Honorlock SageMaker Canvas and Inference Notes

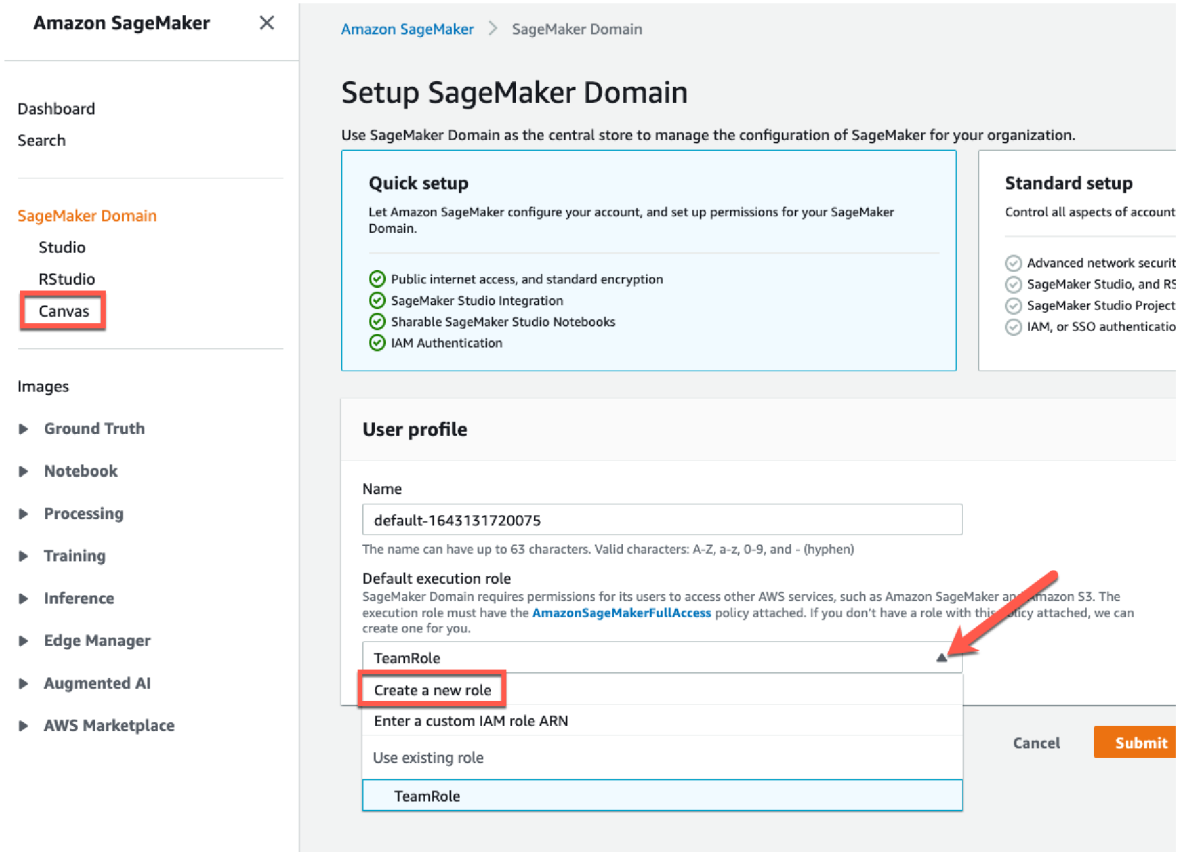
S3 Buckets

Create three s3 buckets, One to work with SageMaker Canvas, one for staging and one for tracking storing outputs.

