云计算课程实验

一、 实验目的及背景：

A. 实验目的：

熟悉云计算平台 Spark 的基本原理，熟悉编程环境和 Spark API。

B. 实验背景

公司最近有大批人员离职，严重影响运作。于是 HR 找到你们团队，希望你们能通过历史数据推测出离职的原因（实验 1,2）。并预测在职员工是否可能会离职（实验 3）。

二、 实验环境及数据说明：

A. 实验环境：

必须在 **Spark** 平台下进行试验，以 ubuntu16.04 下的 pyspark 为例介绍如何搭建试验环境。不同平台也可自行寻找配制方法：

1. 安装 Java，python，ssh 并配置 java 环境变量
2. 安装 hadoop2.7.4 并配置
   http://www.apache.org/dyn/closer.cgi/hadoop/common/hadoop-2.7.4/hadoop-2.7.4.tar.gz
   http://www.cnblogs.com/kinglau/p/3794433.html
3. 安装 Spark 并配置
   https://spark.apache.org/downloads.html
   https://www.cnblogs.com/lijingchn/p/5573898.html
4. 安装 pyspark
5. （可选）安装 IDE 环境
   http://blog.csdn.net/fishseeker/article/details/70188167

B. 数据说明

CSV 文件收集了 14999 个职工的工作情况，包括在职员工和离职员工。每一行代表一个员工样本的信息，用 10 个变量表示员工最近状态。

文件中对应的数据标签如下：

satisfaction_level	**last_evaluation**	**Number_project**	**average_montly_hour**	**time_spend_company**	**work_accident**	**promotion_last_5years**	**occupation**	**salary**	**left**

分别对应的含义为：

职位满意度	最近绩效	项目数量	月均工时	工作年数	是否有工作意外	**5** 年内升职情况	职位类别	薪水	离职与否

三、 实验流程：

本实验由三个小实验（实验 **1~3**）和一道问答题组成。

建议在正式实验之前了解一下 Spark 的特点以及 Spark RDD 数据类型。

实验 **1**（**20%**）：

以 0.2 为区间，请找出各个满意度（satisfaction_level）区间下离职人数占该区间总人数的百分比。

实验 **2**（**30%**）：

查看以下特征（last_evaluation，avg_montly_hour，time_spend_company，

occupation,salary）与离职率的关系。你觉得哪些特征与离职率的关系最明显？请自行设计试验，并通过对比不同特征间的结果（可以参考实验 1）。

实验 **3**（**30%**）：

参考附加资料 A，使用 **KNN(K=5)**来预测以下员工样本是否可能会离职，并解释原因。（使用 CSV 文件中的全部数据作为训练集，使用 **satisfaction_level** 和 **last_evaluation** 为特征值（2 维空间中），距离采用欧式距离。）

satisfaction_level	last_evaluation	number_project	average_montly_hour	Time_spend_company	Work_accident	promotion_last_5years	occupation	salary
0.79	0.90	4	262	4	0	0	hr	low

问答题（**20%**）：

当处理真实世界中的数据分析任务时，数据量可能会非常大。此时，你的实验环境及所编写的代码可能会遇到哪些问题？可以通过什么方法来避免这些问题?

四、 评分标准：

1. 需要将代码和实验报告（2 页左右）一并打包（zip 或 rar）并提交到助教邮箱，每组提交一份，同组同分。打包文件命名方式（**CC2017_**组长名称**_**组员 **1** 名称**_**组员 **2** 名称**.rar**）。
2. 报告整洁度，实验设计的清晰度，实验结论的完整度（图表、解释说明等）会影响评分。
3. 请在 Spark 提供的 RDD、DataFrame 和 DataSet 数据结构的基础上使用提供的 API 进行操作。未完全使用 Spark API 会酌情扣分。

附加资料

A. KNN 算法概要

（1） KNN 是机器学习和数据挖掘中的经典分类算法。

（2） KNN 算法概要：

假设有一堆标签已知的样本集，每个样本在空间中的用 x 表示，对应标签为 y。则对于第 i 个样本可表示为 $\{x_i, y_i\}$，对于这些标签已知的样本集我们称之为训练样本集。KNN 的目的是根据训练样本集，预测新来的标签未知的样本。

KNN 算法过程非常简单，就是将标签未知的样本映射到空间中。计算该未知标签样本与各个训练样本的距离，选择距离最近的 K 个训练样本中占绝大多数的标签作为判定值。

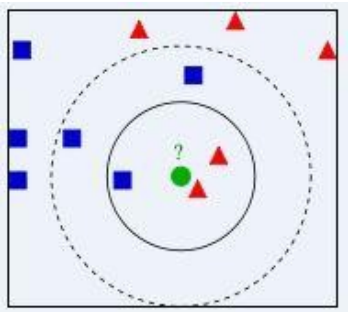

图 1.KNN 算法

图 1 很好的描述了 KNN 算法。样本处于二维空间当中。标签分为两类，用红色三角和蓝色方框表示。此时我们有一个标签未知的样本（用绿色圆圈表示），KNN 根据最近 K 个训练样

本的标签来进行预测。对于 KNN（K=3）判定为红色三角（黑色实线圆圈），对于 KNN（K=5）判定为蓝色方框（黑色虚线圆圈）。

B. **Spark API**：http://spark.apache.org/docs/latest/api.html
C. **Others**：
http://blog.csdn.net/stark_summer/article/details/50218641