

PRAGMATIC MACHINE LEARNING AND ML ENGINEERING IN THE CLOUD WITH AMAZON SAGEMAKER

[CONF42 MACHINE LEARNING]

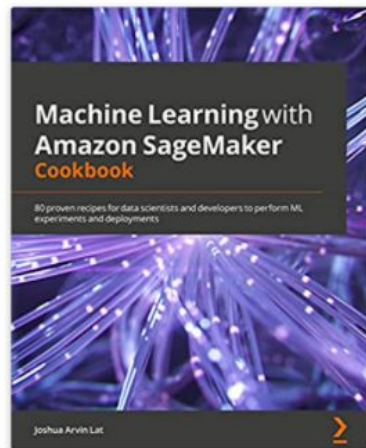


JOSHUA “ARVS” LAT

- Chief Technology Officer of NuWorks Interactive Labs
- AWS Machine Learning Hero
- Author of “Machine Learning with Amazon SageMaker Cookbook”

Machine Learning with Amazon SageMaker Cookbook: 80 proven recipes for data scientists and developers to perform ML experiments and deployments Kindle Edition

by [Joshua Arvin Lat](#) (Author) | Format: Kindle Edition



ISBN-13: 978-1800567030

ISBN-10: 1800567030

[Why is ISBN important?](#)



Kindle

\$31.99

Paperback

\$44.99

Other Sellers

See all 2 versions

Buy

eBook features:

- Highlight, take notes, and search in the book

Read with the free Kindle apps (available on iOS, Android, PC & Mac), Kindle E-readers and on Fire Tablet devices. [See all supported devices](#)

Sold by: Amazon.com Services LLC

\$31.99

Pre-order with 1-Click*

This title will be auto-delivered to your Kindle on **October 11, 2021**.

Deliver to:

Your Kindle Library

[Enter a promotion code or Gift Card](#)

A step-by-step problem solution based guide to prepare, build, train, and deploy high-quality machine learning (ML) models with Amazon SageMaker

Key Features

- Perform ML experiments with built-in and custom algorithms in SageMaker
- Explore proven solutions when working with TensorFlow, PyTorch, Hugging Face Transformers, and scikit-learn.
- Use the different features and capabilities of SageMaker to automate relevant ML processes

MACHINE LEARNING



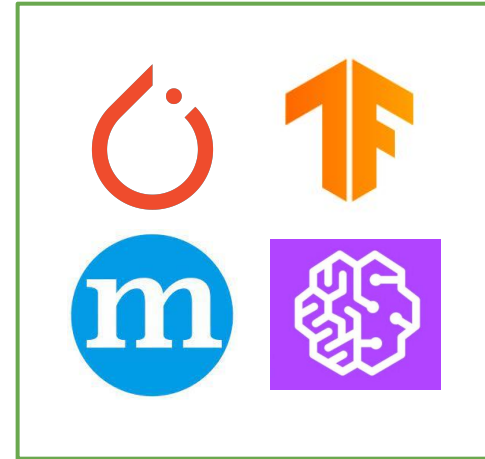
CAT

OR

NOT CAT

**BUILD EVERYTHING
FROM SCRATCH**

VS



Requirement

Possible Solution

Anomaly Detection



SageMaker Random Cut Forest Algorithm

Product Recommendation



Amazon Personalize

Forecasting



SageMaker DeepAR Algorithm

Image and Video Analysis



Amazon Rekognition

Document Classification



SageMaker BlazingText Algorithm

Language Translation



Amazon Translate



Data Collection



Data Preparation and Cleaning



Data Visualization and Analysis



Feature Engineering



Model Training and Parameter Tuning



Model Evaluation



Model Deployment



SageMaker Python SDK

<https://github.com/PacktPublishing/Amazon-SageMaker-Cookbook>

```
estimator = sagemaker.estimator.Estimator(  
    container,  
    role,  
    instance_count=1,  
    instance_type='ml.c4.2xlarge',  
    output_path=training_s3_output_location,  
    sagemaker_session=session  
)
```

```
estimator.set_hyperparameters(  
    time_freq=freq,  
    context_length=str(context_length),  
    prediction_length=str(prediction_length),  
    num_cells=40,  
    num_layers=3,  
    likelihood="gaussian",  
    epochs=20,  
    mini_batch_size=32,  
    learning_rate=0.001,  
    dropout_rate=0.05,  
    early_stopping_patience=10  
)
```

```
data_channels = {"train": train, "test": test}
```

