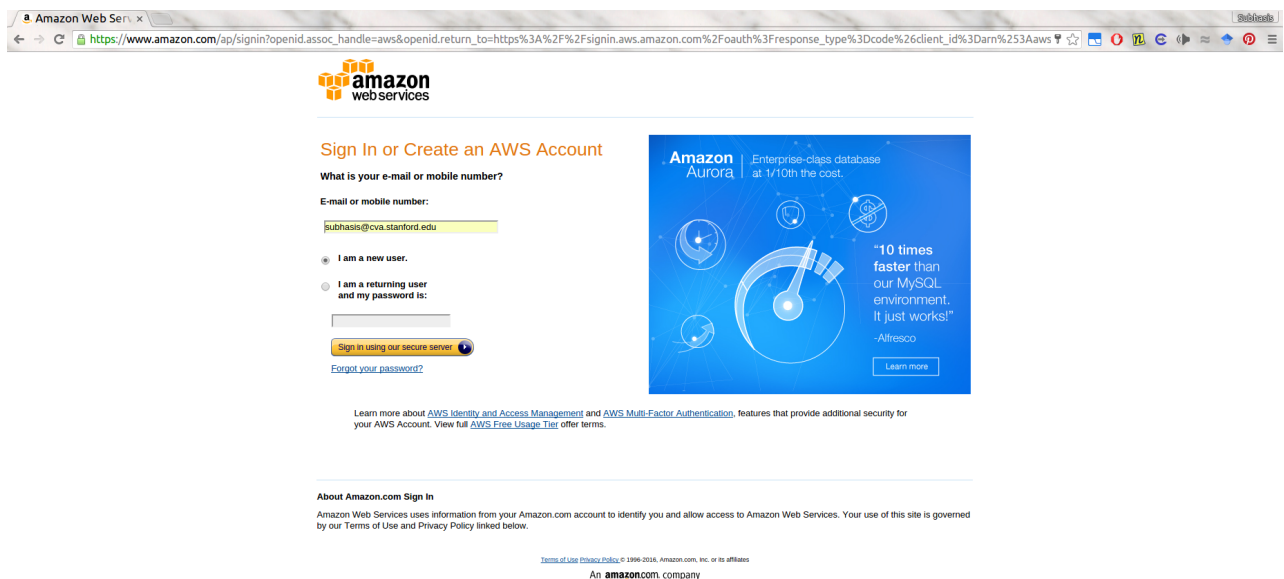


AWS Tutorial

For GPU instances, we also have an Amazon Machine Image (AMI) that you can use to launch GPU instances on Amazon EC2. This tutorial goes through how to set up your own EC2 instance with the provided AMI. **We do not currently distribute AWS credits to CS231N students but you are welcome to use this snapshot on your own budget.**

TL;DR for the AWS-savvy: Our image is `cs231n_caffe_torch7_keras_lasagne_v2`, AMI ID: `ami-125b2c72` in the us-west-1 region. Use a `g2.2xlarge` instance. Caffe, Torch7, Theano, Keras and Lasagne are pre-installed. Python bindings of caffe are available. It has CUDA 7.5 and CuDNN v3.

First, if you don't have an AWS account already, create one by going to the [AWS homepage](#), and clicking on the yellow "Sign In to the Console" button. It will direct you to a signup page which looks like the following.



Select the "I am a new user" checkbox, click the "Sign in using our secure server" button, and follow the subsequent pages to provide the required details. They will ask for a credit card information, and also a phone verification, so have your phone and credit card ready.

Once you have signed up, go back to the [AWS homepage](#), click on “Sign In to the Console”, and this time sign in using your username and password.

