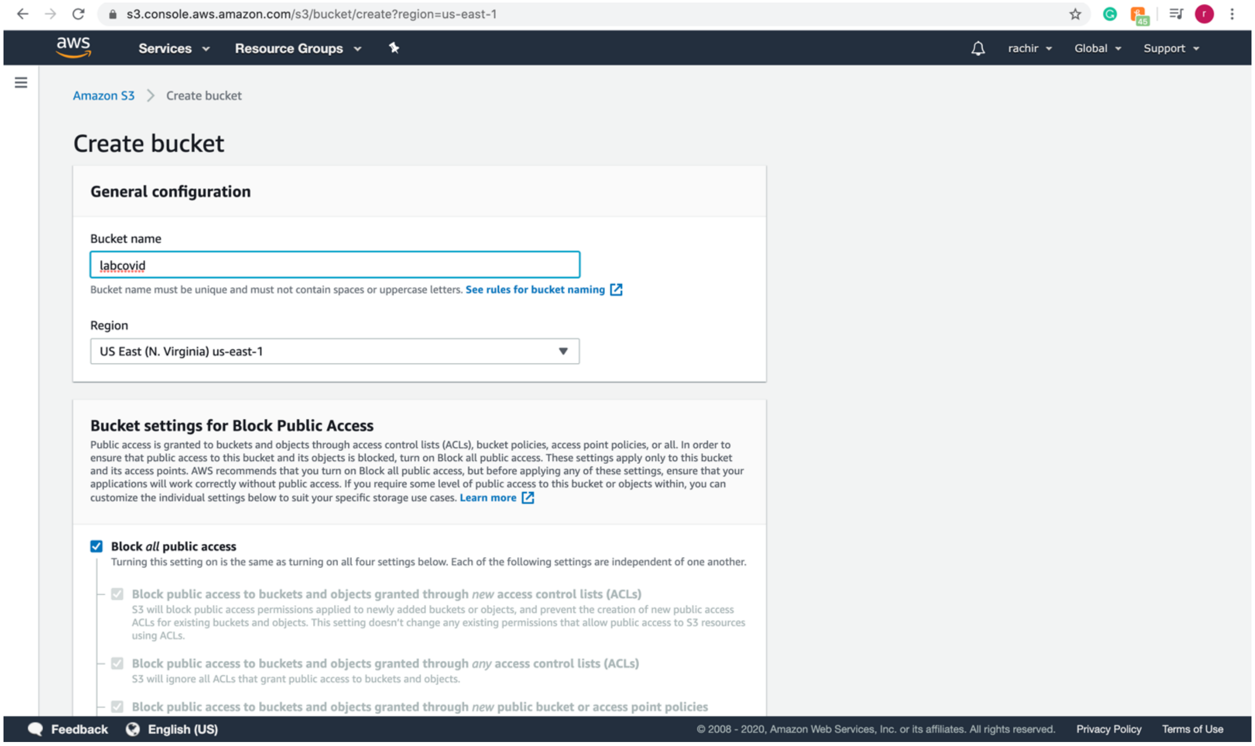
- Downloaded the five data and code provided from the google drive in the Lab 5.
- Creating an Amazon S3 bucket.


s3.console.aws.amazon.com/s3/bucket/create?region=us-east-1
aws
Services
Resource Groups
rachir
Global
Support
Amazon S3 > Create bucket
Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Create bucket

General configuration

Bucket name

labcovid

Bucket name must be unique and must not contain spaces or uppercase letters. See rules for bucket naming

Region

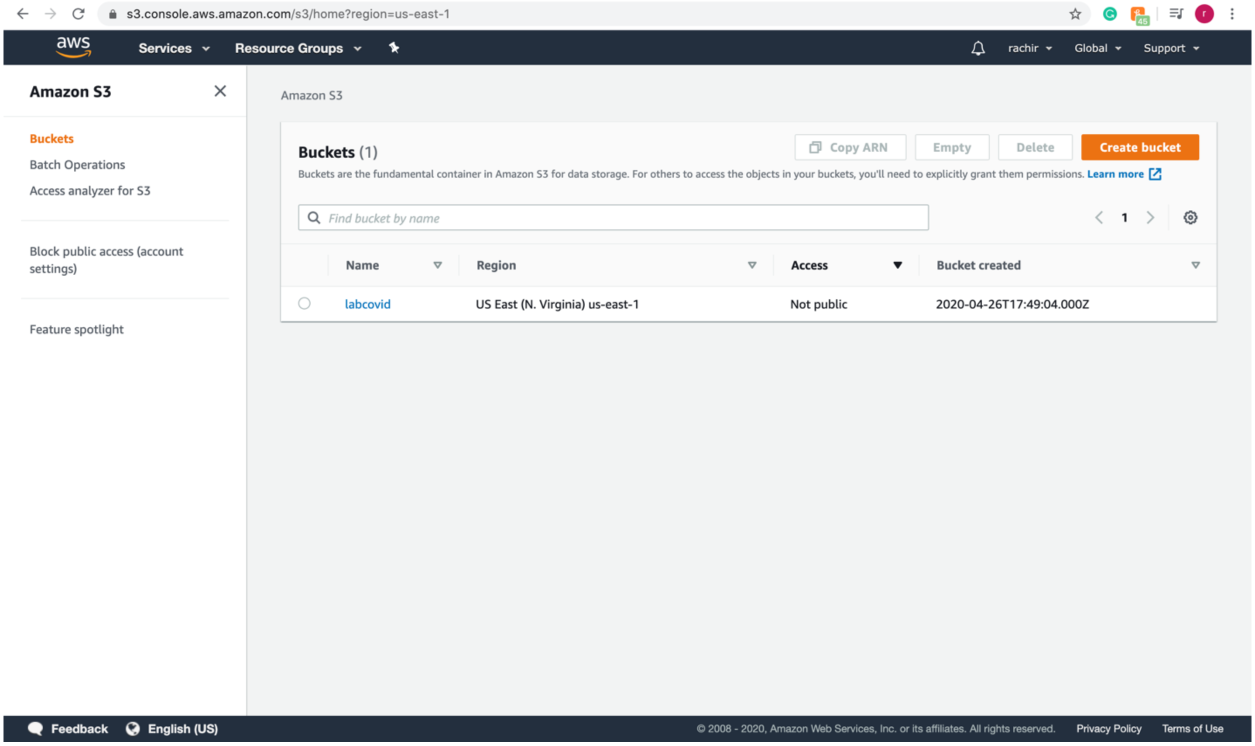
US East (N. Virginia) us-east-1

Bucket settings for Block Public Access

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

☑ Block *all* public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- ☑ Block public access to buckets and objects granted through *new* access control lists (ACLs)
  S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- ☑ Block public access to buckets and objects granted through *any* access control lists (ACLs)
  S3 will ignore all ACLs that grant public access to buckets and objects.
- ☑ Block public access to buckets and objects granted through *new* public bucket or access point policies

