Data Engineering Basics

Lacey Conrad
MSDS 610
Regis University

May, 2020

1 Introduction

The rise of Big Data has resulted in the development of methods to deal with the 5Vs associated with it: volume, velocity, variety, veracity, and value (Yang et al., 2016). Data Science, on one hand, deals with modeling and analyzing Big Data, whereas Data Engineering primarily focuses on developing and maintaining the data infrastructure (*state of data engineering* — Stitch benchmark report, 2019). Given the massive datasets data engineers are beginning to work with, they began to look for new ways to warehouse, process, and maintain the architecture of their data. Data engineers have started turning to cloud services for most of their needs, which has eliminated the need to maintain local servers infrastructure and software (Castrounis, 2017; Hufford, 2018).

The written portion of this assignment sought to provide foundational knowledge on the topic of data engineering. The technical portion of this exercise involved setting up a Google Cloud Platform account, including updating it, and then practicing Linux commands. After I had refreshed my Linux command knowledge, I followed the lab supplement and installed Docker on the Virtual Machine I had created in the previous step. Lastly, I performed a word count analysis on one of the works of Shakespeare, which is considered the ”Hello World” analog for Hadoop.

2 Written Portion - Questions from week 1

1. What is Linux, and why is it important for data engineers and data scientists?

   Linux is an open source operating system, which means it relays instructions between programs and the hardware of a computer (”What is Linux?,” 2019). Linux-based operating systems include a Linux kernel (which manages hardware resources) and software packages. Additional components are frequently included to aid in resource allocation, security, software installation, and more (”Understanding Linux,” 2020).

   Linux has several characteristics that make it one of the most popular operating systems:

   (a) It is open source and free. One of its most popular features is its establishment as an open-source operating system which allows anyone to view the code and edit it. Also, this allows Linux to have superb support, since many people are knowledgeable of the operating system and have access to the code (”What is Linux?,” 2019).

   (b) It is highly customizable. Linux has many distributions, allowing it to be highly customizable for the user and their project(s). Common distributions include Elementary, Mint, Fedora and Ubuntu (”What is Linux?,” 2019).

   (c) It is secure. This is partly due to its handling of permissions. Without proper permissions, you cannot execute a .exe file. Also, since it is open source, there are many people capable of detecting places where the code could be more resilient to viruses (”What is Linux?,” 2019).

   (d) Similarly, it is reliable. Server reboots, for example, only occur when a kernel is updated, which in some cases may be once every year or more (”What is Linux?,” 2019).

Linux is vitally important to data scientists and engineers due to its computing power. The data handled by data scientists is so large that it becomes difficult to handle, and the computing speed offered by Linux makes it an ideal platform to use compared to other options. Additionally, the use of Docker makes Linux an ideal operating system for a data scientist. Docker, which enables programs to run simultaneously without interfering with each other, runs at its fastest speeds on a GPU. In many cases, the GPU based Docker containers will only run on Linux operating systems. Several of the characteristics listed above hugely benefit the data scientist when using Linux, but of central importance is the ability of Linux to handle the large amounts of data to be processed or analyzed by a data scientists and engineers. This increase in ability to handle Big Data is in part due to Linux's resource management (Misal, 2019).

2. Who are the main cloud services providers, and what sort of things do they provide?

   Cloud computing enables users access to software whenever and wherever they are, assuming they have an internet connection. This is a great improvement over the past need to set-up and maintain physical servers and manage software. Cloud computing services are provided by a 3rd-party which hosts a network of remote servers that store and process data. In addition to these services, cloud service providers also manage the maintenance, performance and security of their servers (Hufford, 2018). The three main models of cloud services are as follows (although there are several other cloud services, they are not as frequently encountered):

   (a) Software as a service (SaaS): This is a licensing and delivery model where software (or an application) is made available to a user via a web browser. The software lives and runs on servers within the cloud and does not need to be installed locally (Hufford, 2018).

   (b) Platform as a service (PaaS): In PaaS, a cloud service provider builds and maintains software development infrastructure so enterprises can create and manage applications without the need of interacting with the underlying infrastructure (Violino, 2019).

   (c) Infrastructure as a service (IaaS): Here, the components that typically comprise a physical data center are hosted and managed by a cloud provider. IaaS services can include servers, storage, and network hardware in addition to the services needed to utilize, manage, and protect these resources (Rouse, 2018).

   According to Hufford (2018), the following three cloud service providers are likely to account for roughly 80% of all cloud revenue in 2020:

   (a) Amazon Web Services (AWS): AWS has a substantial tool kit with a focus on public cloud computing.

   (b) Microsoft Azure: Microsoft Azure is the go-to cloud service for enterprises. It uses a hybrid cloud, allowing Azure to work with enterprise data centers. Microsoft Azure is particularly noted for their SaaS products such as Microsoft 365.

   (c) Google Cloud: Google Cloud is the best option for technical cloud services pertaining to deep learning, artificial intelligence, machine learning, and data analytics (Harvey & Patrizio, 2020).

3. What can we do with the cloud providers related to data engineering?

   Data engineering is moving at a very rapid rate to the cloud. Frequently, the datasets Data Engineers manage are simply too big to be housed on a local machine, and similarly, many these huge datasets require too much processing power to perform tasks in a reasonable amount of time (Castrounis, 2017). Cloud services can be used to warehouse huge datasets cheaply and can keep them in any format or size they choose. The data can be organized, merged (or separated), and analyzed in the cloud without any pressure on the local development machine. Cloud providers will also provide the support to manage the hardware, maintenance, server performance, and so on (Hufford, 2018).

   Many cloud providers also promise elasticity and flexibility to data processing infrastructure. While this shows how important it is for data engineers to keep up-to-date with data engineering technology, it also shows how the cloud providers are promising to the data engineers that they (the cloud providers) will provide cutting edge data processing technology ("Data engineering and science in the cloud," 2019).

