大数据挖掘实验指导

——基于DataOne大数据实验平台

深圳大学计算机与软件学院

2021年5月

01 DataOne大数据实验平台

02 Hadoop集群计算基础

03 Spark编程

04 机器学习库MLlib

05 教学实验案例

PART ONE

DataOne大数据实验平台

DataOne Big Data Experimental Platform

1.1 DataOne大数据实验平台简介

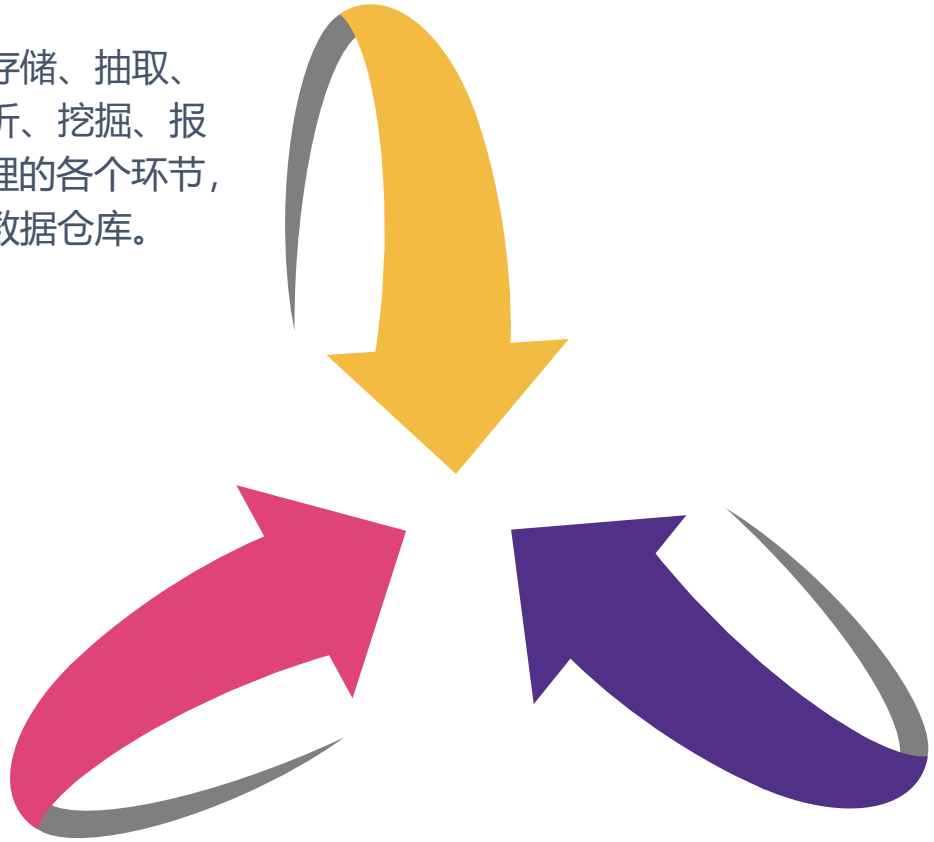
大数据统一处理平台：

大数据统一处理平台（DataOne）是基于开源社区，提供简单易用、一站式的大数据处理、计算、管理平台，覆盖集群部署与管理，数据同步，任务开发，任务管理开发，数据管理和数据运维等功能，使企业能轻松的完成数据采集、抽取、转换、建模、分析、挖掘、报表展示等数据处理的各个环节。DataOne极大的降低了企业构建大数据中心的成本，使其能聚焦在业务层和数据价值变现上，占领商业先机。

1.2 DataOne大数据实验平台应用场景

数据仓库建设：

覆盖数据采集、存储、抽取、转换、建模、分析、挖掘、报表展现等数据处理的各个环节，可快速搭建企业数据仓库。



实时数据处理：

基于Spark Streaming、Storm等套件，实现对企业实时业务的风险监控与告警，比如：工业生产线的实时故障预警、网站的实时流量分析等应用场景。

离线批处理计算：

基于Hadoop、Hive、Spark等套件，提供对数据进行同步、抽取、转换、分析等离线数据处理功能，快速挖掘企业海量历史数据的商业价值。

1.3 DataOne大数据实验平台产品优势

降低门槛

一站式、拖拽式的IDE开发，数据采集、抽取转换、分析、挖掘、报表展现等数据处理的各个环节，均可在Web式IDE上轻松完成开发。

强大的任务调度

支持多任务并发，支持小时、天、周、月等多种调度周期，支持小时、天、周、月的混合调度方式。

便捷运维

管理和监控任务运行状态，可通过邮件、短信、云之家等方式及时告警，避免业务故障。

智慧路由式数据集成

支持MySQL、Oracle、MongoDB、日志采集等各种数据源，且支持离线、实时、增量等各种方式的数据集成，数据集成工具由机器智慧选择。

降低成本

低门槛的数据集成、数据计算、数据分析和挖掘平台，业务专家也可以快速上手平台，从而降低企业人力成本。

PART TWO

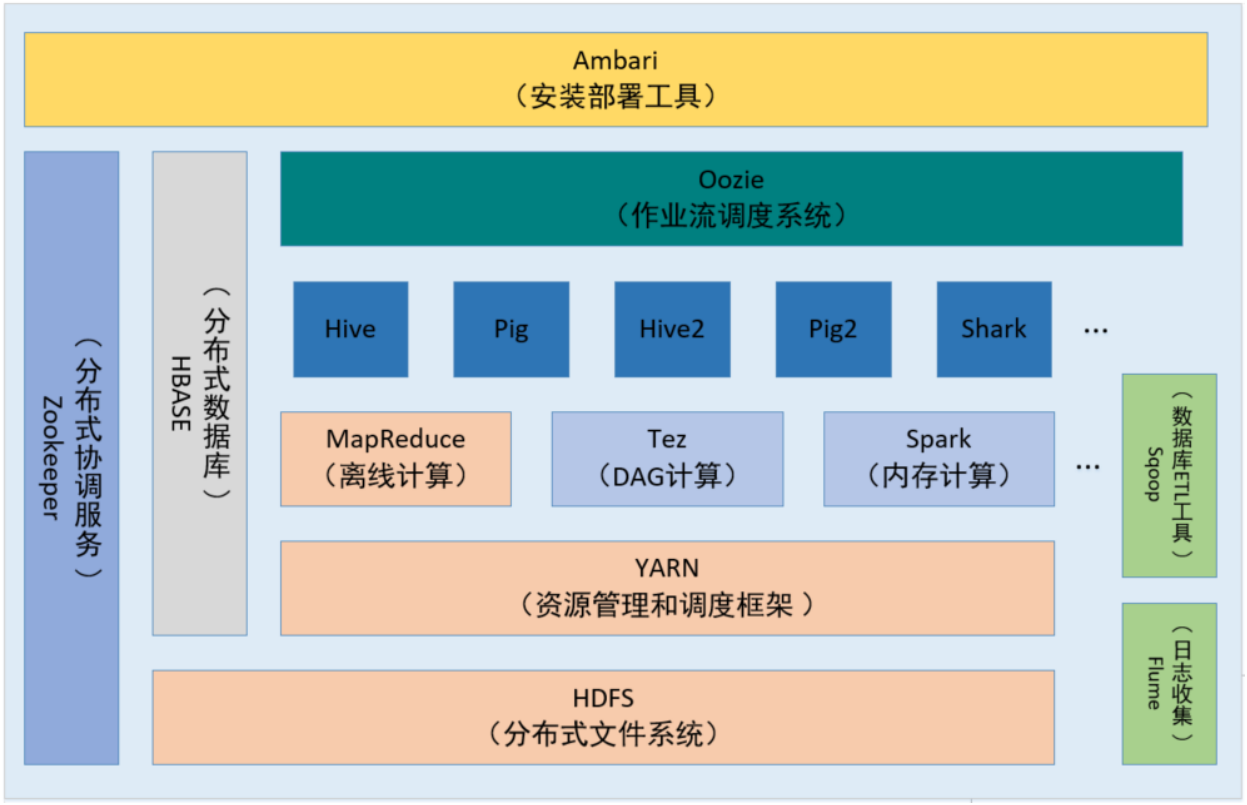
Hadoop集群计算基础

Hadoop Cluster Computing Basics

2.1 Hadoop概述

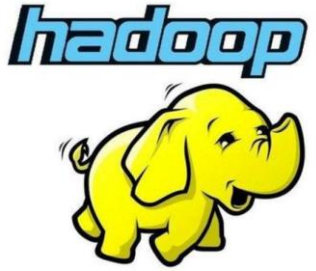
Hadoop简介：

Hadoop是Apache软件基金会旗下的一个开源分布式计算平台，它为分布式环境提供了对海量数据进行处理的能力。



Hadoop优势与特征：

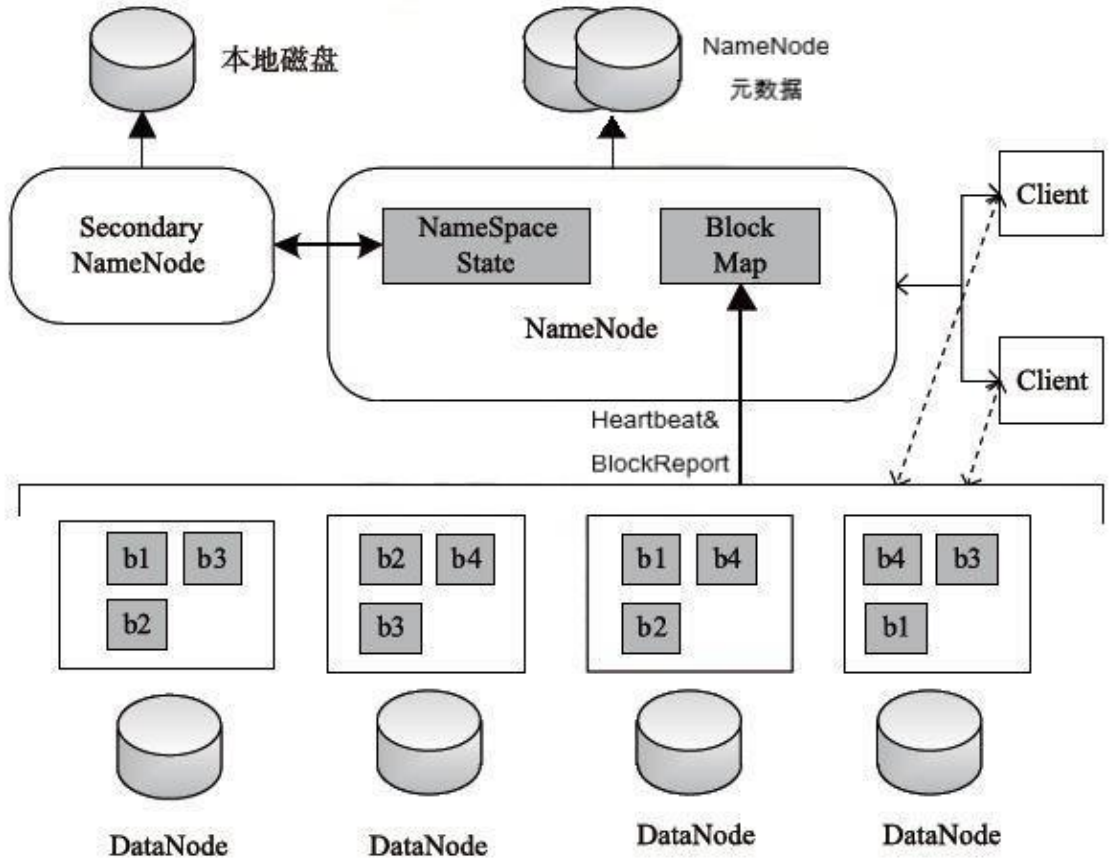
1. 高可靠性
2. 高扩展性
3. 高效性
4. 高容错性
5. 成本低
6. 可构建在廉价机器上
7. 支持多种编程语言



