AWS re:Invent

NOV. 28 – DEC. 2, 2022 | LAS VEGAS, NV

AIM208

Idea to production on Amazon SageMaker, with Thomson Reuters

Kimberly Madia (she/her)
Sr. Manager, AI/ML Product Marketing
AWS

Denis V. Batalov, PhD
WW Tech Leader, AI/ML
AWS

Maria Apazoglou
Vice President
Thomson Reuters

Agenda

What is Amazon SageMaker?

Amazon SageMaker key capabilities

Demo Amazon SageMaker Studio notebooks

Demo Amazon SageMaker Canvas

Why do customers use Amazon SageMaker?

Thomson Reuters’ perspective on Amazon SageMaker

Five year of Amazon SageMaker

2017

Easily build machine learning models with performance-optimized algorithms

Fast, fully managed training

Deploy models into production with one click

2018

Unstructured data preparation

2019

World's first IDE

Studio notebooks

Experiments

Autopilot

Debugger

Model Monitor

2020

Data Wrangler

JumpStart

Feature Store

Distributed training

Pipelines

Edge Manager

Clarify

2021

No Code ML with Canvas

Massive scale

DATA LABELING	1M+ labeling tasks/day
SINGLE CUSTOMER DEPLOYMENT	Million+ models
MODEL TRAINING	Billions of parameters
PREDICTIONS	100B+/month

Business analysts

Make ML predictions using a visual interface with Amazon SageMaker Canvas

Data scientists

Prepare data and build, train, and deploy ML models with Amazon SageMaker Studio

Amazon SageMaker

Infrastructure, tools, visual interfaces, workflows, orchestration, and collaboration

MLOps engineers

Deploy and manage models at scale with Amazon SageMaker MLOps

Build ML models

Fully managed shareable notebooks on Amazon EC2

Fully managed, sharable Jupyter notebooks
Run notebooks on elastic compute resources

Built-in algorithms
Many built-in algorithms available in prebuilt container images

Prebuilt solutions and open-source models
Over 150 popular open-source models

AutoML
Automatically create ML models with full visibility

Support for major frameworks and toolkits
Optimized for popular deep learning (DL) frameworks such as TensorFlow, PyTorch, Apache MXNet, and Hugging Face

Train ML models

Fast and cost-effective ML model training

Experiment management and model tuning
Save weeks of effort by automatically tracking training runs and tuning hyperparameters

Debug and profile training runs
Use real-time metrics to correct performance problems

Distributed training
Complete distributed training up to 40% faster

Training compiler
Accelerate training times by up to 50% through more efficient use of GPUs

Managed spot training
Reduce the costs of training by up to 90%

Deploy ML models

Fully managed deployment for inference at scale

Wide selection of infrastructure
70+ instance types with varying levels of compute and memory to meet the needs of every use case

Single-digit millisecond overhead latency
For use cases requiring real-time responses

Asynchronous inference
Supports large models with long-running processing times

Cost-effective deployment
Multi-model/multi-container endpoints, serverless inference, and elastic scaling

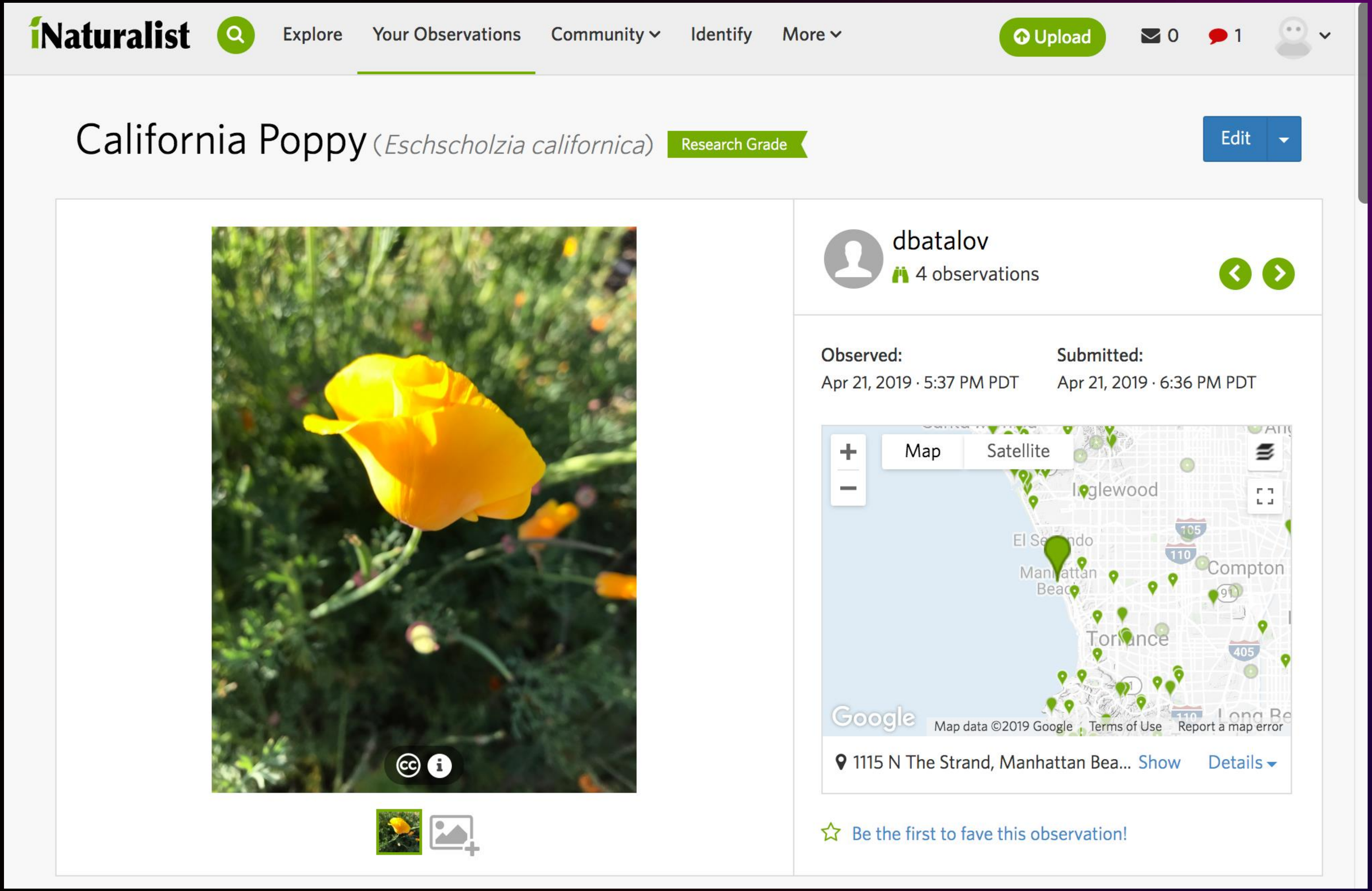
Built-in integration for MLOps
ML workflows, CI/CD, lineage tracking, and catalog

Automatic deployment recommendations
Optimal instance type/count and container parameters, and fully managed load testing

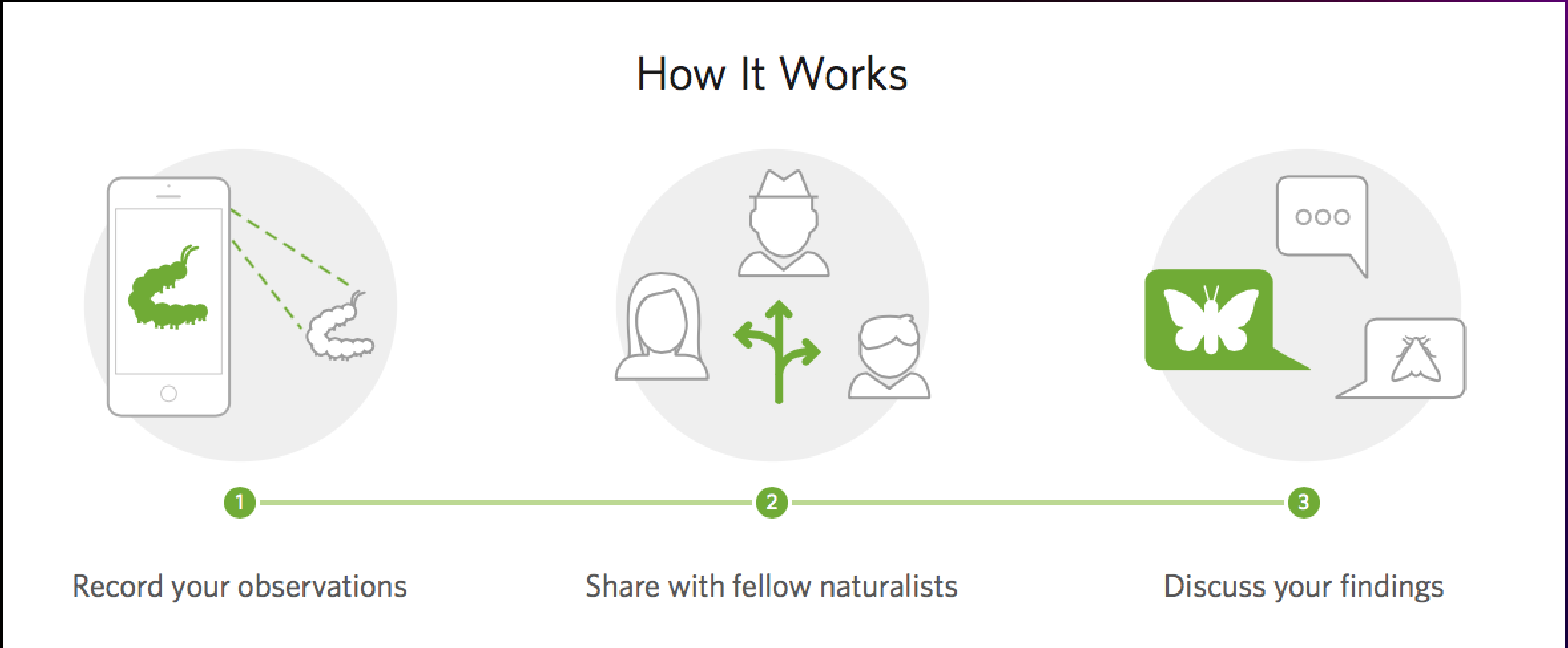
Objective: Detect bees in images

Training data?

iNaturalist.org



iNaturalist.org



iNaturalist.org allows export of a dataset

500 images of bees under Creative Commons CC0 license

https://creativecommons.org/share-your-work/public-domain/cc0/

Amazon SageMaker Ground Truth to label data

Ground Truth

Set up and manage labeling jobs for highly accurate training datasets using active learning and human labeling.

Notebook

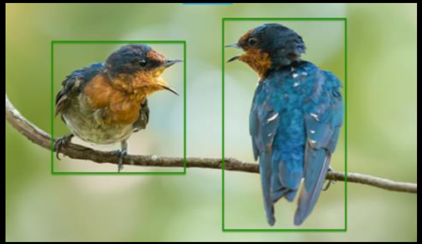
Availability of AWS and SageMaker SDKs and sample notebooks to create training Jobs and deploy models.

Training

Train and tune models at any scale. Leverage high performance AWS algorithms or bring your own.

Inference

Create models from training jobs or import external models for hosting to run inferences on new data.

