



Data Science Service Accelerated Data Science Hands-on Lab.

—
2020年4月22日
日本オラクル株式会社
園田憲一

Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

Statements in this presentation relating to Oracle's future plans, expectations, beliefs, intentions and prospects are "forward-looking statements" and are subject to material risks and uncertainties. A detailed discussion of these factors and other risks that affect our business is contained in Oracle's Securities and Exchange Commission (SEC) filings, including our most recent reports on Form 10-K and Form 10-Q under the heading "Risk Factors." These filings are available on the SEC's website or on Oracle's website at <http://www.oracle.com/investor>. All information in this presentation is current as of September 2020 and Oracle undertakes no duty to update any statement in light of new information or future events.

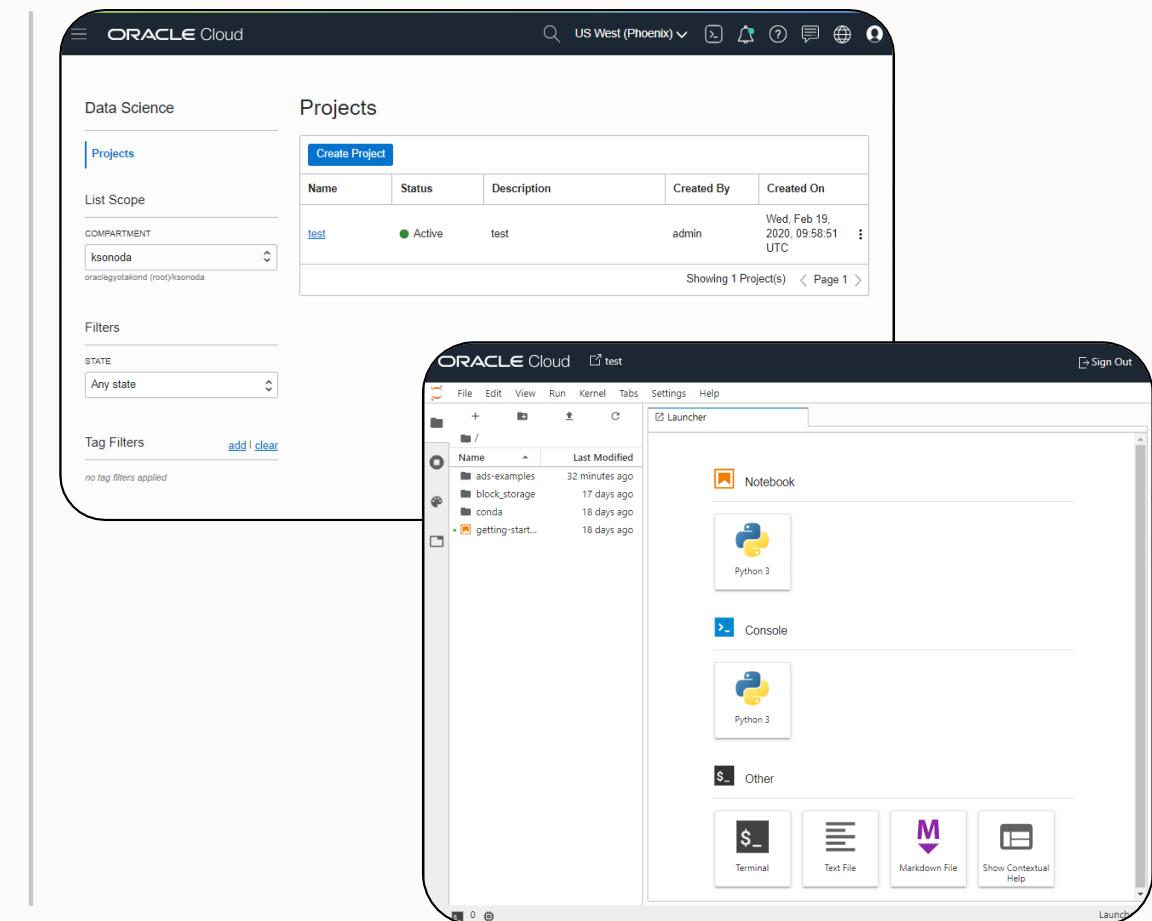


Hands-on Lab.について

- ・ 本ハンズオンは、初学者を対象に、サンプルコードを用い機械学習のワークフローと、そのコードの骨格を理解することを目的としています。
- ・ 各ワークフロー毎に机上の説明をうけ、実際にサンプルコードを実行するという流れを繰り返し、ADSの基本的なワークフロー全てを体感します。
- ・ 従って、応用的なテクニックや、より深い情報を得るためのオプション引数などを極力排除し極めてシンプルなサンプルコードを使用します。
- ・ 本ハンズオンの所要時間は机上の説明含め約2時間30分程度となります。

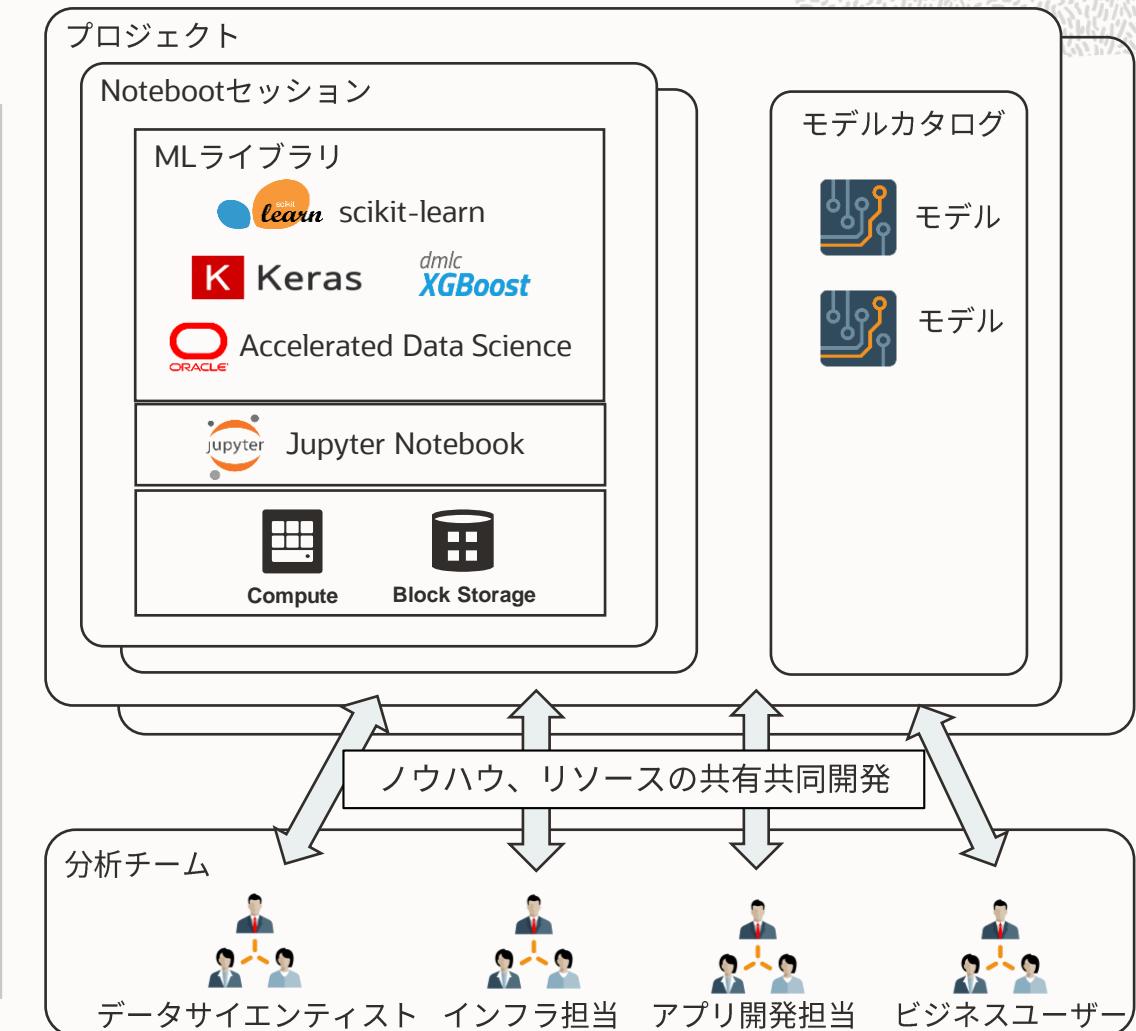
Data Science Service Overview

- 分析プロジェクトにおける共同ワークスペースを提供するサービス
 - データサイエンティストチームだけでなく、異なる部門の専門家がチームとして、分析プロジェクトを推進
 - 複数のデータサイエンティスト間のノウハウの共有
 - ソースコード
 - トレーニングデータ
 - 予測モデル
- 機械学習(ML)のOSSテクノロジー、ライブラリ、パッケージ、Oracle Accelerated Data Science(ADS)をプリインストール済み
- 他社MLライブラリでトレーニング済のモデルをインポート可能
- PaaSとしては無償、IaaSのみの課金



コンポーネント

- プロジェクト
 - 全てのリソースを保持する共同ワークスペース
- Notebookセッション
 - モデルを構築、学習するためのコーディング環境
 - Jupyter Notebook、MLライブラリ群がプリインストールされたComputeインスタンス
 - 作成時にCompartment、VCN、Subnet、Computeシェイプ、Block Volumeの容量を指定
- MLライブラリ
 - Keras
 - scikit-learn
 - XGBoost
 - Oracle Accelerated Data Science(ADS)
- モデルカタログ
 - 構築したモデルを登録、共有するストレージ領域



開発開始までの3ステップ

①プロジェクトの作成

OCIコンソール

Create Project Help Close

Projects enable you to organize your team's data science work.