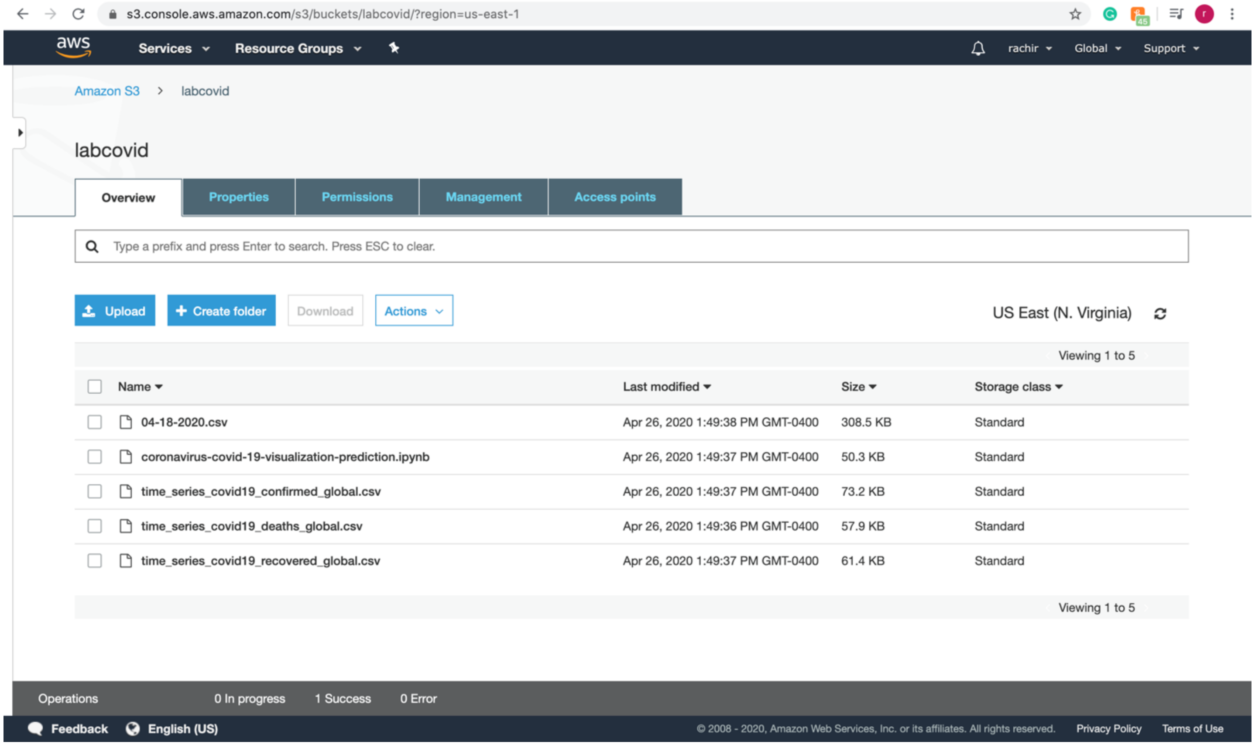


Buckets (1)

Buckets are the fundamental container in Amazon S3 for data storage. For others to access the objects in your buckets, you'll need to explicitly grant them permissions. Learn more

	Name	Region	Access	Bucket created
○	labcovid	US East (N. Virginia) us-east-1	Not public	2020-04-26T17:49:04.000Z

- Uploaded the datas and code provided into the S3 bucket and leave it as default making it not public.



s3.console.aws.amazon.com/s3/buckets/labcovid/?region=us-east-1

Amazon S3 > labcovid

labcovid

Overview | Properties | Permissions | Management | Access points

Type a prefix and press Enter to search. Press ESC to clear.

Upload | Create folder | Download | Actions

US East (N. Virginia)

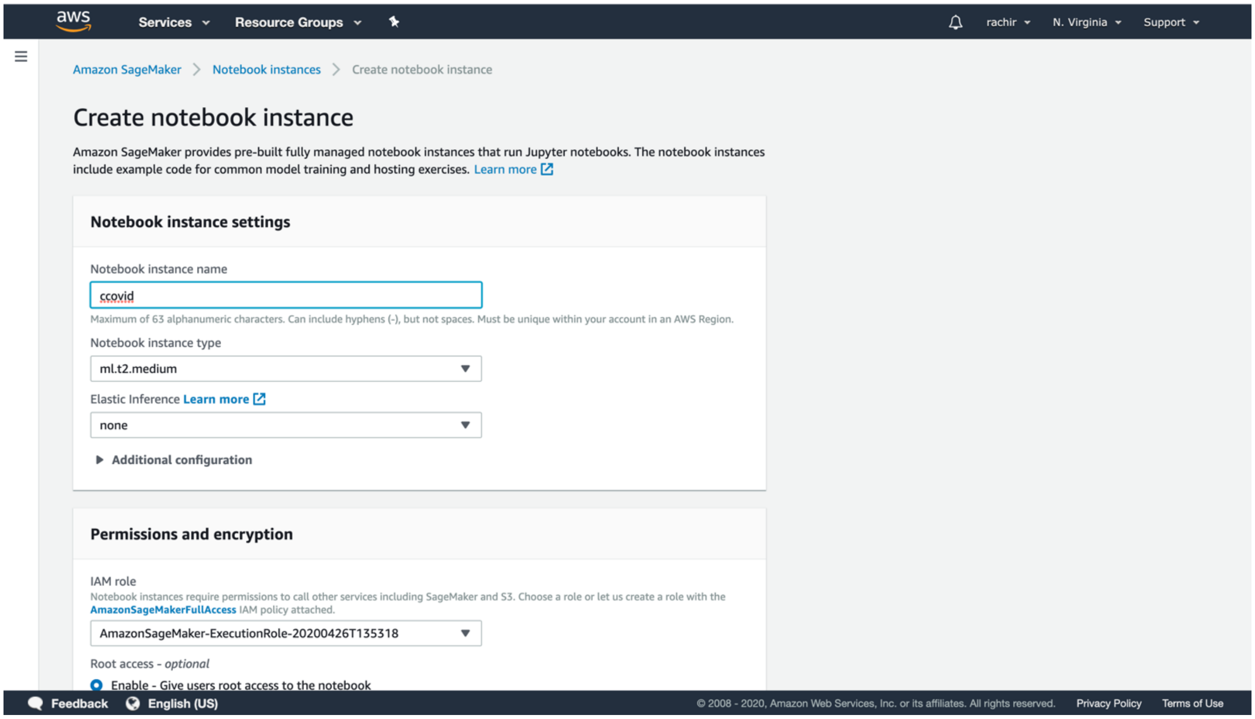
Viewing 1 to 5

Name	Last modified	Size	Storage class
04-18-2020.csv	Apr 26, 2020 1:49:38 PM GMT-0400	308.5 KB	Standard
coronavirus-covid-19-visualization-prediction.ipynb	Apr 26, 2020 1:49:37 PM GMT-0400	50.3 KB	Standard
time_series_covid19_confirmed_global.csv	Apr 26, 2020 1:49:37 PM GMT-0400	73.2 KB	Standard
time_series_covid19_deaths_global.csv	Apr 26, 2020 1:49:36 PM GMT-0400	57.9 KB	Standard
time_series_covid19_recovered_global.csv	Apr 26, 2020 1:49:37 PM GMT-0400	61.4 KB	Standard

Viewing 1 to 5

Operations 0 In progress 1 Success 0 Error

- Open Amazon SageMaker from Services, Create notebook instance in Amazon SageMaker:



Amazon SageMaker > Notebook instances > Create notebook instance

Create notebook instance

Amazon SageMaker provides pre-built fully managed notebook instances that run Jupyter notebooks. The notebook instances include example code for common model training and hosting exercises. Learn more

Notebook instance settings

Notebook instance name: ccovid

Maximum of 63 alphanumeric characters. Can include hyphens (-), but not spaces. Must be unique within your account in an AWS Region.

