



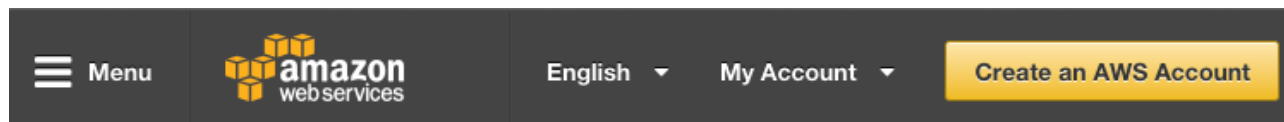
How to setup AWS cloud for Machine Learning

Table of Contents

How to setup AWS cloud for Machine Learning	1
Table of Contents	2
Create an AWS Account	3
View Your Current Limit	4
Submit a Limit Increase Request	4
Wait for Approval	6
How to get AWS credits?	7
Launch an Instance	9
Select the Instance Type	9
Configure the Security Group	11
Launch the Instance	11
Create an Authentication Key Pair	12
Be Careful!	13
Login to the Instance	14
Configure Jupyter notebook settings	14
Test the Instance	15
From your local machine	15
Important: Cost	16
More Resources	17

Create an AWS Account

Visit aws.amazon.com and click on the "Create an AWS Account" button.



If you have an AWS account already, sign in.

If you do not have an AWS account, sign up.

When you sign up, you will need to provide a credit card. But don't worry, you won't be charged for anything yet.

Furthermore, when you sign up, you will also need to choose a support plan. You can choose the free Basic Support Plan.

Once you finish signing up, wait a few minutes to receive your AWS account confirmation email. Then return to aws.amazon.com and sign in.

View Your Current Limit

Amazon Web Services has a service called [Elastic Compute Cloud \(EC2\)](#), which allows you to launch virtual servers (or “instances”), including instances with attached GPUs. The specific type of GPU instance you should launch for this tutorial is called “p2.xlarge”.

We will use [this AMI \(Amazon Machine Image\)](#) to define the operating system for your instance, and to make use of its pre-installed software. In order to use this AMI, you **must** change your AWS region to one of the following (and you are encouraged to select the region in the list that is closest to you):

- EU (Ireland)
- Asia Pacific (Seoul)
- Asia Pacific (Tokyo)
- Asia Pacific (Sydney)
- US East (N. Virginia)
- US East (Ohio)
- US West (Oregon)

After changing your AWS region, view your EC2 Service Limit report at: <https://console.aws.amazon.com/ec2/v2/home?#Limits>, and find your "Current Limit" for the p2.xlarge instance type.

Running On-Demand p2.16xlarge instances	0	Request limit increase
Running On-Demand p2.8xlarge instances	0	Request limit increase
Running On-Demand p2.xlarge instances	0	Request limit increase

By default, AWS sets a limit of 0 on the number of p2.xlarge instances a user can run, which effectively prevents you from launching this instance.

Submit a Limit Increase Request

If your limit of p2.xlarge instances is 0, you'll need to increase the limit before you can launch an instance. From the EC2 Service Limits page, click on “Request limit increase” next to “p2.xlarge”.

You will not be charged for requesting a limit increase. You will only be charged once you actually launch an instance.

Running On-Demand p2.16xlarge instances	0	Request limit increase
Running On-Demand p2.8xlarge instances	0	Request limit increase
Running On-Demand p2.xlarge instances	0	Request limit increase

On the service request form, you will need to complete several fields.

For the “Region” field, select the AWS region you chose in **Step 2**.

For the “New limit value”, enter a value of 1 (or more, if you wish).

For the “Use Case Description”, you can simply state: “I would like to use GPU instances for deep learning.”

Regarding* ☐ Account and Billing Support ☒ **Service Limit Increase** ☐ Technical Support
Unavailable under the Basic Support Plan

Limit Type* EC2 Instances

Request 1

Region* US West (Oregon)

Primary Instance Type* p2.xlarge

Limit* Instance Limit

New limit value* 1

Add another request

Use Case Description* I would like to use GPU instances for deep learning.

Note: If you have never launched an instance of any type on AWS, you might receive an email from AWS Support asking you to initialize your account by creating an instance before they approve the limit increase.

Wait for Approval

You must wait until AWS approves your Limit Increase Request. AWS typically approves these requests within 48 hours.

How to get AWS credits?

This should bring up a page like the one below. Fill in the data for this page. In your AWS account, **your AWS Account ID can be found under 'My Account'**, which is itself found in the dropdown under the name of your account, between the bell and the 'Global' dropdown.



