

Simplifying the
Machine Learning Lifecycle

Agenda

- / Broad Adoption of ML ... and its issues
- / The need for standardization
- / ML development challenges
- / How MLflow tackles these

Login to Databricks Community Edition

<https://databricks.com/try>

- Sign up for Databricks Community Edition for free
- We will use this for the tutorial
- Once you sign up, you can continue to use it to learn and experiment on a dedicated data sciences engineering environment

Go to databricks.com/try

The screenshot shows the Databricks website's "Try Databricks" landing page. At the top, there is a navigation bar with links for Platform, Solutions, Customers, Learn, Partners, Events, Open Source, and Company. To the right of the navigation are search, language (EN), support, contact, log in, and a prominent orange "TRY DATABRICKS" button. Below the navigation, the page title "Try Databricks" is displayed in a large, bold font. A sub-headline describes it as "An open and unified data analytics platform for data engineering, machine learning, and analytics" and credits them as "From the original creators of Apache Spark™, Delta Lake, MLflow, and Koalas". A form is present for users to provide their information: "First Name" and "Last Name" in input fields, "Company Name" and "Work Email" in input fields, "How would you describe your role?" and "What is your intended use case?" in dropdown menus labeled "Select...", and a "Phone Number" input field. A note below the form states, "By Clicking "Sign Up", you agree to the [Privacy Policy](#)". There is also a checked checkbox for "Keep me informed with occasional updates about Databricks and Apache Spark™". A teal "SIGN UP" button is located at the bottom left of the form. In the bottom right corner, there is a "Help" button with a question mark icon.

Sign up for Community Edition

 Platform Solutions Customers Learn Partners Events Open Source Company

TRY DATABRICKS

Try Databricks

An open and unified data analytics platform for data engineering, machine learning, and analytics

From the original creators of Apache Spark™, Delta Lake, MLflow, and Koalas

Select a platform

DATABRICKS PLATFORM - FREE TRIAL
For businesses

- Collaborative environment for Data teams to build solutions together
- Unlimited clusters that can scale to any size, processing data in your own account
- Job scheduler to execute jobs for production pipelines
- Fully collaborative notebooks with multi-language support, dashboards, REST APIs
- Native integration with the most popular ML frameworks (scikit-learn, TensorFlow, Keras,...), Apache SparkTM, Delta Lake, and MLflow
- Advanced security, role-based access controls, and audit logs
- Single Sign On support
- Integration with BI tools such as Tableau, Qlik, and Looker
- 14-day full feature trial (excludes cloud charges)

GET STARTED ON

 OR 

Please note that Azure Databricks is provided by Microsoft and is subject to Microsoft's terms.

By clicking on the "AWS" button to get started, you agree to the [Databricks Terms of Service](#).

TRY DATABRICKS

COMMUNITY EDITION
For students and educational institutions

- Single cluster limited to 6GB and no worker nodes
- Basic notebooks without collaboration
- Limited to 3 max users
- Public environment to share your work

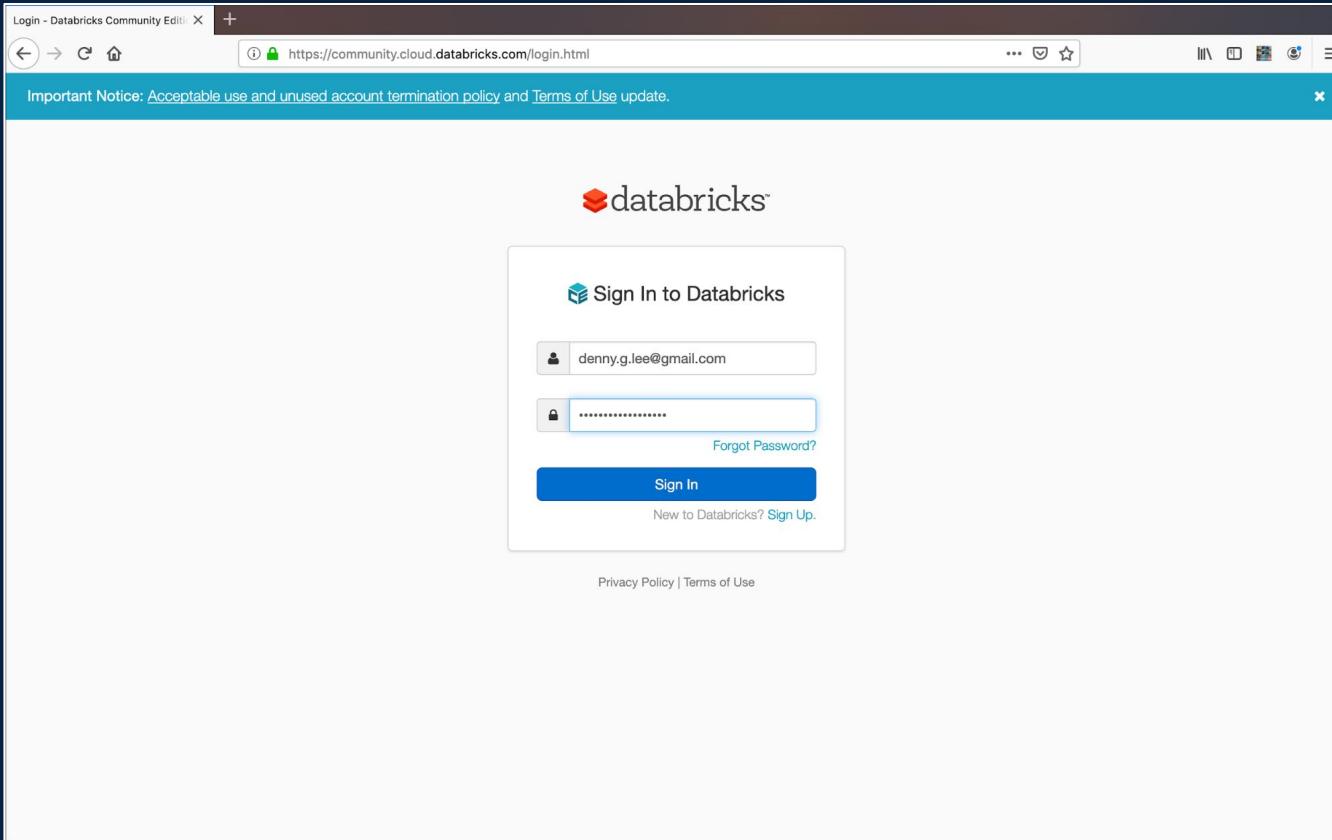
GET STARTED

By clicking "Get Started" for the Community Edition, you agree to the [Databricks Community Edition Terms of Service](#).

Help



Sign up for Community Edition



Log into DBCE

Databricks Community Edition X +

https://community.cloud.databricks.com/?o=57901

Upgrade ?

Welcome to  databricks™

 Explore the Quickstart Tutorial

Spin up a cluster, run queries on preloaded data, and display results in 5 minutes.

 Import & Explore Data

Drop files or [click to browse](#)

 Create a Blank Notebook

Create a notebook to start querying, visualizing, and modeling your data.

Common Tasks

-  New Notebook
-  Upload Data
-  Create Table
-  New Cluster
-  New Job
-  Import Library
-  Read Documentation

Recents

-  Video Identification of Suspicious Behavior

What's new in v2.100

- Databricks Light GA

[View latest release notes](#)

Create a Cluster on DBCE

The screenshot shows the Databricks Clusters interface. On the left is a sidebar with icons for Databricks, Home, Workspace, Recents, Data, and Clusters. The 'Clusters' icon is selected. The main area is titled 'Clusters' and contains a blue button labeled '+ Create Cluster'. Below it are two sections: 'Interactive Clusters' and 'Automated Clusters', each with a large empty white box.

Create a Cluster on DBCE

The screenshot shows the 'Create Cluster' interface in the Databricks Control Room. The left sidebar includes icons for Home, Workspace, Recents, Data, Clusters, Jobs, and Search, with 'Clusters' currently selected. The main form is titled 'New Cluster' and contains the following fields:

- Cluster Name:** An input field with a placeholder icon.
- Databricks Runtime Version:** A dropdown menu showing 'Runtime: 5.5 LTS (Scala 2.11, Spark 2.4.3)'.
- Python Version:** A dropdown menu showing '3'.
- Instance:** A note stating 'Free 6GB Memory: As a Community Edition user, your cluster will automatically terminate after an idle period of two hours.' followed by a link to 'upgrade your Databricks subscription'.
- Availability Zone:** A dropdown menu showing 'us-west-2c'.

At the top right of the form are 'Cancel' and 'Create Cluster' buttons. Above the buttons, status information is displayed: '0 Workers: 0.0 GB Memory, 0 Cores, 0 DBU' and '1 Driver: 6.0 GB Memory, 0.88 Cores, 1 DBU'.