2.2 HDFS分布式文件系统

HDFS简介：

HDFS是以分布式进行存储的文件系统，主要负责集群数据的存储和读取。



HDFS组成：

1. **NameNode**
   用于存储元数据以及处理客户端发出的请求。

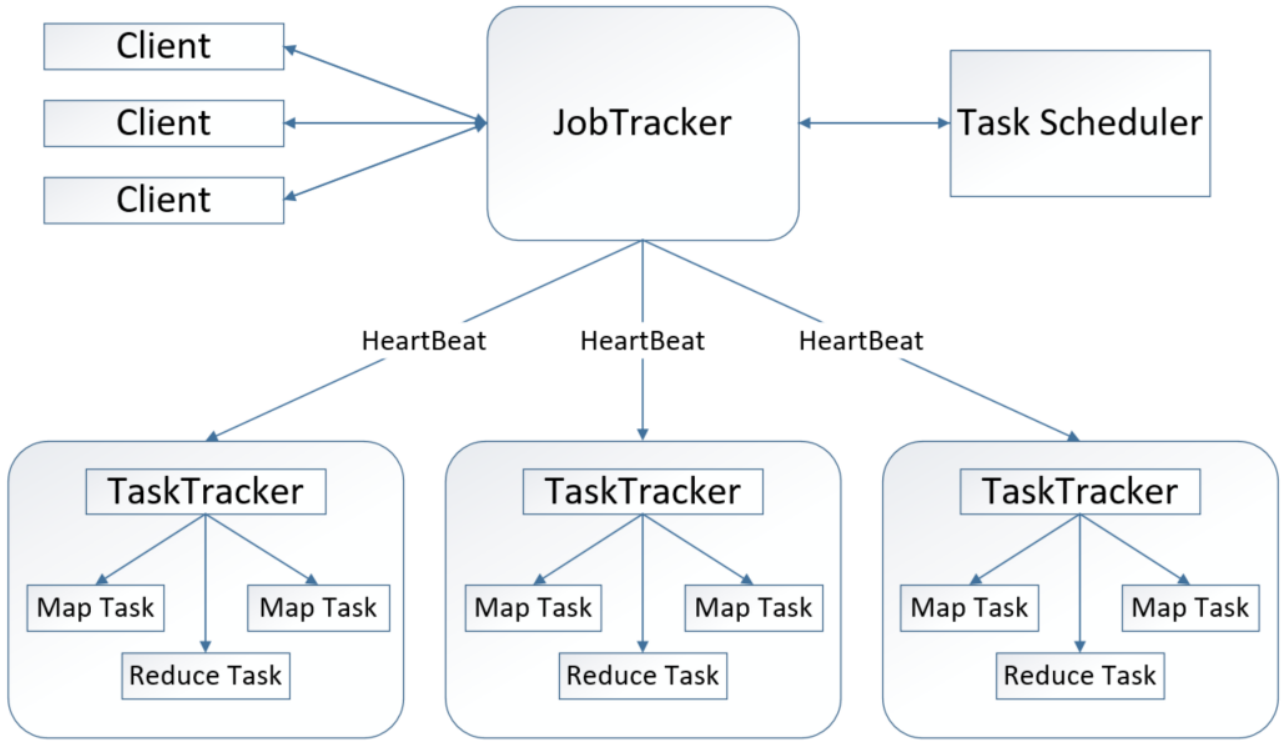
2. **Secondary NameNode**
   用于备份NameNode的数据。

3. **DataNode**
   真正存储数据的地方，在DataNode中，文件以数据块的形式进行存储。

2.3 MapReduce分布式并行编程模型

MapReduce简介：

MapReduce是基于磁盘进行计算的、用于大规模数据集离线并行运算的编程模型。



MapReduce组成：

1. **Client**
   用户也可通过Client进行MapReduce程序的提交，和查看作业的运行状态。

2. **JobTracker**
   JobTracker负责资源监控和作业调度。

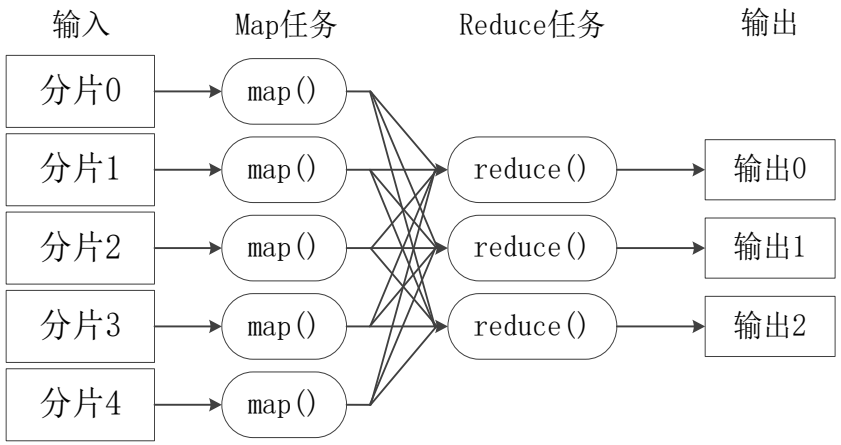
3. **TaskTracker**
   TaskTracker通过“心跳”周期性地将本节点上资源的使用情况，和任务运行进度汇报给JobTracker，同时接收JobTracker发送过来的命令并执行相应的操作。

4. **Task**
   Task分为Map Task和Reduce Task两种，均由TaskTracker启动。

2.3 MapReduce分布式并行编程模型

MapReduce处理流程：

MapReduce采用的是“分而治之”的处理策略，将复杂的、运行于大规模集群上的并行计算过程高度地抽象为两个函数：Map和Reduce。



函数	输入	输出	说明
Map	**<k_1,v_1>** 例如： < “a a b” ,1>	**List(<k_2,v_2>)** 例如： <“a”,1> <“a”,1> <“b”,1>	首先，将小数据集进一步解析成一批<key,value>对，即<k_1,v_1>，并输入Map函数中进行处理。 然后，Map函数中每一个输入的<k_1,v_1>会转化成一批<k_2,v_2>进行输出，即List(<k_2,v_2>)。这些<k_2,v_2>是MapReduce计算的中间结果。
Reduce	**<k_2,List(v_2)>** 例如： <“a”,<1,1>>	**<k_3,v_3>** 例如： <“a”,2>	输入的中间结果<k_2,List(v_2)>中的List(v_2)表示是一批属于同一个k_2的value。 Reduce函数会将这些中间结果<k_2,List(v_2)>中具有相同键的键值对以某种特定的方式组合起来，输出组合后的最终结果<k_3,v_3>。

Spark编程

Spark Programming

3.1 Spark概述

Spark简介：

Spark是专为大规模数据处理而设计的快速通用的计算引擎。

Spark生态系统中的组件	适用场景
Spark	复杂的批量数据处理
Spark SQL	基于历史数据的交互式查询
Spark Streaming	基于实时数据流的数据处理
MLlib	基于历史数据的数据挖掘
GraphX	图结构数据的处理

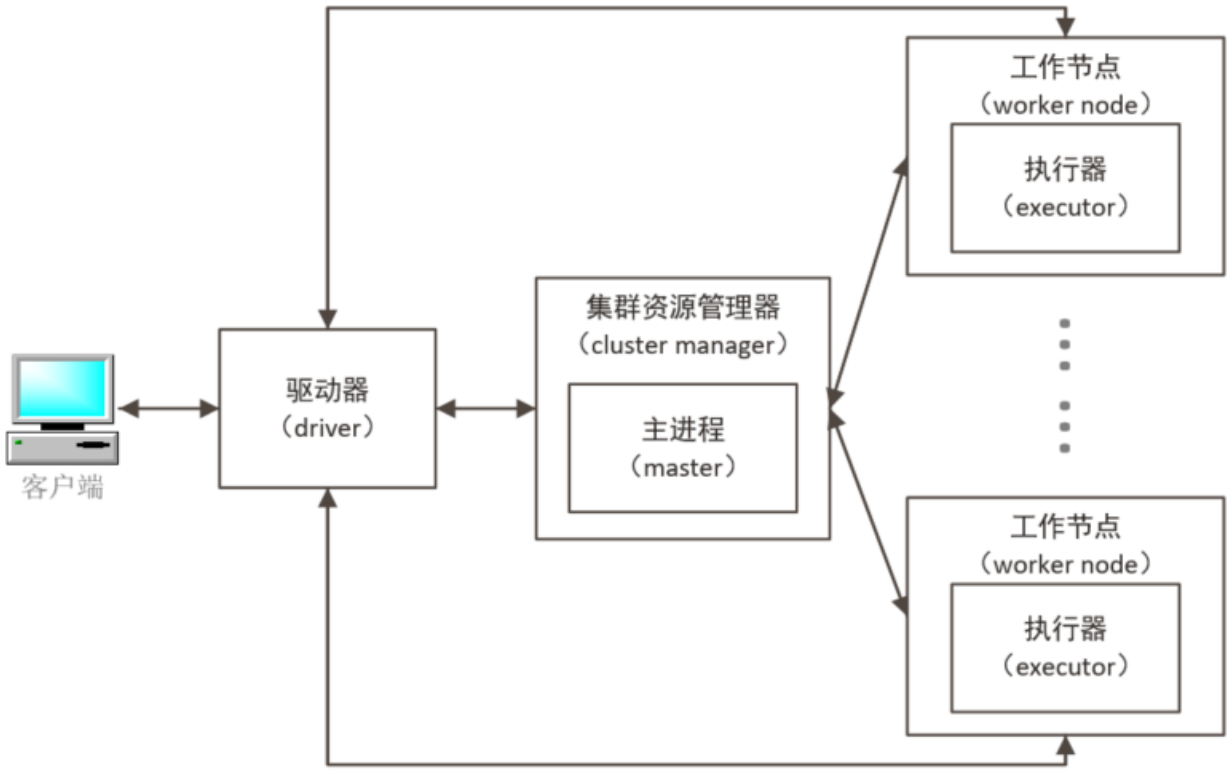
Spark优势与特征：

Spark在继承了Hadoop MapReduce的优点的同时，也完善了Hadoop MapReduce的一些不足。相比于Hadoop MapReduce，Spark主要有如下两大优势：