Notebook instance type: ml.t2.medium

Elastic Inference Learn more: none

▶ Additional configuration

Permissions and encryption

IAM role

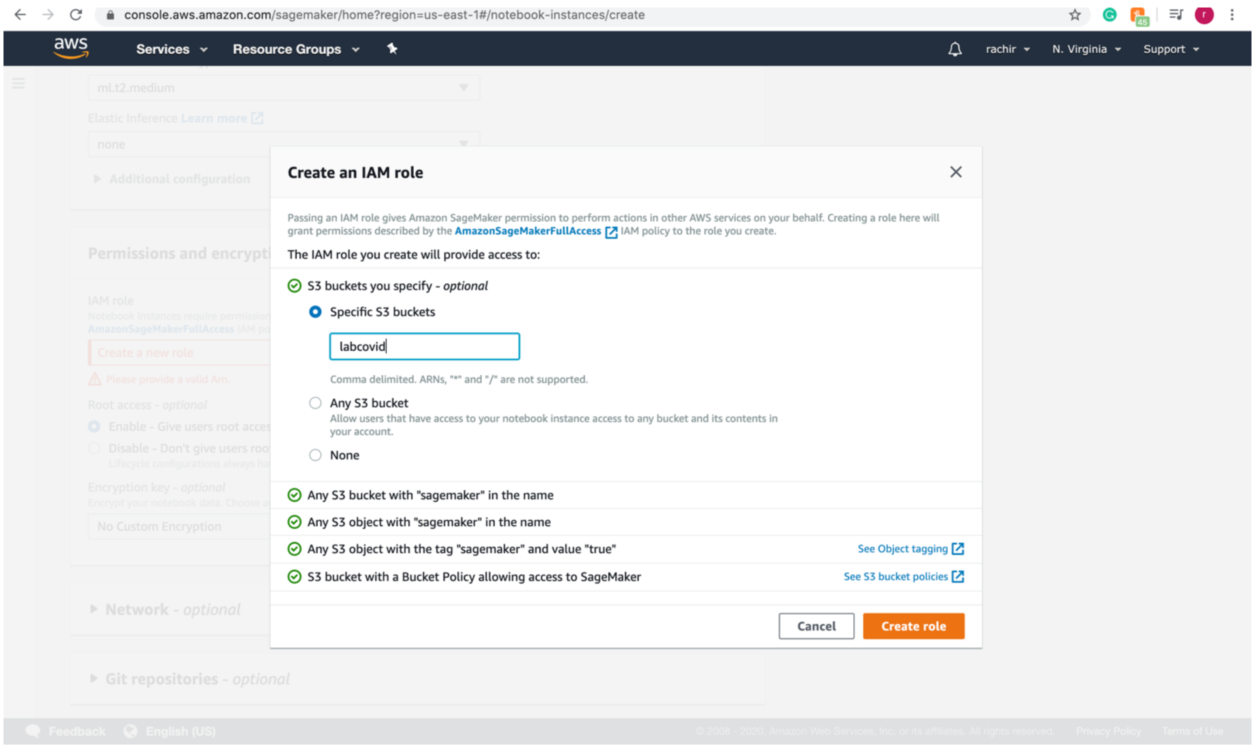
Notebook instances require permissions to call other services including SageMaker and S3. Choose a role or let us create a role with the AmazonSageMakerFullAccess IAM policy attached.

AmazonSageMaker-ExecutionRole-20200426T135318

Root access - *optional*

● Enable - Give users root access to the notebook

- Launching a Notebook instance and same time creating the new IAM permissions for the Amazon S3 bucket we had created.



console.aws.amazon.com/sagemaker/home?region=us-east-1#/notebook-instances/create

Create an IAM role

Passing an IAM role gives Amazon SageMaker permission to perform actions in other AWS services on your behalf. Creating a role here will grant permissions described by the AmazonSageMakerFullAccess IAM policy to the role you create.

The IAM role you create will provide access to:

✓ S3 buckets you specify - *optional*

● Specific S3 buckets

labcovid

Comma delimited. ARNs, "*" and "/" are not supported.

○ Any S3 bucket
Allow users that have access to your notebook instance access to any bucket and its contents in your account.

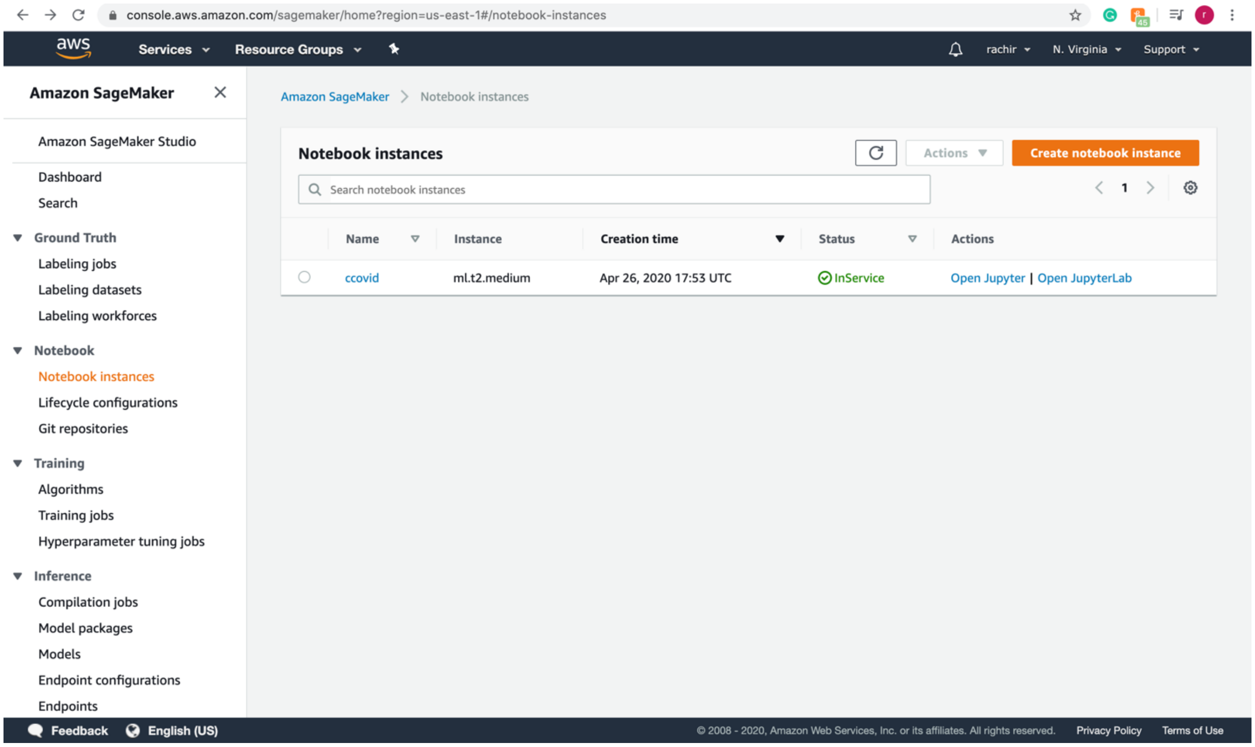
○ None

✓ Any S3 bucket with "sagemaker" in the name

✓ Any S3 object with "sagemaker" in the name

✓ Any S3 object with the tag "sagemaker" and value "true" — See Object tagging

✓ S3 bucket with a Bucket Policy allowing access to SageMaker — See S3 bucket policies



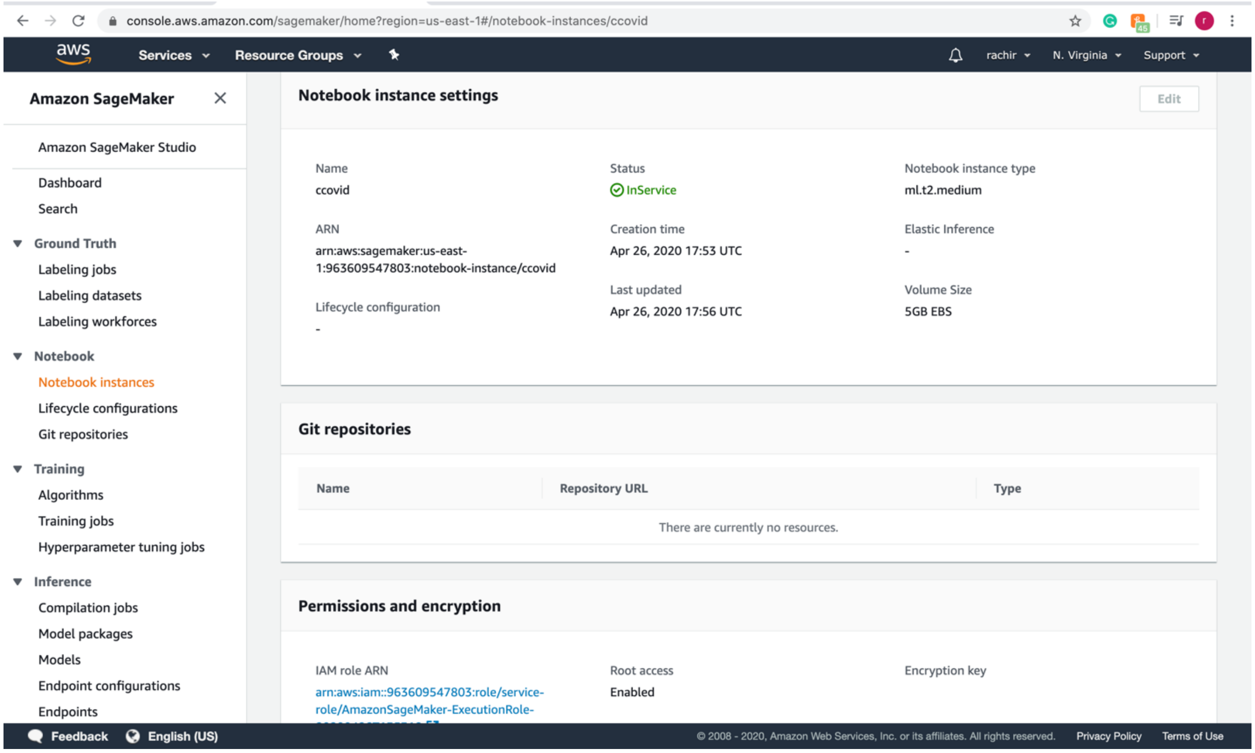
console.aws.amazon.com/sagemaker/home?region=us-east-1#/notebook-instances

Amazon SageMaker > Notebook instances

Notebook instances

	Name	Instance	Creation time	Status	Actions
○	ccovid	ml.t2.medium	Apr 26, 2020 17:53 UTC	✓ InService	Open Jupyter | Open JupyterLab

- Load the notebook, Open Jupyter or Open JupyterLab.



