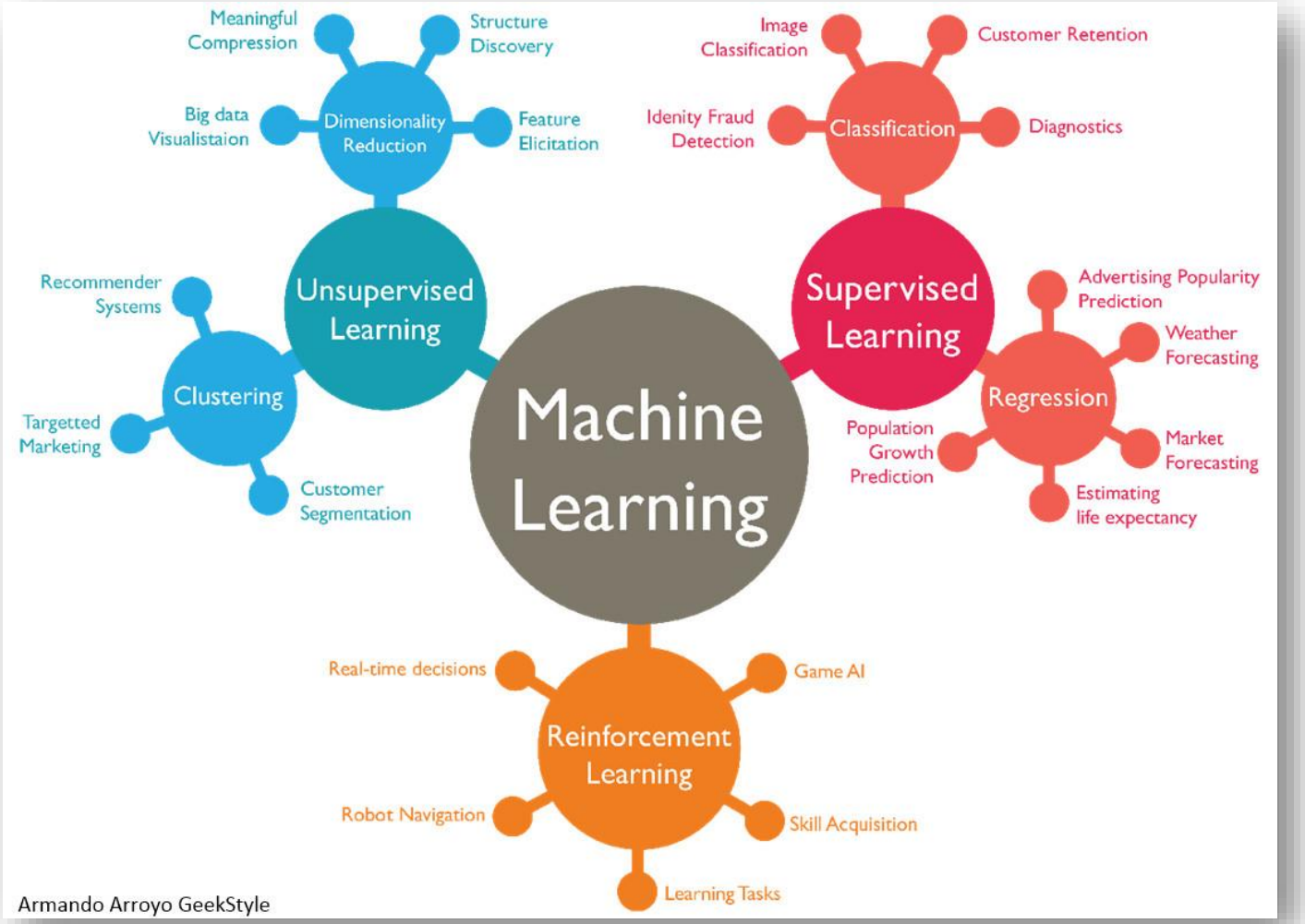
Python与金融数据挖掘(16)

文欣秀

wenxinxiu@ecust.edu.cn

机器学习

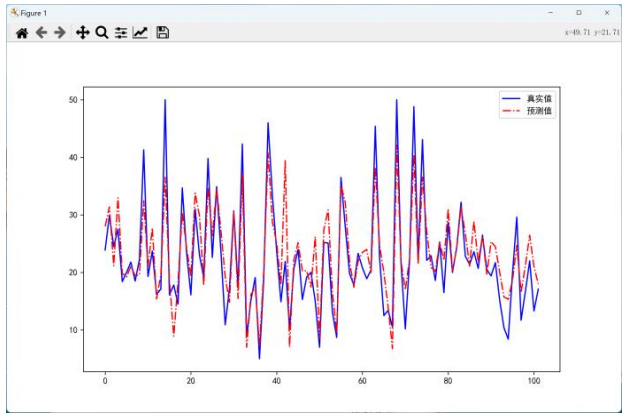

Machine Learning
Unsupervised Learning
Supervised Learning
Reinforcement Learning
Dimensionality Reduction
Clustering
Classification
Regression
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Recommender Systems
Targetted Marketing
Customer Segmentation
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks
Armando Arroyo GeekStyle

机器学习经典数据集

导入数据的函数名称	对应的数据集
load_boston()	波士顿房价数据集
load_breast_cancer()	乳腺癌数据集
load_iris()	鸢尾花数据集
load_diabetes()	糖尿病数据集
load_digits()	手写数字数据集
load_linnerud()	体能训练数据集
load_wine()	红酒品类数据集