The screenshot shows the AWS sign-in page. At the top, it says "Sign In or Create an AWS Account". Below this, it asks "What is your e-mail or mobile number?". There is a text input field containing "subhasis@cva.stanford.edu". Below the input field, there are two radio buttons: "I am a new user." and "I am a returning user and my password is:". The second option is selected. Below the radio buttons, there is a password input field. To the right of the input fields, there is a blue box for "Amazon Aurora" with the text "Enterprise-class database at 1/10th the cost." and "10 times faster than our MySQL environment. It just works!" by -Alfresco. Below the input fields, there is a button "Sign in using our secure server" and a link "Forgot your password?". At the bottom, there is a link "Learn more about AWS Identity and Access Management and AWS Multi-Factor Authentication, features that provide additional security for your AWS Account. View full AWS Free Usage Tier offer terms." and a footer "About Amazon.com Sign In" with a link to the Terms of Use and Privacy Policy.

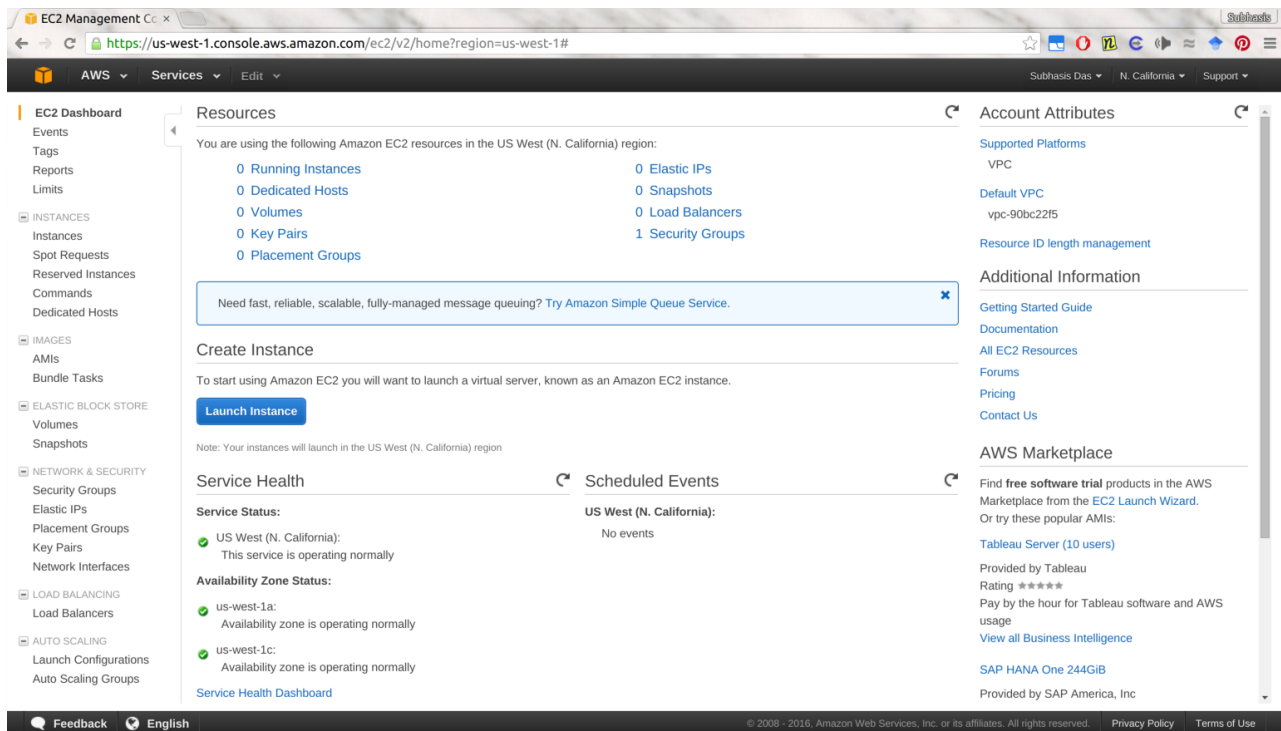
Once you have signed in, you will be greeted by a page like this:

The screenshot shows the AWS Management Console home page. The top navigation bar includes "AWS", "Services", and "Edit". The main content area is titled "Amazon Web Services" and lists various services categorized into Compute, Storage & Content Delivery, Database, Networking, Developer Tools, Management Tools, Security & Identity, Analytics, Internet of Things, Mobile Services, Application Services, and Enterprise Applications. On the right side, there is a "Resource Groups" section with a "Create a Group" button and a "Tag Editor" button. Below this, there is an "Additional Resources" section with links to "Getting Started", "AWS Console Mobile App", "AWS Marketplace", and "AWS re:Invent Announcements". At the bottom right, there is a "Service Health" section showing the status of "AWS Storage Gateway (Tokyo)".

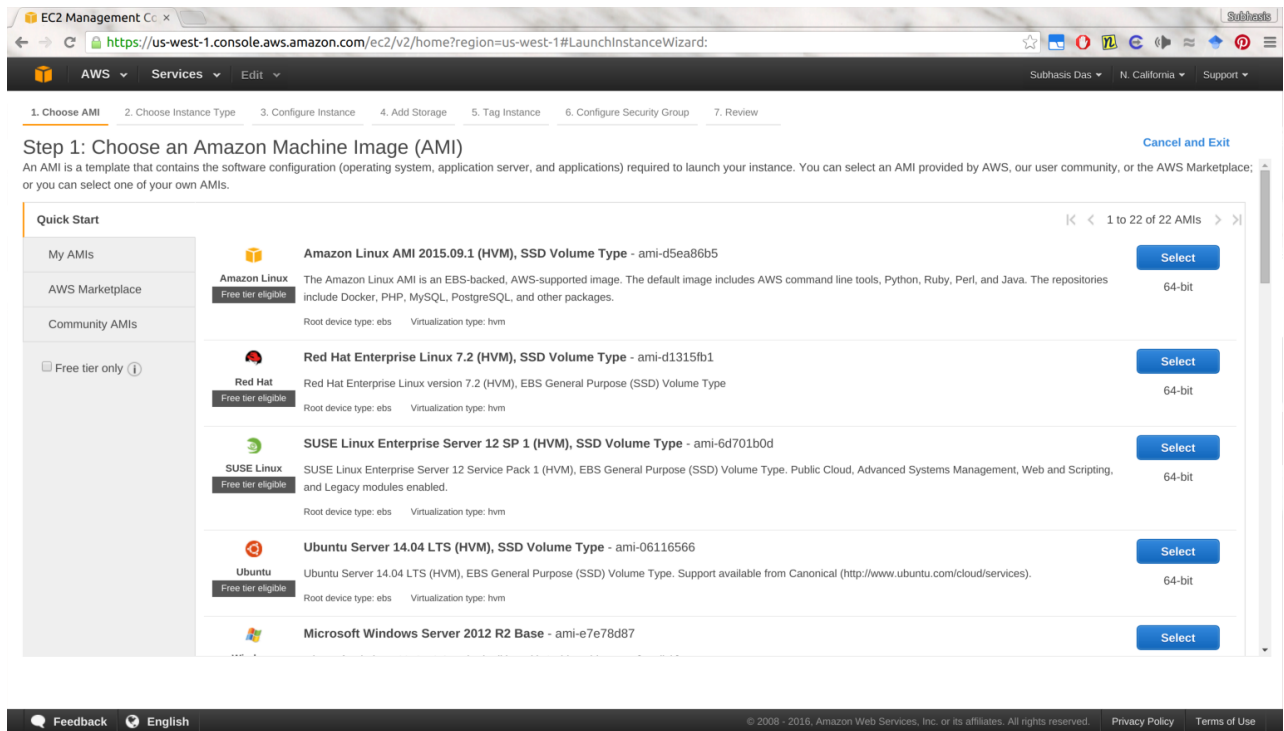
Make sure that the region information on the top right is set to N. California. If it is not, change it to N. California by selecting from the dropdown menu there.