Udacity Deep Learning Foundations -2018

AWS Educate is working with Udacity to provide you resources to support your coursework. To receive AWS promotional credits, enter your name, the email address you used to register for the course, and your AWS ID. After your email address is verified, you will receive a message with an AWS promotional credit code.

Country |

City |

Field of Study | Please select the most appropriate

First Name |

Last Name |

Email |

AWS Account ID |

[Don't have one? Sign up now](#)

Completion year for current course |

The graduation year of your current degree program.

Completion month for current course |

The graduation month of your current degree program.

Please click the box below to help assure that a person and not an automated program is submitting this application. If a set of letters is displayed enter them on the line. If you have any difficulty with the letters, you can click the reload icon to get a new set of letters, or click the headphones to hear audio of what to enter.

After you've gone through all the steps, you'll receive an email with your code. The email will be sent to the email address you entered on the AWS credits application. It may take 48 hours to receive this email, though it is much quicker in most cases

Under "AWS Promotional Credit " in the email, you'll find your code. Use this code on the link provided, which is your [account credits page](#).

Launch an Instance

Once AWS approves your GPU Limit Increase Request, you can start the process of launching your instance.

Visit the [EC2 Management Console](#), and click on the “Launch Instance” button.

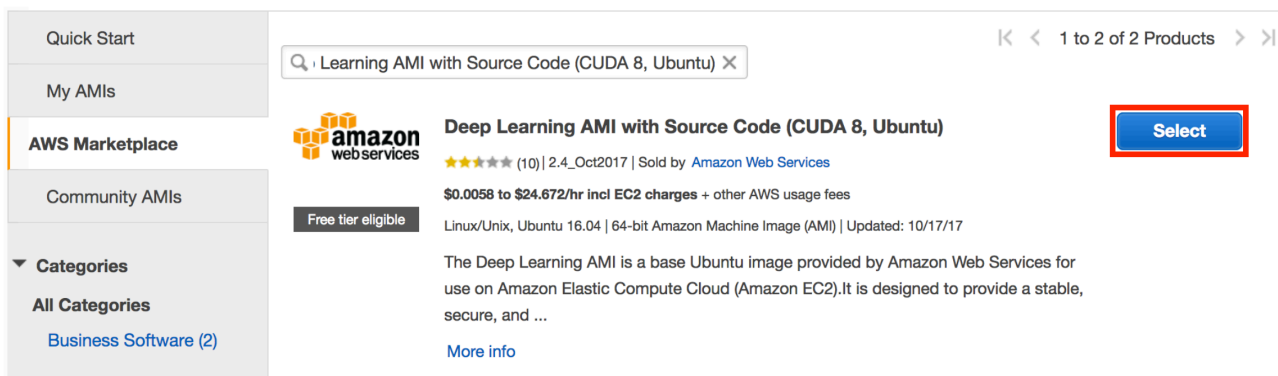
Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.



Next, you must choose an AMI (Amazon Machine Image) which defines the operating system for your instance, as well as any configurations and pre-installed software.

Click on **AWS Marketplace**, and search for **Deep Learning AMI with Source Code (CUDA 8, Ubuntu)**. Once you find the appropriate AMI, click on the "Select" button.





This [Amazon Machine Image \(AMI\)](#) contains all the environment files and drivers for you to train on a GPU. It has [cuDNN](#), and many the other packages required for this course. Any additional packages required for specific projects will be detailed in the appropriate project instructions.



Select the Instance Type

You must next choose an instance type, which is the hardware on which the AMI will run.



Filter the instance list to only show "GPU compute":

Filter by: **GPU compute**  **Current generation** 

Currently selected: p2.xlarge (11.75 ECUs, 4 vCPUs, 2.7 GHz,






	Family 	Type 	
<input checked="" type="checkbox"/>	GPU compute	p2.xlarge	
<input type="checkbox"/>	GPU compute	p2.8xlarge	
<input type="checkbox"/>	GPU compute	p2.16xlarge	

Select the p2.xlarge instance type:

Filter by: **GPU compute**  **Current generation**  [Show/Hide Columns](#)

Currently selected: p2.xlarge (11.75 ECUs, 4 vCPUs, 2.7 GHz, E5-2686v4, 61 GiB memory, EBS only)

Note: The vendor recommends using a **p2.xlarge** instance (or larger) for the best experience with this p

	Family 	Type 	vCPUs 	Memory (GiB) 	Instance Storage (GB) 	
<input checked="" type="checkbox"/>	GPU compute	p2.xlarge	4	61	EBS only	
<input type="checkbox"/>	GPU compute	p2.8xlarge	32	488	EBS only	
<input type="checkbox"/>	GPU compute	p2.16xlarge	64	732	EBS only	

Finally, click on the "Review and Launch" button:

[Cancel](#)[Previous](#)[Review and Launch](#)[Next: Configure Instance Details](#)

Configure the Security Group

Running and accessing a Jupyter notebook from AWS requires special configurations.

