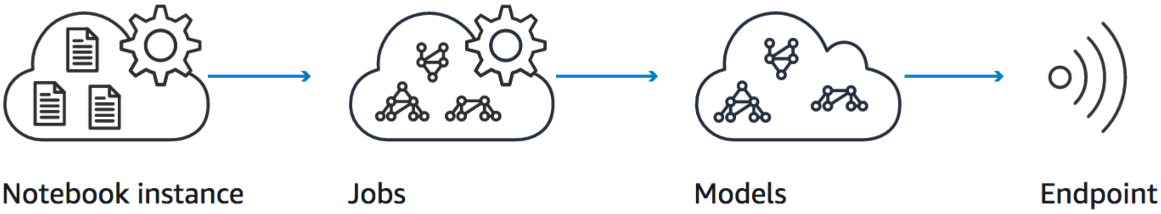
Automatic Labelling and Model Tuning with Amazon SageMaker

Developing machine learning models requires a lot of effort which often needs to be repeated over time as data distributions change. In this session you will learn about some of the latest concepts in Automatic Machine Learning including how to apply them to speed up development and achieve robust models over time. You will learn how to run a custom labelling job using Amazon SageMaker GroundTruth to build a data set to fine-tune your model. You will also learn how to tune your model's hyperparameters using SageMaker's Automatic Model Tuning capabilities and understand the theory of how Bayesian Optimisation is automatically applied for more accurate results and faster tuning.



Prerequisites

AWS Account

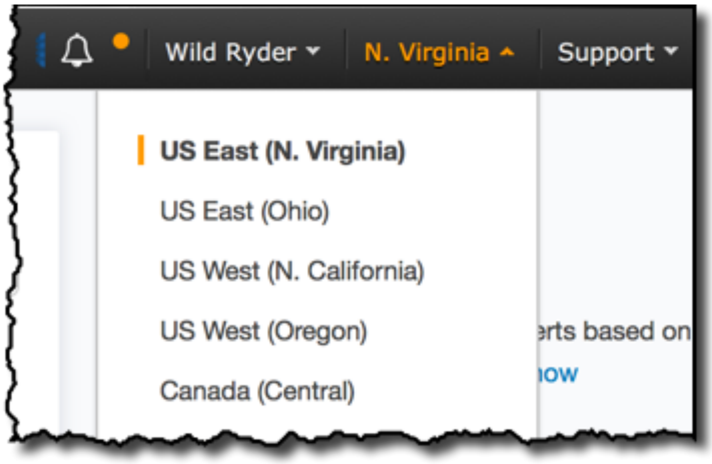
In order to complete this workshop you'll need an AWS Account with access to create AWS IAM, S3 and SageMaker resources. The code and instructions in this workshop assume only one student is using a given AWS account at a time. If you try sharing an account with another student, you'll run into naming conflicts for certain resources. You can work around these by appending a unique suffix to the resources that fail to create due to conflicts, but the instructions do not provide details on the changes required to make this work.

Some of the resources you will launch as part of this workshop are eligible for the AWS free tier if your account is less than 12 months old. See the AWS Free Tier page for more details. For other resources AWS credit vouchers are supplied.

AWS Region

SageMaker is not available in all AWS Regions at this time. Accordingly, we recommend running this workshop in one of the following supported AWS Regions: N. Virginia, Oregon, Ohio, or Ireland.

Once you've chosen a region, you should create all of the resources for this workshop there, including a new Amazon S3 bucket and a new SageMaker notebook instance. Make sure you select your region from the dropdown in the upper right corner of the AWS Console before getting started.



Browser

We recommend you use the latest version of Chrome or Firefox to complete this workshop.

Modules

This workshop is divided into multiple labs. Lab 1 must be completed first, followed by Lab 2.

1. Automatic Labelling
2. Private Workforce Labelling
3. Automatic Model Tuning

Be patient as you work your way through the notebook-based modules. After you run a cell in a notebook, it may take several seconds for the code to show results. For the cells that start training jobs, it may take several minutes. In particular, the last two modules have training jobs that may last up to 10 minutes.

After you have completed the workshop, you can delete all of the resources that were created by following the Cleanup Guide provided with this lab guide.

Lab 1: Automatic Model Tuning with Amazon Sagemaker GroundTruth

In this module we'll start by requesting a limit increase for ml.p3.2xlarge instances with can optionally be used at the end of the lab for training an object detection model. Then we will create an Amazon S3 bucket that will be used throughout the workshop. We'll then create a SageMaker notebook instance, which we will use to run the other workshop modules.

