



Deep Learning - Recitation 1

What is Amazon Web Services (AWS)

Cloud Based Computational Resource



What does AWS offer?

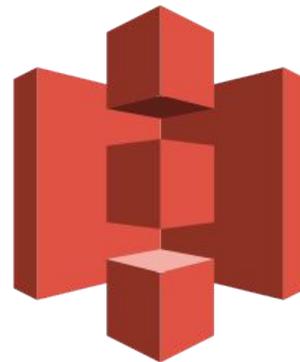
Many many things but here are the two main things to care about for DL...

EC2 - Compute Resources



Train the models

S3 - Data Storage



Store training data,
models, etc

EC2 - What kinds of machines are available?

Different types and different subtypes (you can mix and match what you want)...
Here are the ones you may care about

General Purpose:

T2 - Webservices

M3/M4 - Databases,
Fileservers, etc

Compute Optimized:

C2 - Multiplayer Gaming
Servers, scientific
computing

C3/C4 - Ad serving
machines, MMO servers,
etc

GPU Optimized:

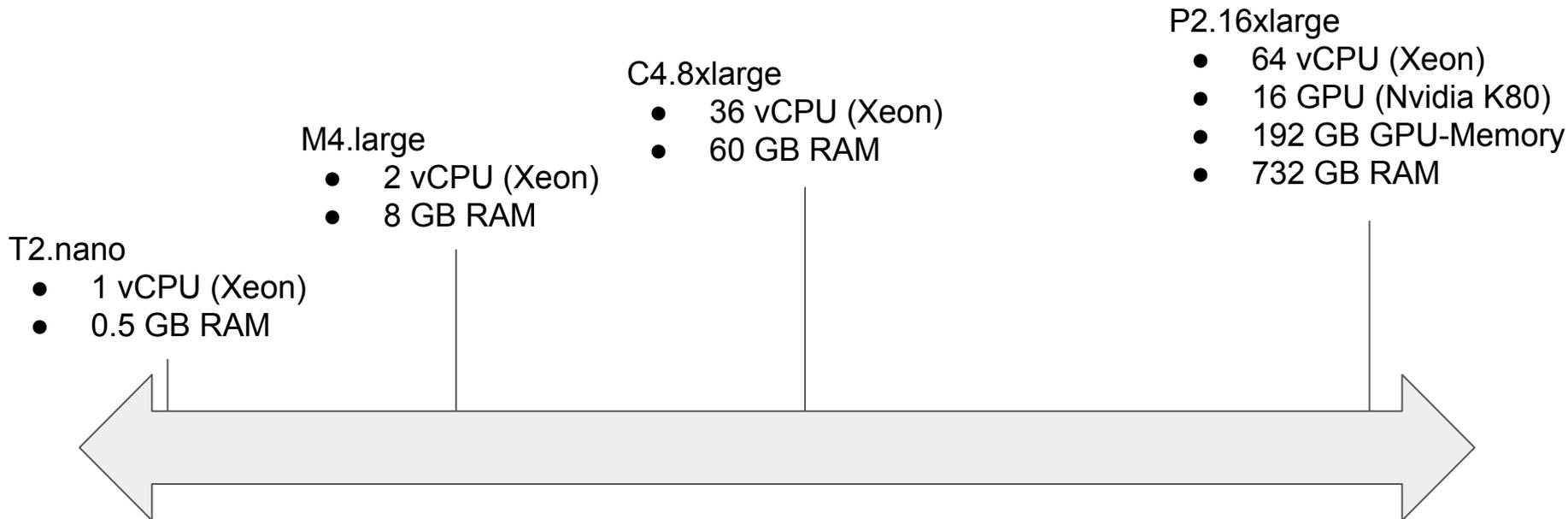
P3/P2 - Machine
Learning

G3 - Fluid dynamics,
graphics rendering, etc

Machine sizes - nano, micro, medium, large, xlarge, 2xlarge, ..., 16xlarge

EC2 - What kinds of machines are available?

Different types and different subtypes (you can mix and match what you want)...



EC2 - So what do we put on these machines?

Amazon Machine Instances (AMIs)

- Virtual images of existing machines
 - You can create an image of your machine
 - Transfer it to a different machine
 - Save it as a backup
- Use cases
 - Software packages that are incredibly difficult to install
 - Need to create multiple different machines with the exact same data for parameters servers
 - Load balancing - create a new machine with the same AMI to be used in a different region depending on load

Now you know what AWS is and what you
can do with it

Create An Account:

<https://aws.amazon.com>

Click “Sign Up” in the top right and follow the instructions

(If you already have an account you can skip this step)

Start Building on AWS Today

Whether you're looking for compute power, database storage, content delivery or other functionality, AWS has the services to help you build sophisticated applications with increased flexibility, scalability and reliability.

Create A Free Account

[View AWS Free Tier Details »](#)

AWS re:Invent

AWS re:Invent Live Streams

We'll send you a reminder with more information closer to the event

Your Email

Register Now

Broad & Deep Platform

AWS has more than 90 services and is continually launching new features and functionality.

[Learn more »](#)

Customer Success

Explore how millions of active customers every month are innovating with AWS.

[Learn more »](#)

Pace of Innovation

The AWS Cloud platform expands daily. Take a look at what we launched this week.

[Learn more »](#)

Global Infrastructure

AWS operates 43 Availability Zones within 16 geographic Regions around the world, with 11 more Availability Zones and 4 more Regions coming online soon.