COMPARTMENT i
ksonoda
oraclegytakond (root)/ksonoda

NAME i
Marketing Analysis

DESCRIPTION i
Revenue Analysis for Product A

TAGS

Tagging is a metadata system that allows you to organize and track resources within your tenancy. Tags are composed of keys and values that can be attached to resources.

[Learn more about tagging](#)

TAG NAMESPACE	TAG KEY	VALUE
None (add a free-form)		

②プロジェクト内にNotebookセッションの作成

OCIコンソール

Create Notebook Session Help Close

COMPARTMENT i
ksonoda
oraclegytakond (root)/ksonoda

NAME i
test

INSTANCE SHAPE
Standard.E2.2

LOCK STORAGE SIZE (IN GB) i
1000
The size must be between 50 GB and 1,024 GB (1 TB)

VCN IN KSONODA [\(CHANGE COMPARTMENT\)](#)
ksonoda_vcn01

SUBNET IN KSONODA [\(CHANGE COMPARTMENT\)](#)
Public Subnet Qump:PHX-AD-1

③Notebookにログイン後、Pythonでコーディング開始

Jupyter Notebook

ORACLE Cloud test

File Edit View Run Kernel Tabs Settings Help

Launcher getting-started.ipynb ads_data_visualizations.ipynb

Save this modeler in the model catalog

▪ Loading a Model from the catalog

```
[ ]: %load_ext autoreload
%autoreload 2

import pandas as pd
from sklearn.utils import Bunch
from sklearn.ensemble import RandomForestClassifier

import os

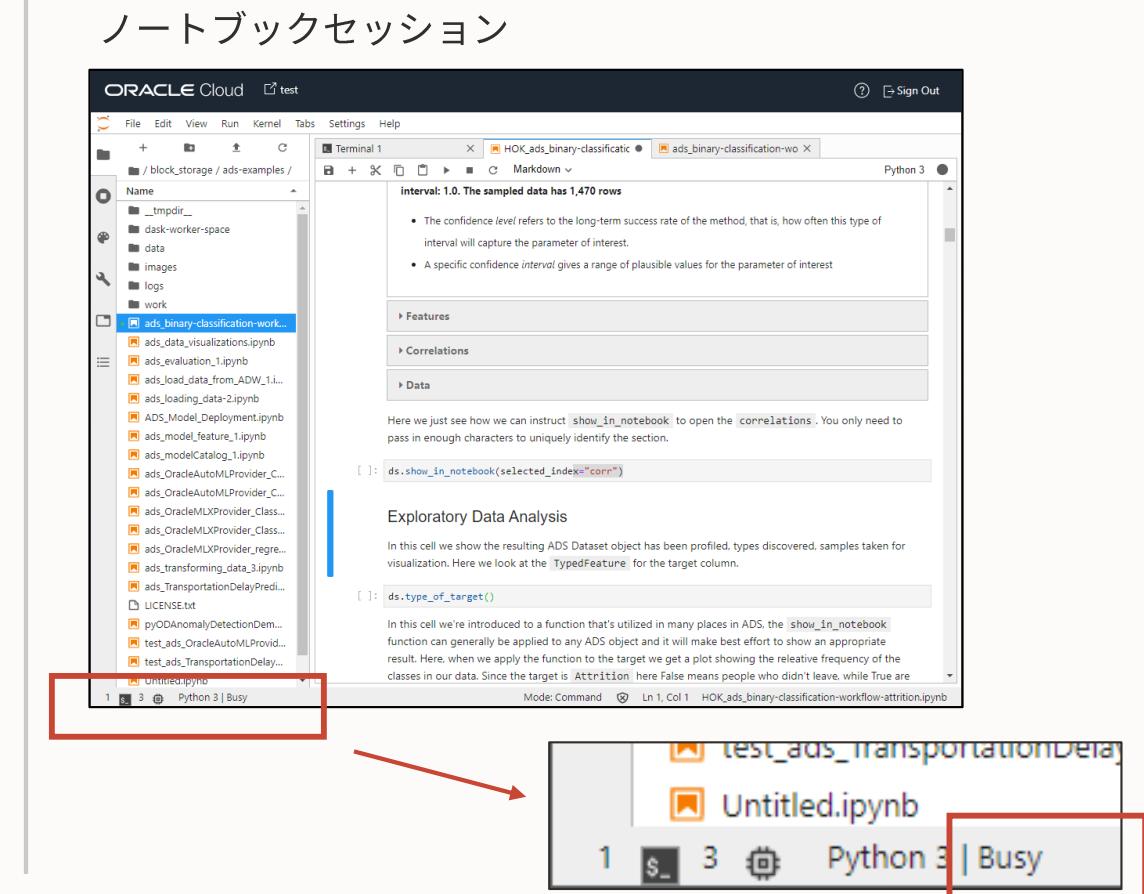
import warnings
warnings.filterwarnings('ignore')

import logging
logging.basicConfig(format='%(levelname)s:%(message)s', level=logging.ERROR)

import ads
from ads.dataset.factory import DatasetFactory
from ads.common.model import ADSModel
from ads.catalog.model import ModelsSummaryList, ModelCatalog
from ads.catalog.project import ProjectSummaryList, ProjectCatalog
from ads.catalog.summary import SummaryList
from ads.common.model_artifact import ModelArtifact
from sklearn.utils import Bunch
```

[Hands-on Lab.] ノートブックセッションの作成と注意事項

- 「OCI Console for Data Science.ppt」の手順に沿ってノートブックセッション作成までを実施します
- ノートブックでファイルを開く際に時間がかかる場合があります。
- 保存したいファイルは /home/datasience/block_storage 配下にコピーしてください。
- 処理中はステータスが Busy になります。 Idle になるまで次の処理の実行をしないでください。

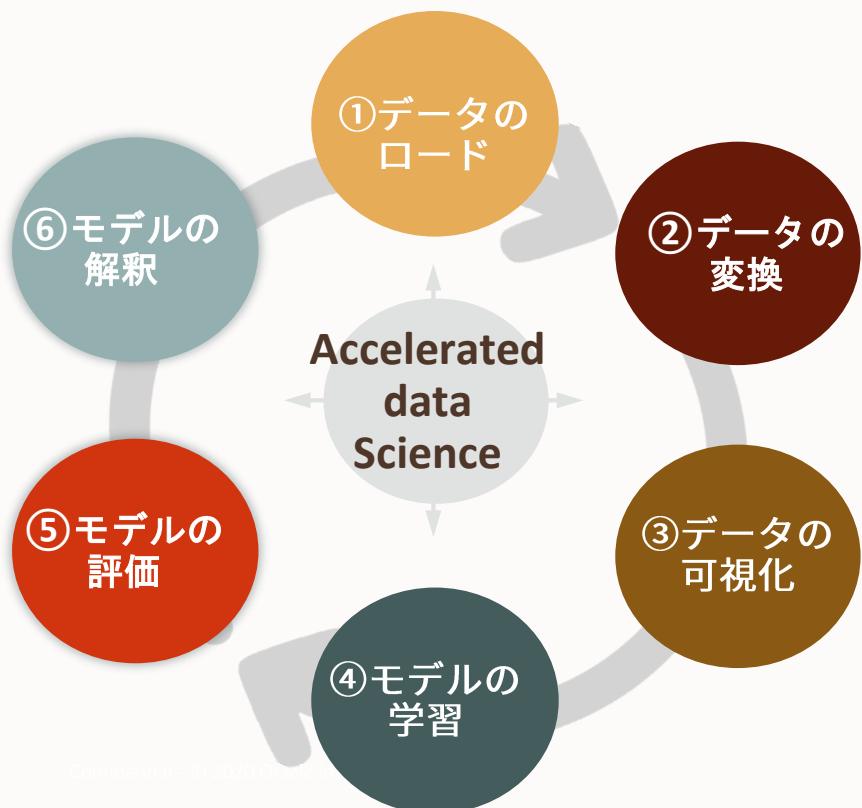


Oracle Accelerated Data Science(ADS)