波士顿房价

sklearn提供的波士顿房价数据集包含**506**条记录，**13**个特征指标，第**14列**通常为目标列房价。构建**回归模型**并训练模型，获取模型的预测结果，最后绘制折线图对比预测值和真实。



波士顿房价回归模型（一）

```
# （1）导入库
from sklearn.datasets import load_boston
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt
from matplotlib import rcParams
from warnings import simplefilter
simplefilter(action='ignore', category=FutureWarning)
```

波士顿房价回归模型（二）

```
# （2）加载数据集
boston=load_boston()
x=boston['data']
y=boston['target']
names=boston['feature_names’]

# 分割数据为训练集和测试集
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=22)
print('x_train前3行数据为：', x_train[0:3])
print('y_train前3行数据为：',y_train[0:3])
```

```
x_train前3行数据为： [[2.24236e+00 0.00000e+00 1.95800e+01 0.00000e+00 6.05000e-
01 5.85400e+00
  9.18000e+01 2.42200e+00 5.00000e+00 4.03000e+02 1.47000e+01 3.95110e+02
  1.16400e+01]
 [2.61690e-01 0.00000e+00 9.90000e+00 0.00000e+00 5.44000e-01 6.02300e+00
  9.04000e+01 2.83400e+00 4.00000e+00 3.04000e+02 1.84000e+01 3.96300e+02
  1.17200e+01]
 [6.89900e-02 0.00000e+00 2.56500e+01 0.00000e+00 5.81000e-01 5.87000e+00
  6.97000e+01 2.25770e+00 2.00000e+00 1.88000e+02 1.91000e+01 3.89150e+02
  1.43700e+01]]
y_train前3行数据为： [22.7 19.4 22. ]
```

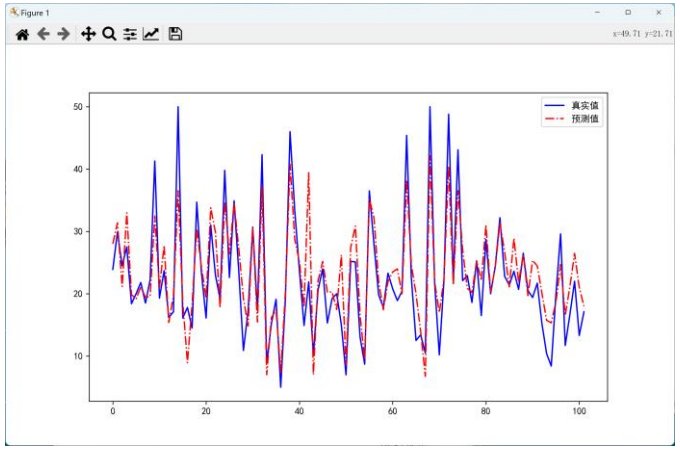
波士顿房价回归模型（三）

```
# （3）创建线性回归模型对象
lr=LinearRegression()

#使用训练集训练模型
lr.fit(x_train,y_train)

# （4）使用测试集获取预测结果
print("预测结果:",lr.predict(x_test[:5]))
```

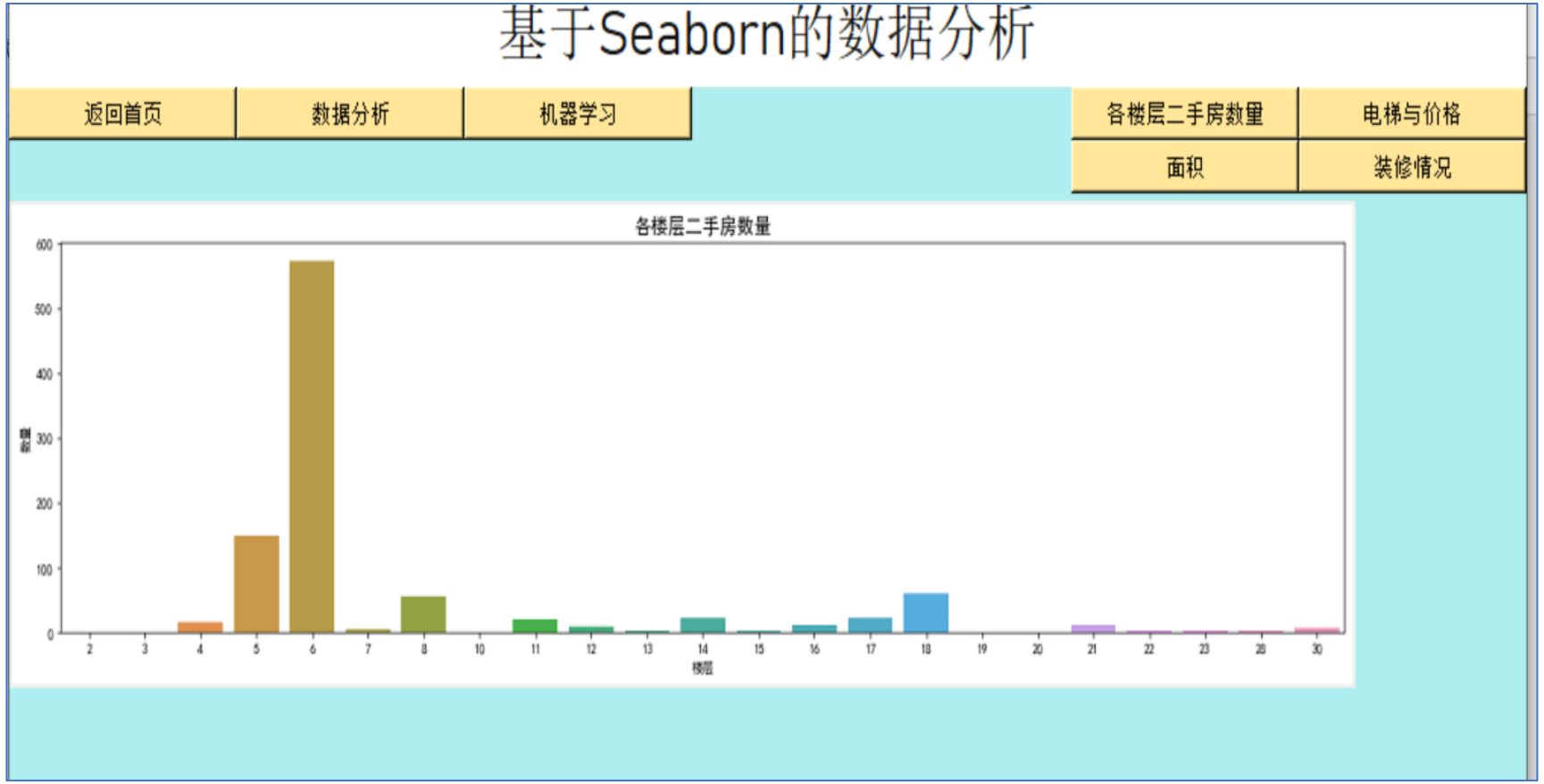
```
预测结果: [27.99617259 31.37458822 21.16274236 32.97684211 19.85350998]
```

