

# Amazon SageMaker Operators for Kubernetes

Use Kubernetes to train, tune, and deploy machine learning (ML) models in Amazon SageMaker

# Using Kubernetes for ML is hard to manage and scale

Build and manage services within Kubernetes cluster for ML

+

Make disparate open-source libraries and frameworks work together in a secure and scalable way

+

Requires time and expertise from infrastructure, data science, and development teams

=

Need an easier way to use Kubernetes for ML

# Introducing Amazon SageMaker Operators for Kubernetes

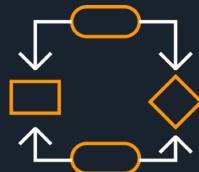
## Kubernetes customers can now train, tune, & deploy models in Amazon SageMaker



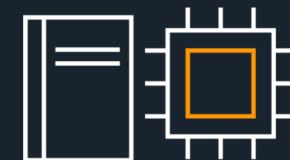
Train, tune, and deploy  
models in SageMaker



Orchestrate ML workloads  
from your Kubernetes  
environments



Create pipelines and  
workflows in Kubernetes

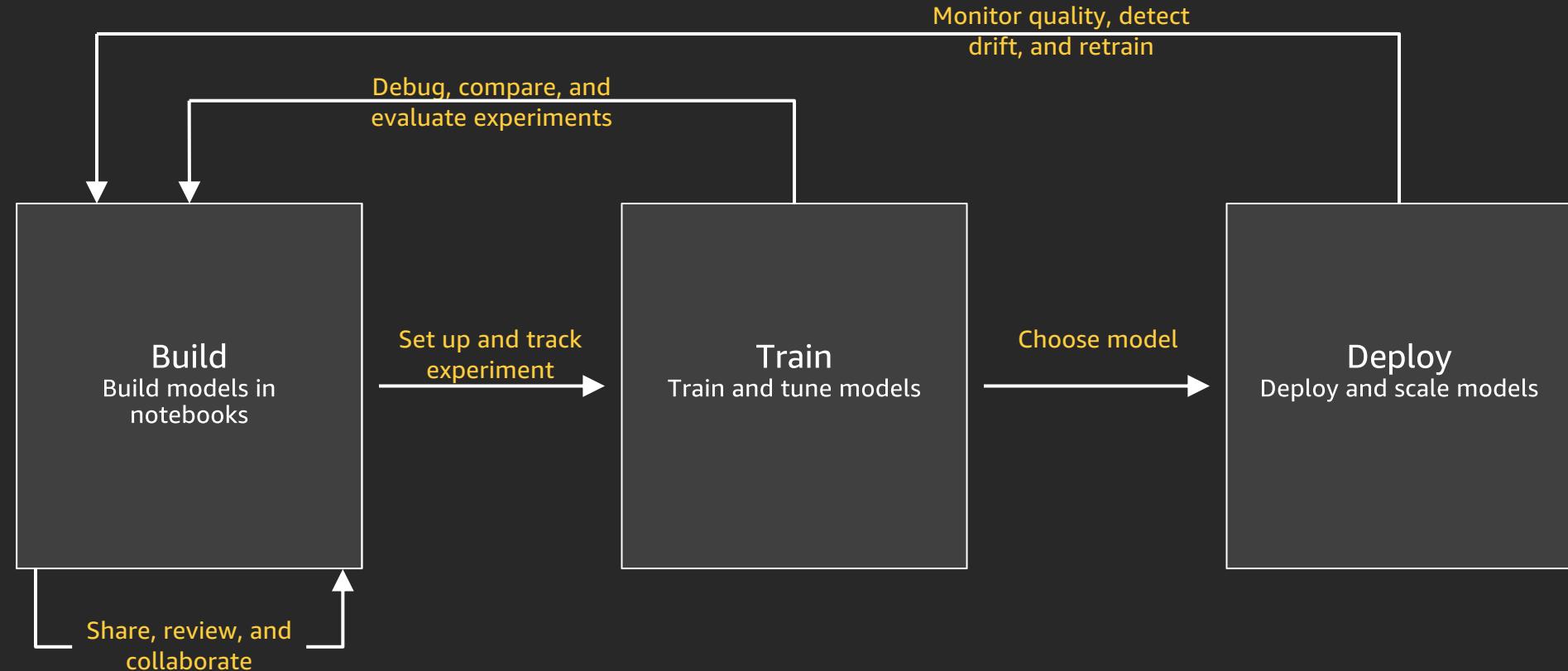


Fully managed  
infrastructure in SageMaker



# Amazon SageMaker Studio

# Machine learning is iterative



# Introducing Amazon SageMaker Studio



## Amazon SageMaker Notebooks

Quick start collaborative notebooks



## Amazon SageMaker Experiments

Organize, track, and compare ML experiments



## Amazon SageMaker Debugger

Debug models with real-time, automated alerts



## Amazon SageMaker Model Monitor

Monitor models in production and detect drift



## Amazon SageMaker Autopilot

Get from data to models automatically

# Amazon SageMaker Studio

Amazon SageMaker Studio

xgboost\_customer\_churn.ipynb

File Edit View Run Kernel Git Tabs Settings Help

• Have the predictor variable in the first column  
• Not have a header row

But first, let's convert our categorical features into numeric features.

```
[ ]: model_data = pd.get_dummies(churn)
model_data = pd.concat([model_data['Churn_True'], model_data.drop(['Churn_Tru
<   >
...
And now let's split the data into training, validation, and test sets. This will help prevent us from overfitting the model, and allow us to test the models accuracy on data it hasn't already seen.
```

```
[ ]: train_data, validation_data, test_data = np.split(model_data.sample(frac=1
train_data.to_csv('train.csv', header=False, index=False)
validation_data.to_csv('validation.csv', header=False, index=False)
<   >
...
Now we'll upload these files to S3.
```

```
[ ]: boto3.Session().resource('s3').Bucket(bucket).Object(os.path.join(prefix,
boto3.Session().resource('s3').Bucket(bucket).Object(os.path.join(prefix,
<   >
...
0 2 conda_amazonei_mxnet_p27 | Idle Mode: Command ✘ Ln 1, Col 1 xgboost_customer_churn.ipynb
```