(Note that the subsequent steps requires your account to be “Verified” by Amazon. This may take up to 2 hrs, and you may not be able to launch instances until your account verification is complete.)

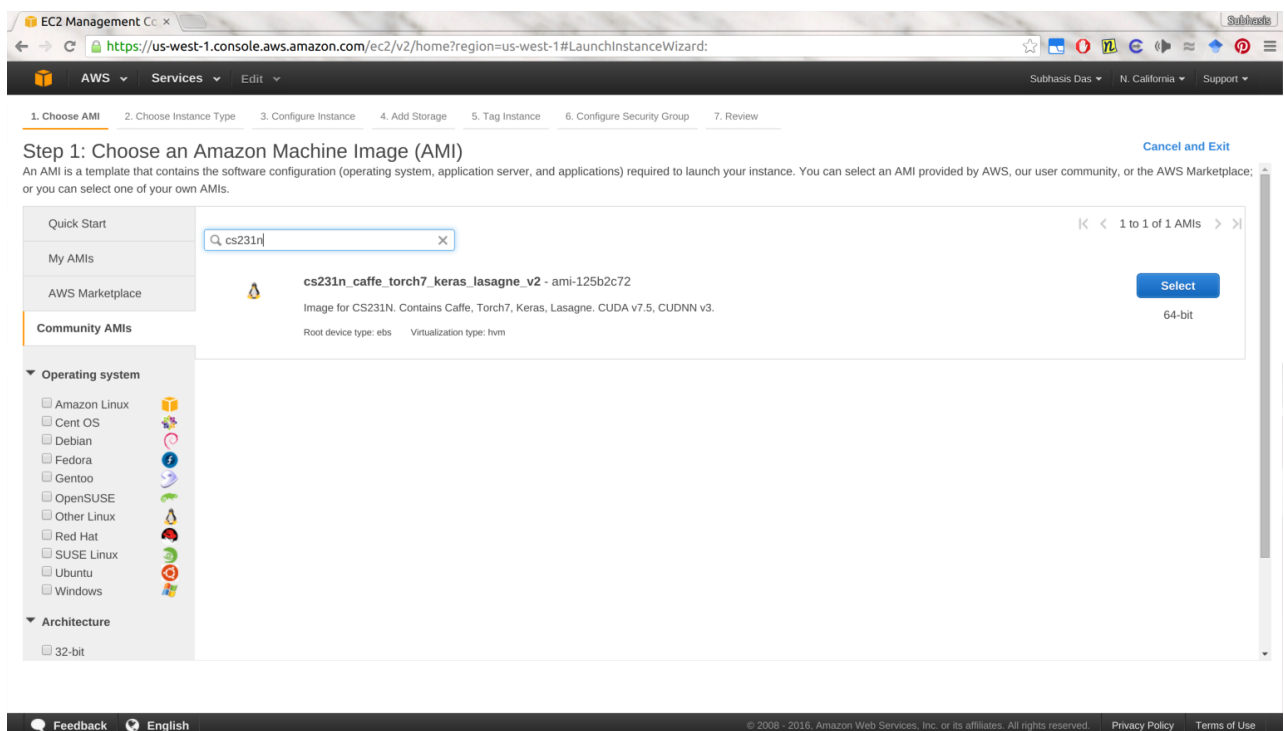
Next, click on the EC2 link (first link under the Compute category). You will go to a dashboard page like this:



Click the blue “Launch Instance” button, and you will be redirected to a page like the following:



Click on the “Community AMIs” link on the left sidebar, and search for “cs231n” in the search box. You should be able to see the AMI `cs231n_caffe_torch7_keras_lasagne_v2` (AMI ID: `ami-125b2c72`). Select that AMI, and continue to the next step to choose your instance type.



Choose the instance type `g2.2xlarge`, and click on “Review and Launch”.