Create a Cluster on DBCE

The screenshot shows the 'Create Cluster' interface in the Databricks Cloud Environment (DBCE). On the left is a sidebar with icons for Home, Workspace, Recents, Data, Clusters, Jobs, and Search. The main area is titled 'Create Cluster' and 'New Cluster'. It includes a 'Cancel' button and a 'Create Cluster' button. Below these are sections for 'Cluster Name' (containing 'delta-rocks'), 'Databricks Runtime Version' (set to 'Runtime: 6.1 Beta (Scala 2.11, Spark 2.4.4)'), 'Databricks Runtime' (listing '6.1 Beta' as selected, '6.1 ML Beta', '6.1 ML Beta', and '20 more' options), and 'Instances' (set to 'Spark'). At the bottom is an 'Availability Zone' dropdown set to 'us-west-2c'. A tooltip for the runtime dropdown explains it will automatically terminate after an idle period of two hours.

Create Cluster

New Cluster | Cancel | Create Cluster

0 Workers: 0.0 GB Memory, 0 Cores, 0 DBU
1 Driver: 6.0 GB Memory, 0.88 Cores, 1 DBU

Cluster Name

delta-rocks

Databricks Runtime Version

Runtime: 6.1 Beta (Scala 2.11, Spark 2.4.4)

Databricks Runtime

- 6.1 Beta
- 6.1 ML Beta
- 6.1 ML Beta
- 20 more

Instances

Spark

Availability Zone

us-west-2c

Automatically terminate after an idle period of two hours.

Create a Cluster on DBCE

The screenshot shows the Databricks Clusters interface. On the left, a sidebar menu includes icons for Home, Workspace, Recents, Data, and Clusters, with 'Clusters' currently selected. The main area is titled 'Clusters' and features a 'Create Cluster' button. A filter bar at the top right allows switching between 'All' and 'Created by me' clusters, with 'Created by me' being selected. The 'Interactive Clusters' section contains one entry:

Name	State	Nodes	Driver	Worker	Runtime	Creator	Actions
delta-rocks	Running	1 (0 spot)	Community ...	Community ...	6.1 Beta (inclu..	denny.g.lee@...	0

The 'Automated Clusters' section below it displays the message 'No clusters found'.

Attach a Notebook to your Cluster

The screenshot shows the Databricks Community Edition interface. On the left, there's a sidebar with icons for Home, Workspace, Recents, Data, Clusters, Jobs, and Search. The main workspace is titled "Delta Lake Lab". A context menu is open over the lab name, with "Import" highlighted. Other options in the menu include Create, Clone, Rename, Move, Delete, Export, and Permissions. Below the menu, there's a section titled "Import & Explore Data" with a placeholder "Drop files or click to browse" and a "Cloud" icon. To the right, there's a "Create a Blank Notebook" button featuring a notebook icon with a plus sign. At the bottom, there are sections for "Recents" (listing "Video Identification of Suspicious Behavior") and "What's new in v2.100" (mentioning "Databricks Light GA" and a link to "View latest release notes"). The URL in the browser bar is https://community.cloud.databricks.com/?o=57901#.

Attach a Notebook to your Cluster

The screenshot shows the Databricks Community Edition interface. The main window title is "01-Delta Lake Workshop - Delta Lab". The URL in the browser is <https://community.cloud.databricks.com/?o=57901>. On the left, there's a sidebar with "Workspace" selected, showing options like Home, Workspace, Recents, Data, Clusters, Jobs, and Search. The main workspace area has a title "Delta Lake Lab". A modal dialog box titled "Import Notebooks" is open in the center. It has two radio button options: "File" (unchecked) and "URL" (checked). Below the radio buttons is a red rectangular input field. Underneath it, text says "Accepted formats: .dbc, .scala, .py, .sql, .r, .ipynb, .Rmd, .html". Below that, it says "(To import a library, such as a jar or egg, [click here](#))". At the bottom of the dialog are "Cancel" and "Import" buttons. In the background, there are sections for "Recents" (listing "Video Identification of Suspicious Behavior") and "What's new in v2.100" (listing "Databricks Light GA" and a link to "View latest release notes").

Attach a Notebook to your Cluster

The screenshot shows the Databricks Community Edition web interface. The main navigation bar at the top includes tabs for "Data" and "Clusters". The left sidebar contains icons for "databricks", "Home", "Workspace", "Recents", "Data", "Clusters", "Jobs", and "Search". The main workspace is titled "Delta Lake Lab". A modal dialog box is centered over the workspace, titled "Import Notebooks". It has two radio button options: "File" (unchecked) and "URL" (checked). Below the radio buttons is a text input field containing the URL: "https://community.cloud.databricks.com/@o=57901". A note below the URL says "Accepted formats: .dbc, .scala, .py, .sql, .r, .ipynb, .Rmd, .html". There is also a link "(To import a library, such as a jar or egg, click here)". At the bottom of the dialog are "Cancel" and "Import" buttons. The background of the workspace shows a "Recent" section with a thumbnail for "Video Identification of Suspicious Behavior" and a "What's new in v2.100" section with a bullet point about "Databricks Light GA".

Attach a Notebook to your Cluster

The screenshot shows a Databricks workspace interface. On the left is a dark sidebar with navigation icons: Databricks logo, Home, Workspace, Recents, Data, Clusters, Jobs, and Search. The main area is titled "SAISEu19 - Delta Lake Python Tutorial (Python)". A "Attached" tab is selected. A modal dialog box titled "Attach:" lists a cluster named "delta-rocks" with specifications: 6.00 GB | 0.88 Cores | DBR 6.1 Beta | Spark 2.4.4 | Scala 2.11. Below the dialog, a slide titled "Delta Lake" provides an overview of the storage format:

An open-source storage format that brings ACID transactions to Apache Spark™ and big data workloads.

- **Open format:** Stored as Parquet format in blob storage.
- **ACID Transactions:** Ensures data integrity and read consistency with complex, concurrent data pipelines.
- **Schema Enforcement and Evolution:** Ensures data cleanliness by blocking writes with unexpected.
- **Audit History:** History of all the operations that happened in the table.
- **Time Travel:** Query previous versions of the table by time or version number.
- **Deletes and upserts:** Supports deleting and upserting into tables with programmatic APIs.
- **Scalable Metadata management:** Able to handle millions of files are scaling the metadata operations with Spark.
- **Unified Batch and Streaming Source and Sink:** A table in Delta Lake is both a batch table, as well as a streaming source and sink. Streaming data ingest, batch historic backfill, and interactive queries all just work out of the box.

At the bottom of the slide, there is a logo for "SPARK+AI SUMMIT EUROPE" with the text "15 - 17 OCTOBER 2019 | AMSTERDAM | ORGANIZED BY databricks". The notebook title bar also includes "SAISEu19 - Delta Lake Python Tutorial" and the Python language indicator.