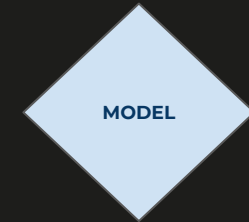
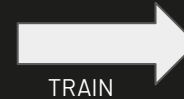
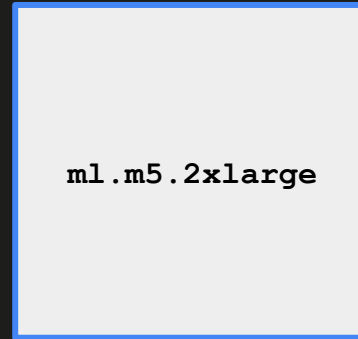
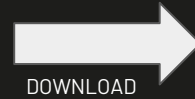
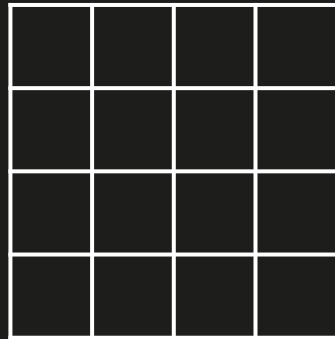
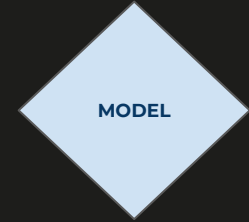
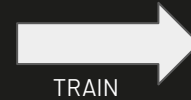
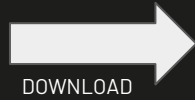
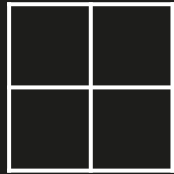
```
%%time
```

```
estimator.fit(inputs=data_channels)
```

```
estimator.model_data
```

```
predictor = estimator.deploy(  
    initial_instance_count=1,  
    instance_type="ml.m4.xlarge"  
)
```

OPTIMIZING COSTS BY USING TRANSIENT ML INSTANCES FOR TRAINING MODELS







SAGEMAKER PROCESSING	SAGEMAKER EXPERIMENTS	AUTOMATED MODEL TUNING
BUILT-IN ALGORITHMS	ML AND DL FRAMEWORK SUPPORT	SAGEMAKER CLARIFY
SAGEMAKER DEBUGGER	SAGEMAKER FEATURE STORE	SAGEMAKER AUTOPILOT
SAGEMAKER STUDIO	SAGEMAKER GROUND TRUTH	SAGEMAKER MODEL MONITOR
MANAGED SPOT TRAINING	SAGEMAKER PIPELINES	SAGEMAKER DATA WRANGLER

SageMaker Debugger

<https://github.com/PacktPublishing/Amazon-SageMaker-Cookbook>

```
from sagemaker.debugger import rule_configs, Rule, DebuggerHookConfig, CollectionConfig
```

```
save_interval = 2
prefix = "debugger"
bucket_path = 's3://{}/{}/{}'.format(s3_bucket, prefix)
```