Notebook instance settings

Name ccovid	Status InService	Notebook instance type ml.t2.medium
ARN arn:aws:sagemaker:us-east-1:963609547803:notebook-instance/ccovid	Creation time Apr 26, 2020 17:53 UTC	Elastic Inference -
Lifecycle configuration -	Last updated Apr 26, 2020 17:56 UTC	Volume Size 5GB EBS

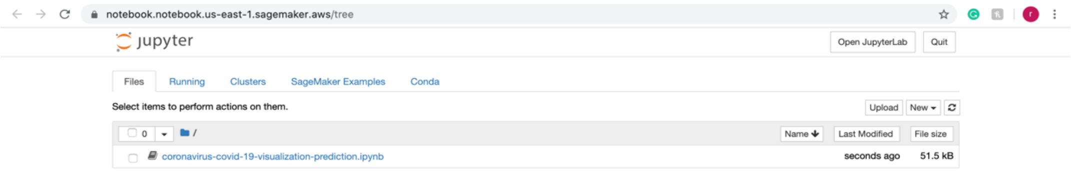
Git repositories

Name	Repository URL	Type
There are currently no resources.		

Permissions and encryption

IAM role ARN arn:aws:iam::963609547803:role/service-role/AmazonSageMaker-ExecutionRole-	Root access Enabled	Encryption key

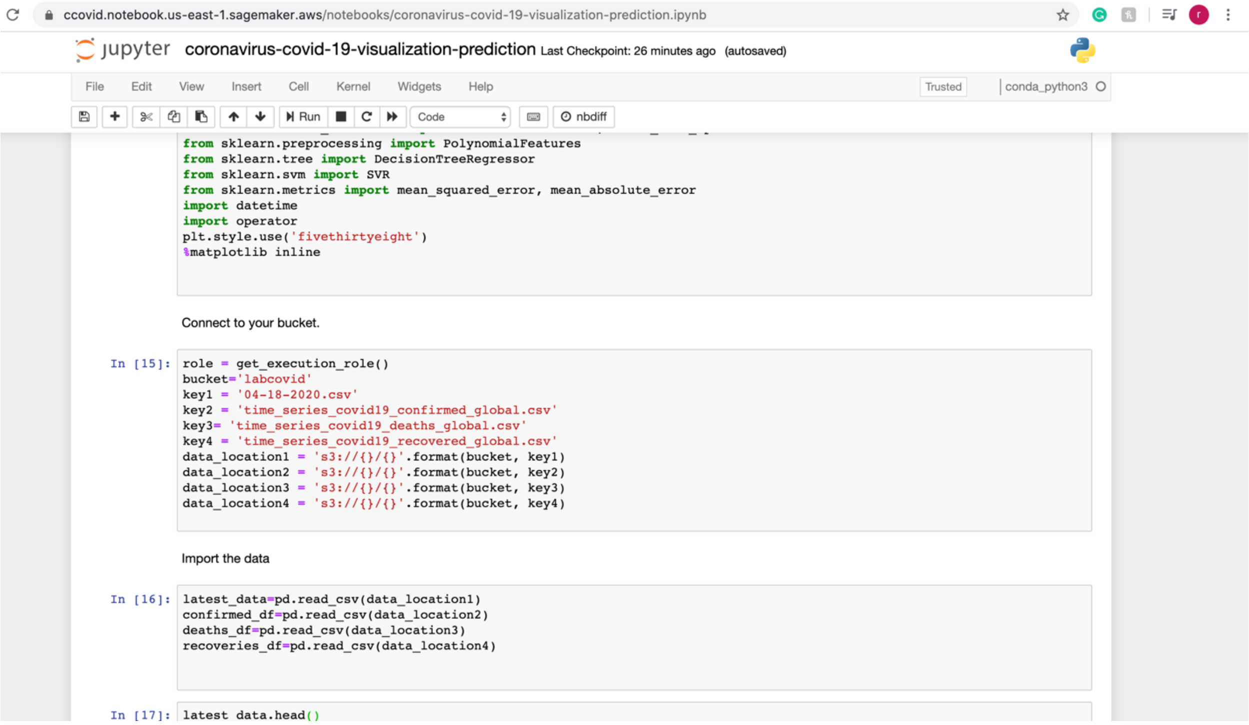
- Uploading the python file code into the Jupyter notebook.



Select items to perform actions on them.

Name	Last Modified	File size
coronavirus-covid-19-visualization-prediction.ipynb	seconds ago	51.5 kB

- Editing the bucket name and running the code on each line of notebook.


ccovid.notebook.us-east-1.sagemaker.aws/notebooks/coronavirus-covid-19-visualization-prediction.ipynb
jupyter coronavirus-covid-19-visualization-prediction Last Checkpoint: 26 minutes ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Trusted
conda_python3
Run
Code
nbdiff

```
from sklearn.preprocessing import PolynomialFeatures
from sklearn.tree import DecisionTreeRegressor
from sklearn.svm import SVR
from sklearn.metrics import mean_squared_error, mean_absolute_error
import datetime
import operator
plt.style.use('fivethirtyeight')
%matplotlib inline
```

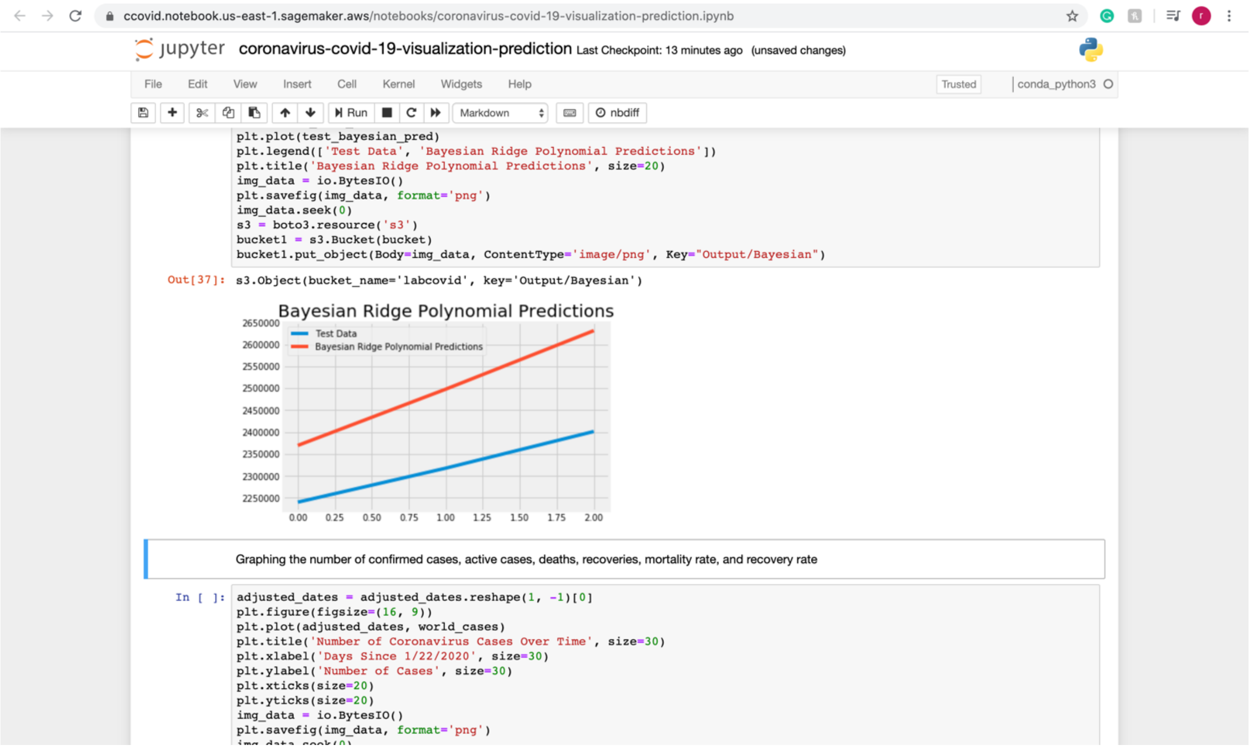
Connect to your bucket.

```
In [15]: role = get_execution_role()
         bucket='labcovid'
         key1 = '04-18-2020.csv'
         key2 = 'time_series_covid19_confirmed_global.csv'
         key3= 'time_series_covid19_deaths_global.csv'
         key4 = 'time_series_covid19_recovered_global.csv'
         data_location1 = 's3://{}/{}'.format(bucket, key1)
         data_location2 = 's3://{}/{}'.format(bucket, key2)
         data_location3 = 's3://{}/{}'.format(bucket, key3)
         data_location4 = 's3://{}/{}'.format(bucket, key4)
```