4. What is Docker, how is it useful, and why is it important to know?

   In a way, Docker can be thought of as a lightweight virtual machine (Husain, 2018). Docker is a tool that uses containers to allow easier creation, deployment, and execution of applications. Docker sought to reduce the reliance on system resources previously encountered in virtual machine hypervisors by utilizing containers, which are much more efficient and secure (Vaughan-Nichols, 2018). Containers are portable units of software that can be used reliably across different computing environments. These lightweight units are packages of code, its dependencies, run-time, tools, libraries and settings. Containers include everything needed to execute an application, and in essence they have their own file system, storage, CPU, RAM, etc (”What is a container?,” 2020; Vaughan-Nichols, 2018). Additionally, since containers use a shared operating system they place a lot less stress on system resources, reducing application size and increasing performance (”What is docker?,” 2019; Vaughan-Nichols, 2018). This allows for four to six times the number of application instances running at once. Docker does require containers to use the same operating system and kernel, which is not the case with other options like Microsoft Azure (Vaughan-Nichols, 2018).

   Docker provides Data Scientists with multiple benefits:

   (a) Makes the work of data scientists and engineers easily reproducible. Using Docker containers allows you to wrap up your computing environment, making it easier for others to reproduce your work.

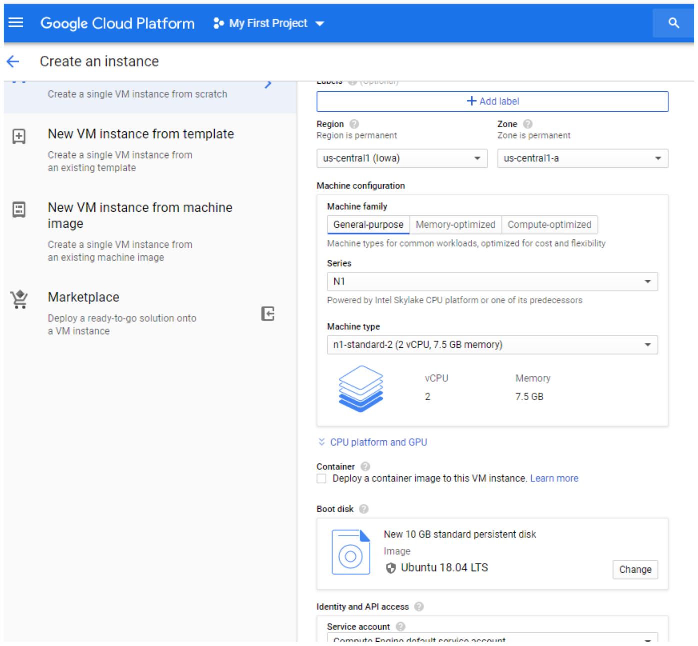
   (b) Allows the data scientist to have a more portable computing environment. This allows data scientists to access the computing environment they need without worrying about having to re-create a local environment every time they change environments.

   (c) Makes is easy to deploy applications, which increases accessibility to a data scientists work. This also allows for reproducibility and quick replication of research (Husain, 2018).

3 Methods and Code

To begin this lab, I created an account with Google Cloud Platform (GCP), and then set up a budget. After I set up my account, I was able to spin up my first GCP VM instance using the settings provided in the lab write-up:

3.1 Linux practice and Setting up Google Cloud Platform



Create a single VM instance from scratch

New VM instance from template

Create a single VM instance from an existing template

New VM instance from machine image

Create a single VM instance from an existing machine image

Marketplace

Deploy a ready-to-go solution onto a VM instance

Region: us-central1 (Iowa) — Region is permanent

Zone: us-central1-a — Zone is permanent

Machine configuration

Machine family: General-purpose | Memory-optimized | Compute-optimized

Machine types for common workloads, optimized for cost and flexibility

Series: N1

Powered by Intel Skylake CPU platform or one of its predecessors

Machine type: n1-standard-2 (2 vCPU, 7.5 GB memory)

vCPU	Memory
2	7.5 GB

Container

□ Deploy a container image to this VM instance.

Boot disk

New 10 GB standard persistent disk

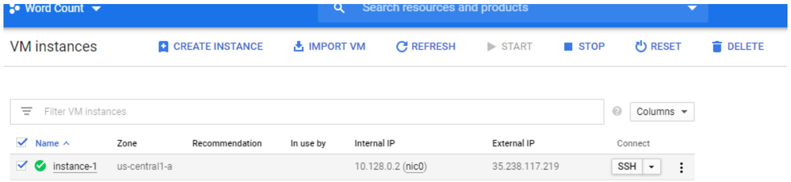
Image

Ubuntu 18.04 LTS

Identity and API access

Service account

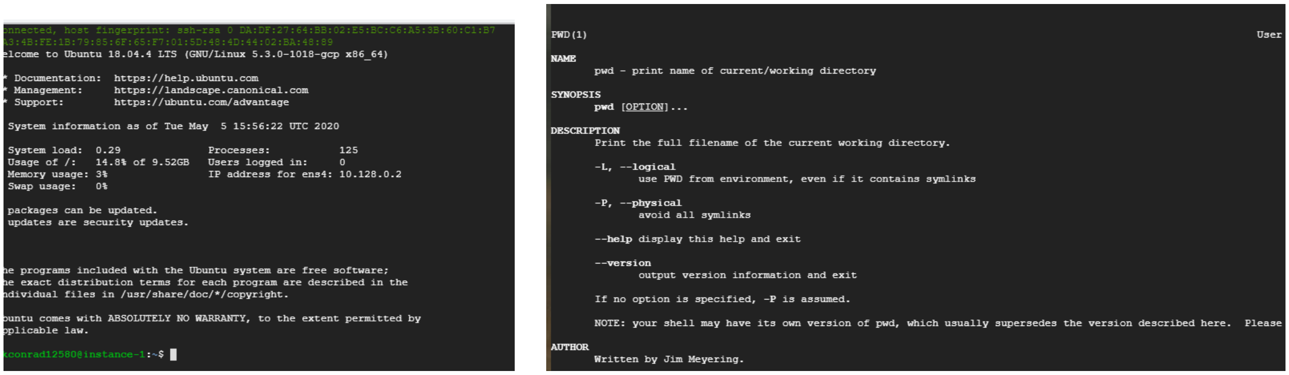
Figure 1: Setting up a VM instance in Google Cloud Platform.



☑	Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
☑	✓ instance-1	us-central1-a			10.128.0.2 (nic0)	35.238.117.219	SSH

Figure 2: All created VM instances in GCP showing the instance I just created.