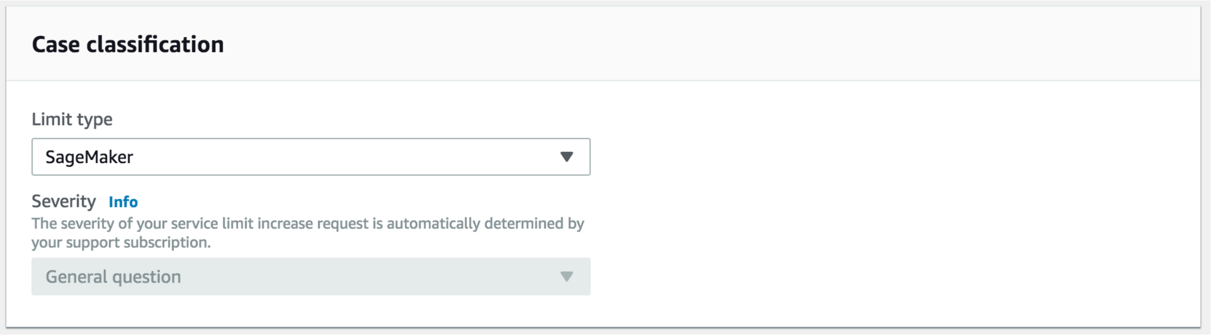
To begin, sign into the AWS Management Console, https://console.aws.amazon.com/.

1. Requesting Limit Increase

Amazon SageMaker by default limits access to GPU instances. Our first task is to use the console to request a limit increase:

1. Open the AWS Support Center page by clicking in the top right of the console, sign in if necessary, and choose **Create case**.

2. Choose **Service limit increase**.
3. Under Limit type select ‘SageMaker’



Case classification

Limit type

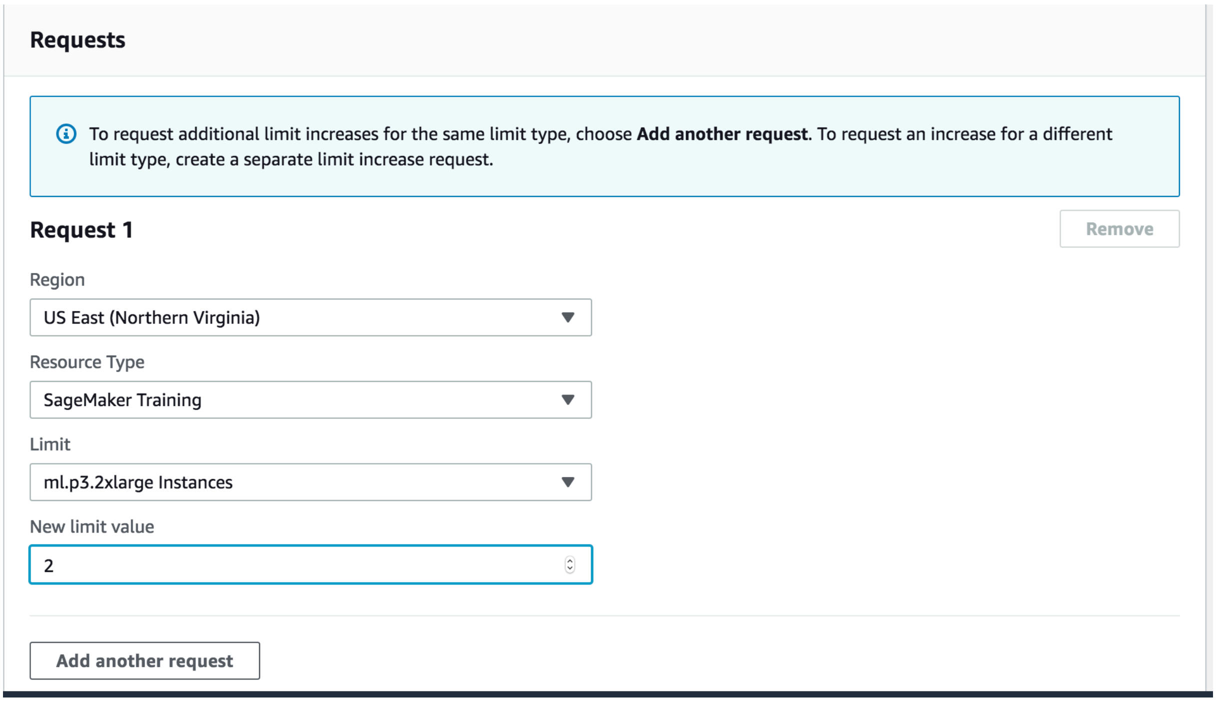
SageMaker ▼

Severity Info

The severity of your service limit increase request is automatically determined by your support subscription.