[Learn more »](#)

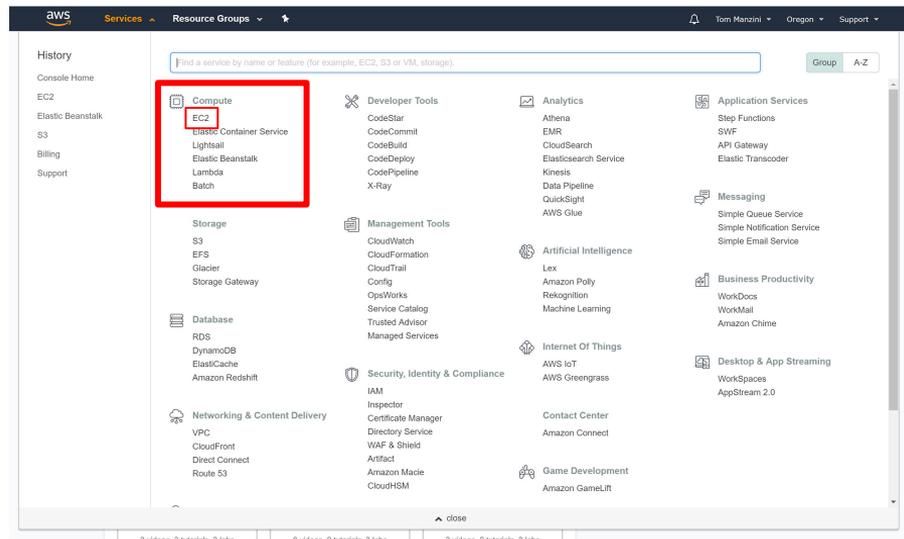
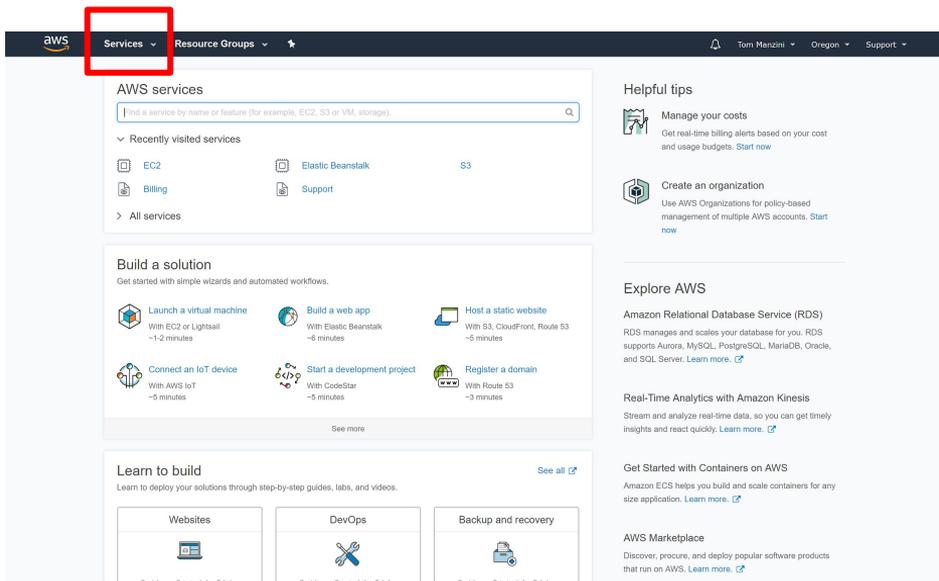


Learn with 10-Minute Tutorials



Start Building with AWS

Let's set up a basic machine



Then, under “Compute”, select “EC2”

Click on “Services” in the top left

Let's set up a basic machine

Make sure you are in the "US West (Oregon)" region

Then Click on "Running Instances"

The screenshot displays the AWS Management Console interface. At the top, the navigation bar shows the AWS logo, 'Services', 'Resource Groups', and a user profile for 'Tom Manzini' in the 'Oregon' region. The main content area is titled 'Resources' and lists various EC2 resources in the US West (Oregon) region. A red box highlights the '0 Running Instances' link. Below this, there are counts for other resources: 0 Dedicated Hosts, 1 Volumes, 2 Key Pairs, 0 Placement Groups, 0 Elastic IPs, 4 Snapshots, 0 Load Balancers, and 9 Security Groups. A promotional banner for Amazon EC2 Spot Instances is visible. The 'Create Instance' section provides instructions on how to launch a virtual server. The 'Service Health' section shows that the US West (Oregon) service is operating normally across all availability zones. The right-hand sidebar contains 'Account Attributes', 'Additional Information', and 'AWS Marketplace' sections.

Resources

You are using the following Amazon EC2 resources in the US West (Oregon) region:

- 0 Running Instances
- 0 Elastic IPs
- 0 Dedicated Hosts
- 4 Snapshots
- 1 Volumes
- 0 Load Balancers
- 2 Key Pairs
- 9 Security Groups
- 0 Placement Groups

EC2 Spot. Save up to 90% off On-Demand Prices. Turbo Boost your Workloads. Get started with Amazon EC2 Spot Instances.

Create Instance

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

[Launch Instance](#)

Note: Your instances will launch in the US West (Oregon) region

Service Health

Service Status:

- US West (Oregon): This service is operating normally

Availability Zone Status:

- us-west-2a: Availability zone is operating normally
- us-west-2b: Availability zone is operating normally
- us-west-2c: Availability zone is operating normally

[Service Health Dashboard](#)

Scheduled Events

US West (Oregon):

- No events

Account Attributes

Supported Platforms

- VPC

Default VPC

- vpc-f6d6a891

Resource ID length management

Additional Information

- Getting Started Guide
- Documentation
- All EC2 Resources
- Forums
- Pricing
- Contact Us

AWS Marketplace

Find free software trial products in the AWS Marketplace from the EC2 Launch Wizard. Or try these popular AMIs:

- [Barracuda NextGen Firewall F-Series - PAYG](#)

Provided by Barracuda Networks, Inc.
Rating ★★★★★
Starting from \$0.60/hr or from \$4,599/yr (12% savings) for software + AWS usage fees
[View all Software Infrastructure](#)

- [Splunk Insights for AWS Cloud Monitoring](#)

Provided by Splunk Inc.
Rating ★★★★★
Bring Your Own License + AWS usage fees
[View all Developer Tools](#)

Feedback English (US)

© 2008 - 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Let's set up a basic machine

The screenshot displays the AWS Management Console interface. At the top, the navigation bar shows 'Services', 'Resource Groups', and user information. The left sidebar contains navigation options for EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES, IMAGES, NETWORK & SECURITY, and LOAD BALANCING. The main content area shows the 'Instances' page with a table of instances. A red box highlights the table containing one instance with ID 'i-02d40dd9b0a197529', type 't2.micro', and state 'stopped'. Below the table, the instance details for 'i-02d40dd9b0a197529' are shown, with a red box highlighting the 'Description' tab. The details include Instance ID, state, type, availability zone, security groups, and various network and storage configurations.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
	i-02d40dd9b0a197529	t2.micro	us-west-2a	stopped		None	