Import the data

```
In [16]: latest_data=pd.read_csv(data_location1)
         confirmed_df=pd.read_csv(data_location2)
         deaths_df=pd.read_csv(data_location3)
         recoveries_df=pd.read_csv(data_location4)
```

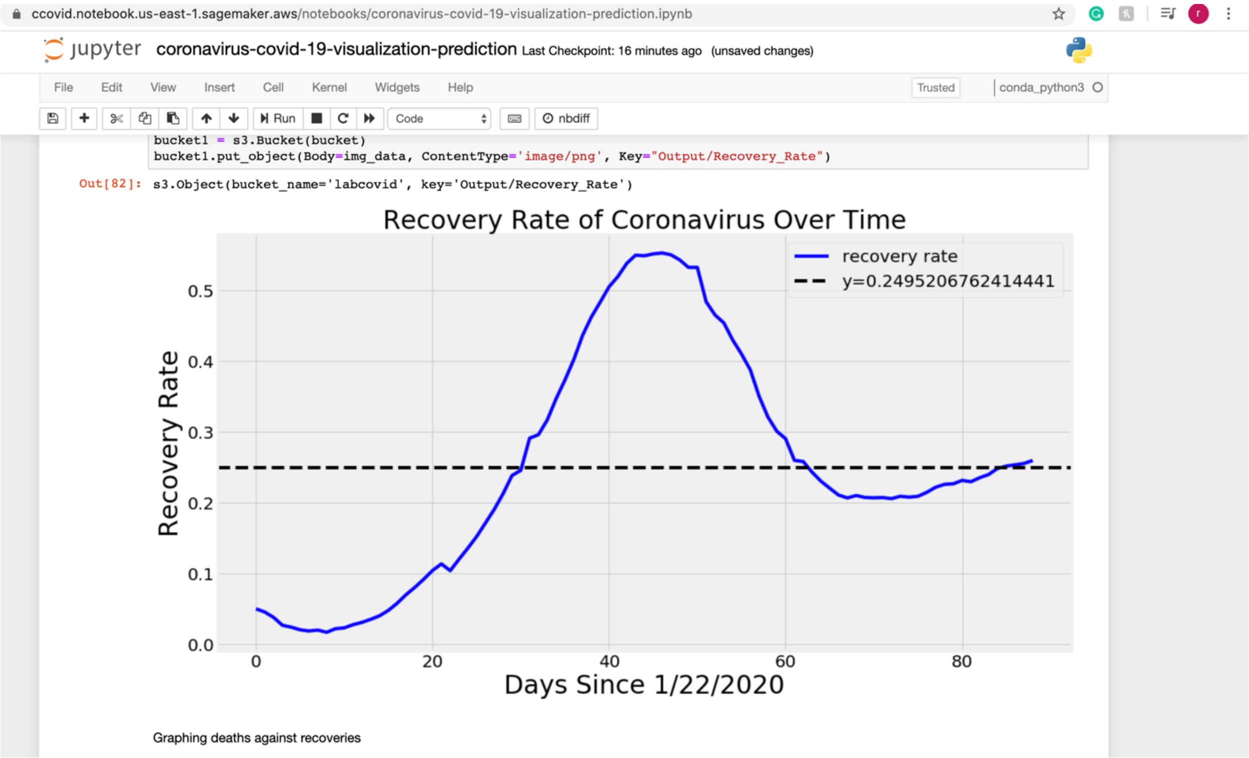
```
In [17]: latest data.head()
```


ccovid.notebook.us-east-1.sagemaker.aws/notebooks/coronavirus-covid-19-visualization-prediction.ipynb
jupyter coronavirus-covid-19-visualization-prediction Last Checkpoint: 13 minutes ago (unsaved changes)
File Edit View Insert Cell Kernel Widgets Help
Trusted
conda_python3
Run
Markdown
nbdiff
Bayesian Ridge Polynomial Predictions
Test Data
Bayesian Ridge Polynomial Predictions
2650000
2600000
2550000
2500000
2450000
2400000
2350000
2300000
2250000
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00

```
plt.plot(test_bayesian_pred)
plt.legend(['Test Data', 'Bayesian Ridge Polynomial Predictions'])
plt.title('Bayesian Ridge Polynomial Predictions', size=20)
img_data = io.BytesIO()
plt.savefig(img_data, format='png')
img_data.seek(0)
s3 = boto3.resource('s3')
bucket1 = s3.Bucket(bucket)
bucket1.put_object(Body=img_data, ContentType='image/png', Key="Output/Bayesian")

Out[37]: s3.Object(bucket_name='labcovid', key='Output/Bayesian')
```

Graphing the number of confirmed cases, active cases, deaths, recoveries, mortality rate, and recovery rate

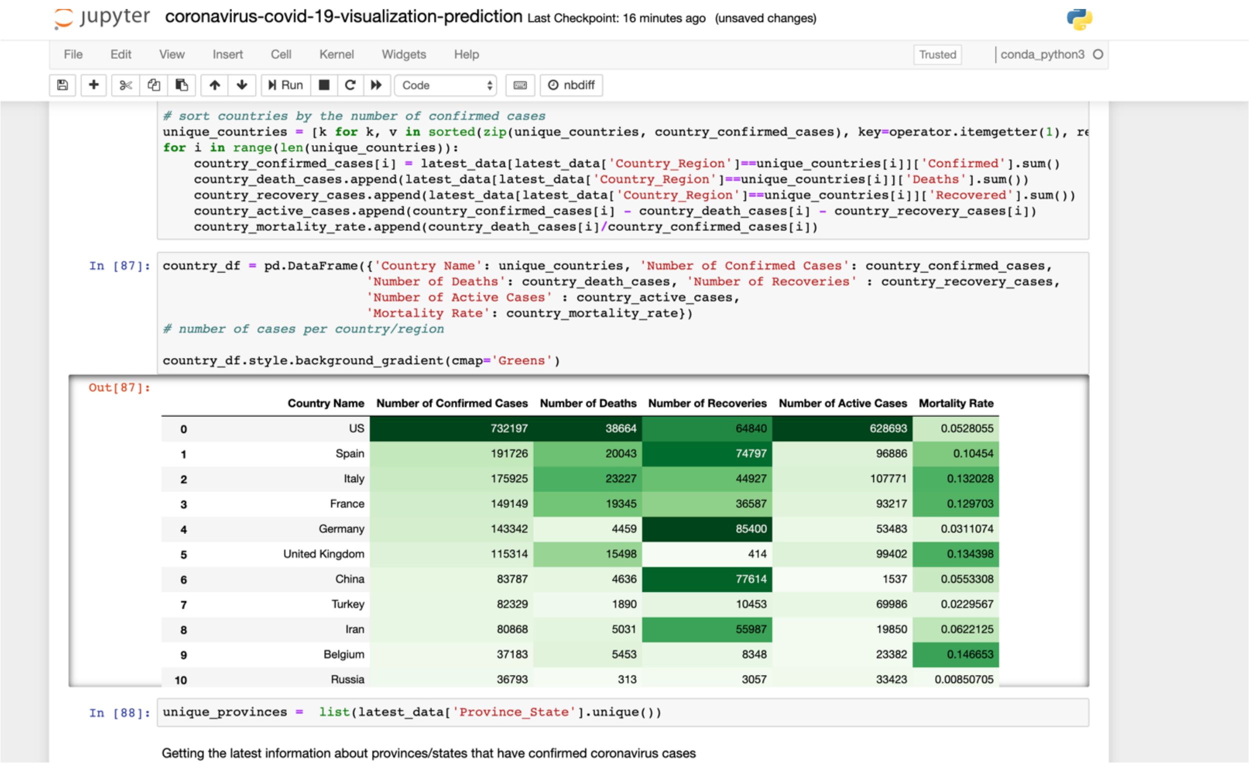
```
In [ ]: adjusted_dates = adjusted_dates.reshape(1, -1)[0]
        plt.figure(figsize=(16, 9))
        plt.plot(adjusted_dates, world_cases)
        plt.title('Number of Coronavirus Cases Over Time', size=30)
        plt.xlabel('Days Since 1/22/2020', size=30)
        plt.ylabel('Number of Cases', size=30)
        plt.xticks(size=20)
        plt.yticks(size=20)
        img_data = io.BytesIO()
        plt.savefig(img_data, format='png')
```


