pd-1. データサイエンス，散布図，平均，分布

（Python によるデータサイエンス演習）

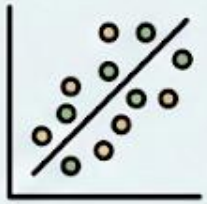
URL: https://www.kkaneko.jp/ai/pd/index.html

金子邦彦

CC BY NC SA

データサイエンス入門：学習内容の全体像とゴール

散布図



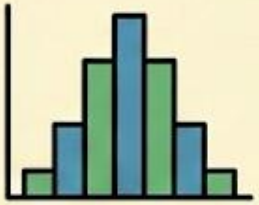
2つの量の関係を
視覚的に把握

平均

データ集合の
代表値を算出

データから
正しい知見を導く

ヒストグラム



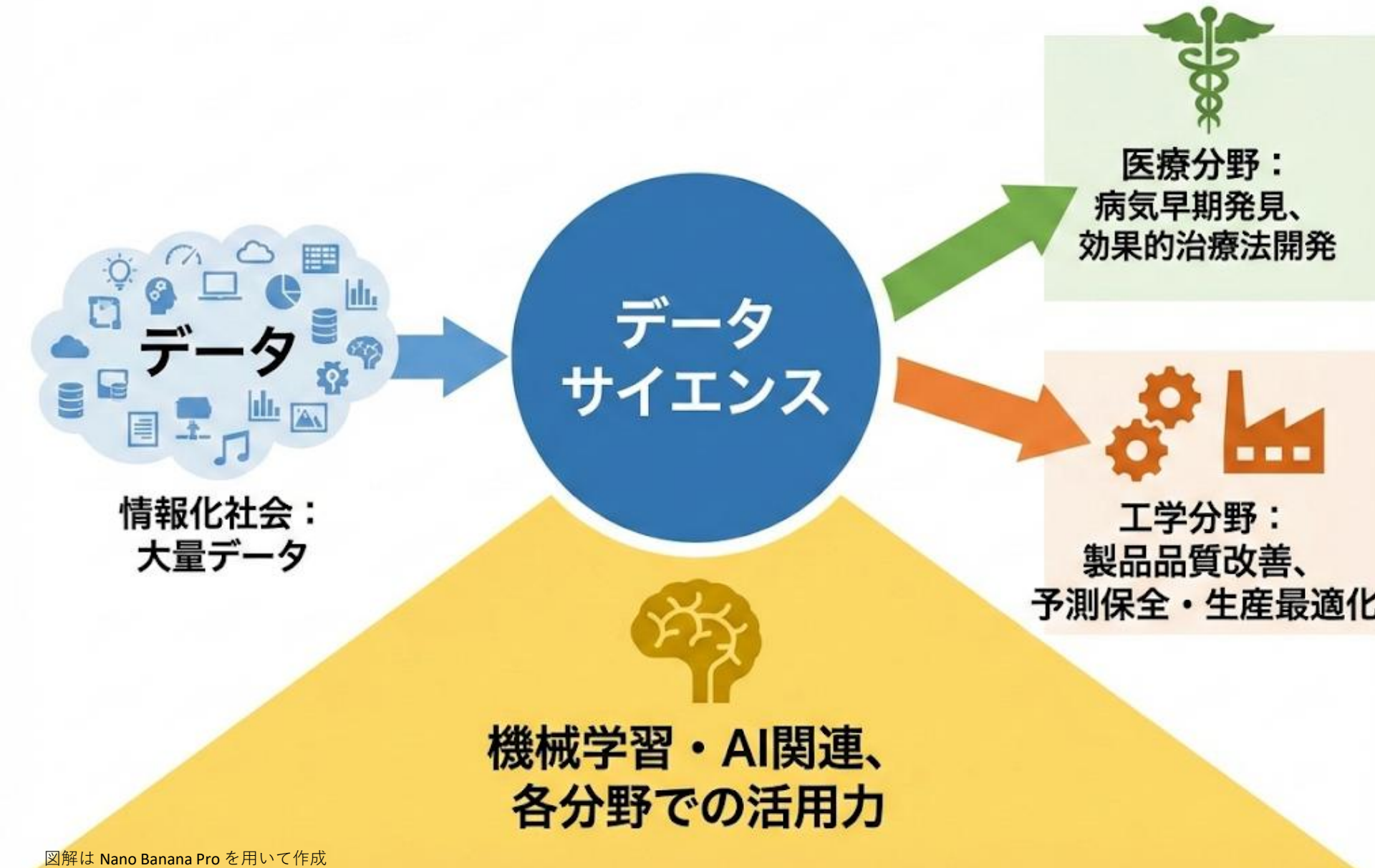
データの分布と
密度を把握

意義

研究レポート作成や
データ活用の実力向上

図解は Nano Banana Pro を用いて作成

データサイエンス：データから知見・結論を導く



図解は Nano Banana Pro を用いて作成

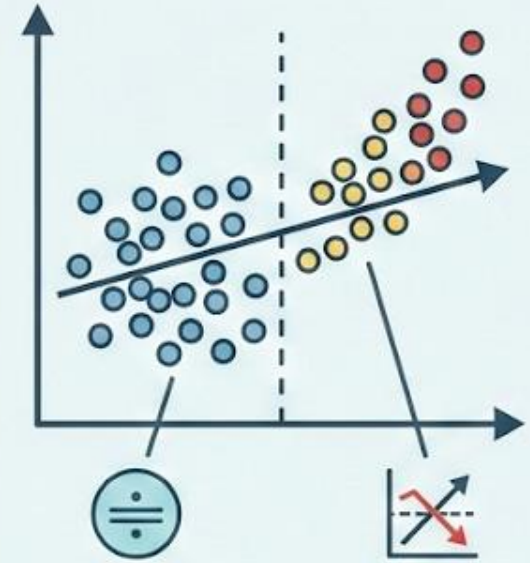
研究レポートの6つの要素