- Data Science Serviceの一部として機能するPythonライブラリ
- 機械学習のライフサイクル全てのフェーズで使いやすくシンプルなAPI
- Oracle AutoML
 - ワークフロー内の処理の自動化
 - 処理時間の削減
 - モデル精度の向上



機械学習のワークフロー



データのロード

分析対象のデータをデータストアから読み込みデータセットを作成するフェーズ

- ADSによる容易な接続とデータロード
 - DatasetFactoryクラスの利用
- 対応データストア
 - ローカルファイルシステム
 - OCI Object Storage, Amazon S3, Google Cloud Storage, Azure Blob
 - Oracle DB, ADW, MongoDB, HDFS, NoSQL DB, Elastic Search, etc.
- 対応ファイルフォーマット
 - CSV, TSV, Parquet, libsvm, json, Excel, HDF5, SQL, xml, Apache Server Logfile(clf, log), arff

データロード サンプルコード

ローカルファイルシステムからのロード

```
ds = DatasetFactory.open("/path/to/data.data", format='csv', delimiter=" ")
```

OCI Object Storage Serviceからのロード

```
ds = DatasetFactory.open("oci://<bucket-name>/<file-name>", storage_options = {  
    "config": "~/.oci/config",  
    "profile": "DEFAULT_USER"  
})
```

Amazon S3からのロード

```
ds = DatasetFactory.open("s3://bucket_name/iris.csv", storage_options = {  
    'key': 'aws key',  
    'secret': 'aws secret',  
    'blocksize': 1000000,  
    'client_kwargs': {  
        "endpoint_url": "https://s3-us-west-1.amazonaws.com"  
    }  
})
```

ADWからのロード

```
uri = f'oracle+cx_oracle://{{os.environ["ADW_USER"]}}:{{os.environ["ADW_PASSWORD"]}}@{{os.environ["ADW_SID"]}}'  
ds = DatasetFactory.open(uri, format="sql", table=table, index_col=index_col, target='label')
```



[Hands-on Lab. Task 1]

Loading Data (10 min.)

データの可視化

データからより多くの洞察を直感的に得るために、データセットを様々なチャートで可視化するフェーズ

- データ列のタイプを自動的に検出し、データをプロットする最適な方法を提供するスマートな可視化ツールを提供
- 可視化の種類
 - 組み込み関数によるデータセットのサマリ、その他チャート
 - 可視化API(Seaborn, Matplotlib, GIS)を使ったコーディングによる可視化

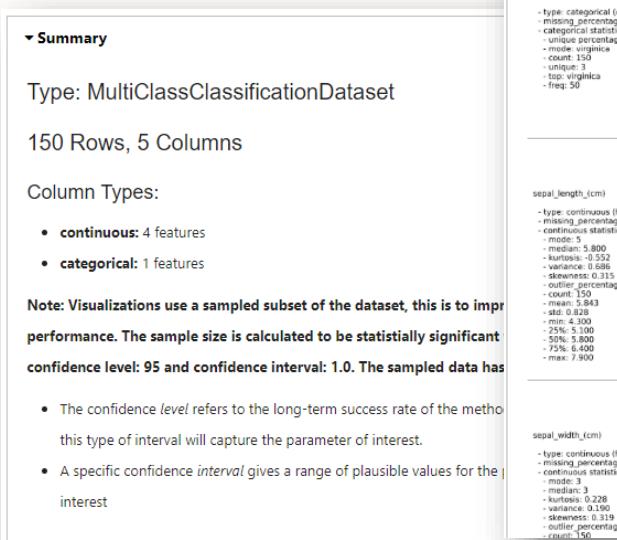
データの可視化

データセットのサマリ

データセットに関する基本情報の包括的なプレビューを可視化

```
#データセットをshow_in_notebook()関数に渡す  
ds.show_in_notebook()
```

データセットに関する
列数、行数、特徴タイプ



データセットの特徴量、値の分布



データセットのWARNING

▼ Warnings (6)

6 WARNING(S) found

`name` has a high cardinality: 1453 distinct values

high-cardinality

zeros

`NumCompaniesWorked` has 197 (13.40%) zeros

zeros

`StockOptionLevel` has 631 (42.93%) zeros

zeros

`YearsAtCurrentLevel` has 244 (16.60%) zeros

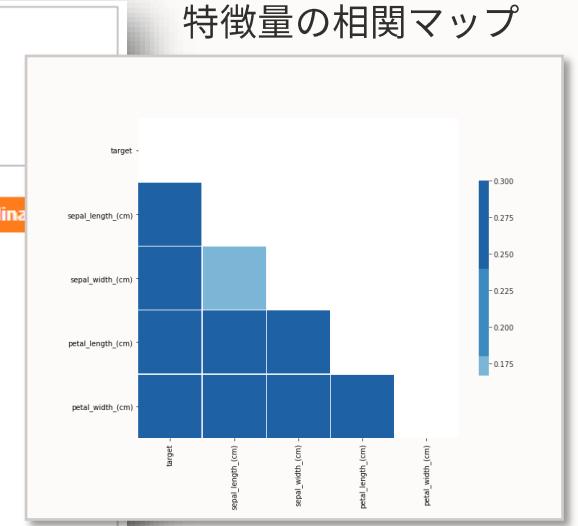
zeros

`YearsSinceLastPromotion` has 581 (39.52%) zeros

zeros

`YearsWithCurrManager` has 263 (17.89%) zeros

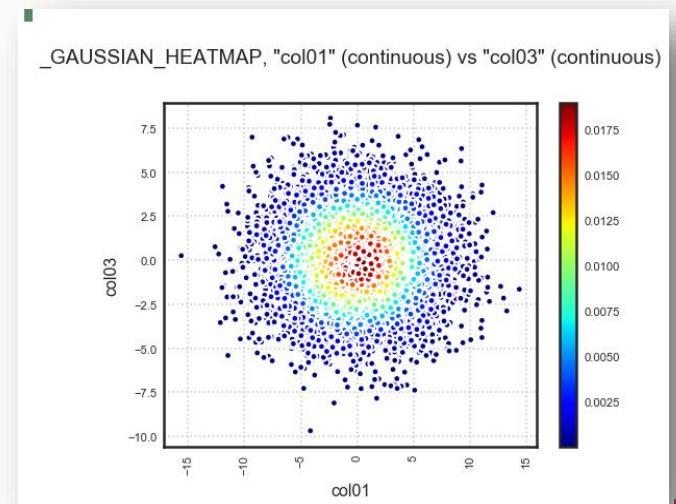
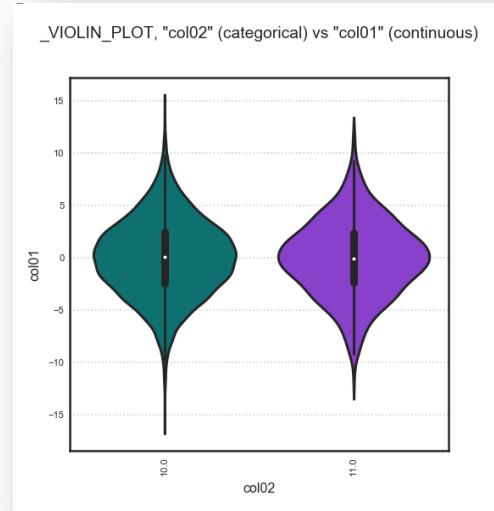
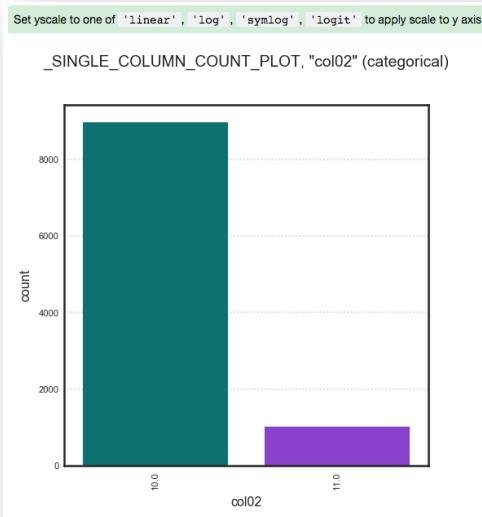
特徴量の相関マップ