ccovid.notebook.us-east-1.sagemaker.aws/notebooks/coronavirus-covid-19-visualization-prediction.ipynb
jupyter coronavirus-covid-19-visualization-prediction Last Checkpoint: 16 minutes ago (unsaved changes)
File Edit View Insert Cell Kernel Widgets Help
Trusted
conda_python3
Run
Code
nbdiff
Recovery Rate of Coronavirus Over Time
recovery rate
y=0.2495206762414441
Recovery Rate
0.0
0.1
0.2
0.3
0.4
0.5
0
20
40
60
80
Days Since 1/22/2020

```
bucket1 = s3.Bucket(bucket)
bucket1.put_object(Body=img_data, ContentType='image/png', Key="Output/Recovery_Rate")
```

```
Out[82]: s3.Object(bucket_name='labcovid', key='Output/Recovery_Rate')
```

Graphing deaths against recoveries


jupyter coronavirus-covid-19-visualization-prediction Last Checkpoint: 16 minutes ago (unsaved changes)
File Edit View Insert Cell Kernel Widgets Help
Trusted
conda_python3
Run
Code
nbdiff

```
# sort countries by the number of confirmed cases
unique_countries = [k for k, v in sorted(zip(unique_countries, country_confirmed_cases), key=operator.itemgetter(1), re
for i in range(len(unique_countries)):
    country_confirmed_cases[i] = latest_data[latest_data['Country_Region']==unique_countries[i]]['Confirmed'].sum()
    country_death_cases.append(latest_data[latest_data['Country_Region']==unique_countries[i]]['Deaths'].sum())
    country_recovery_cases.append(latest_data[latest_data['Country_Region']==unique_countries[i]]['Recovered'].sum())
    country_active_cases.append(country_confirmed_cases[i] - country_death_cases[i] - country_recovery_cases[i])
    country_mortality_rate.append(country_death_cases[i]/country_confirmed_cases[i])
```

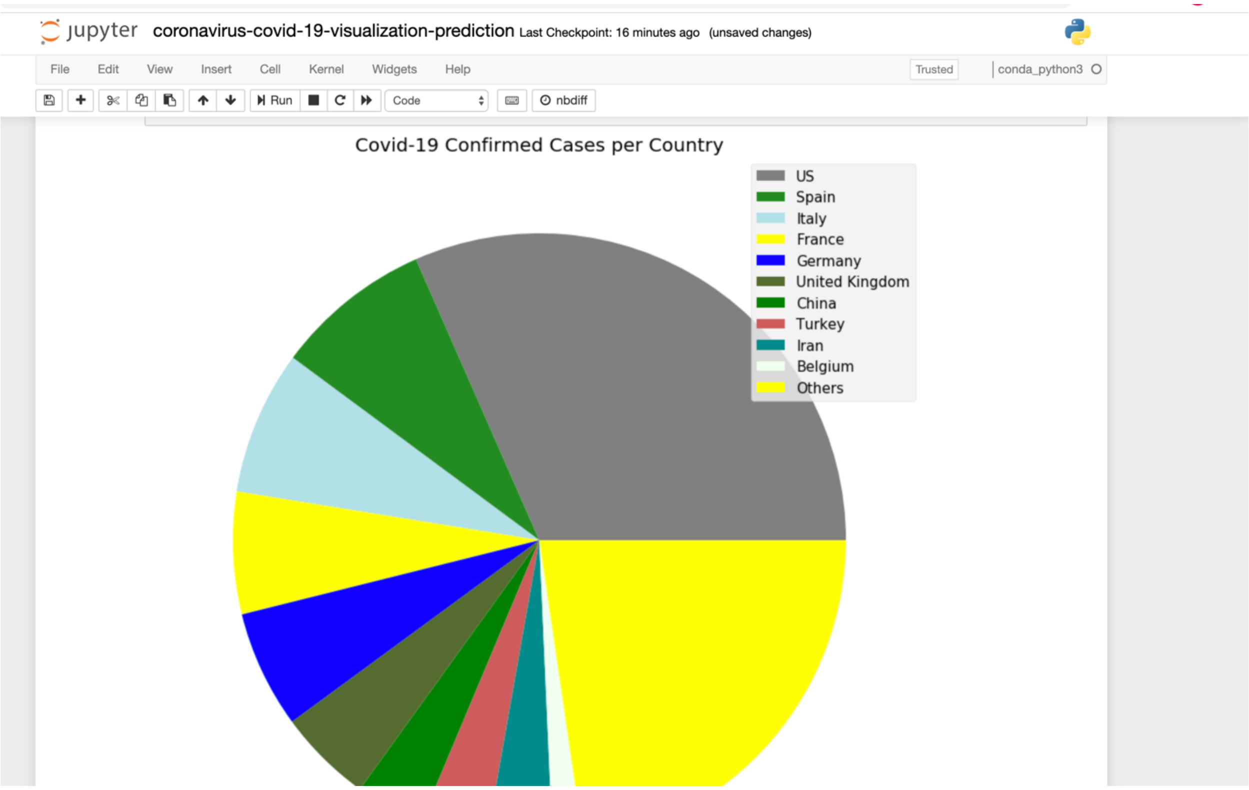
```
In [87]: country_df = pd.DataFrame({'Country Name': unique_countries, 'Number of Confirmed Cases': country_confirmed_cases,
                                   'Number of Deaths': country_death_cases, 'Number of Recoveries' : country_recovery_cases,
                                   'Number of Active Cases' : country_active_cases,
                                   'Mortality Rate': country_mortality_rate})
         # number of cases per country/region

         country_df.style.background_gradient(cmap='Greens')
```

