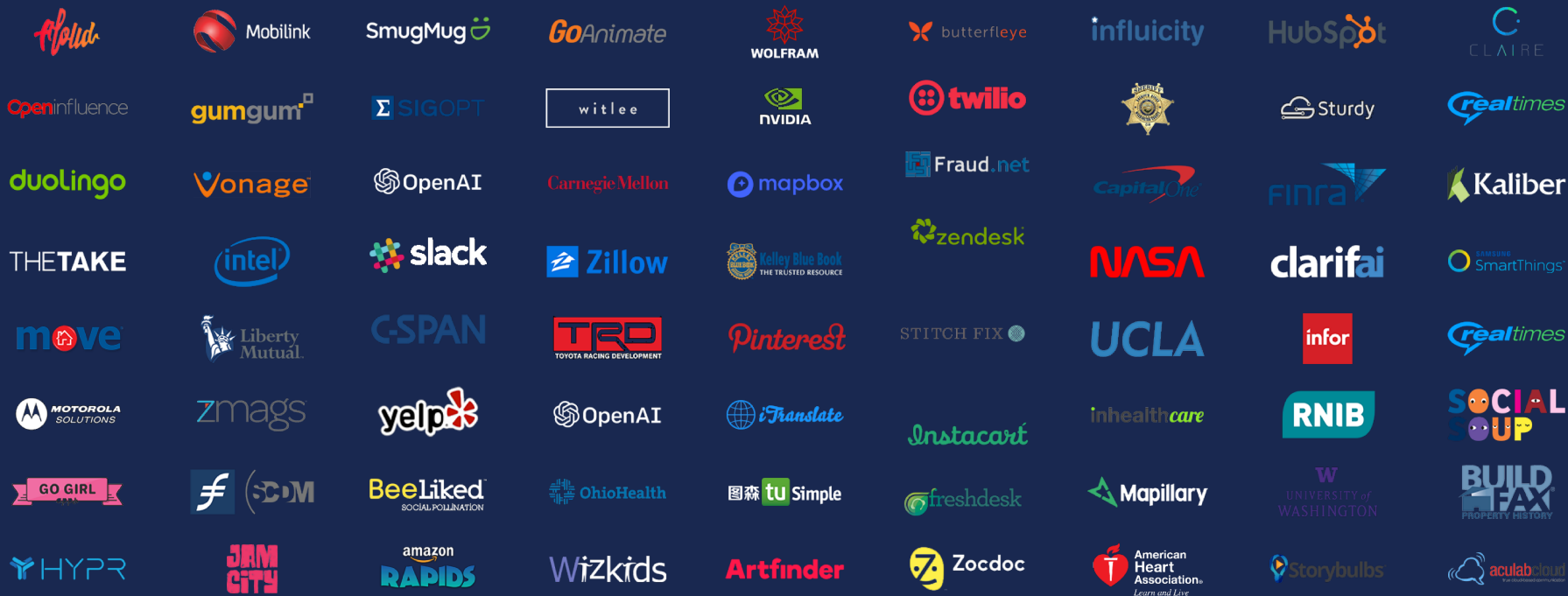




# Recommender Workshop

## Part 1: Introduction

# Customers Running Machine Learning On AWS Today

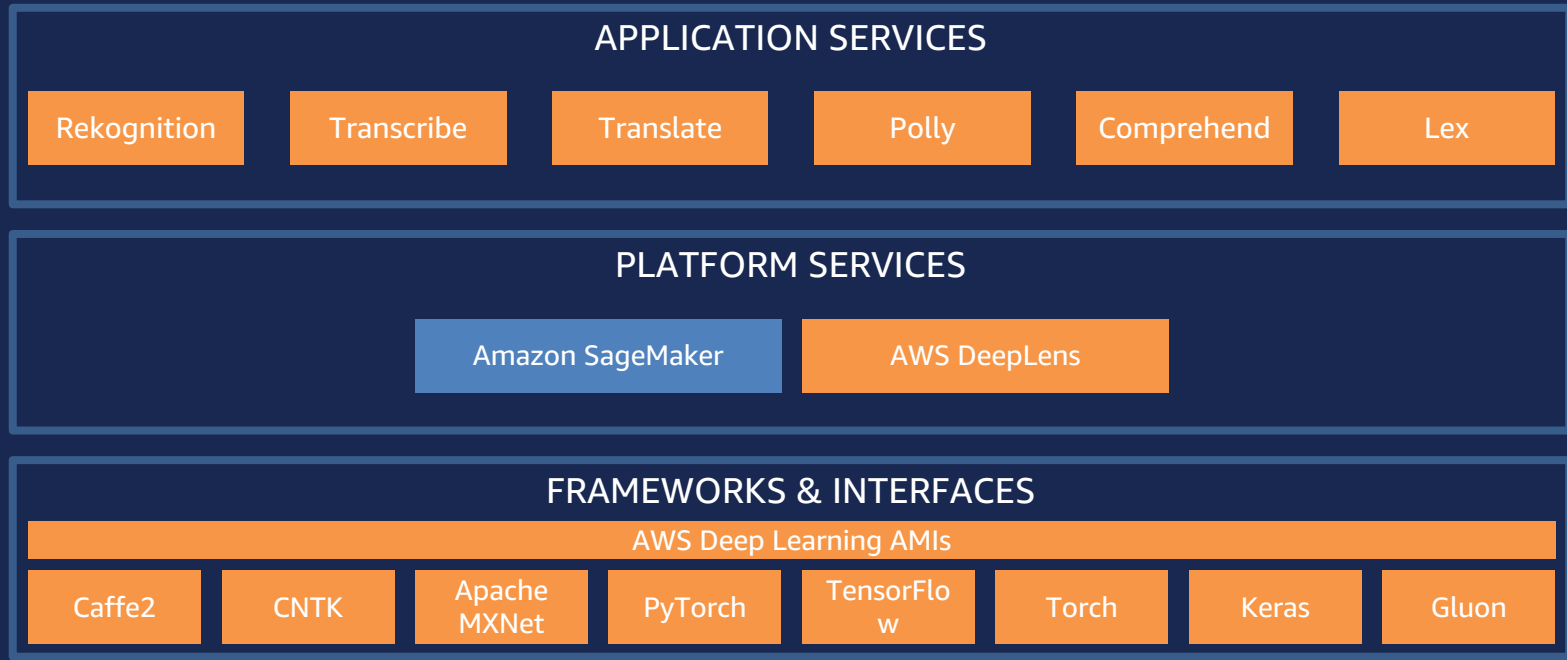


# Machine Learning at AWS

Our mission:

Put machine learning in the hands of every  
developer and data scientist

# The Amazon Machine Learning Stack



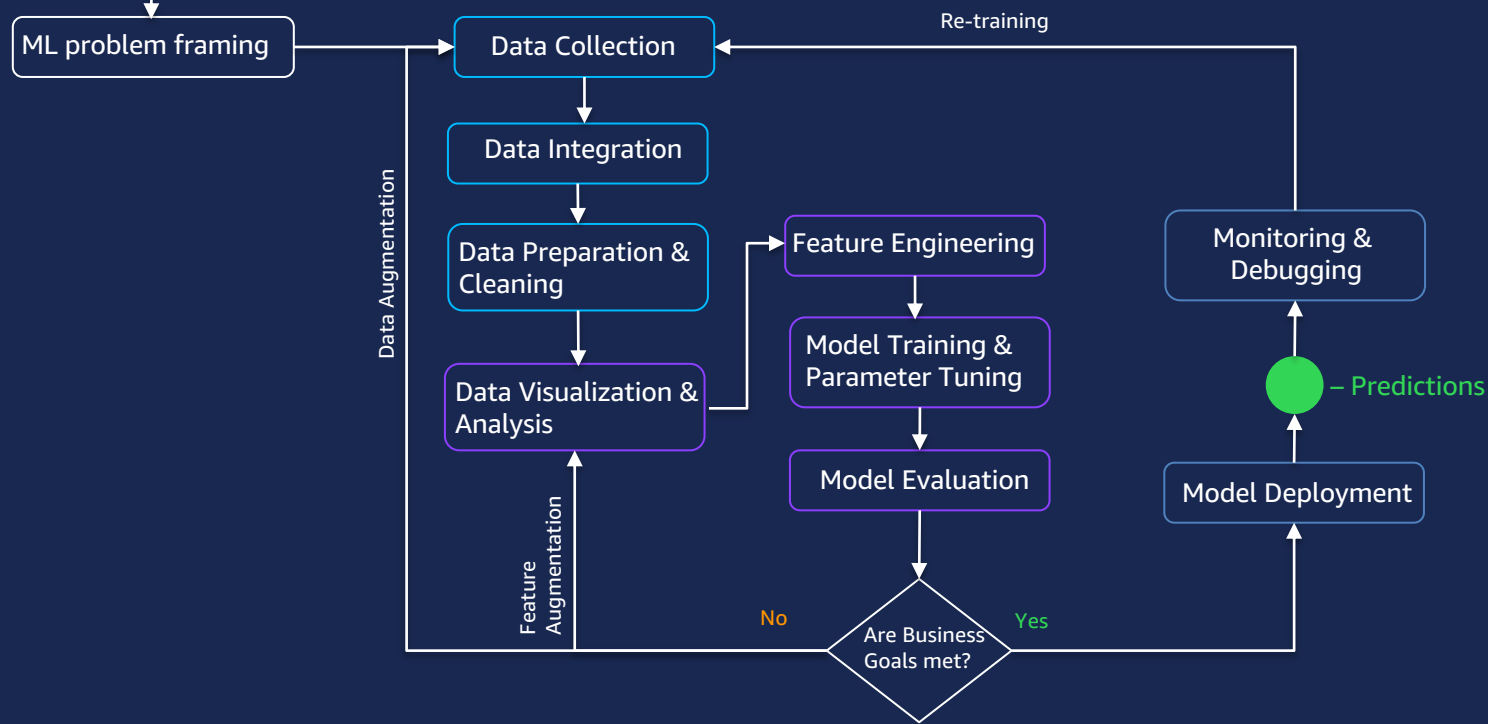
# Recommender Workshop Agenda

- Part 1: Introduction (You Are Here)
  - Overview of Machine Learning Process, Amazon SageMaker
  - Hands-on: Data Exploration
- Part 2: Collaborative Filtering
  - Core Concepts for Recommendations
  - Hands-on: K-Means Clustering
- Part 3: Matrix Factorization
  - Refining Recommendations
  - Hands-on: Factorization Machine
- Part 4: Hyperparameter Tuning
  - Key Concepts
  - Hands-on: Hyperparameter Tuning

