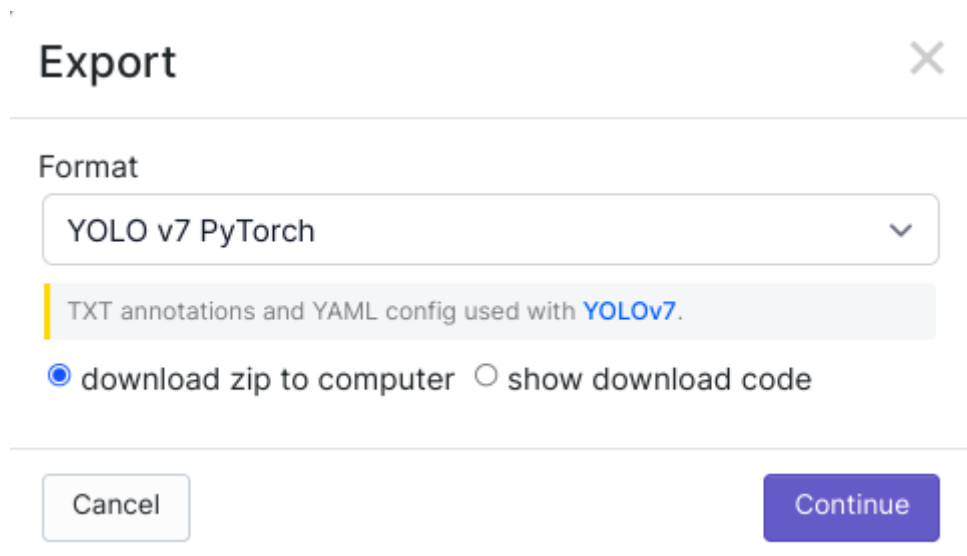


# AutoDL 训练 YoloV7

## 准备数据

- 推荐使用Roboflow标注数据
  - 在上传图片后可分配多人共同标注
  - 可快速划分训练集验证集和测试集比例
  - 可对图像进行预处理和数据增强
  - 可直接导出YoloV7对应格式的数据集
  - 在Export中选择YOLOv7格式下载



- 下载后文件结构



train



valid



data.  
yaml



README.  
dataset  
.txt



README.  
roboflo  
w.txt

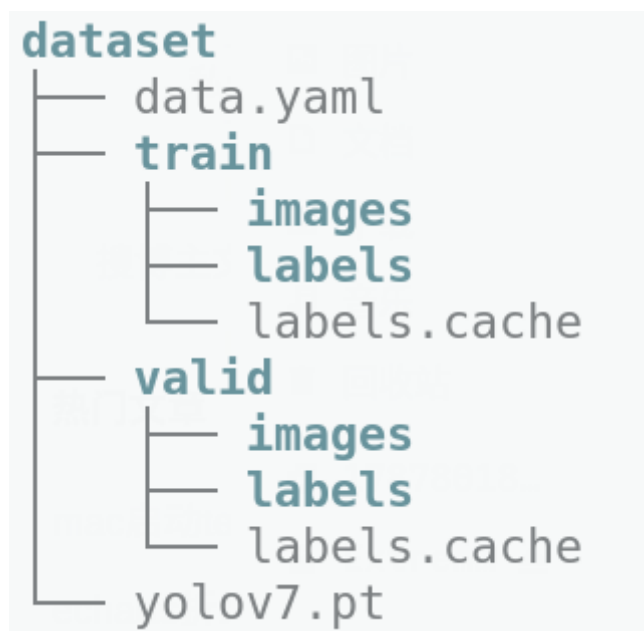
## 在AutoDL中上传数据集

- 修改data.yaml (“dataset”为这个文件夹的名字)

```
train: ./dataset/train/images
val:   ./dataset/valid/images

nc: 3
names: ['bottle', 'person', 'apple']
```

- 可以把权重文件也一同放入，压缩dataset文件夹后在“我的网盘”中上传



## 租用实例

- 在“我的实例”中选择需要的GPU型号
- 需要使用网盘中文件的话要选择和网盘相同区，并勾选“支持网盘”

选租主机: 高可用 芜湖区 高可用 内蒙A区 高可用 北京A区 高可用 南京新手区 高可用 特惠/泉州A区 [如何选择GPU?](#)

将在您选择主机上创建实例，主机之间有差异，请根据需要进行选择

算力型号: ☐ 全部 ☒ RTX A5000 (104/360) ☐ RTX 3090 (101/232) ☐ A100-SXM4-80GB (0/48) ☐ A40 (13/184) ☐ V100-32GB (0/2)