Description		Status Checks	Monitoring	Tags	
Instance ID	i-02d40dd9b0a197529	Instance state	stopped	Public DNS (IPv4)	-
Instance type	t2.micro	Elastic IPs	-	IPv4 Public IP	-
Availability zone	us-west-2a	Security groups	launch-wizard-1 . view inbound rules	IPv6 IPs	-
Private DNS	ip-172-31-45-0.us-west-2.compute.internal	Scheduled events	-	Private DNS	ip-172-31-45-0.us-west-2.compute.internal
Private IPs	172.31.45.0	AMI ID	Loading ami-92f420ea...	Private IPs	172.31.45.0
Secondary private IPs	-	Platform	-	VPC ID	vpc-f6d6a891
VPC ID	vpc-f6d6a891	IAM role	-	Subnet ID	subnet-18ed4551
Subnet ID	subnet-18ed4551	Key pair name	tmanzini-personal	Network interfaces	eth0
Network interfaces	eth0	Owner	180942078756	Source/dest. check	True
Source/dest. check	True	Launch time	November 7, 2017 at 11:35:49 PM UTC-5 (448 hours)	EBS-optimized	False
EBS-optimized	False	Termination protection	-	Root device type	ebs
Root device type	ebs	Lifecycle	normal	Root device	/dev/sda1
Root device	/dev/sda1	Monitoring	basic	Block devices	/dev/sda1
Block devices	/dev/sda1	Alarm status	None	Elastic GPU	-
Elastic GPU	-	Kernel ID	-	Elastic GPU type	-
Elastic GPU type	-			Elastic GPU status	-

Here you see your current instances

(I have 1 that is stopped, you shouldn't have any)

Here are the details of that instance

Let's set up a basic machine

Click "Launch Instance"

The screenshot displays the AWS Management Console interface. At the top, the navigation bar includes the AWS logo, 'Services', 'Resource Groups', and user information (Tom Manzini, Oregon, Support). A red box highlights the 'Launch Instance' button in the top-left navigation pane. The main content area shows a table of EC2 instances with the following columns: Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS (IPv4). A single instance is listed with Instance ID 'i-02d40dd9b0a197529', Instance Type 't2.micro', Availability Zone 'us-west-2a', and Instance State 'stopped'. Below the table, the details for this instance are shown, including its Private IP (172.31.45.0) and various configuration parameters.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
	i-02d40dd9b0a197529	t2.micro	us-west-2a	stopped		None	

Instance: i-02d40dd9b0a197529		Private IP: 172.31.45.0	
Description	Status Checks	Monitoring	Tags
Instance ID	i-02d40dd9b0a197529	Public DNS (IPv4)	-
Instance state	stopped	IPv4 Public IP	-
Instance type	t2.micro	IPv6 IPs	-
Elastic IPs	-	Private DNS	ip-172-31-45-0.us-west-2.compute.internal
Availability zone	us-west-2a	Private IPs	172.31.45.0
Security groups	launch-wizard-1 . view inbound rules	Secondary private IPs	-
Scheduled events	-	VPC ID	vpc-f6d6a891
AMI ID	Loading ami-92f420ea...	Subnet ID	subnet-18ed4551
Platform	-	Network interfaces	eth0
IAM role	-	Source/dest. check	True
Key pair name	tmanzini-personal	EBS-optimized	False
Owner	180942078756	Root device type	ebs
Launch time	November 7, 2017 at 11:35:49 PM UTC-5 (448 hours)	Root device	/dev/sda1
Termination protection	-	Block devices	/dev/sda1
Lifecycle	normal	Elastic GPU	-
Monitoring	basic	Elastic GPU type	-
Alarm status	None	Elastic GPU status	-
Kernel ID	-		

Let's set up a basic machine

1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review

Step 1: Choose an Amazon Machine Image (AMI)

[Cancel and Exit](#)

[Launch a database using RDS](#)

Ubuntu Server 16.04 LTS (HVM), SSD Volume Type - ami-076e276d85f524150 (64-bit x86) / ami-05e1b2aec3b47890f (64-bit Arm)

Free tier eligible

Ubuntu Server 16.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)
 64-bit (Arm)

Microsoft Windows Server 2016 Base - ami-019e99815e07ceb49

Windows

Free tier eligible

Microsoft Windows 2016 Datacenter edition. [English]

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)

Deep Learning AMI (Ubuntu) Version 20.0 - ami-0d0ff0945ae093aea

With latest deep learning frameworks pre-installed: MXNet, TensorFlow, PyTorch, Keras, Chainer, Caffe/2, Theano & CNTK, configured with NVIDIA CUDA, cuDNN, NCCL & Intel MKL-DNN. For a fully managed experience, check: <https://aws.amazon.com/sagemaker>

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)

Deep Learning AMI (Amazon Linux) Version 20.0 - ami-0305a0d7a68489e58

Amazon Linux

With latest deep learning frameworks pre-installed: MXNet, TensorFlow, PyTorch, Keras, Chainer, Caffe/2, Theano & CNTK, configured with NVIDIA CUDA, cuDNN, NCCL & Intel MKL-DNN. For a fully managed experience, check: <https://aws.amazon.com/sagemaker>

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)

Deep Learning Base AMI (Ubuntu) Version 14.0 - ami-015eb46ac552e435f

Comes with foundational platform of NVidia CUDA, cuDNN, NCCL, GPU Drivers, Intel MKL-DNN and other system libraries to deploy your own custom deep learning environment. For a fully managed experience, check: <https://aws.amazon.com/sagemaker>

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)

Let's set up a basic machine

aws Services Resource Groups Tom Manzini Oregon Support

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All instance types Current generation Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro <small>Free tier eligible</small>	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes
<input type="checkbox"/>	General purpose	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
<input type="checkbox"/>	General purpose	m4.large	2	8	EBS only	Yes	Moderate	Yes
<input type="checkbox"/>	General purpose	m4.xlarge	4	16	EBS only	Yes	High	Yes
<input type="checkbox"/>	General purpose	m4.2xlarge	8	32	EBS only	Yes	High	Yes
<input type="checkbox"/>	General purpose	m4.4xlarge	16	64	EBS only	Yes	High	Yes

