GV918 - Week 05 Class

Akitaka Matsuo

1/11/2023

## Class Exercise

### Contents

1. Launch EC2 instances
2. Work with EC2
   * Python and Jupyter Notebook
   * Web-server
3. Work with S3
4. (Try RStudio Server)

### 0. Go to AWS console

1. Go to AWS Academy Login <https://awsacademy.instructure.com/login/canvas>
2. Go to AWS Academy Learner Lab
3. Go to Module -> “Learner Lab - Foundational Services”
4. Click Start Lab (and wait)
5. Open AWS console (Click AWS)

### 1. Create an EC2 instance

Now we create an EC2 instance

1. Go to EC2 dashboard
2. Select region (N. Virginia US-East-1)
3. Select “Launch Instance”
4. Configure the instance
   * AMI: Amazon Linux 2 AMI
   * Instance Type: t3.large
   * Key pair
     + vockey
   * Storage: 20GB
   * Tags:
     + Key: Name
     + Value: GV918-yourname
   * Security Group:
     + Name: GV918-secgroup
     + Description: GV918-secgroup
     + Add rule: HTTP (from your IP or anywhere)
     + Add rule: Custom, Port 8888 (from your IP or anywhere)
     + Modify rule: Source (for your IP or anywhere)
5. Click “Launch instance” (on the right)

### 2. Connect to your EC2 instance

#### Get pem file (optional)

#### Find address of EC2 instance

1. Go to EC2 console
2. Select “Instances” on the list
3. Check your instance state
4. Copy “Public DNS value” and paste it to somewhere
   * the value should look like: ec2-##-##-##-##.compute-1.amazonaws.com

#### Log in to the instance (from Learner lab page)

1. Go back to AWS academy page
2. Open Learner Lab
3. Log in your EC2 instance via ssh
   * ssh -i .ssh/labsuser.pem ec2-user@ec2-##-##-##-##.compute-1.amazonaws.com
   * say *yes* to everything
4. Your console before “$” now says [ec2-user@ip-\*\*-\*\*-\*\*-\*\*]

#### Log in to the instance (from **Terminal**)

1. Go back to AWS academy page
2. (Click AWS detail)
3. Click “Download PEM”
4. Open **Terminal** (or git bash for windows users)
5. Change working directory to the folder you have **\*.pem** file
   * cd /path/to/the/folder
   * in Mac, you can drag and drop the folder into the terminal window after typing cd.
6. Change the permission of **\*.pem** file
   * chmod 400 labsuser.pem
7. Log in your EC2 instance via ssh
   * ssh -i labsuser.pem ec2-user@ec2-##-##-##-##.compute-1.amazonaws.com
   * say *yes* to everything
8. Your console before “$” now says [ec2-user@ip-\*\*-\*\*-\*\*-\*\*]

### 3. Install Miniconda

Now you install python3 through miniconda installation

1. Update linux: sudo yum update
2. Download installation file
   * In the EC2 console, enter wget https://repo.anaconda.com/miniconda/Miniconda3-py38\_4.12.0-Linux-x86\_64.sh
   * bash Miniconda3-py38\_4.12.0-Linux-x86\_64.sh
   * for the licence screen, press q to proceed
   * say ‘yes’ to all
3. Type exec bash to refresh
4. Type which python
   * the output should be ~/miniconda3/bin/python

### 4. Setup conda (virtual) environment

1. Type conda create -n gv918
2. Type conda activate gv918

(To deactivate, type conda deactivate)

### 5. Install Jupyter notebook

1. Type conda install -c conda-forge jupyterlab
2. Type jupyter-lab --generate-config

### 6. Jupyter notebook setup

1. Type jupyter notebook password
2. Type vim .jupyter/jupyter\_lab\_config.py
   * hit a
   * type

* conf = get\_config()  
   conf.NotebookApp.ip = '0.0.0.0'  
   conf.NotebookApp.port = 8888
  + hit esc
  + hit :, w, q