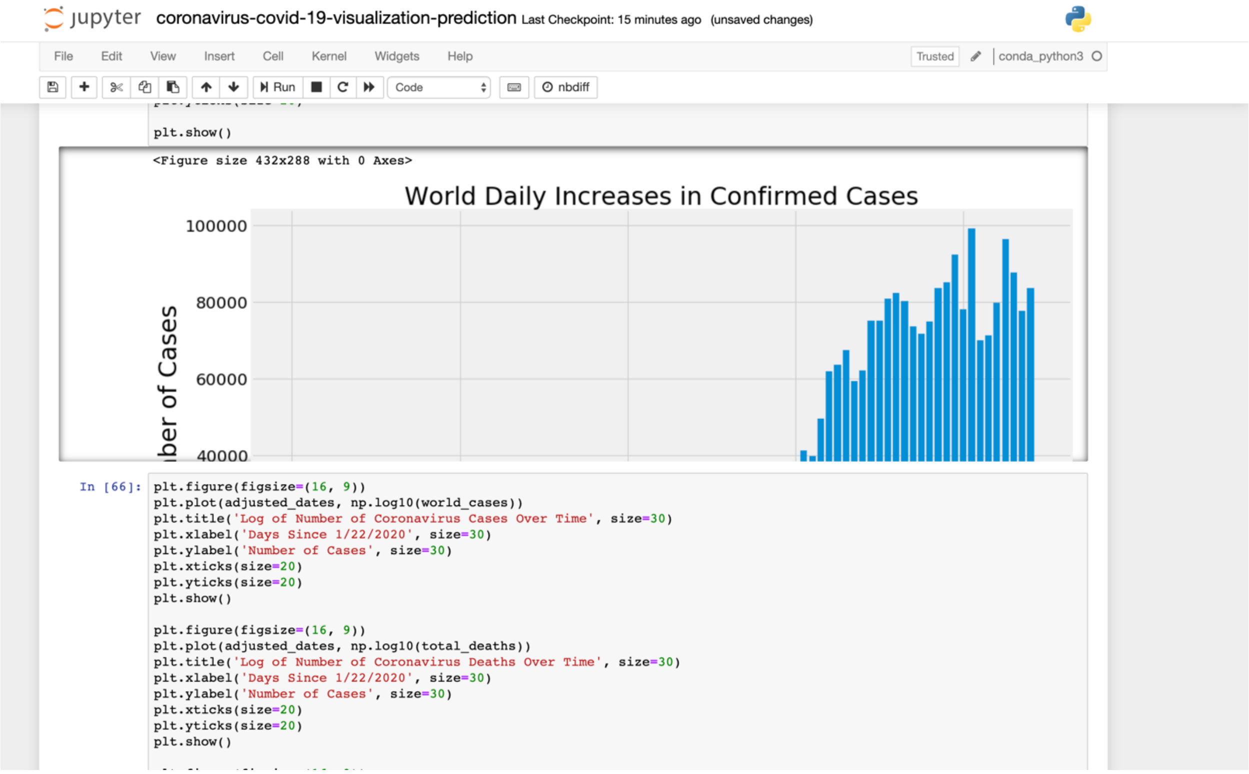
Out[87]:

	Country Name	Number of Confirmed Cases	Number of Deaths	Number of Recoveries	Number of Active Cases	Mortality Rate
0	US	732197	38664	64840	628693	0.0528055
1	Spain	191726	20043	74797	96886	0.10454
2	Italy	175925	23227	44927	107771	0.132028
3	France	149149	19345	36587	93217	0.129703
4	Germany	143342	4459	85400	53483	0.0311074
5	United Kingdom	115314	15498	414	99402	0.134398
6	China	83787	4636	77614	1537	0.0553308
7	Turkey	82329	1890	10453	69986	0.0229567
8	Iran	80868	5031	55987	19850	0.0622125
9	Belgium	37183	5453	8348	23382	0.146653
10	Russia	36793	313	3057	33423	0.00850705

```
In [88]: unique_provinces =  list(latest_data['Province_State'].unique())
```

Getting the latest information about provinces/states that have confirmed coronavirus cases




jupyter
coronavirus-covid-19-visualization-prediction Last Checkpoint: 15 minutes ago (unsaved changes)
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
conda_python3
Run
Code
nbdiff

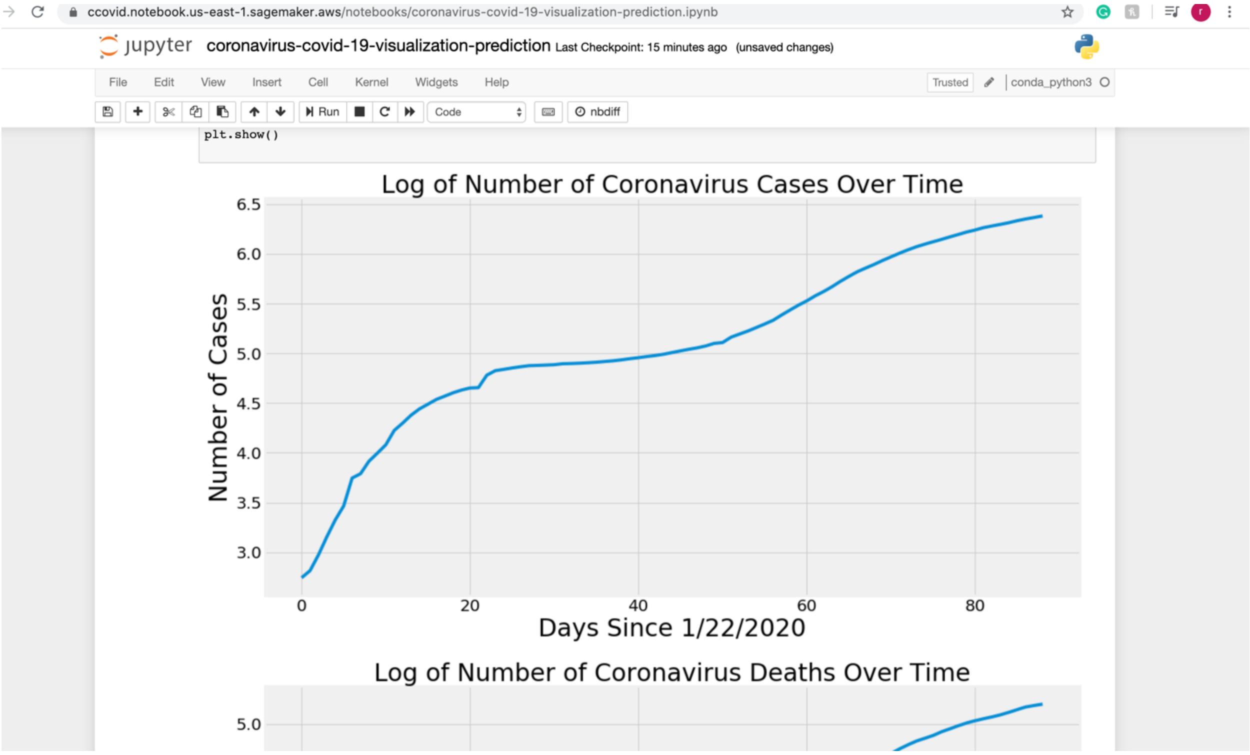
```
plt.show()
```

```
<Figure size 432x288 with 0 Axes>
```

