

Virtual Machines

Presented by Dan Sanz
December 2019

 **FLATIRON SCHOOL**

Top up skills

- Setting up Virtual Machines
- Taking models to production



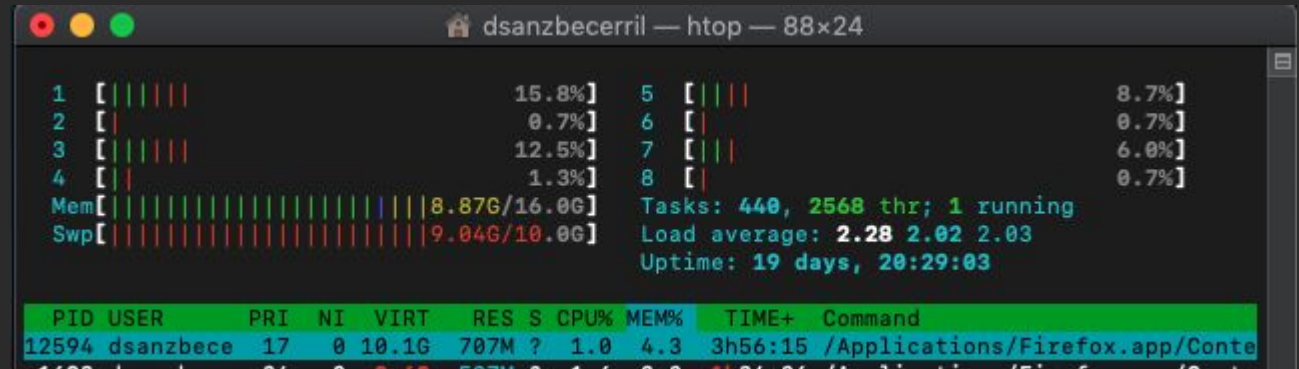
Setting up Virtual Machines

- Hardware
- Networking
- Software
- Your code and data
- Launch



Hardware

- Processor
- Memory
- Disk
- Swap



Processor

- If your learning process is parallelisable, more processors will allow you to split jobs across more threads and your training/optimisation will finish faster
- The number of threads will often make a bigger difference than the speed of the processor



Memory

- Your data needs to be in memory in order to be accessed by your learners
- More data, higher memory requirements
- Different models will have different requirements
- $\text{memory} = \text{data_size}^2$; I found it to be excessive
- Without big data tools this is usually what will make you pay

//

more

EC2 Instances

- Processor/s + Memory
- Minimum Viable Virtual Machine



Swap

- Memory use is not flat. You will see frequent spikes.
- You can use a Disk volume as ‘virtual memory’ to accommodate those spikes



Disk

- For persistence (what you want to use in more than one session)
- As low as \$0.025/GB-month



Networking & Security

- IP
- Port
- Signature
- SSH Tunneling



Information you should protect

- You will be working with data that you will need to protect:
 - Personal Identifiable Information (PII) and genetic data, biometric data processed solely to identify a human being
 - Sensitive information (personal data revealing racial or ethnic origin, political opinions, religious, philosophical beliefs, trade-union membership or data concerning a person's sex life or sexual orientation and health-related data)
 - Information that your business would prefer not to share with competitors



Security Groups

- List the IPs that will have access
 - Most companies use a Virtual Private Network (VPN) so just one IP will do for the whole company independently of location
 - <https://www.whatsmyip.org/>
- List the ports through which access will happen (ie. 8888-8890)



SSH keys

- The 'secret handshake' between computers
- You have used them already with github
- Create them in your mac with ssh-keygen
- You generate a public key that your remote will host
<https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ec2-key-pairs.html#how-to-generate-your-own-key-and-import-it-to-aws>
- And a private key that you will keep in your local `/.ssh/` (note:
// it's an invisible folder)

SSH Tunneling

- Accessing your virtual machine as if it was your local
- So that you can use jupyter notebooks GUI with supercharged computing power from your virtual machine!
- `ssh -i /path/to/your/private_key.pem`
`-L localport:remote_IP:remote_port`
`remote_username@remote_ip`



Software

- Operative System
- Anaconda Distribution



Anaconda

- `curl -O`

`https://repo.anaconda.com/archive/Anaconda3-2019.10-Linux-x86_64.sh`

- `bash Anaconda3-2019.10-Linux-x86_64.sh`



Your code & Data

- Git clone
- Large file transfer



Larger file transfer

- `rsync -ah --progress source-file destination-file`
- Will resume the transfer even after brief interruptions
- You will be able to see progress and speed



Launch jupyter

//

```
jupyter notebook  
--no-browser  
-port=8888
```

(in one line)

//

Resources

<https://medium.com/@GuruAtWork/setup-jupyter-notebook-on-aws-for-fast-ai-b6b407a719e9>

<https://medium.com/@GalarnykMichael/aws-ec2-part-2-ssh-into-ec2-instance-c7879d47b6b2>

<https://scottlinux.com/2017/04/28/how-to-add-a-second-disk-to-an-ec2-instance-in-aws/>

