

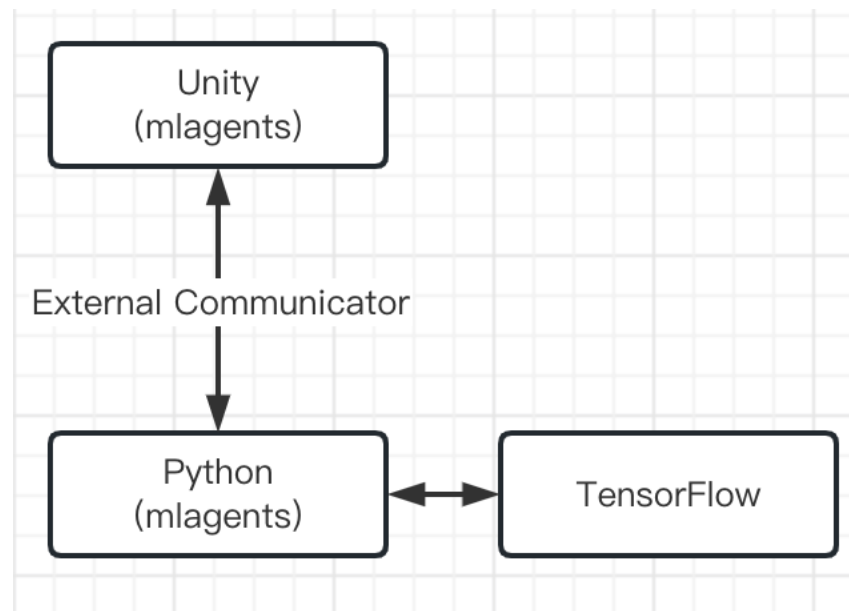
# 游戏项目实践 实践课16-2

- 游戏AI与强化学习

# MLAgent介绍

虽然名字为ML，实际主要部分是强化学习的模块

- Learning Environment – Unity中的场景及角色
- Python API – 包含强化学习算法。Python API 独立于Unity另外安装。
- External Communicator - Unity与Python 通信模块。



## 训练参数说明（设置文件见config文件夹）：

behaviors:

Crawler:

trainer\_type: ppo

hyperparameters:

batch\_size: 2048

buffer\_size: 20480

learning\_rate: 0.0003

network\_settings:

normalize: true

hidden\_units: 512

num\_layers: 3

vis\_encode\_type: simple

keep\_checkpoints: 5

max\_steps: 10000000

time\_horizon: 1000

# 模型名字

# 智能体训练算法

# 超参数

# 会有多少状态元组被用来学习

# 通常值越大越稳定

# 梯度下降的学习率

# 网络模型设置

# 隐藏层的神经元数量

# 训练网络的层数

# 编码器的结构, **simple**是一个两层的卷积网络

# 保留checkpoint的间隔

# 总训练步数

# 使用多少步后的采样对当前行为的预期奖励进行训练



1

# ML Agent实践



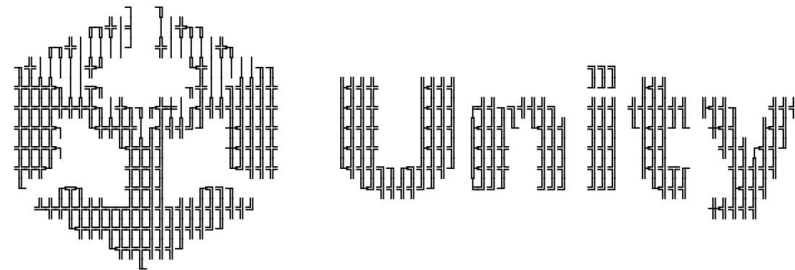
## 练习1-1 安装Python模块

- `conda create -n mlagents python=3.9` (如果不在意python环境, 可不安装)
- `conda activate mlagents` (如果不在意python环境, 可不安装)
- `pip install -i https://pypi.tuna.tsinghua.edu.cn/simple mlagents`
- `cd 16-LunarLanderProj` (进入到16-LunarLanderProj文件夹)
- `mlagents-learn config/lunarlander_config.yaml --run-id=lunarlander --train`

