Interactive Apps中的jupyter notebook申请资源（申请时把jupyter作用的位置设为文件所在位置）

之后在命令行>\_scc什么的中进行命令行操作

module load miniconda

(setup\_scc\_condarc.sh 没显示需要就不用)

conda create —name myenv310 python=3.10 (创建名myenv310的python3.10环境) 3.10的环境的pytorch不能用tensorboard

conda activate myenv310 (conda deactivate)

# conda install jupyter notebook

conda install ipykernel

python -m ipykernel install --user --name myenv --display-name "Python (py37)"

然后在notebook里面切换kernel就行了

Installing Tensorflow via Pip:

(module load python3/3.8.10 应该不需要)

# install tensorflow - this includes both CPU and GPU versions

pip install --prefix=/projectnb/myproject/myfolder/mypip tensorflow==2.8.0

# cuda/11.2 comes with cuDNN 8.3

module load cuda/11.2

# load this next so that the correct cuDNN is found

# by Tensorflow

module load cudnn/8.1.1

python -c 'import tensorflow as tf; print(tf.config.list\_physical\_devices()’

Installing Tensorflow via Conda:

# RUN THIS ON A GPU NODE

(module load miniconda/4.9.2 应该不需要，前面已经load过了)

conda activate myenv

# this brings along its own copy of the req'd CUDA and cuDNN libraries

conda install -c conda-forge tensorflow-gpu==2.8.0

python -c 'import tensorflow as tf; print(tf.config.list\_physical\_devices()'

**使用cuda训练：**

查看哪个gpu能用：

nvidia-smi

选择gpu进行训练

cd 到example文件下

CUDA\_VISIBLE\_DEVICES=gpu\_id python3 train.py

之后可以按照example里面的agent\_policy.py的框架进行修改

models里面的json文件可以放到<https://2021vis.lux-ai.org/>查看动画

后台挂起运行

CUDA\_VISIBLE\_DEVICES=0 nohup python3 train.py &