Cancel Previous Review and Launch Next: Configure Instance Details

Select the t2-micro because it is “free tier eligible”

Select Next

Let's set up a basic machine

- 1. Choose AMI
- 2. Choose Instance Type
- 3. Configure Instance
- 4. Add Storage
- 5. Add Tags
- 6. Configure Security Group
- 7. Review

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1 [Launch into Auto Scaling Group](#)

Purchasing option: Request Spot instances

Network: vpc-f6d6a891 (default) [Create new VPC](#)

Subnet: No preference (default subnet in any Availability Zone) [Create new subnet](#)

Auto-assign Public IP: Use subnet setting (Enable)

IAM role: None [Create new IAM role](#)

Shutdown behavior: Stop

Enable termination protection: Protect against accidental termination

Monitoring: Enable CloudWatch detailed monitoring
[Additional charges apply.](#)

Tenancy: Shared - Run a shared hardware instance
[Additional charges will apply for dedicated tenancy.](#)

▶ Advanced Details

Cancel Previous **Review and Launch** **Next: Add Storage**

Just select next

Let's set up a basic machine

- 1. Choose AMI
- 2. Choose Instance Type
- 3. Configure Instance
- 4. Add Storage
- 5. Add Tags
- 6. Configure Security Group
- 7. Review

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-0b9c16d670f4e2685	8	General Purpose SSD (GP2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

Add New Volume

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

Make sure you choose 8 GB of SSD Storage Space

Cancel Previous Review and Launch Next: Add Tags

Select next

Let's set up a basic machine

- 1. Choose AMI
- 2. Choose Instance Type
- 3. Configure Instance
- 4. Add Storage
- 5. Add Tags
- 6. Configure Security Group
- 7. Review

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key (127 characters maximum)	Value (255 characters maximum)	Instances ⓘ	Volumes ⓘ
------------------------------	--------------------------------	-------------	-----------

This resource currently has no tags

Choose the Add tag button or [click to add a Name tag](#).
Make sure your [IAM policy](#) includes permissions to create tags.

Add Tag (Up to 50 tags maximum)

Select next

Let's set up a basic machine

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more about Amazon EC2 security groups.](#)

Assign a security group: Create a new security group
 Select an existing security group

Security group name:

Description:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
Custom TCP f	TCP	8888	Anywhere 0.0.0.0/0	e.g. SSH for Admin Desktop

Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Cancel Previous **Review and Launch**

Make sure you have an SSH rule set (This should be default) & a TCP rule for 8888 so you can connect your Ipython Notebook

You can set HTTP or other rules here too if you want

Select Review and Launch

Let's set up a basic machine

- 1. Choose AMI
- 2. Choose Instance Type
- 3. Configure Instance
- 4. Add Storage
- 5. Add Tags
- 6. Configure Security Group
- 7. Review

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

⚠ Improve your instances' security. Your security group, launch-wizard-4, is open to the world.
Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only. You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. [Edit security groups](#)

AMI Details [Edit AMI](#)

Ubuntu Server 16.04 LTS (HVM), SSD Volume Type - ami-0a00ce72
Free tier eligible
Ubuntu Server 16.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).
Root Device Type: ebs Virtualization type: hvm

Instance Type [Edit instance type](#)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Security Groups [Edit security groups](#)

Security group name: launch-wizard-4
Description: launch-wizard-4 created 2017-11-26T17:20:35.221-05:00

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	0.0.0.0/0	

Instance Details [Edit instance details](#)

Cancel Previous **Launch**

Take one last look to make sure you are happy with everything...

Select Launch

Let's set up a basic machine

Enter a name
for your private
key

Select an existing key pair or create a new key pair ✕

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into 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

 You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

This window allows you to create a private key to access your machine... when you ssh you will need it

Download your key and save it

Make sure it's in a place where it won't get lost - this key is the only way to connect to this specific instance!

Now you have a basic *AWS* machine up
and running

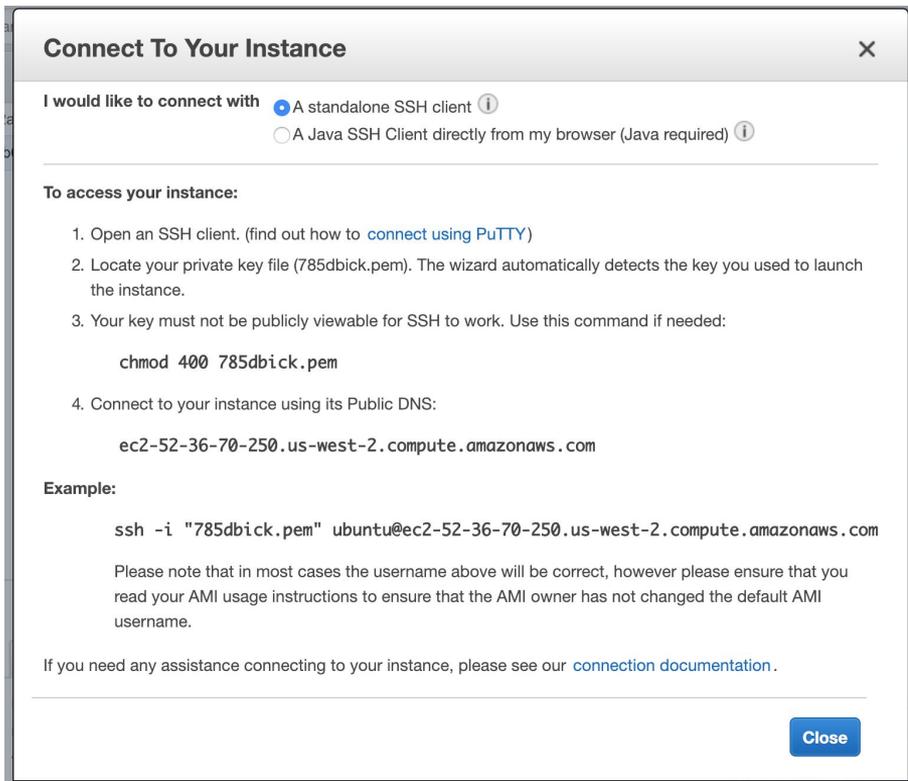
Now let's connect to it

Your instance is now identified by its IP address. You can ssh to your instance by using this public IP address and your address key.