Delta Lake Python Tutorial

Start the notebook in the cluster

Broad Adoption of ML

Huge disruptive innovations are affecting most enterprises on the planet



Healthcare and Genomics



Fraud Prevention



Digital Personalization

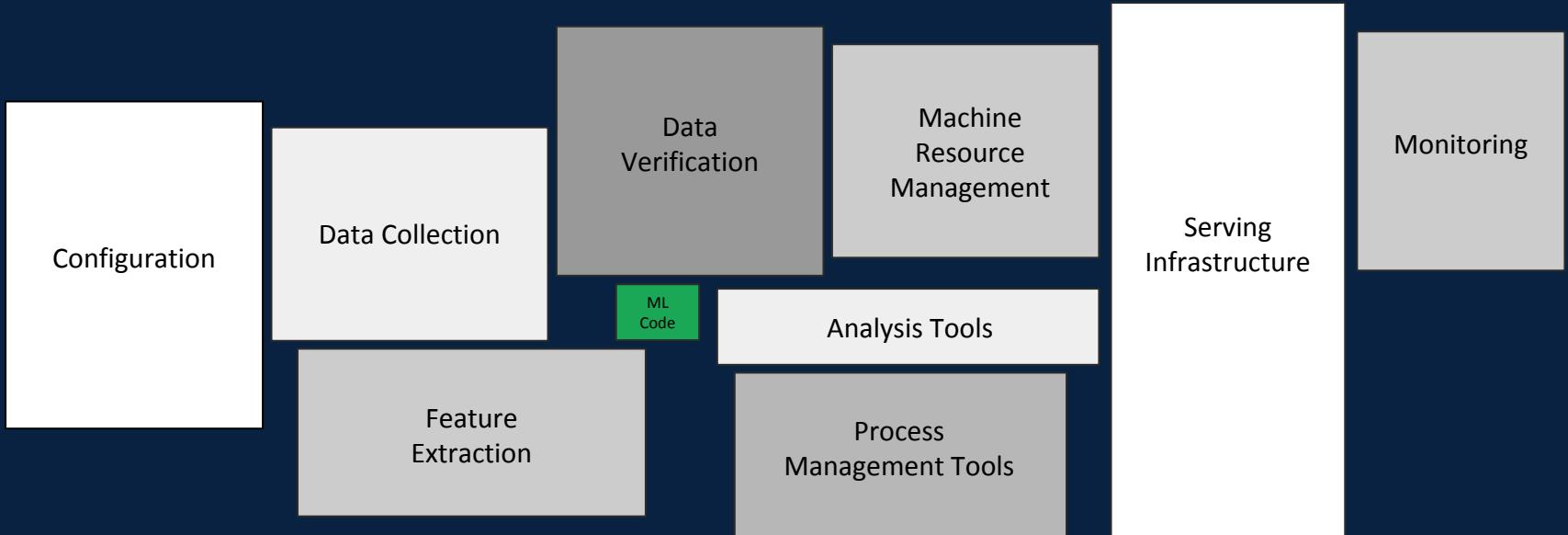


Internet of Things

and many many more customers in different industries and segments

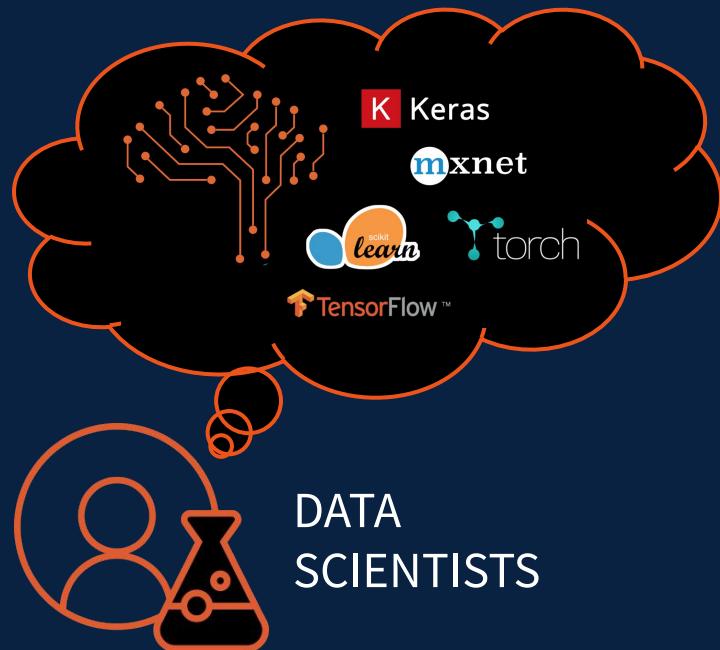
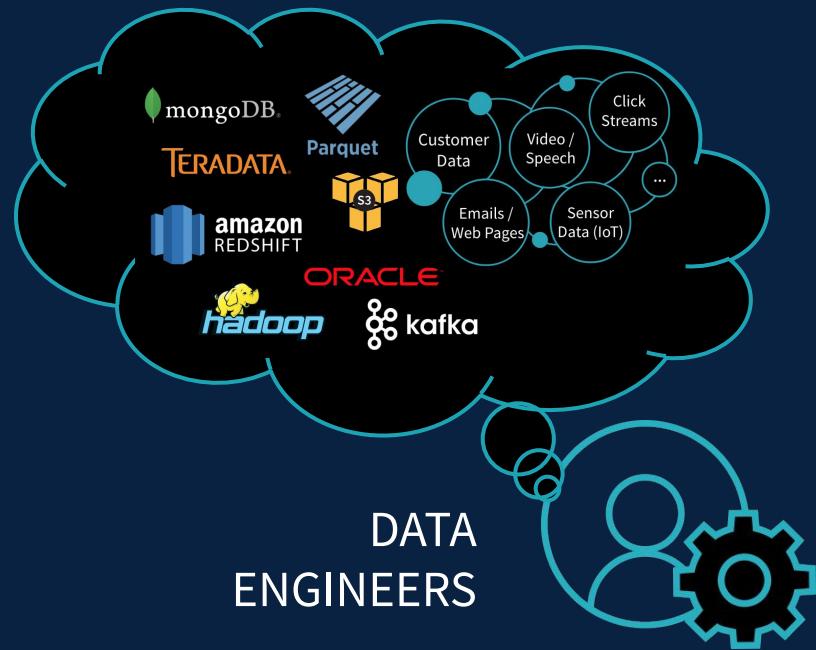
Hardest Part of ML isn't ML, it's Data

“Hidden Technical Debt in Machine Learning Systems,” Google NIPS 2015



Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small green box in the middle. The required surrounding infrastructure is vast and complex.

Data & ML Tech and People are in Silos



ML Lifecycle is Manual, Inconsistent and Disconnected

Prep Data

- Low level integrations for Data and ML
- Difficult to track data used for a model



Build Model

- Ad hoc approach to track experiments
- Very hard to reproduce experiments



Deploy Model

- Multiple tightly coupled deployment options
- Different monitoring approach for each framework



Amazon SageMaker

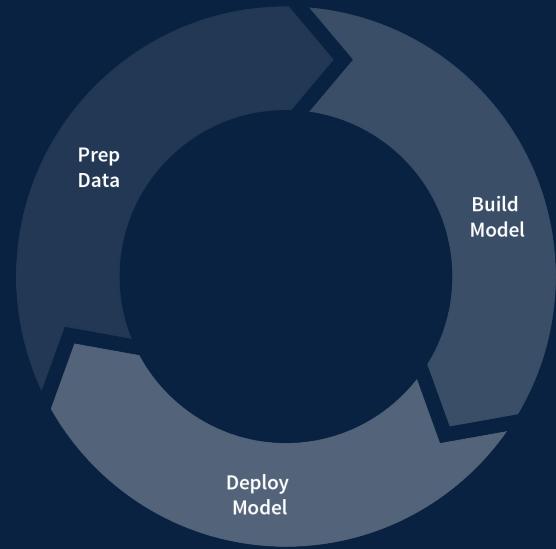


Azure Machine Learning



mlflow

The need for standardization



Day in the life of a data scientist (tracking edition)

```
Elasticnet model (alpha=0.01, l1_ratio=1.0):
```

```
RMSE: ??
```

```
MAE: 51.051828604086325
```

```
R2: 0.3951809598912357
```

```
Elasticnet model (alpha=?, l1_ratio=0.75):
```

```
RMSE: 65.28994906390733
```

```
MAE: 53.759148284349266
```

```
R2: ??
```

```
Elasticnet model (alpha=0.01, l1_ratio=?):
```

```
RMSE: 71.40362571026475
```

```
MAE: ??
```

```
R2: 0.2291130640003659
```

Day in the life of a data scientist (tracking edition)

```
Elasticnet model (alpha=0.01, l1_ratio=1.0):
```

```
RMSE: ??
```

```
MAE: 51.051828604086325
```

```
R2: 0.3951809598912357
```

```
Elasticnet model (alpha=?, l1_ratio=0.75):
```

```
RMSE: 65.28994906390733
```

```
MAE: 53.759148284349266
```

```
R2: ??
```

```
Elasticnet model (alpha=0.01, l1_ratio=?):
```

```
RMSE: 71.40362571026475
```

```
MAE: ??
```

```
R2: 0.2291130640003659
```



Did anything change in the feature engineering?

Day in the life of a data scientist (tracking edition)

```
Elasticnet model (alpha=0.01, l1_ratio=1.0):
```

```
RMSE: ??
```

```
MAE: 51.051828604086325
```

```
R2: 0.3951809598912357
```

```
Elasticnet model (alpha=?, l1_ratio=0.75):
```

```
RMSE: 65.28994906390733
```

```
MAE: 53.759148284349266
```

```
R2: ??
```

```
Elasticnet model (alpha=0.01, l1_ratio=?):
```

```
RMSE: 71.40362571026475
```

```
MAE: ??
```

```
R2: 0.2291130640003659
```



How did the hyperparameters change?

Day in the life of a data scientist (tracking edition)

```
Elasticnet model (alpha=0.01, l1_ratio=1.0):
```

```
RMSE: ??
```

```
MAE: 51.051828604086325
```

```
R2: 0.3951809598912357
```

```
Elasticnet model (alpha=?, l1_ratio=0.75):
```

```
RMSE: 65.28994906390733
```

```
MAE: 53.759148284349266
```

```
R2: ??
```

```
Elasticnet model (alpha=0.01, l1_ratio=?):
```

```
RMSE: 71.40362571026475
```

```
MAE: ??
```

```
R2: 0.2291130640003659
```



What data was this model trained on?

Day in the life of a data scientist (tracking edition)

```
Elasticnet model (alpha=0.01, l1_ratio=1.0):
```

```
RMSE: ??
```

```
MAE: 51.051828604086325
```

```
R2: 0.3951809598912357
```

```
Elasticnet model (alpha=?, l1_ratio=0.75):
```

```
RMSE: 65.28994906390733
```

```
MAE: 53.759148284349266
```

```
R2: ??
```

```
Elasticnet model (alpha=0.01, l1_ratio=?):
```

```
RMSE: 71.40362571026475
```

```
MAE: ??
```

```
R2: 0.2291130640003659
```



How did the offline metrics change?

