



AWS Black Belt Online Seminar

Amazon SageMaker Training

ハンズオン編

本橋 和貴

Partner Solutions Architect, AI/ML
2022/07

© 2022, Amazon Web Services, Inc. or its affiliates.



つづき

1.
Hello World



2-1.
データの持ち込み

2-2.
ディレクトリ
の持ち込み



4.
ライブラリの
追加

3.
アーティファクト
を転送

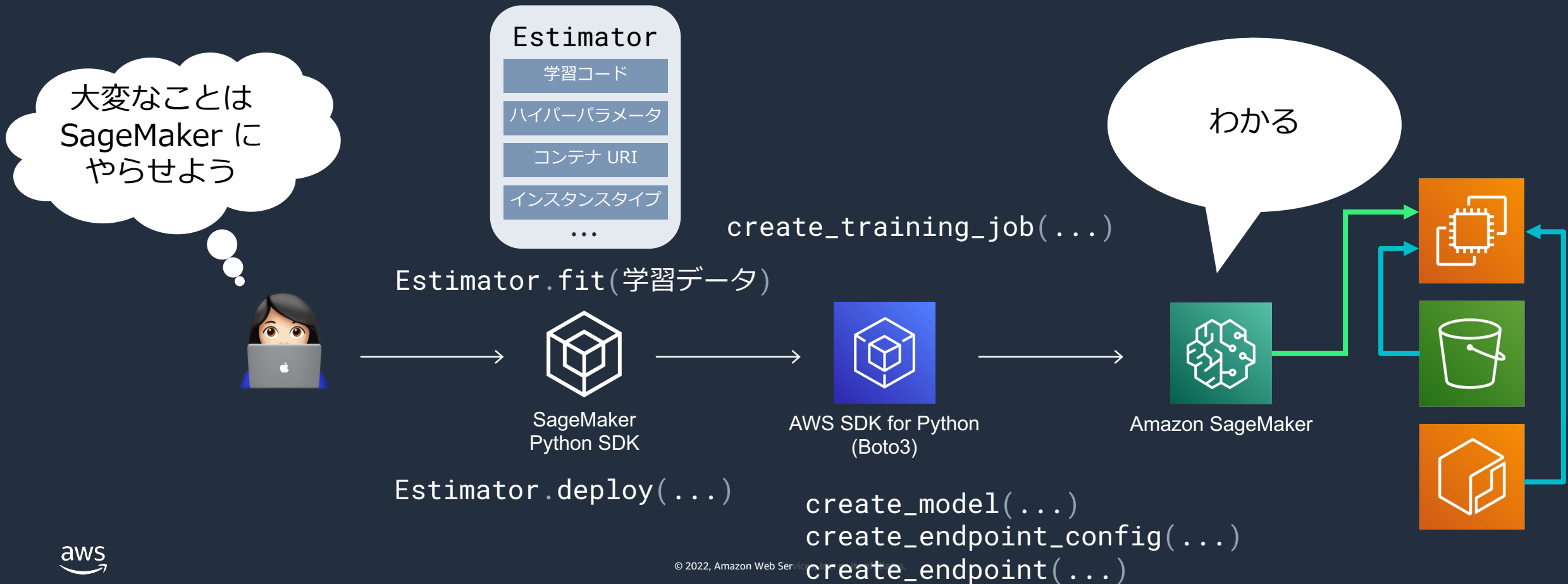


5.
ハイパーパラメータ
の指定



SageMaker Python SDK

- SageMaker でのモデル学習とかデプロイを簡素化する OSS の SDK

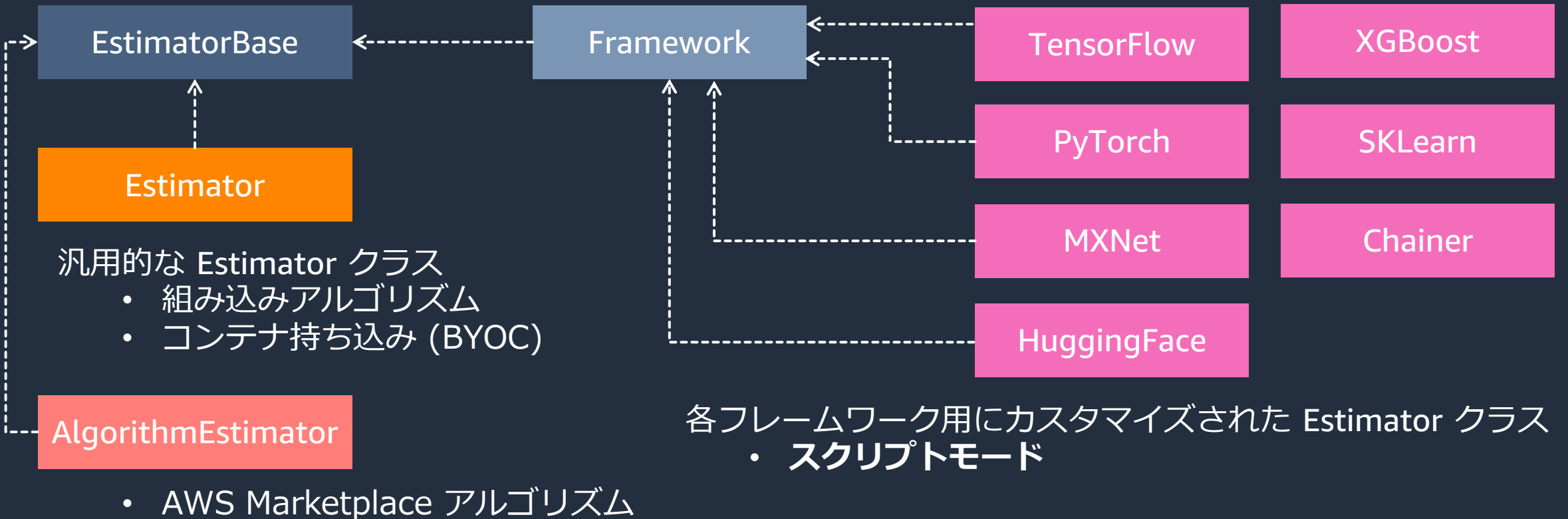


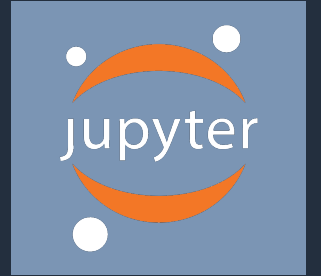
SageMaker でモデルをトレーニングする方法

- Jupyter Notebook 上で直接実行
 - 任意のライブラリ
- SageMaker Training Job で実行
 - 組み込みアルゴリズム
 - Linear Learner, XGBoost, k-NN, Random Cut Forest, ResNet, BrazeingText, ...
 - スクリプトモード
 - TensorFlow, MXNet, PyTorch, HuggingFace, SKLearn, etc