Clicking on the SSH button under Connect allowed me to open a terminal window. Here, I practiced several Linux commands (although I have a decent amount of experience in Linux, so I didn't spend much time here). Below I took a screen capture of the open terminal in addition to one of the Linux commands:

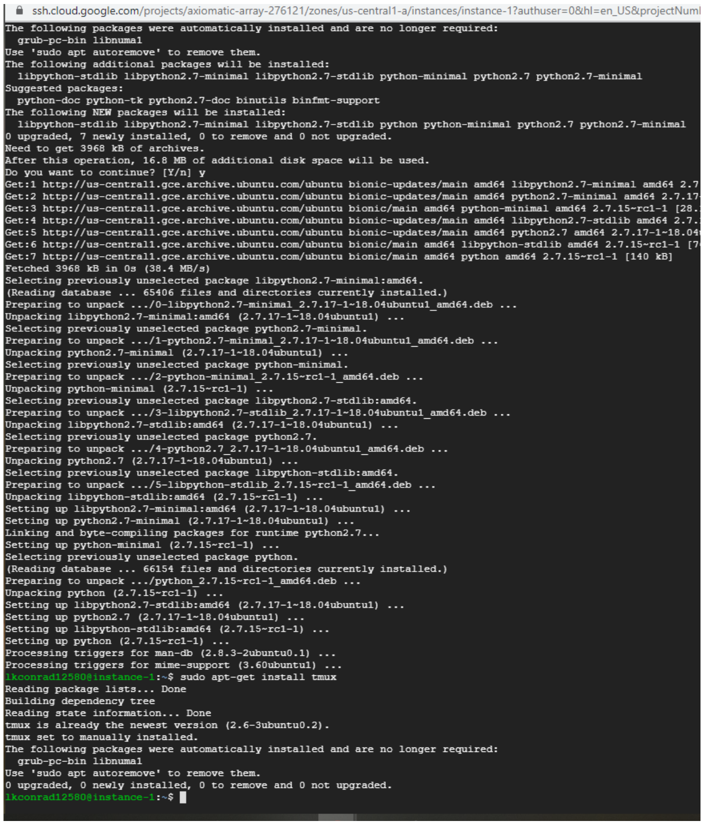


```
onnected, host fingerprint: ssh-rsa 0 DA:DF:27:64:BB:02:E5:BC:C6:A5:3B:60:C1:B7
A3:4B:FE:1B:79:85:6F:65:F7:01:5D:48:4D:44:02:BA:48:89
elcome to Ubuntu 18.04.4 LTS (GNU/Linux 5.3.0-1018-gcp x86_64)

* Documentation:  https://help.ubuntu.com
* Management:     https://landscape.canonical.com
* Support:        https://ubuntu.com/advantage

 System information as of Tue May  5 15:56:22 UTC 2020

 System load:  0.29               Processes:           125
 Usage of /:   14.8% of 9.52GB   Users logged in:     0
 Memory usage: 3%                 IP address for ens4: 10.128.0.2
 Swap usage:   0%

 packages can be updated.
 updates are security updates.



he programs included with the Ubuntu system are free software;
he exact distribution terms for each program are described in the
ndividual files in /usr/share/doc/*/copyright.

buntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
pplicable law.

kconrad12580@instance-1:~$ 
```

```
PWD(1)                                                                                              User

NAME
       pwd - print name of current/working directory

SYNOPSIS
       pwd [OPTION]...

DESCRIPTION
       Print the full filename of the current working directory.

       -L, --logical
              use PWD from environment, even if it contains symlinks

       -P, --physical
              avoid all symlinks

       --help display this help and exit

       --version
              output version information and exit

       If no option is specified, -P is assumed.

       NOTE: your shell may have its own version of pwd, which usually supersedes the version described here.  Please

AUTHOR
       Written by Jim Meyering.
```

Figure 3: Left: After clicking on SSH in the figure above, a terminal window opens up. Right: A screen capture of Linux command practice, in this case `mad pwd`.

I then installed Python on my virtual machine, and checked to see if tmux is installed:

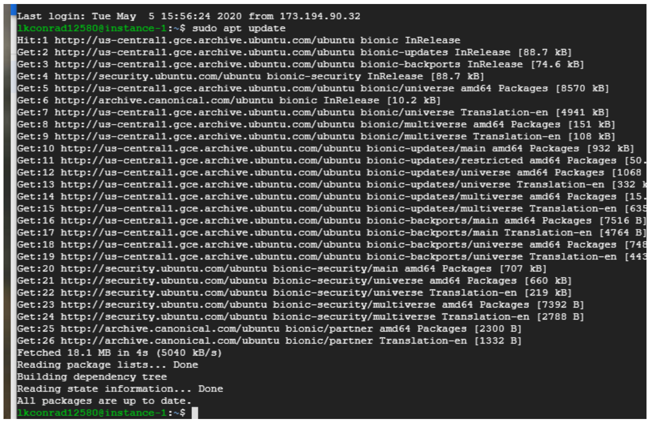


ssh.cloud.google.com/projects/axiomatic-array-276121/zones/us-central1-a/instances/instance-1?authuser=0&hl=en_US&projectNuml