Day in the life of a data scientist (tracking edition)

```
Elasticnet model (alpha=0.01, l1_ratio=1.0):
```

```
RMSE: ??
```

```
MAE: 51.051828604086325
```

```
R2: 0.3951809598912357
```

```
Elasticnet model (alpha=?, l1_ratio=0.75):
```

```
RMSE: 65.28994906390733
```

```
MAE: 53.759148284349266
```

```
R2: ??
```

```
Elasticnet model (alpha=0.01, l1_ratio=?):
```

```
RMSE: 71.40362571026475
```

```
MAE: ??
```

```
R2: 0.2291130640003659
```



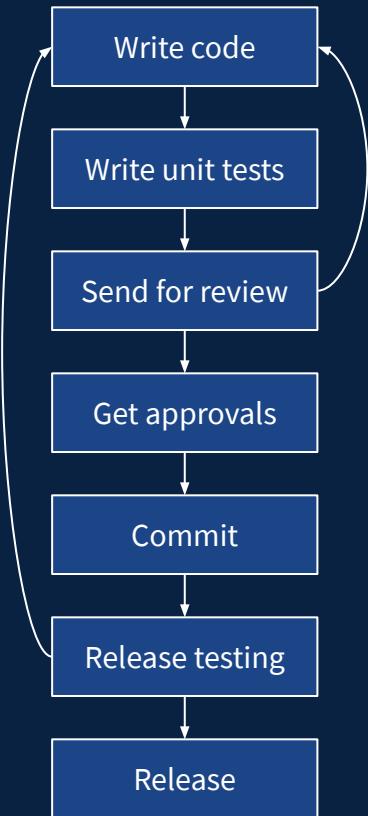
What else am I missing?

The difference between releasing Software
and deploying ML Models

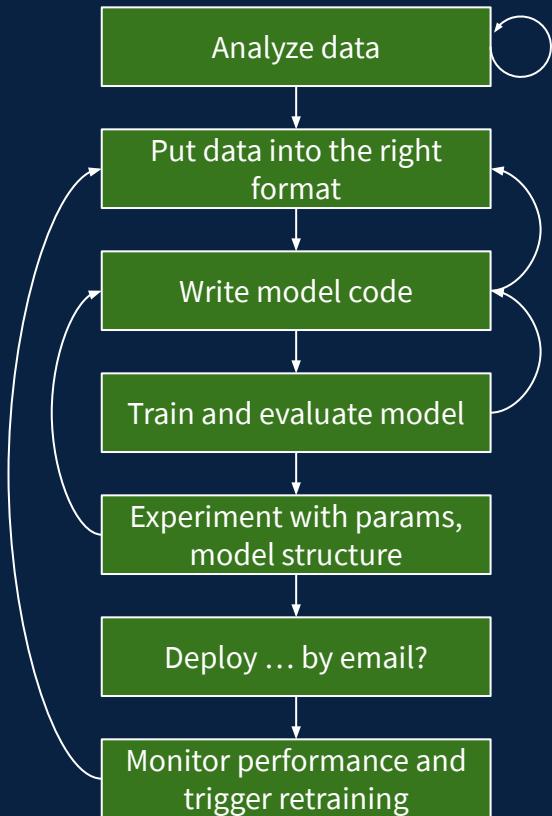
Software



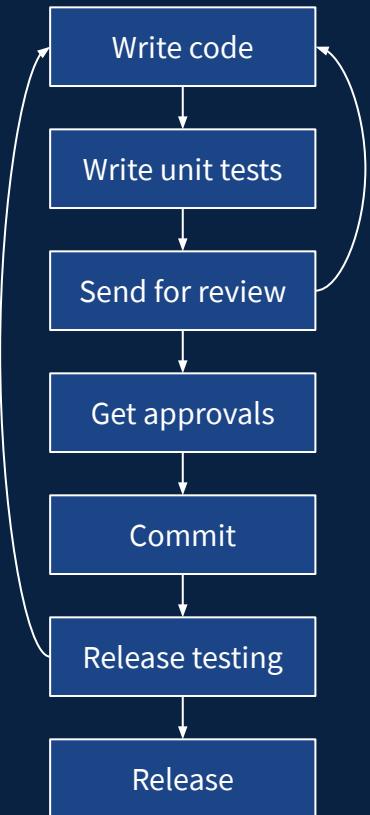
Software



ML Models



Software

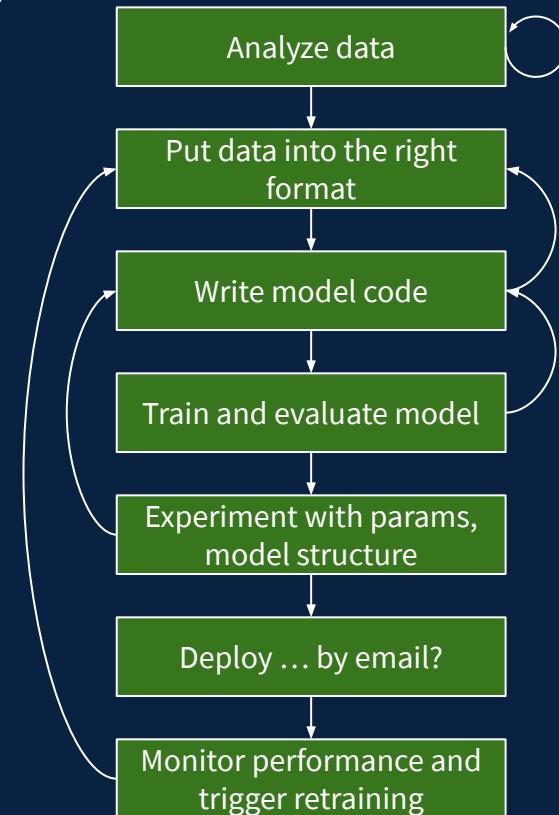


Goal

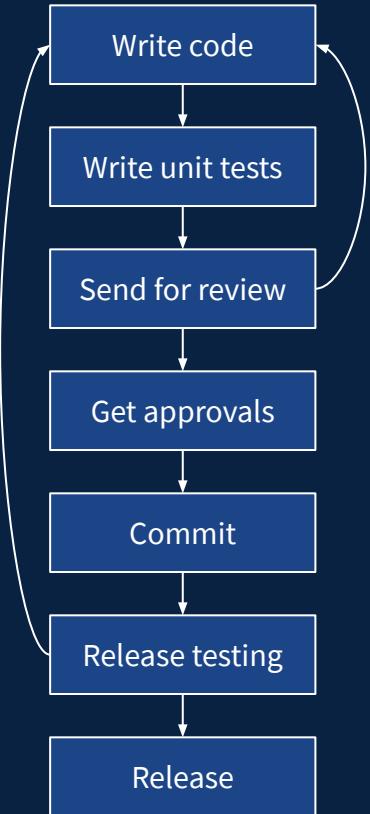
Meet a functional specification

Optimize a metric,
e.g. CTR

ML Models



Software



Goal

Meet a functional specification

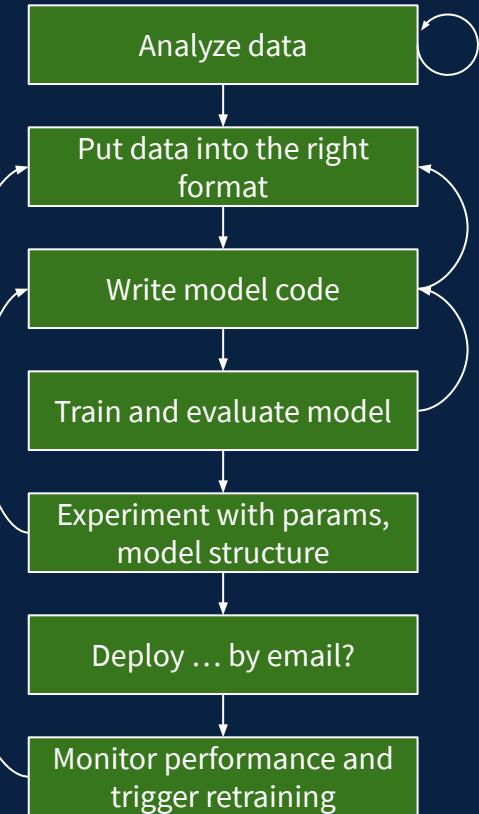
Optimize a metric,
e.g. CTR

Quality

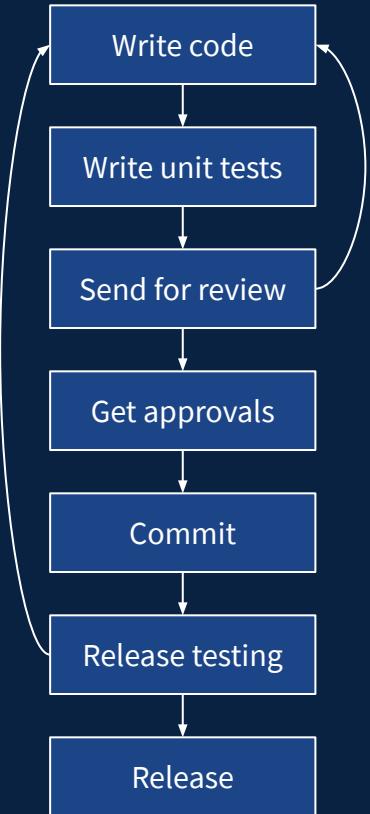
Depends on code

Depends on data,
code, model, params,
...