EC2 Management Console - Step 2: Choose an Instance Type

Instance Type	Category	Instance Type	VCpus	Memory (GiB)	Storage	EBS-Optimized	Network Performance
<input type="checkbox"/>	Compute optimized	c3.xlarge	4	7.5	2 x 40 (SSD)	Yes	Moderate
<input type="checkbox"/>	Compute optimized	c3.2xlarge	8	15	2 x 80 (SSD)	Yes	High
<input type="checkbox"/>	Compute optimized	c3.4xlarge	16	30	2 x 160 (SSD)	Yes	High
<input type="checkbox"/>	Compute optimized	c3.8xlarge	32	60	2 x 320 (SSD)	-	10 Gigabit
<input checked="" type="checkbox"/>	GPU instances	g2.2xlarge	8	15	1 x 60 (SSD)	Yes	High
<input type="checkbox"/>	GPU instances	g2.8xlarge	32	60	2 x 120 (SSD)	-	10 Gigabit
<input type="checkbox"/>	Memory optimized	r3.large	2	15	1 x 32 (SSD)	-	Moderate
<input type="checkbox"/>	Memory optimized	r3.xlarge	4	30.5	1 x 80 (SSD)	Yes	Moderate
<input type="checkbox"/>	Memory optimized	r3.2xlarge	8	61	1 x 160 (SSD)	Yes	High
<input type="checkbox"/>	Memory optimized	r3.4xlarge	16	122	1 x 320 (SSD)	Yes	High
<input type="checkbox"/>	Memory optimized	r3.8xlarge	32	244	2 x 320 (SSD)	-	10 Gigabit
<input type="checkbox"/>	Storage optimized	i2.xlarge	4	30.5	1 x 800 (SSD)	Yes	Moderate
<input type="checkbox"/>	Storage optimized	i2.2xlarge	8	61	2 x 800 (SSD)	Yes	High
<input type="checkbox"/>	Storage optimized	i2.4xlarge	16	122	4 x 800 (SSD)	Yes	High

Buttons: Cancel, Previous, **Review and Launch**, Next: Configure Instance Details

In the next page, click on Launch.

EC2 Management Console - 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-1**, 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](#)

Your instance configuration is not eligible for the free usage tier
To launch an instance that's eligible for the free usage tier, check your AMI selection, instance type, configuration options, or storage devices. Learn more about [free usage tier](#) eligibility and usage restrictions. [Don't show me this again](#)

AMI Details [Edit AMI](#)

cs231n_caffe_torch7_keras_lasagne_v2 - ami-125b2c72
Image for CS231N. Contains Caffe, Torch7, Keras, Lasagne, CUDA v7.5, CUDNN v3.
Root Device Type: ebs Virtualization type: hvm

Instance Type [Edit instance type](#)