波士顿房价回归模型（四）

```
# （5）绘图对比预测值和真实值
rcParams['font.sans-serif']='SimHei'
fig=plt.figure(figsize=(10,6))
y_pred=lr.predict(x_test)
plt.plot(range(y_test.shape[0]),y_test,color="blue",linestyle="-")
plt.plot(range(y_test.shape[0]),y_pred,color="red",linestyle="-.")
plt.legend(['真实值','预测值'])
plt.show()
```



学生创新作品


基于Seaborn的数据分析
返回首页
数据分析
机器学习
各楼层二手房数量
电梯与价格
面积
装修情况
各楼层二手房数量
600
500
400
300
200
100
0
数量
2
3
4
5
6
7
8
10
11
12
13
14
15
16
17
18
19
20
21
22
23
28
30
楼层

学生创新作品


127.0.0.1:8081
数据展示类型
首页
碳排放(国家)
碳排放(人均)
GDP(国家)
GDP(年份)
碳排放(国家对比)
GDP(国家对比)
GDP与碳排放分析
数据导出
各国GDP
与
碳排放展示与分析
图表展示类型
曲线
直方图
饼型图
小提琴图
散点图
矩阵图
折线图
词云

获奖作品分析

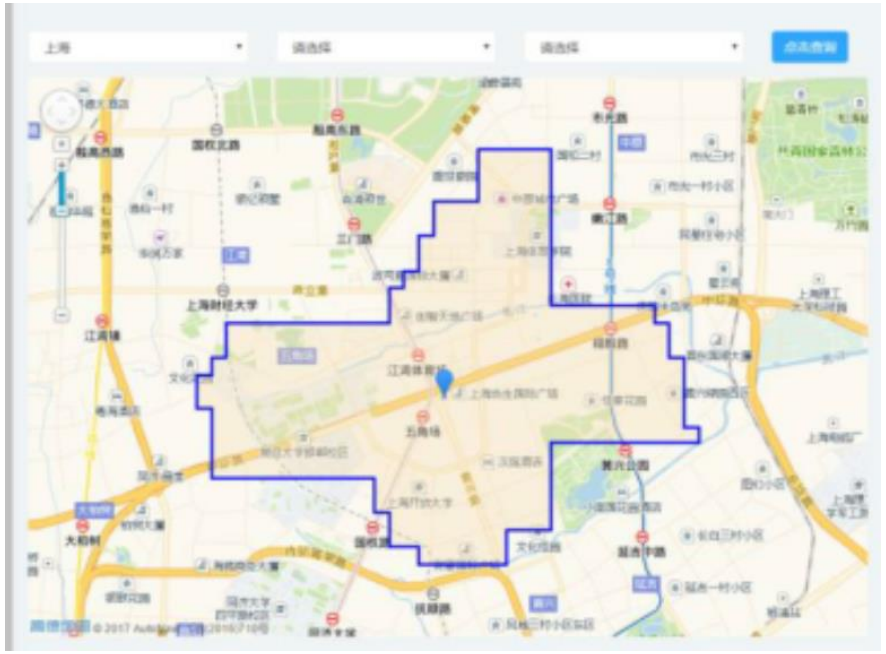
获 奖 证 书

荣获2017年（第九届）上海市大学生计算机应用能力大赛

一 等 奖

作品名称：面向智慧商圈的多源数据采集与分析系统
作　　者：王家辉 邢政 谈天
指导教师：文欣秀 过弋
单　　位：华东理工大学

上海市教育委员会



机器学习经典数据集

导入数据的函数名称	对应的数据集
load_boston()	波士顿房价数据集
load_breast_cancer()	乳腺癌数据集
load_iris()	鸢尾花数据集
load_diabetes()	糖尿病数据集
load_digits()	手写数字数据集
load_linnerud()	体能训练数据集
load_wine()	红酒品类数据集

