**“机器学习算法实现”实验报告**

**课程名称：**人工智能导论

**专业：**数据科学与大数据技术

**年级：**2017级

**姓名：**刘颖凡

**学号：**10172100123

**目录**

[一、实验目的 2](#_Toc8753324)

[二、实验前期准备 2](#_Toc8753325)

[三、实验内容及要求 3](#_Toc8753326)

[四、实验步骤 3](#_Toc8753327)

[1. 回归（Regression） 3](#_Toc8753328)

[2. 分类（Classification） 5](#_Toc8753329)

[2.1 SVM 算法 5](#_Toc8753330)

[2.2 决策树 5](#_Toc8753331)

[2.3 随机森林 5](#_Toc8753332)

[2.4 三种分类算法评估 6](#_Toc8753333)

[3. 聚类 (Clustering) 7](#_Toc8753334)

[3.1 DBScan算法 8](#_Toc8753335)

[3.2 K-Means算法 8](#_Toc8753336)

[3.3 GMM 9](#_Toc8753337)

[3.4 层次聚类算法 10](#_Toc8753338)

[3.5 四种聚类算法评估 11](#_Toc8753339)

[五、总结与体会 12](#_Toc8753340)

[参考文档 12](#_Toc8753341)

# 一、实验目的

1. 动手实践，通过程序实现机器学习算法；

2. 学习并掌握Scikit-learn函数库。

# 二、实验前期准备

1. 数据集来源：AirQualityUCI.csv

iBeacon\_RSSI\_Labeled.csv

iBeacon\_RSSI\_Unlabeled.csv

2. 开发环境：Python 3.0 (jupyter notebook)

# 三、实验内容及要求

对数据集实现下列三种类型算法：

1.回归（Air quality dataset）：线性回归

2.分类（BLE&RSSI dataset）：SVM、决策树、随机森林

3.聚类（BLE&RSSI dataset）：DBScan、K-means、GMM、层次聚类算法

# 四、实验步骤

## 1. 回归（Regression）

通过**线性回归**算法（Linear Regression）实现。

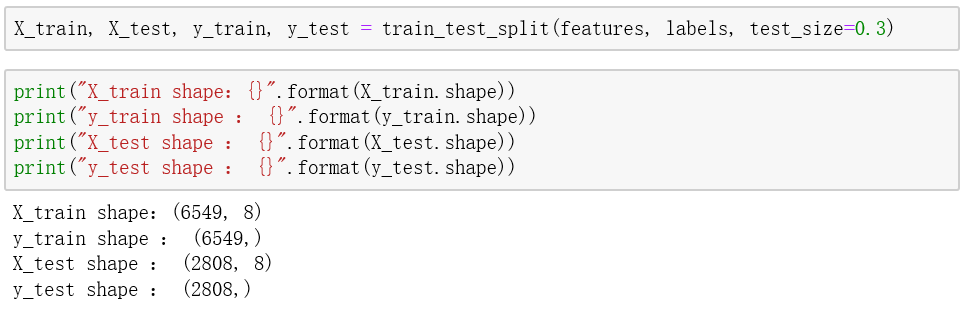
**1.1 数据预处理**

本次实验中用到第一个空气质量数据集：AirQualityUCI.csv

观察数据可知，数据集中-200表示缺失数据，数据集中，“NMHC(GT)”，“NOx(GT)”，“NO2(GT)”三列中含-200行过多，为提高模型的准确度，删除该三列。

**1.2 划分训练-测试集**

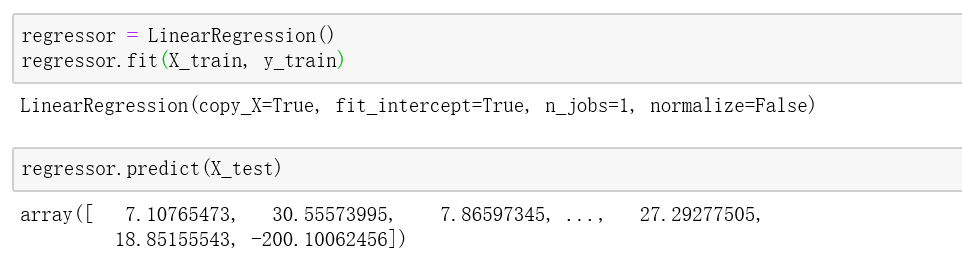
调用Scikit-learn库函数train\_test\_split直接实现。