ML Models



Software



Goal

Meet a functional specification

Optimize a metric,
e.g. CTR

Quality

Depends on code

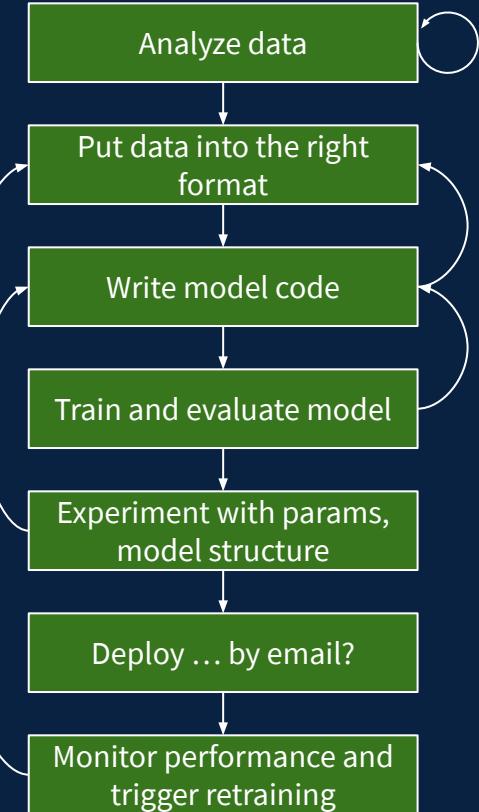
Depends on data,
code, model, params,
...

Tools

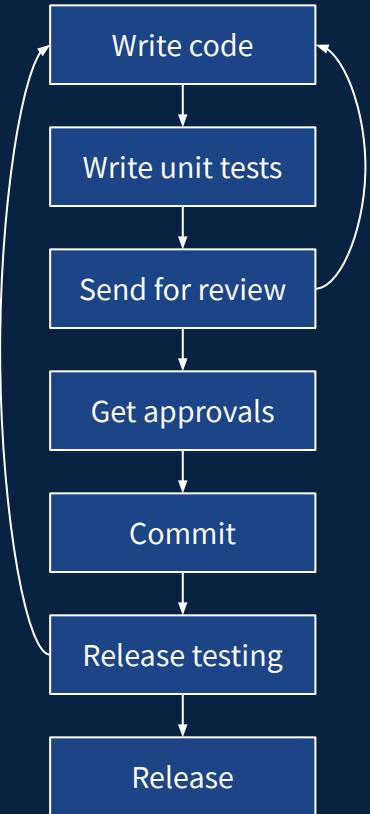
Typically one
software stack

Combination of many
libraries, tools,
...

ML Models



Software



Goal

Meet a functional specification

Optimize a metric,
e.g. CTR

Quality

Depends on code

Depends on data,
code, model, params,
...

Tools

Typically one
software stack

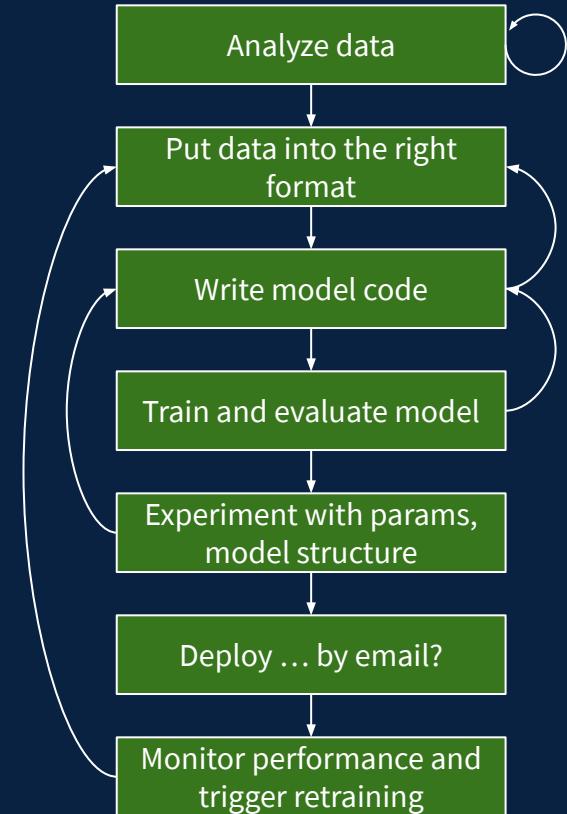
Combination of many
libraries, tools,
...

Outcome

Works
deterministically

Keeps changing with
data, etc.

ML Models

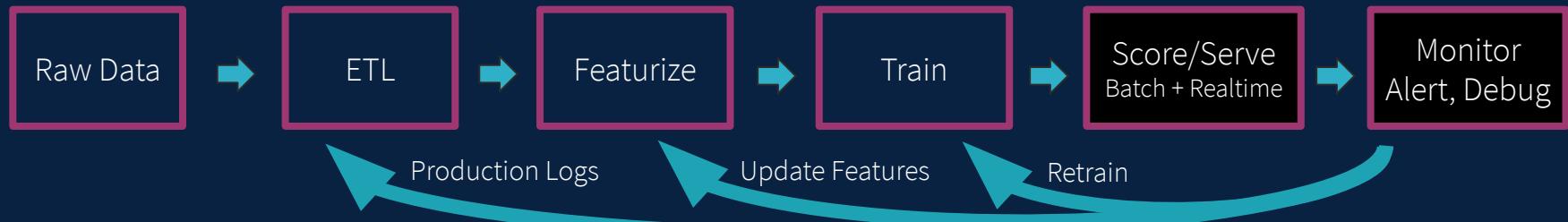


In summary, deploying ML Models is hard!

ML Lifecycle and Challenges

mlflow

An open source platform for the machine learning lifecycle



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.

