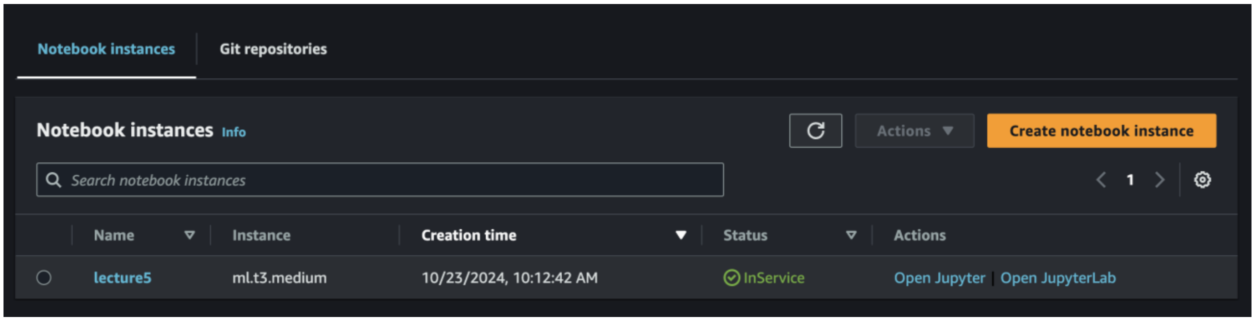
Operationalizing ML on Sagemaker

Training and Deployment on Sagemaker

Initial Setup

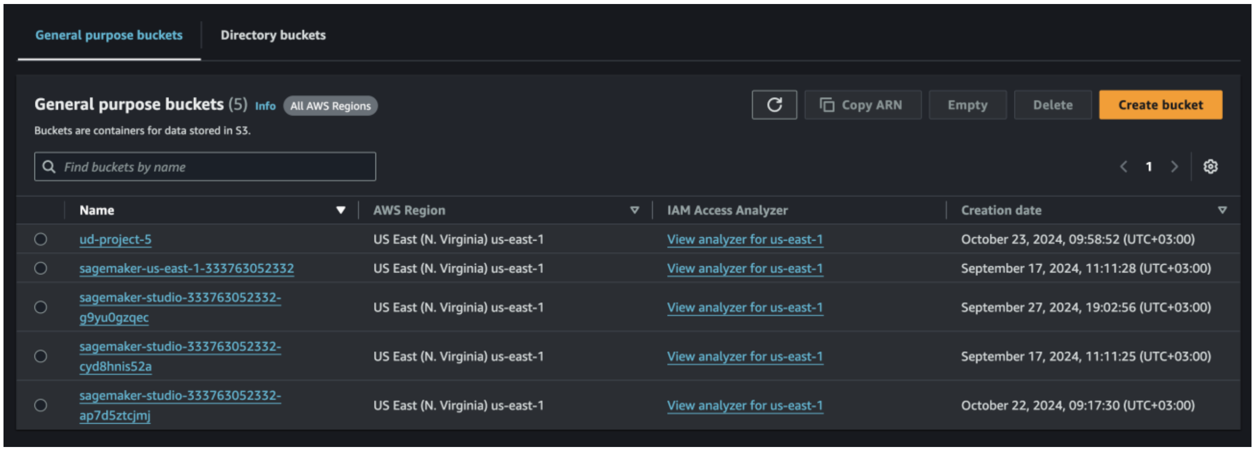
I selected the ml.t3.medium instance as it balances cost and performance for small to medium training tasks. It offers adequate CPU, memory, and network performance to efficiently train and deploy a computer vision model without high expenses. The instance also launches quickly, making it suitable for tasks requiring decent computational speed without heavy GPU use.



	Name	Instance	Creation time	Status	Actions
○	lecture5	ml.t3.medium	10/23/2024, 10:12:42 AM	✓ InService	Open Jupyter | Open JupyterLab

S3 Setup

For storing and managing data, I set up an Amazon S3 bucket. This provides a scalable and secure location for the large datasets required for computer vision tasks. Using S3, I can easily access and manage the data from Sagemaker, facilitating seamless integration between storage and computation.