1. 提供了多种数据集操作类型，编程模式比Hadoop MapReduce更灵活。

2. 提供了内存计算，可将中间结果放到内存中，之后的迭代计算都可以直接使用内存中的中间结果进行运算，避免了从磁盘中频繁读取数据，对于迭代运算效率更高；

3.2 Spark架构



Spark运行架构组成：

1. **驱动器（driver）**
   Spark客户端用来提交应用的进程。主要用于创建SparkSession对象和确定应用的执行计划。

2. **执行器（executor）**
   运行执行计划中描述的任务的进程，其运行在工作节点上。

3. **主进程（master）**
   向集群申请资源，并把资源交给驱动器的进程。

4. **集群资源管理器（cluster manager）**
   负责监控工作节点，并在主进程发起请求时在工作节点上预留资源。

3.3 Spark编程基础

SparkSession：

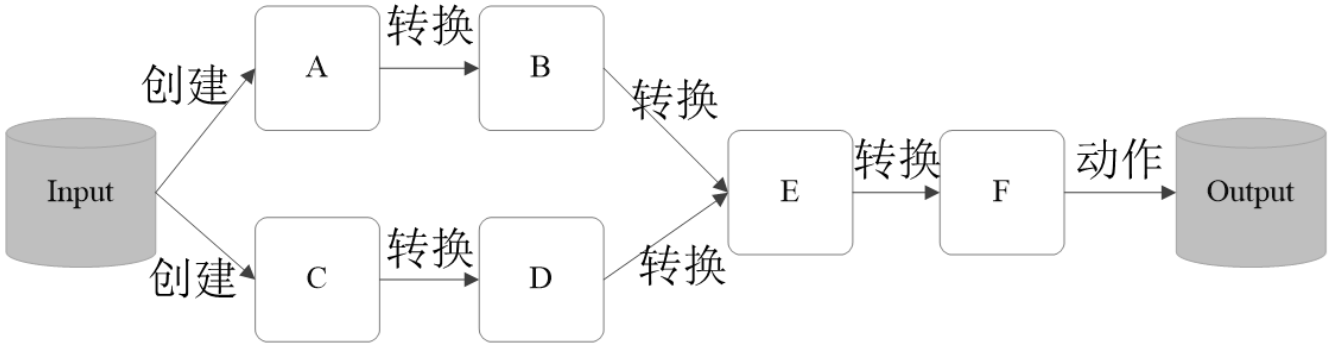
SparkSession是Spark集群的一个连接，是所有Spark程序的统一入口，其在程序整个运行过程中都会使用。

RDD：

RDD（Resilient Distributed Dataset）是Spark编程中最基本的数据对象，它是Spark应用中的数据集。

RDD的操作分为“转换”（Transformation）和“动作”（Action）两种类型。

- **转换操作**：对现有的RDD进行操作来创建新的RDD。
- **动作操作**：在数据集上进行运算，返回计算值的操作。

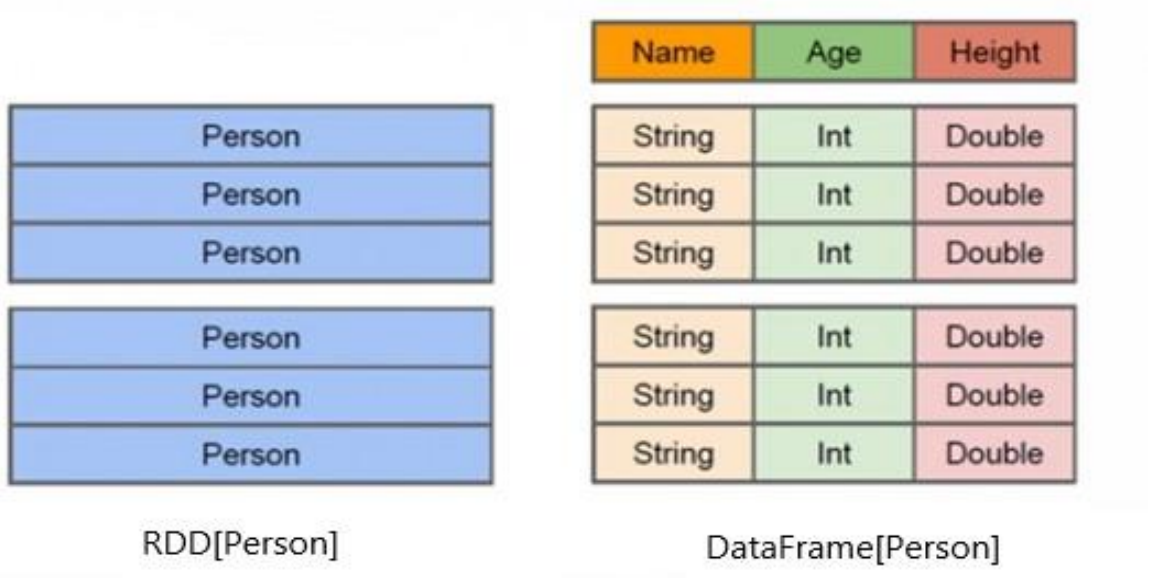


3.3 Spark编程基础

Spark SQL：

Spark SQL在RDD的基础上，增加了DataFrame（带有Schema信息的RDD）。它让Spark具备了处理大规模结构化数据的能力，不仅比原有的RDD转化方式更加简单易用，而且还拥有更高的计算性能。

DataFrame和RDD一样，其各种变换操作也采用惰性机制，分为“转换”（Transformation）和“行动”（Action）两种类型。



机器学习库MLlib

Machine Learning Library MLlib

4 Spark MLlib

Spark MLlib概述：

Spark MLlib（Machine Learning Library）是Spark提供的一个机器学习库，它提供了很多常用机器学习算法的分布式实现。
开发者只需对Spark编程，和机器学习算法的基本原理有所了解，就可以通过简单地调用MLlib中相应的API，来实现基于海量数据的分布式机器学习过程。

基本统计

- 相关性
- 假设检验
- 最值、平均值、方差、合计
- ……

特征处理

- Word2Vec
- TF-IDF
- ChiSqSelector
- PCA
- ……

分类与回归

- 朴素贝叶斯
- 支持向量机
- 线性回归
- 随机森林回归
- ……

聚类

- K-means
- GMM
- LDA
- ……

协同过滤

- 显式反馈
- 隐式反馈
- ……

频繁模式挖掘

- FP-Growth
- PrefixSpan
- ……

05

PART FIVE

教学实验案例

Teaching Experiment Cases

5.1 教学实验要求及内容

适用专业及年级：

高等学校计算机及相关专业的本科高年级学生和研究生。

课程目标与基本要求：

通过本课程的学习，帮助学生了解大数据技术的基本概念、存储、处理、分析和应用；掌握Hadoop HDFS、Hadoop MapReduce，以及Spark的框架与应用；熟悉Spark MLlib机器学习库的使用；拥有运用大数据技术对现实中的一些问题进行分析和求解的能力。
本课程全面系统地介绍现代大数据技术编程、开发方法、语言、环境和工具等，从大量现实案例入手，渐进地展开大数据技术的各个技术层面。

实验教学环境：

DataOne大数据实验平台。

5.2 教学实验案例(1)——原材料采购策略制定

1 实验目的：

- 掌握Spark集群的使用；
- 掌握Spark数据统计的基本方法；
- 了解Spark MLlib的基本用法；
- 掌握Spark MLlib求解线性回归模型的方法。

2 实验环境：

使用DataOne大数据实验平台与Python语言，在spark集群上编写、运行和调试spark程序。

3 实验内容：

- 数据表的选取与HDFS中文件的上传；
- 通过使用dataframe对数据进行统计，统计出每种产品每月的销售额；
- 通过作图（时间-销量），得出每种产品销量的大体趋势；
- 通过LinearRegression分别对每种产品每月的销售额进行线性回归模型的训练，分析训练得到的模型的准确度，并用该模型来预测未来半年内产品的销售额；
- 根据预测的结果，给出未来半年合理的原材料采购方案。

5.2 教学实验案例(1)——原材料采购策略制定

(1) 数据表的选取与HDFS中文件的上传

经过观察与分析，我们得知所要用到的表有（保存在datasets文件夹下）：

- **“销售订制单”**
- **“配方成本表-面包”**
- **“配方成本表-烘焙”**
- **“配方成本表-蛋糕”**

通过“销售订制单”，我们可以得到在某一个时间点，卖出了几件产品。通过“配方成本表”我们可以获得每一种产品生产所需要的原材料种类及数量。

然后通过以下命令将这些数据表上传到HDFS：

$ hdfs dfs –put –f /home/SZU/datasets/. /user/SZU/datasets/.

5.2 教学实验案例(1)——原材料采购策略制定

(2) 通过使用dataframe对数据进行统计，统计出每种产品每月的销售额

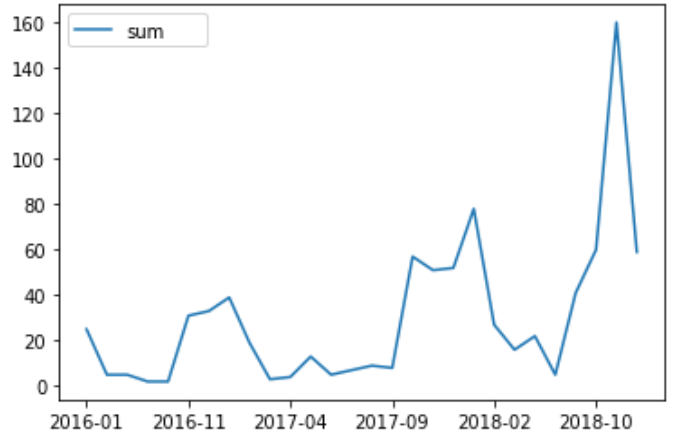
```
#创建sparksession
spark=SparkSession.builder.getOrCreate()

#读取csv文件，并生成dataframe
df0=spark.read.format('com.databricks.spark.csv').options(header='true',
inferschema='true').load('/user/SZU/datasets/销售订制单.csv')

#将string类型的日期转换为timestamp类型的日期
df0=df0.withColumn("下单时间", df0.下单时间.cast('timestamp'))

#选择出需要用到的列
df0=df0.select('商品名称','数量','下单时间' )

#提取出年-月，并将其新建为dataframe的一列，命名为 '时间'
df0=df0.select('*',date_format('下单时间', 'yyyy-MM').alias('时间' ))

#按月份分别统计每种商品的销量
df0=df0.groupBy([df0['商品名称'],df0['时间']]).sum('数量').orderBy(df0['商品
名称'].asc(),df0['时间'].asc())
```

商品名称	时间	数量
DIY蛋糕亲子烘焙	2016-01	25
DIY蛋糕亲子烘焙	2016-05	5
DIY蛋糕亲子烘焙	2016-06	5
DIY蛋糕亲子烘焙	2016-07	2
DIY蛋糕亲子烘焙	2016-10	2
DIY蛋糕亲子烘焙	2016-11	31
DIY蛋糕亲子烘焙	2016-12	33
DIY蛋糕亲子烘焙	2017-01	39
DIY蛋糕亲子烘焙	2017-02	19
DIY蛋糕亲子烘焙	2017-03	3
DIY蛋糕亲子烘焙	2017-04	4
DIY蛋糕亲子烘焙	2017-05	13
DIY蛋糕亲子烘焙	2017-06	5
DIY蛋糕亲子烘焙	2017-07	7
DIY蛋糕亲子烘焙	2017-08	9
DIY蛋糕亲子烘焙	2018-01	8
DIY蛋糕亲子烘焙	2018-02	57
DIY蛋糕亲子烘焙	2018-04	51
DIY蛋糕亲子烘焙	2018-06	52
DIY蛋糕亲子烘焙	2018-08	78