 - 独自コンテナの持ち込み (Bring Your Own Container)
 - AWS Marketplace アルゴリズム

Estimator – SageMaker Python SDK

- エンドツーエンドの学習とモデルのデプロイを制御するクラス





TensorFlow Estimator の使用例

```
from sagemaker.tensorflow import TensorFlow
```

```
tensorflow_estimator = TensorFlow(  
    entry_point='train.py',  
    hyperparameters={  
        'epochs': 20,  
        'batch-size': 64,  
        'learning-rate': 0.1  
    },  
    framework_version='2.6',  
    py_version='py38',  
    instance_type='ml.p3.2xlarge',  
    instance_count=1,  
)
```

学習コードの指定

学習コードに渡すハイパーパラメータの設定

学習コンテナの指定

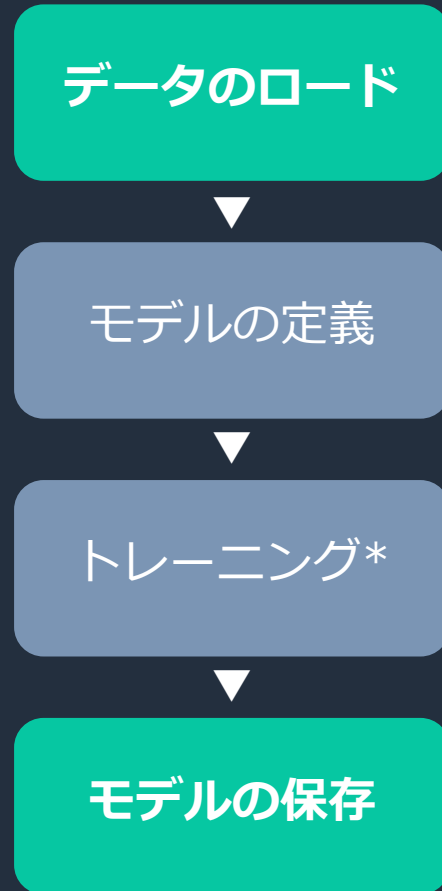
e.g.) 763104351884.dkr.ecr.us-east-1.amazonaws.com/tensorflow-training:2.6-gpu-py38