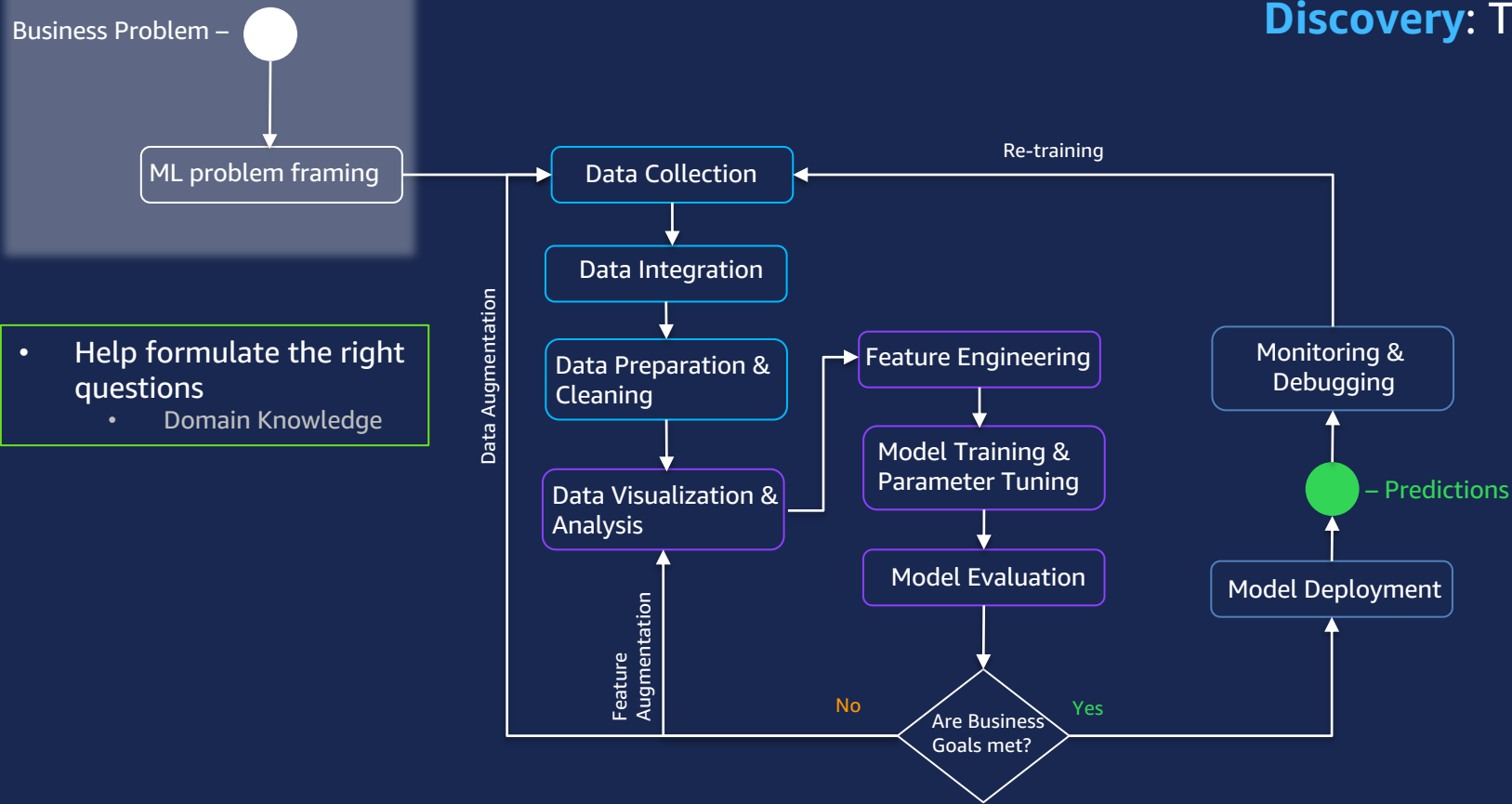
# Let's Review the ML Process

Business Problem – ●

# The Machine Learning Process



# Discovery: The Analysts





Business Problem –

## Integration: The Data Architecture

ML problem framing

- **Build the data platform:**

- Amazon S3
- AWS Glue
- Amazon Athena