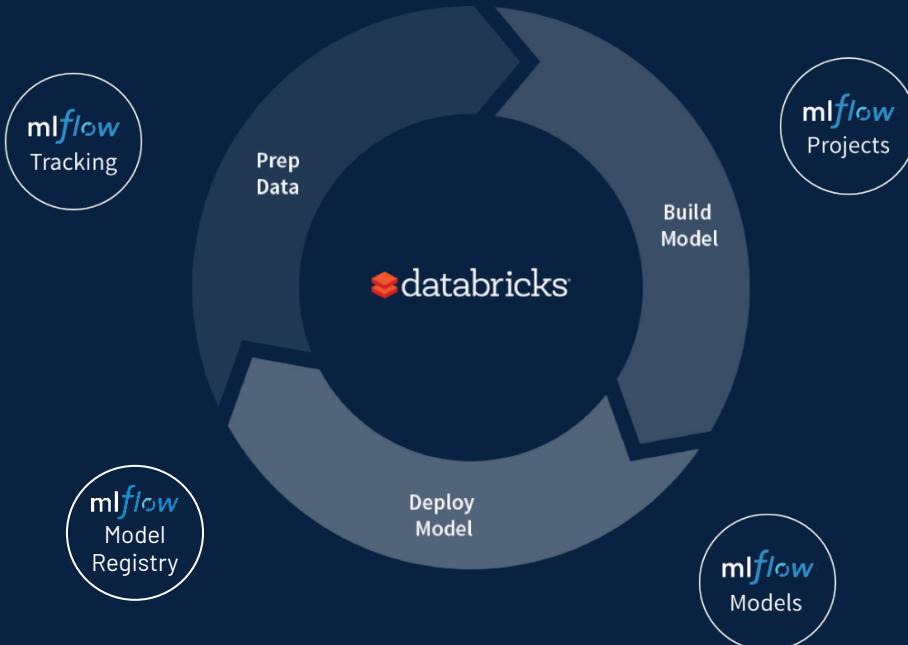
Data Drift

Model Drift

mlflow

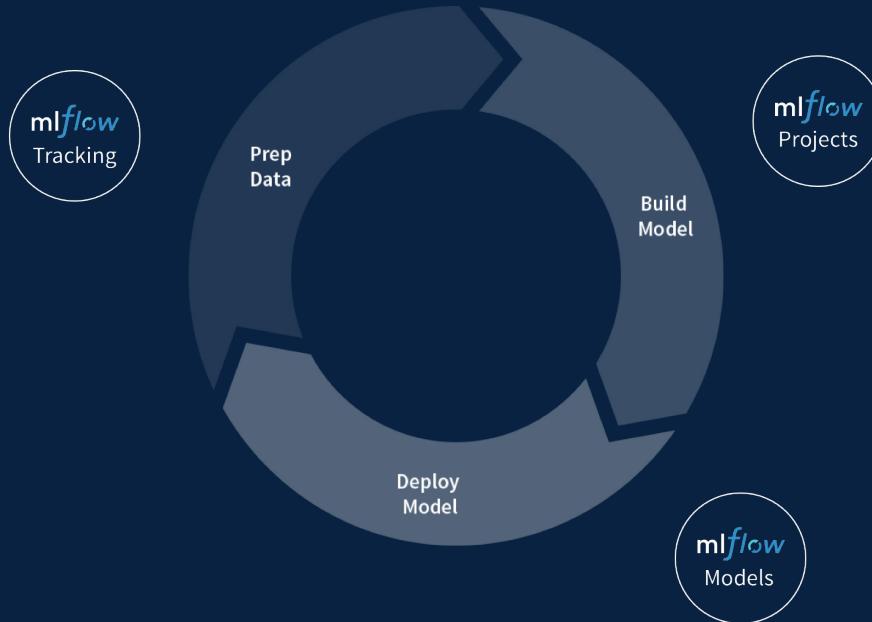
Introducing MLflow

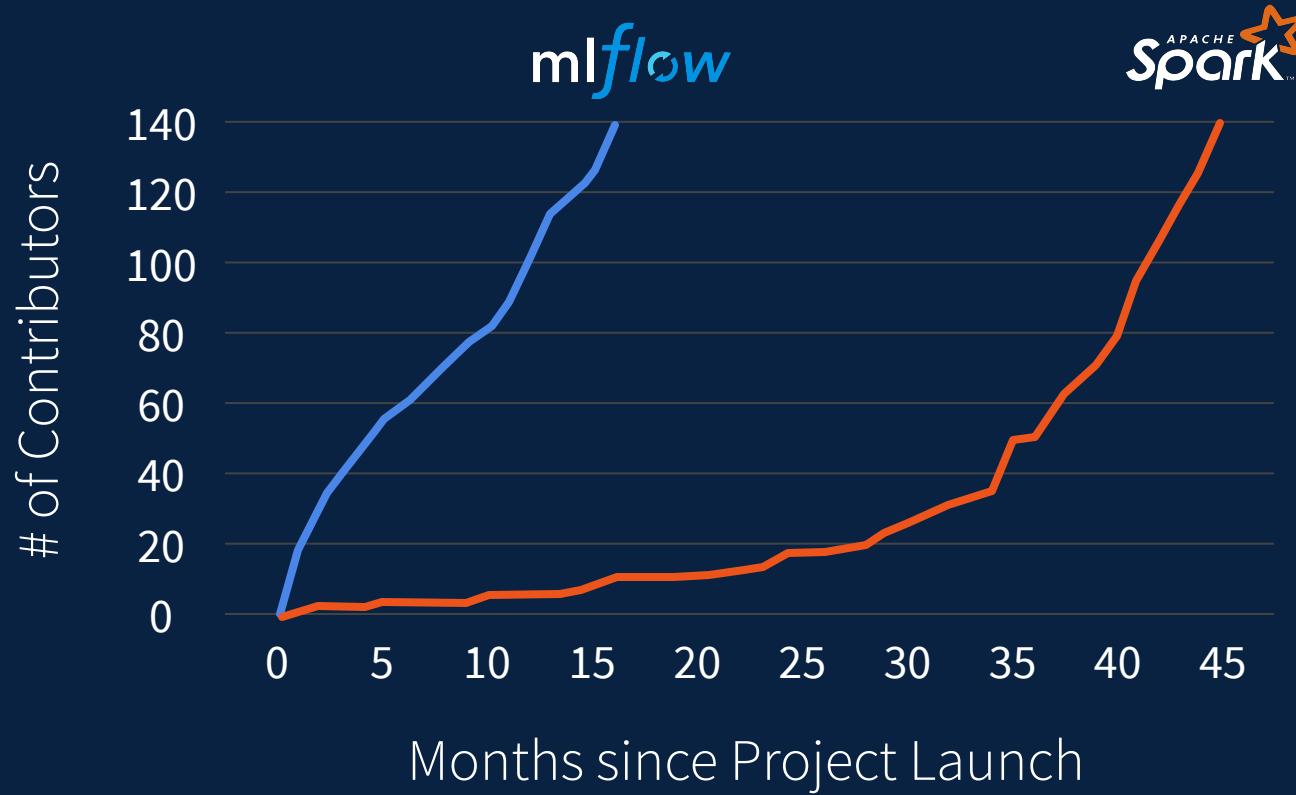
Unveiled in June 2018, MLflow is the only open source framework designed to manage the complete Machine Learning Lifecycle.



Introducing MLflow

Unveiled in June 2018, MLflow is the only open source framework designed to manage the complete Machine Learning Lifecycle.





mlflow Components

mlflow Tracking

Record and query experiments: code, data, config, results

mlflow Projects

Packaging format for reproducible runs on any platform

mlflow Models

General format that standardizes deployment paths

mlflow Model Registry

Centralized and collaborative model lifecycle management

mlflow Components

mlflow Tracking

Record and query experiments: code, data, config, results

mlflow Projects

Packaging format for reproducible runs on any platform

mlflow Models

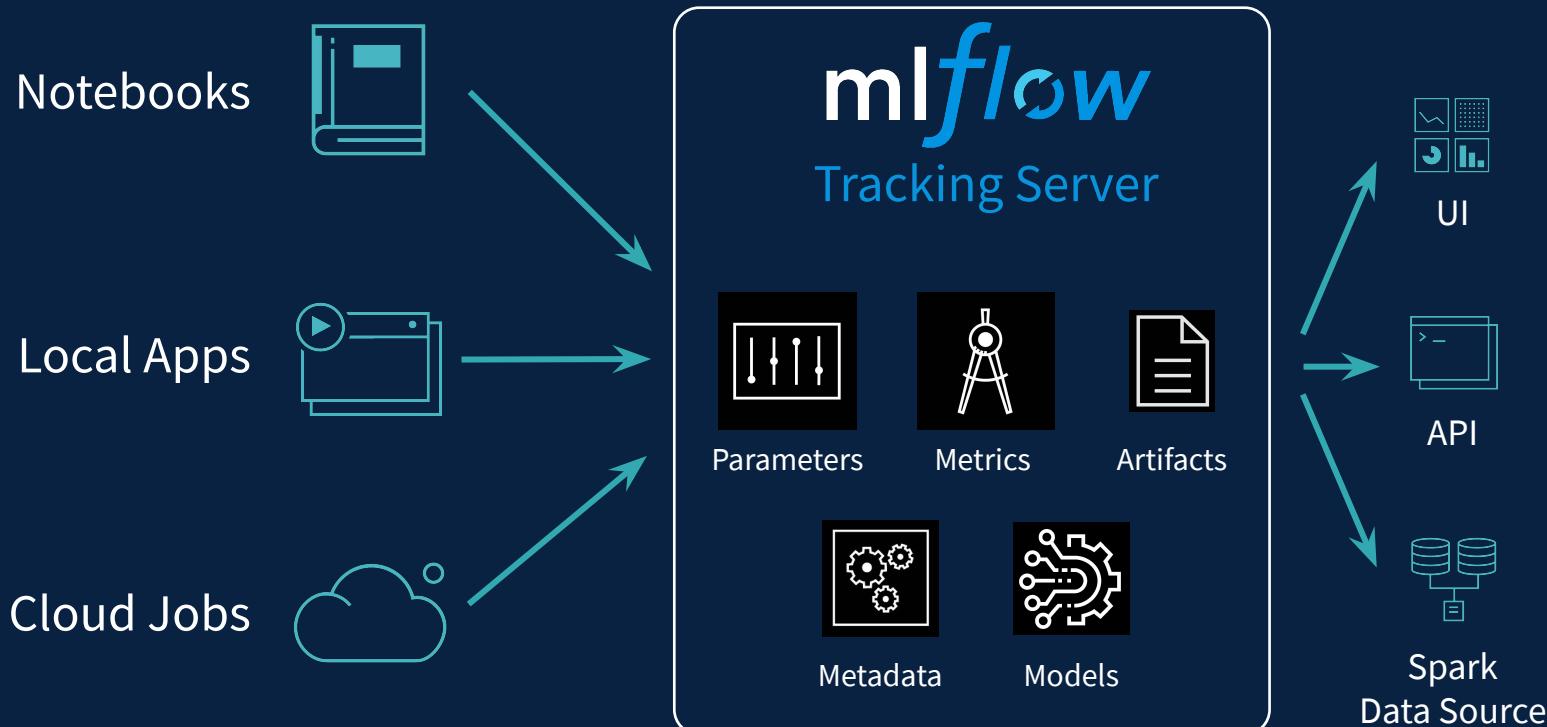
General format that standardizes deployment paths

new

mlflow Model Registry

Centralized and collaborative model lifecycle management

mlflow Tracking



Key Concepts in Tracking

Parameters: key-value inputs to your code

Metrics: numeric values (can update over time)

Artifacts: arbitrary files, including models

Source: what code ran?