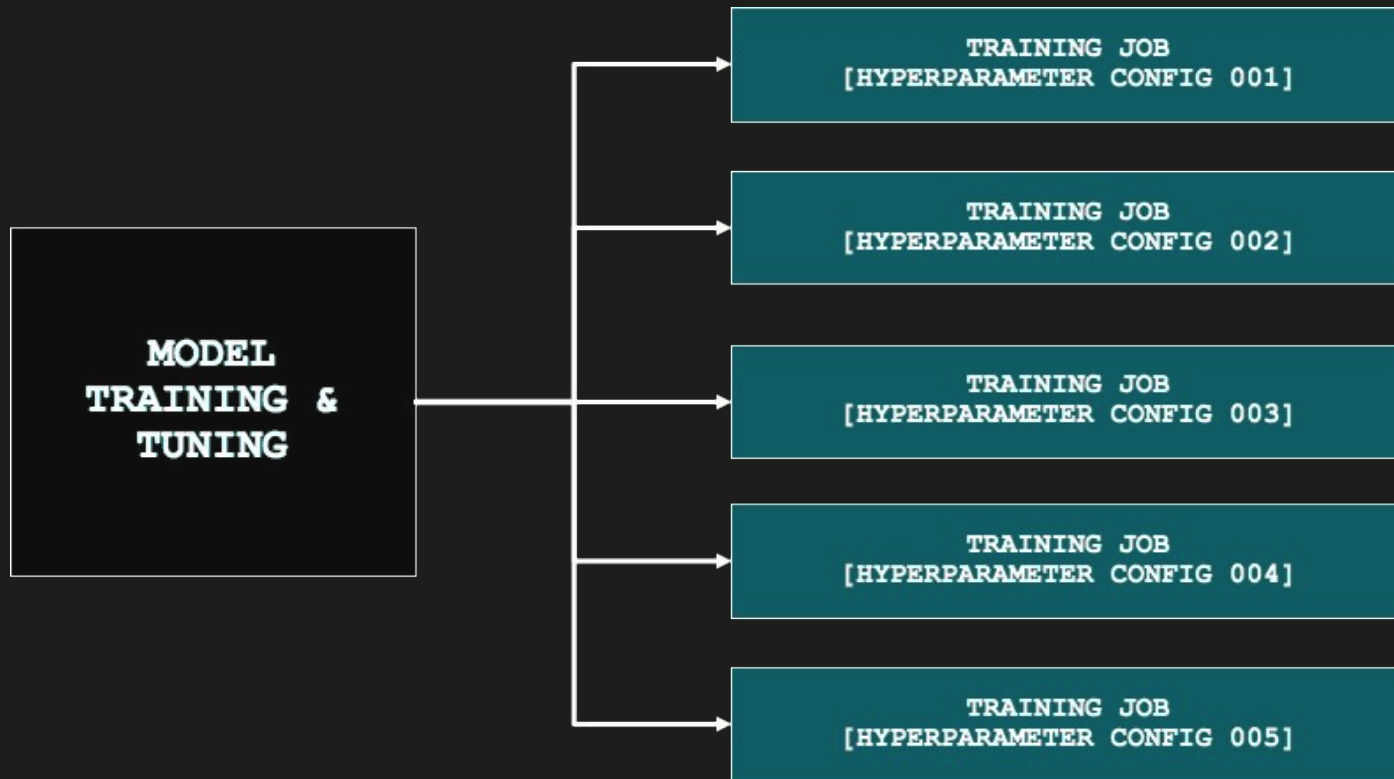
```
metrics_collection_config = CollectionConfig(
    name="metrics",
    parameters={
        "save_interval": str(save_interval)
    })

debugger_hook_config = DebuggerHookConfig(
    s3_output_path=bucket_path,
    collection_configs=[metrics_collection_config]
)
```

```
loss_not_decreasing_rule = Rule.sagemaker(
    rule_configs.loss_not_decreasing(),
    rule_parameters={
        "collection_names": "metrics",
        "diff_percent": "5",
        "num_steps": "2",
    },
)
```

```
rules = [loss_not_decreasing_rule]
```

```
estimator = sagemaker.estimator.Estimator(
    role=role_arn,
    instance_count=1,
    instance_type='ml.m5.2xlarge',
    image_uri=container,
    debugger_hook_config=debugger_hook_config,
    rules=rules,
    sagemaker_session=session)
```