学習インスタンスの指定

```
tensorflow_estimator.fit({  
    'train': 's3://my-bucket/path/to/training/data',  
    'test': 's3://my-bucket/path/to/test/data'  
})
```

学習データの指定

一般的な学習コードの構成



SageMaker Training Job

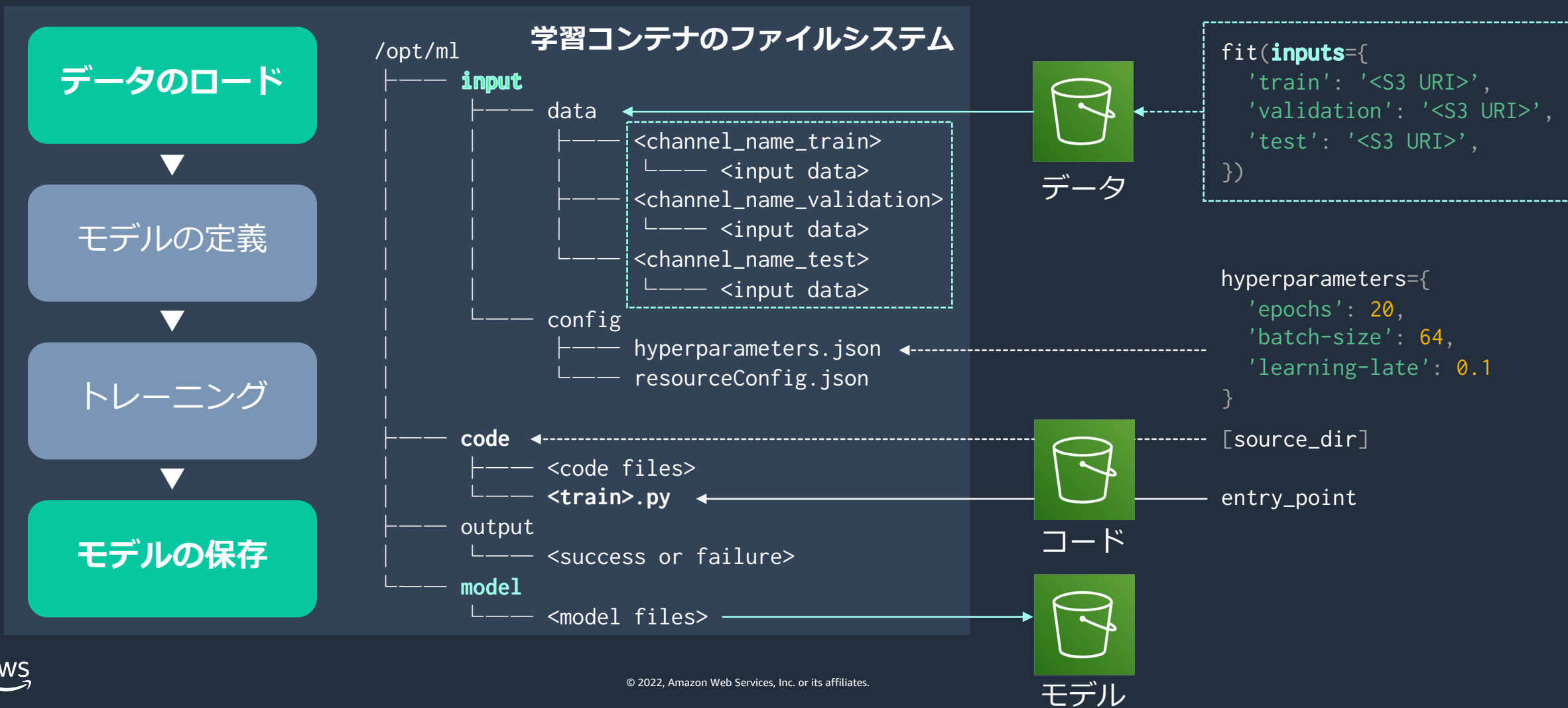
- [ジョブ開始時]: 学習データは S3 から自動ダウンロード
- [ジョブ終了時]: 学習済みモデルは S3 に自動アップロード

コードの書き換えが最低限必要なのは

- **データをどのパスからロードするか**
- **モデルをどのパスに保存するか**
- (*ハイパーパラメータは引数として argparse で取得可能)
- (*分散学習する場合はトレーニング部分も)

コード × コンテナ × SageMaker Python SDK

Estimator





Thank you!