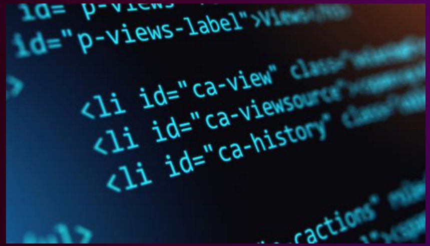


Bounding boxes

Image classification

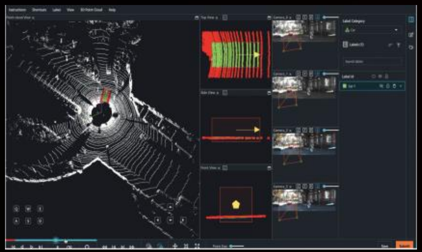
Segmentation

Label verification

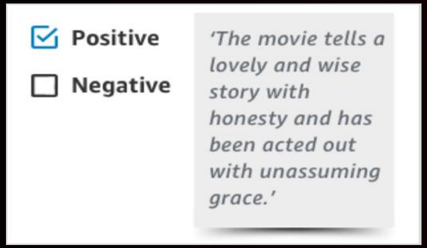


Custom template

Video

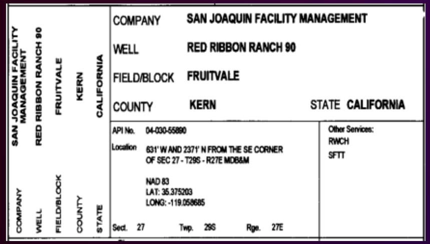


LIDAR 3D Point Cloud



Text classification

Named entity recognition



OCR, form, table

Amazon SageMaker Studio

File Edit View Run Kernel Git Tabs Settings Help

/ demo /

- images
- validation.manifest
- train.manifest
- bee_detector.ipynb

SageMaker Bee Detection Demo

Table of contents

STEP 1: Review of labeled dataset

In this workshop we will use a dataset from the inaturalist.org This dataset contains 500 images of bees that have been uploaded by inaturalist users for the purposes of recording the observation and identification. We only used images that their users have licensed under CC0 license. For your convenience, we have placed the dataset in S3 in a single zip archive here: http://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/DIG-TF-200-MLBEES-10-EN/dataset.zip

The dataset has already been labeled with SageMaker Ground Truth, which means we have an augmented manifest file that contains the results of labeling.

Furthermore, the dataset has already been downloaded and unzipped inside the notebook under the `images` folder. The archive contains the following structure: 500 `.jpg` image files, the already mentioned augmented manifest file `output.manifest` and 10 test images in the `test` subfolder.

Let's review results of the previously completed labeling job called `bees-500` using the provided augmented manifest. This file contains

Python 3 (Data Science) | Idle Kernel: Idle | Instance MEM Mode: Command Ln 1, Col 1 bee_detector.ipynb

SageMaker Bee Detection Demo

Table of contents

STEP 1: Review of labeled dataset

In this workshop we will use a dataset from the inaturalist.org This dataset contains 500 images of bees that have been uploaded by inaturalist users for the purposes of recording the observation and identification. We only used images that their users have licensed under CC0 license. For your convenience, we have placed the dataset in S3 in a single zip archive here: http://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/DIG-TF-200-MLBEES-10-EN/dataset.zip

The dataset has already been labeled with SageMaker Ground Truth, which means we have an augmented manifest file that contains the results of labeling.

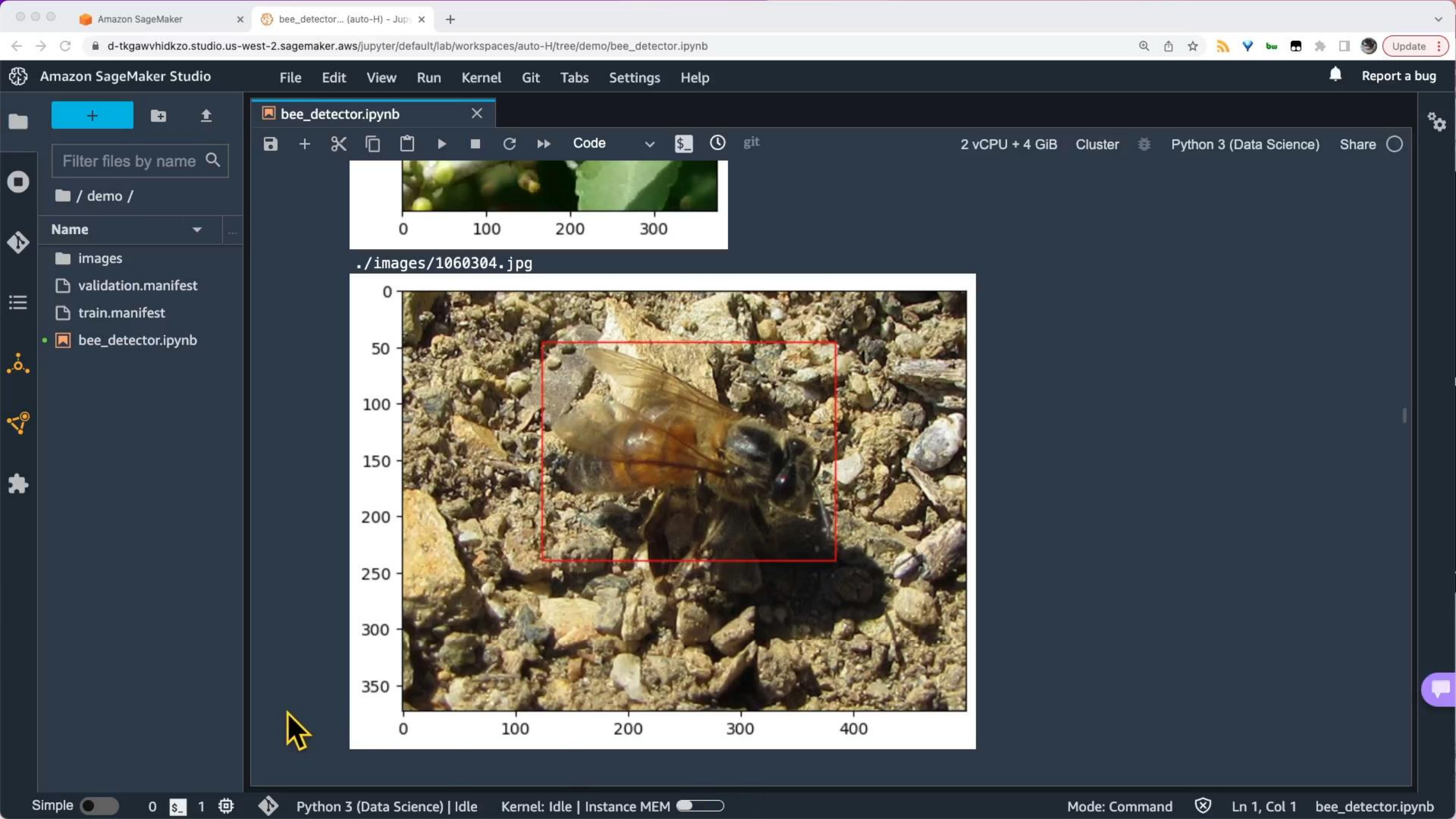
Furthermore, the dataset has already been downloaded and unzipped inside the notebook under the `images` folder. The archive contains the following structure: 500 `.jpg` image files, the already mentioned augmented manifest file `output.manifest` and 10 test images in the `test` subfolder.

Let's review results of the previously completed labeling job called `bees-500` using the provided augmented manifest. This file contains

Let's review results of the previously completed labeling job called `bees-500` using the provided augmented manifest. This file contains 500 lines, each of which contains JSON descriptor of the results of labeling one image. Let's look at the first one.

[2]:
```python
augmented_manifest_file = './images/output.manifest'
labeling_job_name = 'bees-500'

import json

file = open(augmented_manifest_file, 'r')
json_line = json.loads(file.readline())
print(json.dumps(json_line, indent=4))
```


```
{
    "source-ref": "s3://sagemaker-remars/datasets/na-bees/500/10006450.jpg",
    "bees-500": {
        "annotations": [
            {
                "class_id": 0,
                "width": 95.39999999999998,
                "top": 256.2,
                "height": 86.80000000000001,
                "left": 177
            }
        ],
        "image_size": [
            {
                "width": 500,
                "depth": 3,
                "height": 500
            }
        ]
```


Amazon SageMaker
bee_detector... (auto-H) - Jupy
d-tkgawvhidkzo.studio.us-west-2.sagemaker.aws/jupyter/default/lab/workspaces/auto-H/tree/demo/bee_detector.ipynb
Update
Amazon SageMaker Studio
File
Edit
View
Run
Kernel
Git
Tabs
Settings
Help
Report a bug
bee_detector.ipynb
Code
git
2 vCPU + 4 GiB
Cluster
Python 3 (Data Science)
Share
Filter files by name
/ demo /
Name
images
validation.manifest
train.manifest
bee_detector.ipynb
0
100
200
300
./images/1060304.jpg
0
50
100
150
200
250
300
350
0
100
200
300
400
Simple
0
1
Python 3 (Data Science) | Idle
Kernel: Idle | Instance MEM
Mode: Command
Ln 1, Col 1
bee_detector.ipynb

Amazon SageMaker | bee_detector... (auto-H) - Jupy

d-tkgawvhidkzo.studio.us-west-2.sagemaker.aws/jupyter/default/lab/workspaces/auto-H/tree/demo/bee_detector.ipynb

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

Filter files by name

/ demo /

Name
- images
- validation.manifest
- train.manifest
- bee_detector.ipynb

bee_detector.ipynb

Code   2 vCPU + 4 GiB   Cluster   Python 3 (Data Science)   Share

```
            f.write(json.dumps(line))
            f.write('\n')

print(f'training samples: {num_training_samples}, validation samples: {len(lines)-num_training_samples}')
```

```
training samples: 400, validation samples: 100
```

STEP 2.2 Upload training and validation manifests to S3

Next, let's upload the two manifest files to S3 in preparation for training. We will use the same bucket you created earlier.

[6]:

```
import sagemaker
# S3 bucket must be created in the current region
BUCKET = sagemaker.Session().default_bucket()
PREFIX = 'reinvent-artifacts' # this is the root path to our working space

pfx_training = PREFIX + '/training' if PREFIX else 'training'
# Defines paths for use in the training job request.
s3_train_data_path = 's3://{}/{}/{}'.format(BUCKET, pfx_training, augmented_manifest_filename_train)
s3_validation_data_path = 's3://{}/{}/{}'.format(BUCKET, pfx_training, augmented_manifest_filename_validation)

!aws s3 cp train.manifest s3://$BUCKET/$pfx_training/
!aws s3 cp validation.manifest s3://$BUCKET/$pfx_training/
```

```
upload: ./train.manifest to s3://sagemaker-us-west-2-187004457496/reinvent-artifacts/training/train.manifest
upload: ./validation.manifest to s3://sagemaker-us-west-2-187004457496/reinvent-artifacts/training/validation.manif
est
```

STEP 2.3 Configure and kick off training job

We are now ready to kick off the training job. Since we will use a built-in object detection algorithm, you can do the following steps via

Simple   0   1   Python 3 (Data Science) | Idle   Kernel: Idle | Instance MEM   Mode: Command   Ln 1, Col 1   bee_detector.ipynb

```
est
```

STEP 2.3 Configure and kick off training job

We are now ready to kick off the training job. Since we will use a built-in object detection algorithm, you can do the following steps via SageMaker Console UI, but since we are already in the notebook, we can just run the following code with SageMaker Python SDK.