鸢尾花问题K- Means模型（一）

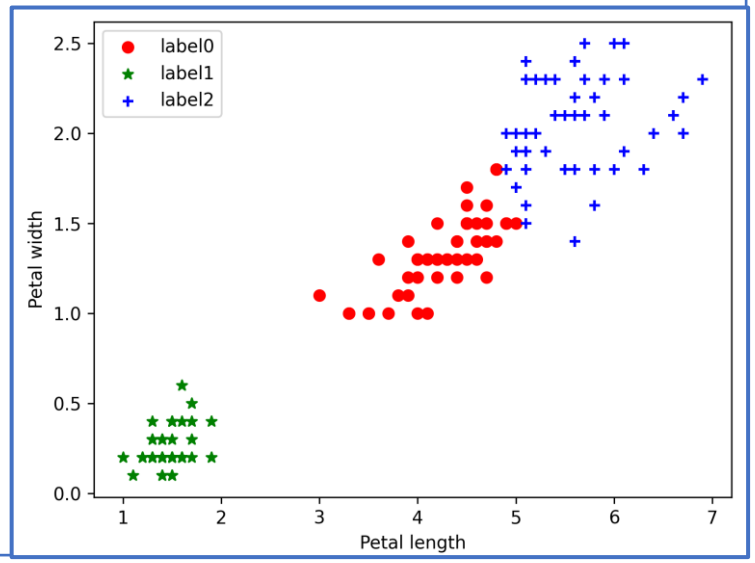
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn import datasets      # 导入数据集包
iris = datasets.load_iris()       # 加载数据据集
X = iris['data']                  #读出数据
```

鸢尾花问题K- Means模型（二）

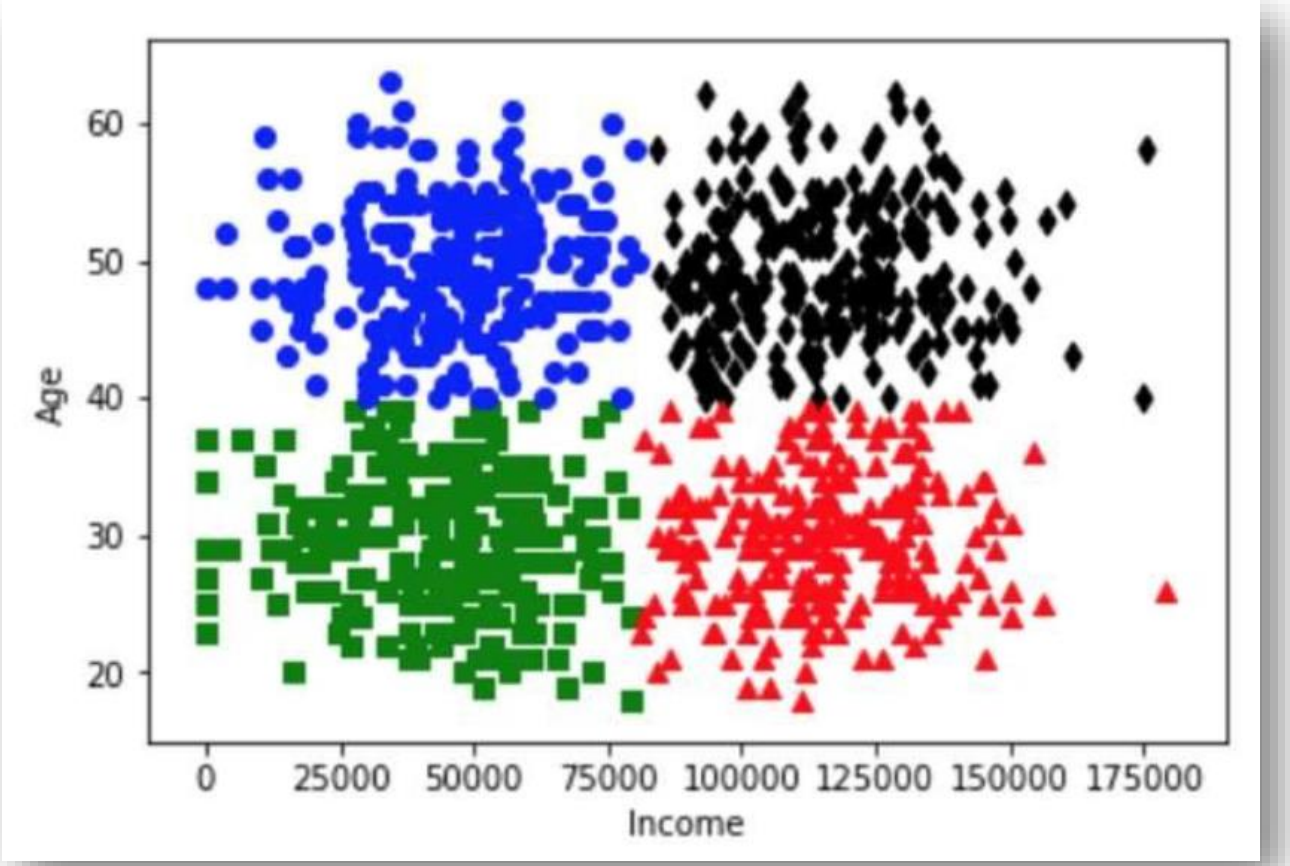
```
estimator = KMeans(n_clusters=3)#模型初始化
estimator.fit(X) #模型学习
label_pred = estimator.labels_ #获取聚类标签
x0 = X[label_pred == 0]
x1 = X[label_pred == 1]
x2 = X[label_pred == 2]
```

鸢尾花问题K- Means模型（三）

