databricks

Clemens Mewald,
Director of Product Management,
Databricks

Joel Thomas,
Senior Solution Architect,
Databricks

WEBINAR

Productionizing Machine Learning: From Deployment to Drift Detection

Logistics

- We can't hear you…
- Recording will be available…
- Slides will be available…
- Code samples and notebooks will be available…
- Submit your questions…
- Bookmark **databricks.com/blog**

databricks

databricks

VISION

Accelerate innovation by unifying data science, engineering and business

SOLUTION

Unified Analytics Platform

WHO WE ARE

- Original creators of Apache Spark, Delta Lake, mlflow
- 2,000+ global companies use our platform across big data & machine learning lifecycle

databricks

UNIFIED ANALYTICS PLATFORM

Databricks Workspace

Collaborative Notebooks, Production Jobs

Databricks Runtime

DELTA LAKE

APACHE Spark

TensorFlow

PyTorch

mlflow

Databricks Cloud Service

Azure

aws

About Today's Presenters

Clemens Mewald, Director of Product Management at Databricks

- MSc in Computer Science from UAS Wiener Neustadt, Austria
- MBA from MIT Sloan
- Previously Product Lead on the Google Brain Team for TensorFlow, TFX
- Product Management Lead for Data Science and ML at Databricks

Joel Thomas, Senior Solutions Architect at Databricks

- MS in Electrical Engineering from University of Houston
- Previously Principal Data Scientist at Micron
- Architect and implement data & AI solutions for customers

databricks

Outline

- ML Development Challenges
- How MLflow, Delta Lake, + Databricks tackles these
- Real Life End to End Data + ML Lifecycle
- Demo of Model Drift Detection on Databricks
- Q&As

ML Lifecycle and Challenges

Raw Data → ETL → Featurize → Train → Score/Serve Batch + Realtime → Monitor Alert, Debug

Production Logs

Update Features

Retrain

kafka S3 Delta Microsoft Azure Blob Storage hadoop mongoDB Spark pandas SQL learn PYTORCH learn TensorFlow R Spark dmlc XGBoost Azure Machine Learning Amazon SageMaker docker Spark