Automatic Model Tuning

<https://github.com/PacktPublishing/Amazon-SageMaker-Cookbook>

```
estimator = sagemaker.estimator.Estimator(  
    container,  
    role,  
    instance_count=1,  
    instance_type='ml.m5.large',  
    output_path=training_s3_output_location,  
    sagemaker_session=session)
```

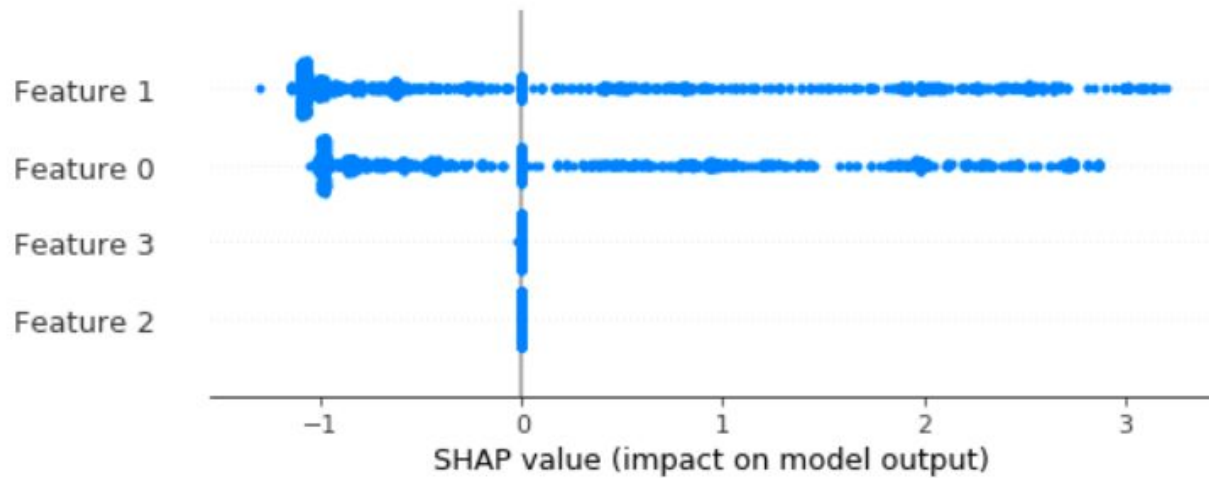
```
estimator.set_hyperparameters(  
    eval_metric='auc',  
    objective='binary:logistic',  
    num_round=50)
```

```
hyperparameter_ranges = {  
    'eta': tuner.ContinuousParameter(0, 1),  
    'min_child_weight': tuner.ContinuousParameter(3, 7),  
    'max_depth': tuner.IntegerParameter(2, 8)  
}
```

```
objective_metric_name = 'validation:auc'
```

```
hyperparameter_tuner = tuner.HyperparameterTuner(  
    estimator,  
    objective_metric_name,  
    hyperparameter_ranges,  
    max_jobs=20,  
    max_parallel_jobs=3)
```

```
hyperparameter_tuner.fit(  
    {'train': s3_input_training,  
     'validation': s3_input_validation},  
    include_cls_metadata=False,  
    wait=False  
)
```



```
{  
  "version": "1.0",  
  "explanations": {  
    "kernel_shap": {  
      "label0": {  
        "global_shap_values": {  
          "a": 0.1173995901699019,  
          "b": 0.37360024663733005,  
          "c": 0.01740283967164966,  
          "d": 0.015364162067494701  
        },  
        "expected_value": 0.34422817826271057  
      }  
    }  
  }  
}
```