```
plt.scatter(x0[:,2], x0[:,3], c = "red", marker='o', label='label0')

plt.scatter(x1[:,2], x1[:,3], c = "green", marker='*', label='label1')

plt.scatter(x2[:,2], x2[:,3], c = "blue", marker='+', label='label2')

plt.xlabel('Petal length')

plt.ylabel('Petal width')

plt.legend(loc=2)

plt.show()
```



客户类型聚类分析


60
50
40
30
20
Age
0
25000
50000
75000
100000
125000
150000
175000
Income

机器学习步骤

一. 导入数据

二. 概述数据

三. 数据可视化

四. 评估算法

五. 实施预测

一. 导入数据

```
# 导入类库
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import SVC
# 导入数据
dataset = pd.read_csv("iris.csv")
```

二. 概述数据

```
#显示数据维度
print('数据维度: 行 %s，列 %s' % dataset. shape)

# 查看数据的前10行
print(dataset. head(10))

# 统计描述数据信息
print(dataset. describe())

# 种类分布情况
print(dataset. groupby('Species').size())
```

```
数据维度: 行 150，列 5
   Sepal_Length  Sepal_Width  Petal_Length  Petal_Width Species
0           5.1          3.5           1.4          0.2  setosa
1           4.9          3.0           1.4          0.2  setosa
2           4.7          3.2           1.3          0.2  setosa
3           4.6          3.1           1.5          0.2  setosa
4           5.0          3.6           1.4          0.2  setosa
5           5.4          3.9           1.7          0.4  setosa
6           4.6          3.4           1.4          0.3  setosa
7           5.0          3.4           1.5          0.2  setosa
8           4.4          2.9           1.4          0.2  setosa
9           4.9          3.1           1.5          0.1  setosa
```

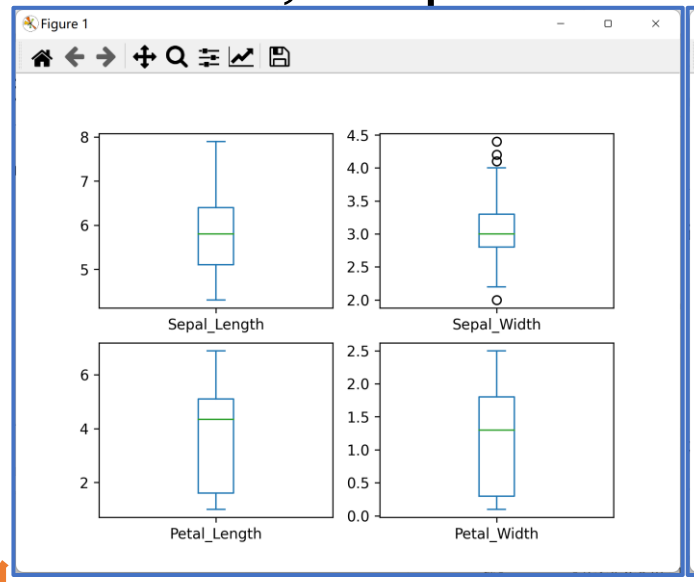
```
       Sepal_Length  Sepal_Width  Petal_Length  Petal_Width
count    150.000000   150.000000    150.000000   150.000000
mean       5.843333     3.057333      3.758000     1.199333
std        0.828066     0.435866      1.765298     0.762238
min        4.300000     2.000000      1.000000     0.100000
25%        5.100000     2.800000      1.600000     0.300000
50%        5.800000     3.000000      4.350000     1.300000
75%        6.400000     3.300000      5.100000     1.800000
max        7.900000     4.400000      6.900000     2.500000
Species
setosa        50
versicolor    50
virginica     50
dtype: int64
```

三. 数据可视化

箱线图

dataset. plot(kind='box', subplots=True, layout=(2,2), sharex=False, sharey=False)

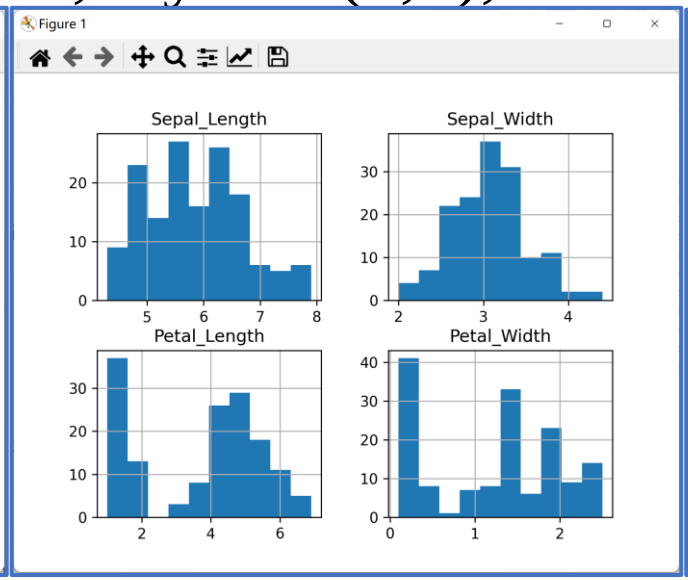
plt. show()



直方图

dataset. hist()

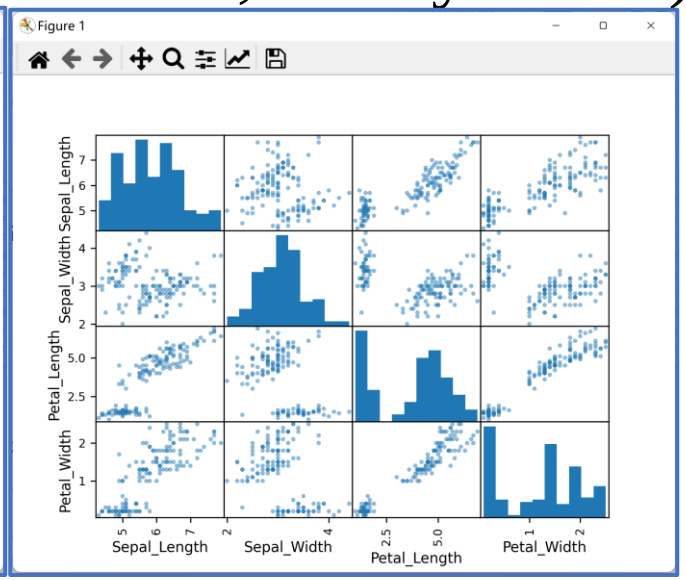
plt. show()



散点矩阵图

pd. plotting. scatter_matrix(dataset)

plt. show()



四. 评估算法（1）