Sagemaker Canvas

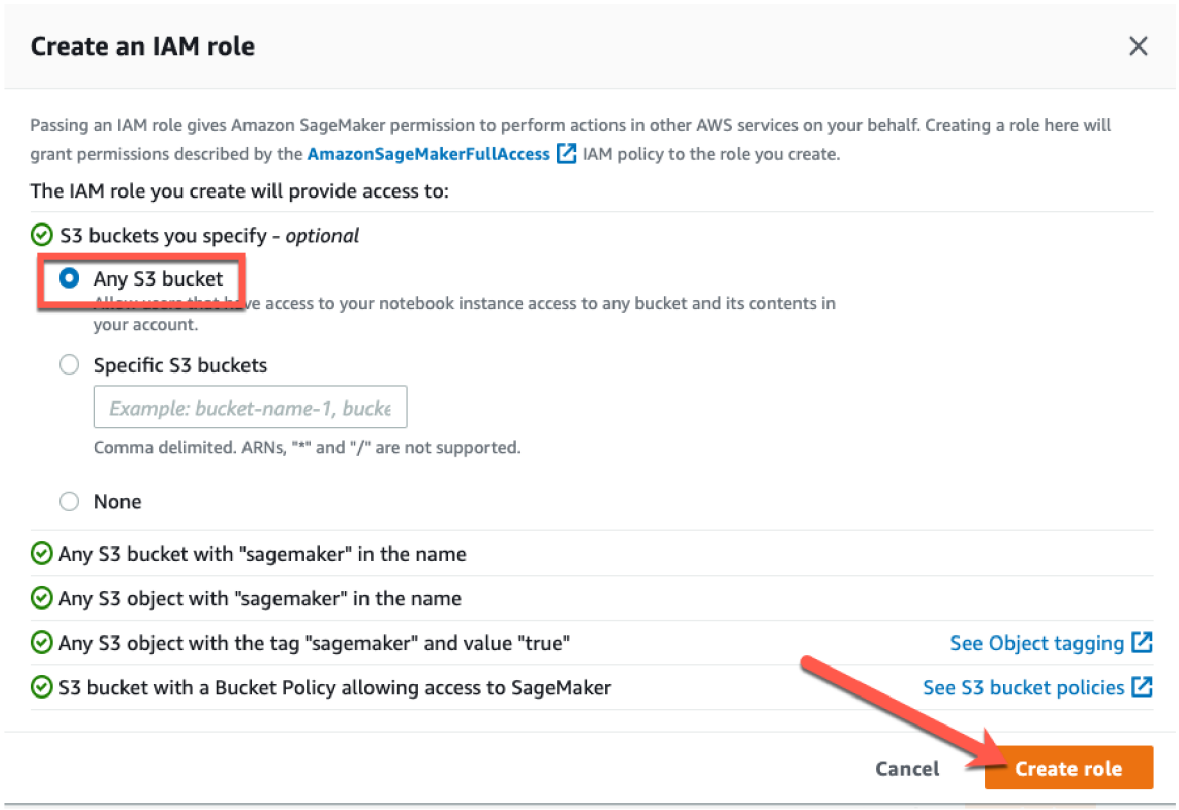
Activate canvas if not already done

Activating SageMaker Canvas

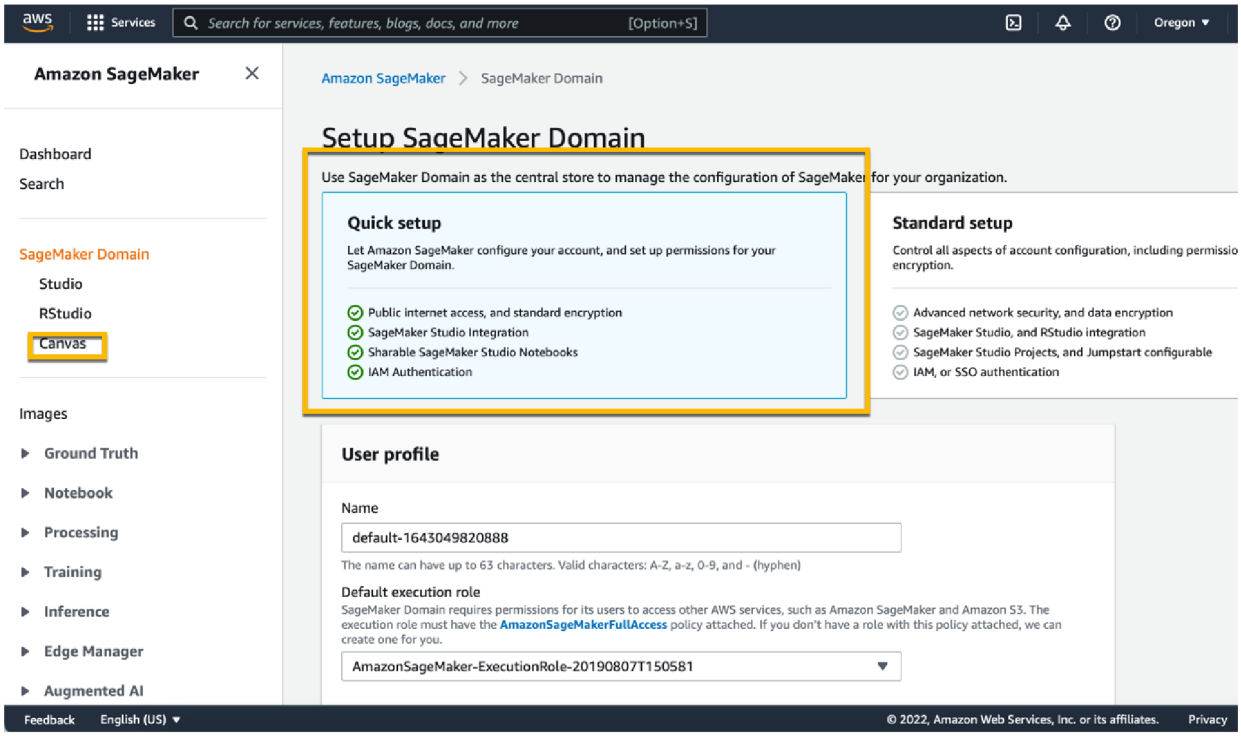
1. Go to Amazon SageMaker Console, select Canvas,
2. Click the drop down for the Default Execution Role and select Create a new role
3. 

4. Ensure that the Any S3 Bucket radio button is checked and click the Create role button

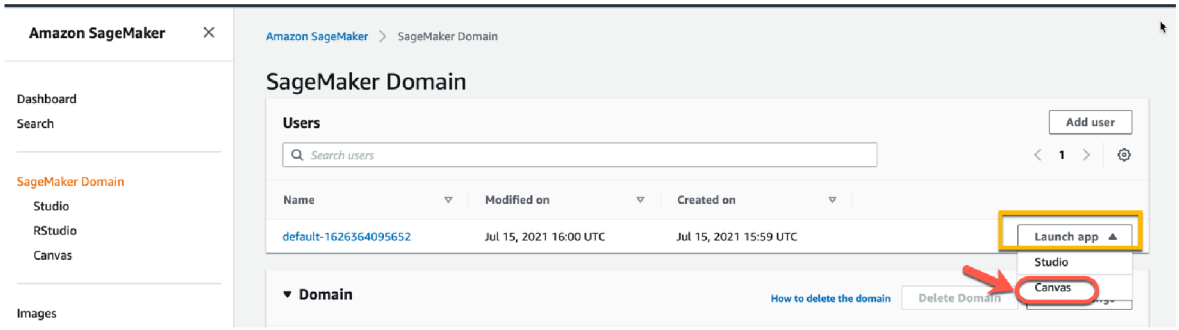
5.



6. Follow Quick setup prompts or Standard setup if you so choose.



Starting Canvas

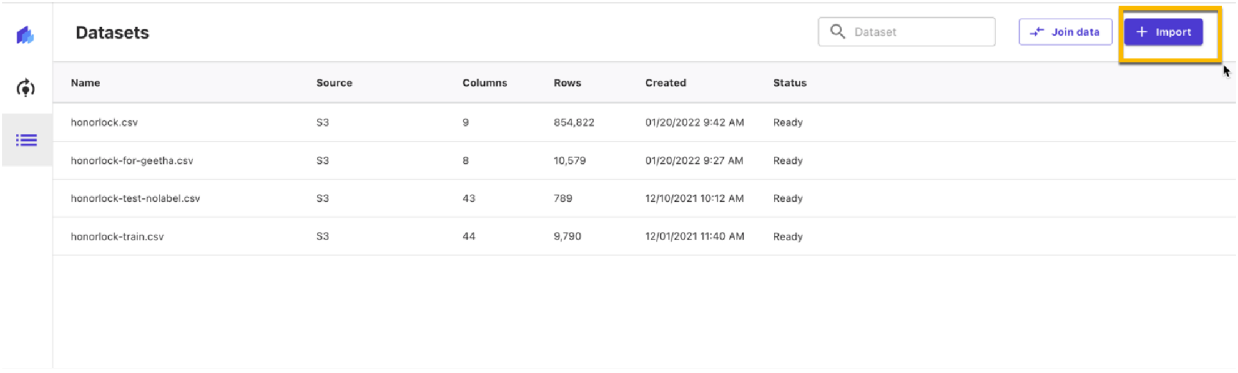
1. Once set up is complete, you can launch Canvas from the SageMaker Console as shown below
2. 

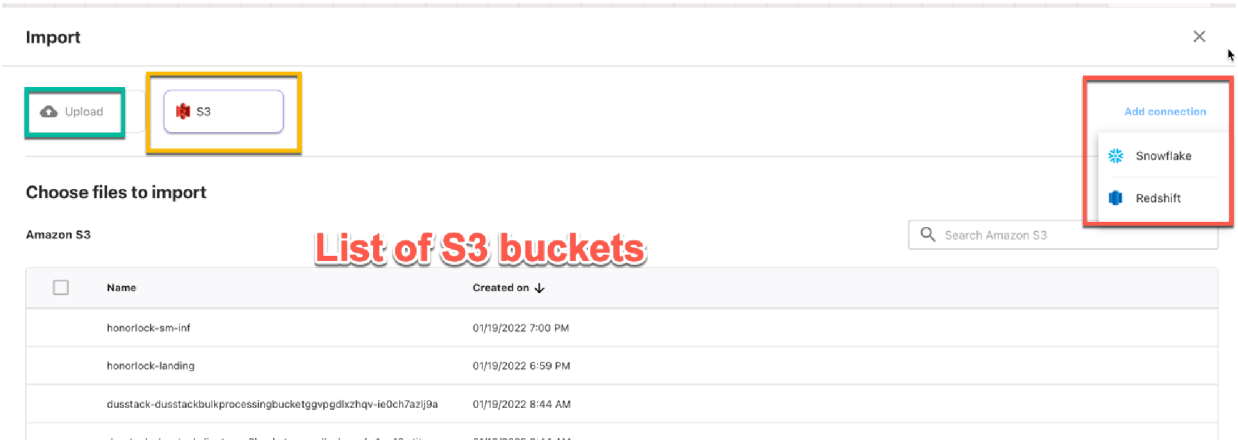
Get started with ML on Canvas

The process of ML with Canvas uses intuitive upload a dataset, examine and decide the columns and target, choose a ML model and then build and train a model followed by inference.