Zoo of Ecosystem Frameworks

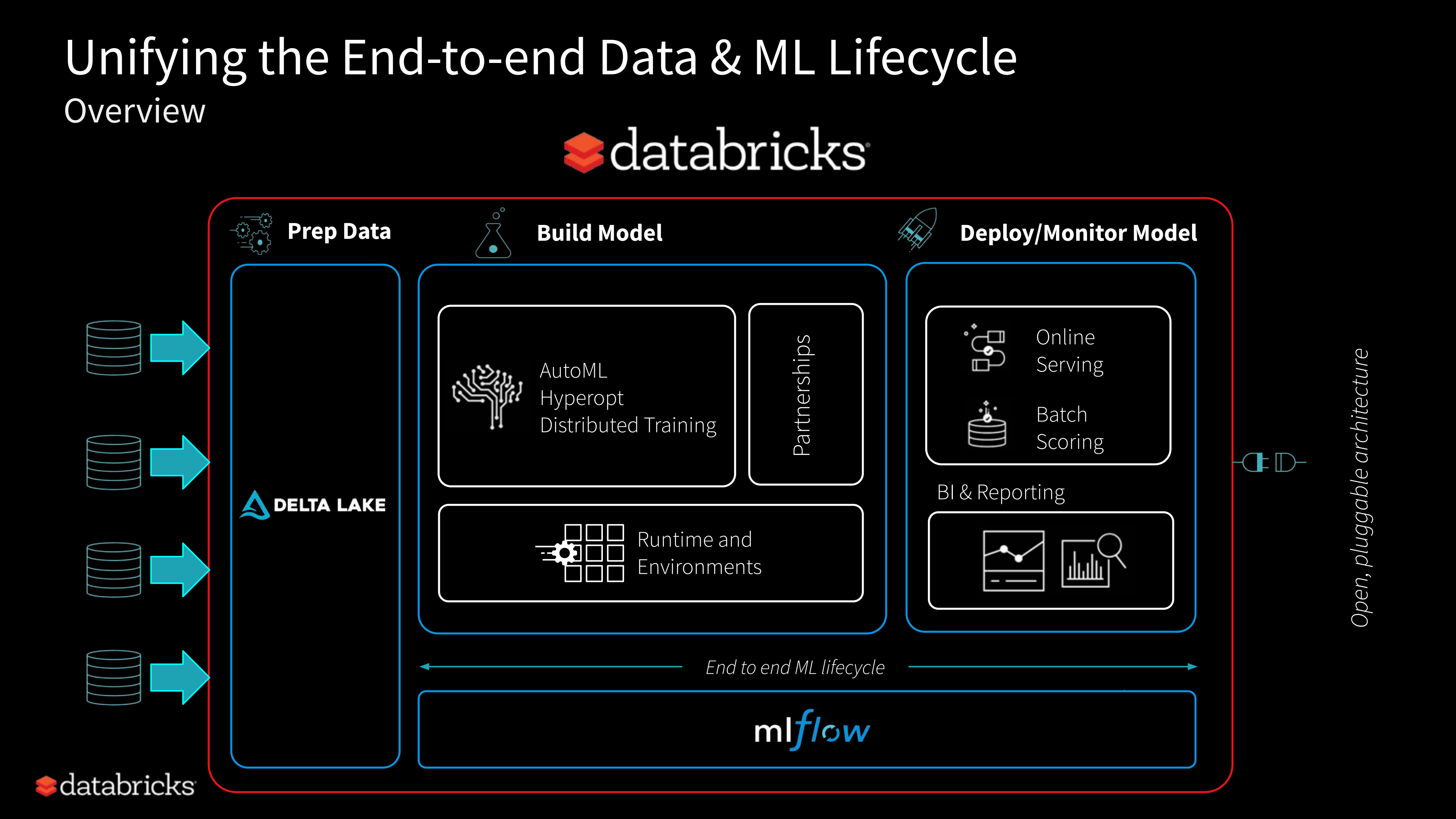
Tuning

Deploy

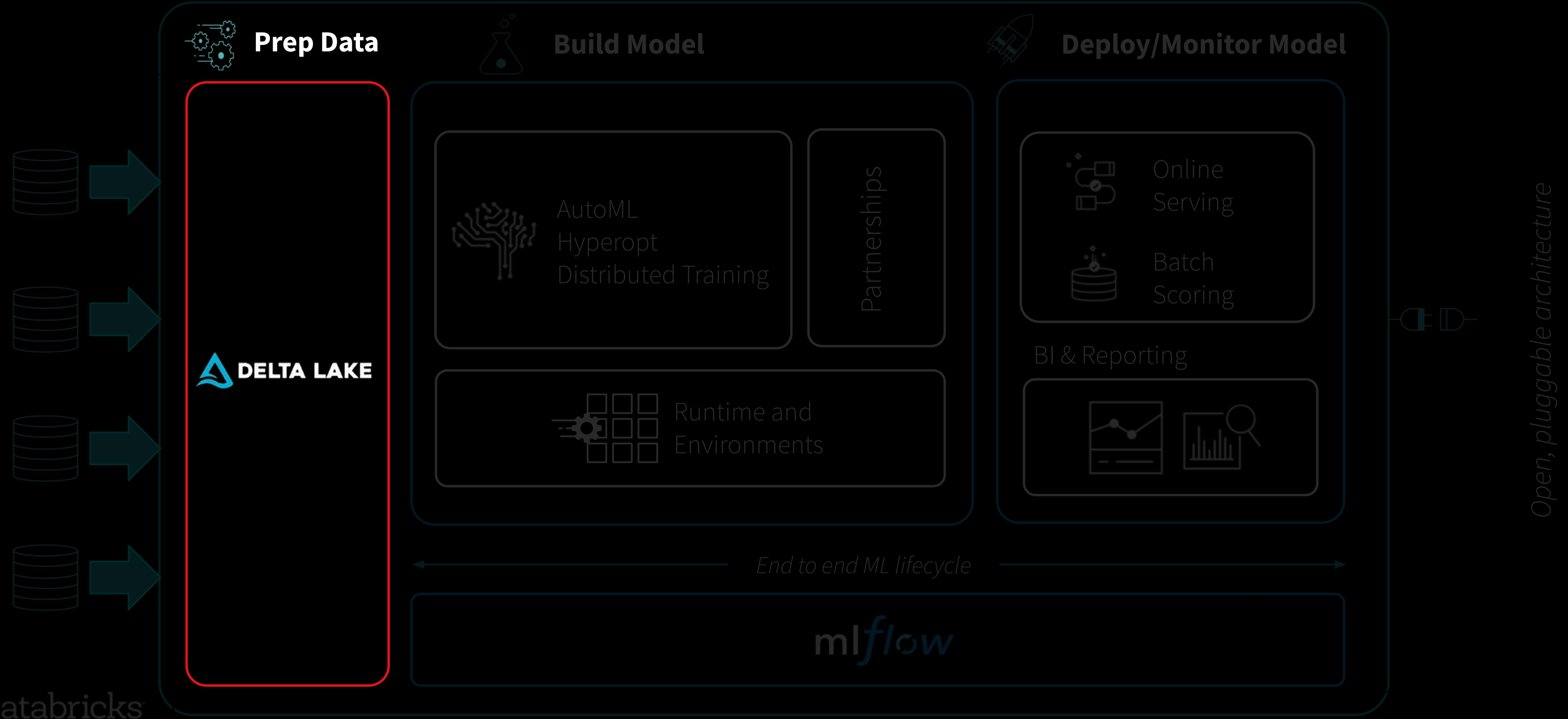
Model Mgmt

Collaboration Scale Governance

Feature Repository	Experiment Tracking	AutoML, Hyper-p. search	Remote Cloud Execution	Project Mgmt (scale teams)	Model Exchange	A/B Testing	CI/CD/Jenkins push to prod	Orchestration (Airflow, Jobs)	Lifecycle mgmt.	Data Drift	Model Drift

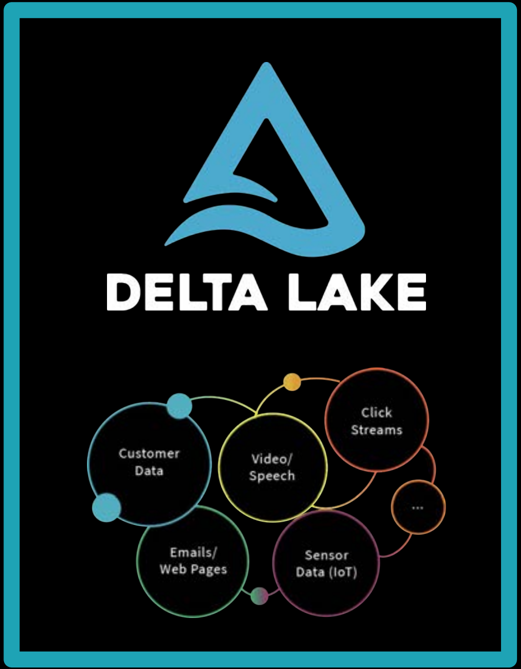

Unifying the End-to-end Data & ML Lifecycle
Overview
databricks
Prep Data
Build Model
Deploy/Monitor Model
DELTA LAKE
AutoML
Hyperopt
Distributed Training
Partnerships
Runtime and
Environments
Online
Serving
Batch
Scoring
BI & Reporting
End to end ML lifecycle
mlflow
Open, pluggable architecture
databricks

Unifying the End-to-end Data & ML Lifecycle
Overview



Reliability & Performance With Open Standards

databricks | DELTA LAKE



Reliability - High Quality Data

- Transactions and schema enforcement
- Data quality validation and expectation

Performance - Fast Queries at Scale

- Optimizations - data skipping, caching & indexing
- Compaction optimizes file layout

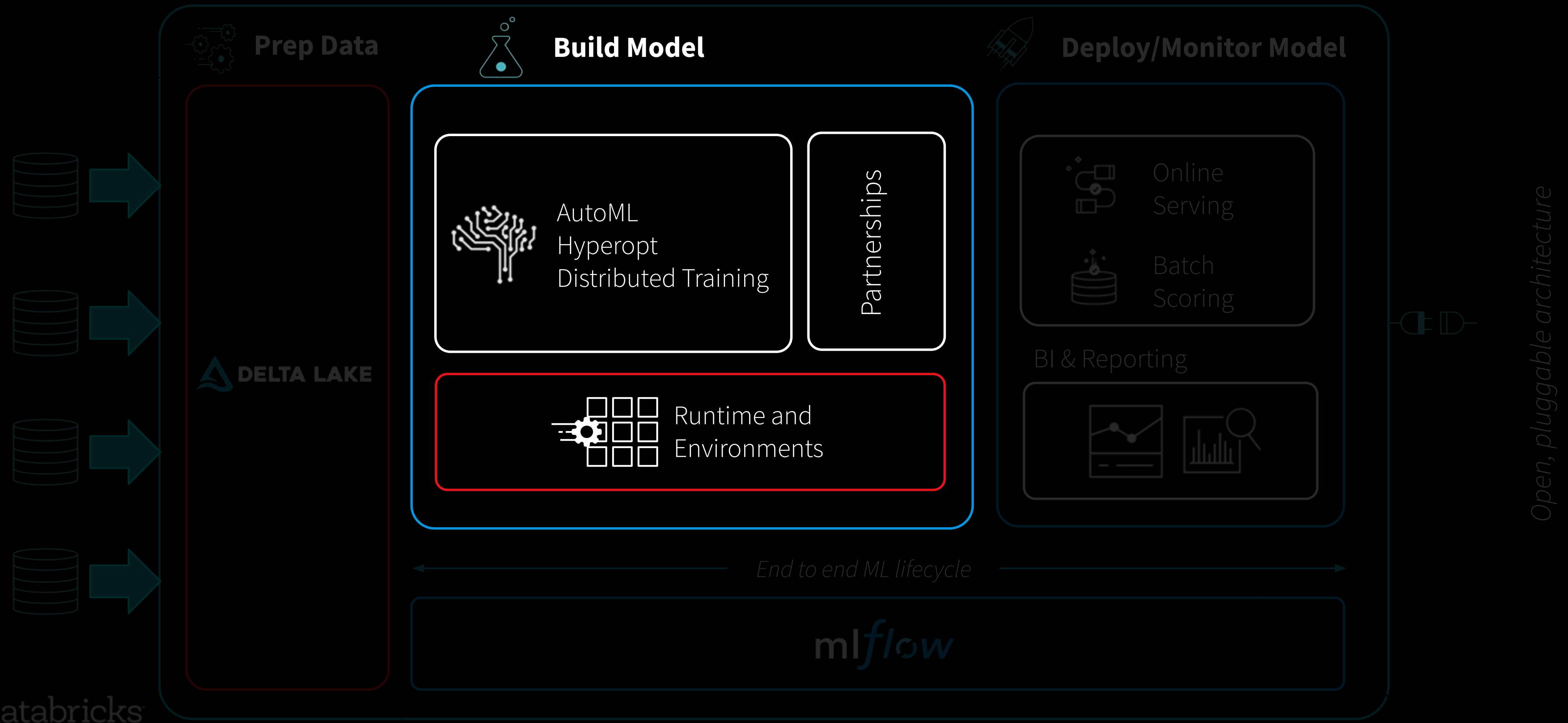
Open Standard - Easier Adoption

- Open source, open format; compatible with Spark API's
- Data stays in your data lake under your control

databricks.com/delta   delta.io   github.com/delta-io

Unifying the End-to-end Data & ML Lifecycle

Overview



databricks

ML environment and Model Training

databricks | Runtime for ML

Packages and optimizes most common ML Frameworks

XGBoost K learn …

Built-in Optimization for Distributed Deep Learning

HOROVOD

Distribute and Scale any Single-Machine ML Code to 1,000’s of machines.