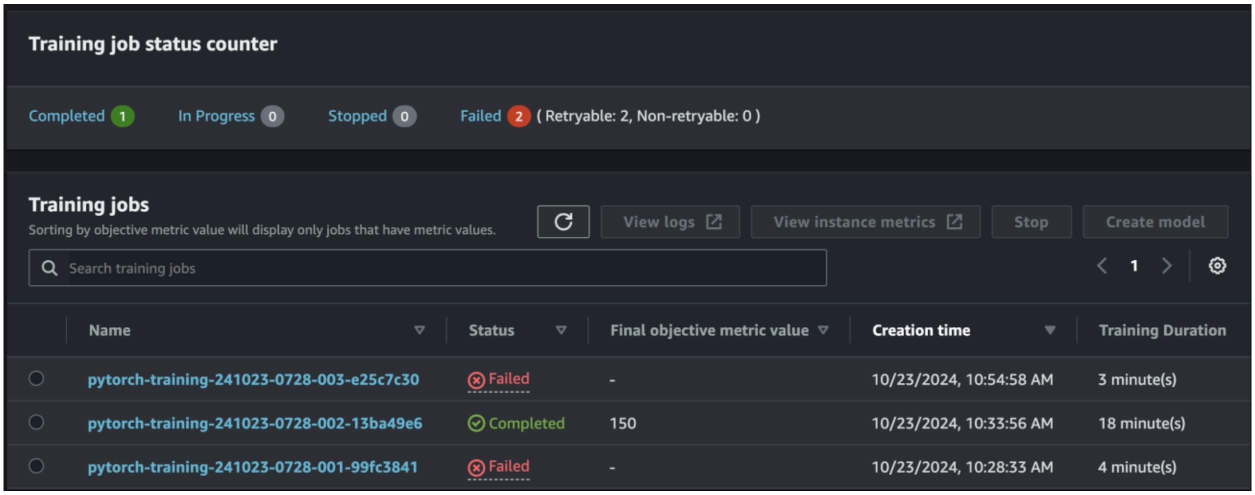
Buckets are containers for data stored in S3.

	Name	AWS Region	IAM Access Analyzer	Creation date
○	ud-project-5	US East (N. Virginia) us-east-1	View analyzer for us-east-1	October 23, 2024, 09:58:52 (UTC+03:00)
○	sagemaker-us-east-1-333763052332	US East (N. Virginia) us-east-1	View analyzer for us-east-1	September 17, 2024, 11:11:28 (UTC+03:00)
○	sagemaker-studio-333763052332-g9yu0gzqec	US East (N. Virginia) us-east-1	View analyzer for us-east-1	September 27, 2024, 19:02:56 (UTC+03:00)
○	sagemaker-studio-333763052332-cyd8hnis52a	US East (N. Virginia) us-east-1	View analyzer for us-east-1	September 17, 2024, 11:11:25 (UTC+03:00)
○	sagemaker-studio-333763052332-ap7d5ztcjmj	US East (N. Virginia) us-east-1	View analyzer for us-east-1	October 22, 2024, 09:17:30 (UTC+03:00)

Training and Deployment

Hyperparameter Tuning

Even though 2 of my trainings failed at first, one of them was a success and I used it.



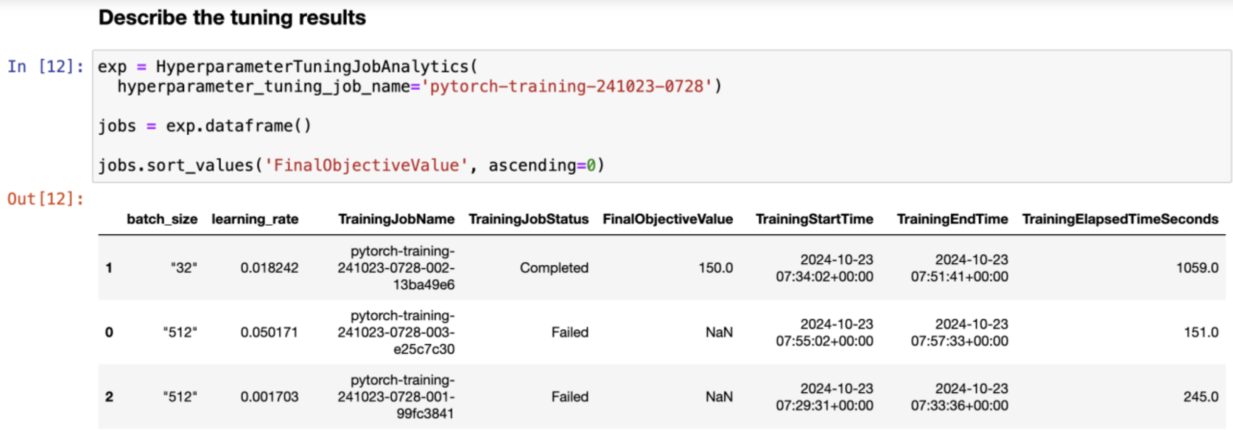
Training job status counter

Completed 1 In Progress 0 Stopped 0 Failed 2 (Retryable: 2, Non-retryable: 0)

Training jobs

Sorting by objective metric value will display only jobs that have metric values.