☐ RTX 3080 (0/80) ☐ TITAN Xp (0/48)

空闲GPU:  ☐ 支持网盘

主机ID	算力型号/显存	空闲GPU	每GPU分配	CPU型号	硬盘	支持网盘	价格(单卡)
<input checked="" type="radio"/> 092机	RTX A5000 24GB	4/8	CPU: 14核 内存: 30GB	Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz	数据盘: 50GB SSD 最大扩容: 4000GB	是	¥1.33/时 ¥1.40/时
<input type="radio"/> 093机	RTX A5000 24GB	4/8	CPU: 14核 内存: 30GB	Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz	数据盘: 50GB SSD 最大扩容: 2005GB	是	¥1.33/时 ¥1.40/时

- 下方选择“算法镜像”，选择yolov7/master



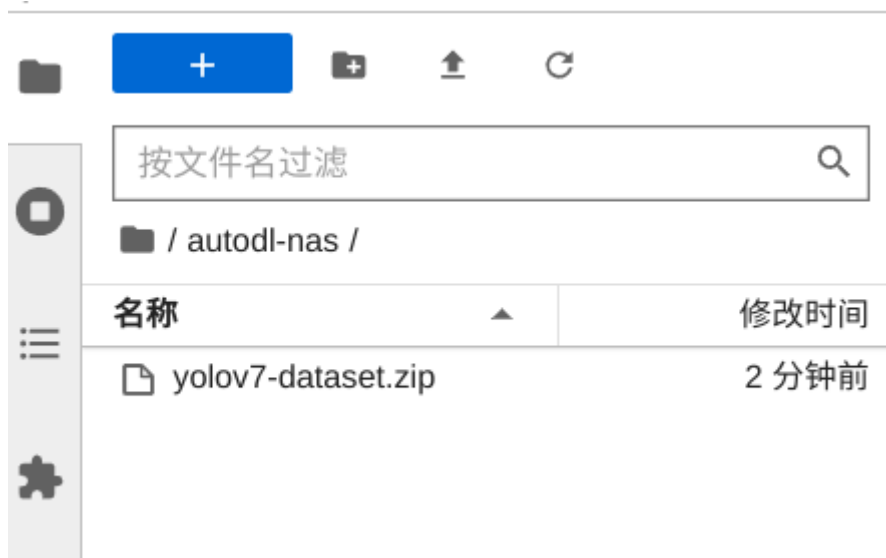
- 点击“立即创建”