5.2 教学实验案例(1)——原材料采购策略制定

（3）绘制折线图得出每种产品销量的大体趋势

```
pddf0=df0.toPandas()
good_names=['芒香’，'芒果蛋糕’，……]

#作图（时间-销量）
for good_name in good_names:
    print(good_name)
    pddf0[pddf0.商品名称==good_name].plot(x='时间’，y='sum(数量)’，
kind='line')
    plt.rcParams['font.sans-serif']=['SimHei']
    plt.show()
```



DIY蛋糕亲子烘焙

5.2 教学实验案例(1)——原材料采购策略制定

(4) 训练LinearRegression模型预测未来半年内各种产品的销量

```
# 对每种产品的销量进行线性回归
for index in range(30):
    # 加载训练数据
    string = '/user/xuyuming/xuyuming/销量' + str(index) + '.txt'
    training = spark.read.format("libsvm").load(string)
    # 训练模型
    lr = LinearRegression(maxIter=10, regParam=0.3,
elasticNetParam=0.8)
    lrModel = lr.fit(training)
    # 输出并保存得到的结果
    print("Coefficients: %s" % str(lrModel.coefficients))
    print("Intercept: %s" % str(lrModel.intercept))
    coeff.append(lrModel.coefficients)
    inter.append(lrModel.intercept)
    # 输出模型的准确度等信息
    trainingSummary = lrModel.summary
    print("RMSE: %f" % trainingSummary.rootMeanSquaredError)
    print("r2: %f" % trainingSummary.r2)
```

	芒香	芒果蛋糕	秘密花园	暗香	提香	黄金彼岸
Coefficients	3.10	4.46	4.52	3.01	4.18	0.14
Intercept	492.97	113.71	127.71	423.67	108.85	95.55
RMSE	80.17	65.20	64.09	81.68	65.26	18.83
r2	0.14	0.34	0.35	0.13	0.31	0.01
	榴莲千层	榴莲双拼	心语心愿	红妆	牛轧糖原料	椰蓉粉
Coefficients	0.58	0.84	2.38	1.39	5.85	4.11
Intercept	61.18	36.49	141.14	99.89	1005.22	819.91
RMSE	22.59	19.43	24.83	22.28	167.79	140.11
r2	0.07	0.18	0.22	0.11	0.12	0.09
	南瓜原料套装	DIY饼干烘焙	DIY蛋糕烘焙	手工烘焙	安佳黄油	kiri奶油奶酪
Coefficients	2.68	3.01	2.16	0.63	0.49	-0.11
Intercept	547.58	22.80	-1.44	8.44	90.92	147.11
RMSE	114.65	59.14	27.88	15.13	30.77	31.92
r2	0.06	0.20	0.29	0.14	0.03	0.00
	蛋挞套装	蛋黄酥原料	提子面包	传统长棍	法式起司	米露提子吐司
Coefficients	4.76	4.09	22.84	22.35	23.58	23.13
Intercept	135.50	102.03	1254.26	1404.36	1041.51	1010.84
RMSE	26.31	23.15	358.51	359.64	339.15	352.41
r2	0.29	0.28	0.31	0.29	0.34	0.32
	温泉吐司	黑糖桂圆欧包	冠军芒果	干酪火腿	布里欧	芥末培根
Coefficients	25.08	3.76	3.76	3.94	11.52	10.84
Intercept	969.47	294.69	283.07	293.43	316.92	299.61
RMSE	353.09	97.91	97.98	98.12	90.28	89.57
r2	0.35	0.14	0.14	0.15	0.40	0.37

5.2 教学实验案例(1)——原材料采购策略制定

(4) 训练LinearRegression模型预测未来半年内各种产品的销量

```
# 对于每种产品
for i in range(30):
    # 计算其未来6个月的销量
    for k in range(6):
        num = coeff[i][0]*(len(goods_list)+k)+inter[i]
        sub_sales_forecast.append(num)
```

	芒香	芒果蛋糕	秘密花园	暗香	提香	黄金彼岸
未来6个月销量预测	585.90	247.47	263.27	513.89	234.28	99.79
	589.00	251.93	267.79	516.89	238.46	99.93
	592.10	256.39	272.31	519.90	242.64	100.08
	595.20	260.84	276.83	522.91	246.82	100.22
	598.29	265.30	281.35	525.92	251.00	100.36
	601.39	269.76	285.87	528.92	255.18	100.50
	榴莲千层	榴莲双拼	心语心愿	红妆	牛轧糖原料	椰蓉粉
未来6个月销量预测	78.68	61.78	212.41	141.74	1180.76	943.30
	79.27	62.62	214.78	143.13	1186.61	947.41
	79.85	63.47	217.16	144.53	1192.46	951.52
	80.43	64.31	219.54	145.92	1198.32	955.64
	81.02	65.15	221.91	147.32	1204.17	959.75
	81.60	66.00	224.29	148.71	1210.02	963.86
	南瓜原料套装	DIY饼干烘焙	DIY蛋糕烘焙	手工烘焙	安佳黄油	kiri奶油奶酪
未来6个月销量预测	627.99	112.99	63.46	27.24	105.51	143.91
	630.67	116.00	65.62	27.86	105.99	143.81
	633.35	119.00	67.79	28.49	106.48	143.70
	636.03	122.01	69.95	29.12	106.97	143.59
	638.72	125.02	72.11	29.74	107.45	143.49
	641.40	128.02	74.28	30.37	107.94	143.38
	蛋挞套装	蛋黄酥原料	提子面包	传统长棍	法式起司	米露提子吐司
未来6个月销量预测	278.21	224.60	1939.51	2074.82	1748.79	1704.79
	282.96	228.69	1962.35	2097.16	1772.37	1727.92
	287.72	232.77	1985.20	2119.51	1795.94	1751.06
	292.48	236.86	2008.04	2141.86	1819.52	1774.19
	297.23	240.94	2030.88	2164.21	1843.10	1797.32
	301.99	245.03	2053.72	2186.56	1866.67	1820.45
	温泉吐司	黑糖桂圆欧包	冠军芒果	干酪火腿	布里欧	芥末培根
未来6个月销量预测	1721.81	407.58	395.84	411.70	662.55	624.94
	1746.89	411.34	399.60	415.64	674.07	635.79
	1771.97	415.10	403.36	419.58	685.59	646.63
	1797.05	418.87	407.12	423.52	697.11	657.48
	1822.13	422.63	410.88	427.46	708.63	668.32
	1847.21	426.39	414.64	431.41	720.15	679.17

5.2 教学实验案例(1)——原材料采购策略制定

(5) 根据销量预测结果给出未来半年的原材料采购方案

```
#第i月的原材料采购策略
for i in range(6):
    #第j种产品
    for j in range(30):
        #第j种产品的第k种原材料
        for k in range(len(cai_liao[j])):
            purchase[cai_liao[j][k]]=purchase[ca

    #打印输出每种原材料在该月的采购计划
    print('【2019年'+str(i+1)+'月的原材料采购
    for n in purchase:
        print(str(n)+':'+str(purchase.get(n)))
        #初始化purchase
        purchase[n]=0
```

	1月的原材料采购策略	2月的原材料采购策略	3月的原材料采购策略	4月的原材料采购策略	5月的原材料采购策略	6月的原材料采购策略
kiri奶油奶酪	898.80	919.48	940.15	960.83	981.51	1002.18
不粘派盘	2361.52	2373.23	2384.93	2396.63	2408.33	2420.04
全脂牛奶	457931.61	461806.34	465681.06	469555.79	473430.52	477305.25
可可粉	82339.72	82864.79	83389.86	83914.93	84440.00	84965.07
培根	6249.44	6357.88	6466.33	6574.77	6683.22	6791.67
大豆油	54406.15	55295.67	56185.19	57074.71	57964.23	58853.75
奶油	358286.28	362121.61	365956.94	369792.28	373627.61	377462.94
奶酪	4116.95	4156.38	4195.80	4235.22	4274.64	4314.06
安佳黄油	1269.23	1312.50	1355.76	1399.03	1442.30	1485.57
小麦粉	797220.50	803295.68	809370.86	815446.04	821521.22	827596.40
巧克力	87810.21	88862.53	89914.86	90967.18	92019.51	93071.83
干酪	3543.43	3578.30	3613.16	3648.03	3682.90	3717.77
提子干	96975.62	98117.71	99259.79	100401.87	101543.96	102686.04
果酱	96325.73	96739.85	97153.97	97568.09	97982.21	98396.33
桂圆肉	12227.33	12340.21	12453.10	12565.98	12678.86	12791.75
椰蓉粉	11230.00	11434.29	11638.58	11842.87	12047.16	12251.45
榴莲	4720.96	4755.97	4790.98	4825.99	4861.00	4896.01
泡打粉	2474.69	2519.27	2563.86	2608.45	2653.03	2697.62
火腿	4116.95	4156.38	4195.80	4235.22	4274.64	4314.06
炼乳	51654.45	52406.79	53159.13	53911.47	54663.82	55416.16
牛乳	36096.63	36609.67	37122.71	37635.75	38148.79	38661.83
牛轧糖	50077.02	50933.28	51789.54	52645.79	53502.05	54358.31
细砂糖	476195.33	480701.38	485207.43	489713.48	494219.54	498725.59
芒果	231882.87	235179.22	238475.57	241771.91	245068.26	248364.61
芥末	1249.89	1271.58	1293.27	1314.95	1336.64	1358.33
草莓	89397.56	90320.38	91243.19	92166.01	93088.82	94011.64
葡萄干	11713.82	11922.87	12131.92	12340.98	12550.03	12759.08
蓝莓	7241.96	7293.72	7345.48	7397.24	7448.99	7500.75
蛋挞皮	527.54	529.97	532.41	534.84	537.27	539.70
豆沙	35978.38	35951.74	35925.10	35898.46	35871.82	35845.18
酵母	68858.07	69663.70	70469.34	71274.98	72080.61	72886.25
面粉	2163066.60	2185953.74	2208840.88	2231728.02	2254615.16	2277502.30
食盐	92050.55	93055.64	94060.74	95065.83	96070.93	97076.02
香咸蛋黄	431.74	431.42	431.10	430.78	430.46	430.14
鸡蛋	908031.11	918599.45	929167.79	939736.13	950304.46	960872.80
黄油	423312.09	426134.01	428955.92	431777.84	434599.75	437421.66
黑糖	9374.29	9460.83	9547.37	9633.92	9720.46	9807.01
黑芝麻	4317.41	4314.21	4311.01	4307.82	4304.62	4301.42

5.3 教学实验案例(2)——供应商聚类分析

1 实验目的：

- 了解hive数据仓库的使用；
- 掌握利用主成分分析法（PCA）对数据进行降维的方法；
- 掌握K-means和DBSCAN两种常用的聚类分析算法；
- 掌握Python图表绘制的方法；
- 掌握基本的数据统计方法。