データの可視化 組み込み関数によるチャート

カウントプロット、バイオリン曲線、ガウスヒートマップ

```
# カウントプロット  
ds.plot("col02").show_in_notebook(figsize=(4,4))  
  
# バイオリンプロット  
ds.plot("col02", y="col01").show_in_notebook(figsize=(4,4))  
  
# ガウスヒートマップ  
ds.plot("col01", y="col03").show_in_notebook()
```



データ可視化

可視化APIを使ったコーディングによる可視化

Seaborn, Matplotlib, GISなど

```
# Matplotlibでのパイチャートのコーディング例
```

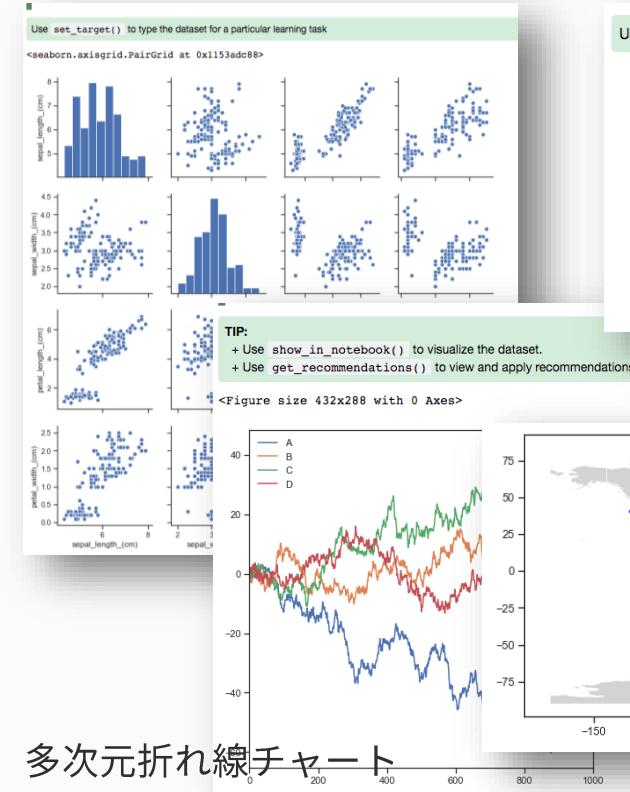
```
from numpy.random import randn

df = pd.DataFrame(randn(1000, 4), columns=list('ABCD'))

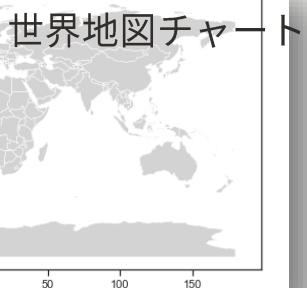
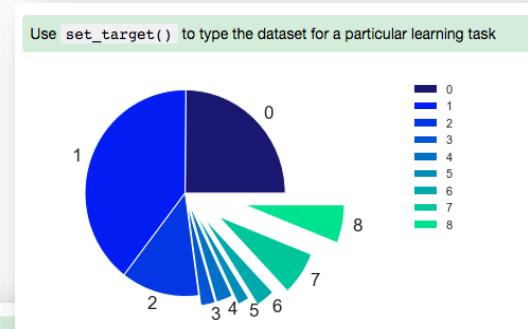
def ts_plot(df, figsize):
    ts = pd.Series(randn(1000), index=pd.date_range('1/1/2000',
    periods=1000))
    df.set_index(ts)
    df = df.cumsum()
    plt.figure()
    df.plot(figsize=figsize)
    plt.legend(loc='best')

ds = DatasetFactory.from_dataframe(df, target='A')
ds.call(ts_plot, figsize=(7,7))
```

散布図



パイチャート



多次元折れ線チャート

[Hands-on Lab. Task 2]

Data Visualization

データ変換

データセットを機械学習に使える形に成形するフェーズ

- 生データをRDBのような「完全に正規化された表形式のデータ」に成形
- 機械学習に特有のデータ成形(特徴量エンジニアリング)

特徴量エンジニアリングの一般的な処理例

- テキストデータの単語分割、頻度カウント
- 文字列の”エンコード”
- 低頻度語や異常値の除外
- 欠損値データの処理
- モデル生成に関係しないデータの削除
- モデル生成に含めるべきでないデータの削除
- etc.



データ変換 行、列、データセットの操作

- 行の操作
 - 削除、追加、フィルタ、重複行の削除
- 列の操作
 - 削除、追加、フィルタ、列名の変更、データ型の変換、ノーマライズ
- String列に対する操作
 - クラスラベルのカウント、ラベルの長さの変更(文字数の変更)、大文字・小文字の変換
- データセットの操作
 - Nullのある列の検出、Nullの変更、結合

データ変換 ADSによる特徴量エンジニアリングの自動化

ADSのAutoMLライブラリにより、一部特徴量エンジニアリングの自動化が可能

1. 定数、主キー列の削除
2. 欠損値の補完、削除
3. 予測精度を下げる可能性のある特徴量の削除
4. データセットの均衡化

```
# 推奨の自動処理内容を確認しながらデータセットに反映するケース
```

```
ds.get_recommendations()  
recommendations_ds = ds.get_transformed_dataset()
```

```
# 推奨の自動処理内容を確認せずにデータセットに反映するケース
```

```
transformed_ds = ds.auto_transform()
```



データ変換 ADSによる特徴量エンジニアリングの自動化

get_recommendations()による定数、主キー列の検出、対応策の実行

```
ds.get_recommendations()
```

Column 'Directs' is constant and will be dropped
Column 'Over18' is constant and will be dropped
Column 'WeeklyWorkedHours' is constant and will be dropped

Potential Primary Key Columns

EmployeeNumber(type: int64) Contains mostly unique values(100.00%)	Drop
Next	Reset All
Drop	
Do nothing	

定数

主キー列

変換前のデータセット(行数, 列数)

```
ds.shape
```

(1469, 36)

“Drop”を選択

変換後のデータセット(行数, 列数)

```
ds_trans.shape
```

(1469, 32)

データ変換 欠損値の補完、削除

get_recommendations()による欠損値の検出、対応策の実行

```
ds.get_recommendations()  
  
Column 'Directs' is constant and will be dropped  
Column 'Over18' is constant and will be dropped  
Column 'WeeklyWorkedHours' is constant and will be dropped  
  
Potential Primary Key Columns  
  
EmployeeNumber(type: int64) Contains mostly unique values(100.00%) Drop  
  
Imputation  
  
Gender(type: category) Contains missing values(1) Drop  
RelationshipSatisfaction(type: float64) Contains missing values(1) Drop  
  
Next Reset All
```

欠損値のある列

変換前のデータセット(行数, 列数)

```
ds_trans.shape  
(1469, 32)
```

“Drop”を選択

変換後のデータセット(行数, 列数)

```
ds_trans.shape  
(1469, 30)
```



データ変換 予測精度を下げる可能性のある特徴量の削除

get_recommendations()による多重共線性の検出、対処策の実行

```
ds.get_recommendations()
Column 'Directs' is constant and will be dropped
Column 'Over18' is constant and will be dropped
Column 'WeeklyWorkedHours' is constant and will be dropped
```

Potential Primary Key Columns

EmployeeNumber(type: int64) Contains mostly unique values(100.00%)	Drop
--	------

Imputation

Gender(type: category)	Contains missing values(1)	Drop
RelationshipSatisfaction(type: float64)	Contains missing values(1)	Drop

Multicollinear Columns

PercentSalaryHike(type: int64) Strongly correlated with PerformanceRating(99.56%).	Drop PercentSalaryHike
--	------------------------

Next Reset All

- Drop PercentSalaryHike
- Drop PerformanceRating
- Combine with PerformanceRating
- Do nothing

欠損値
ある列

変換前のデータセット(行数, 列数)

ds_trans.shape

(1469, 30)

“Drop”を選択

変換後のデータセット(行数, 列数)

ds_trans.shape

(1469, 29)