ML Explainability

<https://github.com/PacktPublishing/Amazon-SageMaker-Cookbook>

```
from sagemaker.clarify import ModelConfig
```

```
model_config = ModelConfig(  
    model_name=model_name,  
    instance_type='ml.c5.xlarge',  
    instance_count=1,  
    accept_type='text/csv'  
)
```

```
from sagemaker.clarify import SageMakerClarifyProcessor
```

```
processor = SageMakerClarifyProcessor(  
    role=role,  
    instance_count=1,  
    instance_type='ml.m5.large',  
    sagemaker_session=session  
)
```

```
baseline = features.iloc[0:200].values.tolist()  
baseline
```

```
from sagemaker.clarify import SHAPConfig
```

```
shap_config = SHAPConfig(  
    baseline=baseline,  
    num_samples=50,  
    agg_method='median'  
)
```

```
headers = training_data.columns.tolist()
```

```
from sagemaker.clarify import DataConfig
```

```
data_config = DataConfig(  
    s3_data_input_path=s3_training_data_path,  
    s3_output_path=s3_output_path,  
    label='approved',  
    headers=headers,  
    dataset_type='text/csv'  
)
```

```
%%time
```

```
processor.run_explainability(  
    data_config=data_config,  
    model_config=model_config,  
    explainability_config=shap_config  
)
```

MAKING THE MOST OUT OF ML FRAMEWORKS AND ML PLATFORMS



MODEL DEPLOYED INSIDE A LAMBDA FUNCTION









LAMBDA TRIGGERING A SAGEMAKER ENDPOINT



API GATEWAY MAPPING TEMPLATES + SAGEMAKER



MODEL DEPLOYED IN FARGATE

		SAGEMAKER MULTI-MODEL ENDPOINT
		SAGEMAKER MULTI-CONTAINER ENDPOINT
		SAGEMAKER A/B TESTING SETUP USING PRODUCTION VARIANTS
		MODEL DEPLOYED INSIDE A LAMBDA FUNCTION + CONTAINER

MAKING THE MOST OUT OF ML FRAMEWORKS AND ML PLATFORMS



ID	Type	Step	Resource	Elapsed Time (ms)	Timestamp
▶ 1	ExecutionStarted		-	0.0	Sep 10, 2020 05:39:24.116 PM
▶ 2	TaskStateEntered	Training Step	-	32.0	Sep 10, 2020 05:39:24.148 PM
▶ 3	TaskScheduled	Training Step	Sagemaker training job	32.0	Sep 10, 2020 05:39:24.148 PM
▶ 4	TaskStarted	Training Step	Sagemaker training job	67.0	Sep 10, 2020 05:39:24.183 PM
▶ 5	TaskSubmitted	Training Step	Sagemaker training job	288.0	Sep 10, 2020 05:39:24.404 PM

Data Science SDK

<https://github.com/PacktPublishing/Amazon-SageMaker-Cookbook>

```
from stepfunctions.inputs import ExecutionInput
```

```
execution_input = ExecutionInput(  
    schema={  
        'ModelName': str,  
        'EndpointName': str,  
        'JobName': str  
    }  
)  
  
ei = execution_input
```

```
from stepfunctions.steps import TrainingStep
```

```
training_step = TrainingStep(  
    'Training Step',  
    estimator=estimator,  
    data={  
        'train': train  
    },  
    job_name=ei['JobName']  
)
```

```
from stepfunctions.steps import ModelStep
```

```
model_step = ModelStep(  
    'Model Step',  
    model=training_step.get_expected_model(),  
    model_name=ei['ModelName']  
)
```

```
endpoint_config_step = EndpointConfigStep(  
    "Create Endpoint Configuration",  
    endpoint_config_name=ei['ModelName'],  
    model_name=ei['ModelName'],  
    initial_instance_count=1,  
    instance_type='ml.m5.xlarge'  
)
```

```
from stepfunctions.steps import EndpointStep
```

```
endpoint_step = EndpointStep(  
    "Deploy Endpoint",  
    endpoint_name=ei['EndpointName'],  
    endpoint_config_name=ei['ModelName']  
)
```

```
from stepfunctions.steps import Chain
```

```
workflow_definition = Chain([  
    training_step,  
    model_step,  
    endpoint_config_step,  
    endpoint_step  
)
```