2 实验环境：

对hive数据仓库中的供应商数据，使用DataOne大数据实验平台与Python语言，在spark集群上编写、运行和调试spark程序。

3 实验内容：

- 连接hive数据仓库并读取供应商数据；
- 使用主成分分析（PCA）对数据的特征值进行降维处理；
- 利用K-means算法对供应商进行聚类，并使用Calinski-Harabasz准则评价模型的拟合度；
- 利用DBSCAN算法对供应商进行聚类，并使用Calinski-Harabasz准则评价模型的拟合度；
- 挑选出存在特项异常的供应商，并找出他们的专长项。

一级评价指标	二级评价指标
企业素质	领导素质 员工素质 经营理念 技术设备
企业能力	年营业额 设备更新期限 揽货能力 固定资产 员工数量 子公司、营业网点数量
服务环境	地理位置 信息传递及时率 服务价格 技术水平
交易能力	交货期 退货率 长期客户数量 业务覆盖率
物流能力	送货频率 车辆数量 信息系统应用覆盖率 物流驻点 仓库数量 物流设施数量
售后服务	客户满意度 次品率 售后维护点 客户维护期限

5.3 教学实验案例(2)——供应商聚类分析

(1) 连接hive数据仓库并读取数据

首先，我们先要使用Python的pyhive库与hive数据仓库进行连接，获取到数据表的内容并用其生成一个dataframe。

```
#连接hive中的数据库
conn = hive.Connectio
cursor = conn.cursor()

#获取数据表
cursor.execute("select

#构造dataframe
result = cursor.fetchall
data = pd.DataFrame(

#关闭连接
cursor.close()
conn.close()
```

	id	leadership_quality	staff_quality	management_idea	technical_equipment	volume	equipment_update_period	freight_quality	fixed_assets
0	SUP007749	6.7	7.2	7.1	7.9	6.7	6.8	7.6	6.2
1	SUP008992	7.2	6.2	7.7	7.3	6	7.1	7.1	7.7
2	SUP010988	6.9	6.6	6.3	6.6	7.9	7.9	7.7	7.2
3	SUP019294	6.1	6.6	7	6.2	6.4	6.2	7.7	7.2
4	SUP019907	4.8	4.3	4.2	5.8	4	4.6	4.4	5.5
...	...	...	...	...	...	...	...	...	...
194	SUP972756	6.9	7.3	7.8	6.5	6.2	6.9	8	6.6
195	SUP980578	6.8	7.7	6.3	8	7.6	7.8	6.8	6.4
196	SUP983470	5.4	4	5.1	4.3	4.5	5.8	5.2	5.5
197	SUP987060	4.3	5.4	4.4	5.5	5.2	4.8	5	4.6
198	SUP988696	7.8	7.6	7.8	7.2	6.3	7.9	6.4	6.1

199 rows × 30 columns

5.3 教学实验案例(2)——供应商聚类分析

（2）使用PCA对数据进行降维处理

因为数据集中样本的特征值非常多，将近30个，也就是将近30维，所以我们要先对样本特征进行降维处理。在这里，我们采用主成分分析（PCA）来对数据进行降维处理。

首先，我们先要对数据集的eigenvalue进行分析，看看要降成几维比较合适。

```
#为了分析其eigenvalue我们先将维数设置为28维（数据集总共28个特征）
pca = PCA(n_components=28)
pca.fit_transform(train)
#分析其eigenvalue，看看要降成几维
eigenvalue=pca.explained_variance_
```

```
array([31.86366621, 14.08570188,  1.22656257,  0.58124522,  0.5594265 ,
        0.53061075,  0.5203859 ,  0.46620109,  0.4569424 ,  0.44604706,
        0.42779386,  0.4157626 ,  0.39070124,  0.34531282,  0.33192607,
        0.32112006,  0.30737716,  0.28239017,  0.28083059,  0.27725171,
        0.26111925,  0.25330198,  0.23606832,  0.22664027,  0.21712987,
        0.21477544,  0.16583797,  0.1456473 ])
```

#因为eigenvalue前两项值很大，后面的值相比都非常小，故将维度降为2维

5.3 教学实验案例(2)——供应商聚类分析

（2）使用PCA对数据进行降维处理

因为数据集中样本的特征值非常多，将近30个，也就是将近30维，所以我们要先对样本特征进行降维处理。在这里，我们采用主成分分析（PCA）来对数据进行降维处理。

首先，我们先要对数据集的eigenvalue进行分析，看看要降成几维比较合适。

```
#为了分析其eigenvalue我们先将维数设置为28维（数据集总共28个特征）
pca = PCA(n_components=28)
pca.fit_transform(train)
#分析其eigenvalue，看看要降成几维
eigenvalue=pca.explained_variance_

#将维度降为2维
pca = PCA(n_components=2)
#将结果生成一个dataframe
data_pca = pd.DataFrame(pca.fit_transform(train))
```

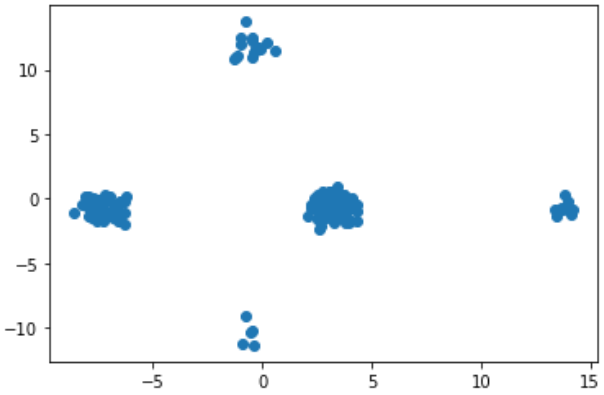
	0	1
0	3.767170	-1.387988
1	3.319501	-0.652953
2	0.193659	12.150278
3	3.126506	-0.594459
4	-7.423938	-1.278174
...	...	...
194	3.171634	-0.108656
195	3.602276	-1.291880
196	-7.238667	-0.008146
197	-6.849328	-0.423364
198	2.988275	0.051548

199 rows × 2 columns

5.3 教学实验案例(2)——供应商聚类分析

(3) 利用K-means算法进行聚类分析

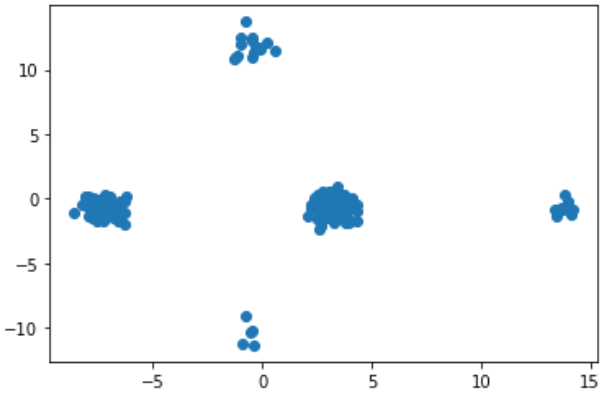
```
#绘制散点图查看数据点大致情况
plt.scatter(data_pca[0],data_pca[1])

#根据上面散点图的分布情况，我们将数据点分类为5类
kmmodel = KMeans(n_clusters=5)      #创建模型
kmmodel = kmmodel.fit(train)        #训练模型
ptarget = kmmodel.predict(train)    #对原始数据进行标注
```



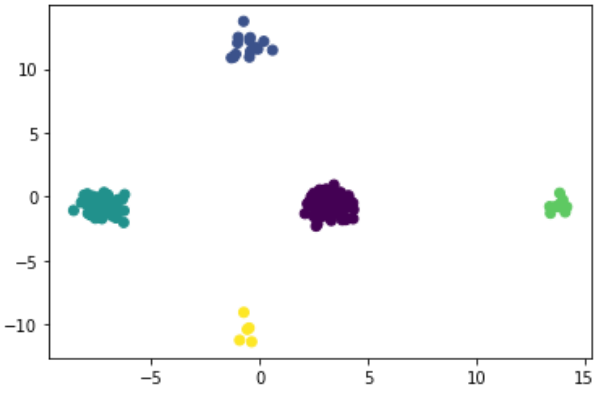
5.3 教学实验案例(2)——供应商聚类分析

(3) 利用K-means算法进行聚类分析

```
#绘制散点图查看数据点大致情况
plt.scatter(data_pca[0],data_pca[1])

#根据上面散点图的分布情况，我们将数据点分类为5类
kmmodel = KMeans(n_clusters=5)      #创建模型
kmmodel = kmmodel.fit(train)        #训练模型
ptarget = kmmodel.predict(train)    #对原始数据进行标注

#交叉表查看各个类别数据的数量
pd.crosstab(ptarget,ptarget)

#查看聚类的分布情况
plt.scatter(data_pca[0],data_pca[1],c=ptarget)

#使用Calinski-Harabasz准则评价模型的拟合度
CH_score_KMeans=calinski_harabasz_score(data_pca, ptarget)
CH_score_KMeans
```



col_0	0	1	2	3	4
row_0					
0	107	0	0	0	0
1	0	15	0	0	0
2	0	0	63	0	0
3	0	0	0	9	0
4	0	0	0	0	5

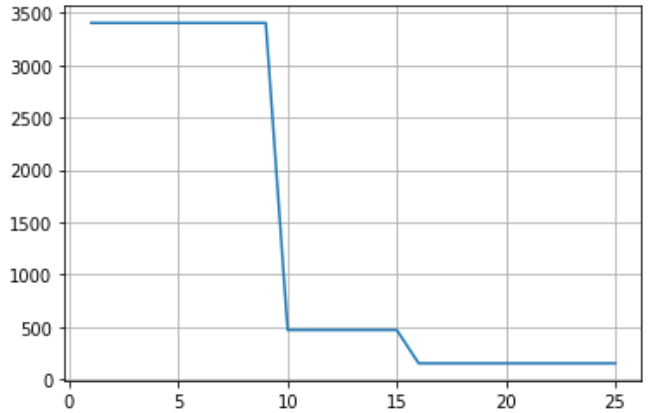


Calinski-Harabasz: 3404.7563

5.3 教学实验案例(2)——供应商聚类分析

(4) 利用DBSCAN算法进行聚类分析

```
#参数选取，固定e=5，MinPts=[1:25]
MinPts_list=[]
CH_score_DBSCAN_list=[]
for i in range(25):
    #邻域的距离阈值
    e = 5
    #样本点要成为核心对象所需要的样本数阈值
    MinPts = i+1
    MinPts_list.append(MinPts)
    #模型训练
    model = DBSCAN(eps=e, min_samples=MinPts)
    #对训练集的数据进行分类
    Type = model.fit_predict(data_pca)
    #使用Calinski-Harabasz准则评价模型的拟合度
    CH_score_DBSCAN=calinski_harabasz_score(data_pca, Type)
CH_score_DBSCAN_list.append(CH_score_DBSCAN)

#对各个MinPts下聚类结果的评分进行折线图绘制
plt.plot(MinPts_list,CH_score_DBSCAN_list)
plt.grid(True)
```