World Daily Increases in Confirmed Cases
ber of Cases
100000
80000
60000
40000

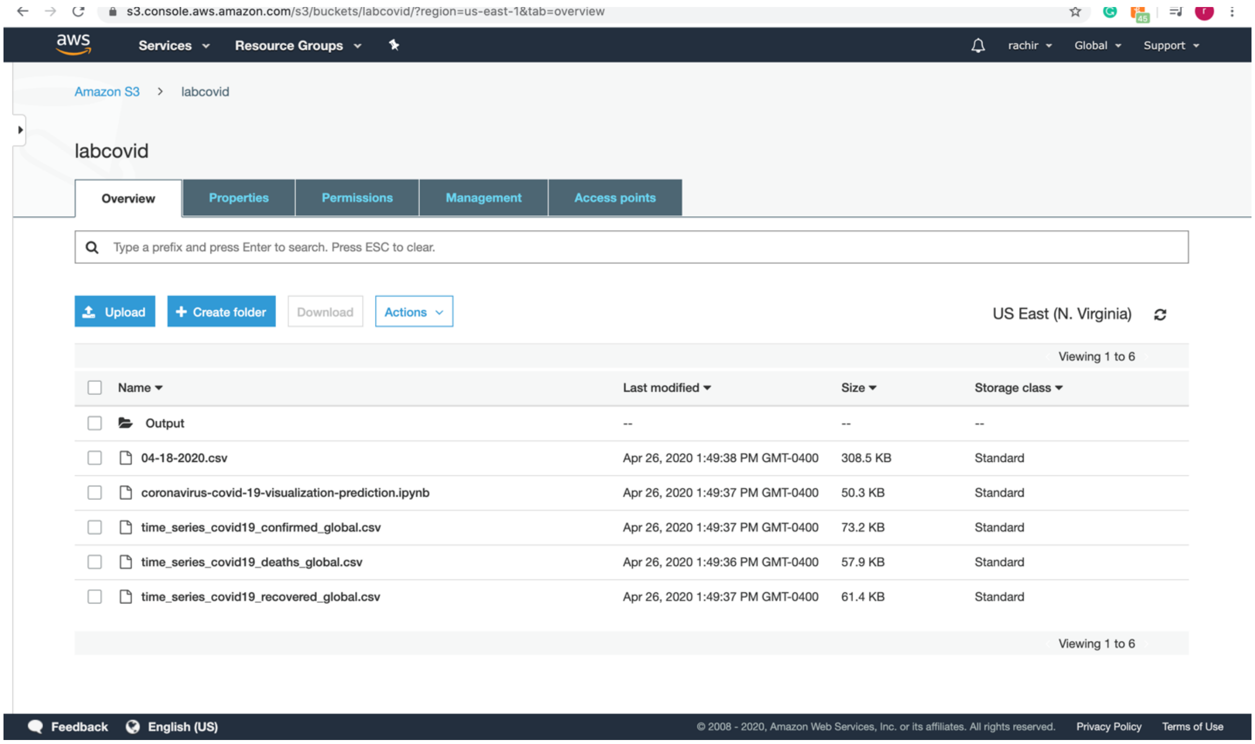
In [66]:

```
plt.figure(figsize=(16, 9))
plt.plot(adjusted_dates, np.log10(world_cases))
plt.title('Log of Number of Coronavirus Cases Over Time', size=30)
plt.xlabel('Days Since 1/22/2020', size=30)
plt.ylabel('Number of Cases', size=30)
plt.xticks(size=20)
plt.yticks(size=20)
plt.show()

plt.figure(figsize=(16, 9))
plt.plot(adjusted_dates, np.log10(total_deaths))
plt.title('Log of Number of Coronavirus Deaths Over Time', size=30)
plt.xlabel('Days Since 1/22/2020', size=30)
plt.ylabel('Number of Cases', size=30)
plt.xticks(size=20)
plt.yticks(size=20)
plt.show()
```

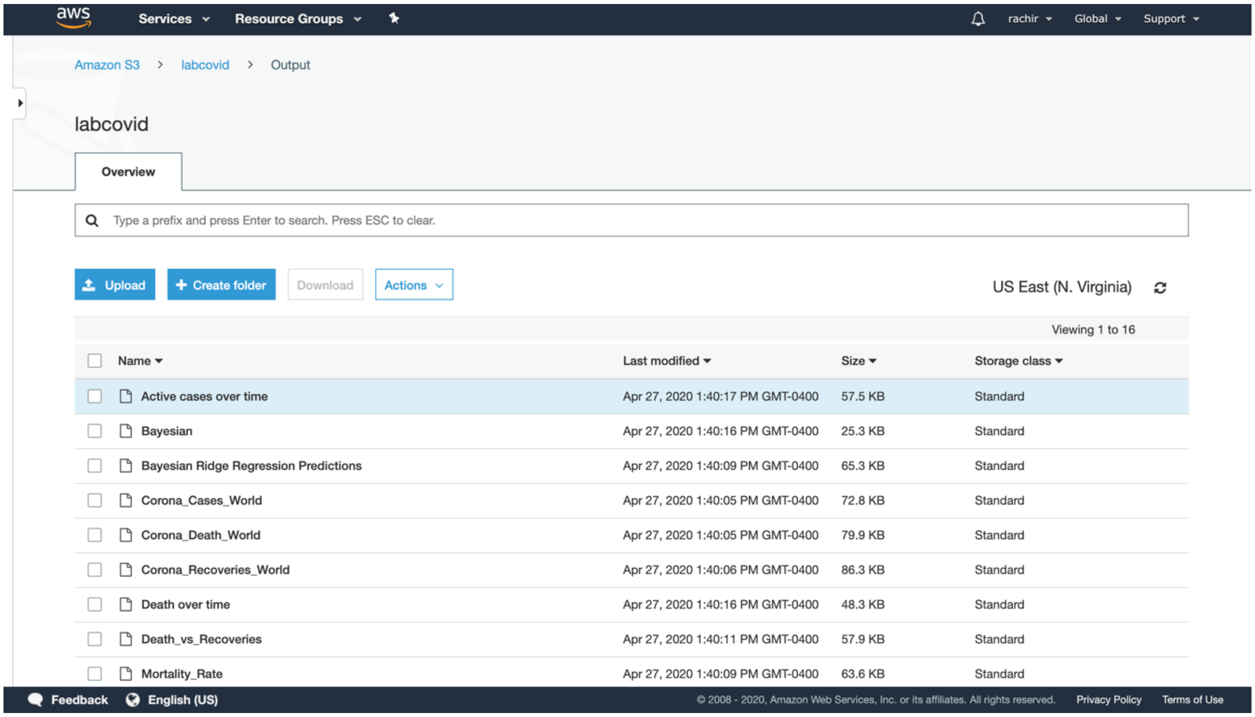

ccovid.notebook.us-east-1.sagemaker.aws/notebooks/coronavirus-covid-19-visualization-prediction.ipynb
jupyter
coronavirus-covid-19-visualization-prediction
Last Checkpoint: 15 minutes ago (unsaved changes)
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
conda_python3
Run
Code
nbdiff
Log of Number of Coronavirus Cases Over Time
Number of Cases
6.5
6.0
5.5
5.0
4.5
4.0
3.5
3.0
0
20
40
60
80
Days Since 1/22/2020
Log of Number of Coronavirus Deaths Over Time
5.0

```
plt.show()
```

- Checking the output folder in the Amazon S3 bucket we had created.



□	Name	Last modified	Size	Storage class
□	Output	--	--	--
□	04-18-2020.csv	Apr 26, 2020 1:49:38 PM GMT-0400	308.5 KB	Standard
□	coronavirus-covid-19-visualization-prediction.ipynb	Apr 26, 2020 1:49:37 PM GMT-0400	50.3 KB	Standard
□	time_series_covid19_confirmed_global.csv	Apr 26, 2020 1:49:37 PM GMT-0400	73.2 KB	Standard
□	time_series_covid19_deaths_global.csv	Apr 26, 2020 1:49:36 PM GMT-0400	57.9 KB	Standard
□	time_series_covid19_recovered_global.csv	Apr 26, 2020 1:49:37 PM GMT-0400	61.4 KB	Standard



Name	Last modified	Size	Storage class
Active cases over time	Apr 27, 2020 1:40:17 PM GMT-0400	57.5 KB	Standard
Bayesian	Apr 27, 2020 1:40:16 PM GMT-0400	25.3 KB	Standard
Bayesian Ridge Regression Predictions	Apr 27, 2020 1:40:09 PM GMT-0400	65.3 KB	Standard
Corona_Cases_World	Apr 27, 2020 1:40:05 PM GMT-0400	72.8 KB	Standard
Corona_Death_World	Apr 27, 2020 1:40:05 PM GMT-0400	79.9 KB	Standard
Corona_Recoveries_World	Apr 27, 2020 1:40:06 PM GMT-0400	86.3 KB	Standard
Death over time	Apr 27, 2020 1:40:16 PM GMT-0400	48.3 KB	Standard
Death_vs_Recoveries	Apr 27, 2020 1:40:11 PM GMT-0400	57.9 KB	Standard
Mortality_Rate	Apr 27, 2020 1:40:09 PM GMT-0400	63.6 KB	Standard

- Amazon SageMaker: Machine learning seems easy to learn but in fact it is not a lot of time is taken for learning data preprocessing, selecting the algorithm and framework, after finding out which algorithm to use we have to train and tune our model and also integrate and deploy the model. Now, this is where Amazon SageMaker steps in : - it reduces complexity by letting it do all the above things in one platform, provides independency we can build a new model or, like we did we have taken some model built somewhere else and then deployed it using AWS SageMaker the choice is given to us. Giving flexibility, AWS SageMaker: builds Machine Learning models, deploy them. Looking at AWS SageMaker does:

- Select and Prepare Training Data
- Choose and Optimize ML Model
- Setup and Manage Environment for Training
- Train and Tune Model (Trial and Error)
- Deploy ML Model to production
- Select and Manage production environment

It manages on demand needs, scaling and managing the environment even that is handled by AWS SageMaker.