```
# 分离数据集
array = dataset. values
X = array[:, 0:4]
Y = array[:, 4]
validation_size = 0.2  # 80%训练集，20%验证数据集
seed = 7 #随机数种子
X_train, X_validation, Y_train, Y_validation = \
    train_test_split(X, Y, test_size=validation_size, random_state=seed)
```

四. 评估算法（2）

```
# 算法审查
from warnings import simplefilter
simplefilter(action='ignore', category=FutureWarning) # 忽略所有警告
models = {}
models['LR'] = LogisticRegression(random_state=0, max_iter=1000)
models['KNN'] = KNeighborsClassifier()
models['CART'] = DecisionTreeClassifier()
models['NB'] = GaussianNB()
models['SVM'] = SVC()
```

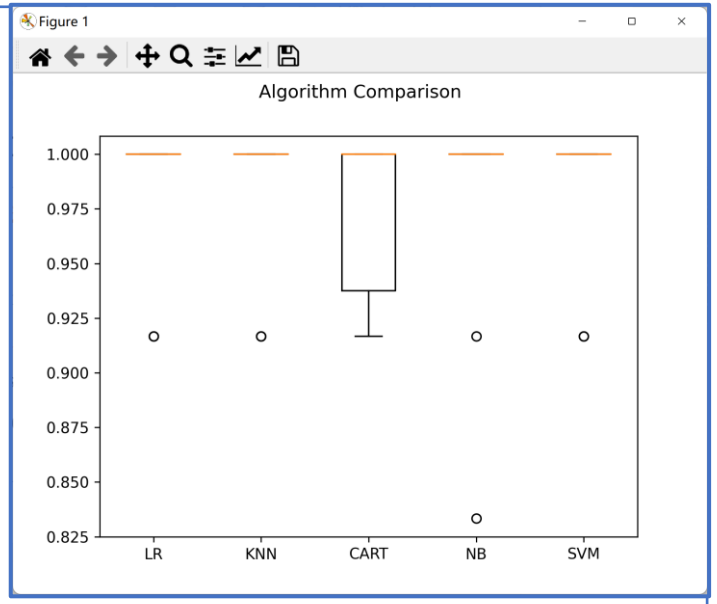
四. 评估算法（3）

```
# 评估算法
results = []
for key in models:
    kfold = KFold(n_splits=10)
    cv_results = cross_val_score(models[key], X_train, Y_train,
cv=kfold, scoring='accuracy')
    results.append(cv_results)
    print('%s: %f (%f)' %(key, cv_results.mean(), cv_results.std()))
```

```
LR: 0.983333 (0.033333)
KNN: 0.983333 (0.033333)
CART: 0.975000 (0.038188)
NB: 0.975000 (0.053359)
SVM: 0.983333 (0.033333)
```

四. 评估算法（4）

```
# 箱线图比较算法
fig = plt.figure()
fig.suptitle('Algorithm Comparison')
ax = fig.add_subplot(111)
plt.boxplot(results)
ax.set_xticklabels(models.keys())
plt.show()
```



五. 实施预测

#使用评估数据集评估算法

```
svm = SVC()

svm.fit(X=X_train, y=Y_train)

predictions = svm.predict(X_validation)

print(accuracy_score(Y_validation, predictions))

print(confusion_matrix(Y_validation, predictions))

print(classification_report(Y_validation, predictions))
```

```
[[ 7  0  0]
 [ 0 10  2]
 [ 0  2  9]]
              precision    recall  f1-score   support

      setosa       1.00      1.00      1.00         7
  versicolor       0.83      0.83      0.83        12
   virginica       0.82      0.82      0.82        11