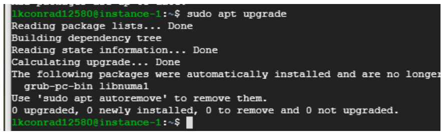
```
The following packages were automatically installed and are no longer required:
  grub-pc-bin libnuma1
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libpython-stdlib libpython2.7-minimal libpython2.7-stdlib python-minimal python2.7 python2.7-minimal
Suggested packages:
  python-doc python-tk python2.7-doc binutils binfmt-support
The following NEW packages will be installed:
  libpython-stdlib libpython2.7-minimal libpython2.7-stdlib python python-minimal python2.7 python2.7-minimal
0 upgraded, 7 newly installed, 0 to remove and 0 not upgraded.
Need to get 3968 kB of archives.
After this operation, 16.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates/main amd64 libpython2.7-minimal amd64 2.7
Get:2 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates/main amd64 python2.7-minimal amd64 2.7.17
Get:3 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic/main amd64 python-minimal amd64 2.7.15~rc1-1 [28.
Get:4 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates/main amd64 libpython2.7-stdlib amd64 2.7.
Get:5 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates/main amd64 python2.7 amd64 2.7.17-1~18.04
Get:6 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic/main amd64 libpython-stdlib amd64 2.7.15~rc1-1 [7
Get:7 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic/main amd64 python amd64 2.7.15~rc1-1 [140 kB]
Fetched 3968 kB in 0s (38.4 MB/s)
Selecting previously unselected package libpython2.7-minimal:amd64.
(Reading database ... 65406 files and directories currently installed.)
Preparing to unpack .../0-libpython2.7-minimal_2.7.17-1~18.04ubuntu1_amd64.deb ...
Unpacking libpython2.7-minimal:amd64 (2.7.17-1~18.04ubuntu1) ...
Selecting previously unselected package python2.7-minimal.
Preparing to unpack .../1-python2.7-minimal_2.7.17-1~18.04ubuntu1_amd64.deb ...
Unpacking python2.7-minimal (2.7.17-1~18.04ubuntu1) ...
Selecting previously unselected package python-minimal.
Preparing to unpack .../2-python-minimal_2.7.15~rc1-1_amd64.deb ...
Unpacking python-minimal (2.7.15~rc1-1) ...
Selecting previously unselected package libpython2.7-stdlib:amd64.
Preparing to unpack .../3-libpython2.7-stdlib_2.7.17-1~18.04ubuntu1_amd64.deb ...
Unpacking libpython2.7-stdlib:amd64 (2.7.17-1~18.04ubuntu1) ...
Selecting previously unselected package python2.7.
Preparing to unpack .../4-python2.7_2.7.17-1~18.04ubuntu1_amd64.deb ...
Unpacking python2.7 (2.7.17-1~18.04ubuntu1) ...
Selecting previously unselected package libpython-stdlib:amd64.
Preparing to unpack .../5-libpython-stdlib_2.7.15~rc1-1_amd64.deb ...
Unpacking libpython-stdlib:amd64 (2.7.15~rc1-1) ...
Setting up libpython2.7-minimal:amd64 (2.7.17-1~18.04ubuntu1) ...
Setting up python2.7-minimal (2.7.17-1~18.04ubuntu1) ...
Linking and byte-compiling packages for runtime python2.7...
Setting up python-minimal (2.7.15~rc1-1) ...
Selecting previously unselected package python.
(Reading database ... 66154 files and directories currently installed.)
Preparing to unpack .../python_2.7.15~rc1-1_amd64.deb ...
Unpacking python (2.7.15~rc1-1) ...
Setting up libpython2.7-stdlib:amd64 (2.7.17-1~18.04ubuntu1) ...
Setting up python2.7 (2.7.17-1~18.04ubuntu1) ...
Setting up libpython-stdlib:amd64 (2.7.15~rc1-1) ...
Setting up python (2.7.15~rc1-1) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for mime-support (3.60ubuntu1) ...
lkconrad12580@instance-1:~$ sudo apt-get install tmux
Reading package lists... Done
Building dependency tree
Reading state information... Done
tmux is already the newest version (2.6-3ubuntu0.2).
tmux set to manually installed.
The following packages were automatically installed and are no longer required:
  grub-pc-bin libnuma1
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
lkconrad12580@instance-1:~$ 
```

Figure 4: Installation of Python and Tmux, showing that Tmux is already installed.

I used `update` to update the current software, and `upgrade` to update the software to its most recent version:



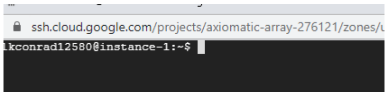
```
Last login: Tue May  5 15:56:24 2020 from 173.194.90.32
lkconrad12580@instance-1:~$ sudo apt update
Hit:1 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:3 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:4 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:5 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [8570 kB]
Get:6 http://archive.canonical.com/ubuntu bionic InRelease [10.2 kB]
Get:7 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic/universe Translation-en [4941 kB]
Get:8 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic/multiverse amd64 Packages [151 kB]
Get:9 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic/multiverse Translation-en [108 kB]
Get:10 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [932 kB]
Get:11 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates/restricted amd64 Packages [50.
Get:12 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [1068
Get:13 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates/universe Translation-en [332 k
Get:14 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates/multiverse amd64 Packages [15.
Get:15 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates/multiverse Translation-en [635
Get:16 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-backports/main amd64 Packages [7516 B]
Get:17 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-backports/main Translation-en [4764 B]
Get:18 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-backports/universe amd64 Packages [748
Get:19 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-backports/universe Translation-en [443
Get:20 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [707 kB]
Get:21 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [660 kB]
Get:22 http://security.ubuntu.com/ubuntu bionic-security/universe Translation-en [219 kB]
Get:23 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 Packages [7392 B]
Get:24 http://security.ubuntu.com/ubuntu bionic-security/multiverse Translation-en [2788 B]
Get:25 http://archive.canonical.com/ubuntu bionic/partner amd64 Packages [2300 B]
Get:26 http://archive.canonical.com/ubuntu bionic/partner Translation-en [1332 B]
Fetched 18.1 MB in 4s (5040 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
lkconrad12580@instance-1:~$
```



```
lkconrad12580@instance-1:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer
  grub-pc-bin libnuma1
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
lkconrad12580@instance-1:~$
```

Figure 5: Left: Updating the software in Ubuntu. Right: Upgrading the current software in Ubuntu.

Once the software was upgraded, I practiced creating a tmux session and attaching and detaching from it:



ssh.cloud.google.com/projects/axiomatic-array-276121/zones/u

```
lkconrad12580@instance-1:~$
```

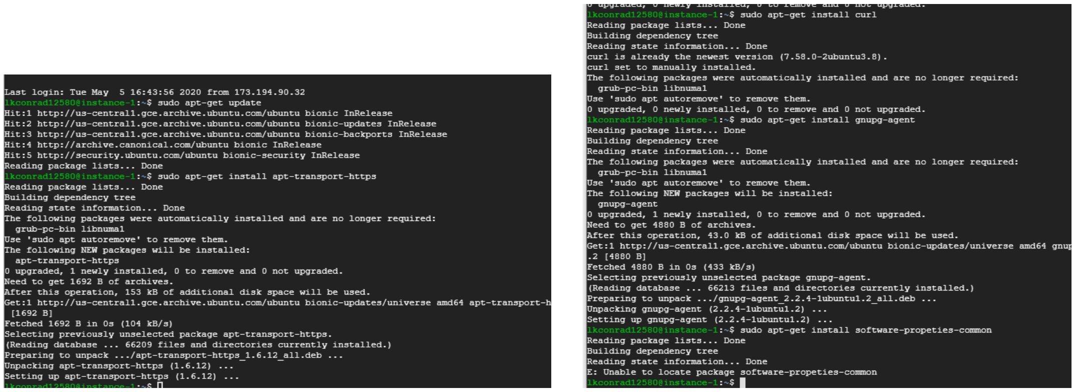
Figure 6: A tmux session window.