Built-In AutoML and Experiment Tracking

mlflow™

AutoML and Tracking / Visualizations with MLflow

Customized Environments using Conda

requirements.txt

conda.yaml

Customization

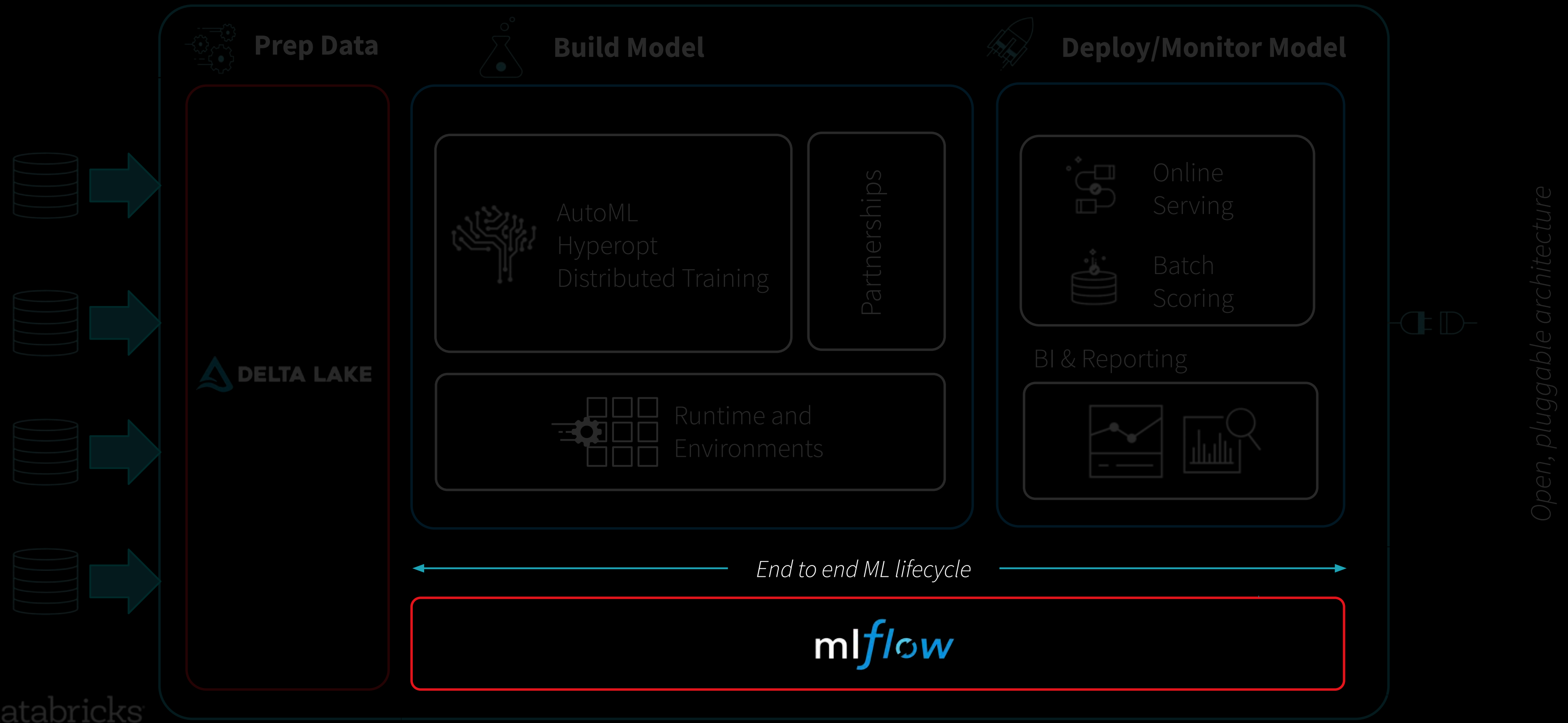
Pre-configured Environment

Machine Learning

Conda-Based

Unifying the End-to-end Data & ML Lifecycle

Overview



An open source framework for the complete ML Lifecycle

databricks | mlflow

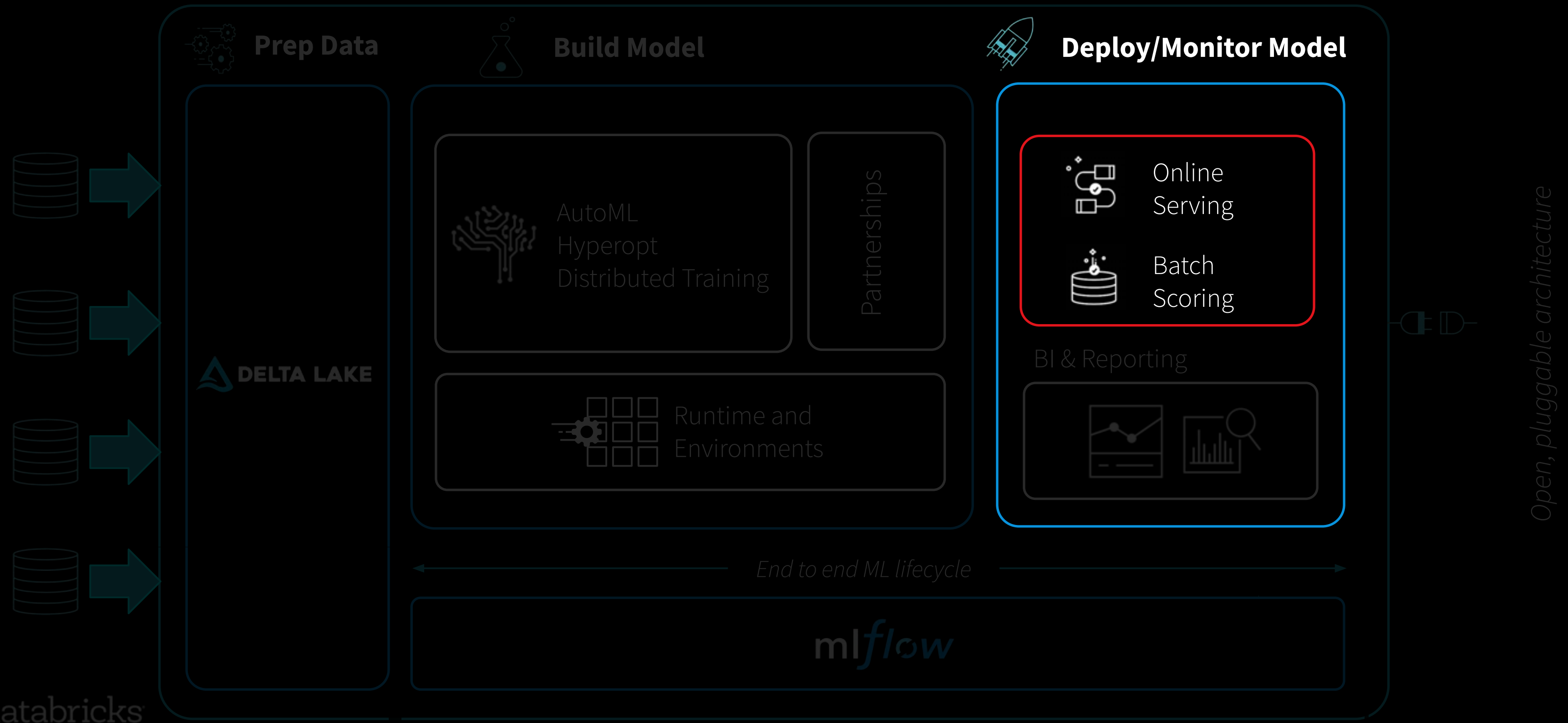
Simplified experiment tracking, reproducibility, and deployment, from experimentation to production

mlflow Tracking	mlflow Projects	mlflow Models
Record and query experiments: code, data, config, results	Packaging format for reproducible runs on any platform	General model format that supports diverse deployment tools

