b3f
[I 21:10:45.541 NotebookApp] or http://127.0.0.1:8889/?token=82693c0e285ee3c3920b0d97e8865bfa03281d85a00f4b3f
[I 21:10:45.541 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 21:10:45.566 NotebookApp]

To access the notebook, open this file in a browser:
file:///C:/Users/Administrator/AppData/Roaming/jupyter/runtime/nbserver-1176-open.html
Or copy and paste one of these URLs:
http://localhost:8889/?token=82693c0e285ee3c3920b0d97e8865bfa03281d85a00f4b3f
or http://127.0.0.1:8889/?token=82693c0e285ee3c3920b0d97e8865bfa03281d85a00f4b3f
```

图 4 Jupyter Notebook 终端信息

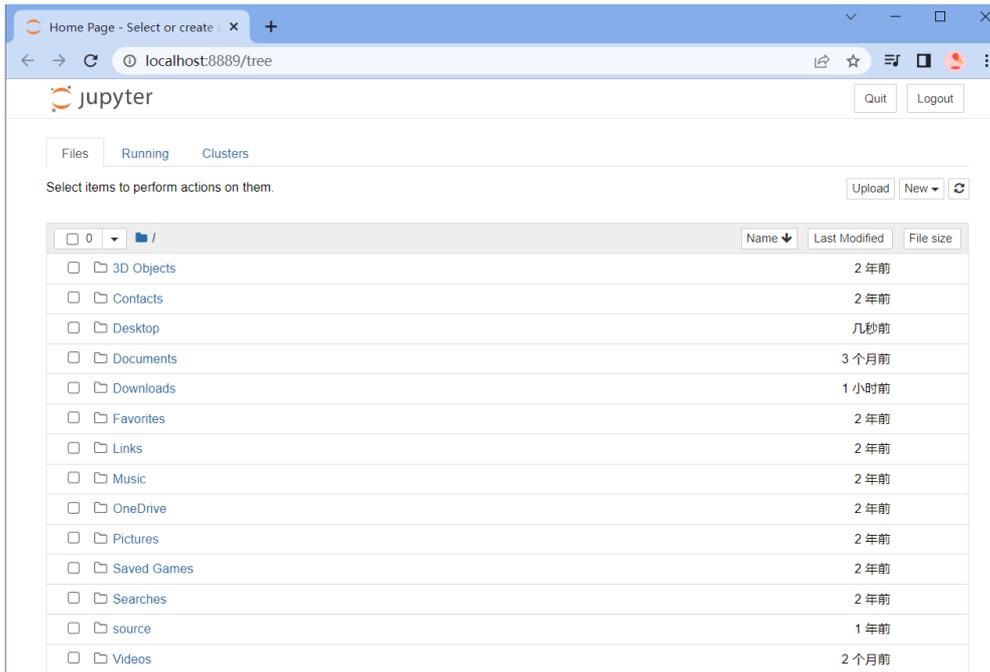


图 5 Jupyter Notebook 网页界面

在 Jupyter Notebook 界面中“New”的下拉列表中选择 Python 环境创建一个新 Notebook，如图 6 所示。

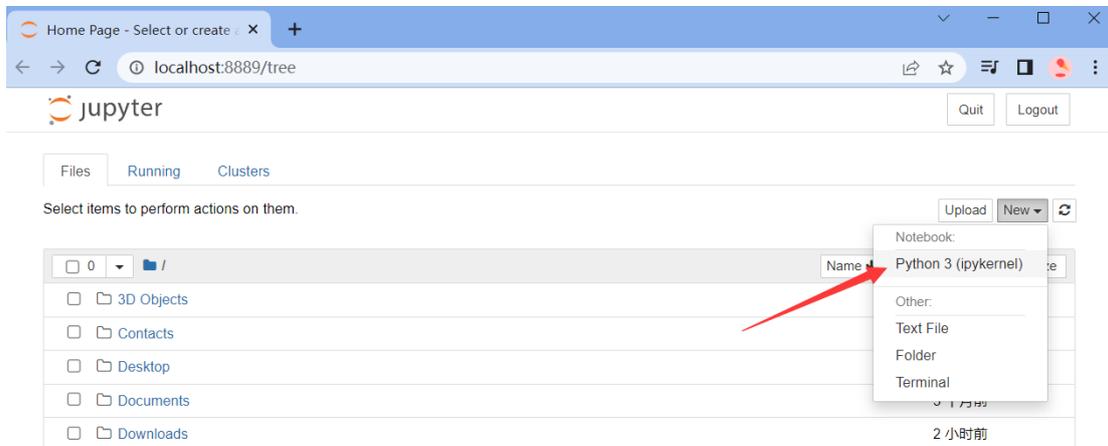


图 6 创建 Python 环境下的新 Notebook

接着在代码单元格内用 Python 编写并运行第一个程序输出“Hello World”，如图 7 所示。

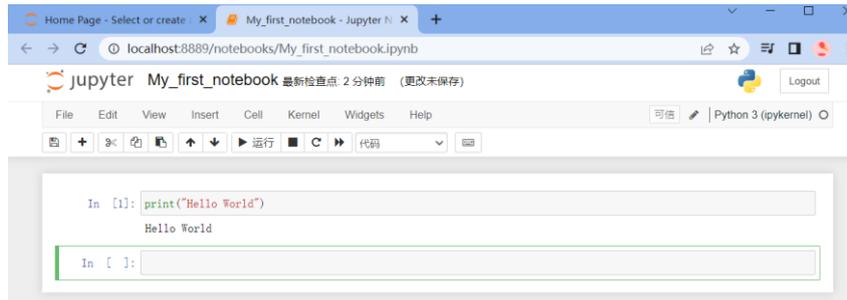


图 6 编写并运行输出“Hello World”

1.2 经典模型的实现

利用 sklearn 机器学习库实现线性回归、逻辑回归、决策树、支持向量机、朴素贝叶斯、K 近邻 6 个经典模型。线性回归代码示例如图 7 所示，结果如图 8 所示。

```
# 载入iris数据集，其中每个特征向量有四个维度，有三种类别
from sklearn import datasets
iris = datasets.load_iris()
print("The iris' target names:", iris.target_names)
x = iris.data # 样本特征向量
y = iris.target # 样本标记向量

# 待分类的两个样本
test_vector = [[1, -1, 2.6, -2], [0, 0, 7, 0.8]]

print("-----")

# 线性回归
from sklearn.linear_model import LinearRegression

linear = LinearRegression()
linear.fit(x,y) # 线性回归模型训练
print("LinearRegression: ", linear.predict(test_vector)) # 输出线性回归模型对测试样本的分类结果

# 逻辑回归
from sklearn.linear_model import LogisticRegression

# 决策树
from sklearn.tree import DecisionTreeClassifier

# 支持向量机
from sklearn.svm import SVC

# 朴素贝叶斯
from sklearn.naive_bayes import GaussianNB

# K近邻
from sklearn.neighbors import KNeighborsClassifier
```

图 7 线性回归模型实现示例

```
The iris' target names: ['setosa' 'versicolor' 'virginica']
-----
LinearRegression: [-0.50935816  2.27441208]
```

图 8 线性回归模型结果展示

2 实验要求

- 参照第一部分实验内容指引完成 Python 环境配置，学习 Jupyter Notebook 的

使用并利用 `sklearn` 机器学习库实现经典分类器；

(注：实验内容不限于第一部分中引导的内容，可在合理范围内拓展发挥。)