	Name	Status	Final objective metric value	Creation time	Training Duration
○	pytorch-training-241023-0728-003-e25c7c30	⊗ Failed	-	10/23/2024, 10:54:58 AM	3 minute(s)
○	pytorch-training-241023-0728-002-13ba49e6	✓ Completed	150	10/23/2024, 10:33:56 AM	18 minute(s)
○	pytorch-training-241023-0728-001-99fc3841	⊗ Failed	-	10/23/2024, 10:28:33 AM	4 minute(s)



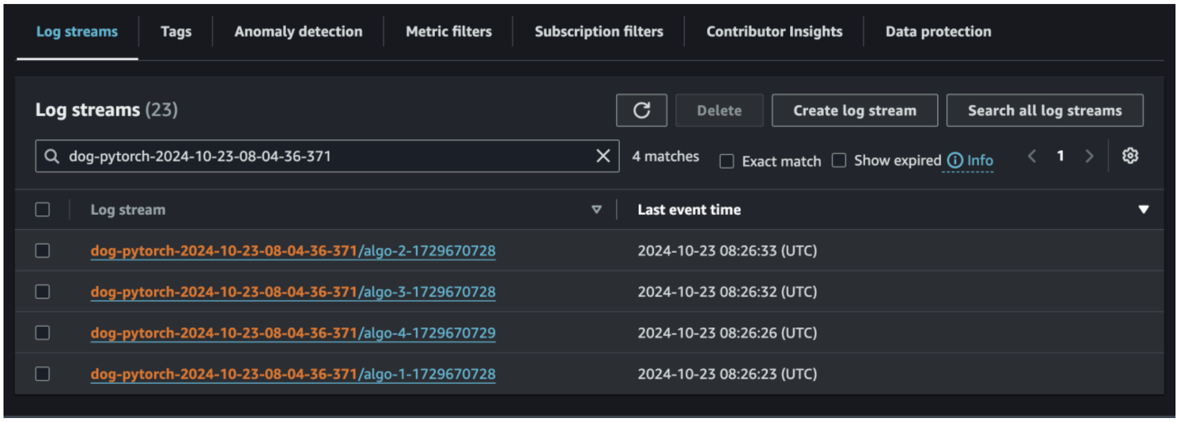
Describe the tuning results

```
In [12]: exp = HyperparameterTuningJobAnalytics(
           hyperparameter_tuning_job_name='pytorch-training-241023-0728')

         jobs = exp.dataframe()

         jobs.sort_values('FinalObjectiveValue', ascending=0)
```

Out[12]:

	batch_size	learning_rate	TrainingJobName	TrainingJobStatus	FinalObjectiveValue	TrainingStartTime	TrainingEndTime	TrainingElapsedTimeSeconds
1	"32"	0.018242	pytorch-training-241023-0728-002-13ba49e6	Completed	150.0	2024-10-23 07:34:02+00:00	2024-10-23 07:51:41+00:00	1059.0
0	"512"	0.050171	pytorch-training-241023-0728-003-e25c7c30	Failed	NaN	2024-10-23 07:55:02+00:00	2024-10-23 07:57:33+00:00	151.0
2	"512"	0.001703	pytorch-training-241023-0728-001-99fc3841	Failed	NaN	2024-10-23 07:29:31+00:00	2024-10-23 07:33:36+00:00	245.0

Training

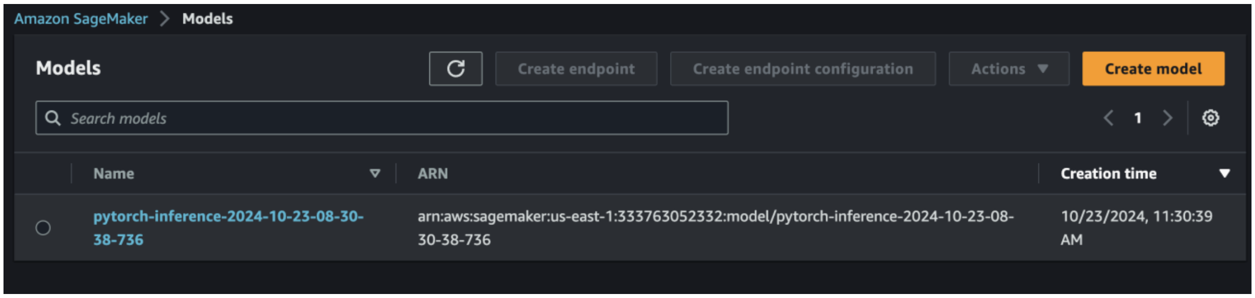


Log streams (23)

dog-pytorch-2024-10-23-08-04-36-371 ✕ 4 matches □ Exact match □ Show expired ⓘ Info 1