By default, AWS restricts access to most ports on an EC2 instance. In order to access the Jupyter notebook, you must configure the AWS Security Group to allow access to port 8888.

Click on "Edit security groups".

[Security Groups](#)[Edit security groups](#)

On the "Configure Security Group" page:

1. Select "Create a **new** security group"
2. Set the "Security group name" (i.e. "Jupyter")
3. Click "Add Rule"
4. Set a "Custom TCP Rule"
 1. Set the "Port Range" to "8888"
 2. Select "Anywhere" as the "Source"
5. Click "Review and Launch" (again)

Assign a security group: ☒ Create a new security group

☐ Select an existing security group

Security group name:

Description:

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	
SSH ▾	TCP	22	Custom ▾ 0.0.0.0/0	✕
Custom TCP Rule ▾	TCP	8888	Anywhere ▾ 0.0.0.0/0	✕

Add Rule

Launch the Instance

Click on the "Launch" button to launch your GPU instance.

Cancel

Previous

Launch

Create an Authentication Key Pair

AWS will ask if you'd like to specify an authentication key pair. You'll need to do so in order to access your instance, so you select "Create a new key pair" and click the "Download Key Pair" button. This will download a .pem file, which you'll need to be able to access your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Create a new key pair

Key pair name

YourKeyName

Download Key Pair

Move the .pem file to a secure and easily remembered location on your computer; you'll need to access your instance through the location you select.

After the .pem file has been downloaded, click the "Launch Instances" button.

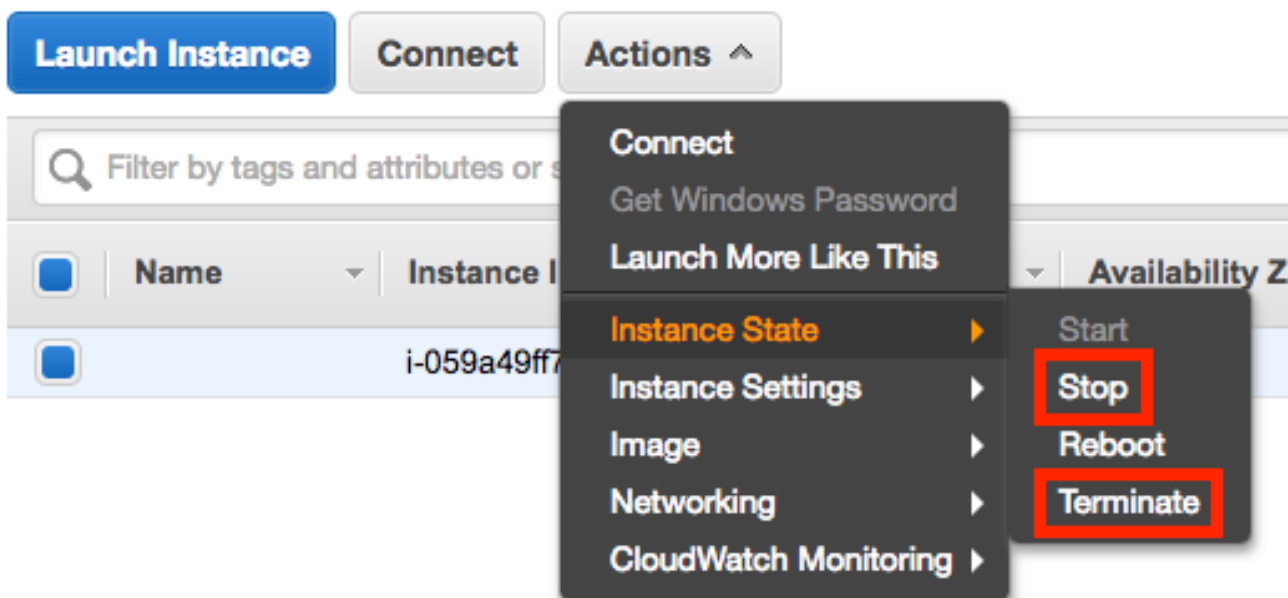
Click the "View Instances" button to go to the EC2 Management Console and watch your instance boot.