- Amazon EMR
- Amazon Redshift Spectrum

Data Augmentation

Data Collection

Data Integration

Data Preparation & Cleaning

Data Visualization & Analysis

Feature Augmentation

Feature Engineering

Model Training & Parameter Tuning

Model Evaluation

Retraining

Monitoring & Debugging

– Predictions

Model Deployment

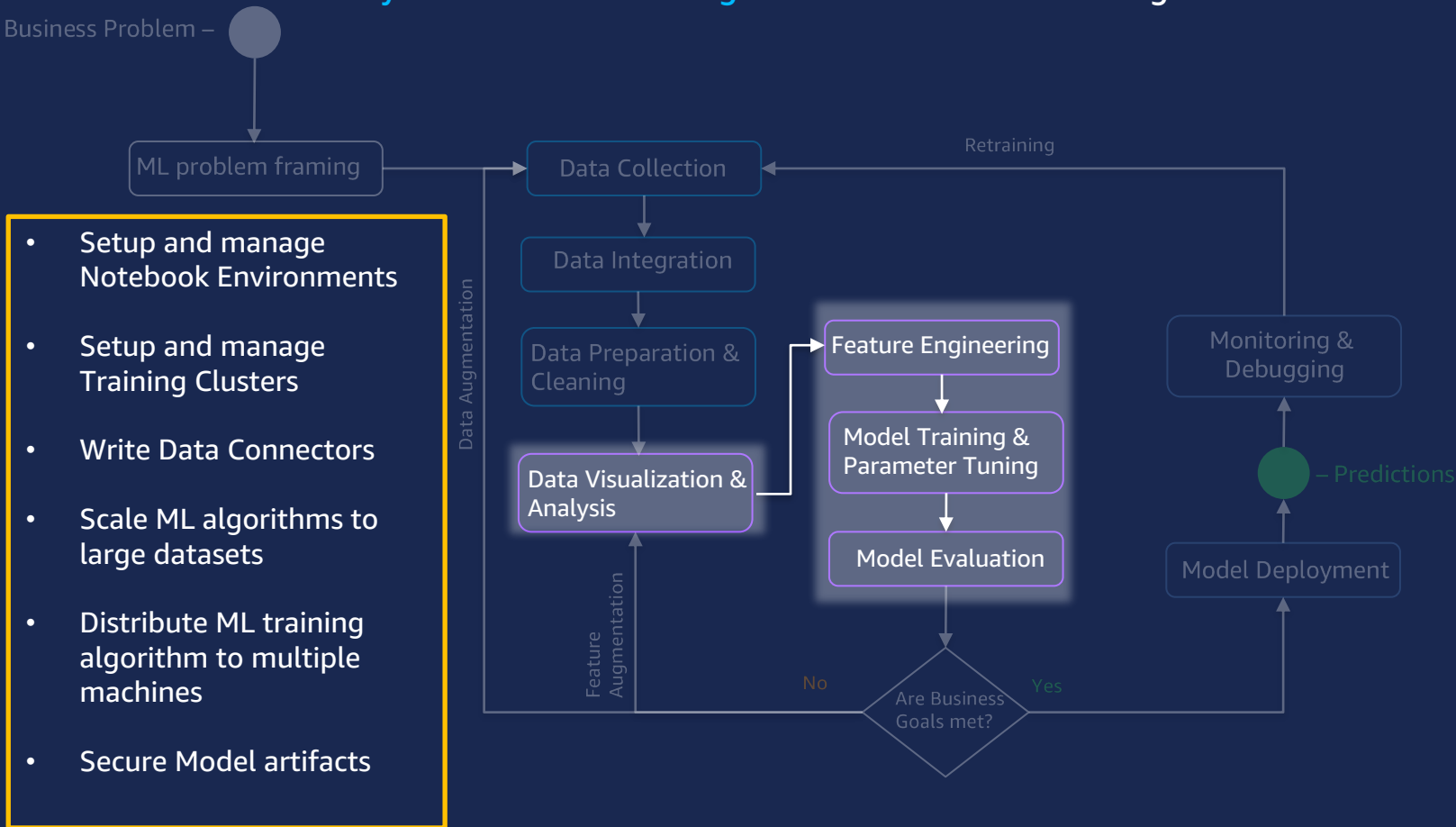
Are Business Goals met?

No

Yes

# Why We built Amazon SageMaker: The Model Training Undifferentiated Heavy Lifting

Business Problem –



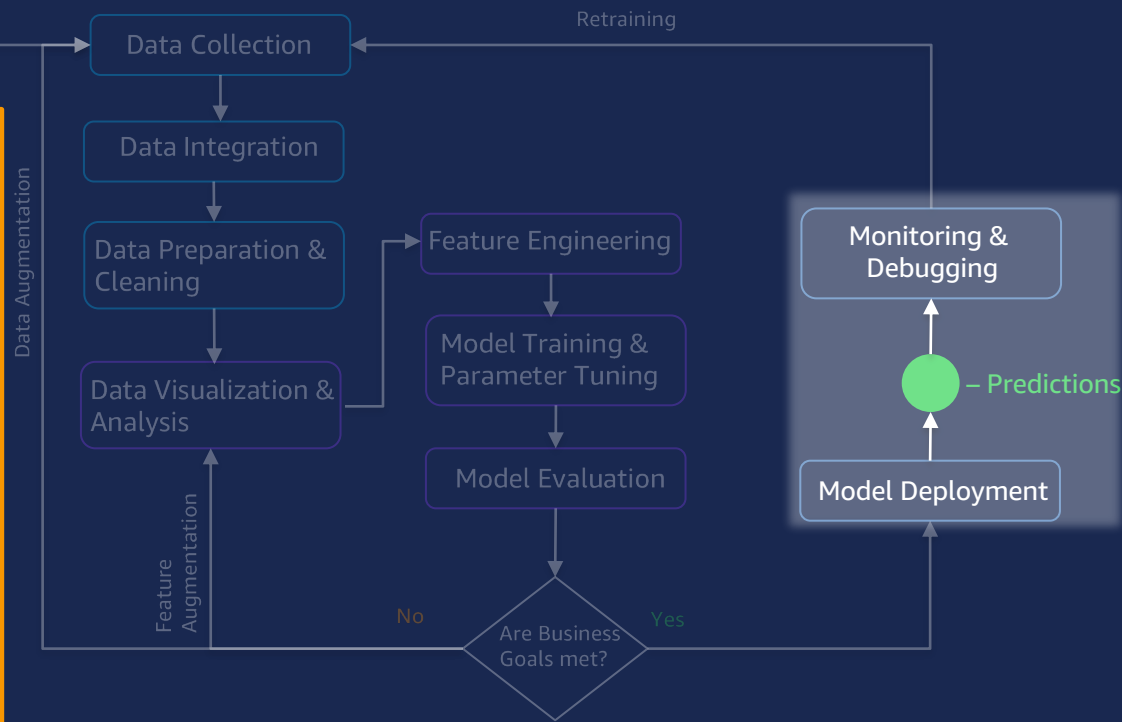
# Why We built Amazon SageMaker: The Model Deployment Undifferentiated Heavy Lifting