[24]:

```
import boto3
client = boto3.client('sagemaker')

role = sagemaker.get_execution_role()
sess = sagemaker.Session()

training_image = sagemaker.image_uris.retrieve('object-detection', boto3.Session().region_name, version='1')
s3_output_path = 's3://{}/{}/output'.format(BUCKET, pfx_training)

model = sagemaker.estimator.Estimator(
    training_image,
    role,
    instance_count=1,
    instance_type="ml.p3.2xlarge",
    input_mode="Pipe",
    output_path=s3_output_path,
    sagemaker_session=sess,
)

# full list of hyperparameters and defaults are here: https://docs.aws.amazon.com/sagemaker/latest/dg/object-detect
model.set_hyperparameters(
    base_network="resnet-50",
    use_pretrained_model=1, # use transfer learning
    num_classes=1, # only bees
```

```
    num_training_samples=num_training_samples,
)

train_data = sagemaker.inputs.TrainingInput(
    s3_train_data_path,
    distribution="FullyReplicated",
    content_type="application/x-recordio",
    record_wrapping="RecordIO",
    s3_data_type="AugmentedManifestFile",
    attribute_names=['source-ref', labeling_job_name]
)

validation_data = sagemaker.inputs.TrainingInput(
    s3_validation_data_path,
    distribution="FullyReplicated",
    content_type="application/x-recordio",
    record_wrapping="RecordIO",
    s3_data_type="AugmentedManifestFile",
    attribute_names=['source-ref', labeling_job_name]
)

data_channels = {"train": train_data, "validation": validation_data}
```

All configuration is done and we are now ready to kick off the training job!

```
[25]: model.fit(inputs=data_channels, wait=False)
```

To check the progess of the training job, you can refresh the console or repeatedly evaluate the following cell. When the training job status reads `'Completed'`, move on to the next part of the tutorial.

```
[27]: training_info = client.describe_training_job(TrainingJobName=model.latest_training_job.name)
```

Amazon SageMaker | bee_detector... (auto-H) - Jup

us-west-2.console.aws.amazon.com/sagemaker/home?region=us-west-2#/jobs/object-detection-2022-11-22-04-10-53-813

aws | Services | Search [Option+S] | Oregon | Admin/denisb-Isengard @ 1870-0445-7496

Amazon SageMaker

Getting started

Control panel
- Studio
- Studio Lab
- Canvas
- RStudio

SageMaker dashboard
- Images
- Lifecycle configurations
- Search

▶ Ground Truth

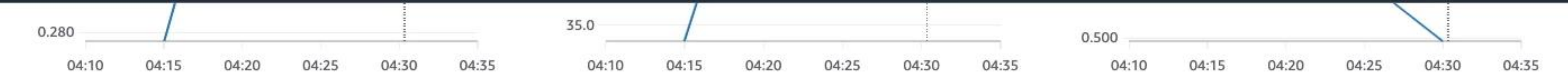
▶ Notebook

▶ Processing

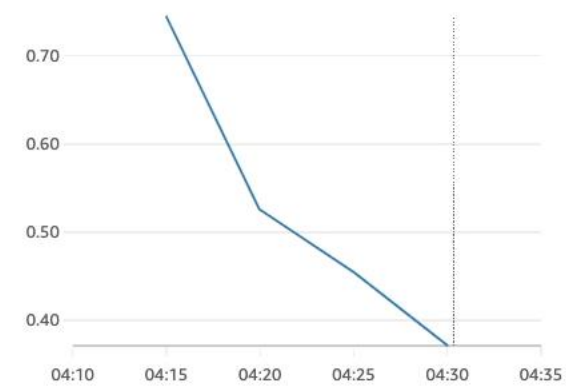
▶ Training

▶ Inference

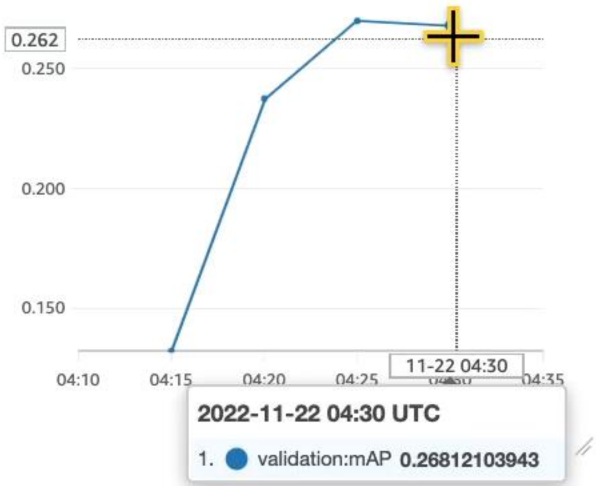
▶ Edge Manager



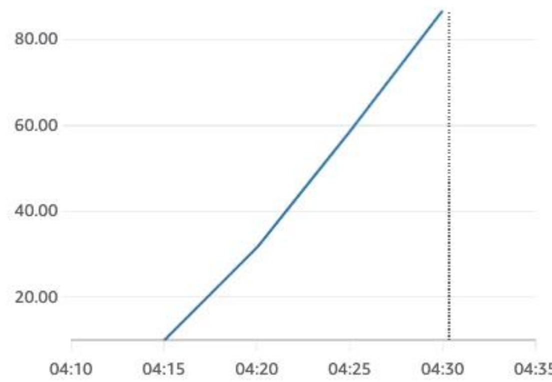
train:smooth_l1



validation:mAP



train:progress



Output

S3 model artifact

Amazon SageMaker | bee_detector... (auto-H) - Jup

d-tkgawvhidkzo.studio.us-west-2.sagemaker.aws/jupyter/default/lab/workspaces/auto-H/tree/demo/bee_detector.ipynb

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

Filter files by name

/ demo /

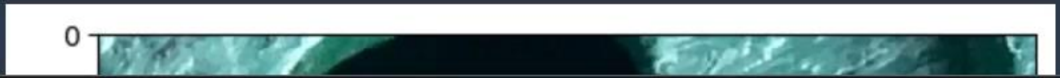
Name

- images
- validation.manifest
- train.manifest
- bee_detector.ipynb

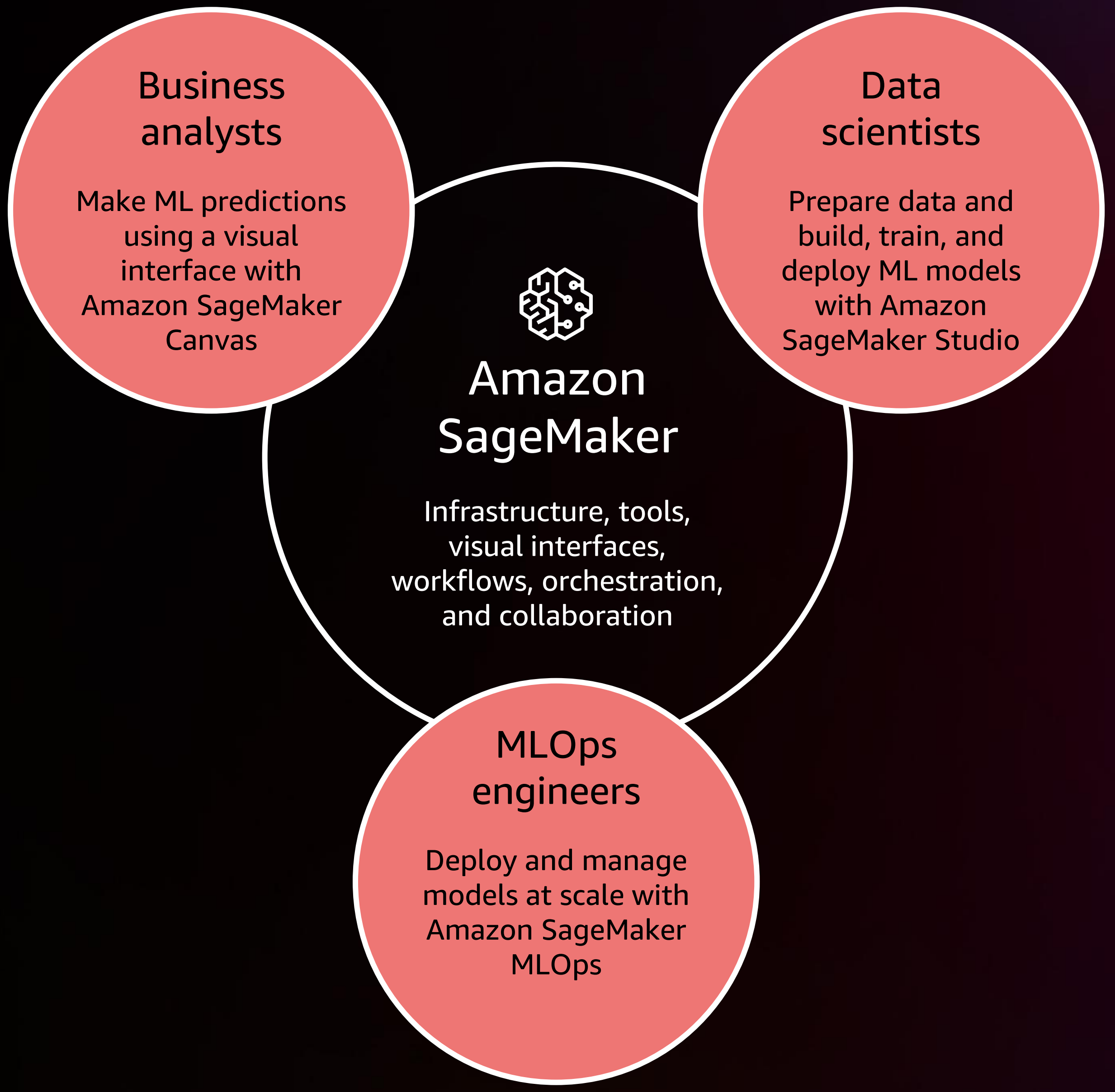
bee_detector.ipynb

Code 2 vCPU + 4 GiB Cluster Python 3 (Data Science) Share

```python
[31]: import matplotlib.pyplot as plt

endpoint_name=bee_detector.endpoint_name

runtime_client = boto3.client('sagemaker-runtime')

# Call SageMaker endpoint to obtain predictions
def get_predictions_for_img(runtime_client, endpoint_name, img_path):
    with open(img_path, 'rb') as f:
        payload = f.read()
        payload = bytearray(payload)

    response = runtime_client.invoke_endpoint(EndpointName=endpoint_name,
                                              ContentType='application/x-image',
                                              Body=payload)

    result = response['Body'].read()
    result = json.loads(result)
    return result


for test_image in test_images:
    result = get_predictions_for_img(runtime_client, endpoint_name, test_image)
    confidence_threshold = .2
    best_n = 3
    # display the best n predictions with confidence > confidence_threshold
    predictions = [prediction for prediction in result['prediction'] if prediction[1] > confidence_threshold]
    predictions.sort(reverse=True, key = lambda x: x[1])
    bboxes = [prediction_to_bbox_data(test_image, prediction) for prediction in predictions[:best_n]]
    show_annotated_image(test_image, bboxes)
```



Simple 0 1 Python 3 (Data Science) | Idle Kernel: Idle | Instance MEM Mode: Edit Ln 25, Col 15 bee_detector.ipynb


Business analysts
Make ML predictions using a visual interface with Amazon SageMaker Canvas
Data scientists
Prepare data and build, train, and deploy ML models with Amazon SageMaker Studio
Amazon SageMaker
Infrastructure, tools, visual interfaces, workflows, orchestration, and collaboration
MLOps engineers
Deploy and manage models at scale with Amazon SageMaker MLOps

Amazon SageMaker Canvas

Generate ML predictions – no code required

- Quickly access and prepare data sources for ML
- AutoML built in to generate accurate predictions
- Share ML models with data science teams

Colorectal cancer survivability

https://seer.cancer.gov/

By National Atlas of the United States:
https://commons.wikimedia.org/w/index.php?curid=50967902

SageMaker Canvas

d-pdovqdwnoub1.studio.us-west-2.sagemaker.aws/canvas/default/datasets

Amazon SageMaker

Canvas

Models

Datasets

Datasets

Dataset

Join data

Import

Name	Source	Columns	Rows	Cells	Created	Status
canvas-sample-loans-part-2.csv	S3	5	1,000	5,000	10/29/2022 2:50 PM	Ready
canvas-sample-maintenance.csv	S3	9	1,000	9,000	10/29/2022 2:50 PM	Ready
canvas-sample-diabetic-readmission.csv	S3	16	1,000	16,000	10/29/2022 2:50 PM	Ready
canvas-sample-shipping-logs.csv	S3	12	1,000	12,000	10/29/2022 2:50 PM	Ready
canvas-sample-housing.csv	S3	10	1,000	10,000	10/29/2022 2:50 PM	Ready
canvas-sample-product-descriptions.csv	S3	5	120	600	10/29/2022 2:50 PM	Ready
canvas-sample-sales-forecasting.csv	S3	5	1,000	5,000	10/29/2022 2:50 PM	Ready
canvas-sample-loans-part-1.csv	S3	19	1,000	19,000	10/29/2022 2:50 PM	Ready

Help

Log out

SageMaker Canvas

d-pdovqdwnoub1.studio.us-west-2.sagemaker.aws/canvas/default/models/My%20Model

My Model

V1 • Draft | Add version | Share

Select | Build | Analyze | Predict

Select a column to predict

Choose the target column. The model that you build predicts values for the column that you select.