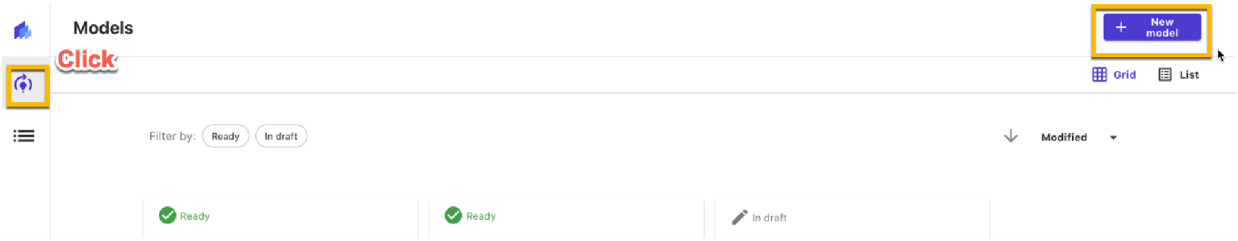
All steps can be carried out in Canvas, However this set up follows training models in Canvas and automating the inference using lambda functions.

1. Canvas allows you to upload datasets from S3 or local, It is recommended that you use S3 for staging your dataset

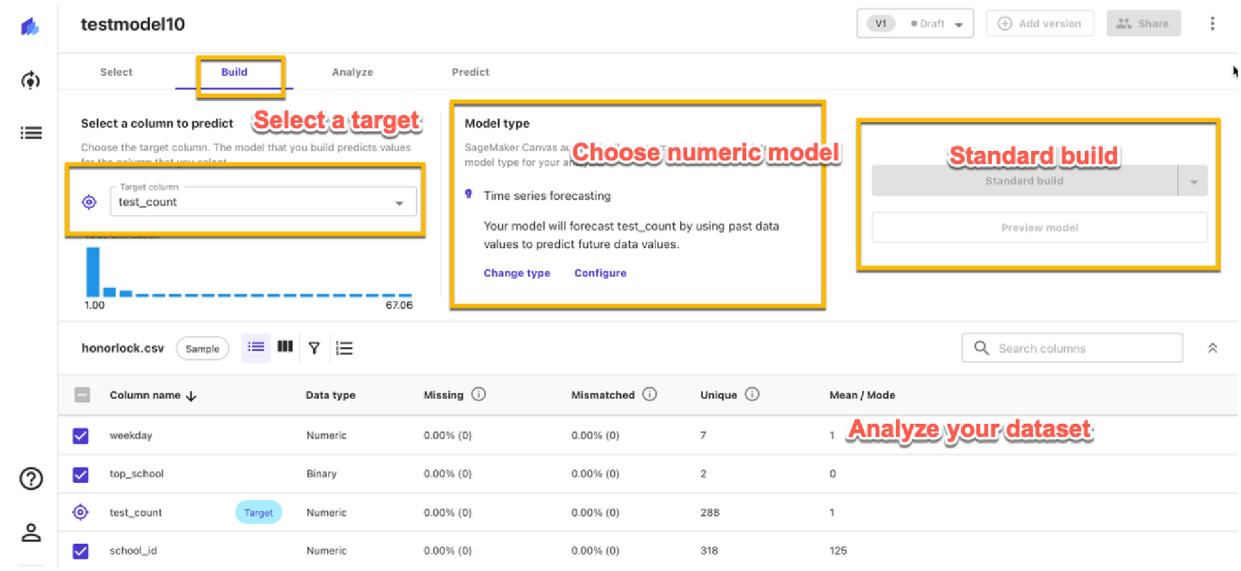


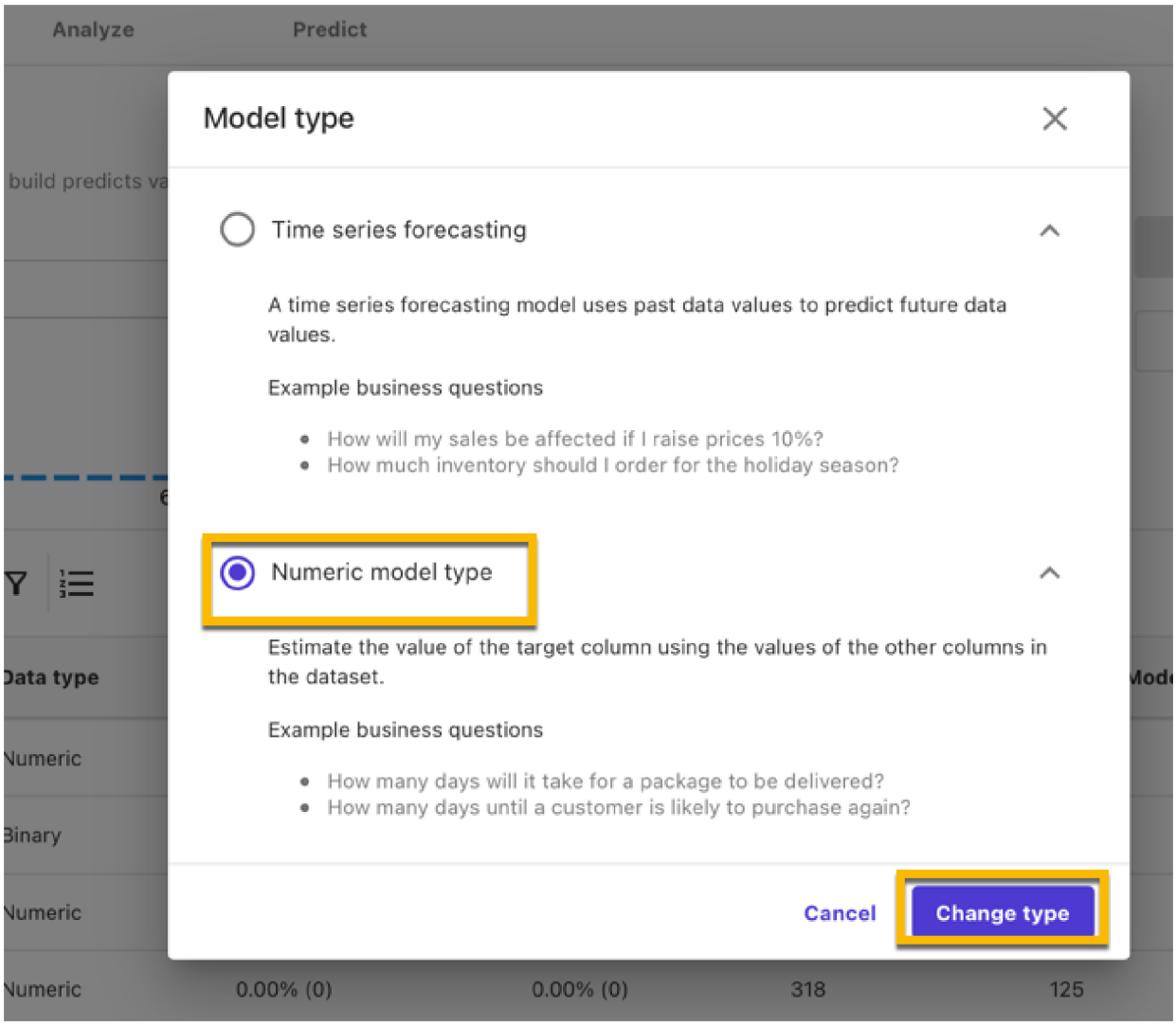


2. Multiple datasets can be joined directly inside canvas, for example holiday dataset can be joined with your dataset on dates
3. the next step is examining the data and looking for any inconsistencies. we do this in model building stage b

