

SAS Viya Workbenchの御紹介

SAS Institute Japan 株式会社

2024/10/18
Toshiaki Habu



SAS Viya Workbench ご紹介

一言で言いますと、利用者自らがクラウドサービスのリソースを使った環境を立ち上げることが出来る、SASプログラミング・Pythonプログラミング環境になります。

Workbench製品特徴

- ✓ クラウド上に分析環境を数クリックで迅速にセットアップ
- ✓ セルフプロビジョニング機能により、ITサポートなしで柔軟にリソース立ち上げ・管理可能
- ✓ 多様なデータソースに安全にアクセス
- ✓ Visual Studio CodeやJupyter Labと統合し、統一された開発環境で既存のプロジェクトを簡単に移行可能
- ✓ SASとPython、R※の多言語をサポート（異なるコーディング言語に対応）
- ✓ クラウドの COMPUTE コストを最小限に抑え、インフラのコスト管理が容易
- ✓ GitHub統合により、分散バージョン管理とチームコラボレーションを容易化

※R言語には今後対応予定



SAS Viya Workbench ご紹介

SAS Viya Workbenchは分析開始から終了までの全ての段階において、簡潔な操作で環境構築に必要なアクションが可能です。開発ユーザーは柔軟性と自由度を保ったまま開発を行うことができ、システム部門はコスト管理とともにガバナンスを保ちながらの環境増設が可能になります。

各部門の分析者



システム部門に頼らずに自分自身で素早く環境利用が可能！

リソースサイズを自由に選択し拡張も可能

自由にエディタを選択

システム部門



セキュリティとガバナンスを保ちつつコストを管理した増設が可能！

SAS・Python
複数言語に対応

全セッションを一
元管理

セッションの削除も1クリック

SAS Viya Workbench ご紹介

SAS Viya Workbenchは全社的なガバナンスに基づいたクラウドベースの環境整備を行うことで、既存の分析資産が乱立していたAI開発の多くの課題点に対する解決方法をご提供いたします。



課題点

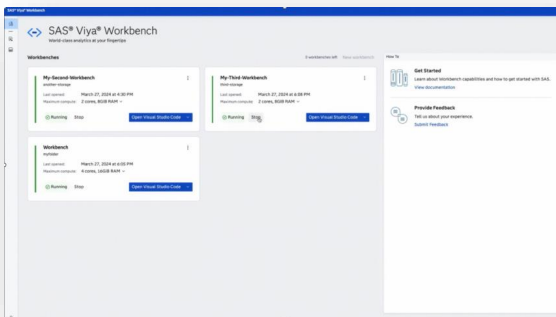
1. 環境が違えばアクセス効率が悪く、データハンドリングがしづらい
2. アドホックな分析を行いたいとなると新しい環境用意が必要となる。複数の異なるプログラム言語の使用による管理効率の悪化
3. 分析結果やコードの管理が非統一的で再検証や更新が困難



WORKBENCHによる解決法

1. ビルトインアクセスエンジンによるスムーズなデータアクセス
2. クラウド環境でセルフプロビジョニングを可能にし、計算リソースを柔軟に管理可能。
SASとPython、Rをサポートし、Editorが統合されたプラットフォーム
3. モデルのデプロイ、パフォーマンスを一括監視、更新を簡素化

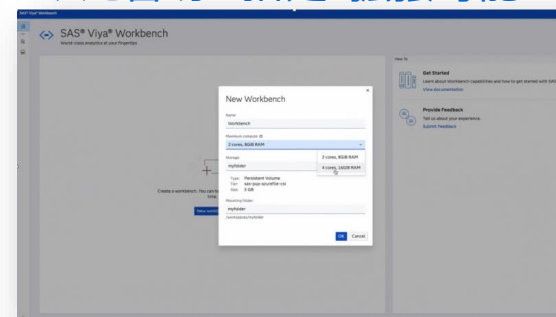
✓ 数クリックで分析セッションを
スピンアップ



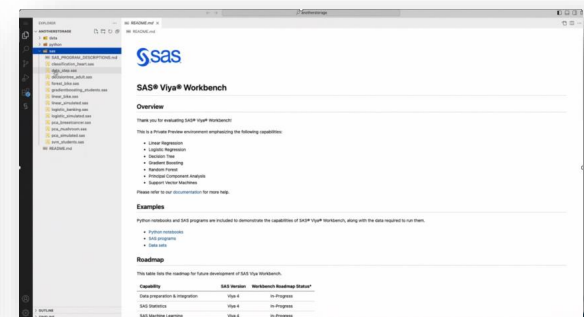
✓ 各種言語に対する共通
プラットフォーム



✓ セルフプロビジョニングに環境サイ
ズを容易に指定・拡張可能



✓ VS Code, Jupyter Labなど
の開発環境の統合



SAS Viya Workbench

demonstration



SAS® Viya® Workbench

World-class analytics on demand.