注: 如果运行不成功, 可将results文件删除, 再执行

如果运行过程中出现protobuf版本过高的问题, 可执行以下代码, 安装低版本

```
pip install protobuf==3.20.0
```



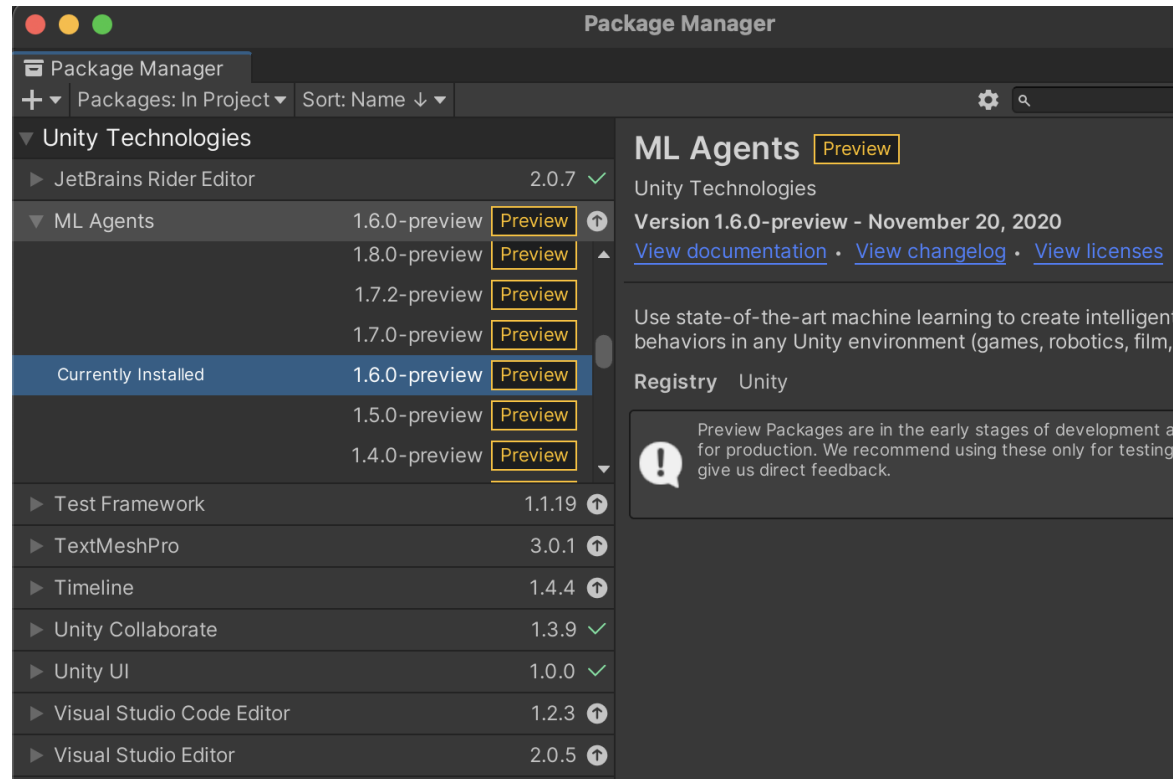
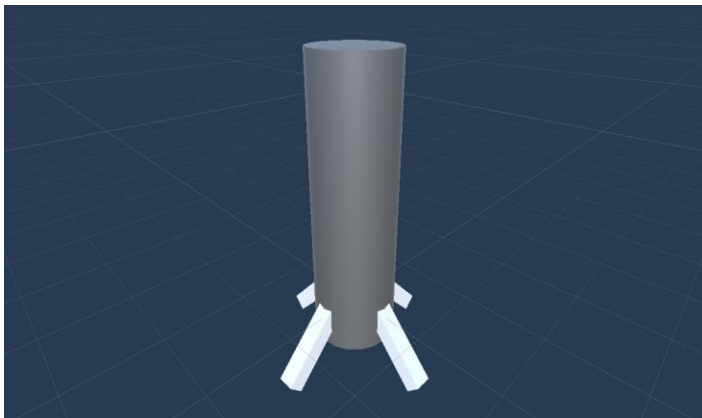
```
Version information:  
ml-agents: 0.30.0,  
ml-agents-envs: 0.30.0,  
Communicator API: 1.5.0,  
PyTorch: 1.11.0  
[INFO] Listening on port 5004. Start training by pressing the Play button in the  
Unity Editor.
```

# MLAgent实践

练习1-2 Unity mlagents插件安装 (已经在LunarLanderProj项目文件Packages/manifest.json中设置, 可不用额外安装)

- 安装的版本为, 如果无法选择指定版本, 则直接修改项目的Packages/manifest.json里的MLAgents版本: 添加或修改为: "com.unity.ml-agents": "1.6.0-preview"