データ変換

データセットの均衡化(アップサンプリング、ダウンサンプリング)

get_recommendations()によるデータ不均衡の検出、対処策の実行

```
ds.get_recommendations()  
Column 'Directs' is constant and will be dropped  
Column 'Over18' is constant and will be dropped  
Column 'WeeklyWorkedHours' is constant and will be dropped  
  
Potential Primary Key Columns  
  
EmployeeNumber(type: int64) Contains mostly unique values(100.00%) Drop  
  
Imputation  
  
Gender(type: category) Contains missing values(1) Drop  
RelationshipSatisfaction(type: float64) Contains missing values(1) Drop  
  
Multicollinear Columns  
  
PercentSalaryHike(type: int64) Strongly correlated with PerformanceRating(99.56%) Drop PercentSalaryHike  
  
Fix imbalance in dataset  
  
Attrition(type: bool) Up-sample  
Apply Up-sample  
Down-sample  
Do nothing
```

データに偏りのある列

変換前のデータセット(行数, 列数)

ds_trans.shape
(1469, 29)

“Up-sample”を選択

“Down-sample”を選択

変換後のデータセット(行数, 列数)

ds_trans.shape
(2464, 29)

ds_trans.shape
(474, 29)



[Hands-on Lab. Task 3]

Data Transformations

モデルの学習

成形済みのデータセットを統計処理にかけ、予測モデルを構築するフェーズ

- ADSのAutoMLにより合理的な時間内で精度の高いモデルを生成
- データサイエンスのノウハウが必要かつ、時間のかかる作業を自動化
 - 最適なアルゴリズム選択の自動化
 - サンプリングの自動化(アダプティブサンプリング)
 - 最適な特徴量選択の自動化
 - ハイパーパラメータ・チューニングの自動化

```
# 成形済みデータセットをトレーニングデータとテストデータに分割  
train, test = transformed_ds.train_test_split(test_size=0.1)
```

```
# トレーニングデータを学習しモデルを構築  
ml_engine = OracleAutoMLProvider()  
automl = AutoML(train, provider=ml_engine)  
model, baseline = oracle_automl.train()
```

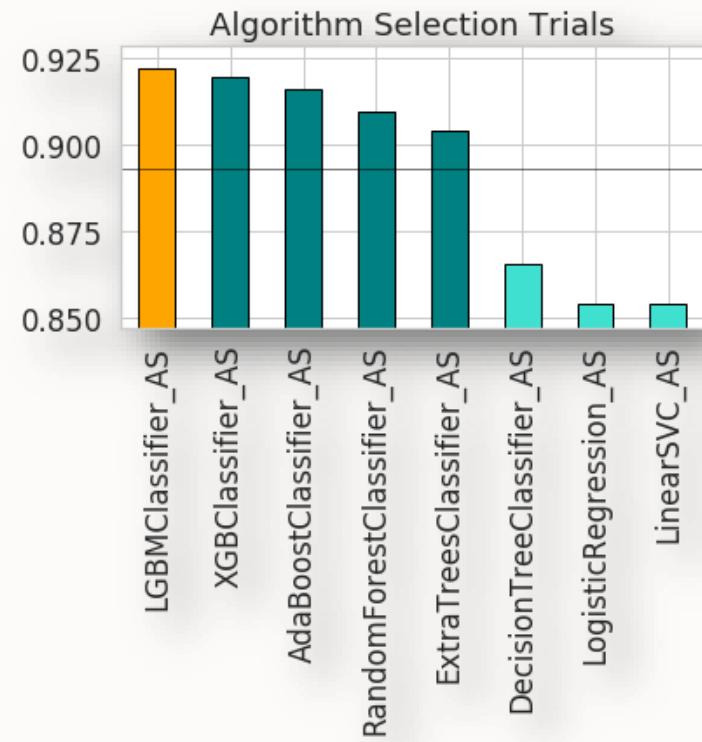
- アルゴリズムのリスト
- AdaBoostClassifier
 - DecisionTreeClassifier
 - ExtraTreesClassifier
 - KNeighborsClassifier
 - LGBMClassifier
 - LinearSVC
 - LogisticRegression
 - RandomForestClassifier
 - SVC
 - XGBClassifier



モデルの学習 Oracle AutoMLによる自動化

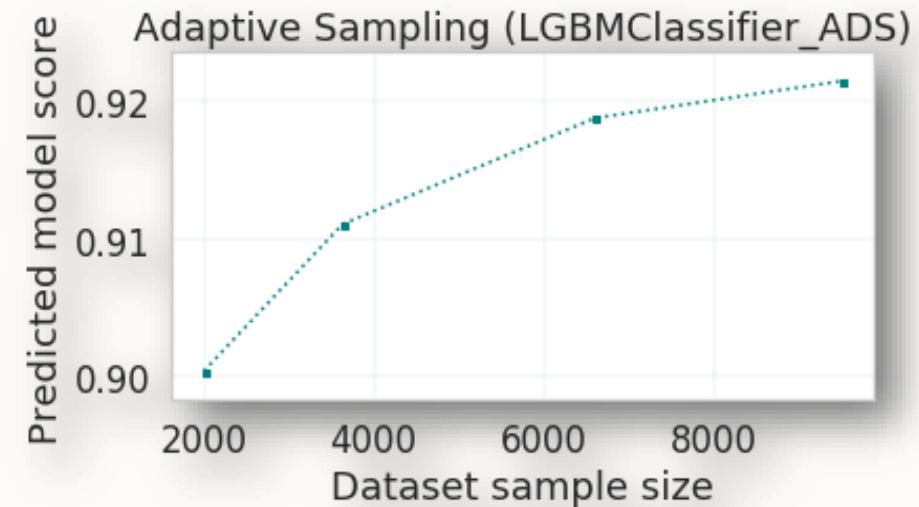
①最適なアルゴリズム選択の自動化

```
oracle_automl.visualize_algorithm_selection_trials()
```



②サンプリングの自動化

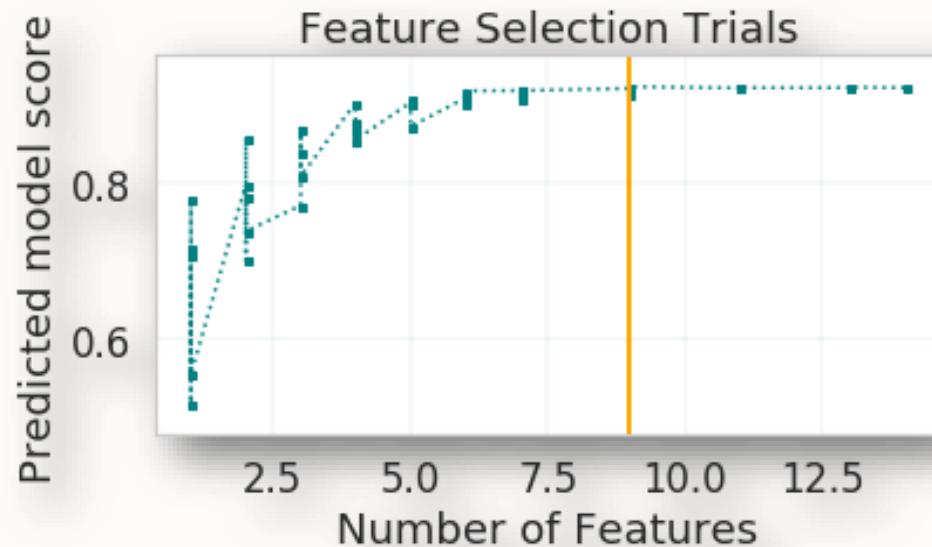
```
oracle_automl.visualize_adaptive_sampling_trials()
```



モデルの学習 Oracle AutoMLによる自動化

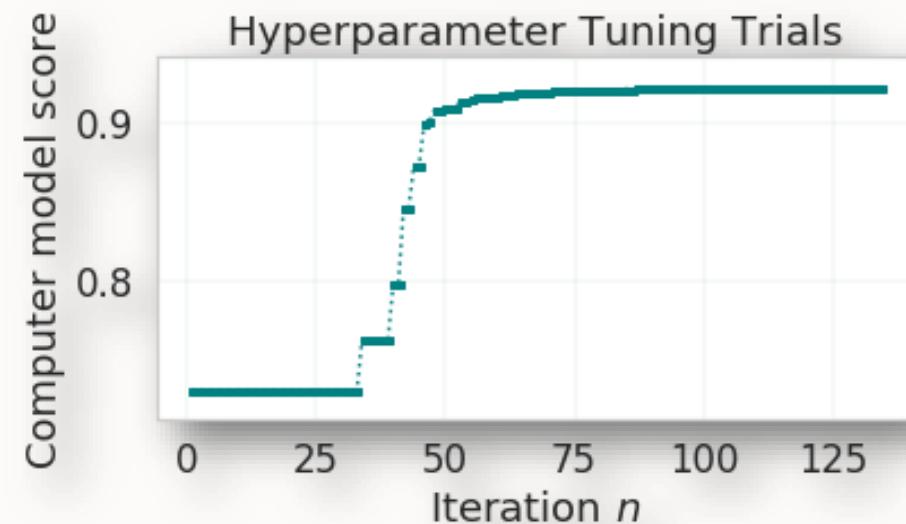
③最適な特徴量選択の自動化

```
oracle_automl.visualize_feature_selection_trials()
```