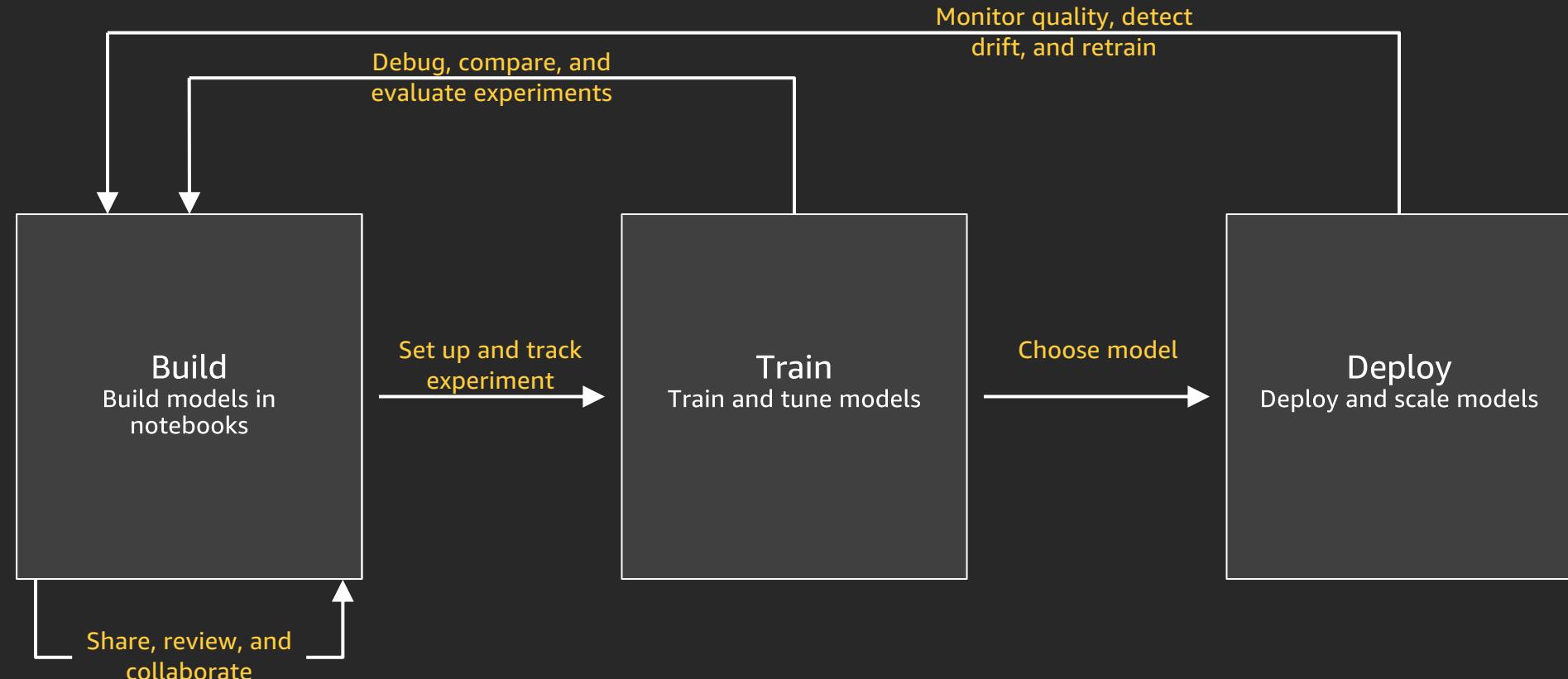
Trial Component Chart

A line chart titled "Trial Component Chart" showing training loss over time. The x-axis is labeled "period" and ranges from 0 to 8. The y-axis is labeled "trainloss\_last" and ranges from 0.0 to 0.4. There are four data series: a blue line starting at ~0.35 and ending at ~0.15; a green line starting at ~0.38 and ending at ~0.15; an orange line starting at ~0.25 and ending at ~0.05; and a red line starting at ~0.25 and ending at ~0.05. All lines show a general downward trend with some fluctuations.

Trial Component List

Status	Experiment	Type	Trial	Trial
Completed	customer-churn-predic...	Training job	Trial-3	Tra
Completed	customer-churn-predic...	Training job	Trial-2	Tra
Completed	customer-churn-predic...	Training job	Trial-1	Tra
Completed	customer-churn-predic...	Training job	Trial-0	Tra