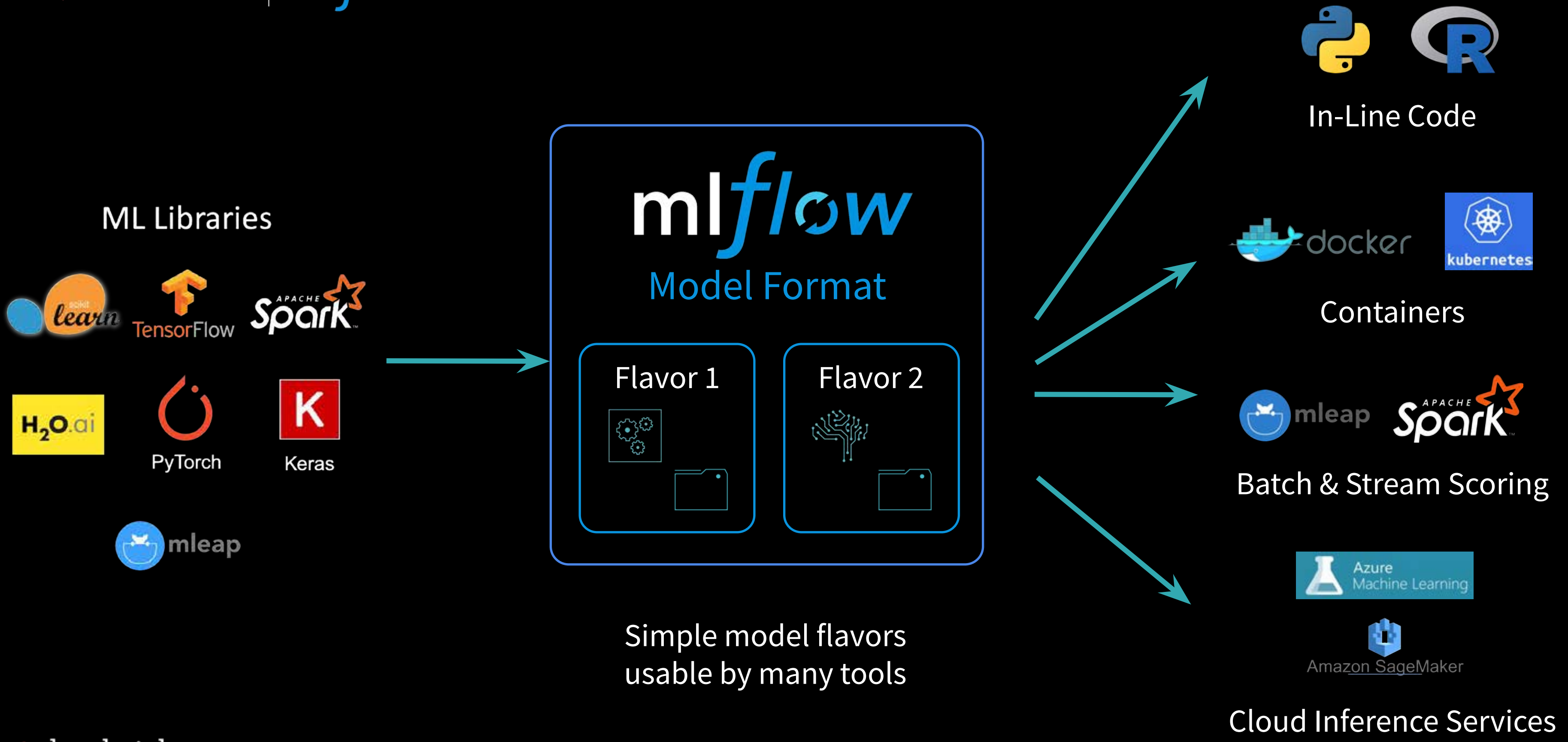
databricks.com/mlflow mlflow.org github.com/mlflow twitter.com/MLflow

Unifying the End-to-end Data & ML Lifecycle

Overview



Model Deployment and Monitoring



databricks

What happens after Deployment?

Heraclitus (paraphrased):

"Change is the only constant in life"

Types of change to worry about in Machine Learning

Concept Drift

- Statistical properties of target variable change (i.e. what you are trying to predict changes)
- E.g. fraud detection

Types of change to worry about in Machine Learning

Concept Drift

- Statistical properties of target variable change (i.e. what you are trying to predict changes)
- E.g. fraud detection

Data Drift

- Statistical properties of input variables change
- E.g. seasonality, personal preferences, trends change

Types of change to worry about in Machine Learning

Concept Drift

- Statistical properties of target variable change (i.e. what you are trying to predict changes)
- E.g. fraud detection

Data Drift

- Statistical properties of input variables change
- E.g. seasonality, personal preferences, trends change

Upstream Data Changes

- Encoding of a feature changes (e.g. switch from Fahrenheit to Celsius)
- Features are no longer being generated (leads to missing values)

Ways to detect and protect against changes

Monitor

Training Data

- Schema & distribution of incoming data
- Distribution of labels

Requests & Predictions

- Schema & distribution of requests
- Distribution of predictions
- Quality of predictions

Ways to detect and protect against changes

Monitor

Training Data

- Schema & distribution of incoming data
- Distribution of labels

Requests & Predictions

- Schema & distribution of requests
- Distribution of predictions
- Quality of predictions

Intervene

Correct Data Problems

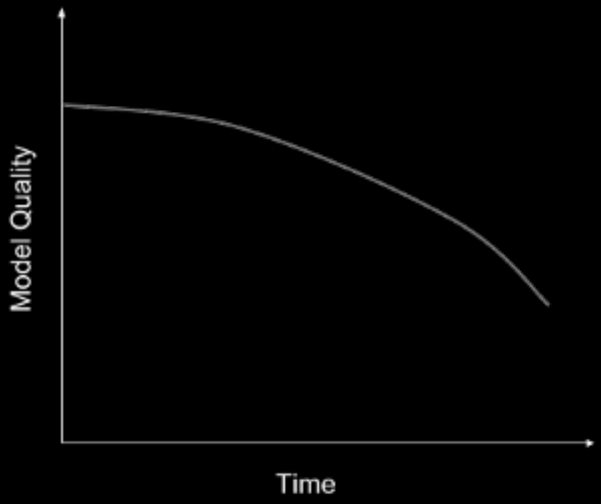
- Detect and correct upstream data problems early
- Clean up erroneous labels

Retrain / Calibrate Model

- In cases where there is concept or data drift, keep models fresh by retraining them

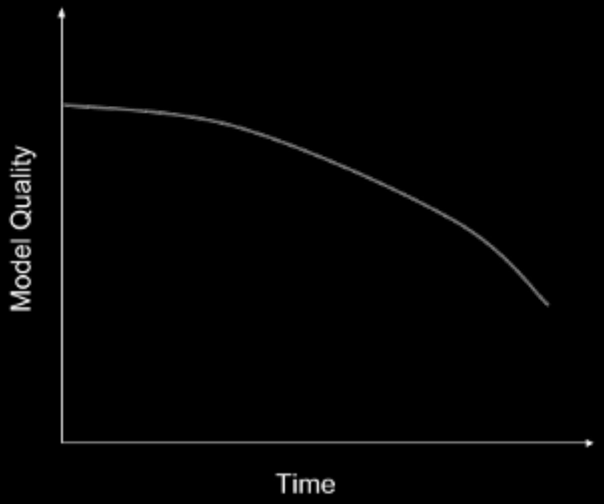
Your Machine Learning Models become stale

Model Staleness over time

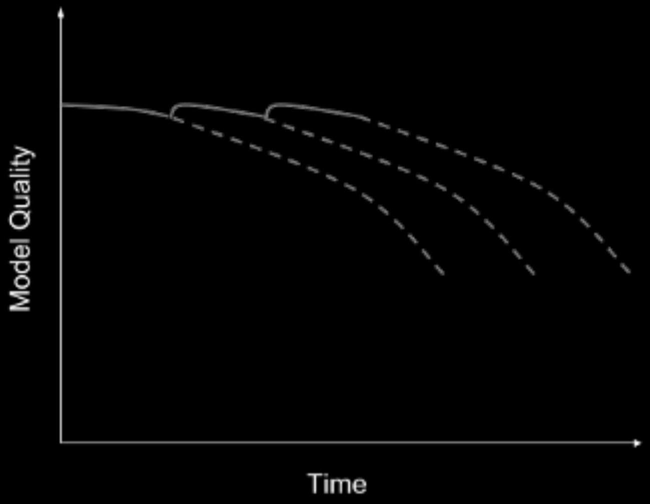


Your Machine Learning Models become stale

Model Staleness over time



Refreshing models over time



Real-Life Examples of Drift

IoT

- Sensor readings can change over time (that's why sensors need to be recalibrated)
- Sensor readings may change with seasonality