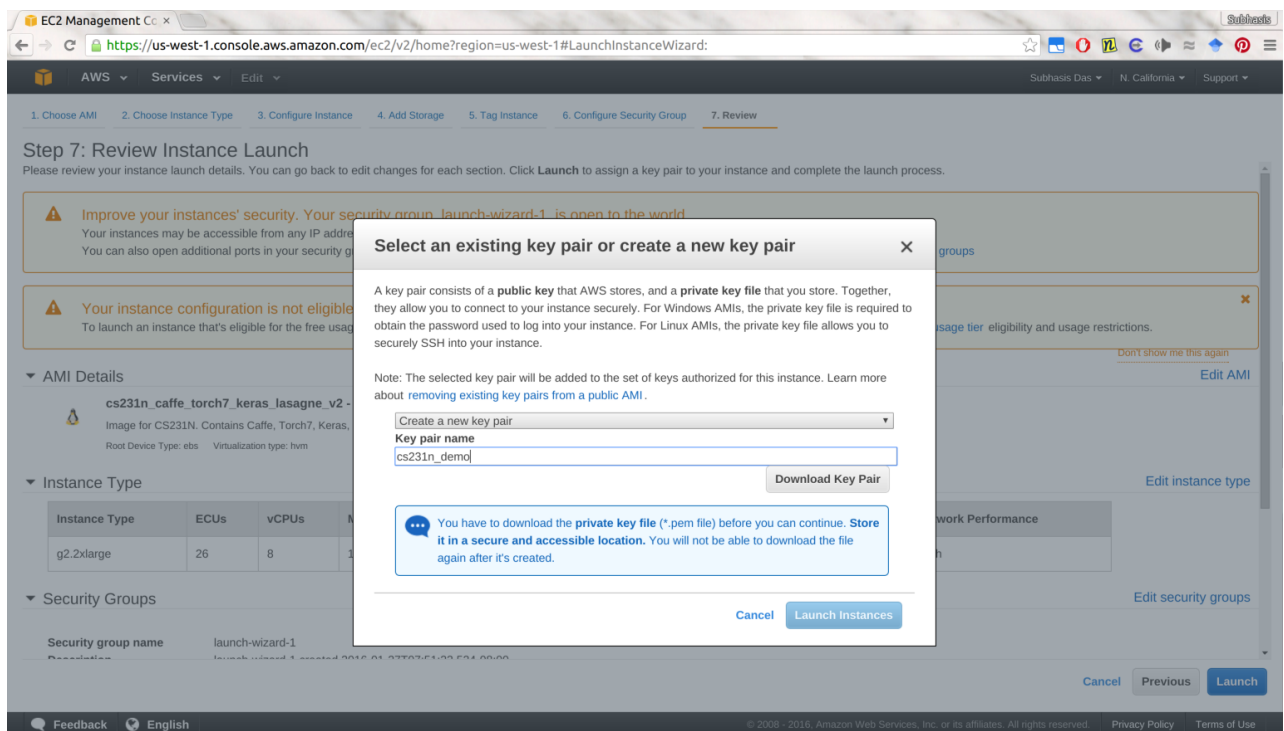
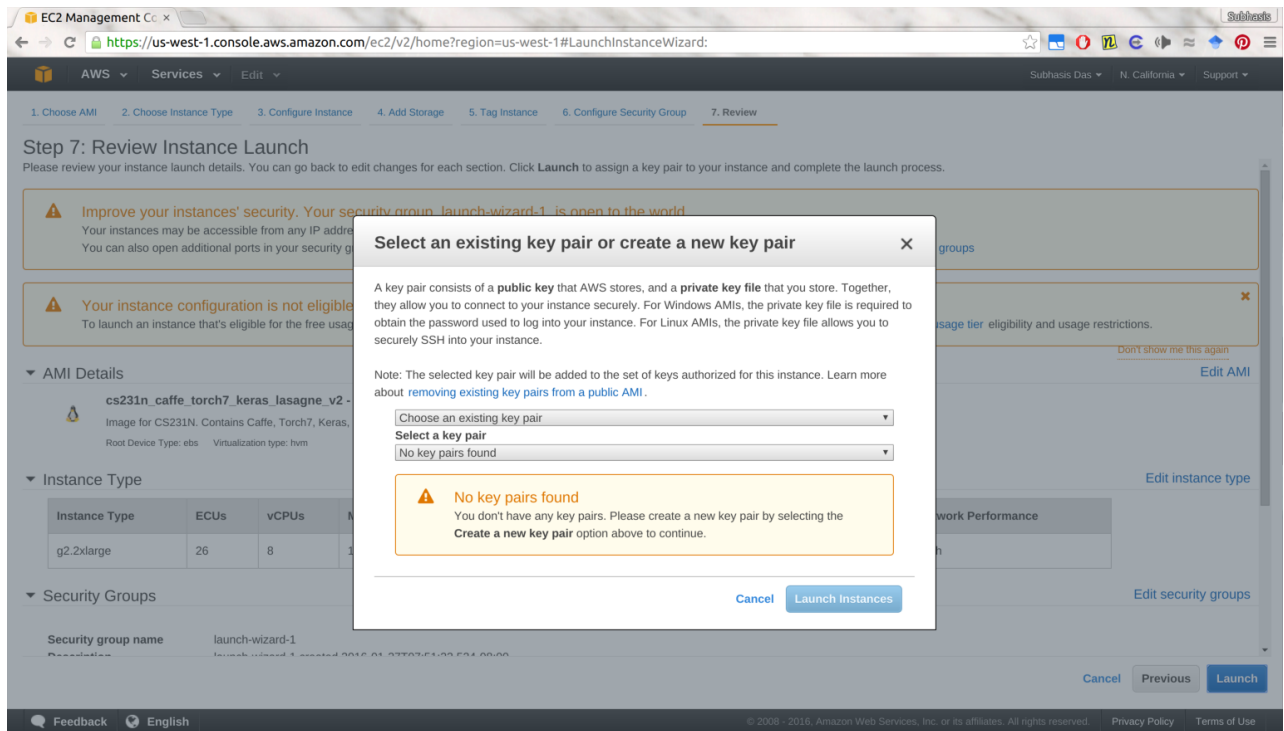
Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
g2.2xlarge	26	8	15	1 x 60	Yes	High