Environments

Filter



New environment

Model Training

[Primary Azure Data](#), [Training Code Repository](#)

Last opened: Mar 6, 2023, 9:57 AM by jdoe

Compute size: 4 vCPU, 16GB RAM (default) ▾

Open Visual Studio Code ▾

Model Deployment

[Primary Azure Data](#), [Deployment Code Repo](#)

Last opened: Feb 23, 2023, 11:36 AM by jdoe

Compute size: 4 vCPU, 16GB RAM (default) ▾

Running Stop

Open Visual Studio Code ▾

環境の作成

環境の起動/停止

Editorの選択

HOW TO



Overview

Watch this video to learn about the capabilities in SAS Viya Workbench.

[Watch video](#)



Get Started

Create your first environment and learn how to get started with SAS analytics.

[View documentation](#)



Import Data

Learn how to import data, preview its schema, create a table, and query it in a notebook.

[Watch video](#)



Provide Feedback

Report any issues you are having or share your thoughts on how we can improve.

[Open feedback form](#)



SAS® Viya® Workbench

World-class analytics at your fingertips

Workbenches

2 workbenches left New workbench

Workbench

myfolder

Last opened: 2024年4月23日 午後2:24

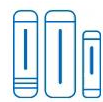
Maximum compute: 4 vCPU, 16GiB RAM ▾

✔ Running Stop

⋮

Open Jupyter Lab ▾

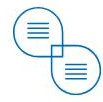
How To



Get Started

Learn about Workbench capabilities and how to get started with SAS.

[View documentation](#)



Provide Feedback

Tell us about your experience.

[Submit Feedback](#)

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

```
[7]: sasrf = ForestClassifier(n_trees=100,
                           max_depth=5,
                           min_samples_leaf=1,
                           max_features=None,
                           criterion='gini',
                           seed=0)

sasrf.fit(X_train, y_train)
```

```
[7]: <scikit-learn.tree._classes.ForestClassifier at 0x7f99537df350>
```

```
[8]: ## now time the training steps - %%timeit doesn't produce output so we run the timing separately from the training
```

```
[9]: %%timeit
rf = RandomForestClassifier(n_estimators=100,
                           max_depth=5,
                           min_samples_leaf=1,
                           max_features=None,
                           bootstrap=True,
                           criterion='gini',
                           random_state=0)

rf.fit(X_train, y_train)
```

```
57.6 s ± 549 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
```

```
[10]: %%timeit
sasrf = ForestClassifier(n_trees=100,
                        max_depth=5,
                        min_samples_leaf=1,
                        max_features=None,
                        criterion='gini',
                        seed=0)

sasrf.fit(X_train, y_train)
```

```
3.65 s ± 23.6 ms per loop (mean ± std. dev. of 7 runs, 1 loop each)
```

```
[11]: # Predict and assess model
rf_misclassification = 1 - accuracy_score(y_valid, rf.predict(X_valid))
sasrf_misclassification = 1 - accuracy_score(y_valid, sasrf.predict(X_valid))
print('Misclassification Rates\n', 'SKLEARN RF:', rf_misclassification, '\n', 'SAS RF:', sasrf_misclassification)
```

```
Misclassification Rates
SKLEARN RF: 0.11571323242399267
SAS RF: 0.1205382313397233
```

```
[12]: rf_score = rf.predict_proba(X_valid)
```


医薬品業界における想定される使い方/メリットについて

SAS Viya Workbench

- Real World Dataのような大規模データ処理環境を容易に構築できる
 - PC SASではスペックが足りない, SAS ServerのCPU/メモリを占有してしまい, 他業務に支障をきたしている, といった悩みがある方には非常に有用
- SASとPythonをシームレスに使い分ける事が出来る. Rも今後対応予定.
 - データサイエンティストとして実行したい事が1つの環境下で出来る.
 - 例 1 : 1つのプログラムで、データ加工はSAS、加工したデータの解析はPython
 - 例 2 : シミュレーションはR, 機械学習系はPython, 結果の信頼性を求められる解析はSAS
- セキュリティと管理コストの低減が図れる (IT管理面の負荷軽減)
 - 中央管理による野良サーバーの乱立によるセキュリティリスクの低減
 - コンピューティングコスト・IT管理コストの最適化

Thank you!!

本セッションにて何か気になることがございましたら、遠慮なくお問い合わせください。

問合せ先：

土生 敏明 (Toshiaki Habu)

Mail: Toshiaki.Habu@sas.com

https://www.sas.com/ja_jp/software/viya.html