Let's start by changing the instance type. For student accounts, t2.micro is free-tier eligible, but let's still start on a t2.nano instance.

Remember your instance's IP address changes every time you restart it

Connect to launched instance



Connect To Your Instance ×

I would like to connect with A standalone SSH client ⓘ
 A Java SSH Client directly from my browser (Java required) ⓘ

To access your instance:

1. Open an SSH client. (find out how to [connect using PuTTY](#))
2. Locate your private key file (785dbick.pem). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:

```
chmod 400 785dbick.pem
```
4. Connect to your instance using its Public DNS:

```
ec2-52-36-70-250.us-west-2.compute.amazonaws.com
```

Example:

```
ssh -i "785dbick.pem" ubuntu@ec2-52-36-70-250.us-west-2.compute.amazonaws.com
```

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

If you need any assistance connecting to your instance, please see our [connection documentation](#).

Close

Right-click and click connect

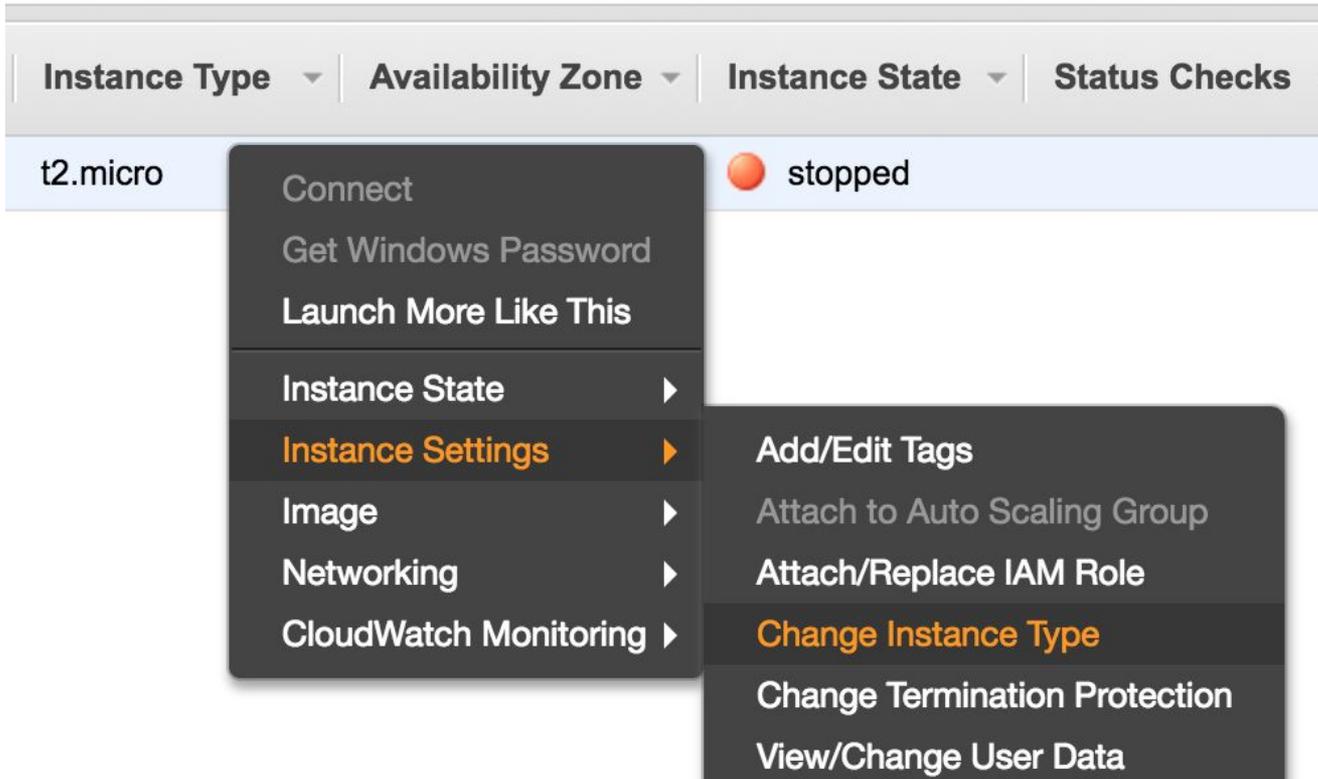
The ssh command shown includes the currently assigned IP address.

This will change at each launch!

Note - the ssh command will use the path to the key associated with this instance.