The screenshot shows the mlflow UI interface. At the top, there's a navigation bar with links for GitHub and Docs. Below that, the main header says "mlflow". On the left, there's a sidebar titled "Experiments" with tabs for "Default" and "Something". The "Default" tab is selected. To its right, the main content area shows experimental runs. It displays the Experiment ID (0) and Artifact Location (/Users/matei/mlflow/mlruns/0). There are search and filter fields for "Search Runs" (metrics.rmse < 1 and params.model = "tree") and "Filter Params" (alpha, lr) and "Filter Metrics" (rmse, r2). A "Clear" button is also present. Below these, it says "4 matching runs" and provides buttons for "Compare Selected" and "Download CSV". A table then lists the runs with columns for Date, User, Source, Version, Parameters, and Metrics. The table shows four rows of data.

Date	User	Source	Version	Parameters	Metrics
2018-06-28 17:09:49	matei	matei_test.py	7cff8e	(n/a)	loss: 2.123
2018-06-28 17:09:06	matei	matei_test.py	7cff8e		loss: 4.543
2018-06-28 17:09:05	matei	matei_test.py	7cff8e		loss: 4.543
2018-06-25 13:08:12	matei	matei_test.py	53ccdc		loss: 4.543

```
# Scikit Learn Linear Regression via ElasticNet
lr = ElasticNet(alpha=alpha, l1_ratio=l1_ratio, random_state=42)
lr.fit(train_x, train_y)

# Predict
predicted_qualities = lr.predict(test_x)

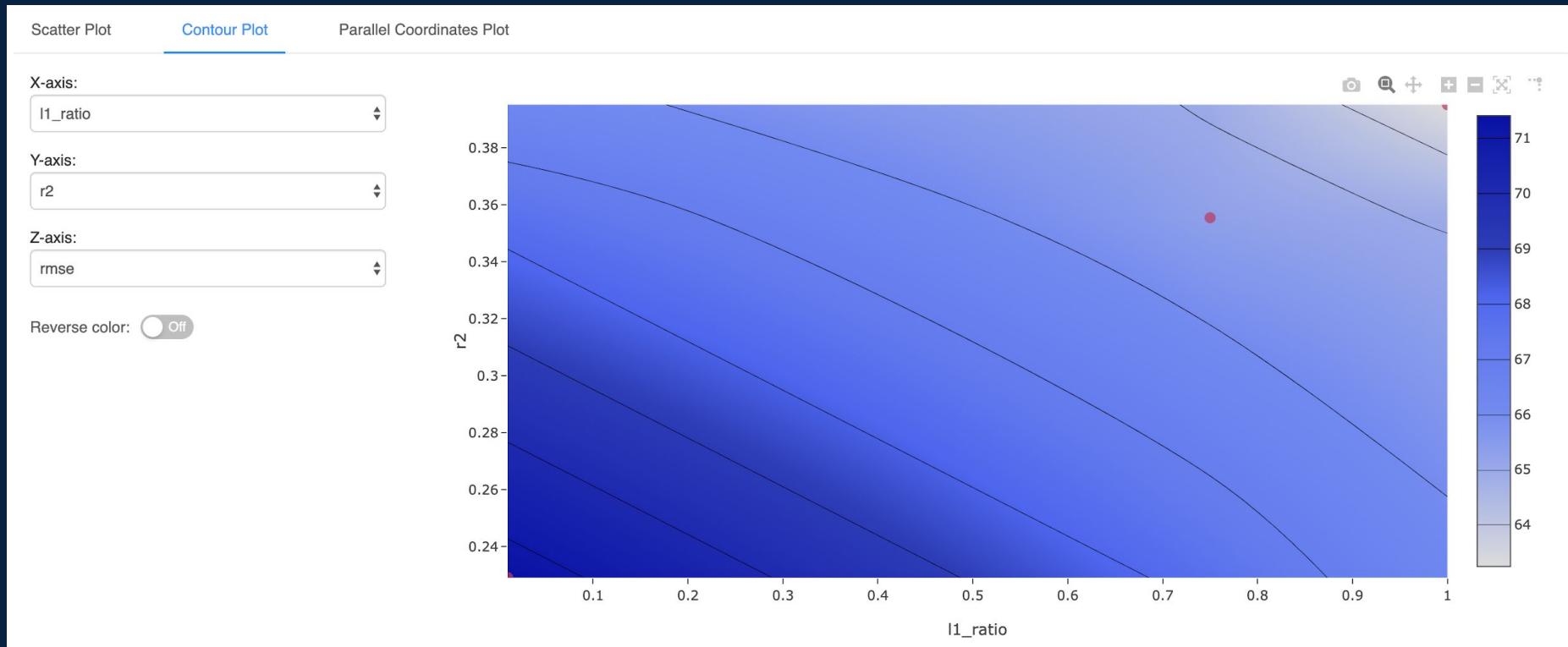
# Evaluate Metrics
(rmse, mae, r2) = eval_metrics(test_y, predicted_qualities)
```

```
with mlflow.start_run() as run:  
  
    # Scikit Learn Linear Regression via ElasticNet  
    lr = ElasticNet(alpha=alpha, l1_ratio=l1_ratio, random_state=42)  
    lr.fit(train_x, train_y)  
  
    # Predict  
    predicted_qualities = lr.predict(test_x)  
  
    # Evaluate Metrics  
    (rmse, mae, r2) = eval_metrics(test_y, predicted_qualities)  
  
    # Log  
    mlflow.log_param("alpha", alpha)  
    ...
```

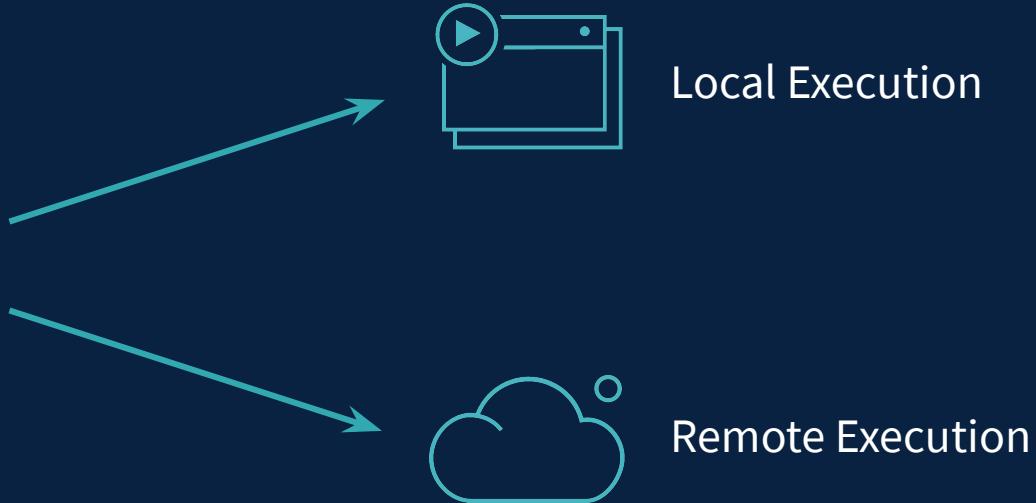
GitHub Demo

<https://github.com/dennyglee/mlflow-diabetes-example>

Comparing Runs Contour Plot



mlflow Projects



Example MLflow Project

```
my_project/
  └── MLproject
      └── conda.yaml
      └── main.py
      └── model.py
      ...
      
```

```
conda_env: conda.yaml

entry_points:
  main:
    parameters:
      training_data: path
      lambda: {type: float, default: 0.1}
    command: python main.py {training_data} {lambda}
```

```
$ mlflow run git://<my_project>
mlflow.run("git://<my_project>", ...)
```