# Build fast and collaborate



# Quick Start collaborative Amazon SageMaker Notebooks (in Preview)



## Single sign-on

Integrated with AWS SSO



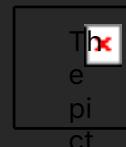
## Secure

Administrators govern access and permissions



## Switch compute environments

Switch compute instances on the fly



## One-click share

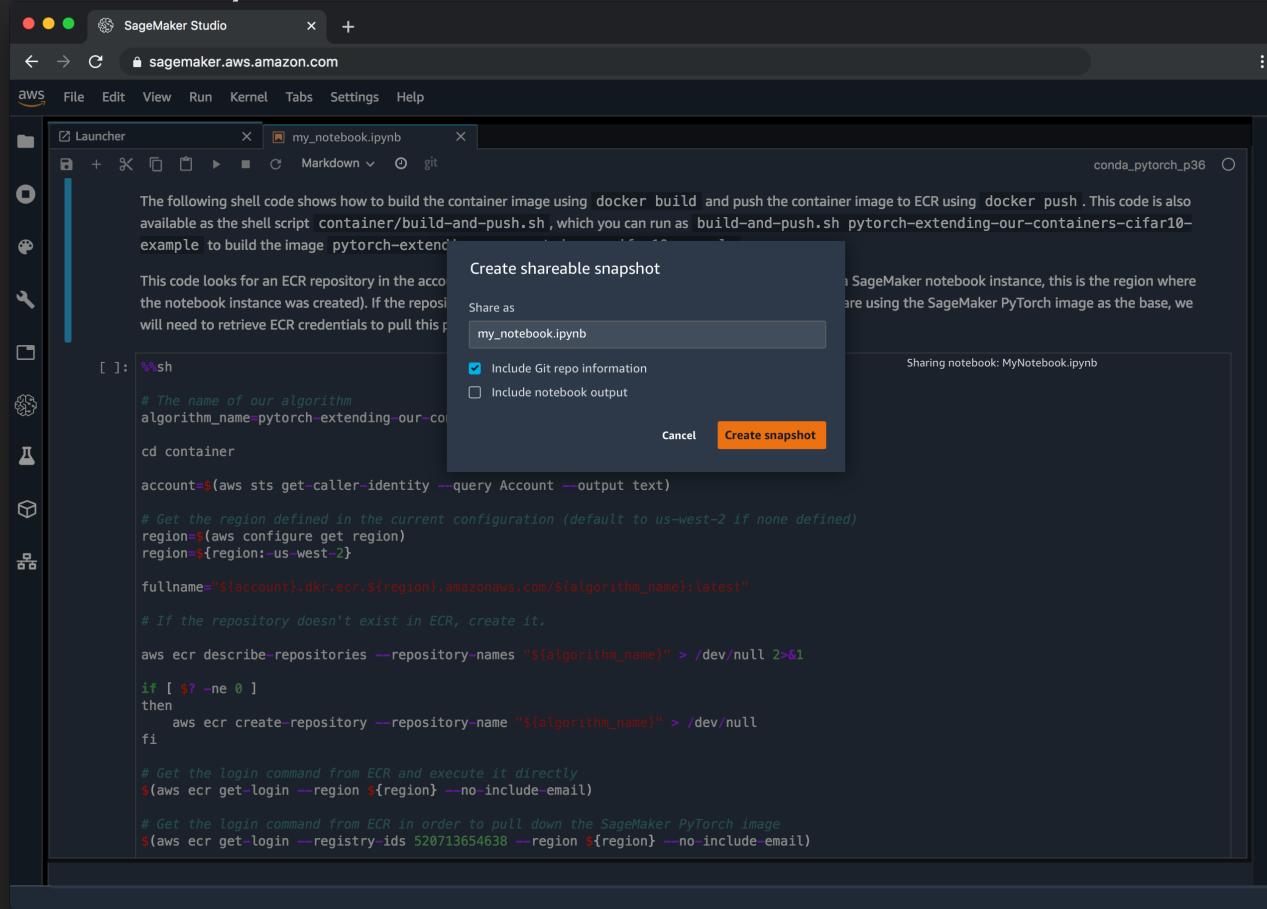
Share URL with a single click



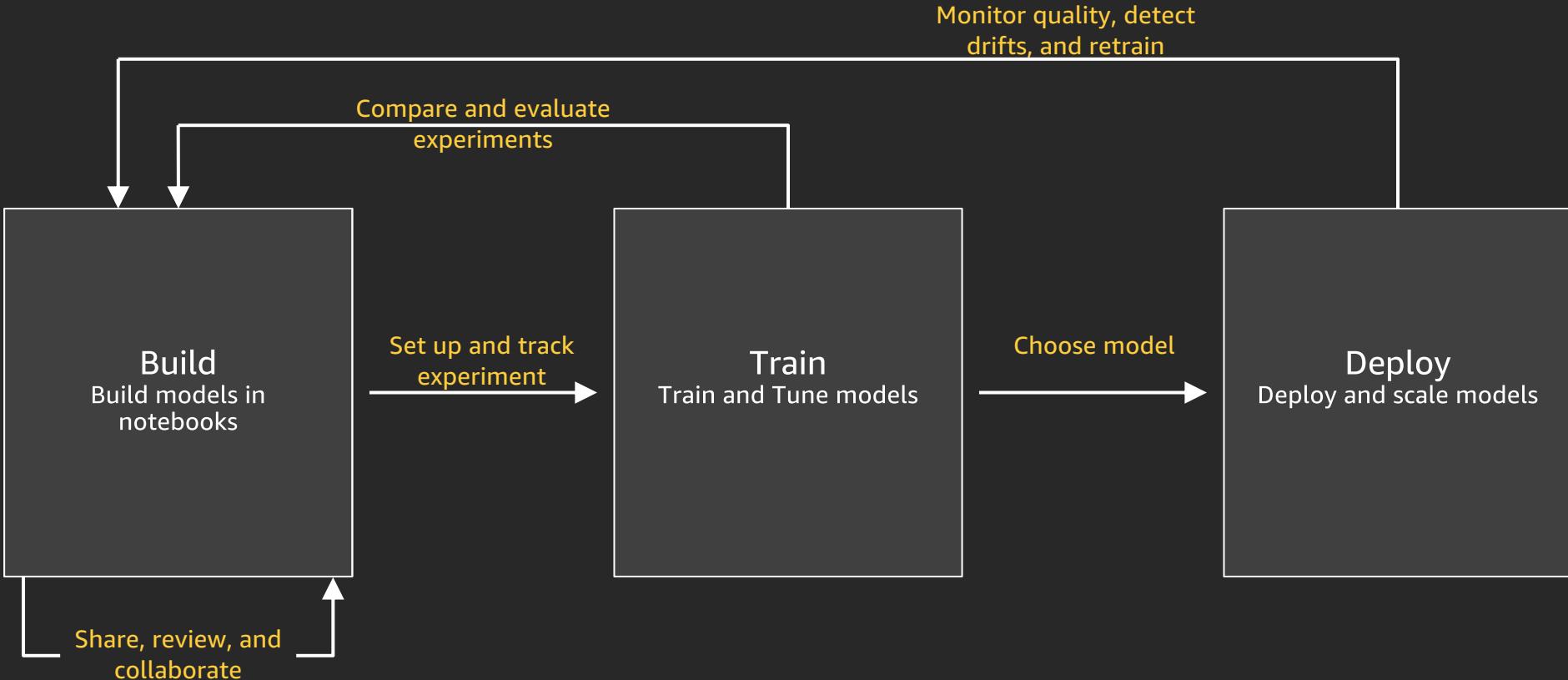
## Reproducible notebooks

Easily reproduce and review notebooks

# Amazon SageMaker Studio Notebooks



# Manage Experiments



# Amazon SageMaker Experiments



## Track experiments

Track parameters, metrics, models, and more



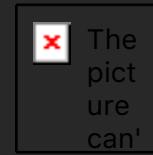
## Organize

Organize your experiments and trials



## Compare experiments

Compare, evaluate, and iterate



## Python SDK

Tracking and analytics APIs

The picture can't be displayed.