Retail

- Inventory changes over time
- Customer preferences change over time
- Regional changes and trends

Media & Entertainment

- Content changes every second
- Today's patterns may not apply tomorrow

From Data Prep to Production: Real Life IoT Example

Process and Prepare Data at Scale

Bronze

Raw data

- Lots and lots of small files on BLOB storage

Streaming

- Data enriched with originating filepath, derived data type and year, month, date (for partitioning)
- Checkpointed for fault tolerance & file listings
- ~150k records per second

Silver

Single Delta Table

- Consistent schema for ML/Analytics
- 70 Billion rows of data
- Query performance improvement of 5.08 hours to 30.13 minutes following an OPTIMIZE

Gold

- Time series resampled & interpolated
- Feature transpose
- Feature reduction

DELTA LAKE™

From Data Prep to Production: Real Life IoT Example

Train, Deploy, and Monitor at Scale

Training data
160k sensors / 60 billion rows

Gold feature tables

DELTA LAKE™

Trained models
100k

Train an sklearn model per sensor via a Pandas UDF

Model Training

Predictions
9 million in 1 hour on a 4 node cluster

Apply an sklearn model per sensor via a Pandas UDF

Model Inference

Compare predicted value to rolling average via a Pandas UDF

Model Monitoring

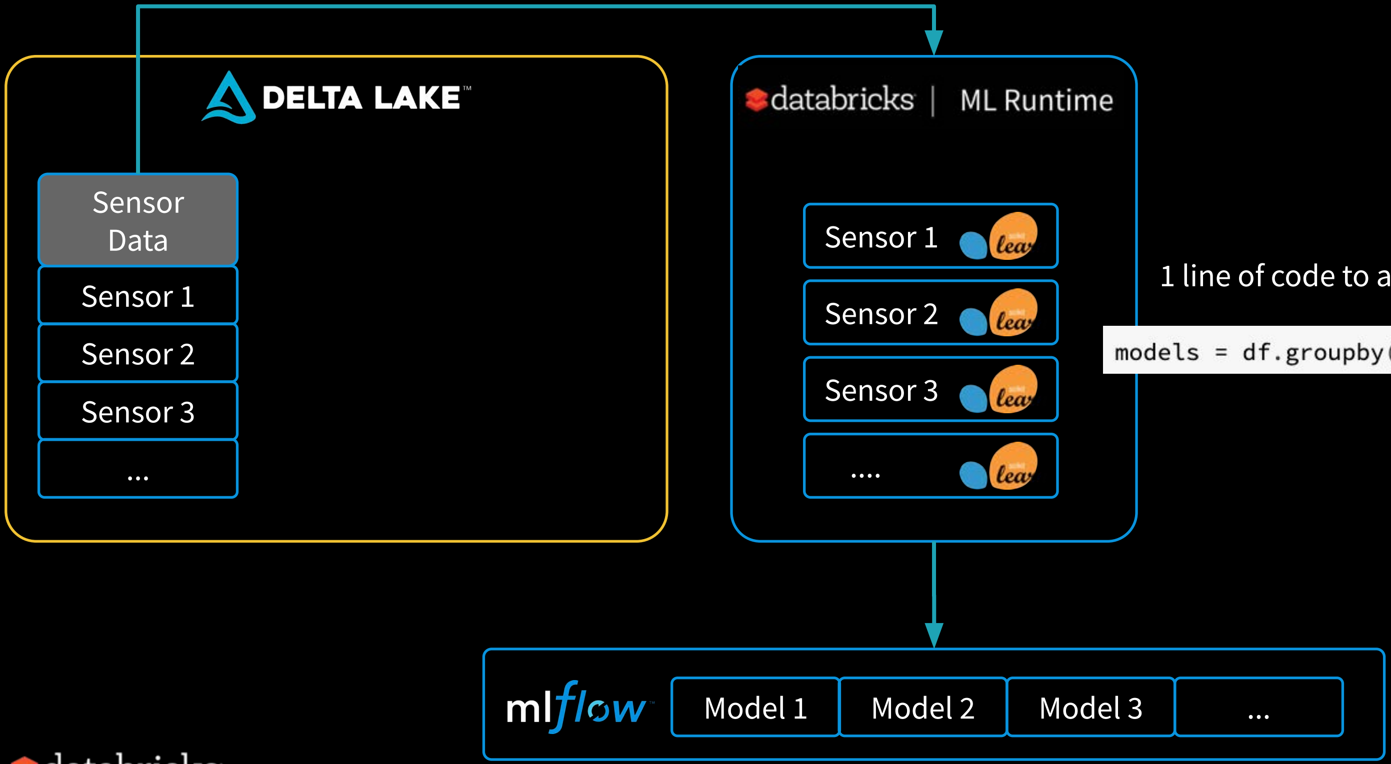
Log params, metrics & model to MLflow

Select & load model from MLflow

mlflow™

From Data Prep to Production: Real Life IoT Example

1) Training at Scale



1 line of code to apply UDF to *n* sensors

```
models = df.groupby("SensorId").apply(train_sklearn)
```

From Data Prep to Production: Real Life IoT Example

2) Deploying at Scale

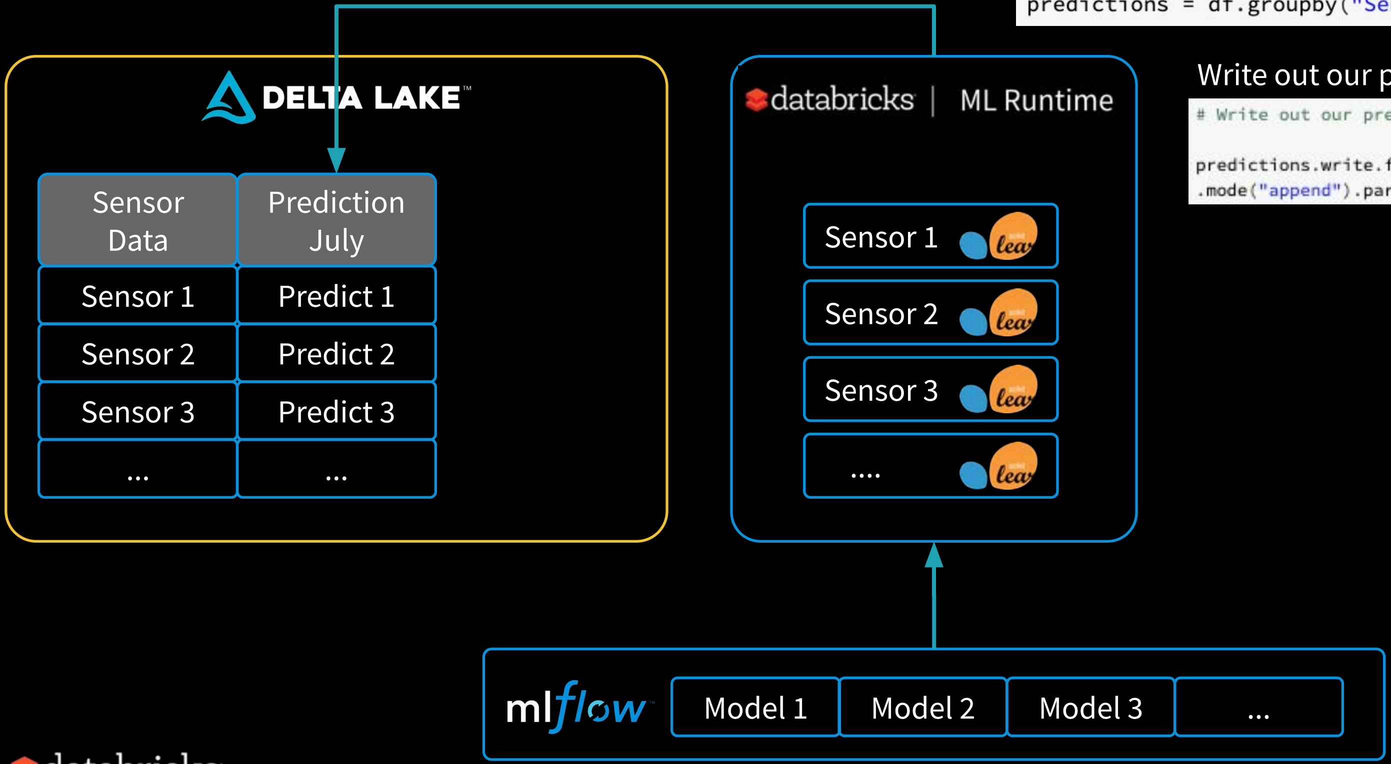
1 line of code to apply UDF to *n* new sensor readings

```
predictions = df.groupby("SensorId").apply(apply_sklearn_model)
```