Security Groups [Edit security groups](#)

Security group name: launch-wizard-1

Buttons: Cancel, Previous, **Launch**

You will be then prompted to create or use an existing key-pair. If you already use AWS and have a key-pair, you can use that, or alternately you can create a new one by choosing "Create a new key pair" from the drop-down menu and giving it some name of your choice. You should then download the key pair, and keep it somewhere that you won't accidentally delete. Remember that there is **NO WAY** to get to your instance if you lose your key.

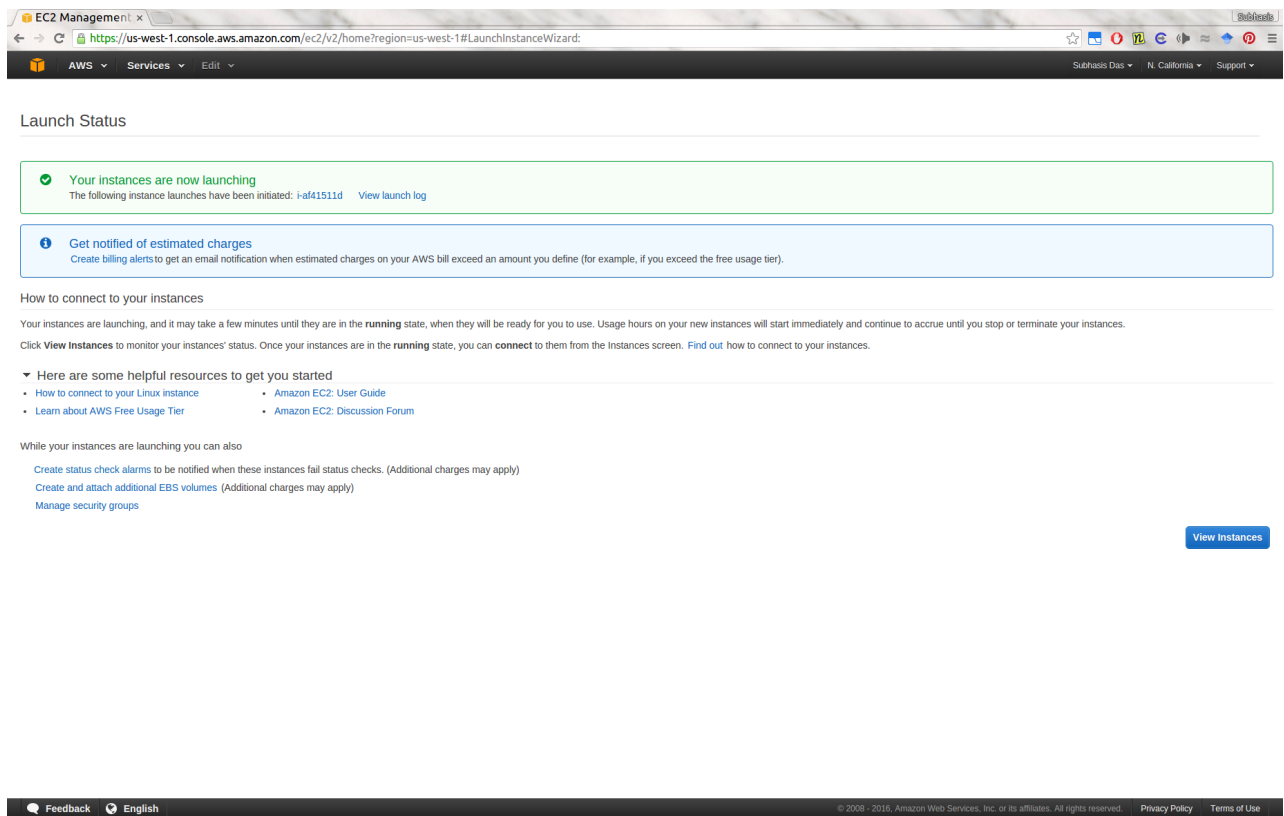


Once you download your key, you should change the permissions of the key to user-only RW, In Linux/OSX you can do it by:

```
$ chmod 600 PEM_FILENAME
```

Here `PEM_FILENAME` is the full file name of the .pem file you just downloaded.

After this is done, click on “Launch Instances”, and you should see a screen showing that your instances are launching:



Click on “View Instances” to see your instance state. It should change to “Running” and “2/2 status checks passed” as shown below within some time. You are now ready to ssh into the instance.

The screenshot shows the AWS Management Console interface. On the left is a navigation menu with categories like INSTANCES, IMAGES, ELASTIC BLOCK STORE, NETWORK & SECURITY, LOAD BALANCING, and AUTO SCALING. The main area displays a table of EC2 instances with columns for Name, Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, Public DNS, and Public IP. One instance, i-af41511d, is shown in the 'running' state. Below the table, a detailed view for instance i-af41511d is shown, including its Public DNS (ec2-54-183-60-80.us-west-1.compute.amazonaws.com) and various configuration details like Instance state, Instance type, Private DNS, Private IPs, Secondary private IPs, VPC ID, Subnet ID, Public DNS, Public IP, Elastic IP, Availability zone, Security groups, Scheduled events, AMI ID, and Platform.