# Amazon SageMaker Experiments

The screenshot shows the Amazon SageMaker Studio interface with the following details:

- Header:** Amazon SageMaker Studio, File, Edit, View, Run, Kernel, Git, Tabs, Settings, Help.
- Left Sidebar:** Includes icons for Home, Trials, Experiments, Training jobs, and a gear icon. The "TRIALS" section shows 0 rows selected, and the "Name" column lists trials from Trial-9 down to Trial-0, each with a timestamp indicating when it was last modified.
- Middle Panel:** A tab titled "xgboost\_customer\_churn.ipynb" is active. Below it is a "Trial Component List" tab.
- Right Panel:** The "Trial Component List" view displays a table of trial components. The columns are: Status, Experiment, Type, Trial, Trial component, Created on, and Last modified. All entries are "Completed" and belong to the "customer-churn-pred..." experiment, type "Training job", under trial names Trial-9 through Trial-0, created and last modified 2 hours ago.
- Bottom:** A footer bar with icons for search, refresh, and help, and the text "Trial Component List".

# Amazon SageMaker Experiments

Amazon SageMaker Studio File Edit View Run Kernel Git Tabs Settings Help

xgboost\_customer\_churn.ipyr Describe Trial Component

Experiment: customer-churn-prediction  
Trial: Trial-5

Trial stages

- Training**  
Created 1 hour ago
- Preprocessing  
Created 2 hours ago

Charts Metrics Debugging Parameters Artifacts AWS Settings Trial Mappings

1 CHARTS Add chart

train:loss\_last with 1-minute aggregation

A line chart titled "train:loss\_last with 1-minute aggregation". The Y-axis is labeled "train:loss\_last" and ranges from 0.0 to 0.4. The X-axis is labeled "time" and shows two segments of 15 minutes each, with markers at 11:46, :15, :30, :45, 11:47, :15, :30, :45, 11:48, and :15. A single blue line starts at approximately 0.35 at 11:46 and decreases steadily to about 0.1 by 11:47, remaining relatively flat until 11:48, and then slightly increasing towards the end of the second segment.

Time	train:loss_last
11:46	0.35
:15	0.28
:30	0.22
:45	0.18
11:47	0.12
:15	0.10
:30	0.09
:45	0.09
11:48	0.09
:15	0.10

trialComponentName  
— Training-Run-5-aws-training-job

CHART PROPERTIES

Chart type

- Histogram
- Line

X-axis dimension

- Epoch
- Time
- Periods from start

X-axis aggregation

- 1-minute
- 5-minute
- 60-minute

Y-axis

train:loss\_last ▾

Describe Trial Component

# Amazon SageMaker Experiments

Amazon SageMaker Studio   File   Edit   View   Run   Kernel   Git   Tabs   Settings   Help

C  
/ customer-churn-prediction / Trial-5 /  
TRIAL COMPONENTS  
1 row selected  
Name  
Training  
Preprocessing

xgboost\_customer\_churn.ipynb X   Describe Trial Component X

Experiment: customer-churn-prediction  
Trial: Trial-5

Trial stages	Charts	Metrics	Debugging	Parameters	Artifacts	AWS Settings	Trial Mappings														
<b>Training</b> Created 1 hour ago		<table border="1"><thead><tr><th>Name</th><th>Minimum</th><th>Maximum</th><th>Standard Deviation</th><th>Final value</th></tr></thead><tbody><tr><td>test:loss</td><td>0.0592</td><td>0.1852</td><td>0.040683282124780014</td><td>0.0592</td></tr><tr><td>test:accuracy</td><td>95</td><td>98</td><td>1.0000000000000016</td><td>98</td></tr></tbody></table>	Name	Minimum	Maximum	Standard Deviation	Final value	test:loss	0.0592	0.1852	0.040683282124780014	0.0592	test:accuracy	95	98	1.0000000000000016	98				
Name	Minimum	Maximum	Standard Deviation	Final value																	
test:loss	0.0592	0.1852	0.040683282124780014	0.0592																	
test:accuracy	95	98	1.0000000000000016	98																	
<b>Preprocessing</b> Created 2 hours ago		<table border="1"><thead><tr><th>Name</th><th>Minimum</th><th>Maximum</th><th>Standard Deviation</th><th>Final value</th></tr></thead><tbody><tr><td>train:loss</td><td>0.051475</td><td>1.617049</td><td>0.20749030578873093</td><td>0.150807</td></tr></tbody></table>	Name	Minimum	Maximum	Standard Deviation	Final value	train:loss	0.051475	1.617049	0.20749030578873093	0.150807									
Name	Minimum	Maximum	Standard Deviation	Final value																	
train:loss	0.051475	1.617049	0.20749030578873093	0.150807																	

Describe Trial Component

# Amazon SageMaker Experiments

Amazon SageMaker Studio   File   Edit   View   Run   Kernel   Git   Tabs   Settings   Help

xgboost\_customer\_churn.ipynb   Trial Component List   Trial Component Chart

TRIAL COMPONENTS 5 rows selected. Select rows to toggle chart visibility. Add Chart

Experiment	Trial	Trial Component	Type	Training
customer-churn-pred...	Trial-8	Training-Run-8-aws-training-job	arn:aws:sagemaker:us...	~5 minu
customer-churn-pred...	Trial-7	Training-Run-7-aws-training-job	arn:aws:sagemaker:us...	~6 minu
customer-churn-pred...	Trial-6	Training-Run-6-aws-training-job	arn:aws:sagemaker:us...	~4 minu
customer-churn-pred...	Trial-5	Training-Run-5-aws-training-job	arn:aws:sagemaker:us...	~4 minu