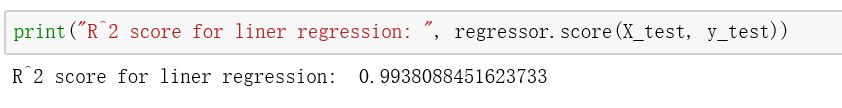
**1.3 算法实现**

将**“温度T”作为labels**，其余特征作为输入参数，来对T进行预测。

调用 sklearn中linear\_model 库函数：



**1.4 实验结果及准确度分析**

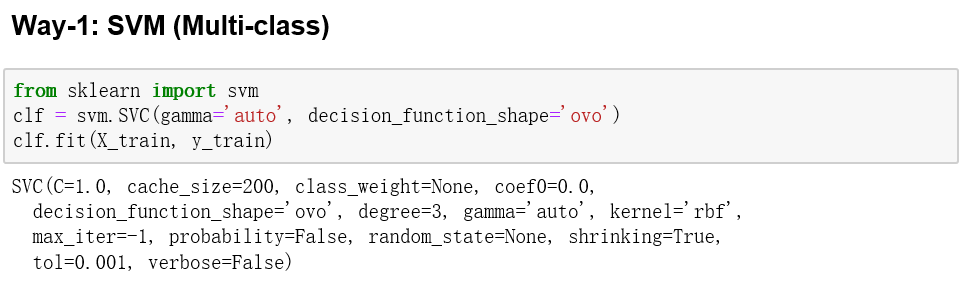


由图片可知，线性回归得出的结果的准确度较高。

## 2. 分类（Classification）

本次实验中用到第二个iBeacon数据集：iBeacon\_RSSI\_Labeled.csv；观察数据可知，数据集中-200表示缺失数据，每个location是一个13维的数据。

