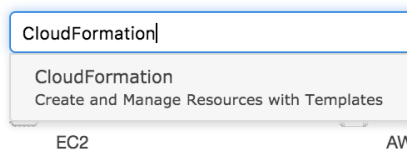


Amazon SageMaker Lab: Churn Predictive Analytics

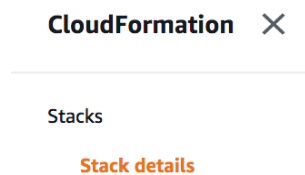
I. Prepare Environment

Before we can get to the meat of this workshop, we need to setup a minimal environment. In the following steps, we're going to launch a CloudFormation template to launch a Redshift cluster, and create an IAM role that will delegate all the permissions required to run the workshop.

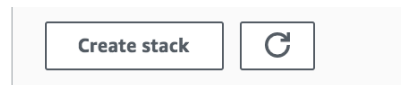
1. Log into the AWS console, and ensure you're running in the region designated for your workshop. Your user should have administrator level rights.
2. Switch over to the CloudFormation console.



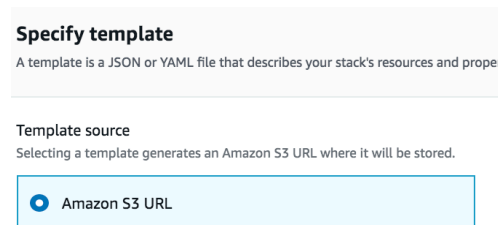
3. Select **Stacks** on the left-hand navigation panel.



4. Select the **Create Stack** button.



5. The template source should be set as **Amazon S3 URL**.



Use the following as the **S3 URL**:

<https://reinvent2018-sagemaker-pytorch.s3-us-west-2.amazonaws.com/cloudformation/workshops/churn-analytics/adv-analytics-lab.yaml>

Amazon S3 URL

<https://reinvent2018-sagemaker-pytorch.s3-us-west-2.amazonaws.com/cloudformation/workshops/churn-analytics/adv-analytics-lab.yaml>

Amazon S3 template URL

S3 URL: <https://reinvent2018-sagemaker-pytorch.s3-us-west-2.amazonaws.com/cloudformation/workshops/churn-analytics/adv-analytics-lab.yaml>

[View in Designer](#)

Select **Next**.

Cancel

Next

6. Provide a unique **Stack Name**.

Stack name

Stack name

dylantong-workshop

Stack name can include letters (A-Z and a-z), numbers

7. Set a password for your Redshift cluster. Remember the password. You will need it later in the lab. You can use the default values for all preceding fields.

DB User Password

The password that is associated with the master user account for the cluster that is being created.

.....

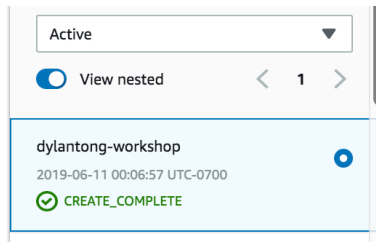
8. Select a subnet to designate a location where your Redshift cluster will launch. The instructions have been adapted for the default VPC. Launch the cluster in one of the **default subnets**.

9. Select a **security group**. Use the **default** security group.

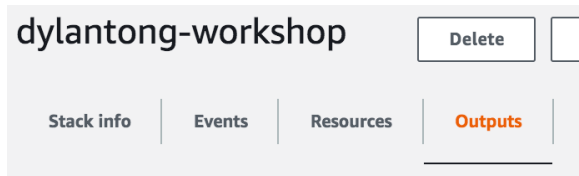
10. Select **Next** until you reach the end of the wizard. The template will create **IAM resources**. Check the box at the end of the wizard to acknowledge this.

Select **Create Stack** to launch the template.