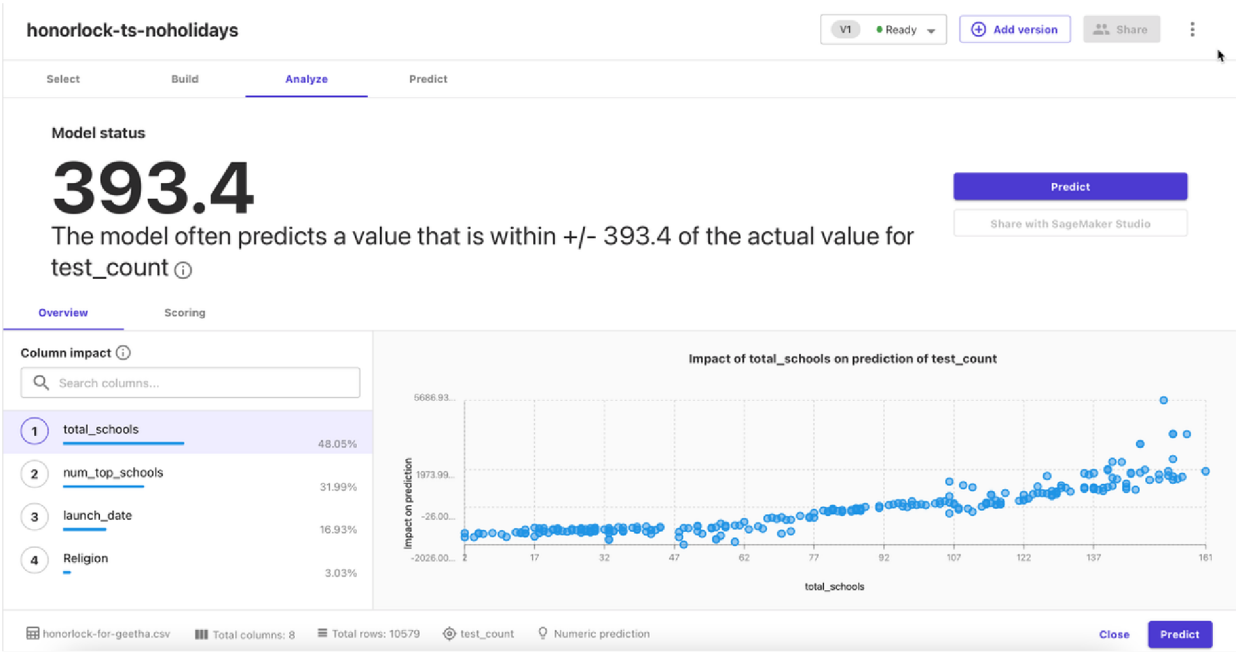


4. click on New Model and create a new model, name it and click create.follow the prompts and select the dataset you want to use
5. The next screen is where you can examine the dataset, select a target, type of model and start a build


testmodel10
Select
Build
Analyze
Predict
Select a column to predict
Select a target
Target column
test_count
Model type
Choose numeric model
Time series forecasting
Your model will forecast test_count by using past data values to predict future data values.
Change type
Configure
Standard build
Standard build
Preview model
honorlock.csv
Sample
Search columns
Column name
Data type
Missing
Mismatched
Unique
Mean / Mode
weekday Numeric 0.00% (0) 0.00% (0) 7 1
top_school Binary 0.00% (0) 0.00% (0) 2 0
test_count Target Numeric 0.00% (0) 0.00% (0) 288 1
school_id Numeric 0.00% (0) 0.00% (0) 318 125
Analyze your dataset


Analyze
Predict
Model type
Time series forecasting
A time series forecasting model uses past data values to predict future data values.
Example business questions
How will my sales be affected if I raise prices 10%?
How much inventory should I order for the holiday season?
Numeric model type
Estimate the value of the target column using the values of the other columns in the dataset.
Example business questions
How many days will it take for a package to be delivered?
How many days until a customer is likely to purchase again?
Cancel
Change type
Numeric 0.00% (0) 0.00% (0) 318 125

6. Once model type is selected, click on standard build and let the model build complete. You can see the model analysis such as influencing columns and their contribution allowing you to explain the model.



7. At this point we exit canvas and go to SageMaker Console. go to Training and Hyperparameter Tuning Jobs and find the Canvas job that just ran.

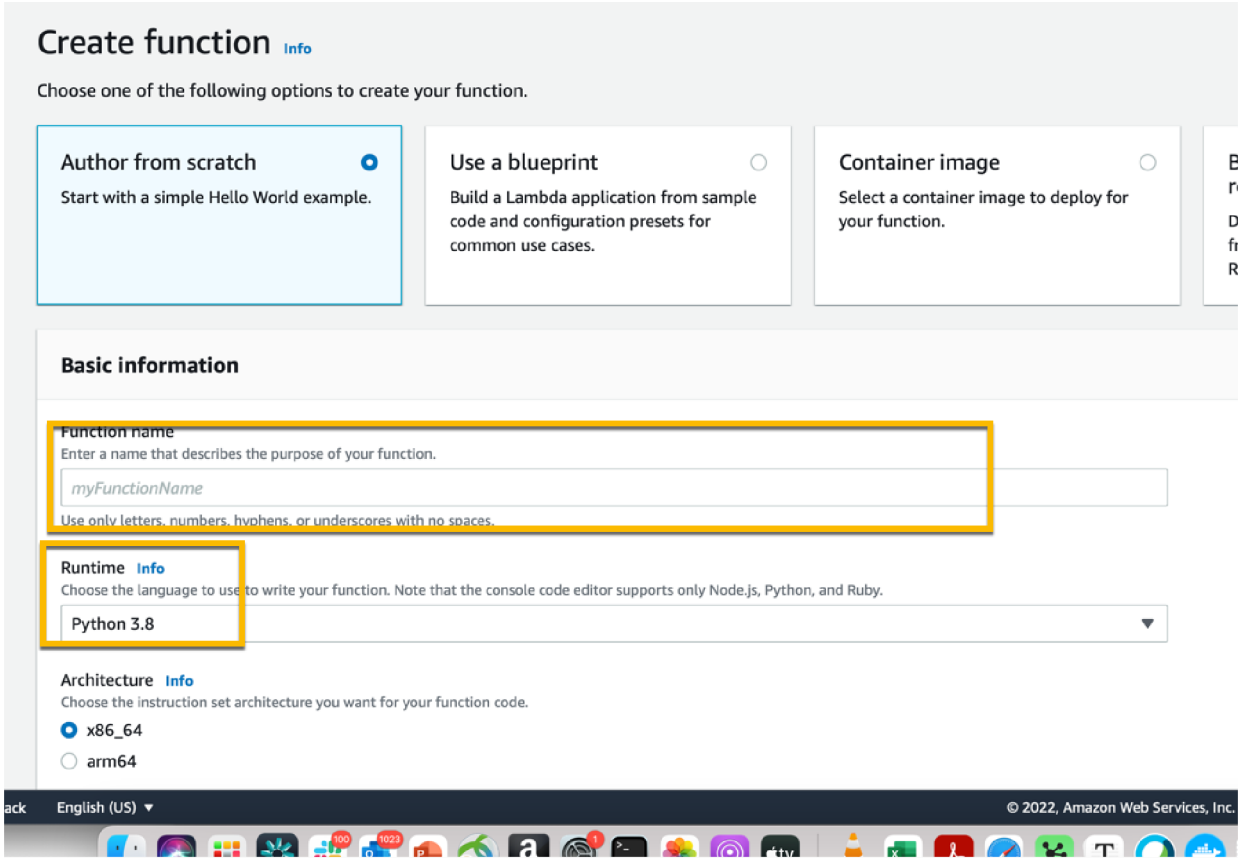
Inference With SageMaker Batch.