General question ▼

4. Select the following under Requests (with your region of choice for this workshop):



Requests

To request additional limit increases for the same limit type, choose **Add another request**. To request an increase for a different limit type, create a separate limit increase request.

Request 1

Region

US East (Northern Virginia) ▼

Resource Type

SageMaker Training ▼

Limit

ml.p3.2xlarge Instances ▼

New limit value

2

5. Under Use Case Description type ‘Urgent request for Sydney Summit workshop’.
6. Choose **Submit**.
7. This may come through before you reach the end of this lab, in which case you can train a model using the labelled images, otherwise come back to the training later.

2. Create a S3 Bucket

SageMaker typically uses S3 as storage for data and model artifacts. In this step you'll create a S3 bucket for this purpose.

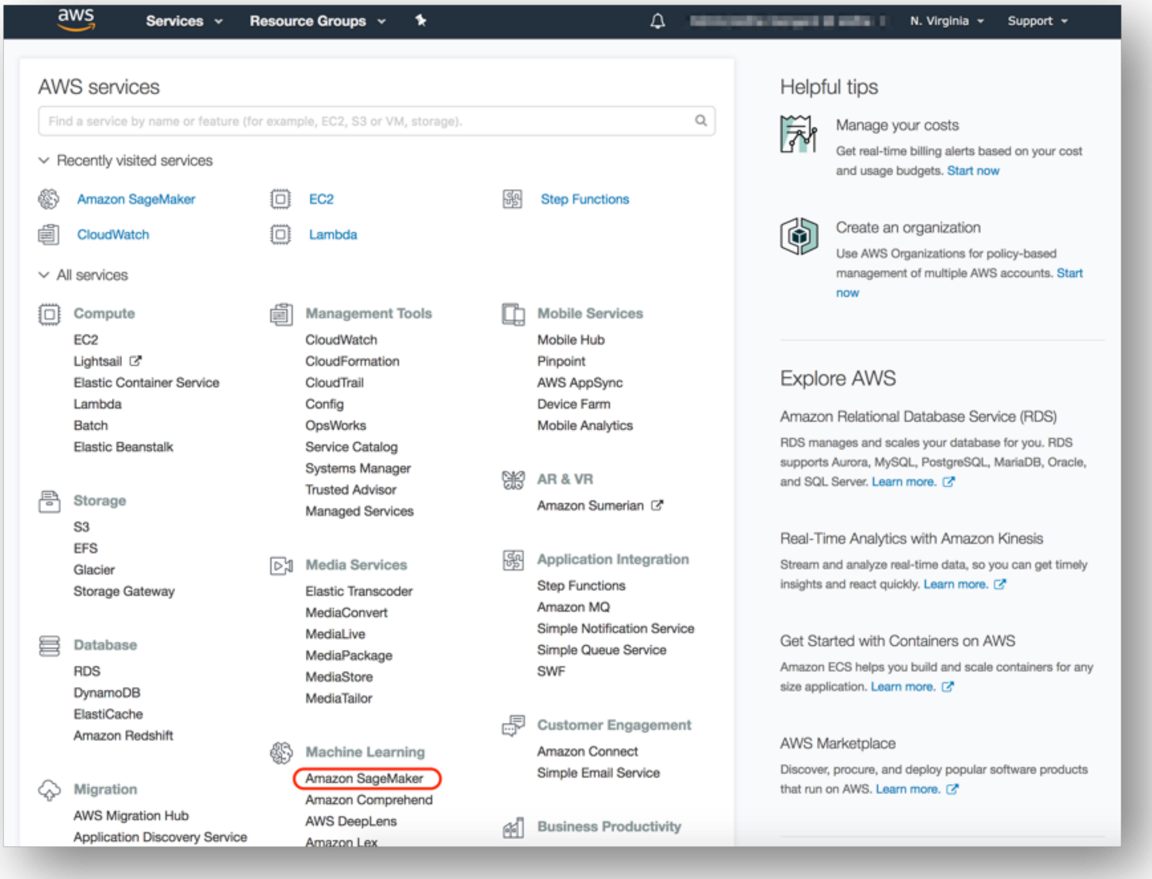
High-Level Instructions

Use the console or AWS CLI to create an Amazon S3 bucket. Keep in mind that your bucket's name must be globally unique across all regions and customers. We recommend using a name like `sagemakerworkshop-firstname-lastname`. If you get an error that your bucket name already exists, try adding additional numbers or characters until you find an unused name.