Target column

Model type

SageMaker Canvas automatically recommends the appropriate model type for your analysis.

To see a recommended model type, specify a value for the target column.

Quick build

Preview model

KentuckySEER.csv

Random sample: 20.0k rows

Extract | Remove rows by | Replace | Functions | Data visualizer

Column name	Data type	Missing	Mismatched	Unique	Mean / Mode
Year of diagnosis	Numeric	0.00% (0)	0.00% (0)	18	2,000
Tumor Grade (thru 2017)	Categorical	0.00% (0)	0.00% (0)	5	Moderately differenti...
Survived >5yrs	Binary	0.00% (0)	0.00% (0)	2	Yes
Summary Stage	Categorical	0.00% (0)	0.00% (0)	5	Localized
Sex	Binary	0.00% (0)	0.00% (0)	2	Female
Race	Categorical	0.00% (0)	0.00% (0)	5	White
Marital status	Categorical	0.00% (0)	0.00% (0)	7	Married (including co...
ICD-Primary Site	Categorical	0.00% (0)	0.00% (0)	11	C18.7-Sigmoid colon

Total columns: 10 | Total rows: 23,981 | Total cells: 239,810 | Show dropped columns

SageMaker Canvas

d-pdovqdwnoub1.studio.us-west-2.sagemaker.aws/canvas/default/models/My%20Model

My Model

V1 Draft | Add version | Share

Select | Build | Analyze | Predict

Select a column to predict

Choose the target column. The model that you build predicts values for the column that you select.

Target column

Model type

SageMaker Canvas automatically recommends the appropriate model type for your analysis.

To see a recommended model type, specify a value for the target column.

Quick build

Preview model

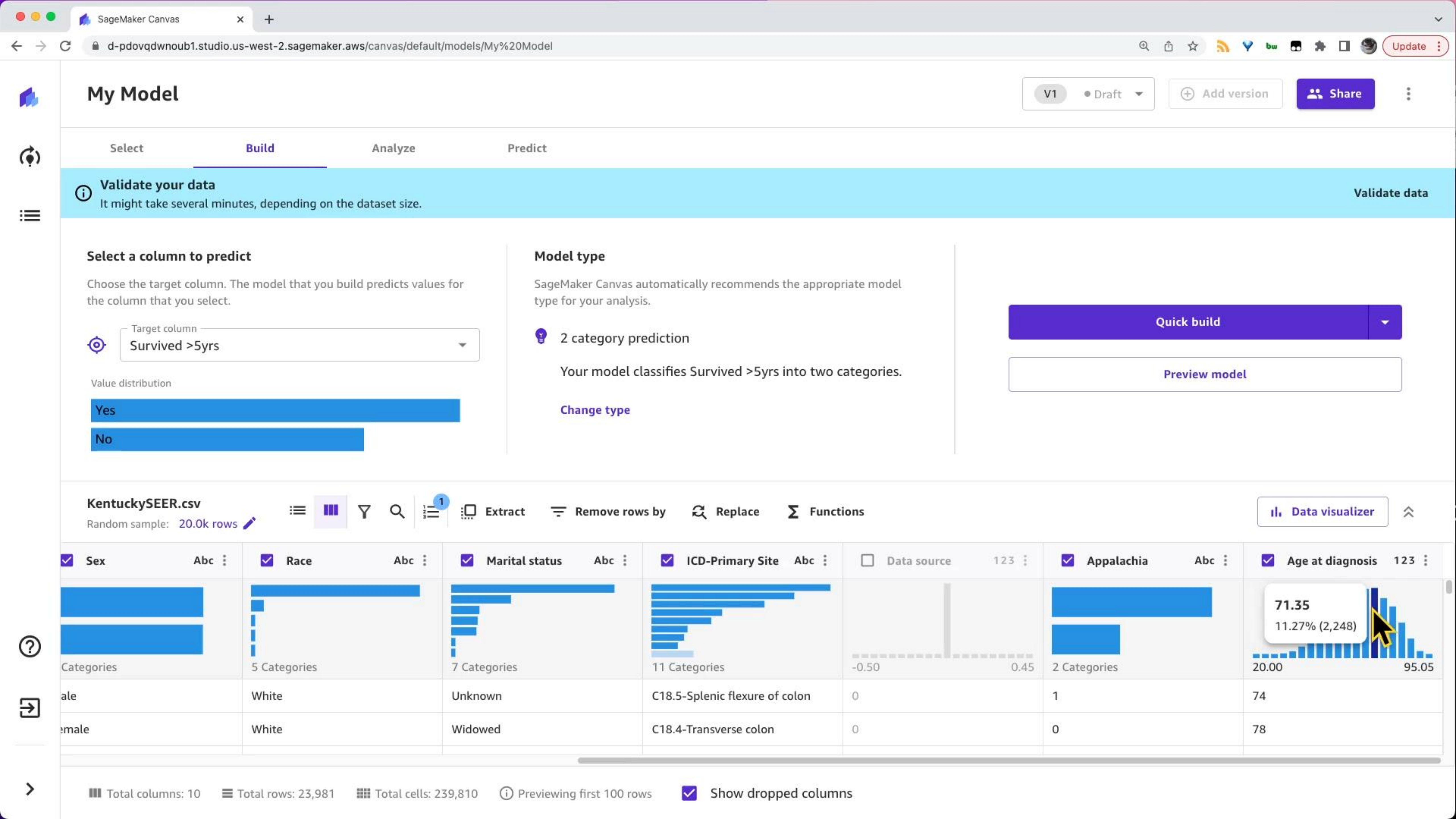
KentuckySEER.csv

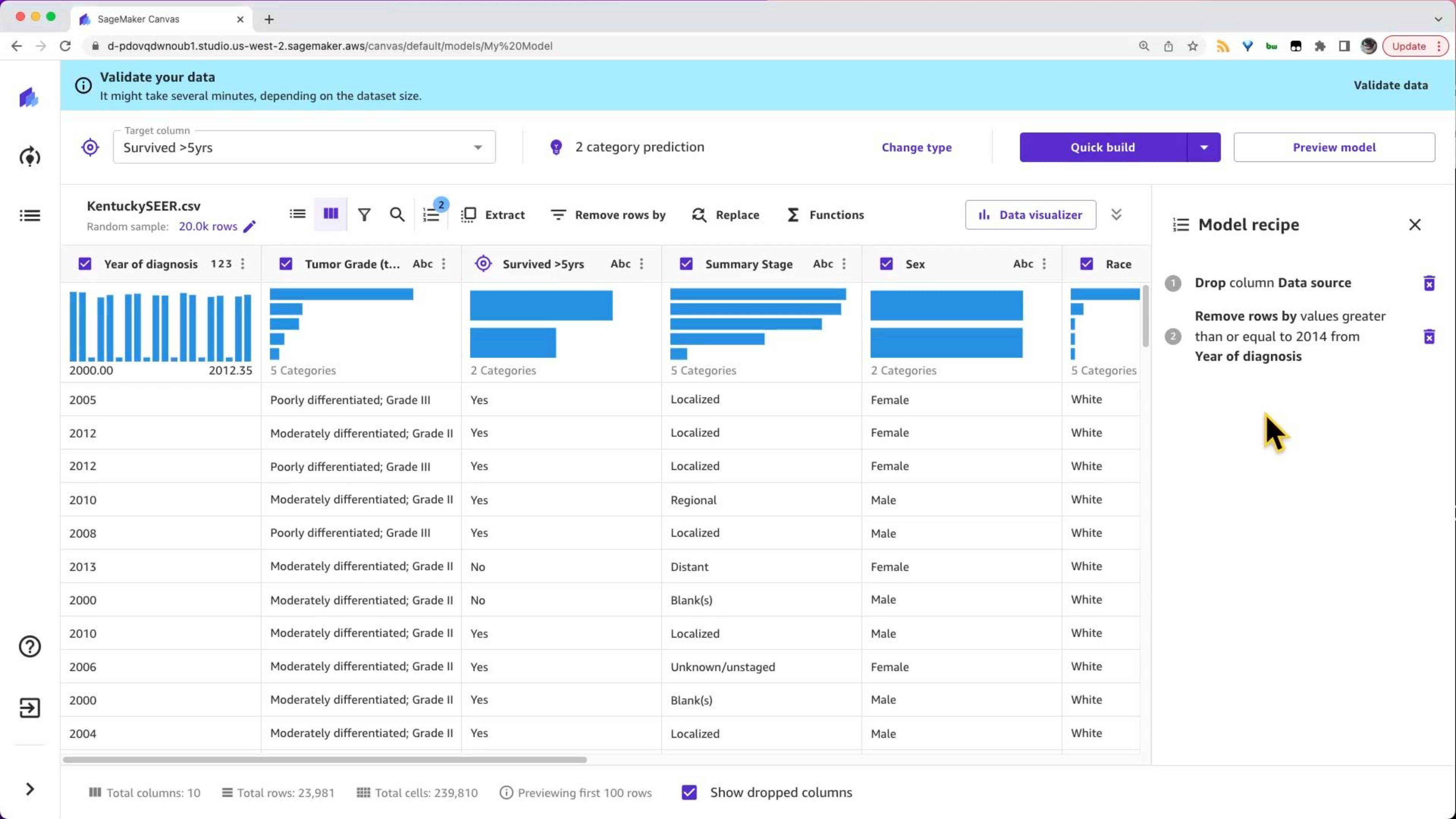
Random sample: 20.0k rows

Extract | Remove rows by | Replace | Functions | Data visualizer

Column name	Data type	Missing	Mismatched	Unique	Mean / Mode
Year of diagnosis	Numeric	0.00% (0)	0.00% (0)	18	2,000
Tumor Grade (thru 2017)	Categorical	0.00% (0)	0.00% (0)	5	Moderately differenti…
Survived >5yrs	Binary	0.00% (0)	0.00% (0)	2	Yes
Summary Stage	Categorical	0.00% (0)	0.00% (0)	5	Localized
Sex	Binary	0.00% (0)	0.00% (0)	2	Female
Race	Categorical	0.00% (0)	0.00% (0)	5	White
Marital status	Categorical	0.00% (0)	0.00% (0)	7	Married (including co…
ICD-Primary Site	Categorical	0.00% (0)	0.00% (0)	11	C18.7-Sigmoid colon

Total columns: 10 | Total rows: 23,981 | Total cells: 239,810 | Show dropped columns