Charts

train:loss\_last with 1-minute aggregation

X-axis dimension: Epoch

X-axis aggregation: 1-minute

Y-axis: train:loss\_last

CHART PROPERTIES

Data type: Summary statistics

Chart type: Line

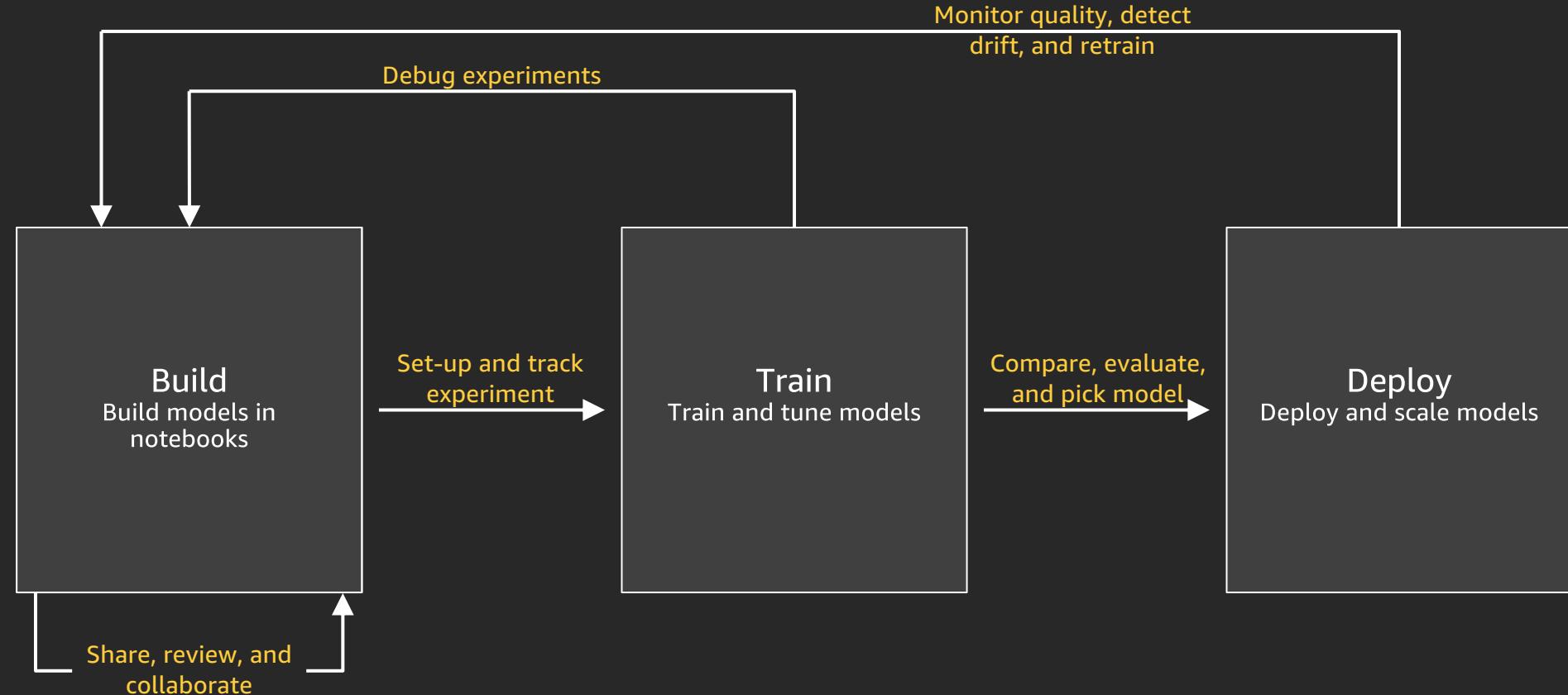
X-axis dimension: Epoch

X-axis aggregation: 1-minute

Y-axis: train:loss\_last

Trial Component Chart

# Debug training runs



# Amazon SageMaker Debugger



## Framework support

TensorFlow, Apache MXNet, PyTorch, XGBoost



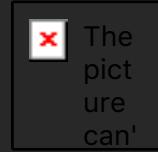
## Introspect models

Introspects, collects and analyzes tensors



## Rules and Alerts

Provides real-time alerts when bottleneck is identified



## Python SDK

The picture can't be displayed  
Write your own debug rules

# Amazon SageMaker Debugger

Amazon SageMaker Studio

File Edit View Run Kernel Git Tabs Settings Help

SMDebugger-CloudWatch-Logs

Using SageMaker Rules

In this example we'll demonstrate how to use SageMaker rules to be evaluated against your training. You can find the list of SageMaker rules and the configurations best suited for using them here.

The rules we'll use are **VanishingGradient** and **PoorWeightInitialization**. As the names suggest, the rules will attempt to evaluate if there are vanishing gradients in the tensors captured by the debugging hook during training and also if the initial weights are poorly initialized. Additionally, we are also adding a custom rule that we created.

```
[42]: estimator = TensorFlow(
    role=sagemaker.get_execution_role(),
    base_job_name='smdebugger-demo-mnist-tensorflow',
    train_instance_count=3,
    train_instance_type='ml.p3.2xlarge',
    image_name='cpu_training_image',
    entry_point='entrypoint_script',
    framework_version='1.15',
    py_version='py3',
    train_max_run=3600,
    script_mode=True,
    sagemaker_session=sess,
    # New parameter
    rules = [Rule.sagemaker(rule_configs.loss_not_decreasing()),
             Rule.sagemaker(rule_configs.poor_weight_initialization()),
             Rule.custom(name='MyCustomRule', # used to identify the rule
                        image_uri='759209512051.dkr.ecr.us-west-2.amazonaws.com/sagemaker-instance-type=ml.c4.xlarge', # instance type to run the rule eval
                        source='my_custom_rule.py', # path to the rule source file
                        rule_to_invoke='CustomGradientRule', # name of the class to invoke
                        volume_size_in_gb=100, # EBS volume size required to be attached
                        collections_to_save=[CollectionConfig(name='gradients')], # collection
                        rule_parameters=[
                            {"threshold": "0.0" # this will be used to initialize 'threshold'
                            } ],
        hyperparameters = {'num_epochs' : 100 }
    )
```

Note that Sagemaker-Debugger is only supported for py\_version='py3' currently.