Step-by-step instructions (expand for details)

3. Launching the Notebook Instance

1. In the upper-right corner of the AWS Management Console, confirm you are in the desired AWS region. Select N. Virginia, Oregon, Ohio, or Ireland.
2. Click on Amazon SageMaker from the list of all services. This will bring you to the Amazon SageMaker console homepage.



Get real-time billing alerts based on your cost and usage budgets. Start now

Use AWS Organizations for policy-based management of multiple AWS accounts. Start now

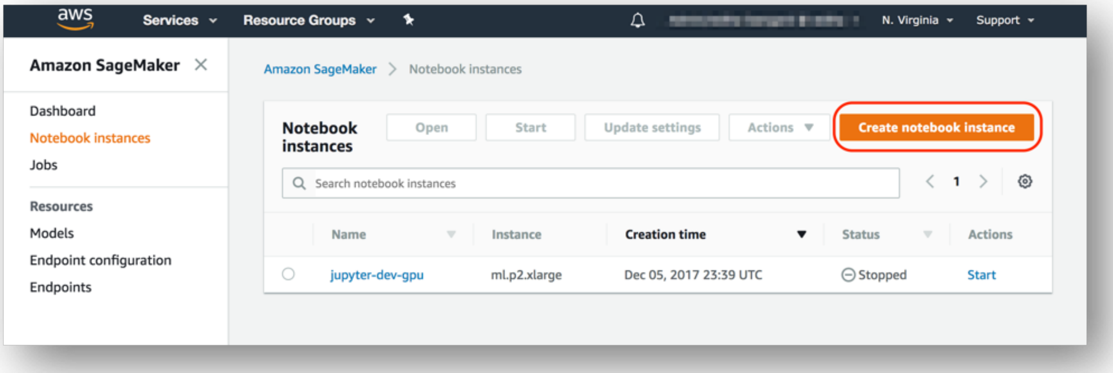
RDS manages and scales your database for you. RDS supports Aurora, MySQL, PostgreSQL, MariaDB, Oracle, and SQL Server. Learn more.

Stream and analyze real-time data, so you can get timely insights and react quickly. Learn more.

Amazon ECS helps you build and scale containers for any size application. Learn more.

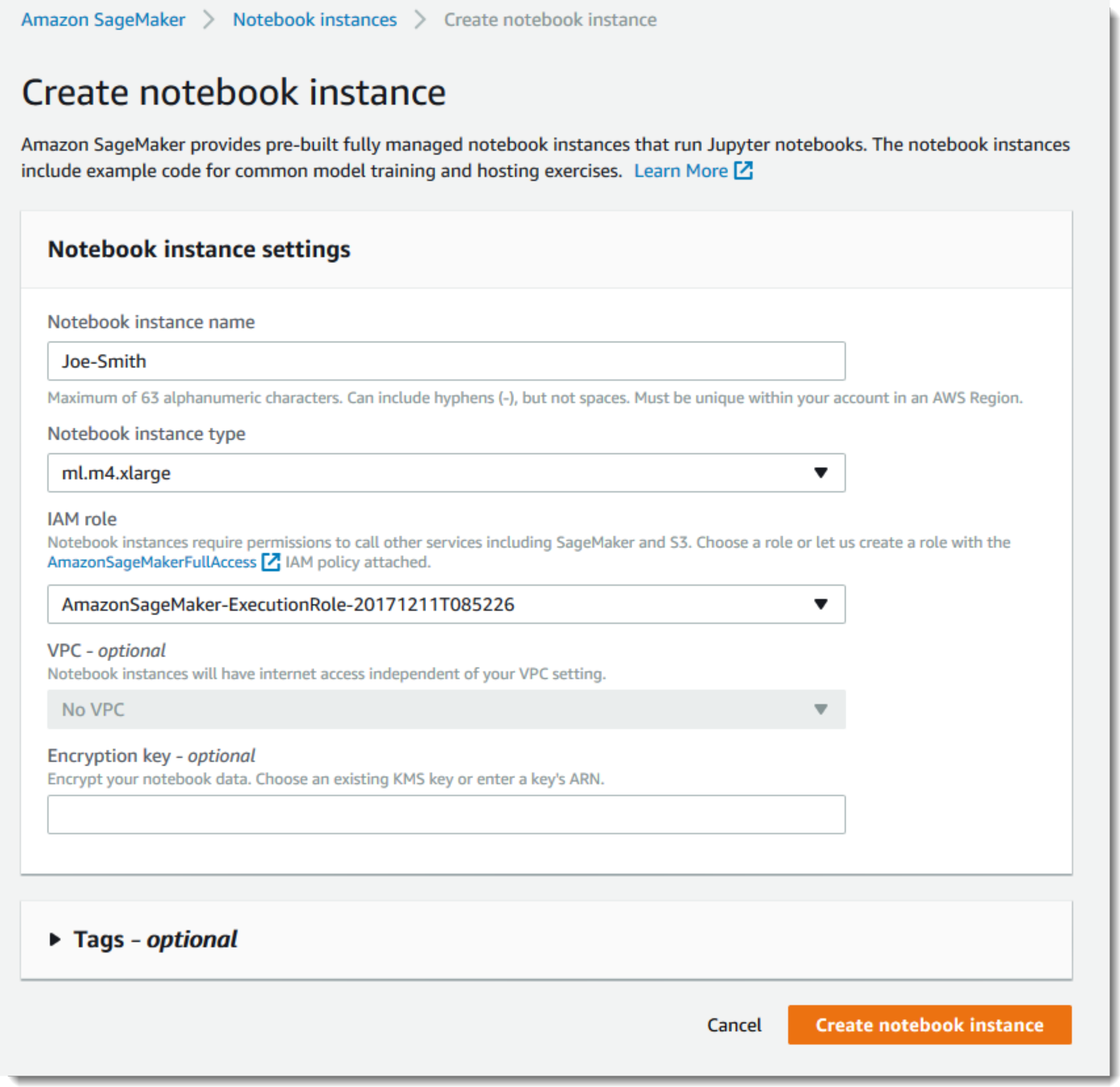
Discover, procure, and deploy popular software products that run on AWS. Learn more.