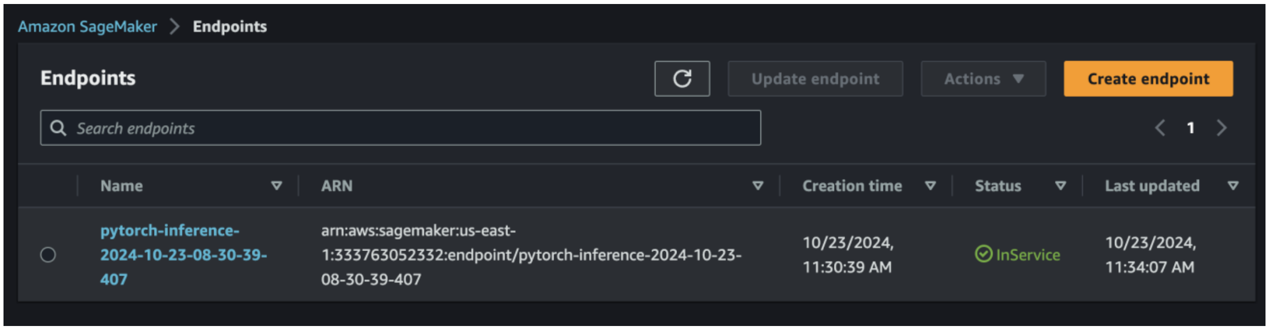
□	Log stream	Last event time
□	dog-pytorch-2024-10-23-08-04-36-371/algo-2-1729670728	2024-10-23 08:26:33 (UTC)
□	dog-pytorch-2024-10-23-08-04-36-371/algo-3-1729670728	2024-10-23 08:26:32 (UTC)
□	dog-pytorch-2024-10-23-08-04-36-371/algo-4-1729670729	2024-10-23 08:26:26 (UTC)
□	dog-pytorch-2024-10-23-08-04-36-371/algo-1-1729670728	2024-10-23 08:26:23 (UTC)

Endpoint



Models

	Name	ARN	Creation time
○	pytorch-inference-2024-10-23-08-30-38-736	arn:aws:sagemaker:us-east-1:333763052332:model/pytorch-inference-2024-10-23-08-30-38-736	10/23/2024, 11:30:39 AM

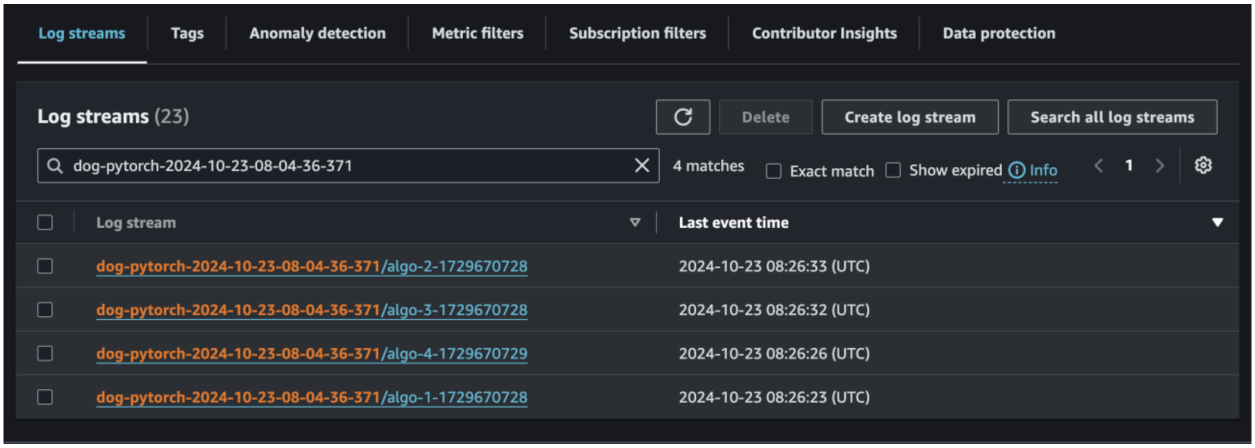


Endpoints

	Name	ARN	Creation time	Status	Last updated
○	pytorch-inference-2024-10-23-08-30-39-407	arn:aws:sagemaker:us-east-1:333763052332:endpoint/pytorch-inference-2024-10-23-08-30-39-407	10/23/2024, 11:30:39 AM	✓ InService	10/23/2024, 11:34:07 AM

Multi Instance Training

I increased the instance count to 4 for my training. I added the code and logs.