Let's start the training by calling `fit()` on the MXNet estimator

```
[43]: # After calling fit, SageMaker will spin off 1 training job and 1 rule job for you
# The rule evaluation status(es) will be visible in the training logs
# at regular intervals

estimator.fit(wait=False)
```

Describe Trial Component

Experiment: Unassigned  
Trial: Unassigned

Trial stages

smdebugger-demo-mnist-tensorflow-2019-12-01-07-26-43-574-aws-training-job	Status	Last modified	Rule name	Job ARN
Created 4 minutes ago	In Progress	1 minute ago	LossNotDecreasing	arn:aws:sagemaker:us-west-2:123456789012:job/12345678901234567890123456789012
	In Progress	1 minute ago	PoorWeightInitialization	arn:aws:sagemaker:us-west-2:123456789012:job/12345678901234567890123456789012
	In Progress	1 minute ago	MyCustomRule	arn:aws:sagemaker:us-west-2:123456789012:job/12345678901234567890123456789012

Trial Component Chart

TRIAL COMPONENTS 2 rows selected. Select rows to toggle chart visibility.

Experiment	Trial	Trial Component	Type
(@) N/A	N/A	smdebugger-demo-mnist-tensorflow-2019-12-01-02-10-58-699-aws-training-job	arn:aws:sagemaker:us-west-2:123456789012:job/12345678901234567890123456789012
(@) N/A	N/A	smdebugger-demo-mnist-tensorflow-2019-12-01-01-57-34-825-aws-training-job	arn:aws:sagemaker:us-west-2:123456789012:job/12345678901234567890123456789012

Charts

sparse\_softmax\_cross\_entropy\_loss/value:0\_avg with 1-minute aggregation

period

trialComponentName

Mode: Command

Ln 11, Col 22

SMDebugger-CloudWatch-Logs.pynb

# Amazon SageMaker Debugger

Amazon SageMaker Studio

File Edit View Run Kernel Git Tabs Settings Help

SMDebugger-CloudWatch-Logs

Markdown conda\_tensorflow\_p36

```
rule_parameters = {
    "threshold": "0.0" # this will be used to initialize 'threshold'
},
hyperparameters = {"num_epochs": 100}
```

Note that Sagemaker-Debugger is only supported for py\_version='py3' currently.

Let's start the training by calling `fit()` on the MXNet estimator

```
[43]: # After calling fit, SageMaker will spin off 1 training job and 1 rule job for you
# The rule evaluation statuses will be visible in the training logs
# at regular intervals
estimator.fit(wait=False)
```

## Result

As a result of calling the `fit()` SageMaker debugger kicked off a rule evaluation job for our custom gradient logic in parallel with the training job that was monitoring the tensors output by the training job. As you can see, in the summary, there was no step in the training which reported vanishing gradients in the tensors. Although, the loss was not found to be decreasing at step 1900.

```
[44]: estimator.latest_training_job.rule_job_summary()
```

```
[44]: [{'RuleConfigurationName': 'LossNotDecreasing',
       'RuleEvaluationJobArn': 'arn:aws:sagemaker:us-west-2:331110439030:processing-job/smdebugger-demo-mnist-tensorflow-lossnotdecreasing-745a1f2e',
       'RuleEvaluationStatus': 'NoIssuesFound',
       'LastModifiedTime': datetime.datetime(2019, 12, 1, 7, 39, 41, 257000, tzinfo=tzlocal()),
       {'RuleConfigurationName': 'PoorWeightInitialization',
        'RuleEvaluationJobArn': 'arn:aws:sagemaker:us-west-2:331110439030:processing-job/smdebugger-demo-mnist-tensorflow-poorweightinitialization-752170c0',
        'RuleEvaluationStatus': 'NoIssuesFound',
        'LastModifiedTime': datetime.datetime(2019, 12, 1, 7, 39, 41, 257000, tzinfo=tzlocal()),
       {'RuleConfigurationName': 'MyCustomRule',
        'RuleEvaluationJobArn': 'arn:aws:sagemaker:us-west-2:331110439030:processing-job/smdebugger-demo-mnist-tensorflow-mycustomrule-3cef575e',
        'RuleEvaluationStatus': 'NoIssuesFound',
        'LastModifiedTime': datetime.datetime(2019, 12, 1, 7, 39, 41, 257000, tzinfo=tzlocal())}]
```

Let's try and look at the logs of the rule job for loss not decreasing. To do that, we'll use this utility function to get a link to the rule job logs

Describe Trial Component

Experiment: Unassigned  
Trial: Unassigned

Trial stages

smdebugger-demo-mnist-tensorflow-2019-12-01-07-26-43-574-aws-training-job	Status	Last modified	Rule name	Job ARN
Created 16 minutes ago	No Issues Found	3 minutes ago	LossNotDecreasing	arn:aws:sagemaker:us-west-2:331110439030:processing-job/smdebugger-demo-mnist-tensorflow-lossnotdecreasing-745a1f2e
	No Issues Found	3 minutes ago	PoorWeightInitialization	arn:aws:sagemaker:us-west-2:331110439030:processing-job/smdebugger-demo-mnist-tensorflow-poorweightinitialization-752170c0
	No Issues Found	3 minutes ago	MyCustomRule	arn:aws:sagemaker:us-west-2:331110439030:processing-job/smdebugger-demo-mnist-tensorflow-mycustomrule-3cef575e

Trial Component Chart

TRIAL COMPONENTS 2 rows selected. Select rows to toggle chart visibility.