We will create 2 lambda functions, a SNS topic and provision IAM permissions to lambda functions.

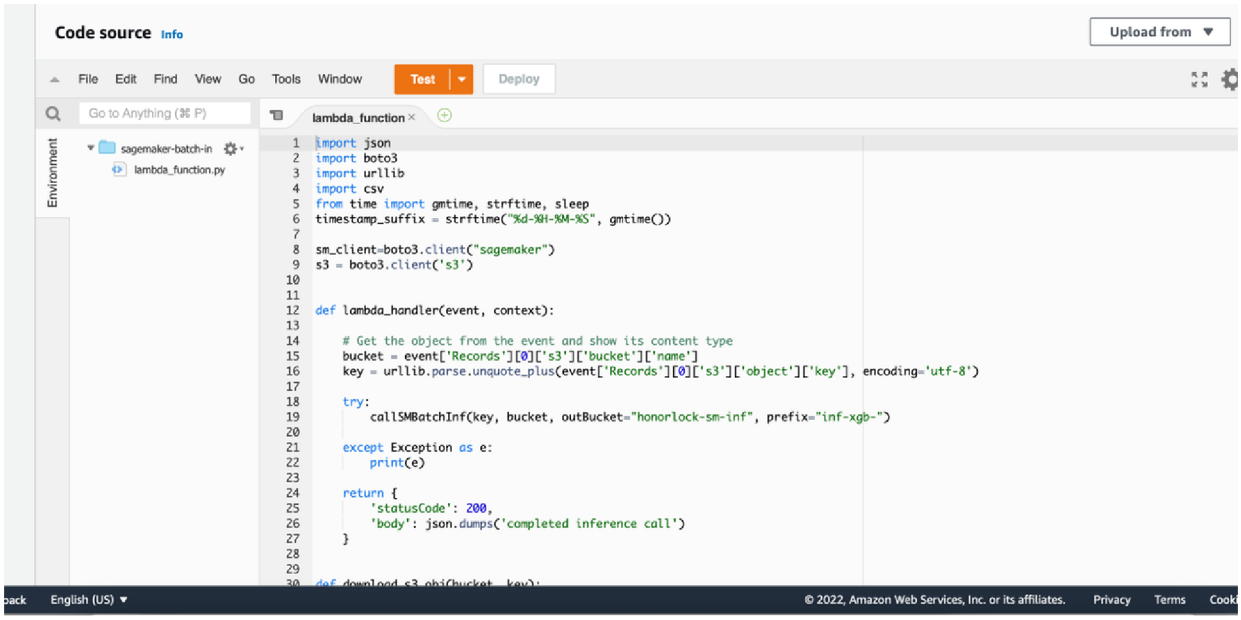
Lambda Functions

Lambda Function for starting SageMaker Batch Inference

1. go to lambda console on aws console and create a new lambda function, select author your own and for language select python3.8