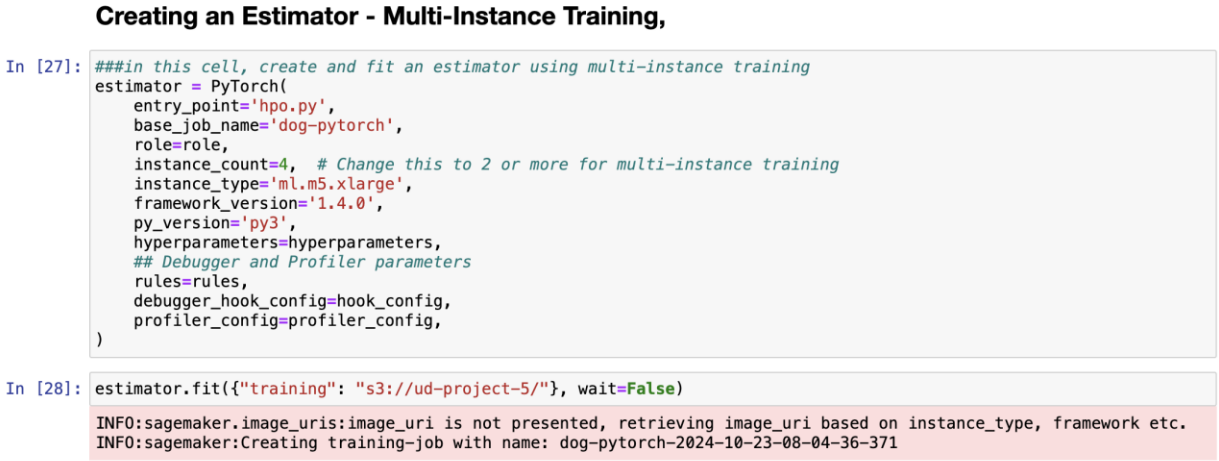


Log streams (23)

dog-pytorch-2024-10-23-08-04-36-371

4 matches

Log stream	Last event time
dog-pytorch-2024-10-23-08-04-36-371/algo-2-1729670728	2024-10-23 08:26:33 (UTC)
dog-pytorch-2024-10-23-08-04-36-371/algo-3-1729670728	2024-10-23 08:26:32 (UTC)
dog-pytorch-2024-10-23-08-04-36-371/algo-4-1729670729	2024-10-23 08:26:26 (UTC)
dog-pytorch-2024-10-23-08-04-36-371/algo-1-1729670728	2024-10-23 08:26:23 (UTC)



Creating an Estimator - Multi-Instance Training,

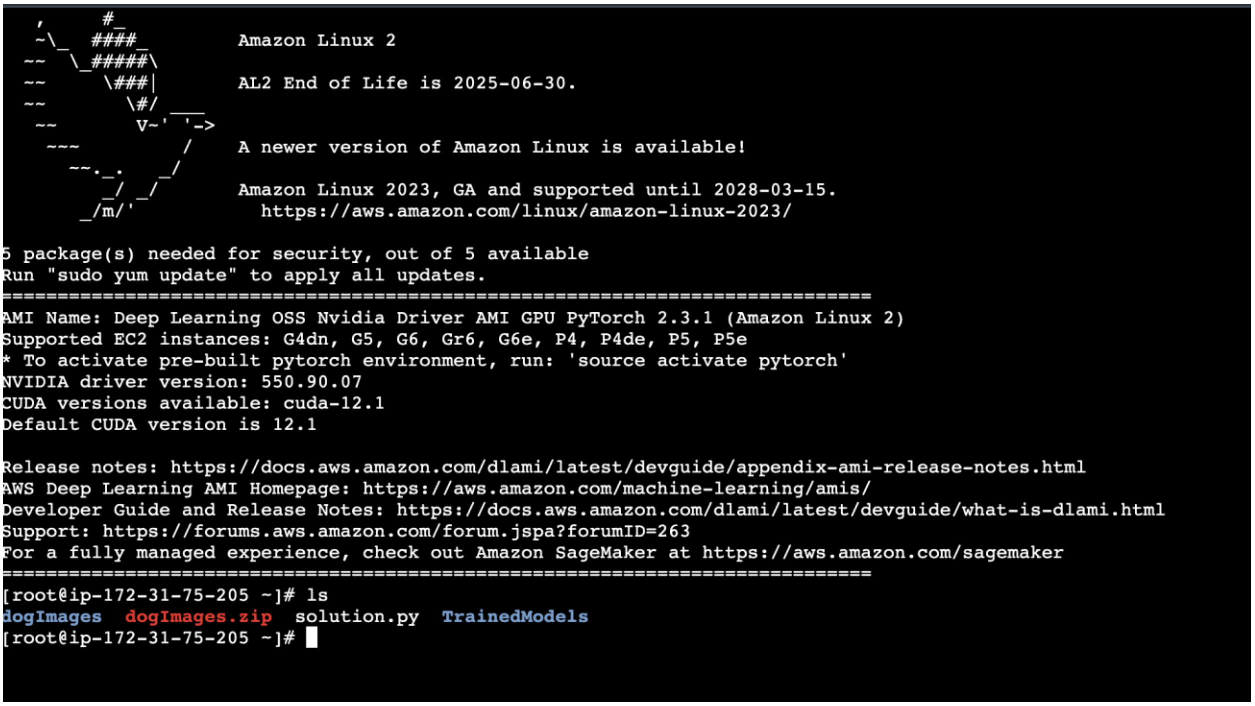
```
In [27]: ###in this cell, create and fit an estimator using multi-instance training
         estimator = PyTorch(
             entry_point='hpo.py',
             base_job_name='dog-pytorch',
             role=role,
             instance_count=4,  # Change this to 2 or more for multi-instance training
             instance_type='ml.m5.xlarge',
             framework_version='1.4.0',
             py_version='py3',
             hyperparameters=hyperparameters,
             ## Debugger and Profiler parameters
             rules=rules,
             debugger_hook_config=hook_config,
             profiler_config=profiler_config,
         )
```

```
In [28]: estimator.fit({"training": "s3://ud-project-5/"}, wait=False)
```