11. The template will take 8-15 minutes to launch the resources.



12. Select the **Output** sub tab.

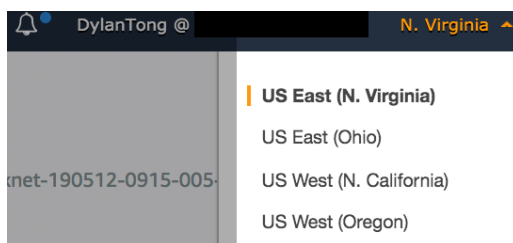


Copy down the **ClusterEndpoint** and the **WorkshopRole**. They should look similar to:

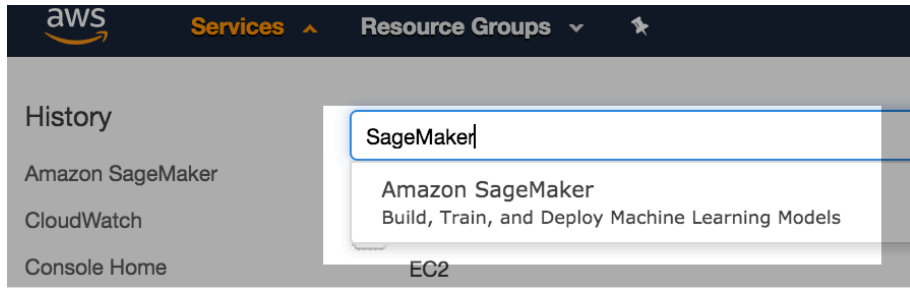
- i. **Cluster endpoint:** dylantong-workshop-rredshiftcluster-byz8ltp751p3.cohiel1b1w2e.us-east-1.redshift.amazonaws.com:5439
- ii. **Workshop Role:** arn:aws:iam::803235869972:role/dylantong-workshop-rWorkshopRole-I7XNY0HQ3WN7

II. Prepare your Development Environment

1. Log into your AWS account and ensure you're in the right region designated for your workshop. The screenshot below indicates that I'm currently in us-east-1 (N. Virginia).