Be Careful!

From this point on, AWS will charge you for running this EC2 instance. You can find the details on the [EC2 On-Demand Pricing page](#).

Most importantly, remember to “stop” (i.e. shutdown) your instances when you are not using them. Otherwise, your instances might run for a day, week, month, or longer without you remembering, and you’ll wind up with a large bill!

AWS charges primarily for running instances, so most of the charges will cease once you stop the instance. However, there are smaller storage charges that continue to accrue until you “terminate” (i.e. delete) the instance.



There is no way to limit AWS to only a certain budget and have it auto-shutdown when it hits that threshold. However, you can set [AWS Billing Alarms](#).

Login to the Instance

After launch, your instance may take a few minutes to initialize.

Once you see “2/2 checks passed” on the EC2 Management Console, your instance is ready for you to log in.

Status Checks ▾	Alarm Status	Public DNS (IPv4) ▾	IPv4 Public IP ▾
✔ 2/2 checks ...	None	 ec2-52-8-198-63.us-we...	52.8.198.638

Instance Status Check and Public IP

Note the "IPv4 Public IP" address (in the format of “X.X.X.X”) on the EC2 Dashboard.

From a terminal, navigate to the location where you stored your .pem file. (For example, if you put your .pem file on your Desktop, `cd ~/Desktop/` will move you to the correct directory.)

Type `ssh -i YourKeyName.pem ubuntu@X.X.X.X`, where:

- `X.X.X.X` is the IPv4 Public IP found in AWS, and
- `YourKeyName.pem` is the name of your .pem file.

Note that if you've used a different AMI or specified a username, `ubuntu` will be replaced with the username, such as `ec2-user` for some Amazon AMI's. You would then instead enter `ssh -i YourKeyName.pem ec2-user@X.X.X.X`

Configure Jupyter notebook settings

In your instance, in order to create a config file for your Jupyter notebook settings, type: `jupyter notebook --generate-config`.

Then, to change the IP address config setting for notebooks (this is just a fancy one-line command to perform an exact string match replacement; you could do the same thing manually using vi/vim/nano/etc.), type: `sed -ie "s/#c.NotebookApp.ip = 'localhost'/#c.NotebookApp.ip = '*' /g" ~/.jupyter/jupyter_notebook_config.py`

Test the Instance

Make sure everything is working properly by verifying that the instance can run a Keras notebook.

On the EC2 instance

- Clone a GitHub repository
 - `git clone https://github.com/udacity/aind2-cnn.git`
- Enter the repo directory
 - `cd aind2-cnn`
- Install the requirements
 - `sudo python3 -m pip install -r requirements/requirements.txt`
- Start Jupyter notebook
 - `jupyter notebook --ip=0.0.0.0 --no-browser`

From your local machine

- You will need the token generated by your jupyter notebook to access it. On your instance terminal, there will be the following line: `Copy/paste this URL into your browser when you connect for the first time, to login with a token:`. Copy everything starting with the `:8888/?token=`.
- Access the Jupyter notebook index from your web browser by visiting: `X.X.X.X:8888/?token=...` (where X.X.X.X is the IP address of your EC2 instance and everything starting with `:8888/?token=` is what you just copied).
- Click on a folder, like "mnist", to enter it and select a notebook, such as the "mnist_mlp.ipynb" notebook.
- Run each cell in the notebook.

For some notebooks, you should see a marked decrease in training time when compared to running the same cells using a typical CPU!

NOTE: Windows users may prefer connecting via the GUI utility PuTTY, by following [these instructions](#).

Important: Cost

From this point on, AWS will charge you for a running an EC2 instance. You can find the details on the [EC2 On-Demand Pricing page](#).

Most importantly, remember to **stop** (i.e. shutdown) your instances when you are not using them. Otherwise, your instances might run for a day or a week or a month without you remembering, and you'll wind up with a large bill!

AWS charges primarily for running instances, so most of the charges will cease once you stop the instance. However, there are smaller storage charges that continue to accrue until you "terminate" (i.e. delete) the instance.

More Resources

You can use your AWS Promotional Credits to get started with AWS Machine Learning services such as Amazon SageMaker. Amazon SageMaker is a fully-managed machine learning platform that enables developers and data scientists to quickly and easily build, train, and deploy machine learning models at any scale. It removes all the barriers that typically slow down developers who want to use machine learning, so it's a perfect starting point on your machine learning journey.

[Click here](#) to learn more!