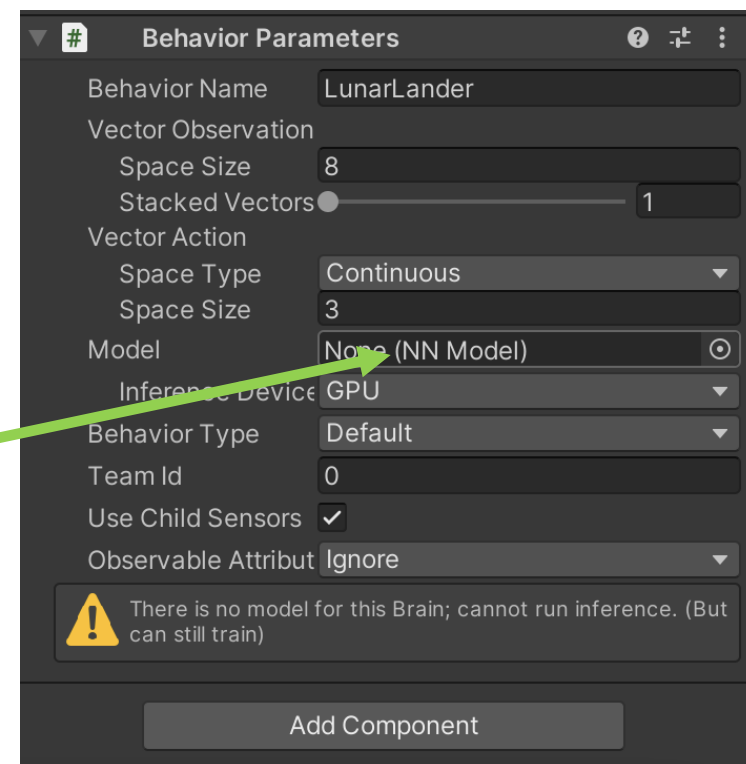
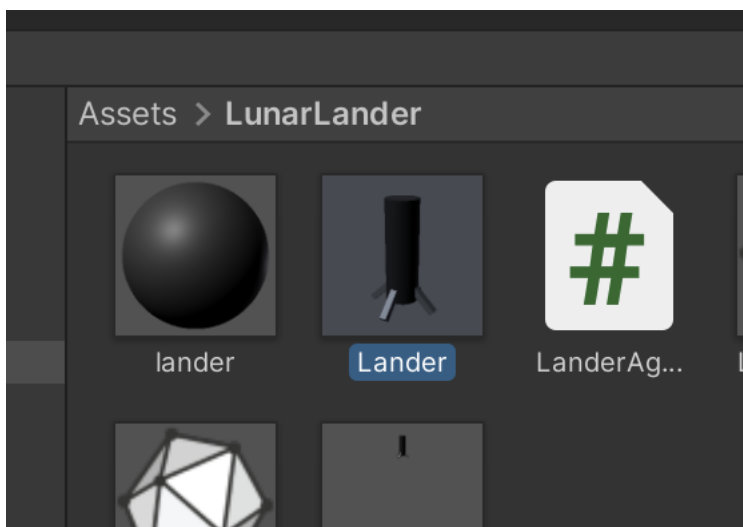
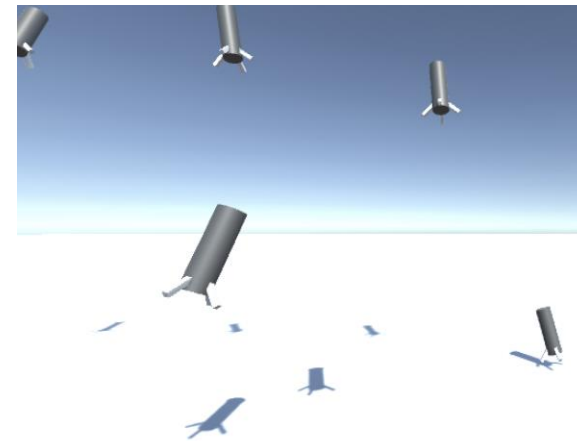
```
manifest.json
{
  "dependencies": {
    "com.unity.collab-proxy": "1.3.9",
    "com.unity.ide.rider": "2.0.7",
    "com.unity.ide.visualstudio": "2.0.5",
    "com.unity.ide.vscode": "1.2.3",
    "com.unity.ml-agents": "1.6.0-preview",
  }
}
```



# MLAgent实践

## 练习1-3 训练

- 确保mlagents-learn已在运行
- 打开LunarLanderProj项目，运行Scenes/LunarLander.unity，开始训练
- 达到一定阶段后，python端可以暂停
- 在results文件里找到训好的模型：LunarLander.onnx，复制到Unity Assets文件里，并设置到Lander对象的Model处



| Q&A