在本地和 GPU 集群上使用 jupyter notebook

刘震（liuzhen@ihep.ac.cn）　2021.10.12

Jupyter notebook 可以实现 python，ROOT, C++等程序的交互式和可视化的处理。
可以在高能所本地计算环境或者 slurm GPU 集群上使用 jupyter notebook，通过 GPU 集群加速运行一些不太耗时间的机器学习等程序，便于程序的查看和调试。

如果不使用机器学习和 GPU 集群的话，可以跳过安装以及关于 GPU 的部分。参考步骤（二）并且在 anaconda 的 root622 环境里直接使用 jupyter notebook：
https://juno.ihep.ac.cn/~offline/Doc/user-guide/appendix/anaconda.html

（一）安装 jupyter notebook 和机器学习框架 pytorch。先激活 anaconda 环境（source /cvmfs/juno.ihep.ac.cn/sw/anaconda/Anaconda3-2020.11-Linux-x86_64/bin/activate）
可以在 root622 环境里安装机器学习框架 pytorch，或者新建一个环境，在里面安装 pytorch 和 jupyter。这里以新建一个名叫 cnn 的环境为例子：

1 修改虚拟环境安装位置到/hpcfs （默认会安装到/afs）
"conda config --add envs_dirs ${dir}"，（例如${dir}可以是/hpcfs/juno/junogpu/…./envs）
"conda config --add pkgs_dirs ${dir}"
（删除：add 替换为 remove）

2 安装 jupyter lab, pytorch, nb_conda, matplotlib 等等
（pytorch 建议安装 1.7.1，因为 GPU cuda 版本是 11.0 及以下： conda install pytorch==1.7.1 torchvision==0.8.2 torchaudio==0.7.2 cudatoolkit=11.0 -c pytorch）

3 修改 jupyter 的 config 等目录地址到/hpcfs：
export JUPYTER_CONFIG_DIR=${jfile}
export JUPYTER_PATH=${jfile}
export JUPYTER_RUNTIME_DIR=${jfile}
（${jfile} 是放在/hpcfs 的目录就行，可以写在提交 jobs 的脚本里面只用于 GPU 集群上运行的情况，也可以写在.bashrc）

4【补充】 GPU 集群里运行 jupyter 出现不能读写的问题：解决办法 disable sqlite writing to the filesystem：

4.1 在安装好的 jupyter 环境里运行命令"jupyter notebook --generate-config"，生成 jupyter config 文件 jupyter_notebook_config.py
4.2 生成的文件在上面第 3 步里的${jfile}
4.3 在 jupyter_notebook_config.py 里面添加下面这句话就可以了
"c.NotebookNotary.db_file=':memory:'"

（二）在本地使用 jupyter notebook（from：林韬老师）

1 Anaconda 环境里运行"jupyter notebook --no-browser"

2 xhell 的话，点击 xshell 终端菜单上的 View -> Tunnel Panel； 然后下面会出来一个 channel panel；点击 Fowarding Rules；在下面空白处右击，Add；然后把 jupyter 的 port （一般是 888 之类的数字） 填到上下的 port 栏中；最后复制终端显示的地址到自己的浏览器里

面就可以了。

3 其他的终端应该类似，或者网上搜索：某某终端软件+ port forwarding rules

4 另外，代码新手可以参考 CERN 为自己的 SWAN 平台准备的例子(https://swan-gallery.web.cern.ch/)，可以在网页直接看或者用 jupyter 运行.

（三）在 GPU 集群上运行 jupyter notebook

参考：https://www.bbsmax.com/A/Ae5R6Mv35Q/

机器学习程序测试好以后，可以用 GPU 资源加速运行。**重要：**在 GPU 集群上，只是用来程序的运行，**不要空闲挂着，**不然会占资源。

1 准备 jobs 的脚本

例如： /hpcfs/juno/junogpu/liuzhen/share/ slurm_sample_script_gpu.sh，修改其中的第 27 行以及其他 job 是设置；修改第 48-55 行之间的 4 个参数，包括：

jupyter notebook 运行的目录${ ifile }，和 configuration 文件目录${ jfile }, pytorch 所在的环境名字，以及想要设置的 port。

2 提交脚本

Sbatch 命令 “sbatch slurm_sample_script_gpu.sh”

3 设置 local forwarding:

Jobs 运行以后 打开 log 文件 复制其中的 “Command to create ssh tunnel:”下面的那句话 例如：

（1） 复制类似的这句话“ssh -N -f -L 8899:gpu034:8899 liuzhen@lxslc7.ihep.ac.cn”

（2） 在本地的 terminal 粘贴并运行 （windows 可以用 WSL 的 terminal）

（3） 然后复制 log 文件中“Use a Browser on your local machine to go to:”下面的那句话，在本地浏览器打开。例如： 复制“localhost:8899 “在浏览器打开

（4） 浏览器打开以后可能需要输入密码或者 token， 可以在 log 文件里面找到，例如 log 里面找到如下形式的链接

“ http://gpu034:8899/?token=6c0c0315511d462c998815cc5beb76bf2852aaa0c03c4093”,

需要输入的密码或者 token 就是其中的：

“6c0c0315511d462c998815cc5beb76bf2852aaa0c03c4093“。

4 不用的时候记得取消 job 避免占用资源： “scancel jobid”。