3.2 Installing and Running docker on Ubuntu

Since Docker will be an important component of this lab and class, it needed to be installed on my virtual machine. I used `sudo apt-get` to install the packages outlined in the lab document. I have lightly commented the code below to reflect its purpose whenever possible.

```
# The following is the code required to install Docker in Ubuntu.  It consists
# of the necessary packages to set up a Docker repository:

# Upgrading software to its current version:
$ sudo apt-get update
$ sudo apt-get upgrade

# Setting up the Docker repository:
$ sudo apt-get install apt-transport-https
$ sudo apt-get install ca-certificates
$ sudo apt-get install curl
$ sudo apt-get install gnupg-agent
$ sudo apt-get install software-properties-common

# Adding Docker's GPG key:
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -

# Verifying the key's fingerprint:
$ sudo apt-key fingerprint 0EBFCD88

# Setting up the stable repository:
$ sudo add-apt-repository \
        "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
        $(lsb_release -cs) \
        stable"

```

```
# Updating the newly installed software:
$ sudo apt-get update

# Installing several Docker containers:
$ sudo apt-get install docker-ce docker-ce-cli containerd.io
```

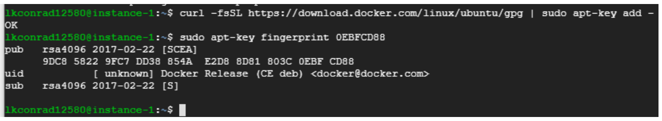


```
Last login: Tue May  5 16:43:56 2020 from 173.194.90.32
lkconrad12580@instance-1:~$ sudo apt-get update
Hit:1 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://archive.canonical.com/ubuntu bionic InRelease
Hit:5 http://security.ubuntu.com/ubuntu bionic-security InRelease
Reading package lists... Done
lkconrad12580@instance-1:~$ sudo apt-get install apt-transport-https
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  grub-pc-bin libnuma1
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 1692 B of archives.
After this operation, 153 kB of additional disk space will be used.
Get:1 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 apt-transport-h
 [1692 B]
Fetched 1692 B in 0s (104 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 66209 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_1.6.12_all.deb ...
Unpacking apt-transport-https (1.6.12) ...
Setting up apt-transport-https (1.6.12) ...
lkconrad12580@instance-1:~$
```

```
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
lkconrad12580@instance-1:~$ sudo apt-get install curl
Reading package lists... Done
Building dependency tree
Reading state information... Done
curl is already the newest version (7.58.0-2ubuntu3.8).
curl set to manually installed.
The following packages were automatically installed and are no longer required:
  grub-pc-bin libnuma1
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
lkconrad12580@instance-1:~$ sudo apt-get install gnupg-agent
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  grub-pc-bin libnuma1
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  gnupg-agent
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 4880 B of archives.
After this operation, 43.0 kB of additional disk space will be used.
Get:1 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 gnup
.2 [4880 B]
Fetched 4880 B in 0s (433 kB/s)
Selecting previously unselected package gnupg-agent.
(Reading database ... 66213 files and directories currently installed.)
Preparing to unpack .../gnupg-agent_2.2.4-1ubuntu1.2_all.deb ...
Unpacking gnupg-agent (2.2.4-1ubuntu1.2) ...
Setting up gnupg-agent (2.2.4-1ubuntu1.2) ...
lkconrad12580@instance-1:~$ sudo apt-get install software-propeties-common
Reading package lists... Done
Building dependency tree
Reading state information... Done
E: Unable to locate package software-propeties-common
lkconrad12580@instance-1:~$
```

Figure 7: Installing Docker.

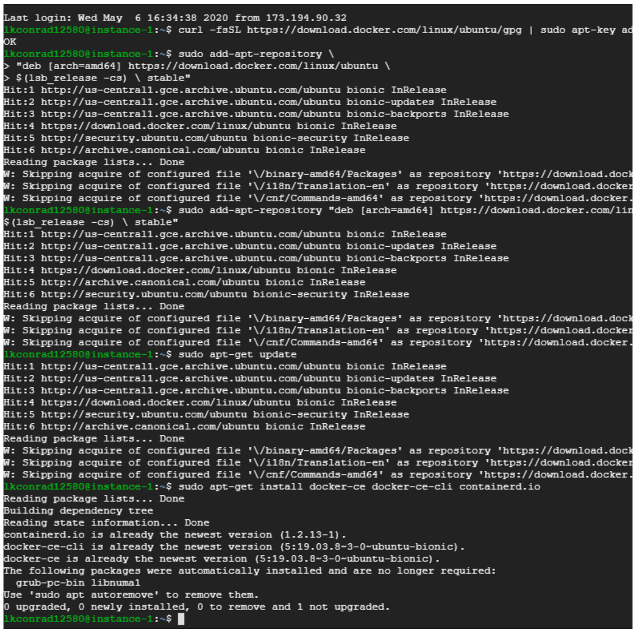
Here, I am verifying the fingerprint of the key:



```
lkconrad12580@instance-1:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
OK
lkconrad12580@instance-1:~$ sudo apt-key fingerprint 0EBFCD88
pub   rsa4096 2017-02-22 [SCEA]
      9DC8 5822 9FC7 DD38 854A  E2D8 8D81 803C 0EBF CD88
uid           [ unknown] Docker Release (CE deb) <docker@docker.com>
sub   rsa4096 2017-02-22 [S]

lkconrad12580@instance-1:~$
```

Figure 8: Verification of key with fingerprint `9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88`.