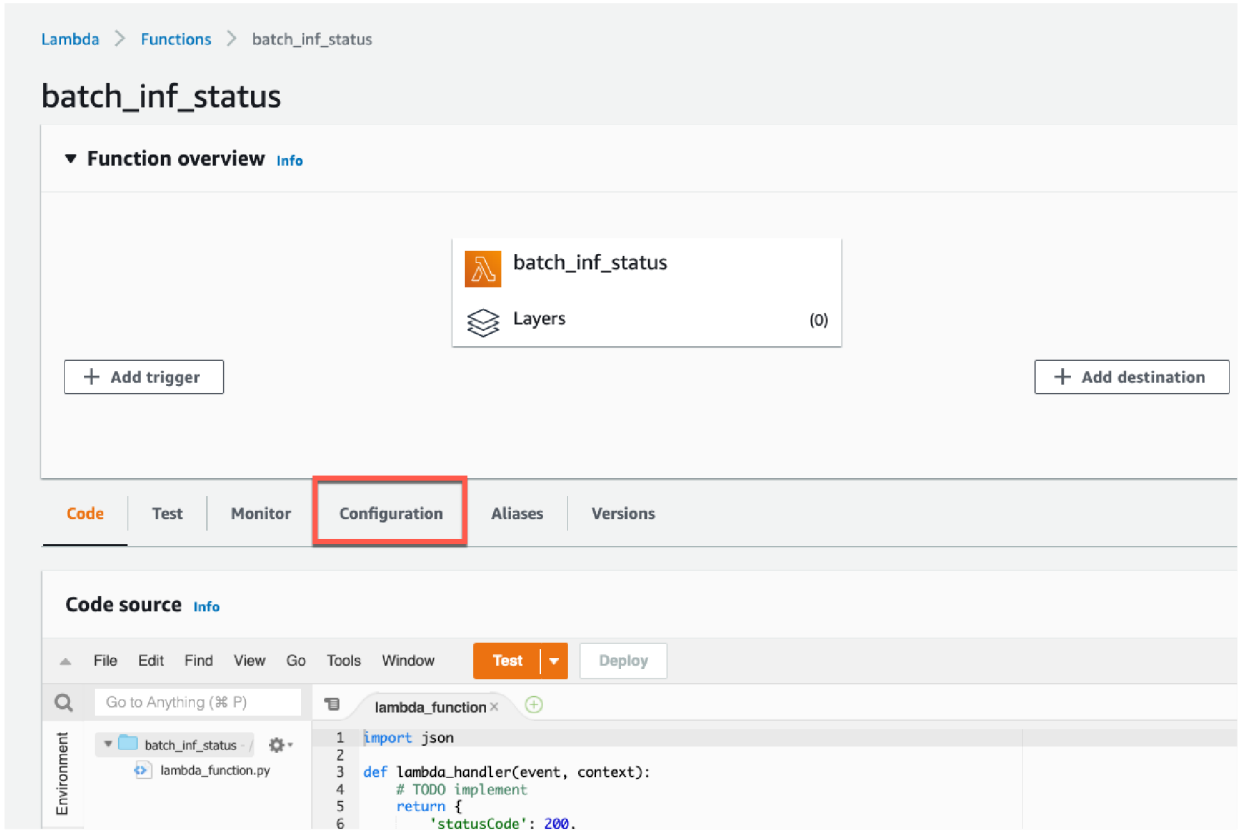


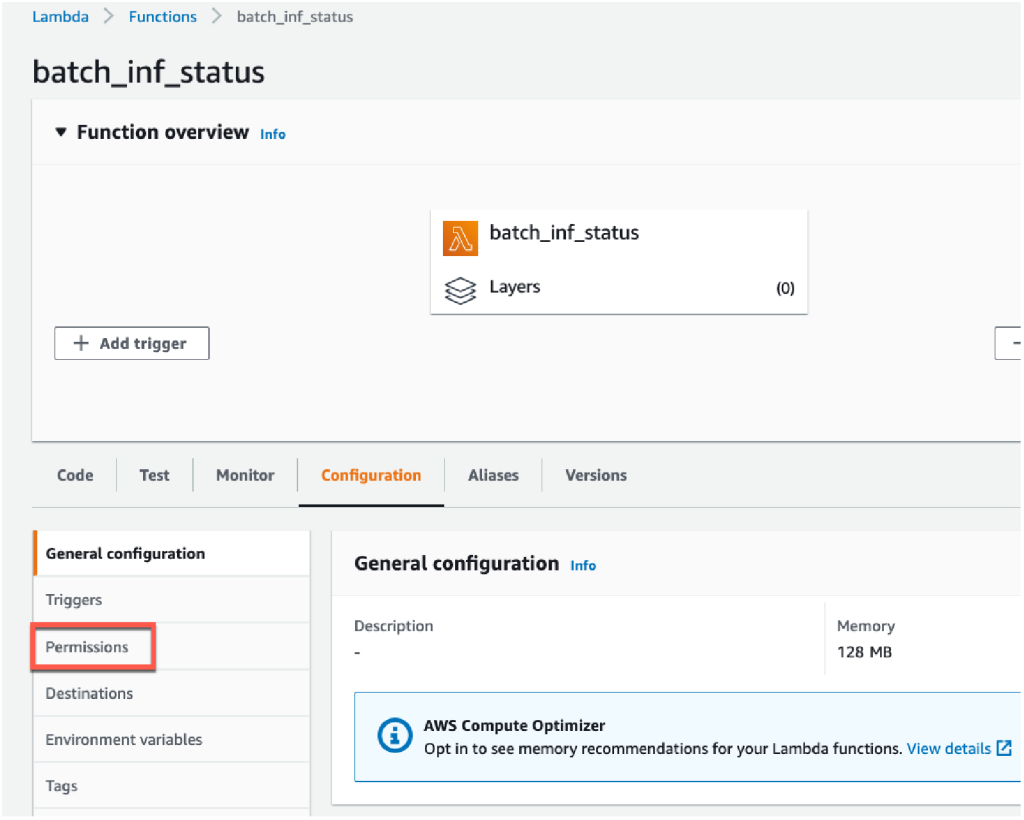
2. Once the function is created, go to code section of the lambda function and paste in the code sm-batch-inf.py
3. click File-->save.
4. Look at the code and change any setting , like naming conventions, logic to use the latest model as you see fit, One way to decouple that from code would be to have a config.json file stored in S3 that can be updated without touching the code. The config file can contain output file naming, model to use for inference or any other variables.

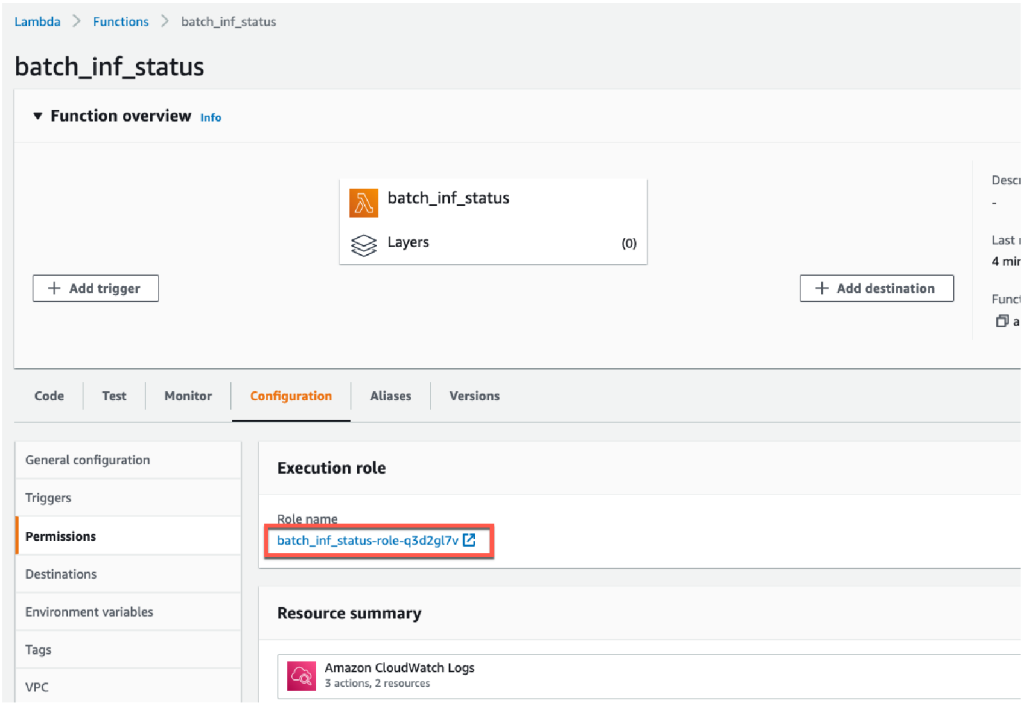


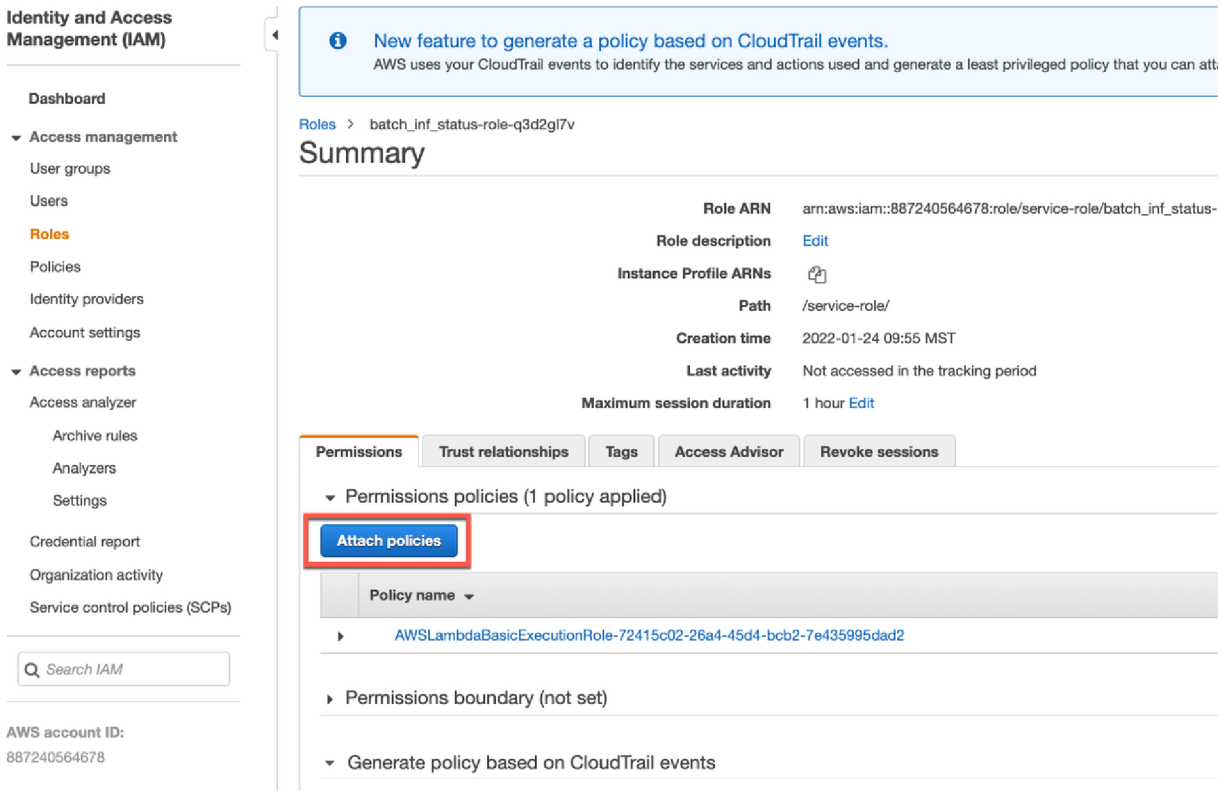
5. Next add a trigger to Lambda function, Choose one of the S3 buckets assigned as landing bucket for inference input.