Experiment	Trial	Trial Component	Type
N/A	N/A	smdebugger-demo-mnist-tensorflow-2019-12-01-07-26-43-574-aws-training-job	arn:aws:sagemaker:us-west-2:331110439030:processing-job/smdebugger-demo-mnist-tensorflow-lossnotdecreasing-745a1f2e
N/A	N/A	smdebugger-demo-mnist-tensorflow-2019-12-01-07-16-30-198-a-trial-component	arn:aws:sagemaker:us-west-2:331110439030:processing-job/smdebugger-demo-mnist-tensorflow-mycustomrule-3cef575e

Charts

sparse\_softmax\_cross\_entropy\_loss/value:0\_avg with 1-minute aggregation

period

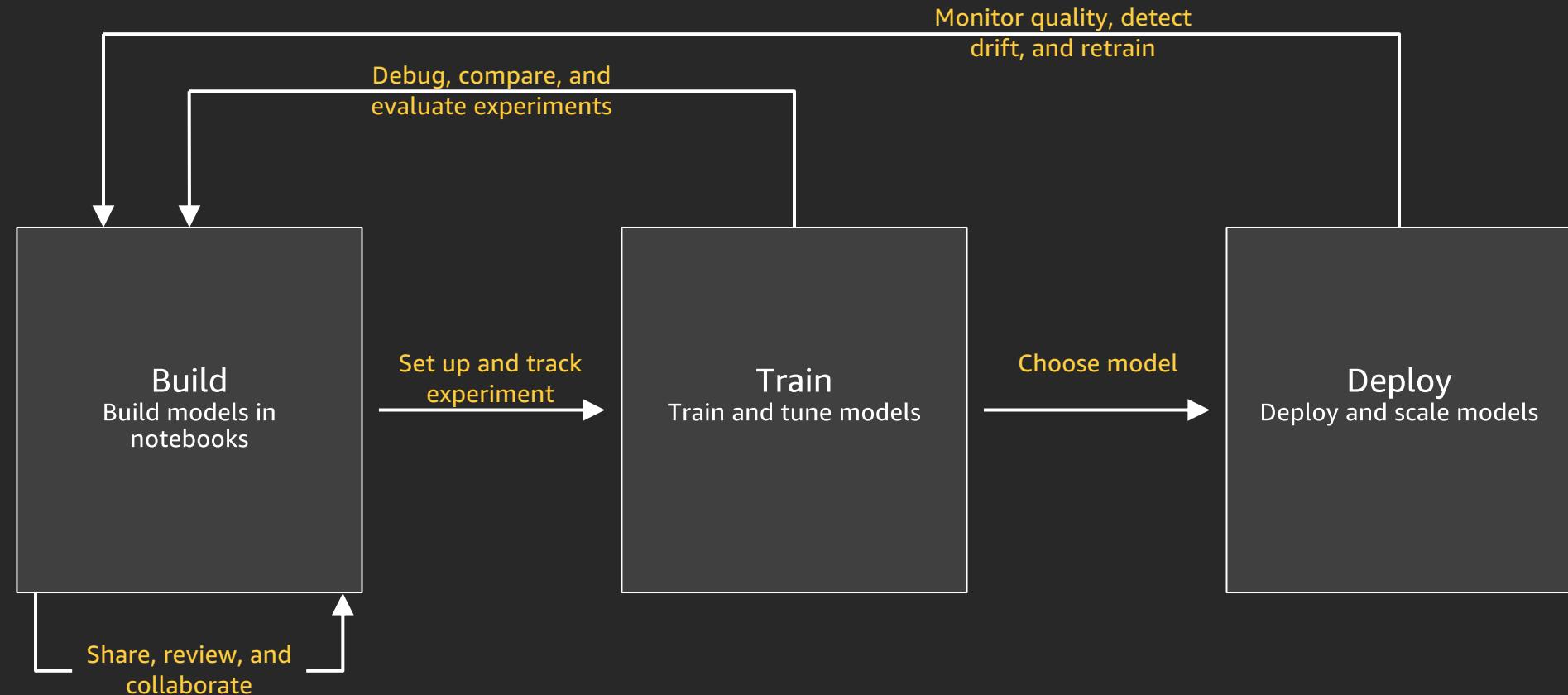
trialComponentName

smdebugger-demo-mnist-tensorflow-lossnotdecreasing-745a1f2e smdebugger-demo-mnist-tensorflow-mycustomrule-3cef575e

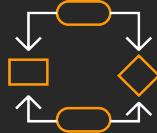
0 5 10 15

Trial Component Chart

# Monitor models



# Amazon SageMaker Model Monitor



## Collect data

Periodically collects data from endpoints into Amazon S3



## Analyze

Computes feature statistics



## Monitor

Monitors trends in data and detects drifts



## Alerts

Alerts on Amazon CloudWatch and Amazon SageMaker Studio

# Amazon SageMaker Model Monitor

aws File Edit View Run Kernel Tabs Settings Help

ENDPOINTS

Search endpoints

Name	Monitoring status
Endpoint-01	Issues
Endpoint-02	No issues
Endpoint-03	Issues
Endpoint-04	No issues
Endpoint-05	Issues
Endpoint-06	No issues
Endpoint-07	Issues

Showing all rows Last updated seconds ago

### MODEL QUALITY MONITORING

Monitor your machine learning models in production to detect data drift, data quality issues, anomalies, and deviation in model quality. [Learn more](#)

Schedule [Schedule\_Name]