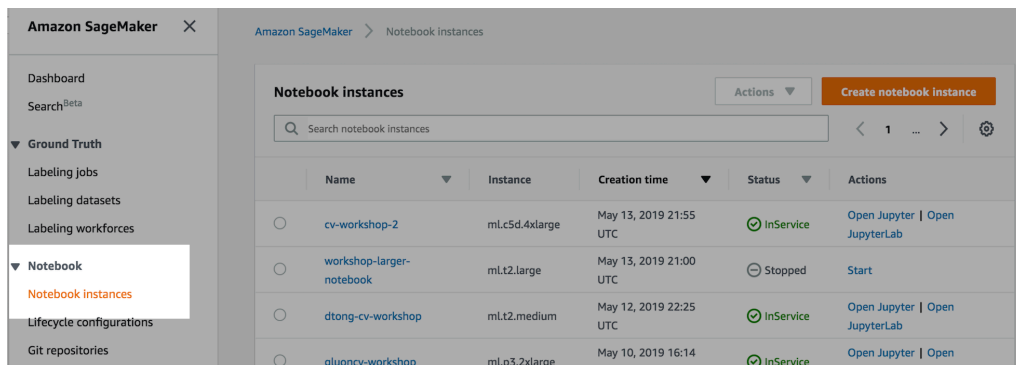


2. Navigate to the Amazon SageMaker console via the search bar.

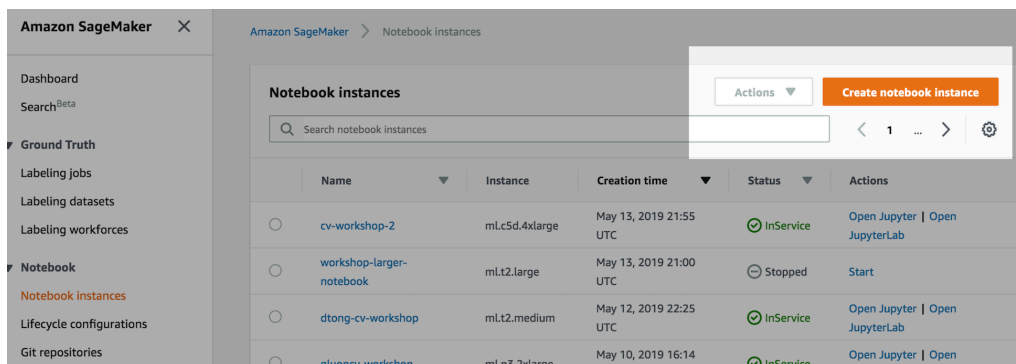


- Next, launch a managed Amazon SageMaker notebook instance. We're going to use this notebook instance to run a number of labs. In this lab, the notebook will be used to stage some raw data that we will annotate.

Switch to the **Notebook Instances** page by using the navigation menu on the left hand side of the console.



- Click on the **Create notebook instance** button.



- Next, we configure our notebook instance by working through the launch wizard. First, provide a **name** for your notebook.

Utilize a **unique prefix** that you can remember, so you more easily find the resources that belong to you.

Notebook instance name


adv-analytics-workshop

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

6. Select the ***ml.m5.4xlarge*** instance type.

Notebook instance type

ml.m5.4xlarge ▼

Elastic Inference [Learn more](#) 

none ▼

► Additional configuration

7. Your instance requires permissions to access data on S3, and execute SageMaker functionality required by this lab.

IAM role

Notebook instances require permissions to call other services including SageMaker and S3. Choose a role or let us create [AmazonSageMakerFullAccess](#) IAM policy attached.

Enter a custom IAM role ARN ▼

Create a new role

Enter a custom IAM role ARN

Use existing role

AmazonSageMaker-ExecutionRole-20171129T110981

AmazonSageMaker-ExecutionRole-20190508T145546

Custom IAM role ARN

arn:aws:iam::803235869972:role/dylantong-workshop-rWorkshopRole-I7XNY0HQ3WN7

8. Under **Network**, select the default VPC. It should appear similar to the screenshot below, but your VPC will have a different unique identifier.

▼ Network - optional

VPC - optional

Your notebook instance will be provided with SageMaker provided internet access because a VPC setting is not specified.

Default vpc-f0bd7797 (172.31.0.0/16) | DefaultVPC ▼

9. Select any subnet among the options available in the drop down.

Subnet

Choose a subnet in an availability zone supported by Amazon SageMaker.

subnet-65fff013 (172.31.32.0/20) | us-west-2a Default-subnet1 ▼

10. Select the default **security group**. This should be the same one that you selected when you deployed your CloudFormation template.

Security group(s)

▼

default

sg-7f241806 (default) | DefaultVPC_SG

sg-d52bc2ad (launch-wizard-3) | DefaultVPC_SG

11. Next, configure the Git integration. We're going to launch the notebook and clone the lab repository over to your notebook instance.

Select **"Clone a public Git repository to this notebook instance only."**

Paste the following link into the text box under **"Git repository URL"**:
<https://github.com/dylan-tong-aws/aws-advanced-analytics-jumpstarter>

▼ Git repositories - optional

▼ Default repository

Repository

Jupyter will start in this repository. Repositories are added to your home directory.

Clone a public Git repository to this notebook instance only ▼

Git repository URL

Clone a repository to use for this notebook instance only.

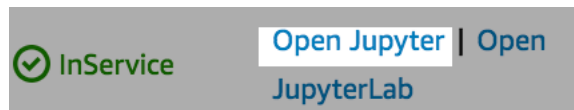
<https://github.com/dylan-tong-aws/aws-advanced-analytics-jumps>

12. These basic configurations will suffice for the lab. In a production setting, you will likely want to launch this [notebook into a VPC](#) for better network security. [Life-cycle configurations](#) also come in handy if you like to automatically bootstrap your notebook instances with packages that aren't already pre-installed by default.