Write out our predictions to a Delta table

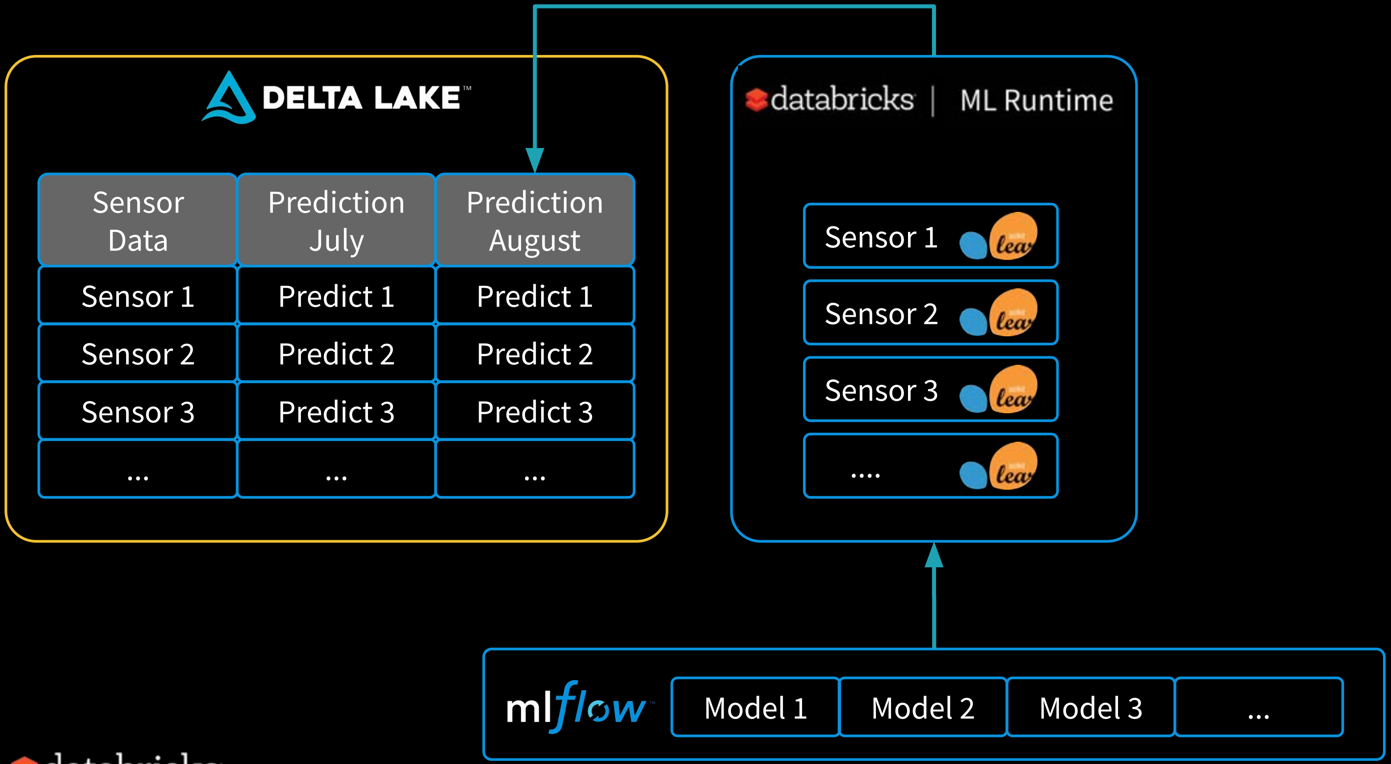
```
# Write out our predictions to a table

predictions.write.format("delta") \
.mode("append").partitionBy("SensorId").save(predictions_path)
```



From Data Prep to Production: Real Life IoT Example

2) Deploying at Scale

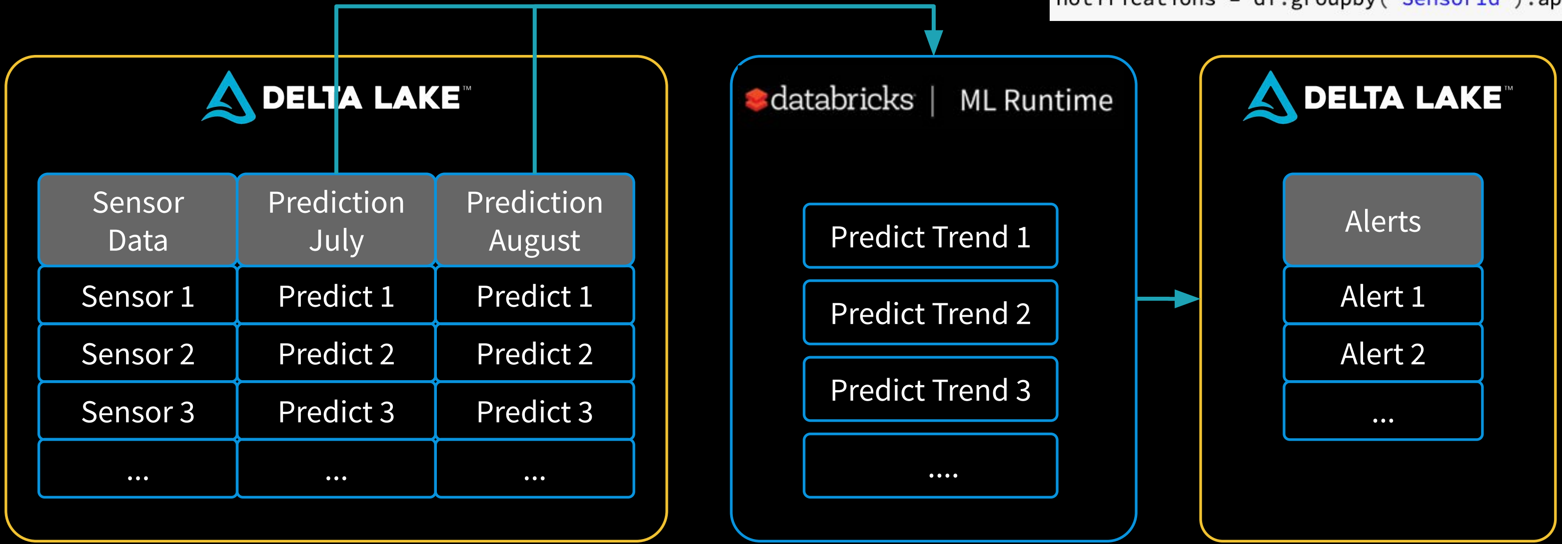


From Data Prep to Production: Real Life IoT Example

3) Monitoring at Scale

1 line of code to apply UDF to *n* predictions

```
notifications = df.groupby("SensorId").apply(monitor_predictions)
```



Write out our alerts to a Delta table

```
# Write out our notifications to a table

notifications.write.format("delta") \
.mode("append").partitionBy("SensorId").save(notifications_path)
```

Demo

Who wants a perfect cake?

