- 说明：标题后带**的内容需要在tutorial上完成；带*的内容可以其他时间进行，但必须完成；其他内容可选或根据说明完成
- 遇到问题尽可能独立通过搜索相关教程解决

1 准备

- 获得pycharm专业版：申请github教育版大礼包 -> 用github账户申请jetbrain大礼包 -> 下载pycharm专业版 （专业版优势：在pycharm上运行jupyter，本地pycharm可连接远程ssh运行代码等）

2 运行前配置**

2.0 集群通信环境准备*

- windows用户安装Git for Windows，使用自带的Git Bash测试ssh连接（亦可使用其他ssh客户端，如：Putty）
- SFTP客户端（windows下如：WinSCP，linux下如FileZila）
- **测试sftp上传/下载文件，ssh登录**

2.1 Git工具配置*

- 可以使用有GUI界面或命令行的git
- GUI界面git：windows可以使用Github Desktop，linux可以使用GitKraken
- 命令行git：使用bash中的git命令（使用git命令前建议使用ssh-keygen并添加public key到github上）
- **完成github上创建一个repository，并使用git工具完成一次pull和push操作**

2.2 Python环境配置**

- 集群上和本地都需要安装anaconda3（集群上可能已经安装）
- 本地安装Pycharm作为IDE

2.3 tensorflow安装**

- 找到anaconda3下的python（windows下比如C:\ProgramData\Anaconda3\python.exe，linux下如~/anaconda3/bin/python），命令行中使用pip安装tensorflow。如

```
C:\ProgramData\Anaconda3\python.exe -m pip install tensorflow
```

- 亦可将上述python所在目录加入环境变量，直接使用pip install XXX

2.4 本地工作目录和pycharm配置**

- 建立一个目录作为pycharm的项目目录
- 建立pycharm项目：可以使用右键Open Folder as Pycharm Project建立，也可打开Pycharm找到工作目录建立

- 设置Interpreter：（windows为例）在pycharm中File - Settings - Project:XXX - Project Interpreter - 右上角... - Add... - System Interpreter (ssh interpreter也是在相应位置设置) - 选择对应目录下的python（之前运行pip的那个python）
- 创建python文件：点开IDE左侧Project栏，选中项目文件夹，右键-New ... 创建一个python file（建议多用IDE创建、删除和重命名文件，而非在系统的文件管理器中进行）
- **在PyCharm上跑通anaconda3中的python，确保import numpy，pandas，sklearn，tensorflow等正常**

2.5 本地jupyter配置**

- Pycharm专业版配置本地jupyter：在pycharm的左边栏project处，右键一个文件夹 - New - jupyter notebook创建空文件，随便写一些代码，点三角运行，ide会提示安装jupyter相关组件，安装完成后可以直接运行，如果jupyter服务没有启动ide会自动启动
- 非专业版Pycharm需要手动配置jupyter环境，可自行搜索教程，如 https://blog.csdn.net/yibo492387/article/details/78774578

2.6 集群上jupyter配置*

- ssh连接集群，安装jupyter相关组件，保证jupyter-notebook命令能够运行
- 参考命令启动服务器端jupyter notebook:

```
jupyter-notebook --ip 10.15.85.198 --port 8890
```

- 运行后浏览器中访问相应ip+port，如10.15.85.198:8890，或者复制jupyter notebook运行后的地址（亦可通过在PyCharm中粘贴相应地址，实现在PyCharm上运行）

3 运行示例jupyter代码LR-Sklearn LR-TF**

1. 测试代码能否全部跑通
2. 打印变量，理解代码含义

4 额外任务

内容

- 完成《Hands-On Machine Learning with Scikit-Learn and TensorFlow》这本书的“End-to-End Machine Learning Project”的代码理解
- 参考https://blog.csdn.net/boywaiter/article/details/86539880

要点

1. 使用pandas读取csv文件
2. DataFrame对象的功能（建议查阅pandas的API文档：https://pandas.pydata.org/pandas-docs/stable/reference/frame.html）

3. 使用matplotlib进行简单的直方图、散点图可视化（matplotlib gallery：https://matplotlib.org/gallery/index.html）
4. 训练集和测试集分割
5. 对label数据进行onehot编码
6. 对特征进行归一化、标准化
7. 使用LR和决策树进行训练，并评估模型在测试集上的表现
8. 进行交叉验证
9. 模型的保存和载入

5 project环境配置：conda虚拟环境中rdkit及deepchem的安装（集群和本地都需要以同样方式安装，参与project的同学需要安装）

- 首先安装rdkit，参考http://www.rdkit.org/docs/Install.html#how-to-install-rdkit-with-conda使用anaconda安装带rdkit的conda虚拟环境
- 完成安装过后，记下所安装虚拟环境中的Python路径，以windows为例

```
C:\Users\wang\AppData\Local\conda\conda\envs\my-rdkit-env\python.exe
```

- 参考deepchem提供的代码，使用pip安装deepchem，注意需要使用有rdkit环境的python进行安装，也就是使用

```
C:\Users\wang\AppData\Local\conda\conda\envs\my-rdkit-env\python.exe -m
pip install joblib pandas sklearn tensorflow pillow deepchem
```

(注意集群上安装tensorflow-gpu版本)

- 在pycharm中的Interpreter设置中添加新安装的env中的python