### 2.1 SVM 算法



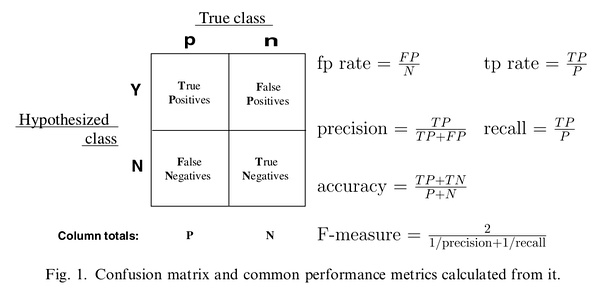
### 2.2 决策树



### 2.3 随机森林



### 2.4 三种分类算法评估



上图表示为一个二分类的混淆矩阵（多分类同理，只需要把不属于当前类的其他类都考虑为负例），表格中的四个参数说明：

True Positive（TP）：预测为正例，实际为正例

False Positive（FP）：预测为正例，实际为负例

True Negative（TN）：预测为负例，实际为负例

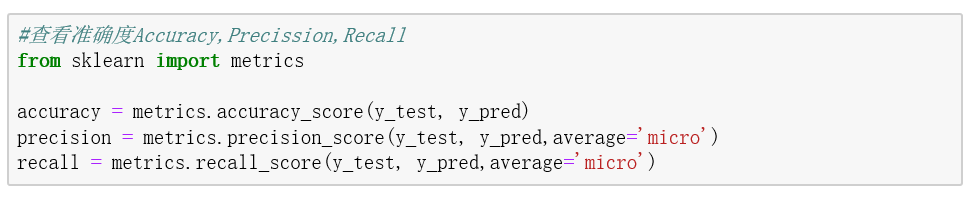
False Negative（FN）：预测为正例，实际为负例

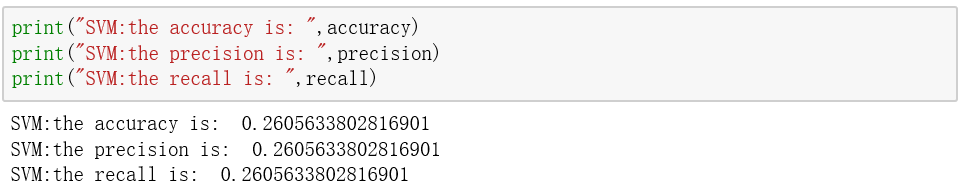
从这我们可以看出，TP和TN都是预测对了，FP和FN都是预测错了。

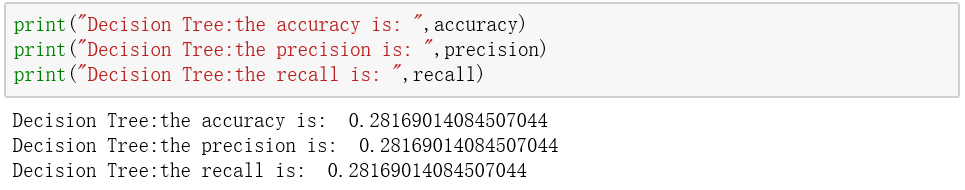
accuracy=\frac{TP+TN}{TP+TN+FP+FN}

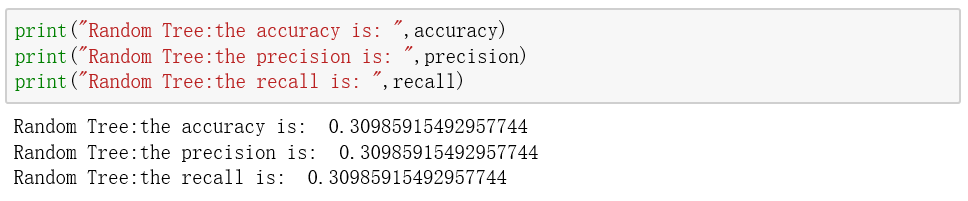
precision=\frac{TP}{TP+FP}

recall=\frac{TP}{TP+FN}









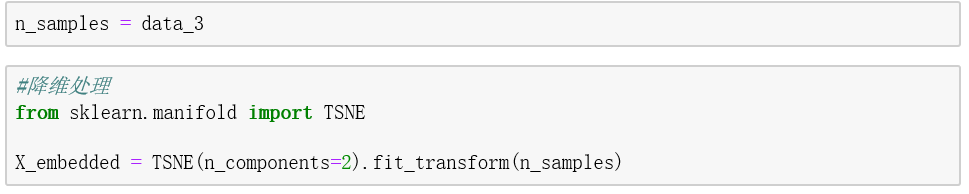
**由于数据集稀疏的原因，导致三种算法的准确度都整体偏低。**

由结果展示得**，随机森林**的准确度最高**，**其次为**决策树，**最差的**为SVM算法。**

## 3. 聚类 (Clustering)

本次实验中用到第三个数据集：iBeacon\_RSSI\_Unlabeled.csv。

在进行可视化处理时，用到t-SNE方法进行降维处理：



### 3.1 DBScan算法

该算法中设置参数**eps=100,min\_samples=10**，最终聚类得出10个**labels**.