④ハイパーパラメータ・チューニングの自動化

```
oracle_automl.visualize_tuning_trials()
```

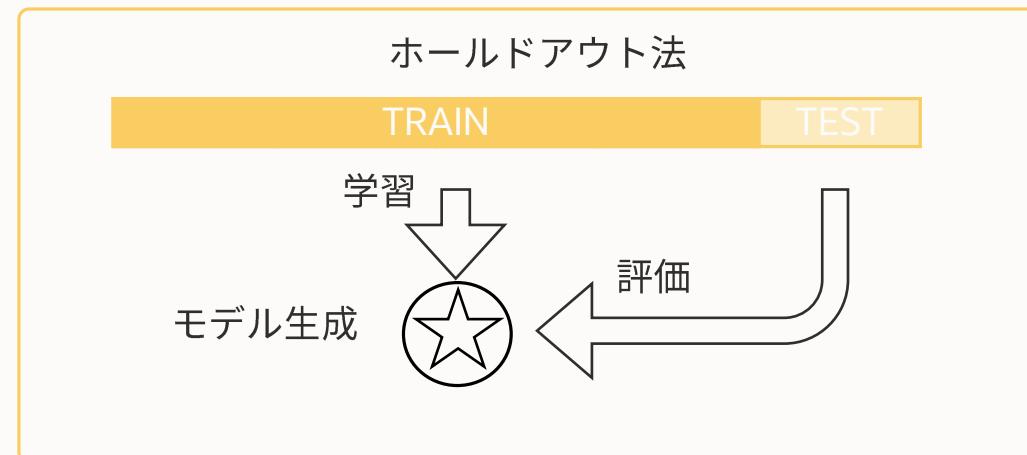


[Hands-on Lab. Task 4] Training Models

モデルの評価

「このモデルは本当に正しいのか？」、「このモデルはどれだけ優れているのか？」を知るフェーズ

- テストデータを用いた予測結果を解釈可能な標準化された一連のスコアとチャートに変換する
- 評価対象：「バイナリ分類」、「マルチクラス分類」、「回帰」
- 評価手法：ホールドアウト法、クロスバリデーション(汎化性が高い)



※1：データをN個に分けて、1回目はそのうち1つをTEST用、残りのN-1個をTRAIN用として使用。2回目は別の1つをTEST用、残りのN-1個をTRAIN用として使用・・・をN回繰り返し、各回で測定した精度の平均を取る

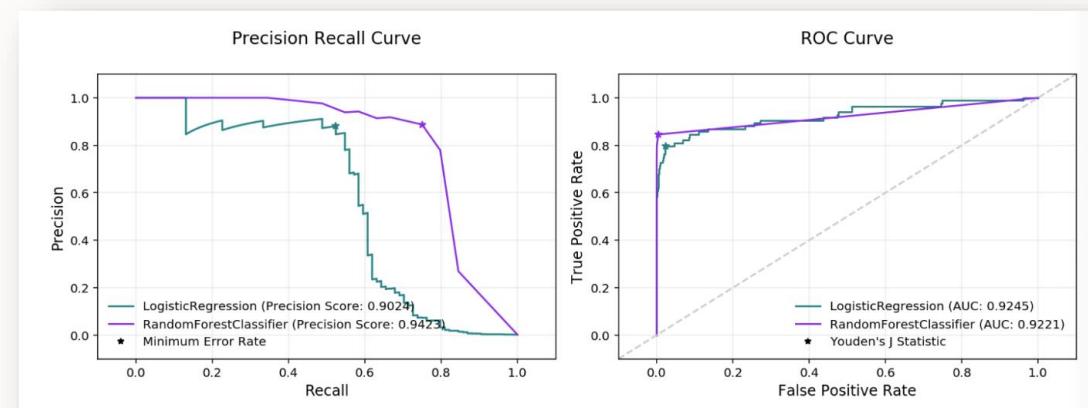
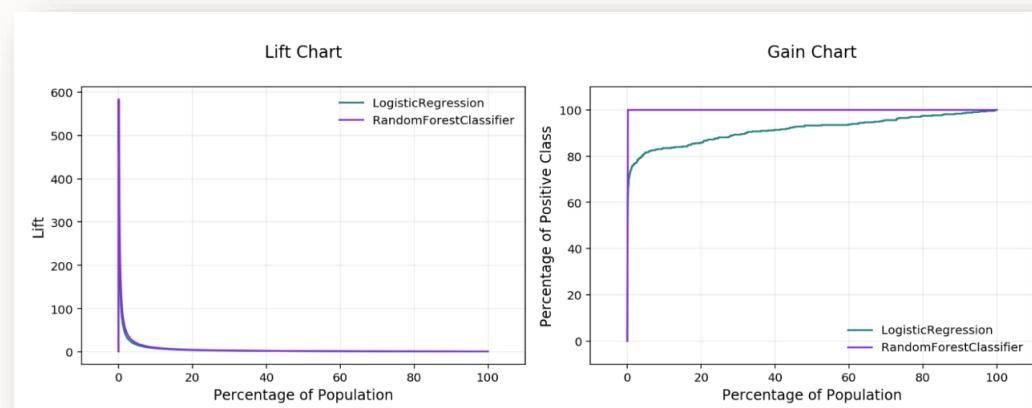
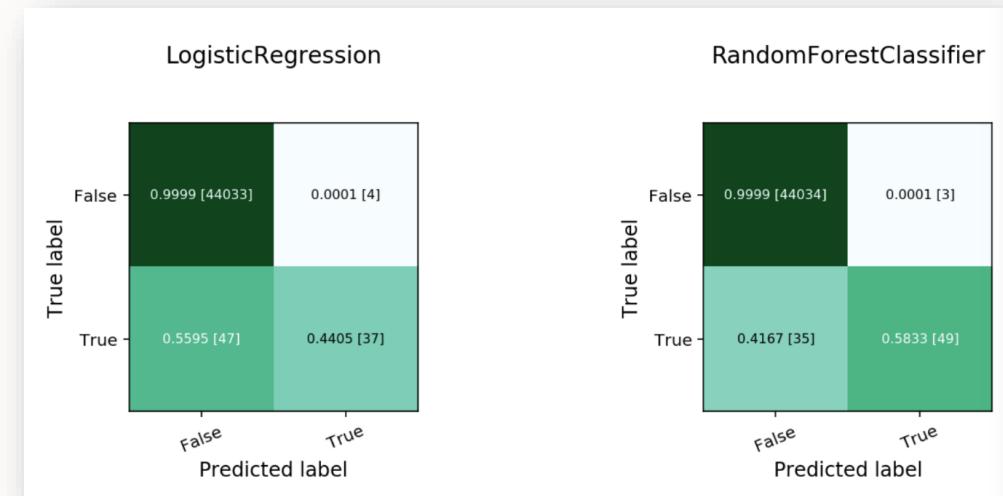
モデルの評価

評価結果のチャート

例)バイナリ分類

- リフトチャート、ゲインチャート
- PR曲線、ROC曲線
- 混合行列

```
# 評価の実行  
bin_evaluator = ADSEvaluator(test, models=[bin_lr_model, bin_rf_model],  
training_data=train)  
  
# 評価結果のチャート化  
bin_evaluator.show_in_notebook(perfect=True)
```