First, note down the Public IP of the instance from the instance listing. Then, do:

```
ssh -i PEM_FILENAME ubuntu@PUBLIC_IP
```

Now you should be logged in to the instance. You can check that Caffe is working by doing:

```
$ cd caffe
$ ./build/tools/caffe time --gpu 0 --model examples/mnist/lenet.protot
```

We have Caffe, Theano, Torch7, Keras and Lasagne pre-installed. Caffe python bindings are also available by default. We have CUDA 7.5 and CuDNN v3 installed.

If you encounter any error such as

```
Check failed: error == cudaSuccess (77 vs. 0) an illegal memory acce
```

you might want to terminate your instance and start over again. I have observed this rarely, and I am not sure what causes this.

About how to use these instances:

- The root directory is only 12GB, and only ~ 3GB of that is free.
- There should be a 60GB `/mnt` directory that you can use to put your data, model checkpoints, models etc.
- Remember that the `/mnt` directory won't be persistent across reboots/terminations.
- Stop your instances when are done for the day to avoid incurring charges. GPU instances are costly. Use your funds wisely. Terminate them when you are sure you are done with your instance (disk storage also costs something, and can be significant if you have a large disk footprint).
- Look into creating custom alarms to automatically stop your instances when they are not doing anything.
- If you need access to a large dataset and don't want to download it every time you spin up an instance, the best way to go would be to create an AMI for that and attach that AMI to your machine when configuring your instance (before launching but after you have selected the AMI).

 [cs231n](#)

 [cs231n](#)

karpathy@cs.stanford.edu