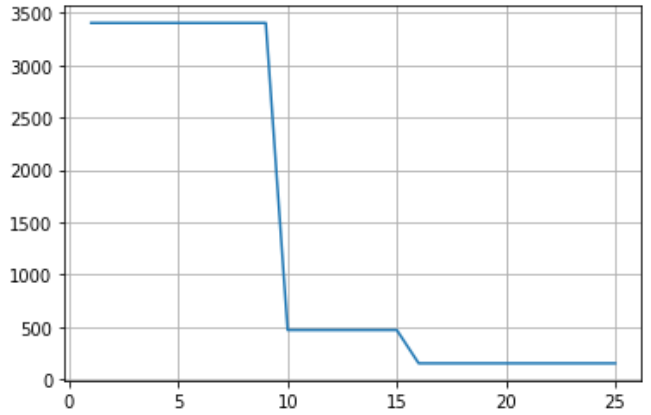


当e=5，且MinPts=[1, 9]时，聚类效果最佳，且聚类效果相同

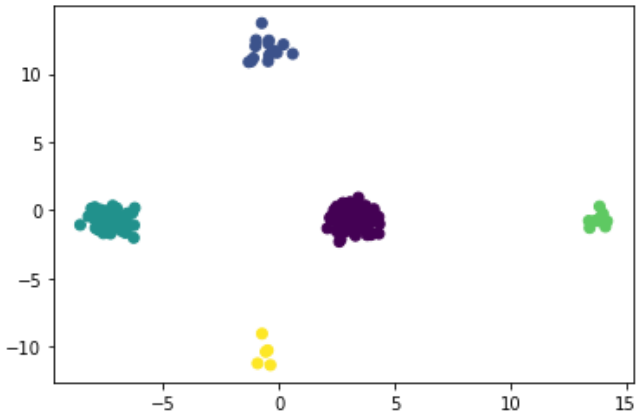
5.3 教学实验案例(2)——供应商聚类分析

(4) 利用DBSCAN算法进行聚类分析

```
#参数设置
e = 5
MinPts = 3

#模型训练
model = DBSCAN(eps=e, min_samples=MinPts)

#对训练集的数据进行分类
Type = model.fit_predict(data_pca)

#交叉表查看各个类别数据的数量
pd.crosstab(Type,Type)

#查看聚类的分布情况
plt.scatter(data_pca[0], data_pca[1], c=Type)

#使用Calinski-Harabasz准则评价模型的拟合度
CH_score_DBSCAN=calinski_harabasz_score(data_pca, Type)
```



col_0	0	1	2	3	4
row_0					
0	107	0	0	0	0
1	0	15	0	0	0
2	0	0	63	0	0
3	0	0	0	9	0
4	0	0	0	0	5

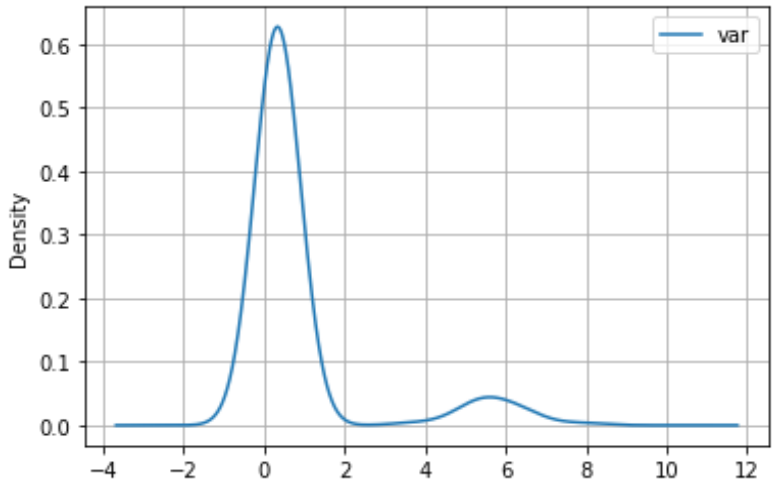


Calinski-Harabasz: 3404.7563

5.3 教学实验案例(2)——供应商聚类分析

(5) 挑选出存在特项异常的供应商，并找出他们的专长项

Step1：找出特项异常供应商



“各个供应商各项评分方差”密度图

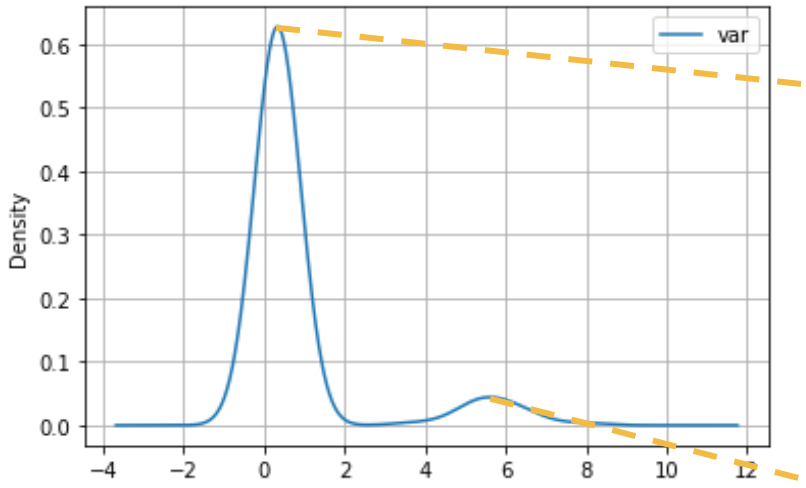
```
#获取各个供应商各项评分方差
var_list=[]
for i in range(199):
    var = np.var(np.array(train[i:i+1]))
var_list.append(var)

#构造dataframe
var_df=pd.DataFrame(data=var_list,columns=['var'])

#绘制密度图
var_df.plot(kind='kde',grid=True)

#显示图表
plt.show()
```

各项评分方差≤1的供应商，为正常的供应商；
各项评分方差>1的供应商，为存在特项异常的供应商。

5.3 教学实验案例(2)——供应商聚类分析

(5) 挑选出存在特项异常的供应商，并找出他们的专长项

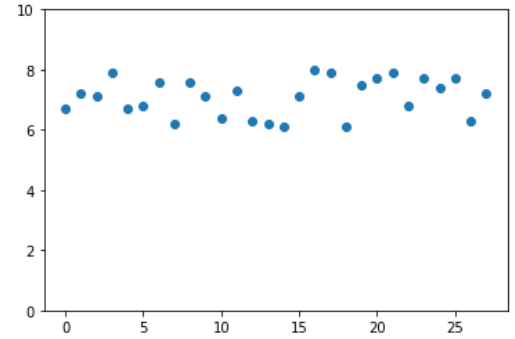
Step1：找出特项异常供应商



“各个供应商各项评分方差”密度图

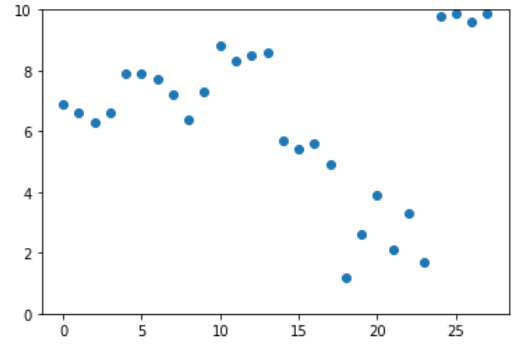
供应商0
方差:0.3752423469387756
均值:7.0892857142857135

正常的供应商



供应商2
方差:6.254642857142858
均值:6.450000000000001

存在特项异常的供应商



各项评分方差≤1的供应商，为正常的供应商；
各项评分方差>1的供应商，为存在特项异常的供应商。

5.3 教学实验案例(2)——供应商聚类分析

(5) 挑选出存在特项异常的供应商，并找出他们的专长项

Step2：分析各个特项异常供应商的专长项

如果一个特项异常供应商某一项的评分排在所有供应商前5%（前10），我们就认为该供应商的该项是它的特项专长。

```
#对各项评分指标进行分析
for f in feature:
    arr=np.array(train[f])
    var = np.var(arr)
    mean = np.mean(arr)

    #挑出该项指标Top10的供应商
    D=train[f].to_dict()
    sorted_D=sorted(D.items(),
    sorted_D=sorted_D[0:10]

    #挑出这Top10里的特项异常供
    sorted_special_D=[]
    for key_value in sorted_D:
        if (key_value[0] in special
            sorted_special_D.appe

    #保存这些特项异常供应商的专
    for kv in sorted_special_D:
        special_suppliers_diction
```

供应商编号	专长项
2	客户满意度、次品率、售后维护点、客户维护期限
5	地理位置、信息传递及时率、客户满意度、售后维护点
18	车辆数量、信息系统应用覆盖率、物流驻点、仓库数量、物流设施的数量
27	车辆数量、信息系统应用覆盖率、物流驻点、仓库数量、物流设施的数量
33	信息传递及时率、客户满意度
52	送货频率、车辆数量、信息系统应用覆盖率、物流驻点、仓库数量、物流设施的数量
60	信息传递及时率、服务价格、技术水平、客户满意度、次品率
63	送货频率、车辆数量、物流驻点、仓库数量
67	地理位置、信息传递及时率、次品率、售后维护点
71	服务价格、技术水平、次品率、客户维护期限
75	技术水平
81	地理位置、技术水平、客户维护期限
116	地理位置、信息传递及时率、服务价格、技术水平、售后维护点、客户维护期限
125	售后维护点
128	地理位置、服务价格
134	地理位置、信息传递及时率、服务价格、客户满意度、次品率、售后维护点
149	年营业额、地理位置、信息传递及时率、服务价格、技术水平、客户维护期限
151	送货频率、车辆数量、信息系统应用覆盖率、物流驻点、仓库数量
164	信息系统应用覆盖率、物流驻点、物流设施的数量
178	地理位置、服务价格、技术水平、客户维护期限

5.4 教学实验案例(3)——客户反馈信息分析

1 实验目的：

- 掌握爬取手机APP数据的方法；
- 掌握使用Python读取json文件的方法；
- 掌握使用jieba库进行中文文本分词的方法；
- 掌握使用wordcloud库进行词云绘制的方法；
- 掌握文本向量化的TF-IDF算法；
- 掌握高维数据降维的t-SNE算法；
- 掌握K-means聚类算法；
- 掌握Python的基本绘图方法。