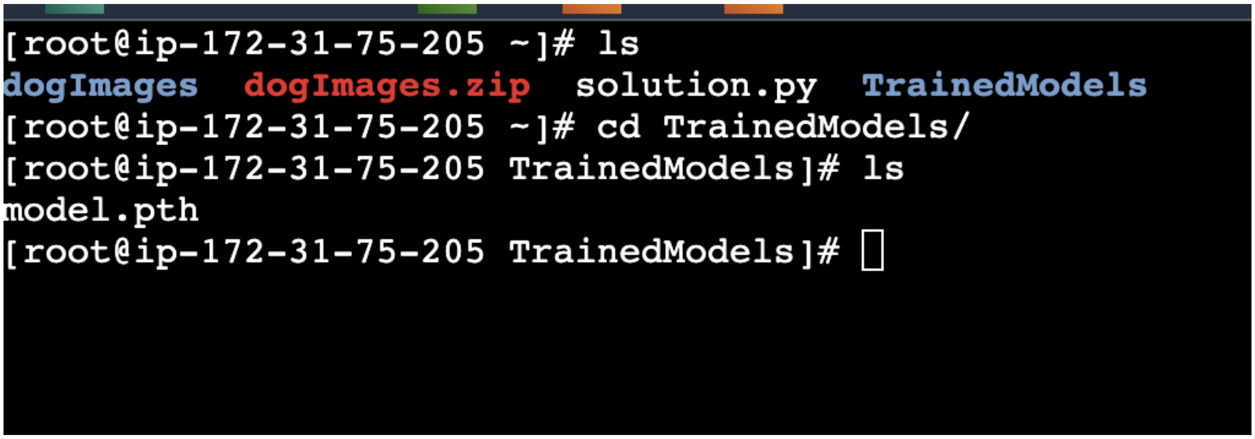
```
INFO:sagemaker.image_uris:image_uri is not presented, retrieving image_uri based on instance_type, framework etc.
INFO:sagemaker:Creating training-job with name: dog-pytorch-2024-10-23-08-04-36-371
```

EC2 Training

I selected “Deep Learning OSS Nvidia Driver AMI GPU PyTorch 2.3 (Amazon Linux 2)” on the AMI
list, so I can proceed with the AMI compatible with the Pytorch environment, which we are using. We chose the G5.xlarge instance type for training our small model, as it performs excellently due to being a new instance.



```
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2025-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

5 package(s) needed for security, out of 5 available
Run "sudo yum update" to apply all updates.
=============================================================================
AMI Name: Deep Learning OSS Nvidia Driver AMI GPU PyTorch 2.3.1 (Amazon Linux 2)
Supported EC2 instances: G4dn, G5, G6, Gr6, G6e, P4, P4de, P5, P5e
* To activate pre-built pytorch environment, run: 'source activate pytorch'
NVIDIA driver version: 550.90.07
CUDA versions available: cuda-12.1
Default CUDA version is 12.1

Release notes: https://docs.aws.amazon.com/dlami/latest/devguide/appendix-ami-release-notes.html
AWS Deep Learning AMI Homepage: https://aws.amazon.com/machine-learning/amis/
Developer Guide and Release Notes: https://docs.aws.amazon.com/dlami/latest/devguide/what-is-dlami.html
Support: https://forums.aws.amazon.com/forum.jspa?forumID=263
For a fully managed experience, check out Amazon SageMaker at https://aws.amazon.com/sagemaker
=============================================================================
[root@ip-172-31-75-205 ~]# ls
dogImages  dogImages.zip  solution.py  TrainedModels
[root@ip-172-31-75-205 ~]#
```