Business Problem –



ML problem framing

- Setup and manage Model Inference Clusters
- Manage and Scale Model Inference APIs
- Monitor and Debug Model Predictions
- Models versioning and performance tracking
- Automate New Model version promotion to production (A/B testing)

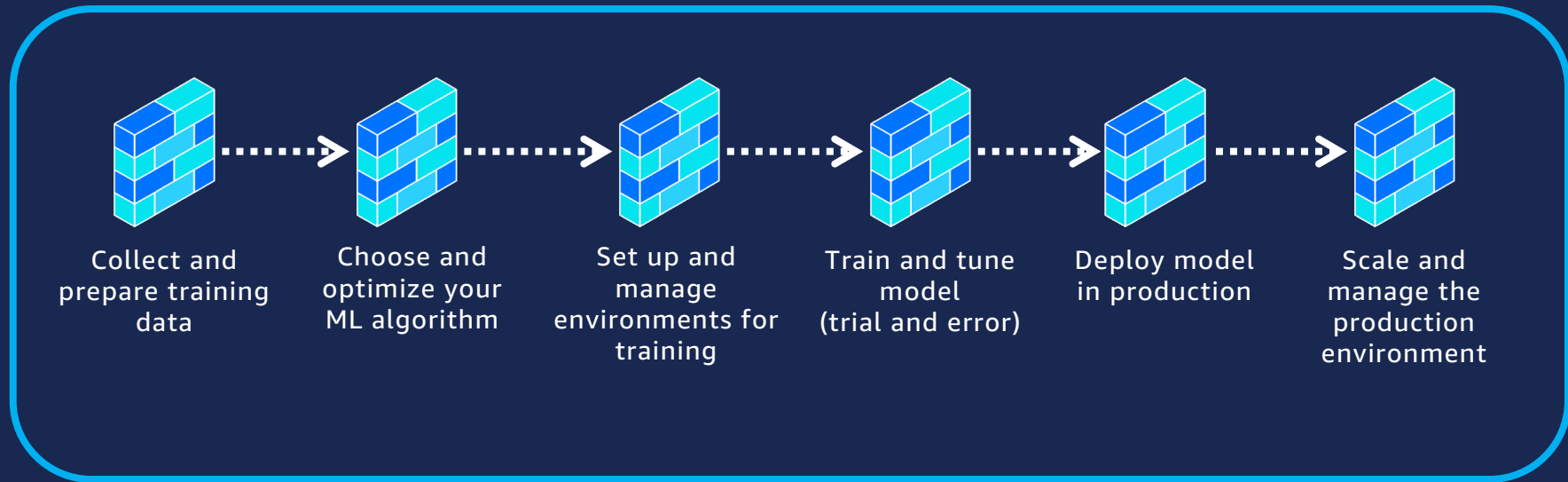




# Amazon SageMaker

# Amazon SageMaker

Easily build, train, and deploy machine learning models



# Amazon SageMaker

Easily build, train, and deploy machine learning models



Pre-built  
notebooks  
for common  
problems



Built-in, high  
performance  
algorithms

**BUILD**



Set up and  
manage  
environments  
for training



Train and  
tune model  
(trial and  
error)



