

プロローグ

テーマ決めから前処理、モデリング、
デプロイまで、まずは体験

ご購入はこちら

Pythonをすぐに試せる Google Colabでデータ分析入門

切通 恵介

```

(2) | from google.colab import drive
    | drive.mount('/content/drive')
    |
    | Mounted at /content/drive
    |
(3) | import pandas as pd # Pandasを読み込んでrdという名前をつける
    |
    | df = pd.read_csv('/drive/My_Drive/data/amerghdata_complete.csv', # このファイルをPandas.DataFrame方式で
    |                 index_col=0, # indexとして認識する
    |                 name_dates=True, # 時刻ごとのデータの場合、時刻型として読み込む
    |                 dayfirst=True) # 日:月:年の順番であることを認識させる
    |
(6) | import matplotlib.pyplot as plt
    |
    | df.plot(figsize=(30,30), subplots=True) # 画像サイズ30x30で全てのデータを別々の枠に可視化する
    |     plt.show() # 可視化結果を表示する
    |
    | --INSERT--
    |
    |
  
```

図1 本稿でやること…センサー・データとGoogle Colaboratoryでデータ分析を体験する

時系列センサー・データを利用したデータ分析プロジェクトを例として、データサイエンス(ここではデータ分析)を体験します。データ分析の流れとして、テーマ決め、可視化、前処理、モデリング、デプロイがあります。特に前処理、モデリングの行程においては機械学習アルゴリズムを利用しつつ進めていきます。分析を行うPythonのプログラムはGoogle Colaboratoryでも体験できます(図1)。

データ分析の全体像

図2に代表的なデータ分析の全体像を示します。これはCRISP-DM⁽¹⁾という、データ分析で現実問題を解決するためのフレームワークです。このフレームワークでは、データ理解、前処理、モデリング、モデルの評価までを何度も繰り返し、最終的にモデルを継続的に利用するモデルのデプロイまでを行います。以下に分析の6段階を記し解説していきます。

- ①分析テーマを決める(図2中の「ビジネスの理解」に当たる)
- ②データの理解
- ③データの可視化
- ④前処理

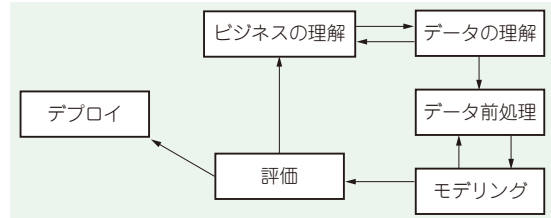


図2(1) 代表的なデータ分析の全体像

データ分析で現実問題を解決するためのフレームワーク「CRISP-DM」のフロー

- ⑤モデリングと評価
- ⑥モデルのデプロイ

この中で何より重要なのは、分析すべきテーマを決めることです。特にビジネスではここを間違えると、実際の業務に生かせなかったり、コストだけが上がり成果が得られなかったりと、悲しい事態を招きます。AIを使えば何とかなる、というような考え方をしていると大体テーマ選定のところでつまづくことになるので、注意が必要です。

①分析テーマを決める

テーマを選択するとき、何を基準にしたらよいでしょうか。趣味の場合はある程度興味に沿った内容でよいのですが、もしビジネスならば、有効なテーマを検討することは非常に重要です。というのも、データ分析やAIの活用がいわゆるPoC(実証実験)で止まってしまうことの多くはテーマの選定が不適切であるためです。

●適切なテーマの決め方

適切なテーマとはどのようなものかを考えます。データ分析のテーマは、課題が解決されたときの価値と課題解決の実現性と言う2つの軸を使って考えることをお勧めします(図3)。

▶課題が解決されたときの価値を考える

例えば、よくあるテーマとして「AIで装置の異常検知をしたい」と言うものがあります。この分析テーマに対して価値を算定する場合、異常が起きた場合にど