Important - will need to run chmod command every time you have a new key

Set instance type



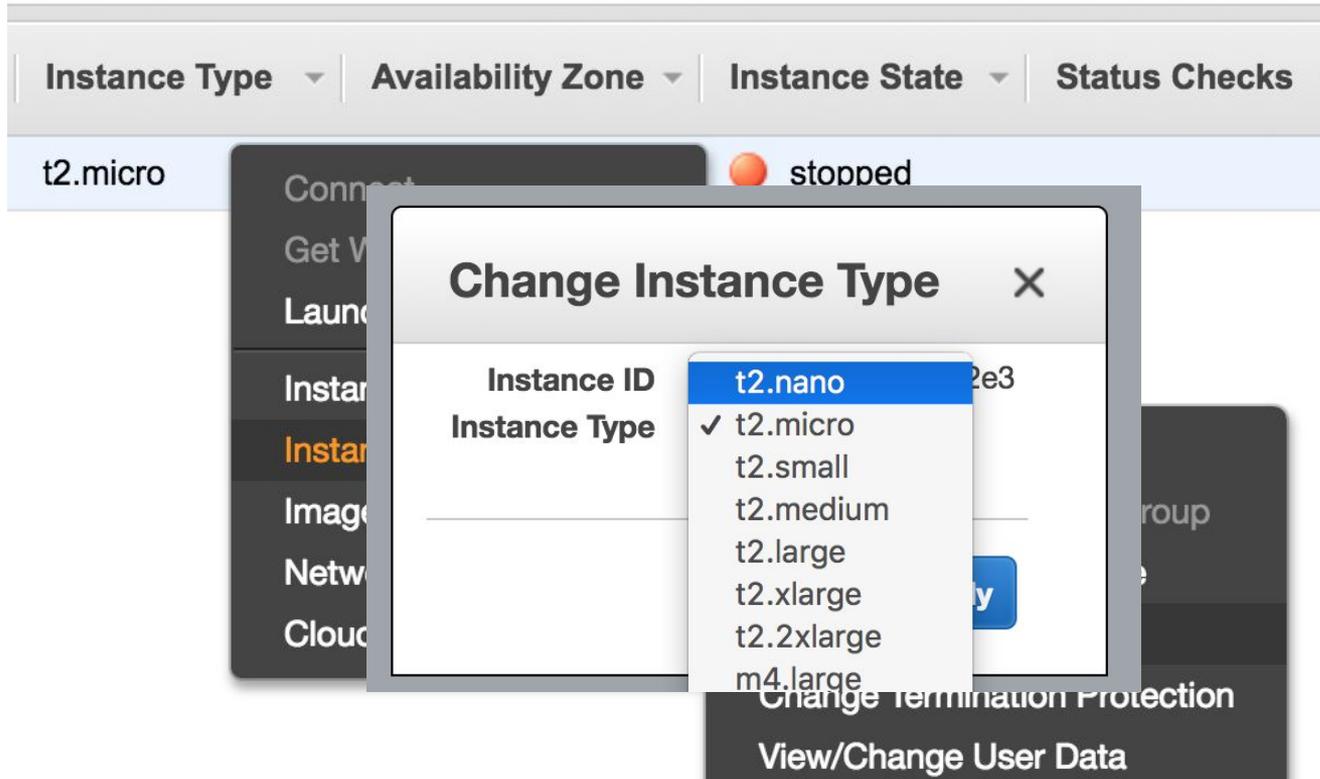
The screenshot shows the AWS Management Console interface for an EC2 instance. At the top, there are four tabs: "Instance Type", "Availability Zone", "Instance State", and "Status Checks". Below these tabs, the instance name "t2.micro" is displayed on the left. To its right, the instance state is shown as "stopped" with a red stop button icon. A context menu is open over the instance, listing various actions. The "Instance Settings" option is highlighted in orange, and its sub-menu is also open, with "Change Instance Type" highlighted in orange.

Instance Type	Availability Zone	Instance State	Status Checks
t2.micro		stopped	

- Connect
- Get Windows Password
- Launch More Like This
- Instance State ▶
- Instance Settings** ▶
 - Add/Edit Tags
 - Attach to Auto Scaling Group
 - Attach/Replace IAM Role
 - Change Instance Type**
 - Change Termination Protection
 - View/Change User Data
- Image ▶
- Networking ▶
- CloudWatch Monitoring ▶

Right-click on your instance under the instance tab, go to instance settings and change instance type to t2.nano

Set instance type



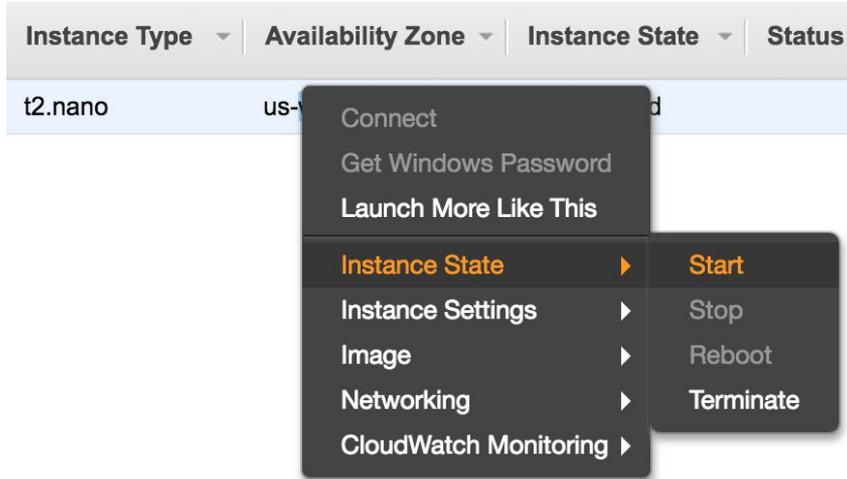
Right-click on your instance under the instance tab, go to instance settings and change instance type to t2.nano

Note

You can spin up a t2.micro for development (which allows access to all the packages in the AMI and can put all your data on the instance), and then change the instance type to a p2.xlarge for GPU access (training).

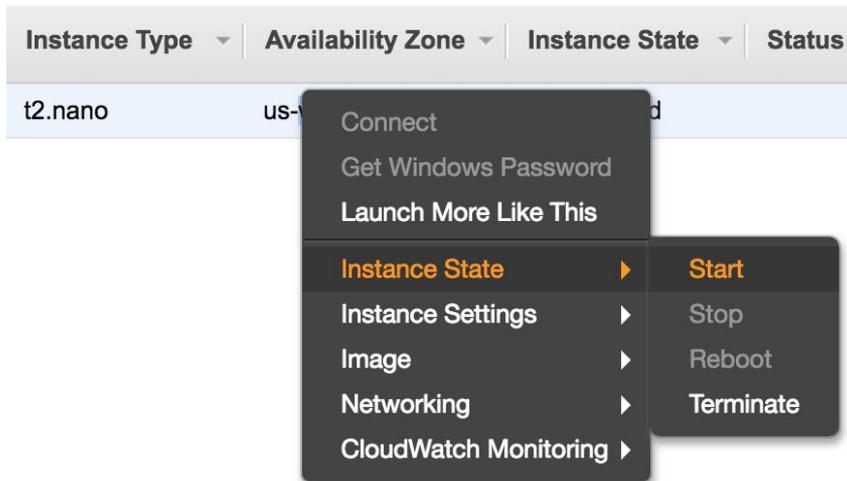
p2.xlarge is what I used for all my training, about \$1 per hour, so with 3 credits of \$50 that's more than enough hours of training for almost allcases.

Launch instance



Right-click and Start your instance.