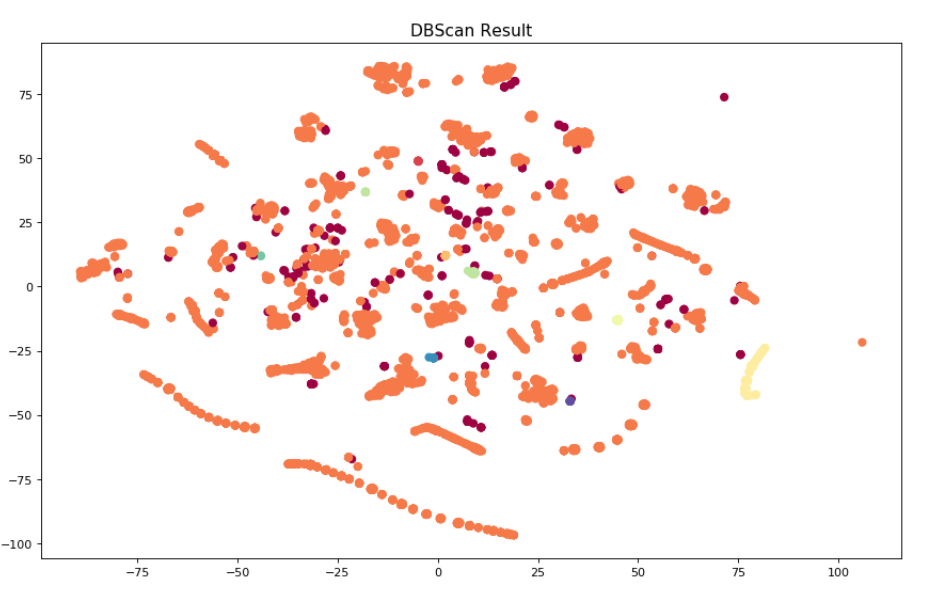
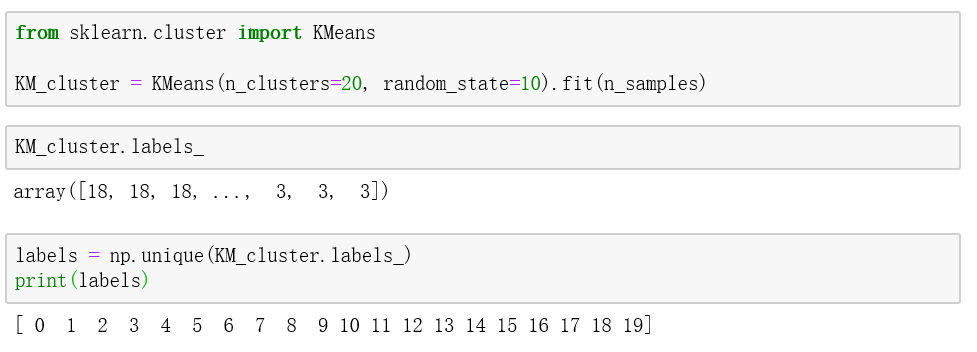


图1：DBScan可视化结果

### 3.2 K-Means算法

该算法中设置参数**n\_clusters=20**，设置20个**labels**.



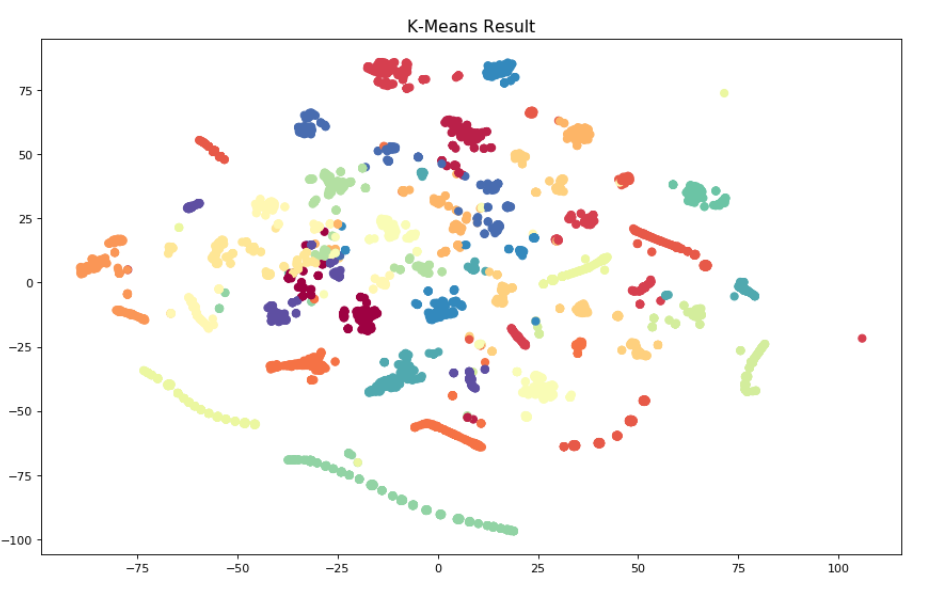


图2：K-Means算法可视化结果

### 3.3 GMM

该算法中设置参数**n\_components=20**，设置20个**labels**.