SageMaker Canvas
d-pdovqdwnoub1.studio.us-west-2.sagemaker.aws/canvas/default/models/My%20Model
Update
My Model
V1
Draft
Add version
Share
Select
Build
Analyze
Predict
Validate your data
It might take several minutes, depending on the dataset size.
Validate data
Select a column to predict
Choose the target column. The model that you build predicts values for the column that you select.
Target column
Survived >5yrs
Value distribution
Yes
No
Model type
SageMaker Canvas automatically recommends the appropriate model type for your analysis.
2 category prediction
Your model classifies Survived >5yrs into two categories.
Change type
Quick build
Preview model
KentuckySEER.csv
Random sample: 20.0k rows
Extract
Remove rows by
Replace
Functions
Data visualizer
Sex Abc
Race Abc
Marital status Abc
ICD-Primary Site Abc
Data source 123
Appalachia Abc
Age at diagnosis 123
71.35
11.27% (2,248)
Categories
5 Categories
7 Categories
11 Categories
-0.50
0.45
2 Categories
20.00
95.05
ale
White
Unknown
C18.5-Splenic flexure of colon
0
1
74
emale
White
Widowed
C18.4-Transverse colon
0
0
78
Total columns: 10
Total rows: 23,981
Total cells: 239,810
Previewing first 100 rows
Show dropped columns



SageMaker Canvas
d-pdovqdwnoub1.studio.us-west-2.sagemaker.aws/canvas/default/models/My%20Model
Update
Validate your data
It might take several minutes, depending on the dataset size.
Validate data
Target column
Survived >5yrs
2 category prediction
Change type
Quick build
Preview model
KentuckySEER.csv
Random sample: 20.0k rows
Extract
Remove rows by
Replace
Functions
Data visualizer
Model recipe
1 Drop column Data source
2 Remove rows by values greater than or equal to 2014 from Year of diagnosis
Year of diagnosis 123
Tumor Grade (t... Abc
Survived >5yrs Abc
Summary Stage Abc
Sex Abc
Race
2000.00
2012.35
5 Categories
2 Categories
5 Categories
2 Categories
5 Categories
2005 Poorly differentiated; Grade III Yes Localized Female White
2012 Moderately differentiated; Grade II Yes Localized Female White
2012 Poorly differentiated; Grade III Yes Localized Female White
2010 Moderately differentiated; Grade II Yes Regional Male White
2008 Poorly differentiated; Grade III Yes Localized Male White
2013 Moderately differentiated; Grade II No Distant Female White
2000 Moderately differentiated; Grade II No Blank(s) Male White
2010 Moderately differentiated; Grade II Yes Localized Male White
2006 Moderately differentiated; Grade II Yes Unknown/unstaged Female White
2000 Moderately differentiated; Grade II Yes Blank(s) Male White
2004 Moderately differentiated; Grade II Yes Localized Male White
Total columns: 10
Total rows: 23,981
Total cells: 239,810
Previewing first 100 rows
Show dropped columns

SageMaker Canvas

d-pdovqdwnoub1.studio.us-west-2.sagemaker.aws/canvas/default/models/My%20Model

My Model

V1 · Ready · Add version · Share

Select | Build | **Analyze** | Predict

Model status

81.386%

The model predicts the correct Survived >5yrs 81.386% of the time.

Predict

Share with SageMaker Studio

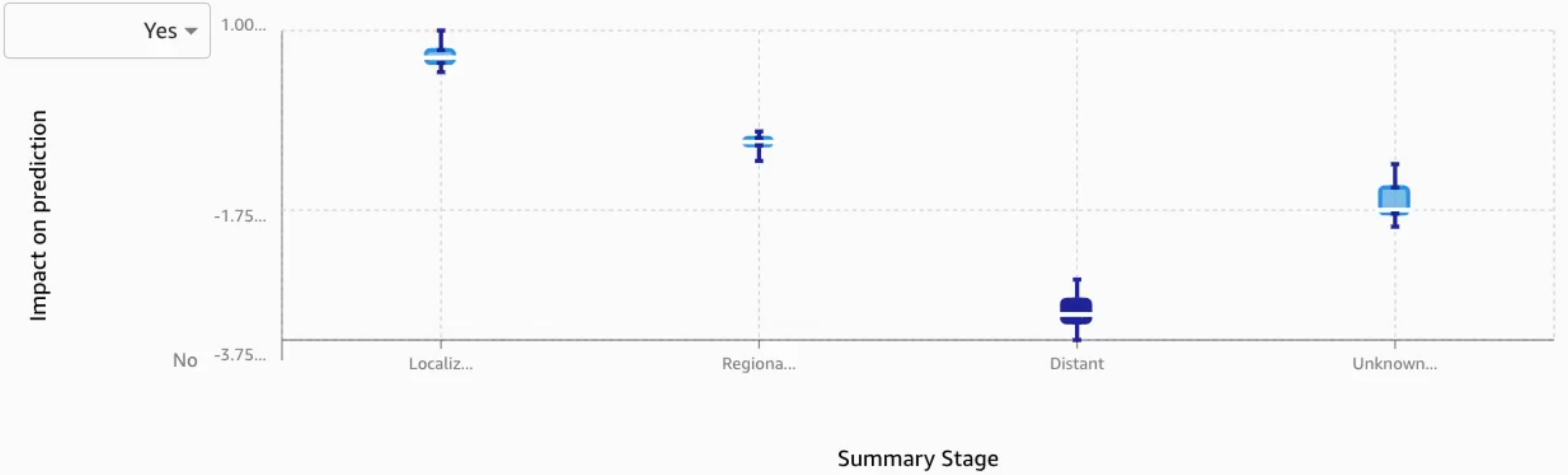
Overview | Scoring

Column impact

Search columns...

#	Column	Impact
1	Summary Stage	50.988%
2	Age at diagnosis	24.244%
3	Tumor Grade (thru 2017)	9.318%
4	Sex	3.93%
5	Marital status	3.355%
6	Year of diagnosis	3.194%

Impact of Summary Stage on prediction of Survived >5yrs



KentuckySEER.csv · Total columns: 10 · Total rows: 23,981 · Total cells: 239,810 · Survived >5yrs · 2 category prediction

Predict

SageMaker Canvas

d-pdovqdwnoub1.studio.us-west-2.sagemaker.aws/canvas/default/models/My%20Model

My Model

V1 ● Ready | Add version | Share

Select | Build | **Analyze** | Predict

Model status

81.386%

The model predicts the correct Survived >5yrs 81.386% of the time.

Predict

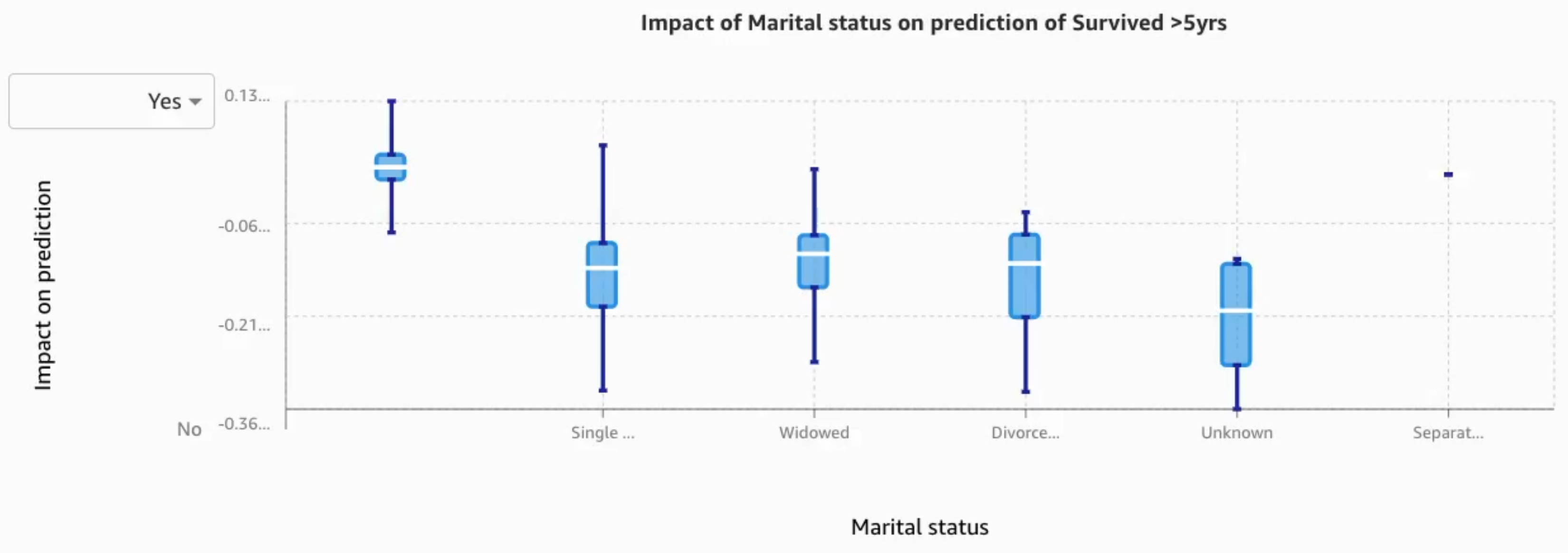
Share with SageMaker Studio

Overview | Scoring

Column impact

Search columns...

1	Summary Stage	50.988%
2	Age at diagnosis	24.244%
3	Tumor Grade (thru 2017)	9.318%
4	Sex	3.93%
5	Marital status	3.355%
6	Year of diagnosis	3.194%



KentuckySEER.csv | Total columns: 10 | Total rows: 23,981 | Total cells: 239,810 | Survived >5yrs | 2 category prediction

Predict

Business analysts

Make ML predictions using a visual interface with Amazon SageMaker Canvas

Data scientists

Prepare data and build, train, and deploy ML models with Amazon SageMaker Studio

Amazon SageMaker

Infrastructure, tools, visual interfaces, workflows, orchestration, and collaboration

MLOps engineers

Deploy and manage models at scale with Amazon SageMaker MLOps

Amazon SageMaker MLOps

Streamline the ML lifecycle

- Automate ML workflows to scale model development
- Build CI/CD pipelines for ML to accelerate model deployment
- Catalog model versions, metadata, metrics, and approvals for traceability and reusability
- Track lineage for troubleshooting and compliance
- Maintain accuracy of predictions after models are deployed
- Enhance governance and security

Machine learning at Thomson Reuters

"Using AWS services, like Amazon SageMaker, we can create our own customized solutions while tapping into core ML functionalities."

Maria Apazoglou

VP, AI/ML and BI Platforms

Thomson Reuters

History of machine learning at Thomson Reuters

1992
WIN (Westlaw Is Natural)

"Try the only legal research service that lets you search in plain English!"

1996
History Assistant

NLP system that analyzed case law documents to find historical relationships between court decisions

. . .