Launch instance



Instance Type	Availability Zone	Instance State
t2.nano	us-west-2a	 pending

Right-click and Start your instance.

This will start the process of allocating resources to your instance. Once this is completed, your instance will be running and you can connect to it.

Stopping this instance removes the compute associated with the current session.

Do NOT terminate! This will wipe the slate clean. Terminate ONLY when you are sure of 'throwing away' the data.

Now you have a running AWS machine
and you can connect

Now you know how to use machines on
EC2

Let's run a trivial little problem

A Toy Problem to Test your AWS Instance

Will Test:

- Working instance
- Good Python installation
- Working PyTorch installed
- GPU / CUDA support
- Your ability to login and execute code on AWS

```
import torch

def main():
    GPU = torch.cuda.is_available()
    mat_size = (100, 100)
    cpu_mat_0 = torch.zeros(size = mat_size)
    cpu_mat_1 = torch.ones(size = mat_size)
    gpu_mat_0 = torch.zeros(size = mat_size)
    gpu_mat_1 = torch.ones(size = mat_size)

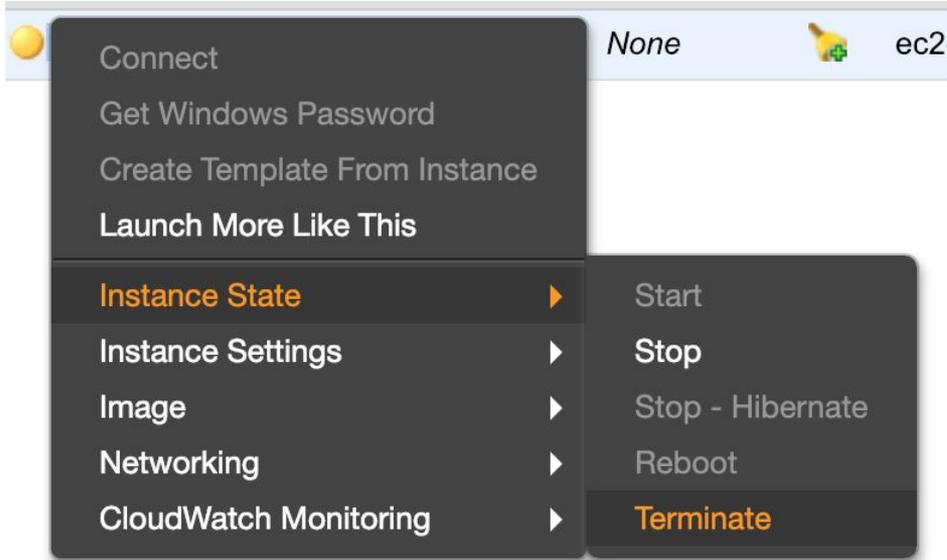
    if GPU:
        gpu_mat_0 = gpu_mat_0.cuda()
        gpu_mat_1 = gpu_mat_1.cuda()
        print("Using GPU")

    cpu_res = cpu_mat_0 + cpu_mat_1
    gpu_res = gpu_mat_0 + gpu_mat_1

    try:
        print(bool(torch.all(cpu_res == gpu_res)))
        return 0
    except:
        print("If using GPU, should be here")
        gpu_res = gpu_res.detach().cpu() # detach is for gradient computations
        print(bool(torch.all(cpu_res == gpu_res)))
        return 0

main()
```

And let's shut it down so you don't get billed



If you stop the instance you can just start the instance and resume whenever you want.

If you are done, terminate.

Now lets redeem some AWS credits

The screenshot shows the AWS Management Console interface. At the top, the navigation bar includes the AWS logo, "Services", "Resource Groups", and a user profile dropdown for "Tom Manzini" in the "Oregon" region. The user menu is open, showing options: "My Account", "My Organization", "My Billing Dashboard" (highlighted with a red box), "My Security Credentials", and "Sign Out".

The main content area is divided into several sections:

- AWS services:** A search bar and a list of "Recently visited services" including EC2, IAM, Billing, Support, and Elastic Beanstalk.
- Build a solution:** A section for getting started with simple wizards and automated workflows, featuring cards for "Launch a virtual machine", "Build a web app", "Host a static website", "Connect an IoT device", "Start a development project", and "Register a domain".
- Learn to build:** A section for deploying solutions through guides, labs, and videos, with categories for Websites, DevOps, Backup and recovery, Big data, Databases, and Mobile.

On the right side, there are sections for "Helpful tips" (Manage your costs, Create an organization), "Explore AWS" (Amazon RDS, Amazon Kinesis), "Get Started with Containers on AWS", and "AWS Marketplace".

Now lets redeem some AWS credits

The screenshot shows the AWS Billing & Cost Management Dashboard. The left-hand navigation menu includes: Dashboard, Bills, Cost Explorer, Budgets, Reports, Cost Allocation Tags, Payment Methods, Payment History, Consolidated Billing, Credits (highlighted with a red box), and DevPay. The main content area is titled "Billing & Cost Management Dashboard" and features a "Spend Summary" section with a "Cost Explorer" button. A large circular gauge displays "\$0". Below the gauge is a table with the following data:

Category	Amount
No Amount Due	\$0.00
Tax	\$0.00
Total	\$0.00

The dashboard also includes a "Month-to-Date Spend by Service" section with a "Bill Details" button and a "What's New in AWS Billing and Cost Management?" section with links to AWS Budgets, Cost Explorer, and Cost and Usage Reports. A line chart shows the current month-to-date balance for December 2017 as \$0.00, compared to last month (November 2017) and a forecast for December 2017, both also at \$0.00. An "Alerts & Notifications" section at the bottom provides information about the AWS Free Usage Tier and billing alerts.

Now lets redeem some AWS credits

aws Services Resource Groups

Tom Manzini Global Support

Dashboard Bills Cost Explorer Budgets Reports Cost Allocation Tags Payment Methods Payment History Consolidated Billing Preferences Credits Tax Settings DevPay

Credits

Please enter your code below to redeem your credits.

Promo Code

Security Check  Refresh Image

Please type the characters as shown above

By clicking "Redeem" you indicate that you have read and agree to the terms of the AWS Promotional Credit Terms & Conditions located here.