3. To create a new notebook instance, go to **Notebook instances**, and click the **Create notebook instance** button at the top of the browser window.



	Name	Instance	Creation time	Status	Actions
○	jupyter-dev-gpu	ml.p2.xlarge	Dec 05, 2017 23:39 UTC	⊖ Stopped	Start

4. Type [First Name]-[Last Name]-workshop into the **Notebook instance name** text box, and select ml.m4.xlarge for the **Notebook instance type**.
5. For IAM role, choose **Select an existing role** and choose one named "AmazonSageMaker-ExecutionRole-XXXX".


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

Joe-Smith

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.m4.xlarge

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-20171211T085226

VPC - *optional*

Notebook instances will have internet access independent of your VPC setting.

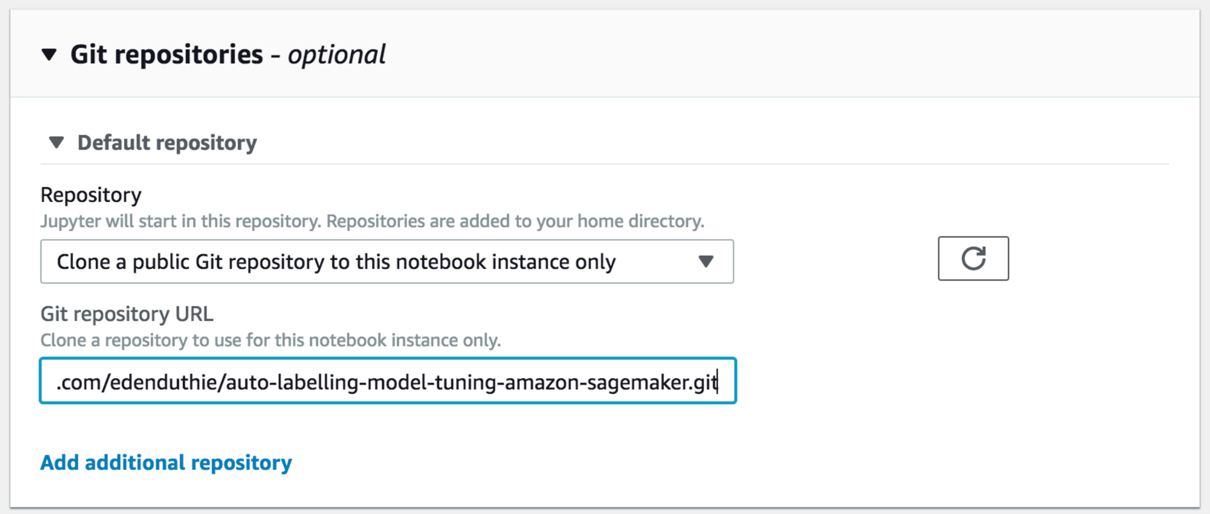
No VPC

Encryption key - *optional*

Encrypt your notebook data. Choose an existing KMS key or enter a key's ARN.

