### **MLflowバージョンアップ情報**

[すべてを見る](https://openstandia.jp/oss_info/pycaret/version/)

更新日：2021-09-07

### **MLflowの概要**



　MLflowは、実験、再現、配備（デプロイ）、モデルの保存など、機械学習のライフサイクルを管理するためのオープンソースプラットフォームです。MLflowを使用すると、機械学習の実験結果を追跡して整理し、他の機械学習エンジニアに説明したり、機械学習モデルをパッケージ化し、デプロイしたりすることができます。大規模な組織での利用が可能になるようにスケーラビリティを考慮した設計がされていますが、個人のユーザーにも最適なOSSです。

TOPに戻る

### **MLflowの機能**

現在、MLflowは次の4つのコンポーネントを提供しています。

* **MLflow Tracking**

実験（コード、データ、設定、結果）の記録と検索を行う

* **MLflow Projects**

任意のプラットフォームでコードの実行を再現できる形式でパッケージ化する

* **MLflow Models**

多様なサービス環境に機械学習モデルをデプロイする

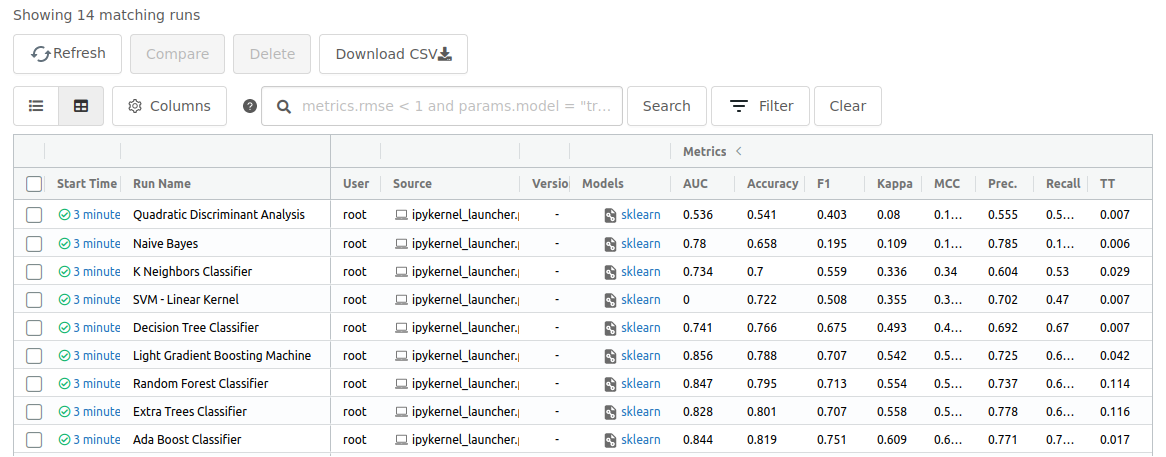
* **MLflow Repositry**

セントラルリポジトリでモデルを保存、アノテーション付け、検出、および管理を行う

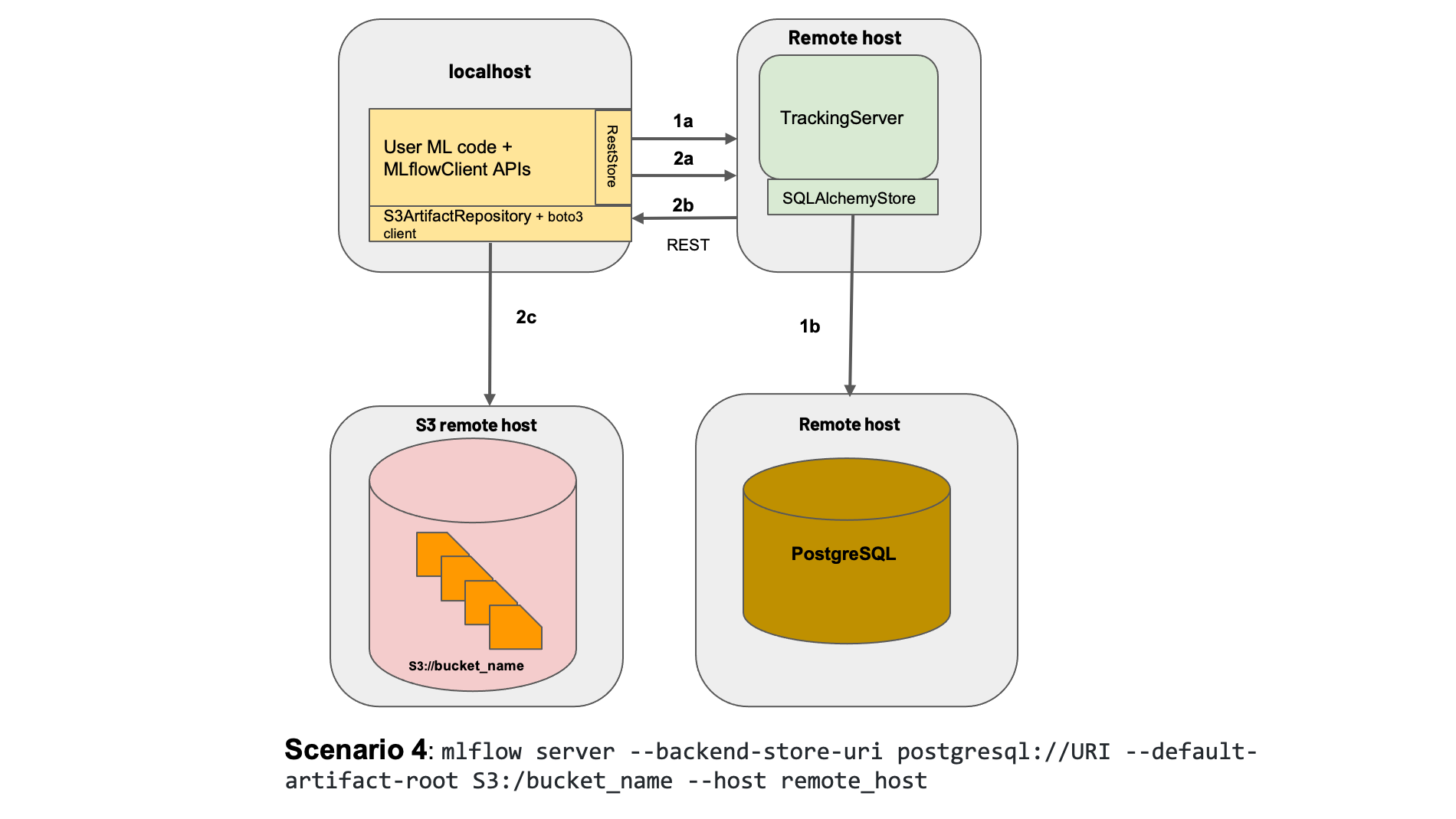
それぞれについてもう少し詳しく紹介します。

**MLflow Tracking**

　MLflow Trackingにより、実験の記録と検索が可能です。コードのバージョンや使用したハイパーパラメータ、実行結果などの情報をファイルやRDBMSに保存し、GUIで解析することができます。



以下の例のように、リモートMLflow Tracking Server、バックエンドエンティティストレージ用のPostgresデータベース、およびアーティファクトストレージ用のS3バケットを備えた分散アーキテクチャもサポートします。

（出典：<https://mlflow.org/docs/latest/tracking.html>）

**MLflow Projects**

　MLflow Projectsは、他者のコードの再利用と実行結果の再現ができるようにするためのパッケージングを行います。以下のようなYAML形式のテキストファイルをプロジェクトのルートディレクトリに追加することで、MLflowプロジェクトを制御します。

| name: My Project  conda\_env: my\_env.yaml  *# Can have a docker\_env instead of a conda\_env, e.g.*  *# docker\_env:*  *# image: mlflow-docker-example*  entry\_points:  main:  parameters:  data\_file: path  regularization: {type: float, default: 0.1}  command: "python train.py -r {regularization} {data\_file}"  validate:  parameters:  data\_file: path  command: "python validate.py {data\_file}" |
| --- |