Below are all the credits you have redeemed with AWS. Credits will automatically be applied to your bill. Only credits that apply to a specific service can be used.

Expiration Date	Credit Name	Credits Used	Credits Remaining	Applicable Products
2018-03-31	EDU_ENG_FY2017_Q1_1_CMU_50USD	\$44.22	\$5.78	See complete list
2018-03-31	EDU_ENG_FY2017_Q1_1_CMU_50USD	\$45.05	\$4.95	See complete list

Total Amount of Credits Remaining: \$10.73

Now you know how to use machines on
EC2 and you can afford to use the
expensive ones

Finally, some useful tidbits

Remember to refer back here later in the semester, likely won't remember all this right now

Editing Volume

If you run out of space, you can adjust your volume without shutting off the instance, happened to me on hw2p2

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-modify-volume.html>

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/console-modify.html>

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Launch Templates

Spot Requests

Reserved Instances

Dedicated Hosts

Scheduled Instances

Capacity Reservations

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Volumes

Snapshots

Lifecycle Manager

NETWORK & SECURITY

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Network Interfaces

LOAD BALANCING

Create Volume

Actions

Modify Volume

Create Snapshot

Delete Volume

Attach Volume

Detach Volume

Force Detach Volume

Change Auto-Enable IO Setting

Add/Edit Tags

Filter by tags

Name

Volume Type

IOPS

Snapshot

Created

Availability Zone

State

Alarm Status

Attachment Information

Monitoring

gp2

225

snap-014390f4...

December 21, 2018...

us-west-2a

in-use

None

i-08d9e0db1b613edb...

Volumes: **vol-0adb44e78d8f4ab21**

Description

Status Checks

Monitoring

Tags

Volume ID	vol-0adb44e78d8f4ab21
Size	75 GiB
Created	December 21, 2018 at 12:37:08 AM UTC-5
State	in-use
Attachment information	i-08d9e0db1b613edb5 /dev/sda1 (attached)
Volume type	gp2

Alarm status	None
Snapshot	snap-014390f4a84518e09
Availability Zone	us-west-2a
Encrypted	Not Encrypted
KMS Key ID	
KMS Key Aliases	

EC2 Dashboard

Events

Tags

Reports

Limits

INSTANCES

Instances

Launch Templates

Spot Requests

Reserved Instances

Dedicated Hosts

Scheduled Instances

Capacity Reservations

IMAGES

AMIs

Bundle Tasks

ELASTIC BLOCK STORE

Volumes

Snapshots

Lifecycle Manager

NETWORK & SECURITY

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Network Interfaces

LOAD BALANCING

Create Volume

Actions ▾

Filter by tags and attributes or search by keyword

<input type="checkbox"/>	Name	Volume ID	Size	Volume Type	IOPS	Snapshot	Created	Availability Zone	State	Alarm Status	Attachment Informati	Monitoring
<input type="checkbox"/>		vol-0adb44e...	75 GiB	gp2	225	snap-014390f4...	December 21, 2018...	us-west-2a	in-use	None	i-08d9e0db1b613edb...	<input type="checkbox"/>

Modify Volume


Volume ID vol-0adb44e78d8f4ab21

Volume Type General Purpose SSD (gp2) ⓘ

Size 75 (Min: 1 GiB, Max: 16384 GiB) ⓘ

IOPS 225 / 3000 (Baseline of 3 IOPS per GiB with a minimum of 100 IOPS, burstable to 3000 IOPS) ⓘ

Cancel

Modify

Volumes: | vol-0adb44e78d8f4ab21

Description

Status Checks

Mod...

Volume ID | vol-0adb44e78d8f4ab21
 Size | 75 GiB
 Created | Dec 21, 2018 10:00 AM
 State | in-use
 Attachment information | i-08d9e0db1b613edb5:/dev/sda1 (attached)
 Volume type | gp2

Encrypted | Not Encrypted
 KMS Key ID |
 KMS Key Aliases |

Here's How to Connect to Jupyter Notebook

Call `ssh -N -L localhost:8888:localhost:8887 -i ~/path/to/key ubuntu@ip`

Create new terminal window, ssh into instance

In ssh window, source activate `pytorch_p36`

Call `jupyter notebook --no-browser --port=8887`

Go to browser, enter `localhost:8888`

Parting wisdom

- Remember to shut down your machines
 - Just because you don't have an open ssh connection doesn't mean your machine is off
 - When you're not running code, Stop; when you can get rid of the data, Terminate
- Use PyTorch on the previously given AMI
 - After you ssh into the instance, run command 'source activate pytorch_p36'
 - Without this you cannot import torch
 - Unless you are already very comfortable with tensorflow, it is much easier to get help from TAs on PyTorch

Parting wisdom

- You need permission to launch a GPU instance
 - You will need to create a support ticket to launch GPU Machines
 - If you attempt to launch on you will be guided through the process
 - You are typically only allowed to launch 1 GPU machine at a time
- Only launch an expensive instance when it is time to train, not develop
 - Launch a basic instance with everything you need to develop and test your code
 - When it comes time to train your system for real, then launch a decked out instance
 - Saves you money

Parting wisdom

If you want to use a local IDE instead of VIM on your ssh window, you can develop locally and run an scp to transfer the file to the aws instance

```
scp -i ~/path/to/key ~/path/to/file/ ubuntu@ec2...
```

If you use jupyter notebook connected to aws, no need to worry about this.

While writing code, I kept a note with common commands that I could just change the IP address on, such as an ssh command, and scp of my local code

Requesting Instances

Must request access to instances from AWS in the Support Center section of AWS. Check the latest status on AWS, as they may have changed something or upgraded offerings. Currently, should request EC2 instance limit increase, but be sure that it has not changed.

How to Request

Instructions to apply for GPU access on AWS

Starting from aws.amazon.com

Top right corner, "My Account" -> "Account Settings"

At top right corner click "Support" -> Support Center

Click "Create case"

Click "Service Limit increase"

Limit type -> EC2 Instance

If Pitt, select Region US East (Ohio). If SV, select Region US West (Oregon)

New limit value -> 1 Instance type -> p2.xlarge (can also request other instance types)

Use case description -> Describe that you are in this class at CMU and you need a GPU to train deep learning models for the homeworks

Specify your contact method