▸ Tags - *optional*

6. You can expand the "Tags" section and add tags here if required.
7. Under Git Repositories select ‘Clone a Git repository to this notebook instance only’
8. Enter **https://github.com/edenduthie/auto-labelling-model-tuning-amazon-sagemaker.git** under the ‘Git repository URL’. This will automatically clone the resources required for the workshop into the notebook instance.



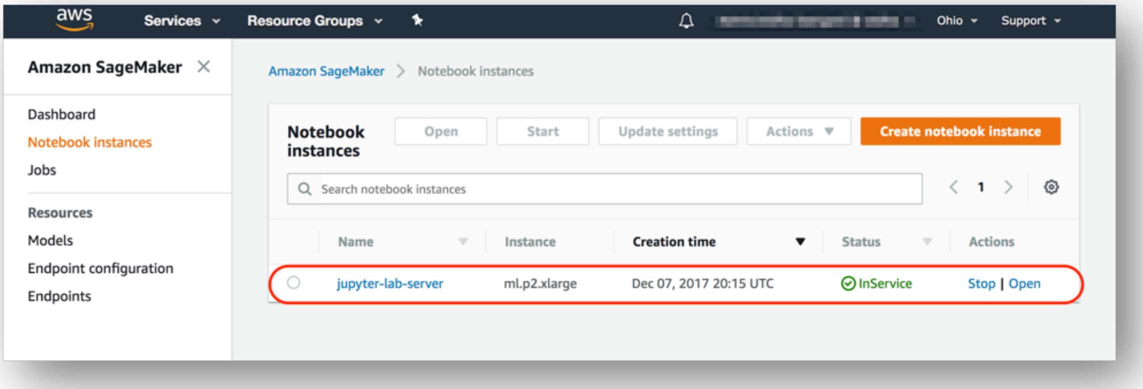
Repository
Jupyter will start in this repository. Repositories are added to your home directory.
Clone a public Git repository to this notebook instance only

Git repository URL
Clone a repository to use for this notebook instance only.
.com/edenduthie/auto-labelling-model-tuning-amazon-sagemaker.git

9. You will be taken back to the Create Notebook instance page. Click **Create notebook instance**. This will take several minutes to complete.

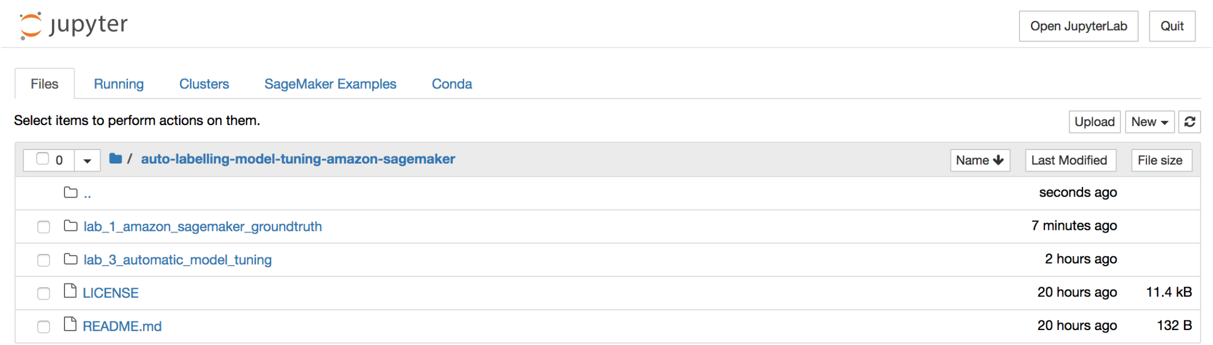
4. Accessing the Notebook Instance

1. Wait for the server status to change to **InService**. This will take a few minutes.



	Name	Instance	Creation time	Status	Actions
○	jupyter-lab-server	ml.p2.xlarge	Dec 07, 2017 20:15 UTC	InService	Stop | Open

2. Click **Open**. You will now see the Jupyter homepage for your notebook instance.



Select items to perform actions on them.