Today

Deep learning center in Boston


REUTERS World Business Legal Markets Breakingviews Technology Investigations More

Trends in privacy & data security: Looking back at 2021 and ahead to 2022

By Thomson Reuters Institute

A man types on a computer keyboard in Warsaw in this February 28, 2013 illustration file picture. REUTERS/Kacper Pempel

Gender and Racial Stereotype Detection in Legal Opinion Word Embeddings
In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI-2022)*

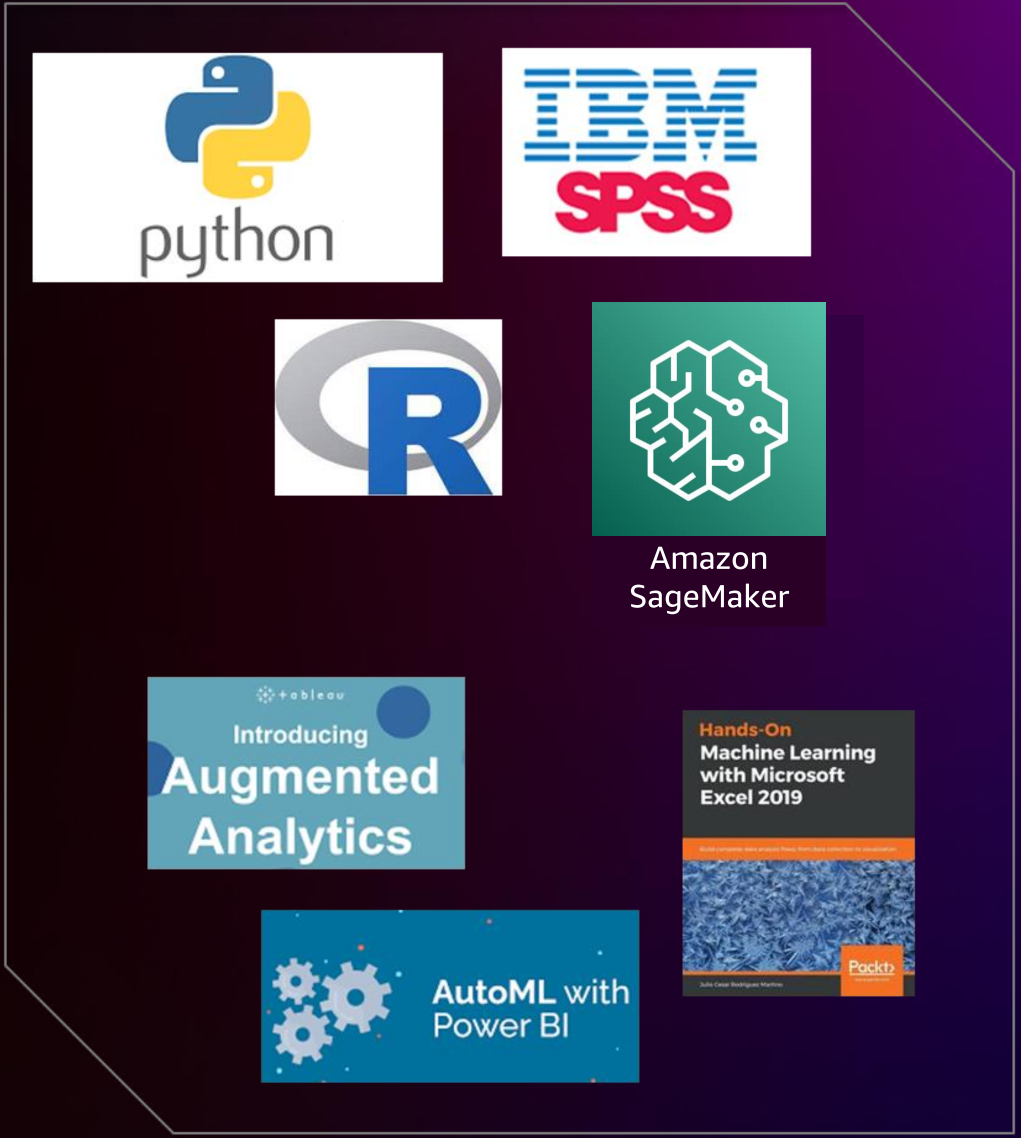
Need for an AI platform

Why?

- Increase of AI use cases
- Teams with different skills and toolsets
- Evolving regulatory requirements
- Need for reproducibility and collaboration

What?

- Access to a secure environment
- Availability of cloud resources at speed
- Seamless model governance integration
- Collaboration using a centralized model repository
- Best practices on code development



Why Amazon SageMaker?

