实验一：数据预处理

1. 实验目的

掌握数据预处理的步骤和方法，包括数据抽样、数据过滤、数据标准化和归一化、数据清洗。理解数据预处理各个步骤在大数据环境下的实现方式。

2. 实验环境

操作系统：Windows、Linux(建议)
框架：伪分布式Hadoop环境
编程语言：Java

3. 实验内容

搭建伪分布式Hadoop环境，将数据集合D导入至HDFS中，然后完成对D的数据预处理，具体过程如下。

3.1 数据抽样

1. 本次实验采取分层抽样的方式，选取D中的某一属性A，按A进行分层抽样，将抽样结果保存至HDFS中命名为D_Sample；
2. Tips：在Map阶段以属性A作为Key，然后在Reduce阶段进行抽样，如果属性是连续属性，可以考虑进行离散化；

3.2 数据过滤

1. 选取D_Sample中的某一连续属性B进行排序，我们认为取值排名在1%~99%之间的值为正常值，即其它值为奇异值，我们需要过滤掉**原始数据集合D**中属性B取值为奇异值的数据；
2. 过滤后的数据保存在HDFS中，命名为D_Filter；

3.3 数据格式转换与归一化

1. 一些数据属性可能存在不同的格式，如日期、温度，我们需要转化为统一的格式；另一些属性则往往需要归一化；
2. 对D_Filter中的数据进行格式转换与归一化，结果保存在HDFS中，仍然命名为D_Filter；
3. 本实验中建议使用的归一化方式为Min-Max归一化：

$$x^{new} = \frac{x - min(x)}{max(x) - min(x)}$$

3.4 数据清洗（缺失值填充）

1. D_Filter中某些数据在属性E上可能存在缺失，我们需要使用某种方法对E进行填充，然后将数据保存在HDFS中，命名为D_Done；
2. 填充策略可以考虑使用默认值，也可以使用平均值、中位数，还可以利用相似度寻找与缺失数据相似的其他数据，然后借此对缺失的数据进行填充；

4. 分数说明

1. 完成实验内容3.1~3.3，并以默认值填充方式完成3.4，可以获得本实验总分的80%；
2. 相比默认值填充，我们认为考虑数据的相关性而进行填充是合理的，因此将视使用的填充算法，获得本实验总分的2%~5%；
3. 显然的，我们可以使用多个MapReduce过程来完成3.1~3.4，过多的MapReduce过程会带来更多的资源消耗，因此思考如何尽可能地减少MapReduce的轮数是有价值的，我们将视减少的轮数和策略，给予本实验总分5%~15%；

5. 数据说明

参考QQ课程群中的数据文件说明。

6. FAQ

Q：必须使用Linux操作系统吗？
A：建议而不强制，但从经验上来讲，Linux系统在环境搭建上更为友好，同时在硬盘空间足够的情况下不建议使用虚拟机（性能可能会引起不适），但OS的选择不会影响分数的获得；如果电脑的空间不足以支撑实验的完成，可以考虑购买学生版阿里云/腾讯云服务器，虽然配置不高，但完成实验绰绰有余；

Q：必须使用Hadoop+Java吗？
A：同样是建议而不强制，因为使用Java和Hadoop能更好的理解本实验想要表达的信息，也可以借助Hadoop-Streaming使用Python等语言完成，甚至可以使用Spark框架完成实验，但考虑到这些并不完全契合我们所设计的实验过程和工作量的加大，评分标准可能会因此而上升；

Q：4.3中关于MapReduce轮数减少的分数获取，是否轮数越少越好？
A：原则上我们鼓励使用更少的轮数完成更多的工作，但如果因此而使用了不合理的设计（如只使用一个Key而导致失去了MapReduce本来的意义），是不可取的，本质上我们更看重同学们思考的过程和对MapReduce过程的理解，只要思路优秀即可；缺失值填充算法也是如此，如何在MapReduce的轮数和算法的复杂度之间做出平衡也是同学们需要考虑的问题；

Q：3.1~3.4中所涉及的属性A、B等是固定的吗，比如是否可以用其他属性分层抽样？
A：不可以，这一点请参考数据文件中的说明并严格按照说明执行；