□ 0 / auto-labelling-model-tuning-amazon-sagemaker	Name	Last Modified	File size
..		seconds ago	
□ lab_1_amazon_sagemaker_groundtruth		7 minutes ago	
□ lab_3_automatic_model_tuning		2 hours ago	
□ LICENSE		20 hours ago	11.4 kB
□ README.md		20 hours ago	132 B

3. Click on the lab_1_amazon_sagemaker_groundtruth folder, and launch the **object_detection_tutorial.ipynb** notebook by double clicking on it.
4. If prompted Choose “Conda_Python3”

5. Running the Notebook

Run all of the cells in the “Introduction” and “Run a Ground Truth labeling job” sections of the notebook. You need to modify some of the cells, so read the notebook instructions carefully. Running these sections:

Creates a dataset with 10 images of birds
Creates object detection instructions for human annotators
Creates an object detection annotation job request
Submits the annotation job request to Ground Truth

The job should take about 45 min.

When it’s done, run all of the cells in the “Analyze Ground Truth labeling job results” and “Compare Ground Truth results to standard labels” sections. This produces a lot of information in plot form.

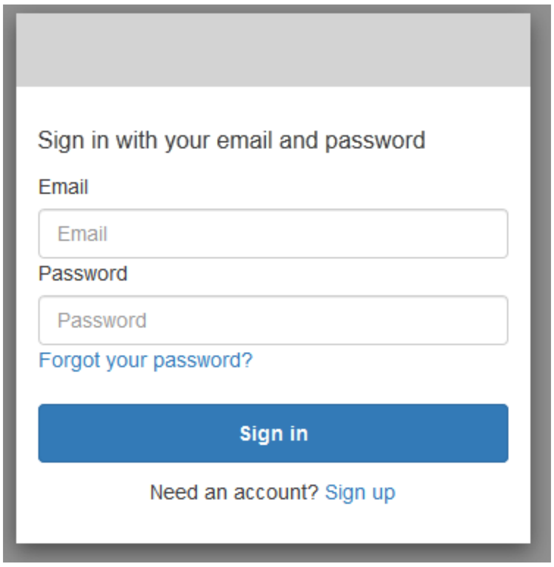
We will also demonstrate the results on the screen of a labelling job using 1,000 images. This takes approximately 4 hours to run. Optionally set RUN_FULL_AL_DEMO = False in the notebook to run this yourself after the workshop.

The labelling job itself will take about 15min as it is going out to a team of humans to label. While this is running follow the instructions in the notebook to create the same labelling job using the console to facilitate your understanding of the process.

Lab 2 – Sagemaker Ground Truth Private workforce

1. Login/signup to the workforce at:

https://jfagq60whn.labeling.us-east-1.sagemaker.aws/



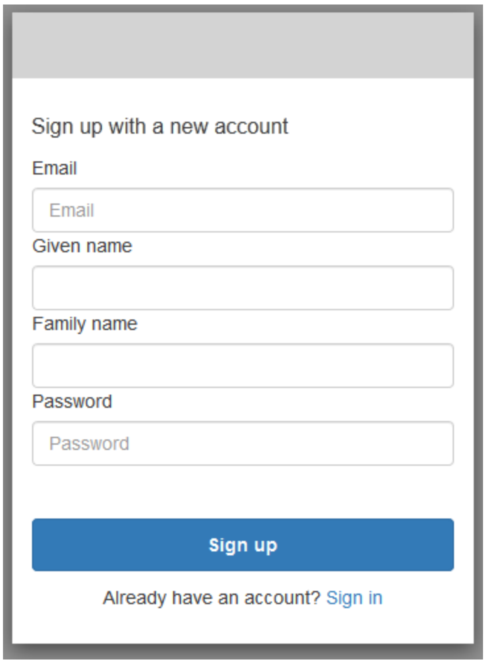
Sign in with your email and password

Email

Password

2. Choose "Sign Up"

Enter
Email
Name
Password
Verification code will be emailed



Sign up with a new account

Email

Given name

Family name

Password

3. Wait for Labeling jobs to come though.

Create a private team to test your task [OPTIONAL]

This step requires you to use the AWS Console. However, we **highly recommend** that you follow it, especially when creating your own task with a custom dataset, label set, and template.

We will create a `private workteam` and add only one user (you) to it. Then, we will modify the Ground Truth API job request to send the task to that workforce. You will then be able to see your annotation job exactly as the public annotators would see it. You could even annotate the whole dataset yourself!