2 实验环境：

使用DataOne大数据实验平台与Python语言，在spark集群上编写、运行和调试spark程序。

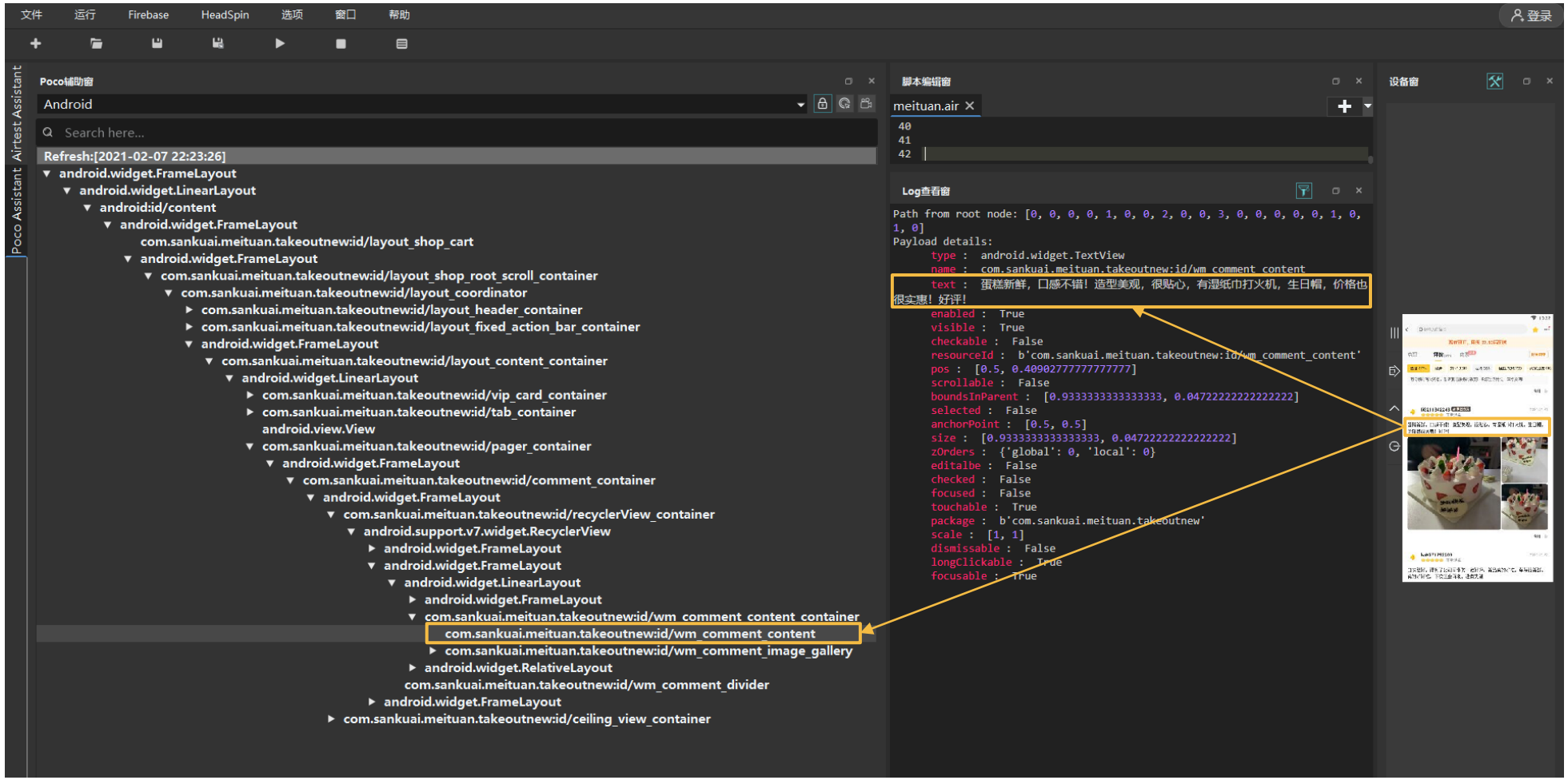
3 实验内容：

- 从“幸福西饼生日蛋糕门店美团评论数据集_1_5k.json”中提取出评价数据；
- 筛选抓取的客户评价，找出评价的高频关键词，并绘制的词云；
- 对抓取的客户评价有效信息进行向量化，采用二维平面对客户评价进行可视化处理。在二维向量可视化图中，彼此相近的点表示具有相似的含义，且用绿色表示正面评价，红色点表示负面评价；
- 对客户评价进行聚类分析，并采用二维平面对聚类结果进行可视化处理。

5.4 教学实验案例(3)——客户反馈信息分析

(1) 爬取美团APP“幸福西饼蛋糕（深圳店）同城配送”门店的外卖评价与到店评价

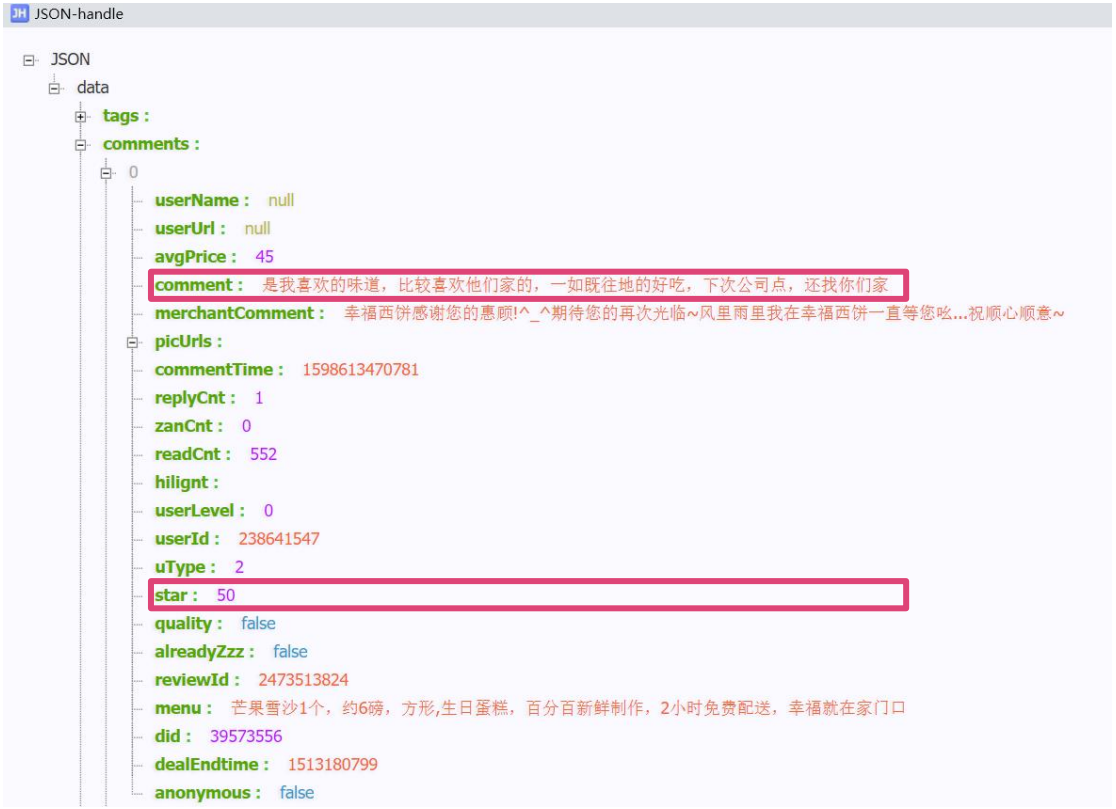
使用Airtest这一UI自动化测试编辑器来对美团APP中的评价进行爬取。



5.4 教学实验案例(3)——客户反馈信息分析

(2) 从“幸福西饼生日蛋糕门店美团评论数据集_1_5k.json”中提取出评价数据

```
#打开json文件
file=open("幸福西饼生日蛋糕门店美团评论数据集_1_5k.json",
"r", encoding="utf-8")

#读取json文件的内容，生成一个字典
dic=json.load(file)

#关闭json文件
file.close()

#从字典中获取每一条评论的内容
for item in dic.get("data").get("comments"):
    print(item.get("comment"))
```



p.s. 因为不同数据集的分析流程都一样，故后面的实验步骤只基于该数据集。

5.4 教学实验案例(3)——客户反馈信息分析

(3) 筛选抓取的客户评价，找出评价的高频关键词，并绘制词云

文本预处理：

- 对评论中的特殊字符进行剔除，只保留中英文和数字
- 对评论中的停用词进行剔除

```
# 结巴中文分词，生成字符串
cut_text_waimai = jieba.cut(text_waimai,cut_all=False)
# 用空格分隔开分词结果，形成字符串，用以绘制词云
result_waimai = " ".join(cut_text_waimai)
# 生成词云图
wc = WordCloud(font_path="simfang.ttf", # 设置字体
                    background_color='white', # 设置背景色
                    width=250, # 设置背景宽
                    height=175, # 设置背景高
                    max_font_size=50, # 最大字体
                    min_font_size=10, # 最小字体
                    collocations=False, # 避免重复统计词语
                    mode='RGBA')
# 产生词云
wc.generate(result_waimai)
# 以图片的形式显示词云
plt.imshow(wc)
# 关闭图像坐标系
plt.axis("off")
plt.show()
```

5.4 教学实验案例(3)——客户反馈信息分析

(4) 利用TF-IDF模型对客户评价信息进行向量化

```
#将文本转为词频矩阵
frequency=vectorizer.fit_transform(corpus)

#计算tf-idf
tfidf=transformer.fit_transform(frequency)

#将tf-idf矩阵抽取出来
weight=tfidf.toarray()
```

id	一下子	一如既往	下午茶	下次	不腻	不贵	不错	两磅	个性化	五星	价格	便宜	值得
0	0	0.3249259	0	0.288596	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0.235653	0	0	0	0
4	0.218811	0.1287407	0	0.114346	0	0	0	0.193168	0	0.196785	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0.168613	0	0
7	0	0	0	0	0	0	0.414651	0	0	0	0	0	0
8	0	0.6359492	0	0	0	0	0	0	0	0	0	0	0
9	0	0.7104067	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0.173381	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0.266394	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0.482084	0	0	0.65114	0	0	0
17	0	0	0	0	0	0	0.134677	0	0	0	0	0.288859	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0.576112	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0.264734	0	0	0	0	0	0
24	0	0	0.264169	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0.45791
29	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0.6174	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0
32	0	0	0.416113	0	0	0	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0	0	0	0	0	0	0
34	0	0	0	0	0	0	0	0	0	0	0	0	0

后续处理：

- 将dataframe中一些无意义的词项剔除
- 将dataframe中一些TF-IDF得分和为0的评论剔除

5.4 教学实验案例(3)——客户反馈信息分析

(5) 利用T-SNE模型将向量化的评价信息的特征维度降至二维

进行10次T-SNE降维，将其Kullback-Leibler离散度作为衡量指标，取KL离散度最小的一次作为降维的结果。

```
for i in range(10):
    #使用t_sne降成2维
    tsne=TSNE(n_components=2, perplexity=50)
    result=tsne.fit_transform(np.array(train))
    result_list.append(result)
    #计算KL离散度（越小越好）
    kl=tsne.kl_divergence_
    if kl<min_kl:
        min_kl=kl
        best_id=i

#构造dataframe
best_result=result_list[best_id]
tsne_train=pd.DataFrame({"feature1":list(best_result[:, 0]),
"feature2":list(best_result[:, 1])})
```

	feature1	feature2
0	23.237724	8.133894
1	11.821193	-8.671532
2	-2.111201	-0.686433
3	7.280168	-34.102917
4	-4.011161	-9.744668
...	...	...
4876	16.088757	-5.072202
4877	-1.086728	-21.216061
4878	-7.892528	-2.125502
4879	-14.761262	12.668215
4880	-15.184138	17.785509