## 训练

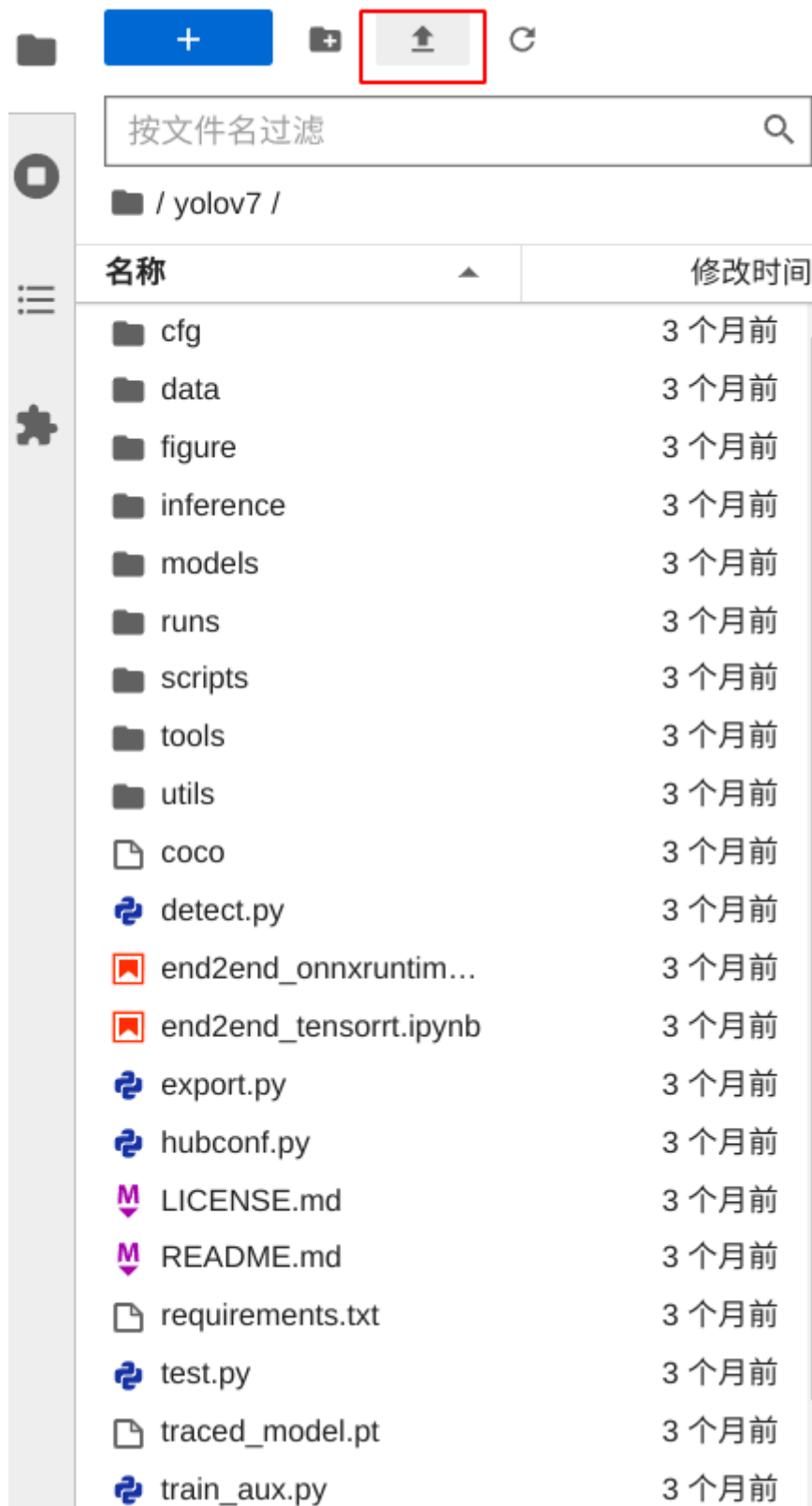
- 在“我的实例”中选择租用的主机开机，点击右侧JupyterLab进入



- 解压上传的文件
- 切换到autodl-nas目录可以看见网盘中之前上传的文件，将其复制到yolov7目录



- 也可以直接点击上传选择文件



- 在终端运行解压命令

```
1 sudo unzip yolov7-dataset.zip
```

- 刷新后可看到解压出的dataset文件夹

- 开始训练

- 终端输入命令

若报错CUDA out of memory则减小batch size为32

```
1 python train.py --weights dataset/yolov7.pt --cfg cfg/training/yolov7.yaml --  
data dataset/data.yaml --device 0 --batch-size 64 --epoch 200
```

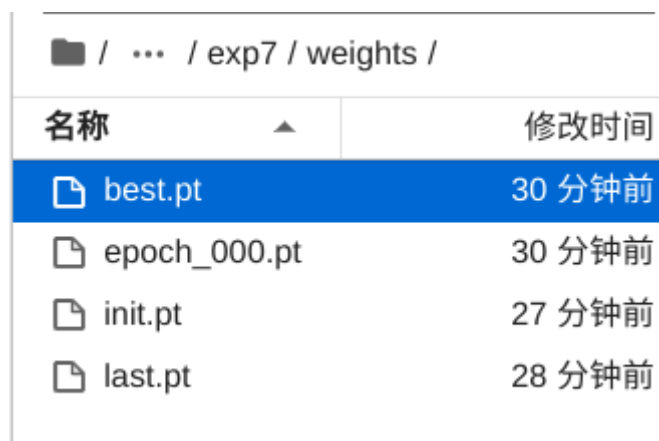
- 若训练中断可传参数 -resume断点续训

```
1 python train.py --weights dataset/yolov7.pt --cfg cfg/training/yolov7.yaml --  
data dataset/data.yaml --device 0 --batch-size 32 --epoch 200 --resume
```

## 测试与部署

- 下载模型

- 训练好后进入/yolov7/runs/train/最新的exp/weights目录，下载best.pt



/ ... / exp7 / weights /	
名称	修改时间
best.pt	30 分钟前
epoch_000.pt	30 分钟前
init.pt	27 分钟前
last.pt	28 分钟前

- 测试

- github下载Object-Detector-YoloV7，并根据README安装依赖

```
1 wget https://ghproxy.com/https://github.com/Cmoon-cyl/Object-Detector-  
YoloV7/releases/download/v1.0.0/Detector_v7.zip
```

- 解压Detector\_v7，并根据README配置运行环境
- 将best.pt放入weights文件夹，终端运行如下指令调用azure kinect识别

```
1 python Detector.py --weights weights/best.pt --device k4a
```

- 部署

- 参考example.py，在实例化时传入best.pt模型路径

```
1 from Detector import Detector
2 yolo=Detector("weights/best.pt")
3 results=yolo.detect()
```