Search by features Timeline 1h 6h 12h 1d **1w**

#### Feature 1: Completeness

Completeness

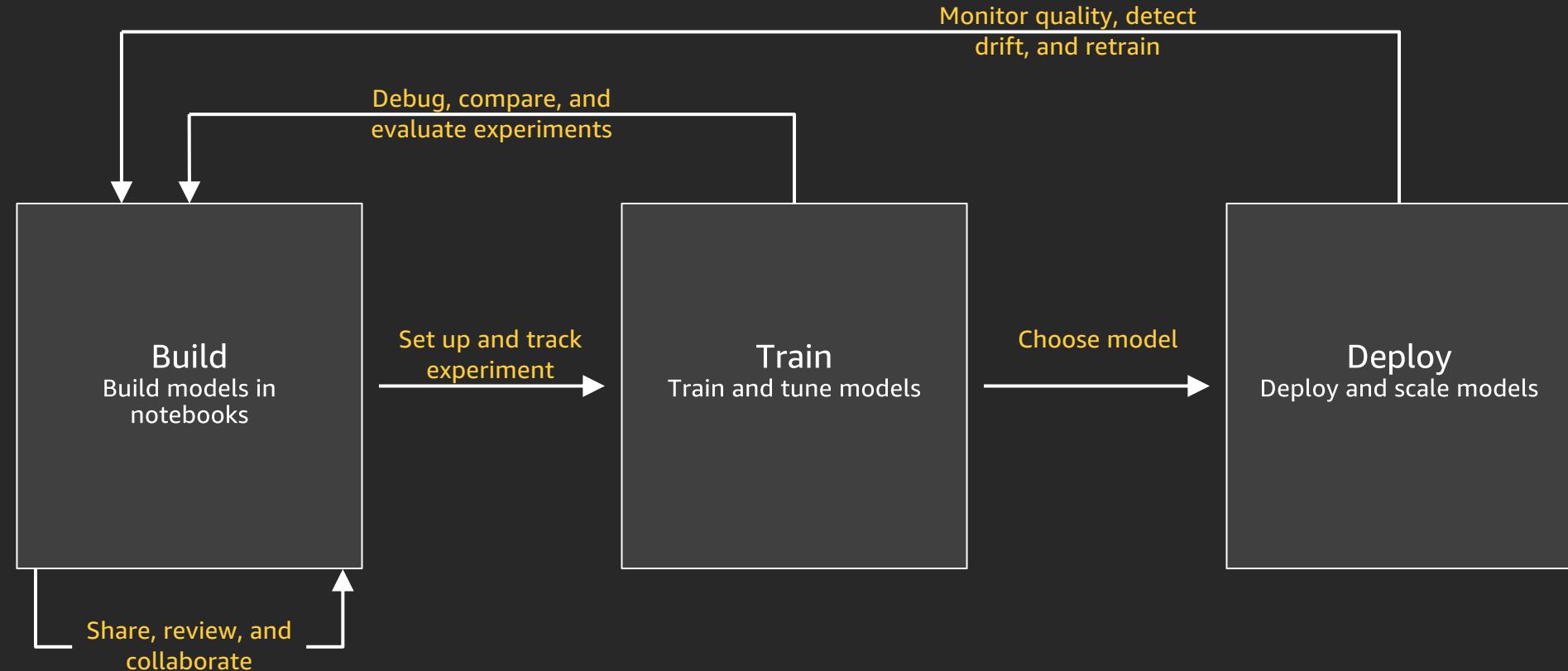
Date and time	Baseline	Current
Sept 01 12:00	0.80	0.78
Sept 02 12:00	0.80	0.80
Sept 03 12:00	0.80	0.81

#### Feature 3: Completeness

Completeness

Date and time	Baseline	Current
Sept 01 12:00	0.80	0.78
Sept 02 12:00	0.80	0.80
Sept 03 12:00	0.80	0.81

# How else can we accelerate ML workflow?



# Amazon SageMaker Autopilot



Provide data

Data in tabular form



Specify column to predict

Support for regression and classification



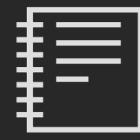
Create model

Feature generation, algorithm selection, and parameter tuning



Track experiment

Automatically tracked as an experiment



Transparent

Get notebook with source code

# Amazon SageMaker Autopilot

Amazon SageMaker Studio File Edit View Run Kernel Git Tabs Settings Help

xgboost\_customer\_churn\_stu Create experiment

### Create SageMaker Autopilot Experiment

**JOB SETTINGS**

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

S3 location for the input data  
Provide the S3 location of your input data for training. To find a path, go to [Amazon S3](#)  
 S3 path

Target attribute name  
Provide the attribute name that you want Auto-ML to predict.  
 Enter attribute name  
This input is case-sensitive. Incorrect input will cause the experiment to fail.

S3 location for the output  
Provide the S3 location for storing the output. To find a path, go to [Amazon S3](#)  
 S3 path

Select the machine learning problem type

Auto

Binary classification

Create Experiment

EXPERIMENTS

1 row selected Create Experiment

Name	Last Modified
Unassigned trial comp...	1 minute ago
customer-churn-auto...	18 hours ago
customer-churn-pred...	22 hours ago
customer-churn-predi...	2 days ago

0 \$ 2

Create experiment

# Amazon SageMaker Autopilot

aws File Edit View Run Kernel Git Tabs Settings Help

+ 📁 ⌂ ⌄ ⌅ ⌆

📁 / automl-preview /

Name	Last Modified
bank-additional	3 hours ago
model	3 hours ago
sagemaker_auto...	2 minutes ago

Terminal 2 | sagemaker\_automl\_direct\_m:● | my-sagemaker-autopilot

EXPERIMENT: MY-SAGEMAKER-AUTOPILLOT

Analyzing Data → Feature Engineering → Model Tuning → Completed

AutoML is tuning the model.

If experiment is taking too long to run, you can [stop the experiment](#).

Trials Job profile

TRIALS

0 row selected Deploy model

Trial name	Status	Start time	End time	Objective
my-sagemaker-tuning-job-...	Completed	4 minutes ago		0.9031320214271545
my-sagemaker-tuning-job-...	Completed	3 minutes ago		0.894877016544342
my-sagemaker-tuning-job-...	Completed	2 minutes ago		0.9175770282745361
my-sagemaker-tuning-job-...	Completed	3 minutes ago		0.9138140082359314
my-sagemaker-tuning-job-...	Completed	2 minutes ago		0.9117500185966492
my-sagemaker-tuning-job-...	Completed	2 minutes ago		0.9158779978752136

2 \$ 1 ⌂ my-sagemaker-autopilot



# Q&A



# Thank You!

Please join us again for another PartnerCast session