Data Science SDK

<https://github.com/PacktPublishing/Amazon-SageMaker-Cookbook>

```
import uuid
from stepfunctions.workflow import Workflow

workflow = Workflow(
    name='{}-{}'.format('Workflow', grs()),
    definition=workflow_definition,
    role=execution_role,
    execution_input=execution_input
)
```

```
workflow.create()
```

```
execution = workflow.execute(
    inputs={
        'JobName': '11-{}'.format(grs()),
        'ModelName': '11-{}'.format(grs()),
        'EndpointName': '11-{}'.format(grs())
    }
)
```

```
execution.list_events()
```




SageMaker Pipelines





SageMaker Pipelines

Graph Parameters Settings

Search for step...

TrainStep

TrainStep

Output Logs Information

Metric	Value
train:objective_loss/final	7907.846858705337
train:rmse	7907.846858705337
train:progress	100
train:throughput	124.50360130185419
train:absolute_loss	72.75599452427456
train:objective_loss	0.6568790177504221

Files

S3 bucket URI

SageMaker Pipelines

<https://github.com/PacktPublishing/Amazon-SageMaker-Cookbook>

```
from sagemaker.sklearn.processing import SKLearnProcessor
```

```
framework_version = "0.23-1"
```

```
sklearn_processor = SKLearnProcessor(  
    framework_version=framework_version,  
    instance_type=processing_instance_type,  
    instance_count=1,  
    role=role,  
)
```

```
from sagemaker.processing import ProcessingInput, ProcessingOutput  
from sagemaker.workflow.steps import ProcessingStep
```

```
step_process = ProcessingStep(  
    name="ProcessingStep",  
    processor=sklearn_processor,  
    inputs=[  
        ProcessingInput(  
            source=input_data,  
            destination="/opt/ml/processing/input"  
        ),  
    ],  
    outputs=[  
        ProcessingOutput(  
            output_name="output",  
            source="/opt/ml/processing/output"  
        ),  
    ],  
    code="tmp/preprocessing.py",  
)
```

SageMaker Pipelines

<https://github.com/PacktPublishing/Amazon-SageMaker-Cookbook>

```
from sagemaker.inputs import TrainingInput
from sagemaker.workflow.steps import TrainingStep

s3_input_data = step_process.properties.ProcessingOutputConfig.Outputs["output"].S3Output.S3Uri

step_train = TrainingStep(
    name="TrainStep",
    estimator=estimator,
    inputs={
        "train": TrainingInput(
            s3_data=s3_input_data,
            content_type="text/csv",
        )
    },
)
```

```
from sagemaker.workflow.pipeline import Pipeline

pipeline_name = f"Pipeline"
pipeline = Pipeline(
    name=pipeline_name,
    parameters=[
        processing_instance_type,
        training_instance_type,
        input_data,
    ],
    steps=[step_process, step_train],
)
```

```
pipeline.upsert(role_arn=role)
```

SAGEMAKER PROCESSING	SAGEMAKER EXPERIMENTS	AUTOMATED MODEL TUNING
BUILT-IN ALGORITHMS	ML AND DL FRAMEWORK SUPPORT	SAGEMAKER CLARIFY
SAGEMAKER DEBUGGER	SAGEMAKER FEATURE STORE	SAGEMAKER AUTOPILOT
SAGEMAKER STUDIO	SAGEMAKER GROUND TRUTH	SAGEMAKER MODEL MONITOR
MANAGED SPOT TRAINING	SAGEMAKER PIPELINES	SAGEMAKER DATA WRANGLER

Thanks!

FEEL FREE TO REACH OUT AND CONNECT:

EMAIL: joshua.arvin.lat@gmail.com

LINKEDIN: <https://www.linkedin.com/in/joshualat/>