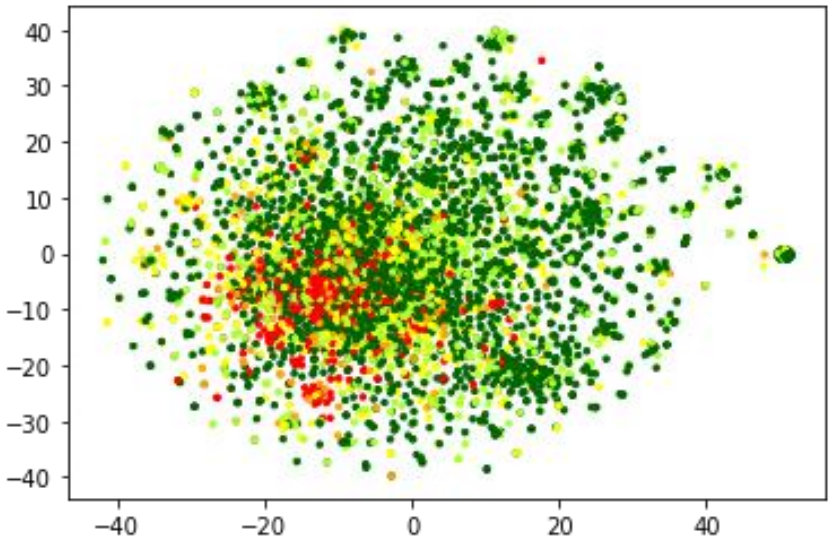
4881 rows × 2 columns

5.4 教学实验案例(3)——客户反馈信息分析

(6) 采用二维平面对客户评价进行可视化处理

根据每条评论中的评分对评论进行二维可视化处理，其颜色所代表的含义为：深绿（非常满意）、绿（满意）、黄（一般）、橙（差）、红（非常差）。

```
#设置颜色列表
label_color=[]
for i in label_list:
    if i=="非常满意":
        label_color.append("DarkGreen")
    elif i=="满意":
        label_color.append("GreenYellow")
    elif i=="一般":
        label_color.append("Yellow")
    elif i=="差":
        label_color.append("Orange")
    elif i=="非常差":
        label_color.append("Red")
```



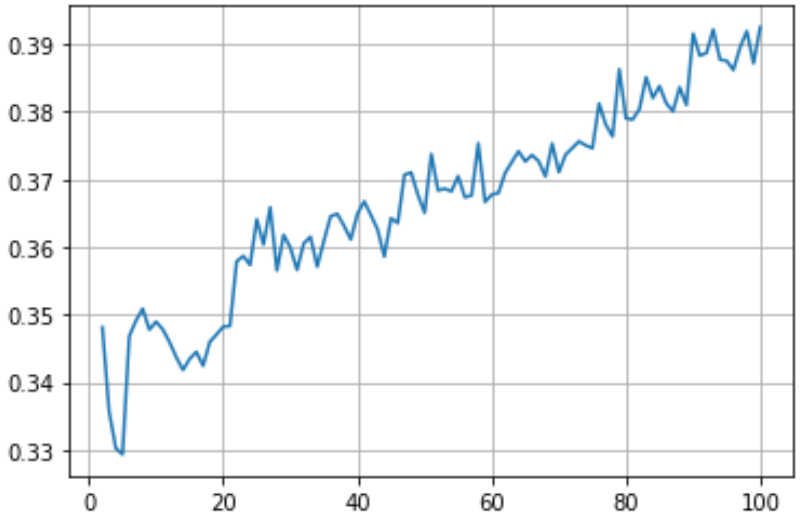

```
#将评论可视化，并将不同评分的评论用不同颜色进行标注
plt.scatter(best_result[:, 0], best_result[:, 1], 20, np.array(label_color))
plt.show()
```

5.4 教学实验案例(3)——客户反馈信息分析

（7）对客户评价进行聚类分析，并采用二维平面对聚类结果进行可视化处理

我们采用轮廓系数作为聚类个数选取的衡量指标。

```
#K-means聚类(2-100)
for k in list(range(2,101)):
    #创建模型
    kmmodel = KMeans(n_clusters=k)
    #训练模型
    kmmodel = kmmodel.fit(tsne_train)
    #对原始数据进行标注
    target=kmmodel.predict(tsne_train)
    target_list.append(target)
    #使用轮廓系数评价模型的拟合度
    S_score=silhouette_score(tsne_train, target)
S_list.append(S_score)

#对各个聚类结果的轮廓系数进行折线图绘制
plt.plot(list(range(2,101)),S_list)
plt.grid(True)
plt.show()
```



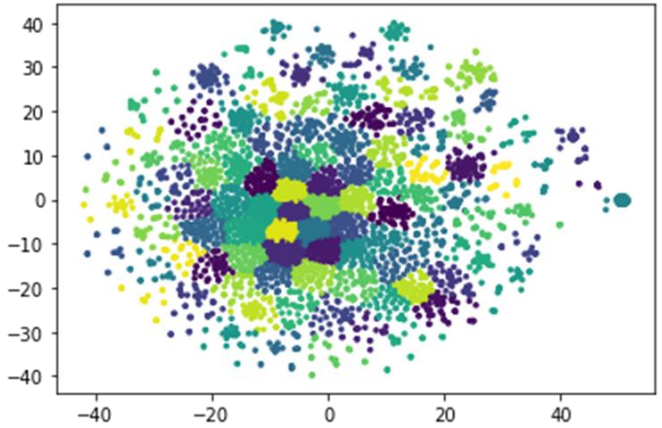
虽然其轮廓系数总体是随着聚类个数的增加而增加，但有时聚类个数过多并没有什么实际意义。因为此时数据规模为5000左右，所以我们就选取聚类个数为2-100时轮廓系数最大的那个结果，即聚100个类。

5.4 教学实验案例(3)——客户反馈信息分析

（7）对客户评价进行聚类分析，并采用二维平面对聚类结果进行可视化处理

选取得分评价最高的那个聚类结果（簇个数为100）。

```
target = target_list[max_score_id]
plt.scatter(tsne_train["feature1"],tsne_train["feature2"],c=target)
```

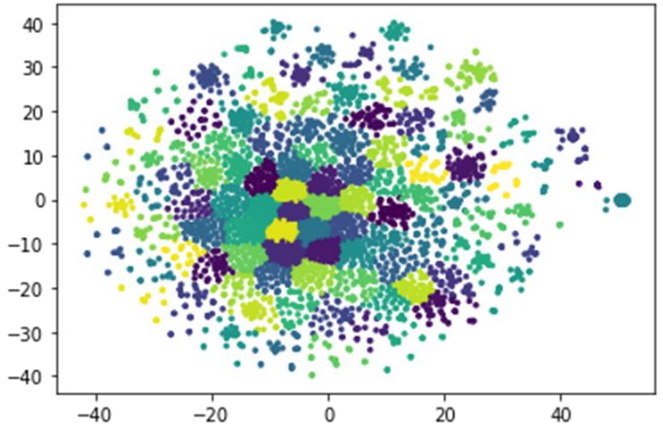


5.4 教学实验案例(3)——客户反馈信息分析

（7）对客户评价进行聚类分析，并采用二维平面对聚类结果进行可视化处理

选取得分评价最高的那个聚类结果（簇个数为100）。

```
target = target_list[max_score_id]
plt.scatter(tsne_train["feature1"],tsne_train["feature2"],c=target)
```



在聚类得到的每一类中，选取TF-IDF指标总和最大的那3个词作为该类的主题。

```
for i in range(num_of_cluster):
    cluster_tfidf=train.iloc[cluster_dictionary[i]]
    sum_c_tfidf=np.sum(np.array(cluster_tfidf),axis=0)
    clister=pd.Series(sum_c_tfidf,words)
    clister=clister.sort_values(ascending=False)
    theme=clister.index.to_list()
    print("第"+str(i+1)+"个类的主题："+theme[0]+"
"+theme[1]+" "+theme[2])
```

序号	主题			序号	主题			序号	主题			序号	主题		
1	幸福	西饼	蛋糕	26	活动	划算	涨价	51	千层	榴莲	芒果	76	服务周到	想到	味道
2	蛋糕	好吃	不错	27	服务	小哥	妈妈	52	草莓	送到	时间	77	小时	蛋糕	承诺
3	榴莲	千层	芒果	28	习惯	可口	不错	53	太甜	一块	不错	78	挺不错	小孩	喜欢
4	榴莲	太甜	蛋糕	29	支持	一如既往	很多年	54	购买	不错	值得	79	分量	味道	价格
5	口味	很快	送货	30	超级	继往	一如	55	四重奏	首选	蛋糕	80	味道	不错	满意
6	实惠	孩子	价格	31	蛋糕	依然	现做	56	好评	一如既往	不错	81	太腻	准确	好不好
7	味道	不错	准时	32	西饼	幸福	蛋糕	57	味道	不错	新鲜	82	完美	祝福语	蜡烛
8	蛋糕	好吃	幸福	33	满意	好吃	不腻	58	送货	送到	好吃	83	超级	好吃	第二次
9	满意	不错	包装	34	蜡烛	生日	生日蛋糕	59	幸福	西饼	蛋糕	84	一如既往	好吃	味道
10	电话	商家	难吃	35	口味	服务	准时	60	客服	打电话	蛋糕	85	不好	拿破仑	蛋糕
11	好吃	美味	蛋糕	36	男朋友	送货员	态度	61	粉丝	忠实	西饼	86	退款	小时	蛋糕
12	榴莲	千层	英寸	37	好吃	蛋糕	真心	62	性价比	米苏	提拉	87	服务态度	送货上门	特别
13	生日蛋糕	感谢	莓莓	38	棒棒	好吃	味道	63	价格	公司	精致	88	不错	味道	款式
14	味道	全心全意	奶油	39	蛋糕	生日	不错	64	便宜	信赖	好吃	89	好吃	蛋糕	喜欢
15	四种	口味	味道	40	朋友	家里人	很足	65	下次	味道	真的	90	幸福	西饼	蛋糕
16	很赞	味道	这次	41	小朋友	喜欢	他家	66	开心	漂亮	反正	91	难吃	一块	好食
17	送货	准时	不错	42	这家	蛋糕	每次	67	好好	哈哈	赞赞	92	好吃	榴莲	好看
18	幸福	西饼	每次	43	芝士	半熟	送到	68	姐姐	即往	男票	93	几次	强烈推荐	买过
19	不错	好吃	蛋糕	44	可惜	麻烦	卡片	69	速度	好吃	食材	94	小贵	好几个	一年
20	两个	吃不完	送人	45	物美价廉	回购	蛋糕店	70	还会	下次	光顾	95	蛋糕	送到	时间
21	好看	好吃	优惠活动	46	好吃	不到	口味	71	优惠	价格	团购	96	还好	味道	购价
22	越来越	蛋糕	西饼	47	新鲜	水果	好吃	72	美味	还来	下次	97	发票	感谢	周到
23	还行	想象	说好	48	顾客	全家	好吃	73	送达	准时	不错	98	好腻	店里	客户
24	挺好吃	不错	蛋糕	49	服务	下次	好吃	74	好吃	榴莲	很浓	99	每次	生日	蛋糕
25	朋友	生日	蛋糕	50	芒果	榴莲	好吃	75	榴芒	双拼	蛋糕	100	榴莲	小孩子	喜欢

谢 谢

深圳大学计算机与软件学院
2021年5月