mlflow Models

ML Libraries



mlflow

Model Format

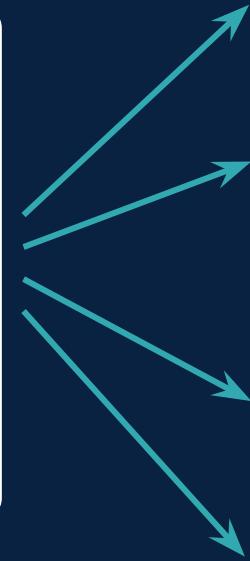
Flavor 1



Flavor 2



Simple model flavors
usable by many tools



In-Line Code



Containers



Batch & Stream Scoring

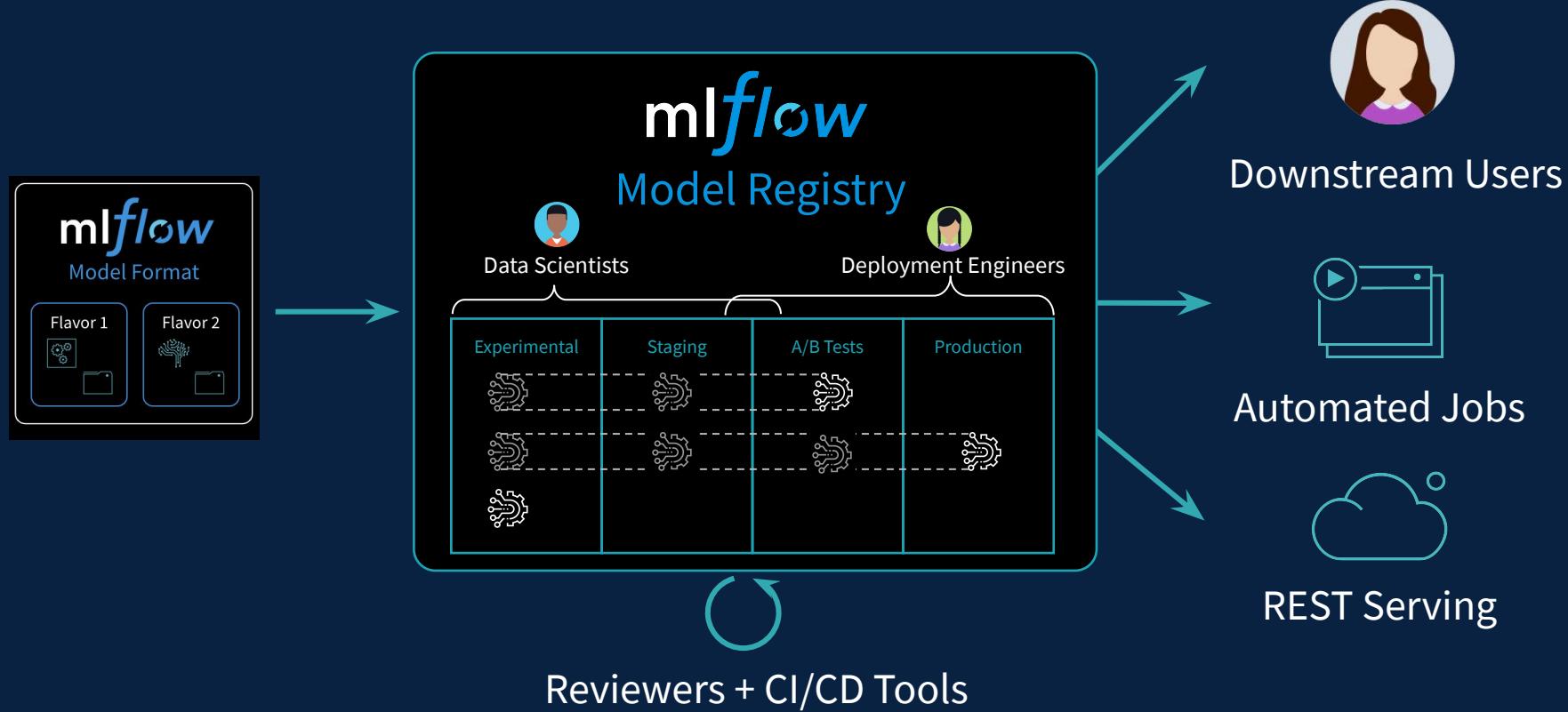


Cloud Inference Services

Example MLflow Model



mlflow Model Registry



mlflow Model Registry: Benefits

The screenshot shows the Databricks MLflow Model Registry interface. On the left, there's a sidebar with icons for Home, Workspace, Projects, Recents, Data, Clusters, and Models (which is highlighted with a green circle and the number 1). The main area is titled "Registered Models" and displays a table of registered models. The columns are: Name, Latest Version, Staging, Production, and Last Modified. The table contains the following data:

Name	Latest Version	Staging	Production	Last Modified
AaronModel	Version 108	—	—	2019-10-18 09:04:20
Airline_Delay_Scikit	Version 3	—	Version 1	2019-10-11 12:41:43
Airline_Delay_SparkML	Version 11	Version 6	Version 5	2019-10-22 10:30:21
Andre_BasicModels_02_Sklearn_Train_Predict	Version 2	Version 1	—	2019-10-19 14:38:11
BertisLarge	Version 1	—	—	2019-10-11 15:18:05
Brooke Keras Model	Version 1	—	—	2019-10-12 08:20:12
holland-forecast-model	Version 1	—	Version 1	2019-10-07 15:38:27
joytesting	—	—	—	2019-10-15 18:23:17
ManiErrorModel	—	—	—	2019-10-14 16:53:10
MateiModel	Version 5	Version 5	Version 3	2019-10-10 14:07:07

Three specific features are highlighted with green circles and numbers: 2 highlights the "Airline_Delay_Scikit" and "Airline_Delay_SparkML" rows; 3 highlights the search bar and the pagination controls at the bottom right.

One Collaborative Hub

- 1 Central Model Repository
- 2 Overview of versions in Staging/Production/etc.
- 3 Search/filter/pagination

mlflow Model Registry: Benefits

The image shows two screenshots of the mlflow Model Registry interface. The left screenshot displays the 'Registered Models > Airline_Delay_SparkML' page. It includes a 'Description' section with a text input field containing the model's purpose, and a 'Versions' section listing two active versions: Version 5 (Production stage) and Version 6 (Staging stage). A green circle labeled '1' highlights the 'Pending Requests' section at the bottom of the 'Versions' table. The right screenshot shows the 'Stage: Staging' page, which contains a list of pending requests for transitioning the model to different stages. One request, 'Request transition to Production', is highlighted with a green border and a green circle labeled '2'. Other requests include 'Request transition to Archived' and 'Transition to None'.

Version	Registered at	Created by	Stage
Version 5	2019-10-11 12:44:44	clemens.mewald@databricks.com	Production
Version 6	2019-10-16 03:15:56	clemens.mewald@databricks.com	Staging

Management of the entire ML Lifecycle (MLOps)

- 1 Overview of active model versions and their deployment stage
- 2 Request/Approval workflow for transitioning deployment stages

mlflow Model Registry: Benefits

Registered Models > Airline_Delay_SparkML > Version 5 ▾

Registered At: 2019-10-11 12:44:44 Creator: clemens.mewald@databricks.com Stage: Production ▾

Last Modified: 2019-10-22 09:03:28 Source Run: [Run 6151fe768a5e49d39076b07448e60d57](#)

▼ Description 

Improved the Airline delay model using a GBDT. See run for improved metrics.

1 Pending Requests

Activities

-  clemens.mewald@databricks.com applied a stage transition None → Production 11 days ago
What can go wrong?
-  clemens.mewald@databricks.com requested a stage transition Production → None 8 days ago
-  clemens.mewald@databricks.com rejected a stage transition → None 8 days ago

Visibility

- 1 Full activity log of stage transition requests, approvals, etc.

mlflow Model Registry: Benefits

The screenshot shows three views illustrating the model's history:

- Top View:** Registered Models > Airline_Delay_SparkML > Version 5. It shows the model was registered at 2019-10-11 12:44:44 by clemens.mewald@databricks.com in Production stage. A green callout labeled 1.a points to the "Source Run" field, which contains the run ID: Run 6151fe768a5e49d39076b07448e60d57.
- Middle View:** /Users/clemens.mewald@databricks.com/Airline Demo/... > Run 6151fe768a5e49d39076b07448e60d57. It shows the run details: Date: 2019-10-11 12:20:44, User: clemens.mewald@databricks.com, Duration: 12.8s. A green callout labeled 1.b points to the "Source" field, which links to "02.2 Model Search".
- Bottom View:** 02.2 Model Search (Python). It shows the notebook revision history: You are viewing a notebook revision from Oct 11 2019, 10:09 AM PDT. A green callout labeled 1.c points to this message.

Code snippet from the notebook:

```
def train(params):
    if params['type'] == 'spark_rf':
        regressor = RandomForestRegressor(featuresCol="features",
                                          labelCol="ArrDelay",
                                          maxBins=348,
                                          seed=42,
```

Governance and Auditability

- Full provenance from Model marked production in the Registry to ...
- Run that produced the model
- Notebook that produced the run
- Exact revision history of the notebook that produced the run

Notebook Demo

[https://github.com/dennyglee/tech-talks/blob/master/samples/MLflow%20Diabetes%20Example%20\(with%20MLflow%20Registry\).ipynb](https://github.com/dennyglee/tech-talks/blob/master/samples/MLflow%20Diabetes%20Example%20(with%20MLflow%20Registry).ipynb)

mlflow: An Open Source ML Platform

Towards more principled
Data Science and ML



Hands-on Workshop

bit.ly/mlflow-boss-2020