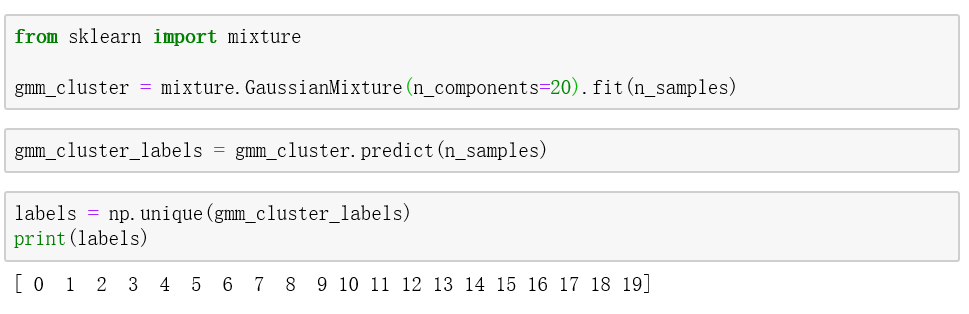
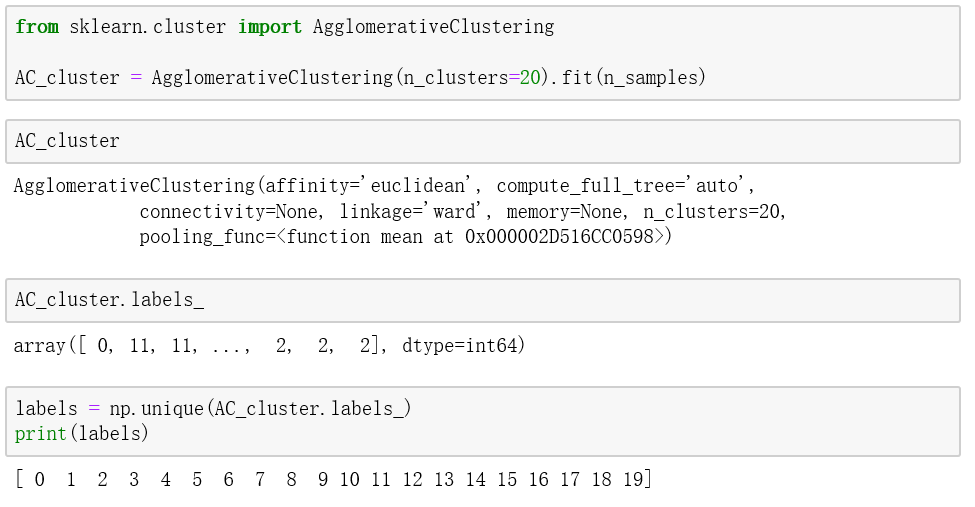




图3：GMM算法可视化结果

### 3.4 层次聚类算法

该算法中设置参数**n\_clusters=20**，设置20个**labels**.