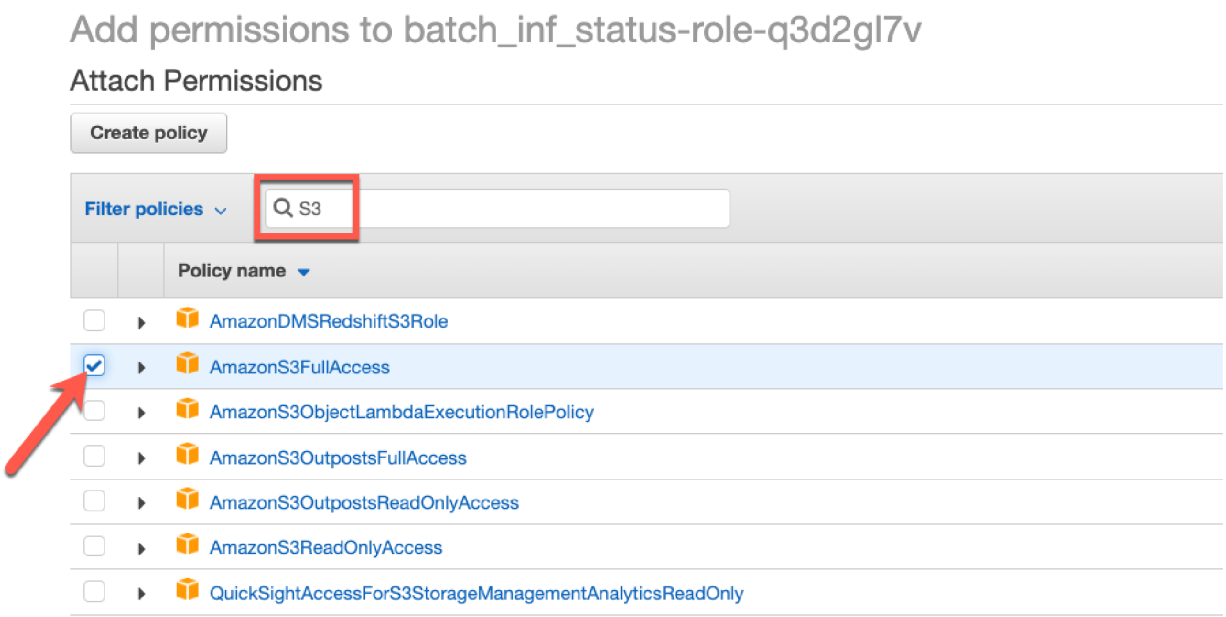
6. Next we will update permissions so lambda can access Sagemaker and S3.

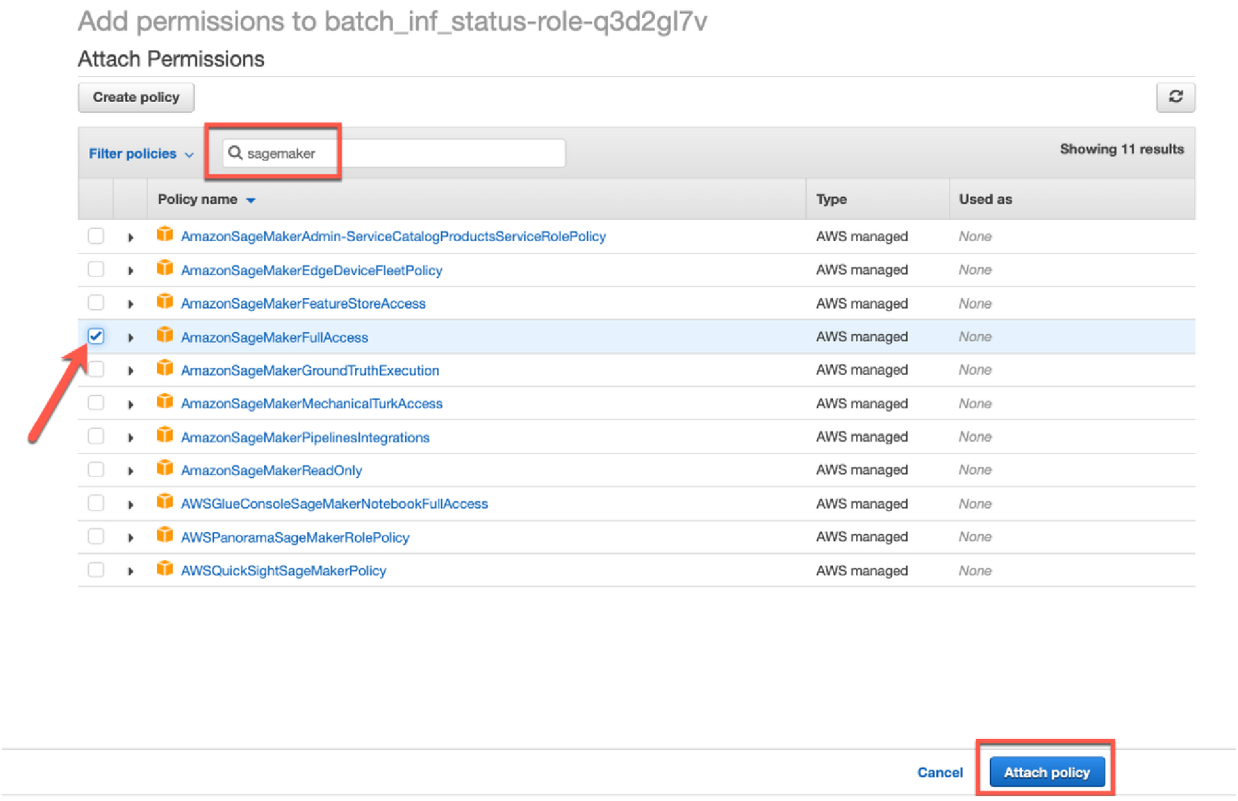



Lambda > Functions > batch_inf_status
batch_inf_status
Function overview Info
batch_inf_status
Layers (0)
+ Add trigger
Code
Test
Monitor
Configuration
Aliases
Versions
General configuration
Triggers
Permissions
Destinations
Environment variables
Tags
General configuration Info
Description
-
Memory
128 MB
AWS Compute Optimizer
Opt in to see memory recommendations for your Lambda functions. View details