[Hands-on Lab. Task 5] Model Evaluation

モデルの解釈

「どのようなモデルが構築されたのか？」、「なぜそのような予測結果になったのか？」を
知るフェーズ

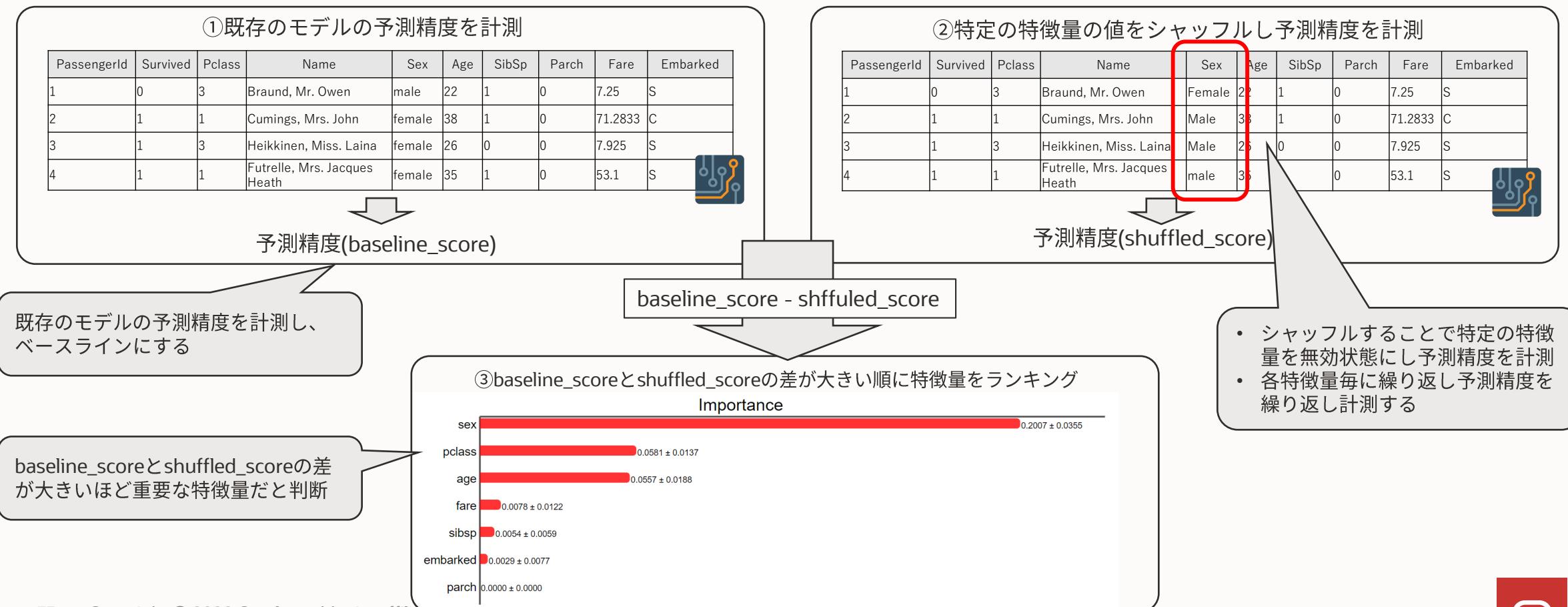
- モデルの解釈とは？
 - モデルや予測結果を人間が解釈可能な状態にチャート化すること
- なぜモデルの解釈が必要なのか？
 - 予測結果のブラックボックス化を防止
 - 開発者がモデルの挙動を理解し、モデルの品質改善やデバッグをより容易にする
- 解釈の種類
 - Global Explainer = モデル自体の解釈
 - 特徴量重要度(Feature Permutation Importance)
 - 個別条件付き期待値(Individual Conditional Expectation(ICE))
 - 部分従属プロット(Partial Dependence Plot(PDP))
 - Local Explainer = 特定のデータに対してモデルが予測した結果の解釈



モデルの解釈

ADS Global Explainer – Feature Permutation Importance

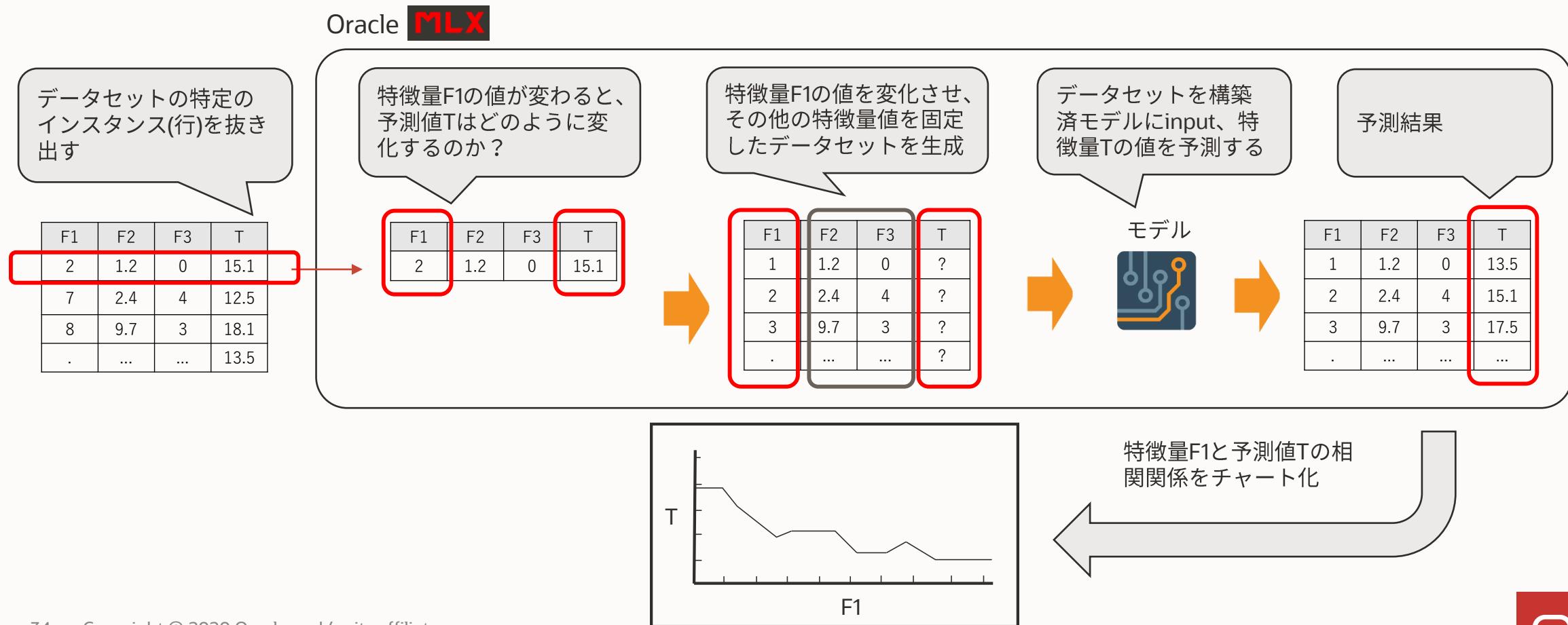
構築済モデルにおいて「どの特徴量が予測モデルに強く影響し、どの特徴量は影響しないのか？」を特定する



モデルの解釈

ADS Global Explainer - Individual Conditional Expectation(ICE/PDP)

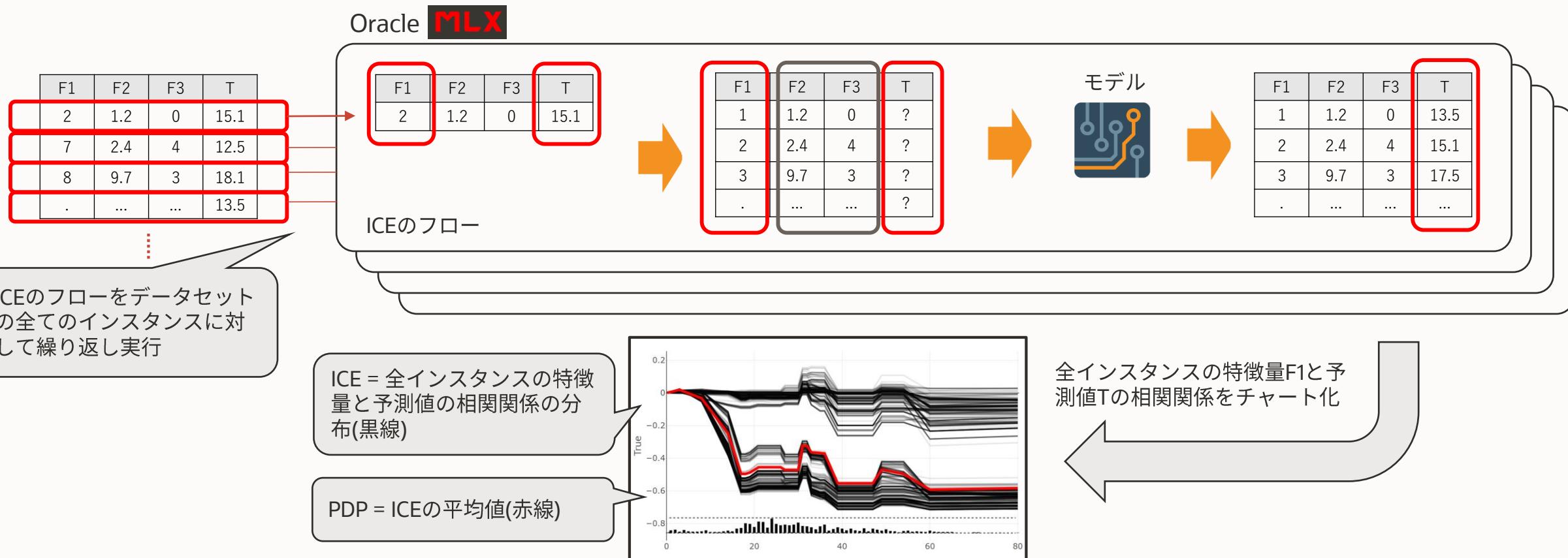
特徴量の取りえる値によって予測値がどのように変化するのかを把握する



モデルの解釈

ADS Global Explainer - Partial Dependence Plot(ICE/PDP)