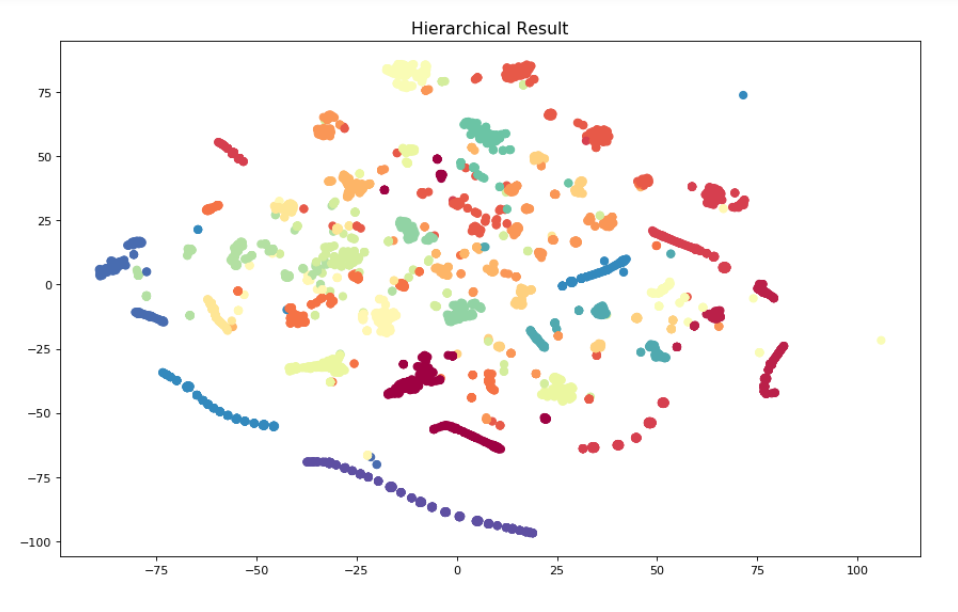


图4：层次聚类算法可视化结果

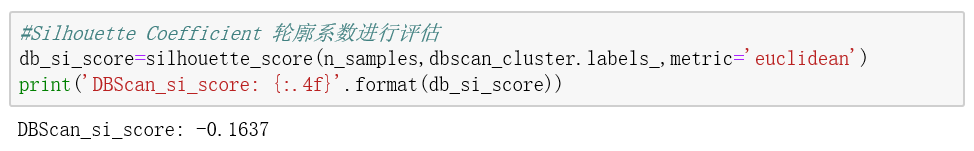
### 3.5 四种聚类算法评估

轮廓系数（Silhouette Coefficient），是聚类效果好坏的一种评价方式。最早由 Peter J. Rousseeuw 在 1986 提出。它结合内聚度和分离度两种因素。可以用来在相同原始数据的基础上用来评价不同算法、或者算法不同运行方式对聚类结果所产生的影响。

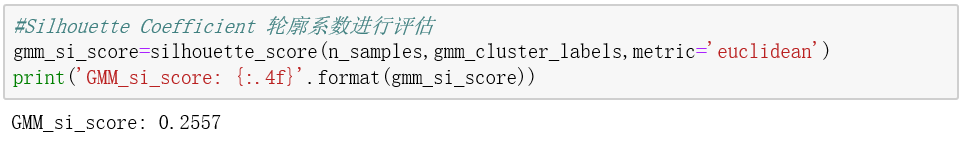
若si接近1，则说明样本i聚类合理；

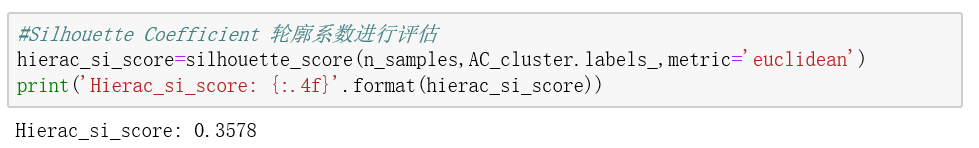
若si接近-1，则说明样本i更应该分类到另外的簇；

若si 近似为0，则说明样本i在两个簇的边界上。









由结果展示得**，K-Means算法**的结果最好**，其**次为**层次聚类算法、GMM算法，**最差的**为DBScan算法。**

# 五、总结与体会

1.整个实验主要通过理论学习机器学习相关算法，了解原理机制，在此基础上通过调用Sklearn库函数以直接实现相关算法，以辅助对理论学习的理解；

2.由于数据集本身的原因，三种类型，八个模型算法的准确度**整体都偏低**，其次，模型设置的参数不同也会影响最终的准确度，因此本次实验结果仍需不断完善；

3.在进行实验的过程中，巩固了对python语言的掌握。

# 参考文档

https://scikit-learn.org/stable/