### 7. Start jupyter notebook

1. Type mkdir my\_jupyter\_prj
2. Type cd my\_jupyter\_prj
3. Type jupyter-lab

### 8. Open jupyter notebook through browser

1. Visit ec2-##-##-##-##.compute-1.amazonaws.com:8888
2. Input the password you created above
3. Create a new notebook
4. Install packages
   * !conda install -c conda-forge numpy pandas matplotlib seaborn -y
5. Load packages and run some example codes

* import numpy as np  
   import pandas as pd  
   import matplotlib.pyplot as plt  
   import seaborn as sns  
   %matplotlib inline   
    
   # Apply the default theme  
   sns.set\_theme()  
    
   # Load an example dataset  
   tips = sns.load\_dataset("tips")  
    
   # Create a visualization  
   sns.relplot(  
   data=tips,  
   x="total\_bill", y="tip", col="time",  
   hue="smoker", style="smoker", size="size",  
   )  
   plt.savefig('test\_fig.png')

1. Upload/downlaod files

### 9. Misc

1. Finishing
2. Stop notebook: Ctrl-C
3. Exit gv918 conda env: conda deactivate
4. Run a long-term job, using screen function
   * screen -S py-long-run
   * conda activate gv918
   * cd ...
   * jupyter-lab
   * Exit screen Ctrl+A+D
   * To resume screen :  
     screen -r py-long-run

### 10. Using S3

We will upload an html file on S3 and directly serve the file from s3. This is a cost efficient way of hosting static contents. Or other files, you can do too.

1. On browser, go to AWS S3 console
2. Create a bucket
   * Name: GV918-\*\*\* (since bucket names are universal, we can’t give the same name to multiple buckets)
   * Region: London
   * Tags:
     + Key: Name
     + Value: GV918
3. Locally Create a simple html file Using your favorite text-editor, create a simple html file and save it as s3-test.html:   
   <html> <body>This is my first S3 webpage.</body> </html>
4. Upload the file
   * Select the bucket
   * Select Upload file
   * Drag and drop the file s3-test.html
   * Choose Grant public read access to this object
5. Check the file
   * Click the object just uploaded
   * In Overview find Link. Click the link and check the file is displayed correctly. url should look like <https://s3.eu-west-2.amazonaws.com/your-bucket-name/s3-test.html>
6. (Optional) Upload other stuffs

### 11. Set up a simple website with apache on EC2 (if we have time)

Here you install an apache server using yum and create a simple html file

1. Update packages: sudo yum update -y
2. Install apache
   * In the EC2 console, enter sudo yum install httpd
3. Start webserver: sudo service httpd start
4. Check if server is running.
   * enter public IP of your EC2 instance in your browser and see if a Test Page is shown
5. Create a simple html file to display
   * Enter cd /var/www/html
   * Enter sudo nano index.html
   * copy and paste following lines
   * <html>  
     <body>Hello this is my website</body>  
     </html>
   * save and exit nano
6. Check whether the page is displayed by refreshing the browser
7. Publish the figure from the previous section
   * Copy the figure into the public folder using cp
   * sudo cp ~/test\_fig.png /var/www/html
   * Check on the browser. url: http://ec2-##-##-##-##.compute-1.amazonaws.com/test\_fig.png

### 12. Install R and RStudio Server

1. Edit security group, open port 8787
2. Log in your EC2 instance via ssh
   * ssh -i labsuser.pem ec2-user@ec2-##-##-##-##.compute-1.amazonaws.com
3. Update linux: sudo yum update
4. Install R: sudo amazon-linux-extras install R4
5. Install Rstudio:
   * wget https://download2.rstudio.org/server/centos7/x86\_64/rstudio-server-rhel-2022.07.2-576-x86\_64.rpm
   * sudo yum install rstudio-server-rhel-2022.07.2-576-x86\_64.rpm
6. Add user:
   * sudo adduser <username>
   * sudo passwd <username>
7. Access with: ec2-##-##-##-##.compute-1.amazonaws.com:8787