**MLflow Models**

　学習したモデルを決められたフォーマットでパッケージングしたり、デプロイする機能です。学習済みのモデルを利用するために必要なCondaやDockerの環境を構築したり、学習済みのモデルで推論を行うREST APIサーバーを起動することが、以下のようにコマンド1つでできます。

| $ mlflow models serve --model-uri my\_model |
| --- |

**MLflow Registry**

　MLflow Registryは、MLflowモデルのライフサイクル全体を共同で管理するための、一元化されたモデルストア、APIのセット、およびUIです。提供される機能は、モデルの系統（モデルが作成されたMLflowの実験と実行）やのバージョンの管理、ステージの移行（ステージングから本番への移行など）、およびアノテーションなどです。

TOPに戻る

### **MLflowの特徴**

MLflowの公式サイトでは、主な特徴として以下の4つを挙げています。



* 機械学習のライブラリやプログラミング言語に依存せず、既存のコードでも動作する
* どのクラウドサービスでも同じように実行できる
* 1人のユーザーの利用から大規模な組織での利用まで利用規模に応じて、スケールするように設計されている
* ビッグデータに対してApache Sparkを使用したスケールが可能

TOPに戻る

### **MLflowのライセンス形式**

MLflowのライセンスは、Apache License 2.0となっており、ユーザーはそのソフトウェアの使用や頒布、修正、派生版の頒布をすることに制限を受けません。

MLflowのライセンスについてご不明な点がありましたら、お気軽に[お問い合わせ](https://openstandia.jp/site/contact.html)ください。

TOPに戻る

### **MLflowと同類のソフトウェア**

機械学習のライフサイクル管理を行う場合、商用のクラウドサービスであればNeptuneやWeights & Biasesなどいくつか選択肢がありますが、OSSでは現時点でMLflowの一択と言えるかもしれません。

TOPに戻る

### **NRIのMLflowサポート（保守/サポートサービス）**

現在、OpenStandiaではMLflowのサポートを実施していません。

※詳細は[お問い合わせ](https://openstandia.jp/site/contact.html)ください。

TOPに戻る

### **関連OSS**

* Jupyter Notebook
* [Pyacret](https://openstandia.jp/oss_info/pycaret/)
* [scikit-learn](https://openstandia.jp/oss_info/scikit-learn/index.html)
* [TensorFlow](https://openstandia.jp/oss_info/tensorflow/)
* [PyTorch](https://openstandia.jp/oss_info/pytorch/)

TOPに戻る

### **参考情報**

* [公式サイト](https://mlflow.org/)
* [公式ドキュメント](https://mlflow.org/docs/latest/index.html)
* [GitHub](https://github.com/mlflow/mlflow/) - MLflow

TOPに戻る