To create a private team:

1. Go to `AWS Console > Amazon SageMaker > Labeling workforces`
2. Click "Private" and then "Create private team".
3. Enter the desired name for your private workteam.
4. Select "Create a new Amazon Cognito user group" and click "Create private team."
5. The AWS Console should now return to `AWS Console > Amazon SageMaker > Labeling workforces`.
6. Click on "Invite new workers" in the "Workers" tab.
7. Enter your own email address in the "Email addresses" section and click "Invite new workers."
8. Click on your newly created team under the "Private teams" tab.
9. Select the "Workers" tab and click "Add workers to team."
10. Select your email and click "Add workers to team."
11. The AWS Console should again return to `AWS Console > Amazon SageMaker > Labeling workforces`. Your newly created team should be visible under "Private teams". Next to it you will see an `ARN` which is a long string that looks like `arn:aws:sagemaker:region-name-123456:workteam/private-crowd/team-name`. Copy this ARN into the cell below.
12. You should get an email from `no-reply@verificationemail.com` that contains your workforce username and password.
13. In `AWS Console > Amazon SageMaker > Labeling workforces > Private`, click on the URL under `Labeling portal sign-in URL`. Use the email/password combination from the previous step to log in (you will be asked to create a new, non-default password).

That's it! This is your private worker's interface. When we create a verification task in Verify your task using a private team below, your task should appear in this window. You can invite your colleagues to participate in the labeling job by clicking the "Invite new workers" button.

The SageMaker Ground Truth documentation has more details on the management of private workteams.

Lab 3: Automatic Model Tuning with Amazon SageMaker

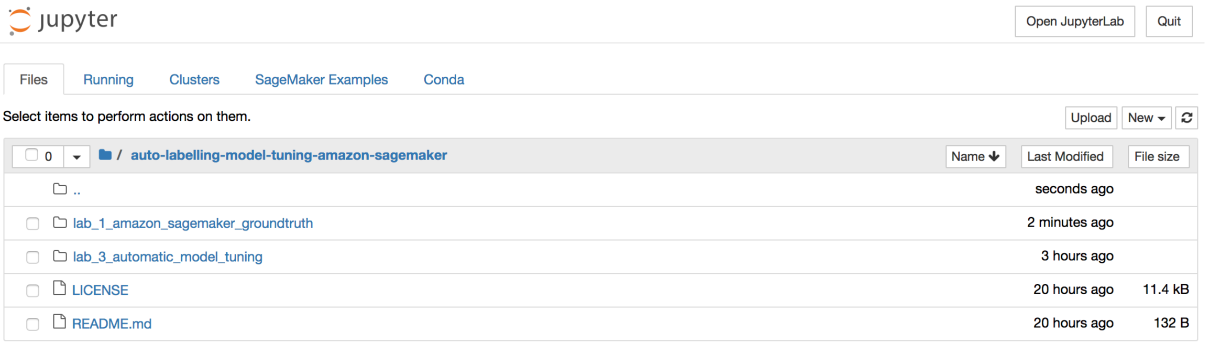
Selecting the right hyperparameter values for your machine learning model can be difficult. The right answer is dependent on your data; some algorithms have many different hyperparameters that can be tweaked; some are very sensitive to the hyperparameter values selected; and most have a non-linear relationship between model fit and hyperparameter values.

Amazon SageMaker Automatic Model Tuning helps with automating the hyperparameter tuning process. In many occasions the tuning process is iterative and requires to run multiple tuning jobs after analyzing the results to get the best objective metric.

Warm start configuration allows you to create a new tuning job with the learning gathered in a parent tuning job by specifying up to 5 parent tuning jobs. If a warm start configuration is specified, Automatic Model Tuning will load the previous hyperparameter set and objective metrics values to warm start the new tuning job. This means, you can continue optimizing your model from the point you finished your previous tuning job experiment.

This lab will train a model which can be used to predict if a customer will enroll for a term deposit at a bank, after one or more phone calls, based on data from the Bank Marketing data set. Hyperparameter tuning, leveraging the warm start feature, will be used in order to try multiple hyperparameter settings and produce the best model.

1. Launch the jupyter notebook interface of the notebook instance created for lab 1:



Select items to perform actions on them.

/ auto-labelling-model-tuning-amazon-sagemaker

Name	Last Modified	File size
..	seconds ago	
lab_1_amazon_sagemaker_groundtruth	2 minutes ago	
lab_3_automatic_model_tuning	3 hours ago	
LICENSE	20 hours ago	11.4 kB
README.md	20 hours ago	132 B

2. Click on the folder **lab_3_automatic_model_tuning** and open the automatic_model_tuning.ipynb notebook.
3. Read through the instructions and then execute all the cells of the notebook. Play close attention as there are sections where you will need to fill in some of the code.