The following shows my establishment of the Docker repository. I also updated the software/packages I had installed, and lastly, installed the Docker containers:

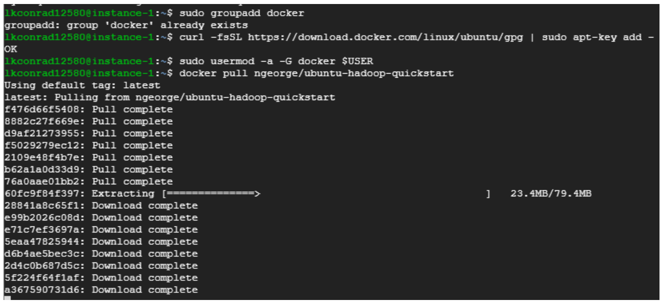


```
Last login: Wed May  6 16:34:38 2020 from 173.194.90.32
lkconrad12580@instance-1:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key ad
OK
lkconrad12580@instance-1:~$ sudo add-apt-repository \
> "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
> $(lsb_release -cs) \ stable"
Hit:1 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu bionic InRelease
Hit:5 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:6 http://archive.canonical.com/ubuntu bionic InRelease
Reading package lists... Done
W: Skipping acquire of configured file '\/binary-amd64/Packages' as repository 'https://download.dock
W: Skipping acquire of configured file '\/i18n/Translation-en' as repository 'https://download.docker
W: Skipping acquire of configured file '\/cnf/Commands-amd64' as repository 'https://download.docker.
lkconrad12580@instance-1:~$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/lin
$(lsb_release -cs) \ stable"
Hit:1 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu bionic InRelease
Hit:5 http://archive.canonical.com/ubuntu bionic InRelease
Hit:6 http://security.ubuntu.com/ubuntu bionic-security InRelease
Reading package lists... Done
W: Skipping acquire of configured file '\/binary-amd64/Packages' as repository 'https://download.dock
W: Skipping acquire of configured file '\/i18n/Translation-en' as repository 'https://download.docker
W: Skipping acquire of configured file '\/cnf/Commands-amd64' as repository 'https://download.docker.
lkconrad12580@instance-1:~$ sudo apt-get update
Hit:1 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu bionic InRelease
Hit:5 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:6 http://archive.canonical.com/ubuntu bionic InRelease
Reading package lists... Done
W: Skipping acquire of configured file '\/binary-amd64/Packages' as repository 'https://download.dock
W: Skipping acquire of configured file '\/i18n/Translation-en' as repository 'https://download.docker
W: Skipping acquire of configured file '\/cnf/Commands-amd64' as repository 'https://download.docker.
lkconrad12580@instance-1:~$ sudo apt-get install docker-ce docker-ce-cli containerd.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
containerd.io is already the newest version (1.2.13-1).
docker-ce-cli is already the newest version (5:19.03.8~3-0~ubuntu-bionic).
docker-ce is already the newest version (5:19.03.8~3-0~ubuntu-bionic).
The following packages were automatically installed and are no longer required:
  grub-pc-bin libnuma1
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
lkconrad12580@instance-1:~$
```

Figure 9: Setting up the stable Docker repository and updating the software.

My ID needed to be attached to Docker, which is shown in the following code. Also, I used a Docker `pull` to download the container for cloudera quickstart, and then ran the docker container:

```
# Added my user to the Docker group:
$ sudo usermod -a -G docker $USER

# Pulling/downloading the Docker container for Cloudera quickstart:
$ docker pull ngeorge/ubuntu-hadoop-quickstart

# Running the Docker container:
$ docker run -it ngeorge/ubuntu-hadoop-quickstart
```



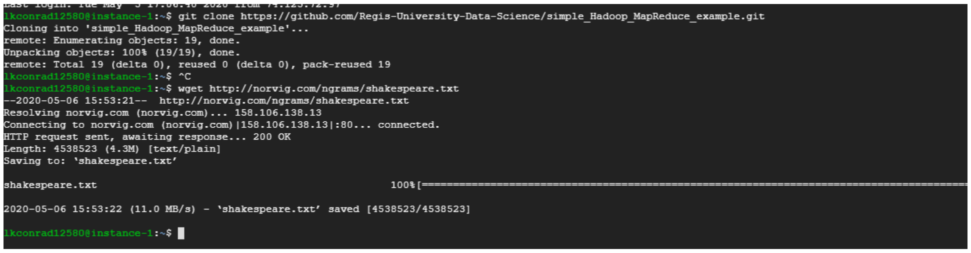
```
lkconrad12580@instance-1:~$ sudo groupadd docker
groupadd: group 'docker' already exists
lkconrad12580@instance-1:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
OK
lkconrad12580@instance-1:~$ sudo usermod -a -G docker $USER
lkconrad12580@instance-1:~$ docker pull ngeorge/ubuntu-hadoop-quickstart
Using default tag: latest
latest: Pulling from ngeorge/ubuntu-hadoop-quickstart
f476d66f5408: Pull complete
8882c27f669e: Pull complete
d9af21273955: Pull complete
f5029279ec12: Pull complete
2109e48f4b7e: Pull complete
b62a1a0d33d9: Pull complete
76a0aae01bb2: Pull complete
60fc9f84f397: Extracting [==============>                                    ]   23.4MB/79.4MB
28841a8c65f1: Download complete
e99b2026c08d: Download complete
e71c7ef3697a: Download complete
5eaa47825944: Download complete
d6b4ae5bec3c: Download complete
2d4c0b687d5c: Download complete
5f224f64f1af: Download complete
a367590731d6: Download complete
```

Figure 10: Downloading the Docker container for cloudera quickstart.

3.3 Word count example

Now that I have a running Docker container, I can practice using Hadoop. I will do this by running MapReduce on a text file as described in the lab document. First off, I needed to obtain the code for the MapReduce from git hub by using `git clone`. Then, I downloaded a text of Shakespeare's work.

```
# Cloning the git hub repo where the MapReduce code is located:
$ git clone https://github.com/Regis-University-Data-Science/
        simple_Hadoop_MapReduce_example.git

# Downloading the Shakespeare text:
$ wget http://norvig.com/ngrams/shakespeare.txt
```

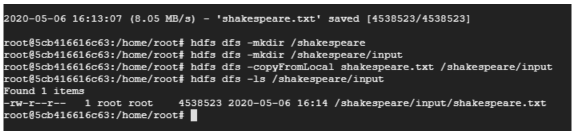


```
Last login: Tue May  5 17:06:40 2020 from 74.125.72.97
lkconrad12580@instance-1:~$ git clone https://github.com/Regis-University-Data-Science/simple_Hadoop_MapReduce_example.git
Cloning into 'simple_Hadoop_MapReduce_example'...
remote: Enumerating objects: 19, done.
Unpacking objects: 100% (19/19), done.
remote: Total 19 (delta 0), reused 0 (delta 0), pack-reused 19
lkconrad12580@instance-1:~$ ^C
lkconrad12580@instance-1:~$ wget http://norvig.com/ngrams/shakespeare.txt
--2020-05-06 15:53:21--  http://norvig.com/ngrams/shakespeare.txt
Resolving norvig.com (norvig.com)... 158.106.138.13
Connecting to norvig.com (norvig.com)|158.106.138.13|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 4538523 (4.3M) [text/plain]
Saving to: 'shakespeare.txt'

shakespeare.txt                                   100%[==========================================================================

2020-05-06 15:53:22 (11.0 MB/s) - 'shakespeare.txt' saved [4538523/4538523]

lkconrad12580@instance-1:~$ 
```

