RandomForest

项目结构

```
.
├── RandomForest
|   ├── CMakeLists.txt
|   ├── RandomForest
|   |   ├── DesionTree.cpp
|   |   ├── DesionTree.h
|   |   ├── FileProcesser.cpp
|   |   ├── FileProcesser.h
|   |   ├── Message.cpp
|   |   ├── Message.h
|   |   ├── RandomForest.cpp
|   |   ├── RandomForest.h
|   |   ├── Run.cpp
|   |   ├── Run.h
|   |   ├── Util.cpp
|   |   ├── Util.h
|   |   ├── main.cpp
|   |   ├── tools.cpp
|   |   └── tools.h
|   ├── Util
|   |   └── util
|   ├── python
|   |   ├── splitData.py
|   |   └── versionData.py
|   ├── readme.md
|   └── run.sh
├── data
├── doc
|   ├── 随机森林-项目报告.pdf
|   └── 配置说明文件.pdf
├── result
└── version
```

依赖

```
mpich2 version >= 3.1.4
g++ version >= 4.1.2
python version >= 2.7
cmake version == 3.0
```

注意，本程序是脚本直接编译运行，需要运行脚本要先安装好上述依赖。因为程序有依赖 mpi

的库文件，所以直接命令编译会出现链接错误。项目里面的 cmake 文件已经设置好依赖关系。所以请安装好上述依赖然后直接运行脚本。

配置

程序的配置文件在 `Util` 文件夹下的 util 文件内。配置变量的描述如下：

```
CVTEST:1  该变量表示是否进行测试。测试是用 trainFin.csv 进行模型的训练，然后用 testCV.csv 进行模型的测试。该值默认为 1，如果不希望进行测试，则将该值改为 0.

PREDICTTEST:0  该变量表示是否进行 test.csv 文件的预测。test.csv 文件是项目测试文件。该值默认为 0，如果需要进行预测，则将该值设置为 1. 预测结果是 result 文件夹下的 result.csv 文件。

CHOSEFEATUREBEFORERUN:0  该变量表示是否进行特征选择操作。特征选择操作的介绍在项目报告有详细说明。该值默认为 0。如果要进行特征选择操作，将该值设为 1。但是要注意的是，进行特征选择操作会花费大量时间，如无必要不要将该值设为 1.

FEATURETHRESHOD:0.001  默认特征选择的阈值

TREENUM:50 该变量表示随机森林的建树的数量，默认是 500 棵。如果需要改变树的数量，直接将该值设置为想要的值即可。

DATASCALE:0.8  该变量表示每棵树的数据集比例。也就是一棵树的数据集数量占原数据集的比例。默认为 0.8.

FEATURESCALE:0.15 该变量表示每棵树的特征列比例。 默认为 0.15
```

注意，上述设置中变量名和值之间以 `:` 相隔，两者之间不应该再有其他符号

运行

在设置好配置文件后，首先将 train.csv 和 test.csv 两个数据文件放到 `data` 文件夹（注意一定要先将这两个文件放到 data 文件夹内，否则程序会执行失败），然后在 `RandomForest` 文件夹下运行如下命令：

```
$bash run.sh
```

这个命令会运行脚本 run.sh。该脚本会先运行两个 python 代码进行测试文件创建和数据可视化的操作。然后进行代码编译。最后运行程序。

注意：

- 如果你的计算机不支持同时开 11 个进程，那么可以通过修改 run.sh 脚本的 35 行的 -np 后面的参数。它代表你设置的进程数。但是这个值最小为 2.不能设置为 1. 原因是程序是主从式控制结构，最少一个控制进程一个执行进程。

- 本 Project 的开发环境是 OS X，运行脚本可以在含有 bash 环境的 *nix 系统下运行。但是该脚本无法直接在 windows 环境下执行。所以在 windows 下需要先在 vs 里面建立工程然后再执行。建议直接用 OSX 或 linux 系统执行。
- 可执行文件是在 OS X 下编译完成的，如果要直接执行可执行文件，可以在 `RandomForest` 文件夹下执行

  ```
  $time mpirun -np 11 bin/main
  ```

  命令（其中的 time 命令是用来计时的，可以去掉）。当然你可以自己控制进程数，也就是修改 -np 后面的参数。这个值也不能小于 2. 这个可执行文件在其他系统中是无法直接运行的。如果你的操作系统不是 OS X，那么请先编译一个新的可执行文件。

运行结果

可视化结果在 `version` 文件夹。

如果将配置中的 `CVTEST` 设置为了 1，那么测试结果直接在终端输出，输出值是错误率。

如果将配置中的 `PREDICTTEST` 设置为了 1，那么 test.csv 的运行结果在 `result` 文件夹，文件名为 result.csv.

可视化

在这里我使用 python 对数据进行可视化处理，随机选择 10 列数据，各自与 label 构成二维坐标，然后画出结果。可以在 RandomForest 文件夹下直接运行如下命令：

```
$python versionData.py
```

来获得结果，但是在 run.sh 脚本里面已经包含这条命令，所以可以通过直接运行 run.sh 脚本来获得结果。 注意，可视化结果在 version 文件夹下面。

Author

E-mail: xiezhw3@163.com

Github: www.github.com/xiezhw3