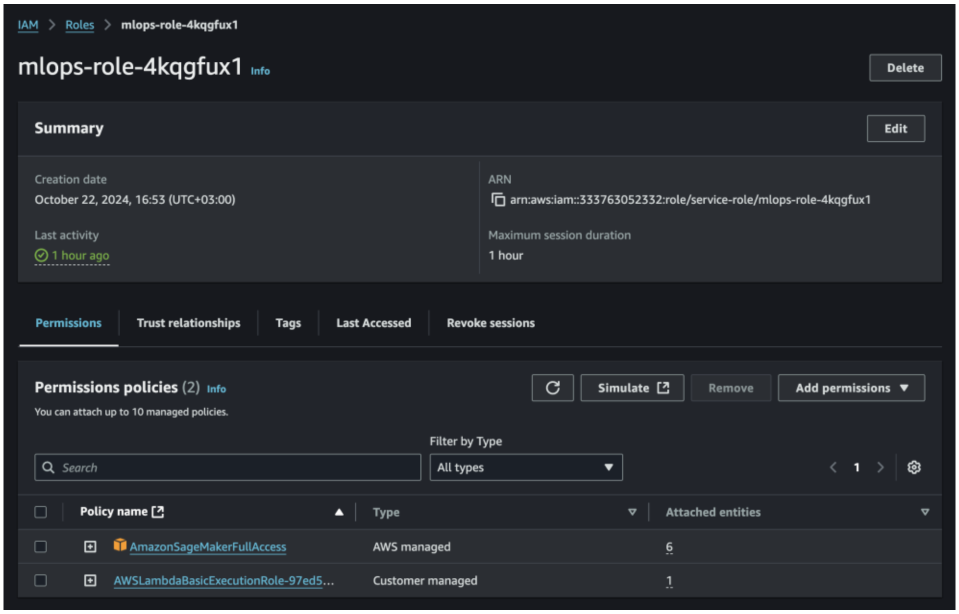


```
[root@ip-172-31-75-205 ~]# ls
dogImages  dogImages.zip  solution.py  TrainedModels
[root@ip-172-31-75-205 ~]# cd TrainedModels/
[root@ip-172-31-75-205 TrainedModels]# ls
model.pth
[root@ip-172-31-75-205 TrainedModels]#
```

We ran a Python script on EC2 rather than a Jupyter Notebook, which requires a distinct environment. Additionally, we didn't leverage Sagemaker's hyperparameter tuning, opting for hard-coded values instead. Apart from these changes, the code closely resembles hpo.py, which was used as our estimator entry point in the notebook.

Lambda Function Setup

The lambda_function.py file receives an event and context, decodes the event with base64, and forwards the decoded data to a SageMaker endpoint. The function then processes the endpoint's response, converting it into a JSON object that consists of HTTP status codes and headers. Furthermore, it employs the Python logging module to log debug messages during the entire procedure.



mlops-role-4kqgfux1

Summary

Creation date
October 22, 2024, 16:53 (UTC+03:00)

Last activity
1 hour ago

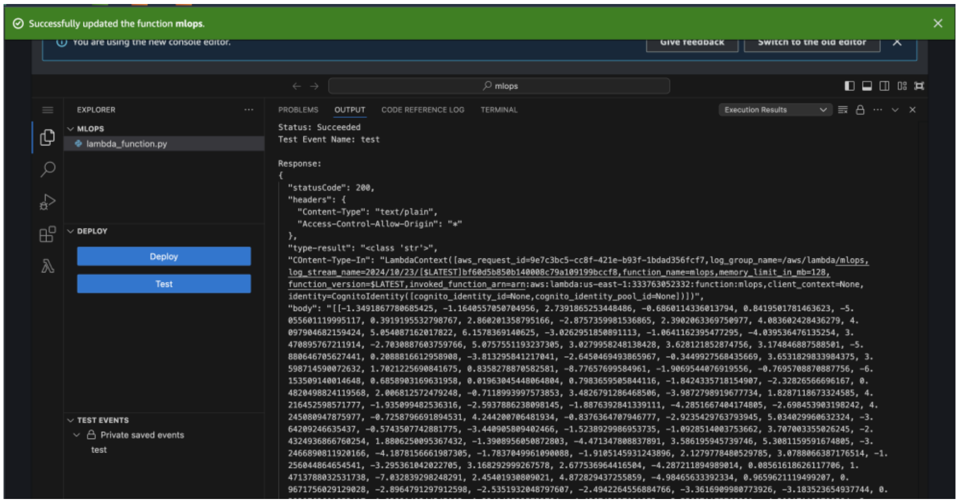
ARN
arn:aws:iam::333763052332:role/service-role/mlops-role-4kqgfux1

Maximum session duration
1 hour

Permissions policies (2)
You can attach up to 10 managed policies.

Filter by Type
All types

	Policy name	Type	Attached entities
□	AmazonSageMakerFullAccess	AWS managed	6
□	AWSLambdaBasicExecutionRole-97ed5...	Customer managed	1



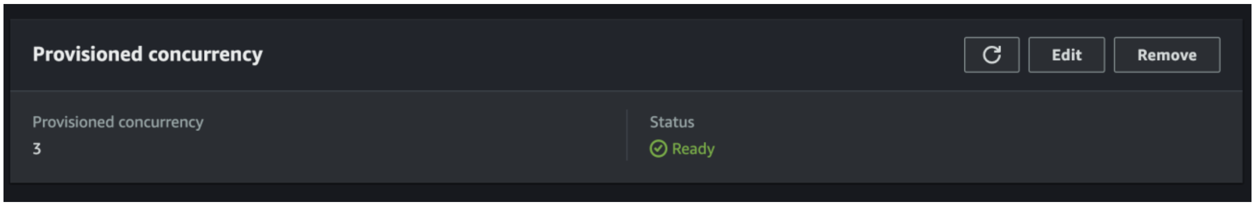
Successfully updated the function **mlops**.