Parameters like temperature and duration have an impact on quality of cake

Glassware Manufacturing Dataset

Data streaming from IoT sensors

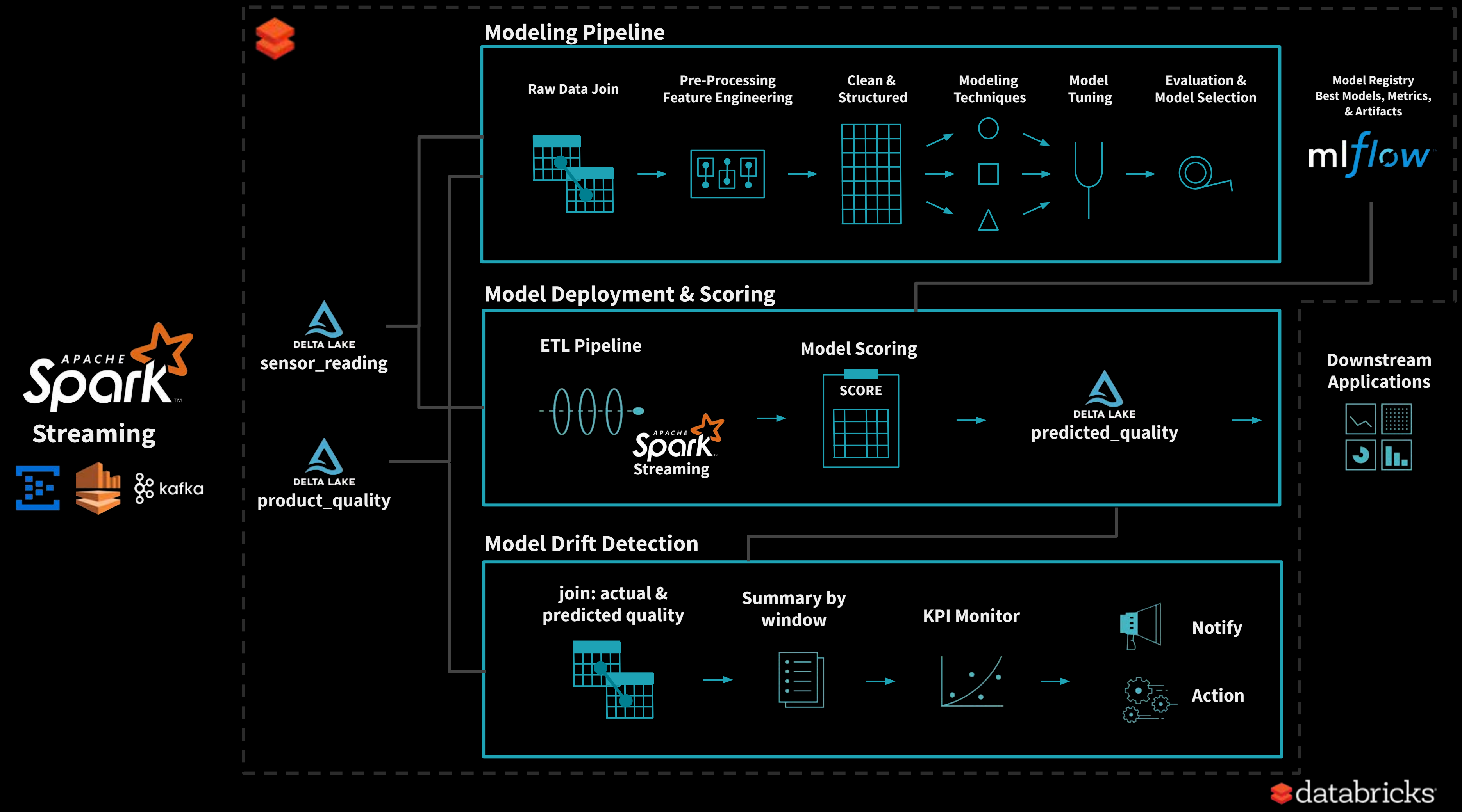
- Temperature
- Pressure
- Duration

These parameters in ideal combination gives good quality product.

Goal: Predict product quality inline to be used for additional manual inspection

Synthesized dataset to showcase model drift

databricks

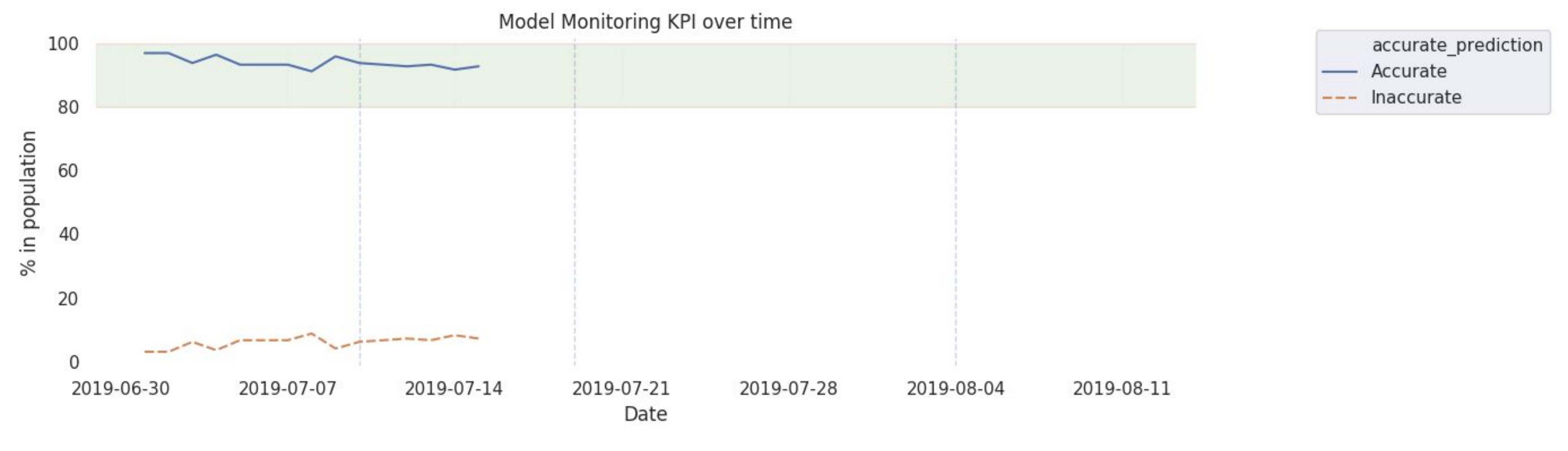


Modeling Pipeline
Raw Data Join
Pre-Processing
Feature Engineering
Clean &
Structured
Modeling
Techniques
Model
Tuning
Evaluation &
Model Selection
Model Registry
Best Models, Metrics,
& Artifacts
mlflow
APACHE Spark
Streaming
kafka
DELTA LAKE
sensor_reading
DELTA LAKE
product_quality
Model Deployment & Scoring
ETL Pipeline
APACHE Spark
Streaming
Model Scoring
SCORE
DELTA LAKE
predicted_quality
Downstream
Applications
Model Drift Detection
join: actual &
predicted quality
Summary by
window
KPI Monitor
Notify
Action
databricks

Deployment to Drift Detection - a Typical Workflow

1. To understand the data, we start with EDA (Exploratory Data Analysis)
2. Using historical data, we explore various modeling methods, tune its hyperparameters, and identify our best model
3. All the experiment runs are tracked using MLflow and we tag the best model for production use
4. While scoring in a streaming pipeline, production model is accessed from MLflow
5. Model is stable for first ‘x’ days