    accuracy                           0.87        30
   macro avg       0.88      0.88      0.88        30
weighted avg       0.87      0.87      0.87        30
```

Python应用领域

文本分析： Jieba、Nltk…

科学计算： Numpy、SciPy…

数据分析： Pandas、Matplotlib…

机器学习： Scikit-Learn、Keras…

深度学习： Pytorch、Mindspore、PaddlePaddle…

常用的深度学习框架

PyTorch： 由Facebook的团队开发，并于2017年在GitHub 上开源。

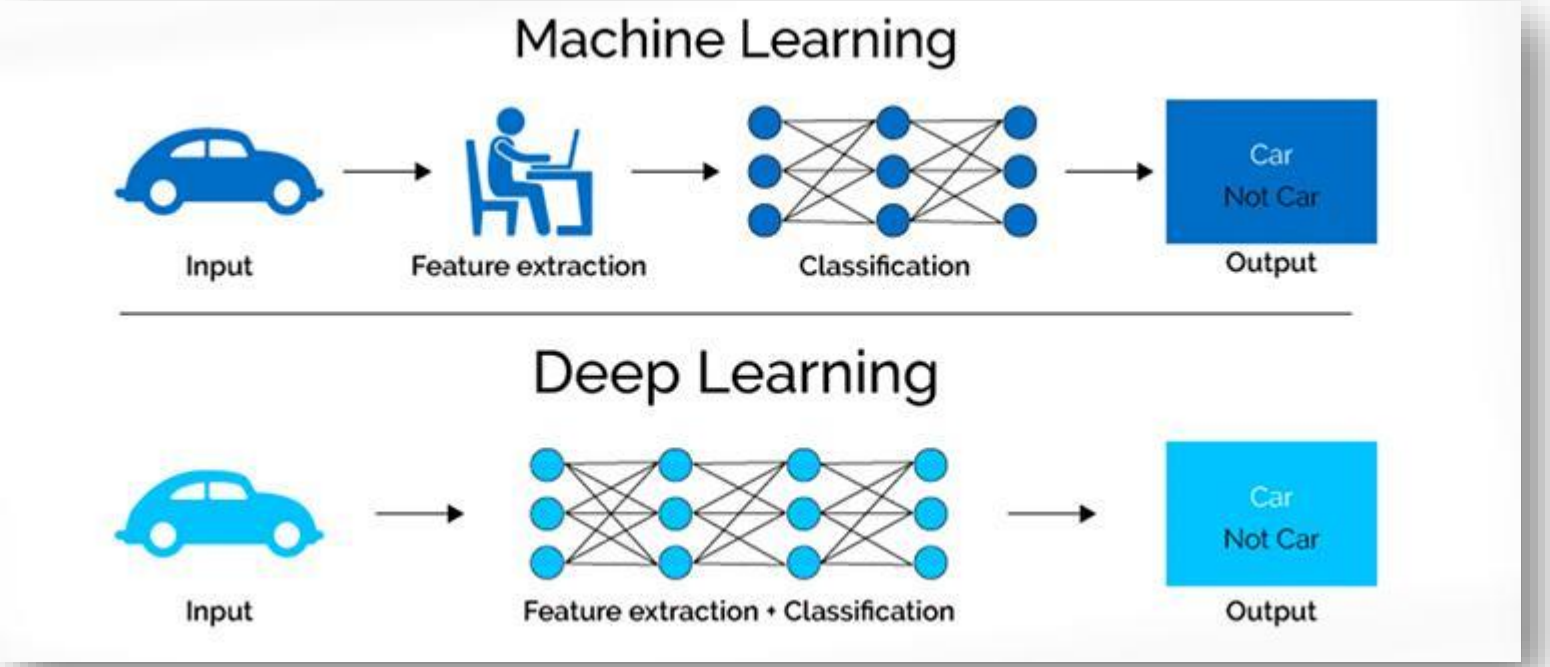
MindSpore： 由华为推出的新一代全场景AI计算框架，2020年MindSpore正式开源。

PaddlePaddle： 由百度推出的中国首个自主研发、功能丰富、开源开放的深度学习平台。

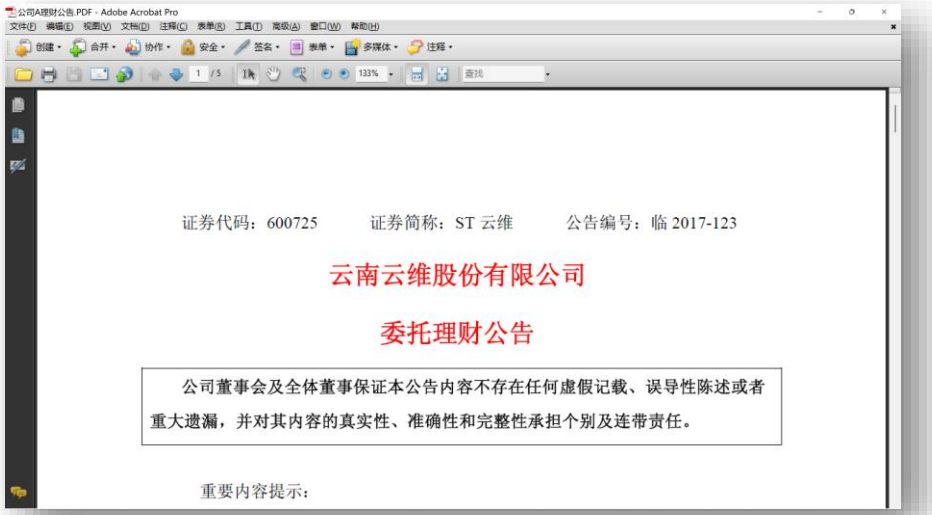
深度学习地位


人工智能
能够感知、推理、行动和适应的程序
机器学习
能够随着数据量的增加不断改进性能的算法
深度学习
机器学习的一个子集：利用多层神经网络从大量数据中进行学习

机器学习与深度学习


Machine Learning
Input
Feature extraction
Classification
Car
Not Car
Output
Deep Learning
Input
Feature extraction + Classification
Car
Not Car
Output

中英文献分析



证券代码：600725　　证券简称：ST 云维　　公告编号：临 2017-123

云南云维股份有限公司

委托理财公告

公司董事会及全体董事保证本公告内容不存在任何虚假记载、误导性陈述或者重大遗漏，并对其内容的真实性、准确性和完整性承担个别及连带责任。

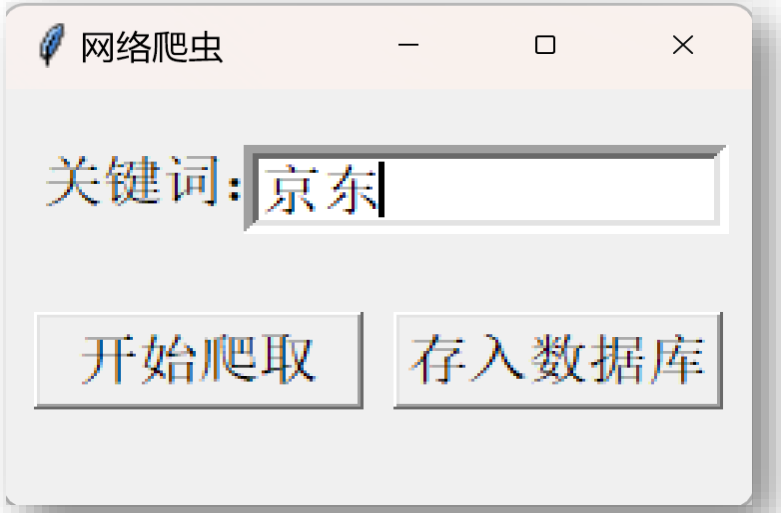
重要内容提示：

AI in Finance: Challenges, Techniques, and Opportunities

LONGBING CAO, University of Technology Sydney, Australia

AI in finance refers to the applications of AI techniques in financial businesses. This area has attracted attention for decades, with both classic and modern AI techniques applied to increasingly broader areas of finance, economy, and society. In contrast to reviews on discussing the problems, aspects, and opportunities of finance benefited from specific or some new-generation AI and data science (AIDS) techniques or the progress of applying specific techniques to resolving certain financial problems, this review offers a comprehensive and dense landscape of the overwhelming challenges, techniques, and opportunities of AIDS research in finance over the past decades. The challenges