Lambda > Functions > batch_inf_status
batch_inf_status
Function overview Info
batch_inf_status
Layers (0)
+ Add trigger
+ Add destination
Code
Test
Monitor
Configuration
Aliases
Versions
General configuration
Triggers
Permissions
Destinations
Environment variables
Tags
VPC
Execution role
Role name
batch_inf_status-role-q3d2gl7v
Resource summary
Amazon CloudWatch Logs
3 actions, 2 resources


Identity and Access Management (IAM)
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access analyzer
Archive rules
Analyzers
Settings
Credential report
Organization activity
Service control policies (SCPs)
Search IAM
AWS account ID:
887240564678
New feature to generate a policy based on CloudTrail events.
AWS uses your CloudTrail events to identify the services and actions used and generate a least privileged policy that you can att
Roles > batch_inf_status-role-q3d2gl7v
Summary
Role ARN
arn:aws:iam::887240564678:role/service-role/batch_inf_status-
Role description
Edit
Instance Profile ARNs
Path
/service-role/
Creation time
2022-01-24 09:55 MST
Last activity
Not accessed in the tracking period
Maximum session duration
1 hour Edit
Permissions
Trust relationships
Tags
Access Advisor
Revoke sessions
Permissions policies (1 policy applied)
Attach policies
Policy name
AWSLambdaBasicExecutionRole-72415c02-26a4-45d4-bcb2-7e435995dad2
Permissions boundary (not set)
Generate policy based on CloudTrail events


Add permissions to batch_inf_status-role-q3d2gl7v
Attach Permissions
Create policy
Filter policies
S3
Policy name
AmazonDMSRedshiftS3Role
AmazonS3FullAccess
AmazonS3ObjectLambdaExecutionRolePolicy
AmazonS3OutpostsFullAccess
AmazonS3OutpostsReadOnlyAccess
AmazonS3ReadOnlyAccess
QuickSightAccessForS3StorageManagementAnalyticsReadOnly



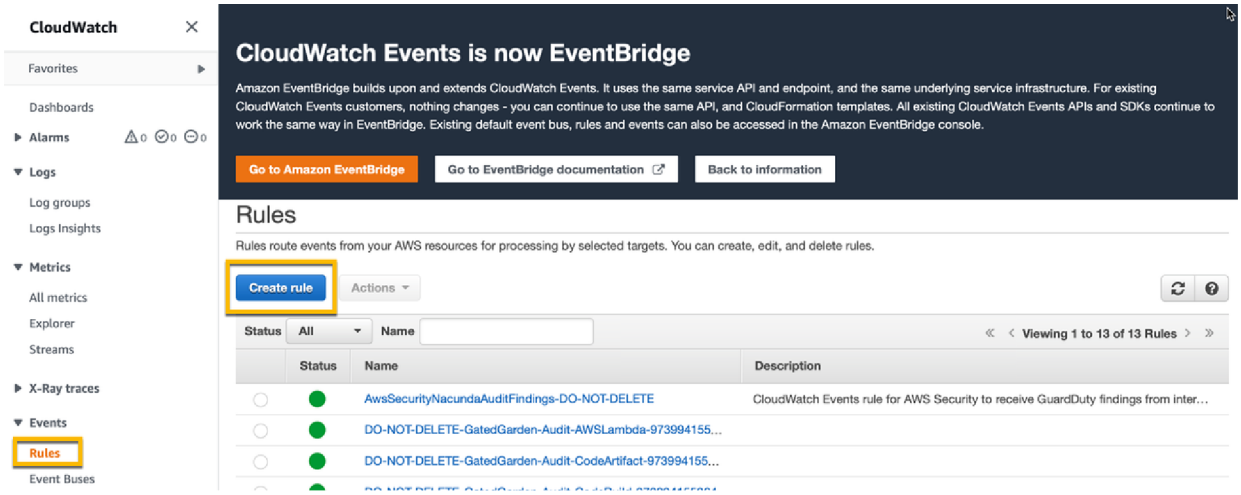
The Lambda function is now ready to react to a file drop in S, look at the file for headers, remove if there is any and call SageMaker Batch Inference with the file. We can do a few more things to the Lambda function to make it efficient.

In configurations, we can remove multiple retries in asynchronous section, we can also increase the memory to 1GB and change the timeout to 15min in General configuration sections.

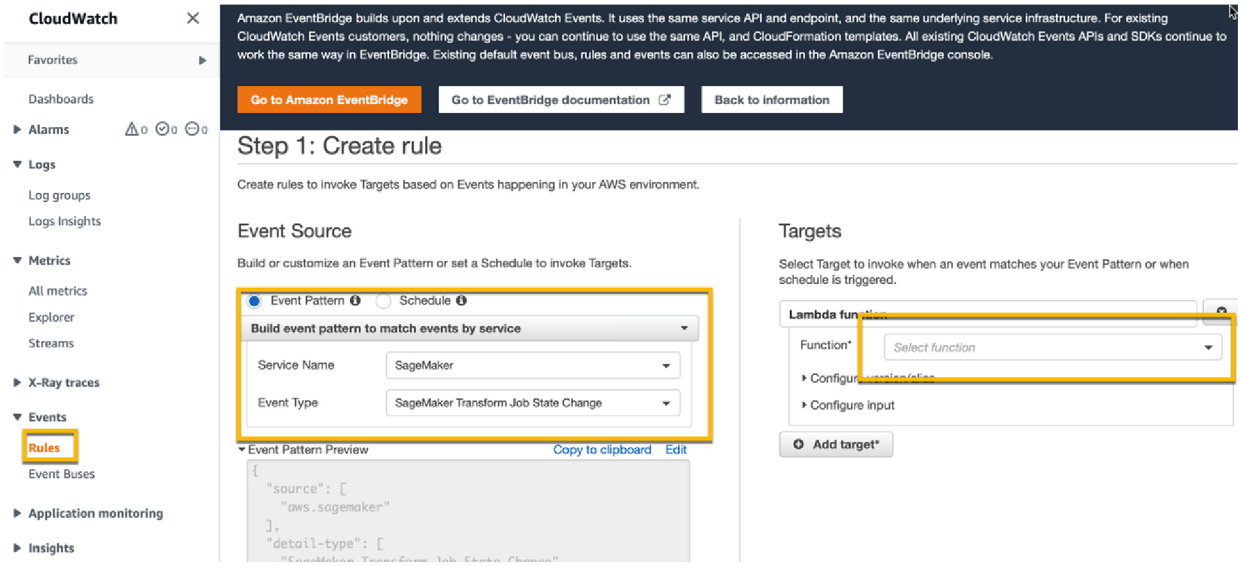
Lambda Function for Checking completion status and re-assemble the output csv

This lambda function is created using the code from batch-ing-status.py code. The trigger for this is cloudwatch events.

1. Let's start with lambda function as above and add an addition permission for SNS to this lambda function.
2. For the Trigger, Let's create a cloudwatch event and associate with this lambda function.



3. Create a rule and select SageMaker as service and Transform Job State change as event type, on the right hand side select the second lambda function created above as your target and follow the rest of the prompts.



This lambda function will kick off after the first lambda function that initiated the inference. As a part of this, we have also provided code to create a SNS topic and subscribe to updates on Inference results. Whoever is on subscription list will get an email asking to confirm subscription and once confirmed they will get an email when the inference is successfully completed. In your case, your operations people will get an email saying new predictions are available.

QuickSight and Q

We will configure the quicksight folder in your output bucket to sync with Amazon Quicksight.

Here, we will work with Quicksight setup I have walked you through in the past and we will add the new s3 location to the manifest.json file to be included in the dataset access.