Breadth of services

Continuous co-innovation

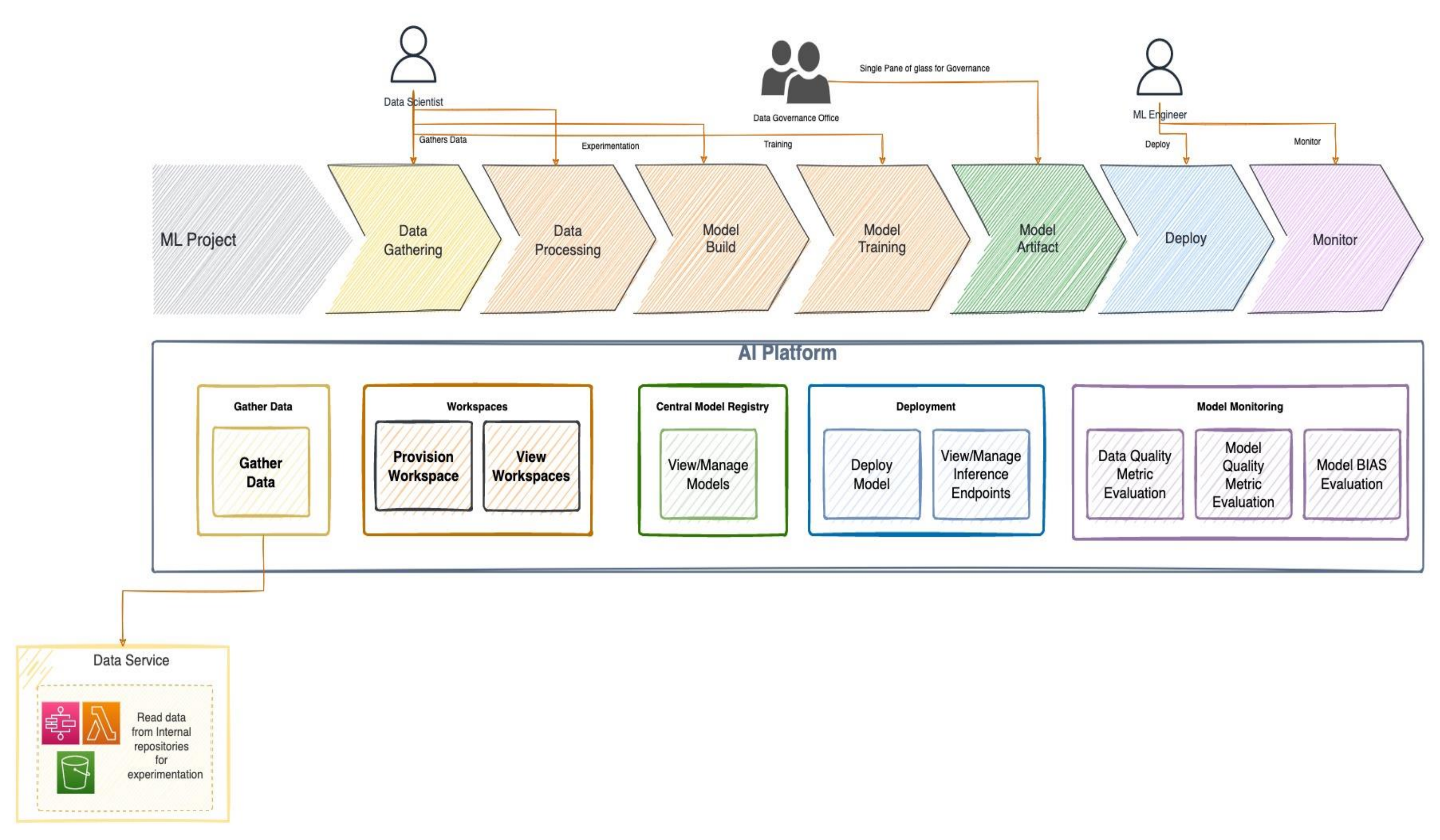
Machine learning at scale

Flexibility to go from native to custom

Great partnership

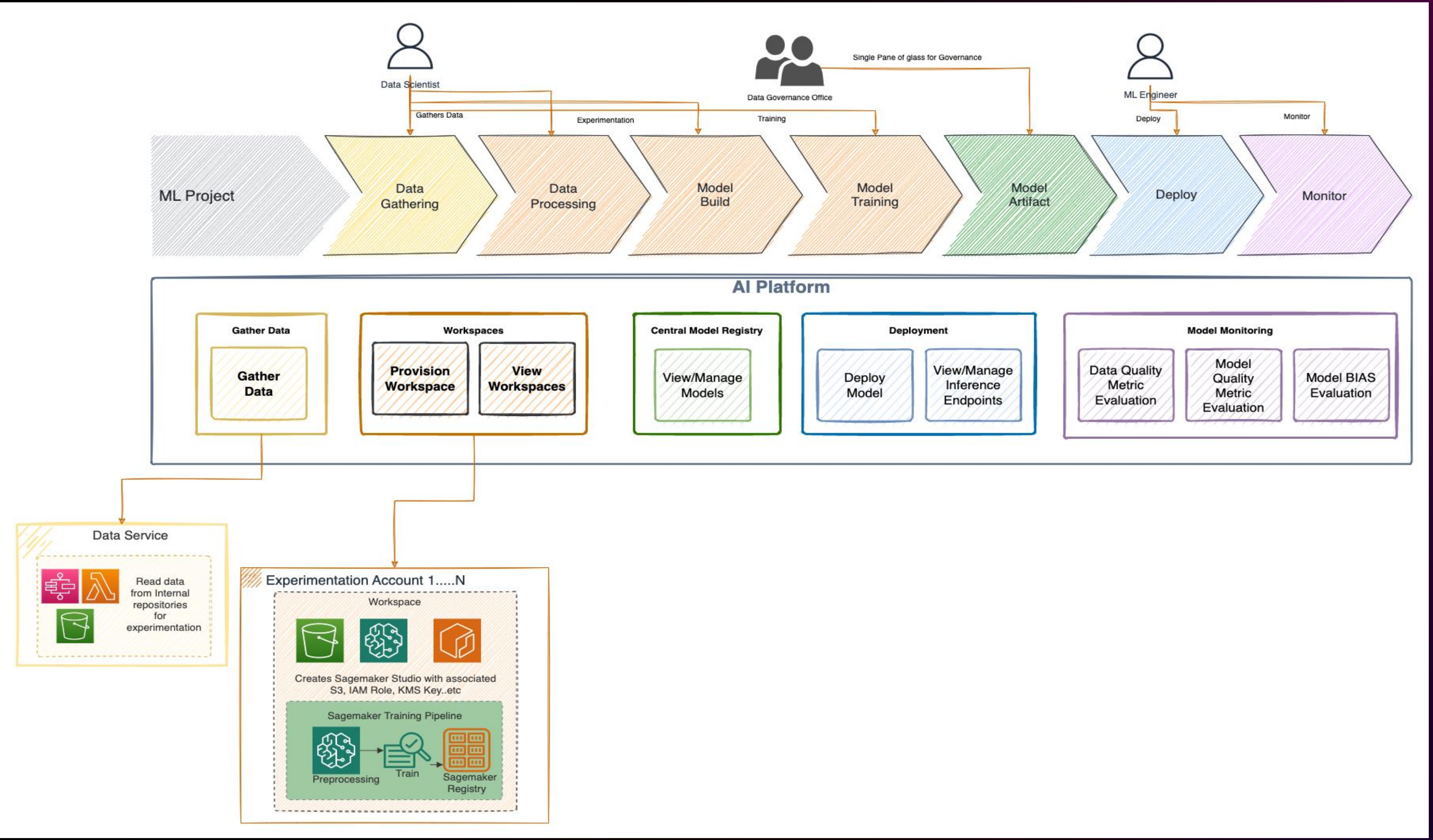
AI platform powered by AWS

AI PLATFORM STANDARDIZES THE FRAMEWORK TO ENABLE MODEL GOVERNANCE



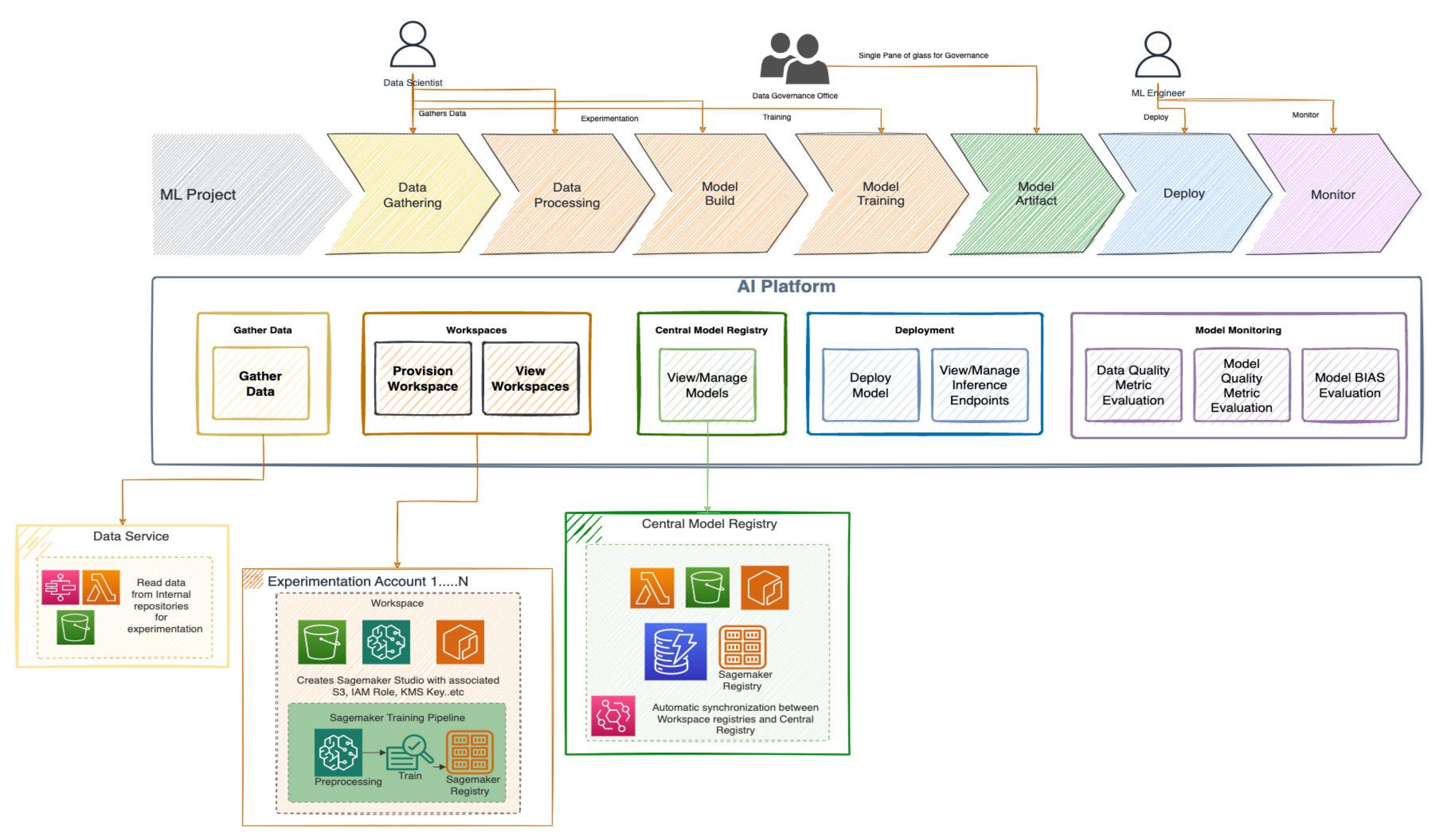
AI platform powered by AWS

AI PLATFORM STANDARDIZES THE FRAMEWORK TO ENABLE MODEL GOVERNANCE



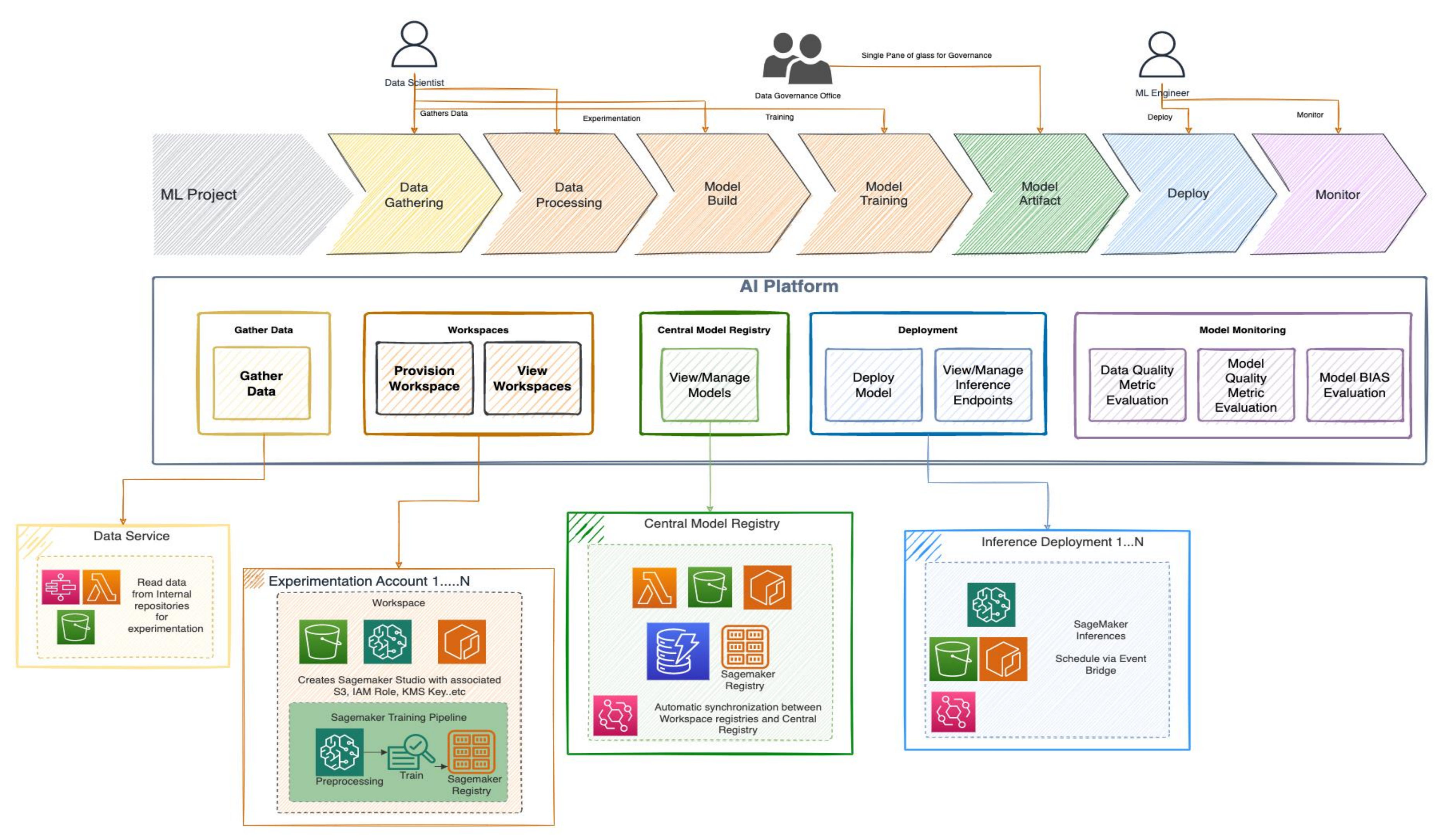
AI platform powered by AWS

AI PLATFORM STANDARDIZES THE FRAMEWORK TO ENABLE MODEL GOVERNANCE



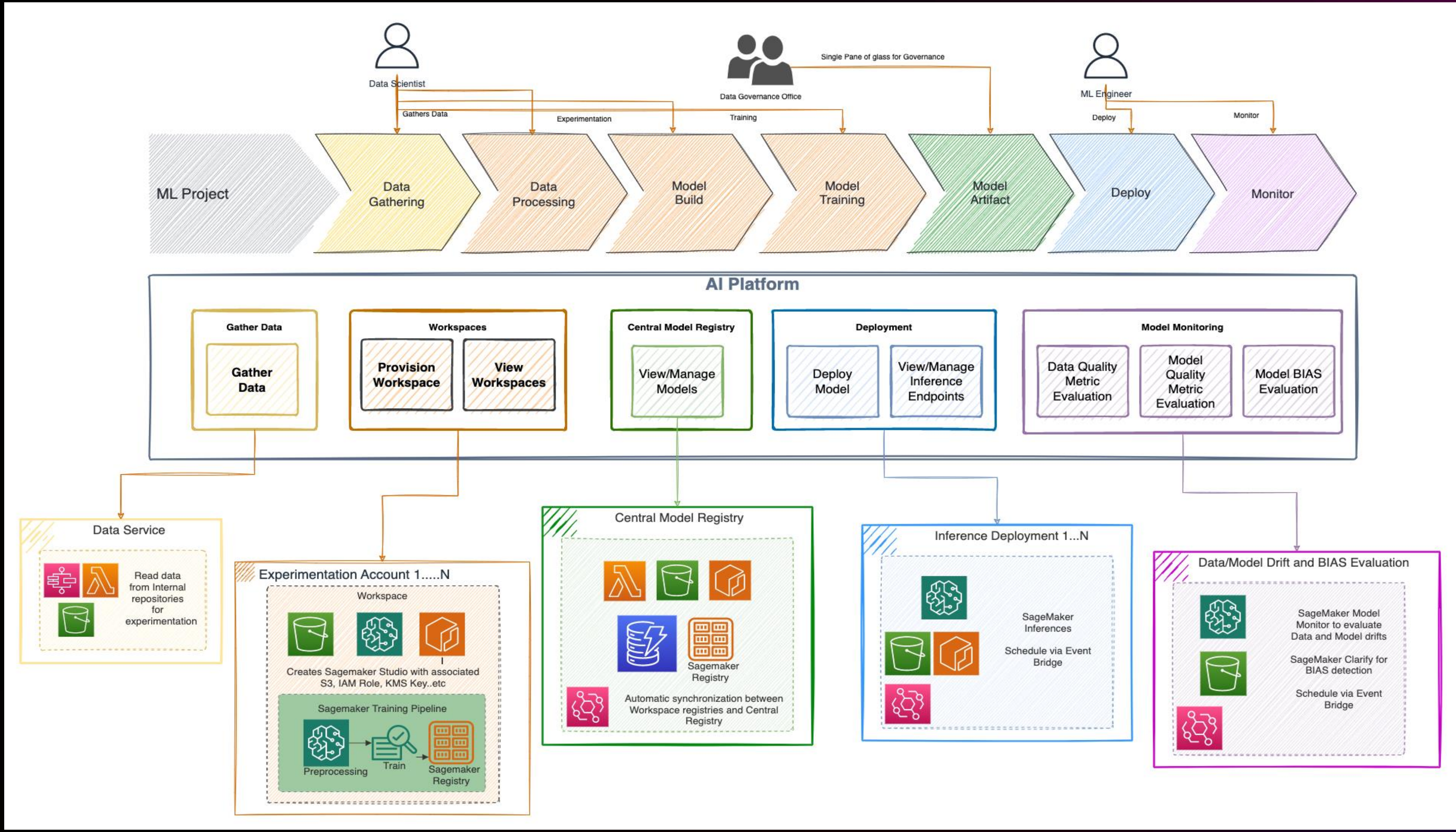
AI platform powered by AWS

AI PLATFORM STANDARDIZES THE FRAMEWORK TO ENABLE MODEL GOVERNANCE



AI platform powered by AWS

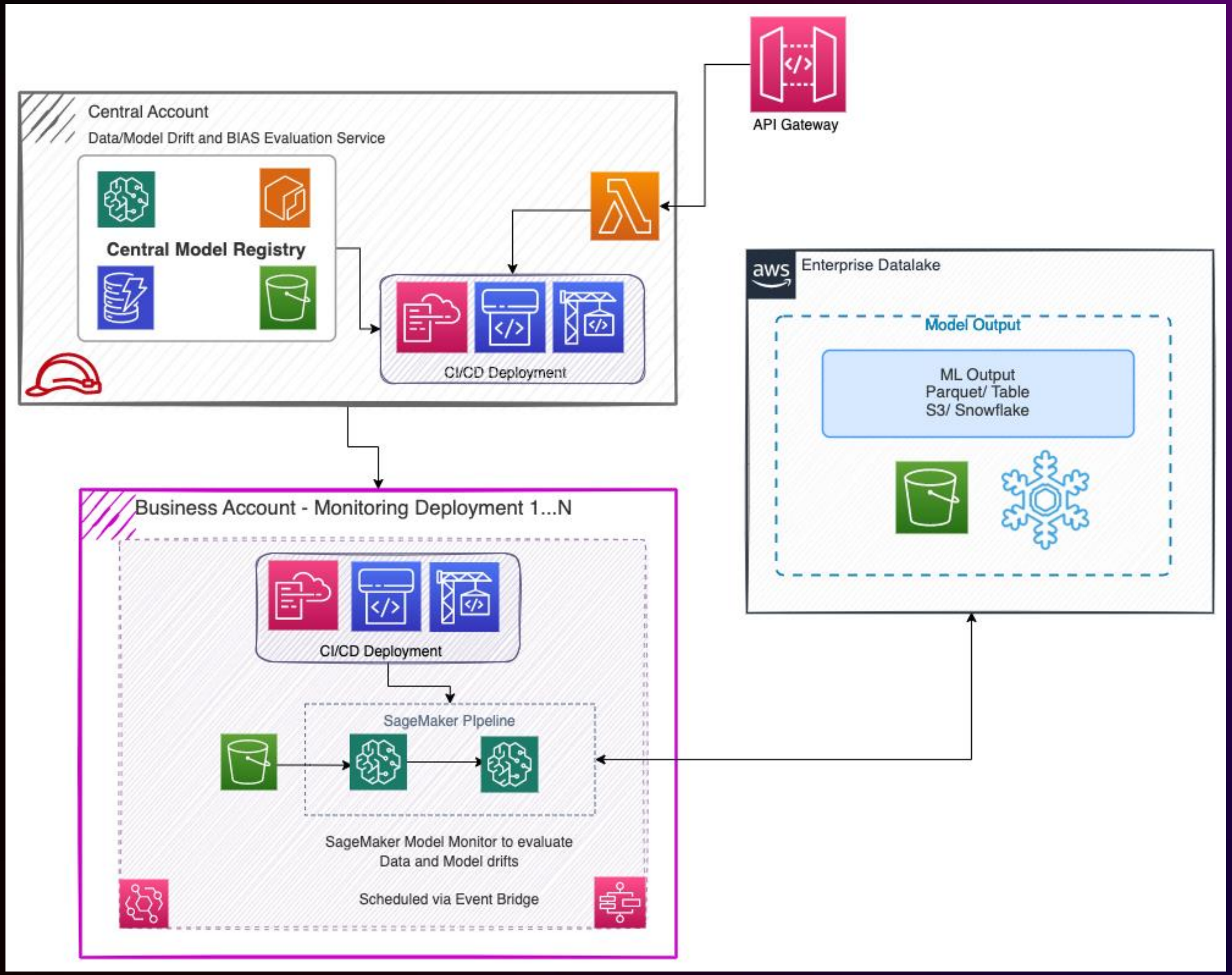
AI PLATFORM STANDARDIZES THE FRAMEWORK TO ENABLE MODEL GOVERNANCE



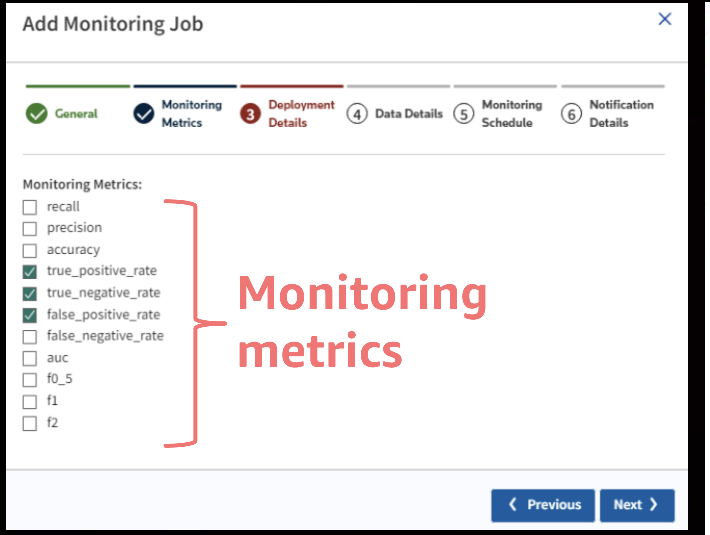
Deep dive on model monitoring

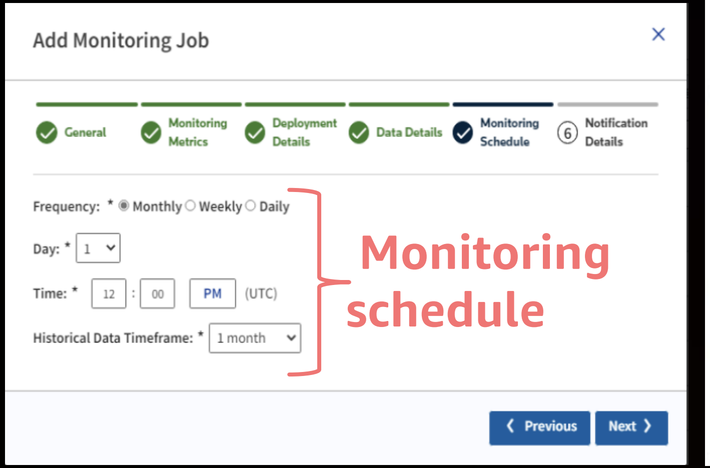
AMAZON SAGEMAKER MODEL MONITOR

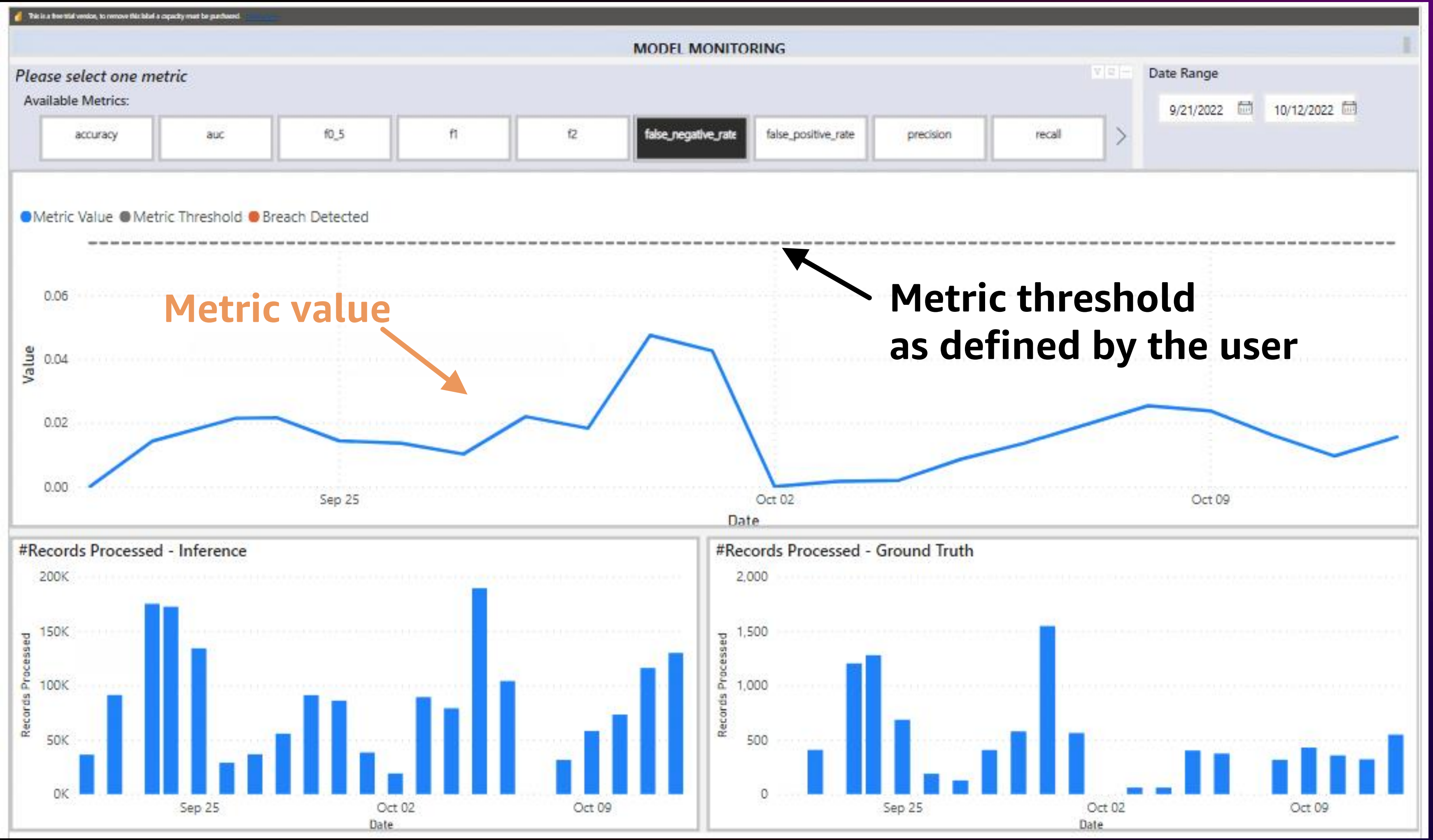
- Models and data drift as a service with the click of a button
- Baseline and continuous evaluation metrics in a central repository
- Custom metrics for model evaluation standardization
- Parametrized services for flexibility
- Deployment into individual business accounts by a central account for security and isolation



Deep dive on model monitoring







aws

What's next?