Figure 11: Cloning the github repo where the MapReduce code is stored. Then, downloading the text to be analyzed

I then needed to create directories in hdfs for the raw and processed data. The hdfs directories are separate from local directories, so we will also need to copy our text document to the hdfs file system.

```
$ hdfs dfs -mkdir /shakespeare
$ hdfs dfs -mkdir /shakespeare/input
$ hdfs dfs -copyFromLocal shakespeare.txt /shakespeare/input
$ hdfs dfs -ls /shakespeare/input
```

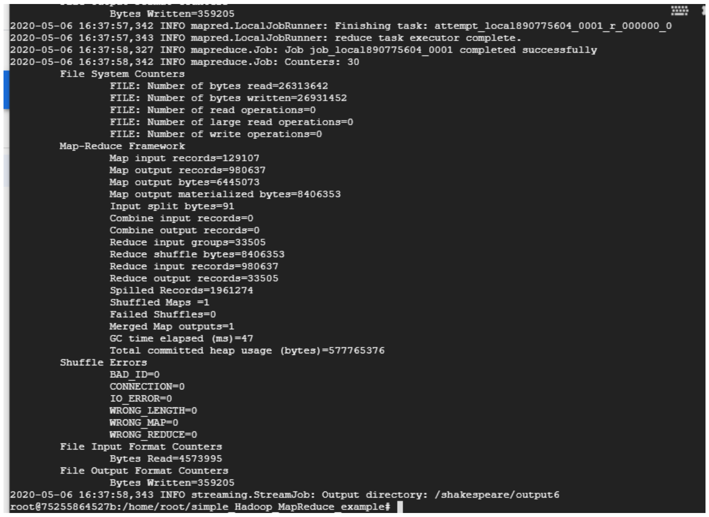


```
2020-05-06 16:13:07 (8.05 MB/s) - 'shakespeare.txt' saved [4538523/4538523]

root@5cb416616c63:/home/root# hdfs dfs -mkdir /shakespeare
root@5cb416616c63:/home/root# hdfs dfs -mkdir /shakespeare/input
root@5cb416616c63:/home/root# hdfs dfs -copyFromLocal shakespeare.txt /shakespeare/input
root@5cb416616c63:/home/root# hdfs dfs -ls /shakespeare/input
Found 1 items
-rw-r--r--   1 root root    4538523 2020-05-06 16:14 /shakespeare/input/shakespeare.txt
root@5cb416616c63:/home/root# 
```

Figure 12: Creating directories in preparation for the MapReduce output.

Now, it's time to run the MapReduce. The MapReduce implementation we are using reads in a text file and then returns how many times a word is encountered in the file. It reads in the file line by line, and as the line it being read in, it separates the line into a list of words which were separated by white space. The mapper program will then parse through each word, and simply assign a 1 to it, creating key-value pairs. The output of the mapper is sorted, in our case alphabetically, and fed into the reducer program. The reducer takes all the occurrences of a word and adds up how many times that words was encountered, thus producing a word count of all words in the document.

```
# The commands to run the MapReduce programs.  Here, our Shakespeare file
# is ran through the mapper, and then the reducer, and the output
# is saved in output6 (yes, I had a few issues that ended up being
# really silly):
$ mapred streaming \
-mapper mapper.py \
-reducer reducer.py \
-input /shakespeare/input \
-output /shakespeare/output6
```



```
                Bytes Written=359205
2020-05-06 16:37:57,342 INFO mapred.LocalJobRunner: Finishing task: attempt_local890775604_0001_r_000000_0
2020-05-06 16:37:57,343 INFO mapred.LocalJobRunner: reduce task executor complete.
2020-05-06 16:37:58,327 INFO mapreduce.Job: Job job_local890775604_0001 completed successfully
2020-05-06 16:37:58,342 INFO mapreduce.Job: Counters: 30
        File System Counters
                FILE: Number of bytes read=26313642
                FILE: Number of bytes written=26931452
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
        Map-Reduce Framework
                Map input records=129107
                Map output records=980637
                Map output bytes=6445073
                Map output materialized bytes=8406353
                Input split bytes=91
                Combine input records=0
                Combine output records=0
                Reduce input groups=33505
                Reduce shuffle bytes=8406353
                Reduce input records=980637
                Reduce output records=33505
                Spilled Records=1961274
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=47
                Total committed heap usage (bytes)=577765376
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=4573995
        File Output Format Counters
                Bytes Written=359205
2020-05-06 16:37:58,343 INFO streaming.StreamJob: Output directory: /shakespeare/output6
root@75255864527b:/home/root/simple_Hadoop_MapReduce_example# 
```

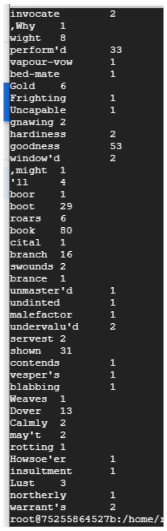
Figure 13: Output of running the MapReduce streaming command.

Finally, I viewed the results of the MapReduce analysis, and copied the output to my local file system:

```
$ hdfs dfs -ls /shakespeare/output6
$ hdfs dfs -cat /shakespeare/output6/part-00000
$ hdfs dfs -copyToLocal /shakespeare/output6/part-00000 result
$ sort -gr -k 2 result | head
```

4 Results and Output

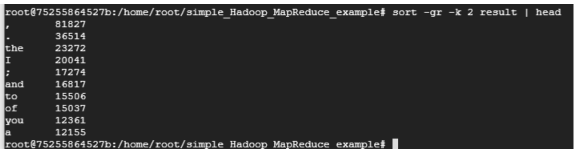
While most of this lab involved setting up our google cloud platform, and then Docker containers, there were some data produced as a result of the MapReduce ran on the Shakespeare text. In the first image below we see the unsorted results of the MapReduce in the format of (word, number of instances). The MapReduce program reads in files line by line while separating each line at its white spaces. This is why we see punctuation in our output, and some odd breaks in words.



```
invocate        2
,Why    1
wight   8
perform'd       33
vapour-vow      1
bed-mate        1
Gold    6
Frighting       1
Uncapable       1
gnawing 2
hardiness       2
goodness        53
window'd        2
,might  1
'll     4
boor    1
boot    29
roars   6
book    80
cital   1
branch  16
swounds 2
brance  1
unmaster'd      1
undinted        1
malefactor      1
undervalu'd     2
servest 2
shown   31
contends        1
vesper's        1
blabbing        1
Weaves  1
Dover   13
Calmly  2
may't   2
rotting 1
Howsoe'er       1
insultment      1
Lust    3
northerly       1
warrant's       2
root@75255864527b:/home/r
```

Figure 14: Results of running MapReduce on Shakespeare text.