全インスタンスにおける、特徴量と予測値の相関関係の分布(ICE)と平均値(PDP)を把握する



[Hands-on Lab. Task 6]

Model Explainability (Global)

モデルの解釈 Local Explainer

なぜそのような予測結果になったかを解釈する

例) タイタニックのデータセット(<https://www.kaggle.com/c/titanic>)

- データセット
 - 乗客データ(乗客のデモグラフィックデータ+α)
 - 生存率(Survived= 0 or 1)を予測するモデルを構築
- 任意の乗客の生存率予測結果がなぜそのようになったのかを解釈

学習データセット

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Fare	Embarked
1	0	3	Braund, Mr. Owen	male	22	1	0	7.25	S
2	1	1	Cumings, Mrs. John	female	38	1	0	71.2833	C
3	1	3	Heikkinen, Miss. Laina	female	26	0	0	7.925	S
...



予測したいデータセット

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Fare	Embarked
500	?	-	Anna. Miss. Bworn	female	36	1	0	71.283	C

予測結果

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Fare	Embarked
500	1	-	Anna. Miss. Bworn	female	36	1	0	71.283	C

Why?

モデルの解釈 Local Explainer

Oracle MLXによる予測結果の説明チャートの生成

データセット

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Fare	Embarked
1	0	3	Braund, Mr. Owen	male	22	1	0	7.25	S
2	1	1	Cumings, Mr. John	male	38	1	0	71.2833	C
3	1	3	Heikkinen, Mrs. Laina	female	26	0	0	7.925	S
...



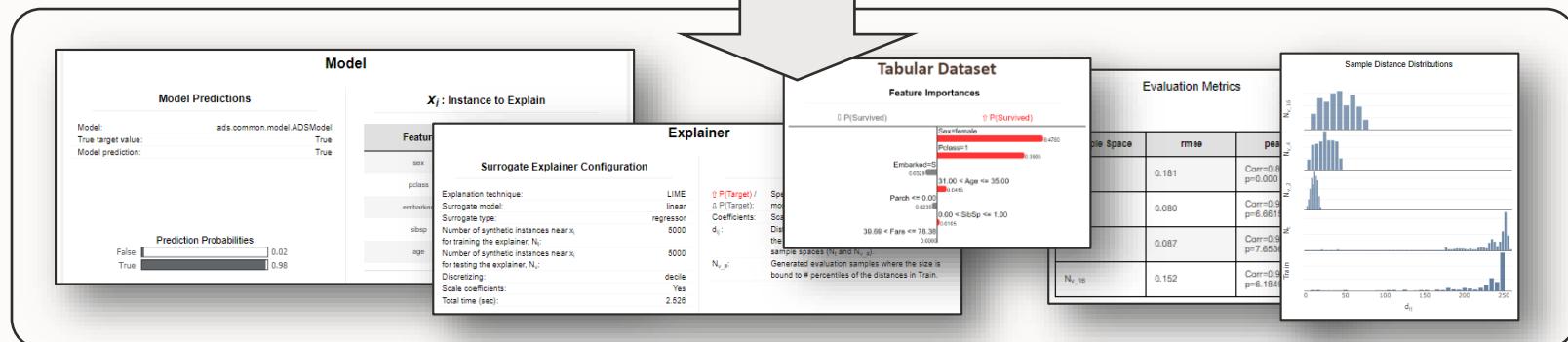
予測したいデータセット

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Fare	Embarked
500	?		Anna. Miss. Bworn	female	36	1	0	71.283	C

Passenger ID = 500 の予測結果と、予測結果の根拠を知りたい



「Passenger ID = 500 の予測結果」に関する解釈のチャート化



モデルの解釈 Local Explainer

PassengerID 500番の乗客の生存率の予測結果

PassengerID 500番の乗客の特徴量

予測結果に影響した
「特徴量」
「特徴量の重要度」
「特徴量の値(条件)」



[Hands-on Lab. Task 6] Model Explainability (Local)

モデルカタログ

- 構築したモデルを登録、共有するための仕組み
- データサイエンティスト間のモデルの共有とモデルの再現性を促進
- モデルカタログからモデルをダウンロードし、Data Science Service以外の環境でモデルを実行
- ADS以外のMLライブラリで作成したモデルを登録可能
 - scikit-learn, keras, xgboost, lightGBMをサポート



scikit-learn



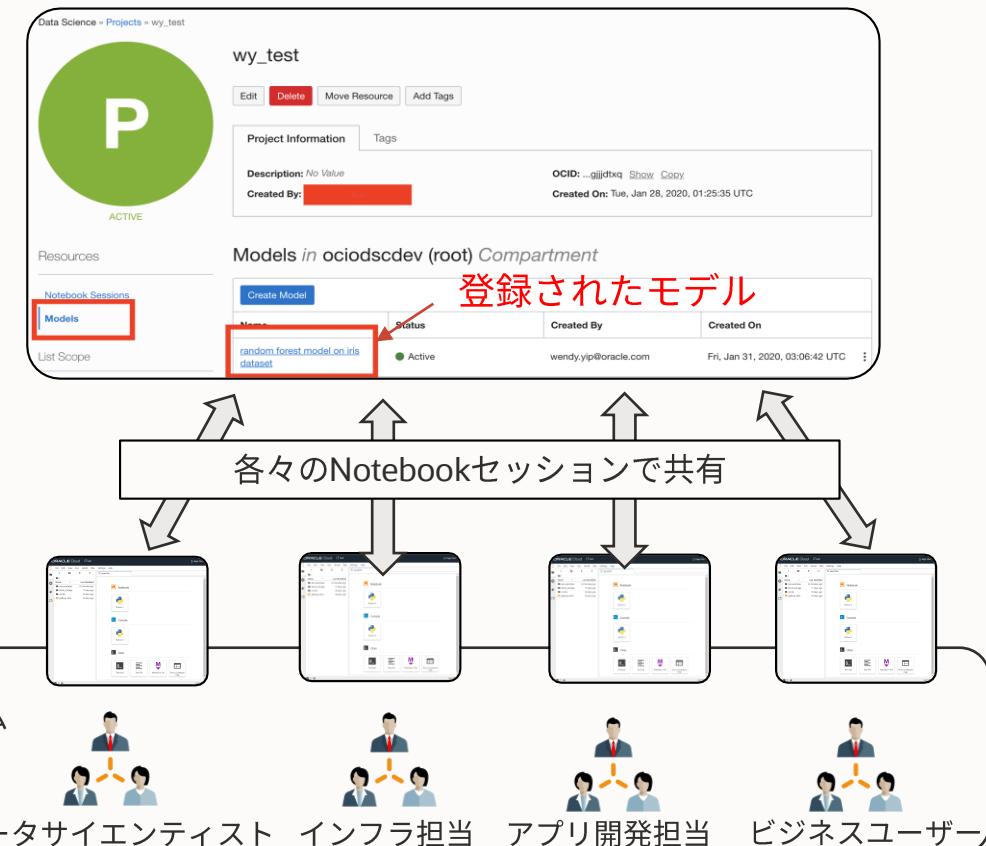
Keras



XGBoost

lightGBM

OCIコンソール ([プロジェクト]> [モデル])



[Hands-on Lab.]

Task 7 : Run prediction and Save your model to your Catalog

A person is seen from the side, wearing a hooded garment with prominent black and white zebra stripes. The hood covers their head, and the pattern continues down the front of the garment.

ORACLE