of financial businesses and data are first outlined, followed by a comprehensive categorization and a dense overview of the decades of AIDS research in finance. We then structure and illustrate the data-driven analytics and learning of financial businesses and data. A comparison, criticism, and discussion of classic versus modern AIDS techniques for finance follows. Finally, the open issues and opportunities to address future AIDS-empowered finance and finance-motivated AIDS research are discussed.

网络爬虫



熟练掌握正则表达式的规则及应用…

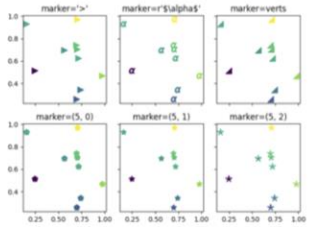
数据库设计

- SQLite
- MySQL
- MongoDB
- Redis
- Microsoft SQL Server 2000
- ….

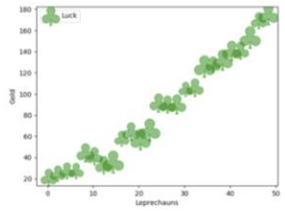
tkinter设计步骤

- 导入tkinter模块
- 创建GUI主窗体
- 添加人机交互控件并编写相应的函数
- 在主事件循环中等待用户触发事件响应

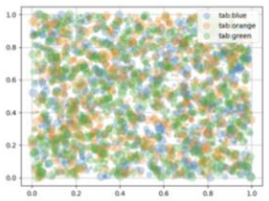
Matplotlib



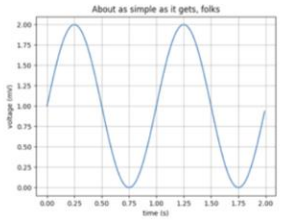
Marker examples



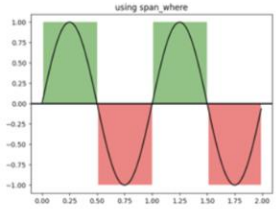
Scatter Symbol



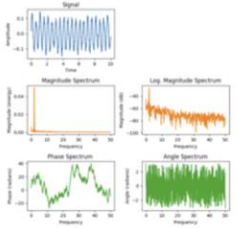
Scatter plots with a legend



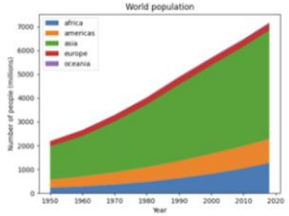
Simple Plot



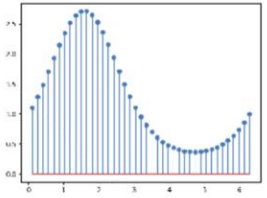
Using span_where



Spectrum Representations



Stackplots and streamgraphs



Stem Plot

Numpy

NumPy(Numerical Python的缩写)：是一个开源的Python科学计算库，NumPy数组在数值运算方面的效率优于列表。它是数据分析、机器学习和科学计算的主力军。

官网：**https://numpy.org/doc/stable/**

Pandas

Pandas ： 基于NumPy 的一种工具，该工具是为了解决数据分析任务而创建的。Pandas 纳入了大量库和一些标准的数据模型，提供了大量能快速便捷地处理数据的函数和方法。Pandas有三个重要的数据结构：一维系列(Series)和二维数据框(DataFrame)、三维(Panel)。

官网： https://pandas.pydata.org/

scikit-learn

scikit-learn： 基于NumPy, SciPy, matplotlib，可以实现数据预处理、分类、回归、降维、聚类、模型选择等常用的机器学习算法，是数据挖掘和数据分析的一个简单有效工具。

机器学习分类：有监督学习、无监督学习

题型及分值

考试题型如下：

选择题：2分*20题=40分，涵盖全部教学内容

程序填空题：2分*3空*5题=30分，涵盖全部教学内容

编程题：10分*3题=30分

复习重点：窗体设计、图形绘制、数据清洗、文献词频统计、爬虫与数据库、机器学习算法应用

谢　谢