Here is the same data after running the sort command, which sorts the output by most frequently encountered word.



```
root@75255864527b:/home/root/simple_Hadoop_MapReduce_example# sort -gr -k 2 result | head
,       81827
.       36514
the     23272
I       20041
;       17274
and     16817
to      15506
of      15037
you     12361
a       12155
root@75255864527b:/home/root/simple_Hadoop_MapReduce_example# 
```

Figure 15: Word count produced by MapReduce, sorted.

5 Analysis

The MapReduce ran on the Shakespeare text produced the results as we would expect; without any sorting the results are simply a list of words and their counts. The sorted results, which look to be sorted, in descending order, according to the most frequently seen ”word,” show results that if we were actually analyzing a text file would likely not be of much use. The most counted word is a comma, and other punctuation marks follow, along with the most frequently used English words such as ”the”, ”and”, ”I”, ”to”. The text would be better analyzed after proper cleaning where we could remove unwanted characters such as the punctuation, make all letters the same case (so island and Island aren’t 2 different words), and removing unwanted text that tends to be found at the beginning and end of a text document.

6 Conclusion

I have heard of Google Cloud Platform, but have never utilized it for anything. In fact, I barely knew what it did until this lab. Same can be said about cloud computing and Docker. After doing the written portion of the lab, I began filling in some of the holes in my knowledge, while also learning about technologies I knew little about. Then, seeing many of these technologies in action really helped to drive home their uses.

I also found myself going back after completing the written questions, and reading more about cloud computing, GCP, Docker, and Hadoop. In particular, I wanted to understand Docker better, as I was struggling to understand what it was and why we used it. I believe I better understand it now, and I am even working through some tutorials at `training.play-with-docker.com` to beef up my Docker knowledge, and to feel more comfortable using it. Now that I have a better idea of what it does and why it's important to Data Science, I can understand why it is in such high demand right now. I can also see why it hasn't and likely wont replace virtual machines anytime soon.

As far as does data science or data engineering interest me more, I am still figuring that out. After discussing this topic on the forums, I do feel that the skills and requirements of each of these careers are more clear to me, yet I still want to learn more about data engineering before I can fully commit to either one.

7 References

1. *5 top cloud service providers companies in the world.* (2019, January 30). DataFlair. `https://data-flair.training/blogs/cloud-service-providers-companies/`

2. Castrounis, A. (2017, September 20). *Cloud computing and architecture for data scientists.* DataCamp Community. `https://www.datacamp.com/community/blog/data-science-cloud`

3. *Data engineering and science in the cloud.* (2019, August 29). Cloudera. `https://www.cloudera.com/products/data-science-and-engineering/data-engineering.html`

4. Harvey, C., & Patrizio, A. (2020, March 17). *AWS vs. Azure vs. Google: Cloud comparison.* Datamation: Emerging Enterprise Tech Analysis and Products. `https://www.datamation.com/cloud-computing/aws-vs-azure-vs-google-cloud-comparison.html`

5. Hufford, J. (18, February 6). *Cloud vs SaaS: What's the difference?* — nChannel blog. eCommerce, ERP, POS & 3PL System Integration — nChannel. `https://www.nchannel.com/blog/cloud-vs-saas/`

6. Husain, H. (2018, January 17). *How docker can help you become a more effective data scientist.* Towards Data Science. `https://towardsdatascience.com/how-docker-can-help-you-become-a-more-effective-da`

7. *LinkedIn's 2017 U.S. Emerging Jobs Report.* 2017. LinkedIn.

   `news.linkedin.com/2017/12/introducing-linkedins-2017-u-s--emerging-jobs-report`.

8. Misal, D. (2019, October 27). *Linux vs Windows: Which is the best OS for data scientists?* Analytics India Magazine.

   `https://analyticsindiamag.com/linux-vs-windows-which-is-the-best-os-for-data-scientists/`

9. Rouse, M. (2018, September 2). *What is infrastructure as a service (IaaS)? - Definition from WhatIs.com.* SearchCloudComputing. `https://searchcloudcomputing.techtarget.com/definition/Infrastructure-as-a-Se`

10. Saltz, J. S., Yilmazel, S., & Yilmazel, O. (2016). Not all software engineers can become good data engineers. textit2016 IEEE International Conference on Big Data (Big Data).

11. *The state of data engineering — Stitch benchmark report.* (2019). Stitch. `https://www.stitchdata.com/resources/the-state-of-data-engineering/?thanks=true`

12. Steven J. Vaughan-Nichols. (2018, March 21). *What is docker and why is it so darn popular?* ZDNet. `https://www.zdnet.com/article/what-is-docker-and-why-is-it-so-darn-popular/`

13. *Understanding Linux.* (2020). Red Hat - We make open source technologies for the enterprise. `https://www.redhat.com/en/topics/linux`

14. Violino, B. (n.d.). *What is PaaS? platform-as-a-service explained.* InfoWorld. `https://www.infoworld.com/article/3223434/what-is-paas-software-development-in-the-cloud.html`

15. *What is a container?* (2020). Docker. `https://www.docker.com/resources/what-container`

16. *What is docker?* (2019). Opensource.com. urlhttps://opensource.com/resources/what-docker

17. *What is Linux?* (2019). Opensource.com. https://opensource.com/resources/linux

18. Yang, C., Huang, Q., Li, Z., Liu, K., & Hu, F. (2016). Big data and cloud computing: Innovation opportunities and challenges. *International Journal of Digital Earth*, 10(1), 13-53. https://doi.org/10.1080/17538947.2016.1239771