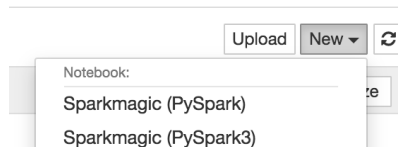
Click on **"Create notebook instance"** to launch your instance.



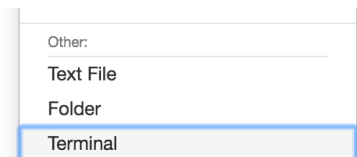
13. It will take about 5 minutes before your notebook is **InService**. Once it is, click on the **"Open Jupyter"** link.



14. Select the **"New"** drop down on the right hand side of the Jupyter admin console.



Scroll to the bottom and select **Terminal**.



15. We're going to create an S3 bucket from the terminal. The AWS CLI has been pre-installed, and it inherits the IAM permissions of the role that you attached to the instance.

Run the following command and replace the parts that are **high-lighted in red** with appropriate values. First, your bucket needs a **unique name**. Secondly, you need to create your bucket in the **same region as your notebook instance**. The example below will create the bucket in Oregon (us-west-2).

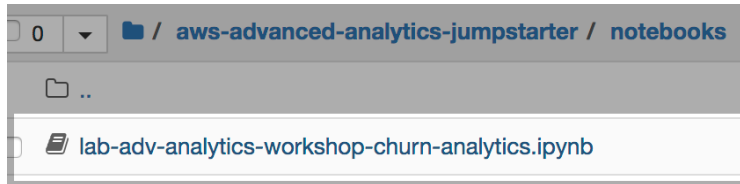
```
aws s3api create-bucket --bucket dtong-jumpstarter-workshop --region us-west-2 --  
create-bucket-configuration LocationConstraint=us-west-2
```

The output should look like the following:

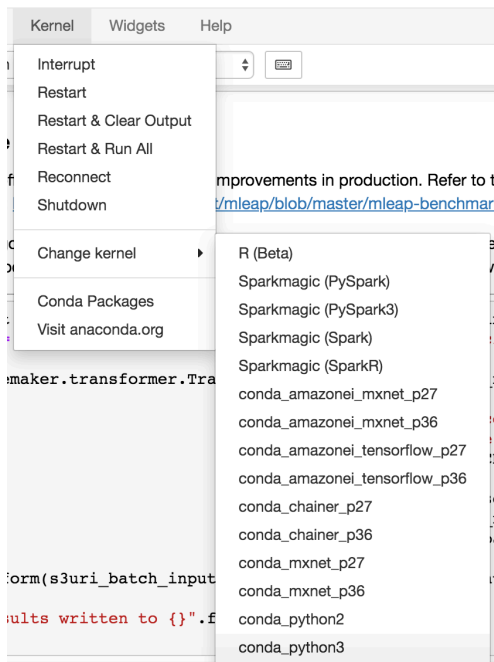

```
sh-4.2$ aws s3api create-bucket --bucket dtong-cv-jumpstarter-workshop --region us-west-2 --create-bucket-configuration LocationConstraint=us-west-2
{
  "Location": "http://dtong-cv-jumpstarter-workshop.s3.amazonaws.com/"
}
sh-4.2$
```

III. Build Your Churn Analytics Solution

1. Return to the Jupyter admin console, and launch the Jupyter notebook named “**lab-adv-analytics-workshop-churn-analytics.ipynb**” by clicking on it:



2. Once your notebook launches ensure that the Kernel is set to **conda_python3**:



3. Follow the instructions provided in the notebook. You'll accomplish the following learning objectives:
 - Learn how to query ground truth data from our data warehouse into a pandas data frame for exploration and feature engineering.
 - Train an XGBoost model to perform churn prediction.
 - Learn how to run a Batch Transform job to calculate churn scores in batch.

- Run a Glue job programmatically to demonstrate data processing and feature engineering at scale using SparkML.
- Create a production-scale inference pipeline that consists of a SparkML feature engineering pipeline that feeds into an XGBoost churn classification model.