KPI to Monitor Model Quality

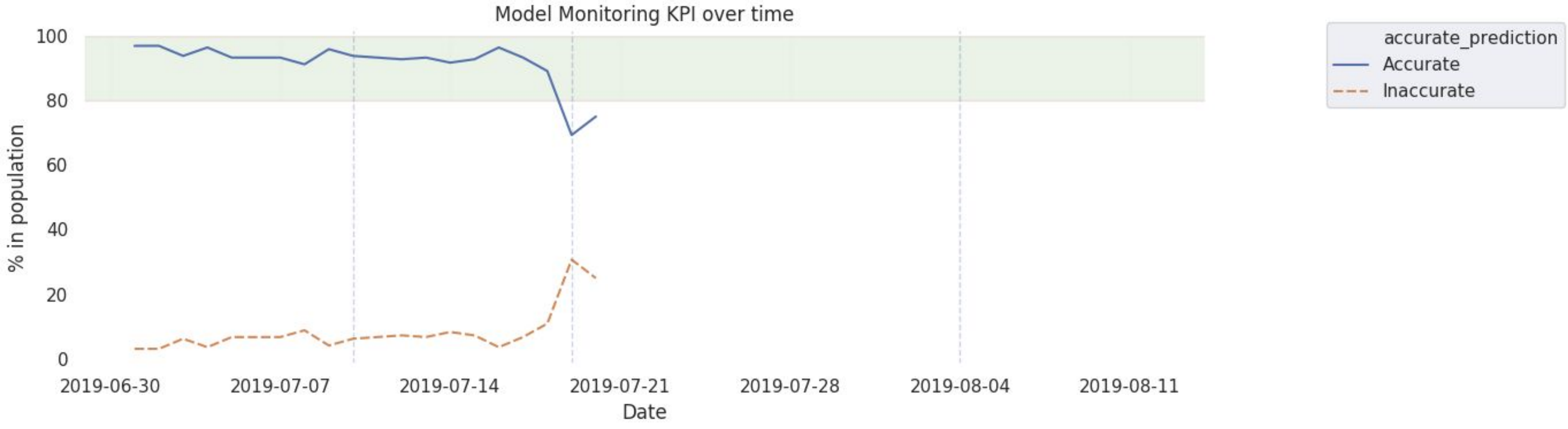


Model Drift KPIs:

- KPIs and its margin depends on the model and business problem
- Sometimes more than 1 KPI maybe needed at times to capture behavior changes

Deployment to Drift Detection - a Typical Workflow

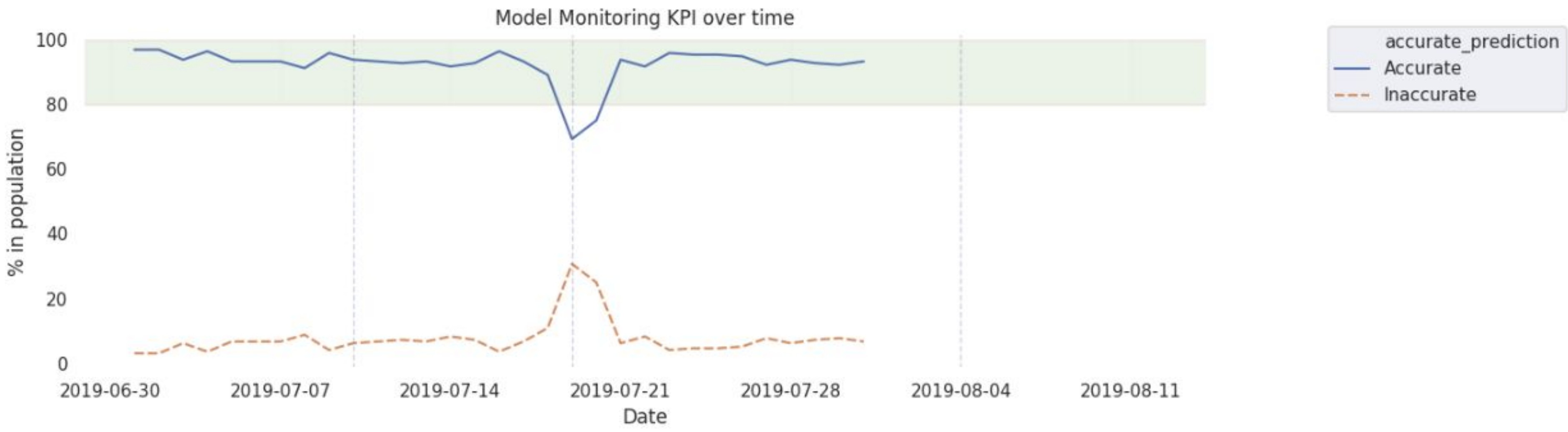
- After ‘y’ days, we see model drift occur, as identified by tracking KPIs
- This triggers re-training process



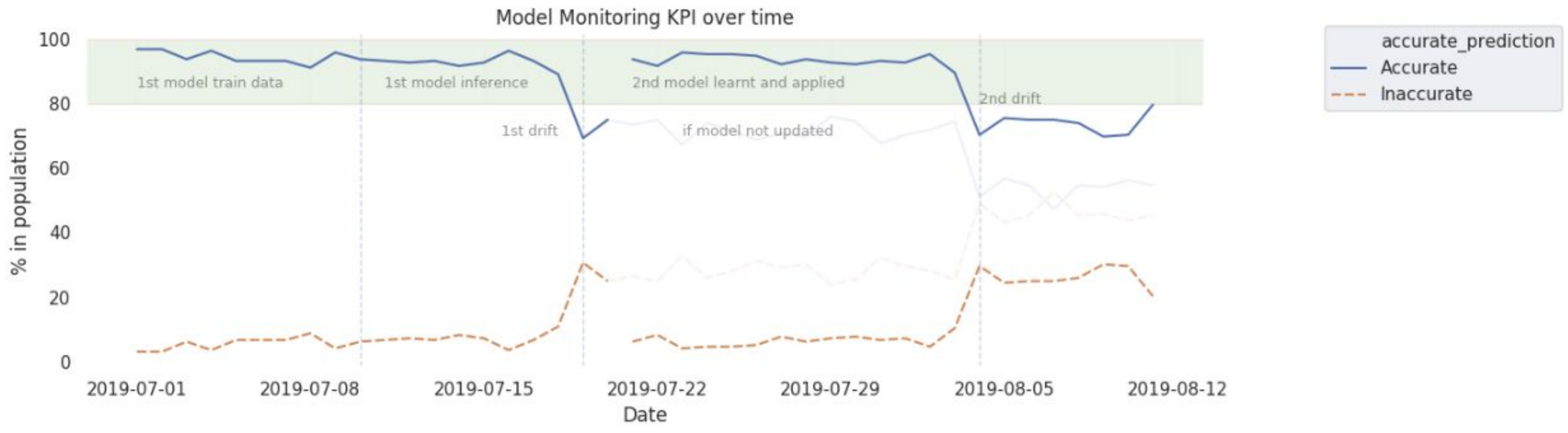
Deployment to Drift Detection - a Typical Workflow

- Once again, we explore various modeling methods, tune its hyperparameters, and identify our new best model
- The new model is tagged as current production model in MLflow
- We once again observe that KPIs are back within acceptable range
- Over time, based on business demands, it may be needed to update KPIs and its acceptable limits

Deployment to Drift Detection - a Typical Workflow



Summary - Deployment to Drift Detection



databricks

Model Drift Demo: Source Code

https://github.com/joelcthomas/modeldrift

Summary

- The only constant is change. Concept drift, data drift, and upstream data problems all affect your ML pipeline.
- Monitoring data and ML model performance is critical.
- In many cases, your ML model needs to adapt to changes.
- Databricks provides an end-to-end platform to deploy, monitor, and automatically retrain your ML models.

Coming soon

mlflow Model Registry

Tagging, sharing and versioning models in the MLflow tracking server

- Register any model in MLflow Model format
- Deploy to serving systems
- Add metadata (e.g. tags/notes) and track model creators and users

SPARK+AI SUMMIT EUROPE

15 - 17 OCTOBER 2019 | AMSTERDAM | ORGANIZED BY databricks

- 15-17 October 2019 | Amsterdam
- New product announcements around Delta Lake, MLflow, etc.
- Sign up at databricks.com/sparkaisummit

Q&A

Get Started - databricks.com/try
Contact us - databricks.com/contact

databricks