You are using the new console editor.

```
Status: Succeeded
Test Event Name: test

Response:
{
  "statusCode": 200,
  "headers": {
    "Content-Type": "text/plain",
    "Access-Control-Allow-Origin": "*"
  },
  "type-result": "<class 'str'>",
  "COntent-Type-In": "LambdaContext([aws_request_id=9e7c3bc5-cc8f-421e-b93f-1bdad356fcf7,log_group_name=/aws/lambda/mlops,
  log_stream_name=2024/10/23/[$LATEST]bf60d5b850b140008c79a109199bccf8,function_name=mlops,memory_limit_in_mb=128,
  function_version=$LATEST,invoked_function_arn=arn:aws:lambda:us-east-1:333763052332:function:mlops,client_context=None,
  identity=CognitoIdentity([cognito_identity_id=None,cognito_identity_pool_id=None])])",
  "body": "[[-1.3491867780685425, -1.1640557050704956, 2.7391865253448486, -0.6860114336013794, 0.8419501781463623, -5.
  055601119995117, 0.3919195532798767, 2.860201358795166, -2.8757359981536865, 2.3902063369750977, 4.083602428436279, 4.
  097904682159424, 5.054087162017822, 6.1578369140625, -3.0262951850891113, -1.0641162395477295, -4.039536476135254, 3.
  470895767211914, -2.7030887603759766, 5.0757551193237305, 3.0279958248138428, 3.628121852874756, 3.174846887588501, -5.
  880646705627441, 0.2088816612958908, -3.813295841217041, -2.6450469493865967, -0.3449927568435669, 3.6531829833984375, 3.
  598714590072632, 1.7021225690841675, 0.8358278870582581, -8.77657699584961, -1.9069544076919556, -0.7695708870887756, -6.
  153509140014648, 0.6858903169631958, 0.01963045448064804, 0.7983659505844116, -1.8424335718154907, -2.32826566696167, 0.
  4820498824119568, 2.006812572479248, -0.7118993997573853, 3.4826791286468506, -3.9872798919677734, 1.8287118673324585, 4.
  216452598571777, -1.935099482536316, -2.5937886238098145, -1.8876392841339111, -4.2851667404174805, -2.698453903198242, 4.
  245080947875977, -0.7258796691894531, 4.244200706481934, -0.8376364707946777, -2.9235429763793945, 5.034029960632324, -3.
  64209246635437, -0.5743507742881775, -3.440905809402466, -1.5238929986953735, -1.0928514003753662, 3.707003355026245, -2.
  4324936866760254, 1.8806250095367432, -1.3908956050872803, -4.471347808837891, 3.586195945739746, 5.3081159591674805, -3.
  2466890811920166, -4.1878156661987305, -1.7837049961090088, -1.9105145931243896, 2.1279778480529785, 3.0788066387176514, -1.
  256044864654541, -3.295361042022705, 3.168292999267578, 2.677536964416504, -4.287211894989014, 0.08561618626117706, 1.
  4713788032531738, -7.032839298248291, 2.45401930809021, 4.872829437255859, -4.98465633392334, 0.965962111949207, 0.
  9671756029129028, -2.8964791297912598, -2.5351932048797607, -2.4942264556884766, -3.3616909980773926, -3.183523654937744, 0.
```

Concurrency and Auto Scaling

I assigned three instances to run my Lambda function concurrently in order to cut costs.

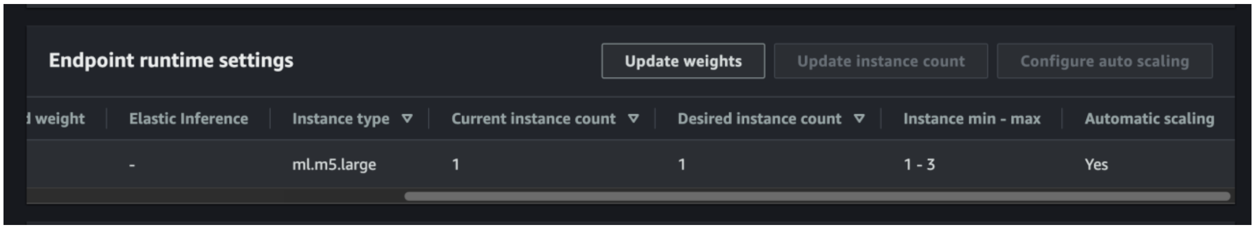


Provisioned concurrency

Provisioned concurrency: 3

Status: ✓ Ready

I set up auto scaling for my endpoint to activate when the number of invocations hits 20 per second, with a scale-in period of 30 seconds and a scale-out period of 30 seconds. This configuration was aimed at optimizing costs while managing high traffic.



Endpoint runtime settings

weight	Elastic Inference	Instance type	Current instance count	Desired instance count	Instance min - max	Automatic scaling
	-	ml.m5.large	1	1	1 - 3	Yes