Deploy model  
in production



Scale and  
manage the  
production  
environment

# Training ML Models Using Amazon SageMaker

## SageMaker Built-in Algorithms

- K-means Clustering
- PCA
- Neural Topic Modelling
- Factorisation Machines
- Linear Learner
- XGBoost
- Latent Dirichlet Allocation
- Image Classification
- Seq2Seq
- DeepAR Forecasting
- BlazingText (word2vec)
- Random Cut Forest
- kNN
- Object Detection

# Training ML Models Using Amazon SageMaker

## SageMaker Built-in Algorithms

K-means Clustering  
PCA  
Neural Topic Modelling  
Factorisation Machines  
Linear Learner – Regression  
XGBoost  
Latent Dirichlet Allocation  
Image Classification  
Seq2Seq  
Linear Learner –  
Classification  
DeepAR Forecasting

## Bring Your Own Algorithms

ML Algorithms  
R  
MXNet  
TensorFlow  
Caffe  
PyTorch  
Keras  
CNTK  
...



TensorFlow

PYTORCH



# Training ML Models Using Amazon SageMaker

## SageMaker Built-in Algorithms

K-means Clustering  
PCA  
Neural Topic Modelling  
Factorisation Machines  
Linear Learner – Regression  
XGBoost  
Latent Dirichlet Allocation  
Image Classification  
Seq2Seq  
Linear Learner – Classification  
DeepAR Forecasting

## Bring Your Own Algorithms

ML Algorithms  
R  
MXNet  
TensorFlow  
Caffe  
PyTorch  
Keras  
CNTK  
...

## DNN SDKs

TensorFlow SDK  
MXNet (Gluon) SDK  
Chainer SDK  
PyTorch SDK



# Training ML Models Using Amazon SageMaker

## SageMaker Built-in Algorithms

K-means Clustering  
PCA  
Neural Topic Modelling  
Factorisation Machines  
Linear Learner – Regression  
XGBoost  
Latent Dirichlet Allocation  
Image Classification  
Seq2Seq  
Linear Learner – Classification  
DeepAR Forecasting

## Bring Your Own Algorithms

ML Algorithms  
R  
MXNet  
TensorFlow  
Caffe  
PyTorch  
Keras  
CNTK  
...

## MXNet & TensorFlow SDK

TensorFlow SDK  
MXNet (Gluon) SDK

## Apache Spark Estimator

Apache Spark Python library  
Apache Spark Scala library



# Amazon SageMaker

Easily build, train, and deploy machine learning models



Pre-built  
notebooks  
for common  
problems



Built-in, high  
performance  
algorithms



One-click  
training



Hyperparameter  
optimization



Deploy model  
in production



Scale and  
manage the  
production  
environment

**BUILD**

**TRAIN**

# Amazon SageMaker

Easily build, train, and deploy machine learning models



Pre-built  
notebooks  
for common  
problems



Built-in, high  
performance  
algorithms



One-click  
training



Hyperparameter  
optimization



One-click  
deployment



Fully managed  
hosting with  
auto-scaling

**BUILD**

**TRAIN**

**DEPLOY**

# Recommender Workshop Repository

<http://bit.ly/2wkaV0N>

# Our Data Set: Movielens

- Public Data Set produced by **GroupLens Research**
- <https://grouplens.org/datasets/movielens/>

```
In [15]: data = pd.read_csv("u.data", sep='\t', header=None,  
                           names=[ 'userid', 'movieid', 'rating', 'timestamp' ] )  
data.head()
```

Out[15]:

	userid	movieid	rating	timestamp
0	196	242	3	881250949
1	186	302	3	891717742
2	22	377	1	878887116
3	244	51	2	880606923
4	166	346	1	886397596

# Recommender Workshop Activity

- Find the [Amazon SageMaker](#) service, then find [Notebooks](#)
- Spin up new notebook instance
  - Recommended: `ml.m4.xlarge` type
- Within notebook instance, open Terminal
  - `cd SageMaker`
  - `git clone https://github.com/shirkeyaws/sagemaker-recommender-workshop` (aka <http://bit.ly/2wkaV0N>)
- In Jupyter, within the repo path, find: `01_exploring_data.ipynb`

# NEXT: Part 2