Model bias and explainability services using SageMaker Clarify

Migrate legacy models into SageMaker environment

Onboard users across business units

Amazon SageMaker

COMMON USE CASES

Predictive maintenance
Manufacturing, automotive, IoT

Demand forecasting
Retail, consumer goods, manufacturing

Fraud detection
Financial services, online retail

Credit risk prediction
Financial services, retail

Extract and analyze data from documents
Healthcare, legal, media/entertainment, education

Computer vision
Healthcare, pharma, manufacturing

Autonomous driving
Automotive, transportation

Personalized recommendations
Media/entertainment, retail, education

Churn prediction
Retail, education, software and internet

Amazon SageMaker

RE:INVENT 2022 SESSIONS

Business analysts

No-code ML

AIM207 Make better business decisions with ML using Amazon SageMaker Canvas (Breakout)

AIM337 Build better ML models for business decisions using Amazon SageMaker Canvas (Workshop)

Data scientists

Single IDE for full-code ML

AIM322 Accelerate data preparation with Amazon SageMaker Data Wrangler (Breakout)

AIM313 Scalable data preparation & ML using Apache Spark on Amazon SageMaker Studio (Workshop)

MLOps engineers

Workflow automation and CI/CD

AIM321 Productionize ML workloads using Amazon SageMaker MLOps (Breakout)

AIM308 Implementing MLOps to deliver high-performing ML models faster (Workshop)

Thank you!

Please complete the session survey in the **mobile app**