データサイエンスのスキルは研究レポートの作成にも有用である。

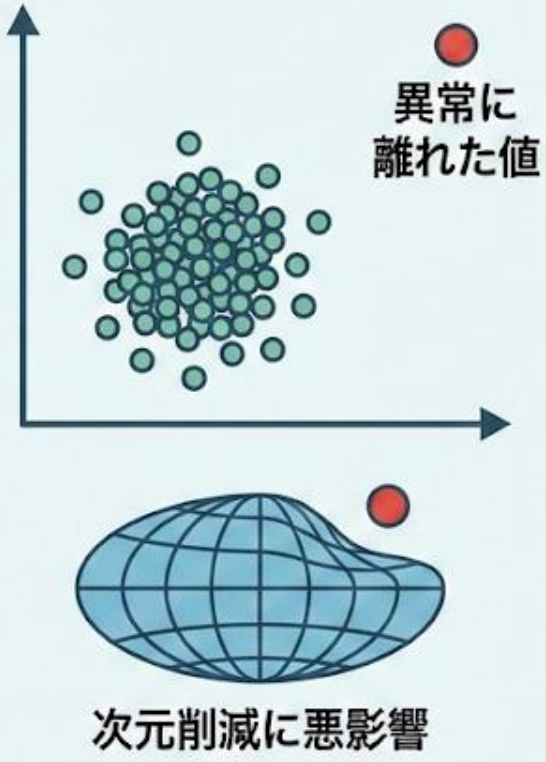
- **問題**：研究の背景と目的、対処する問題の明示
- **仮説**：問題を解決するための自分のアイデア
- **実験手順**：研究手順の明示
- **結果**：研究結果の正確かつ明確な提示。グラフや表による視覚的表現
- **考察**：研究結果に基づく分析
- **引用文献**：先行研究や参考文献の明示

データサイエンスは万能ではない：3つの落とし穴

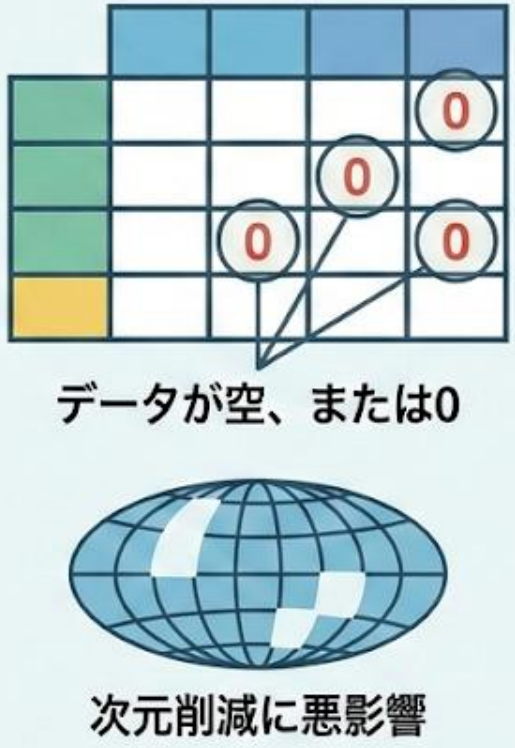
ノイズ



外れ値



計測漏れ



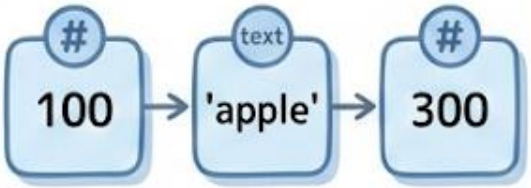
不適切なデータは手作業や
適切な分析手法で除去・補完が必要

図解は Nano Banana Pro を用いて作成

Pythonの基本データ構造

リスト (List)

順序あり・異なる型混在可




```
a = [100, 'apple', 300]
```

```
[100, 'apple', 300]
```

柔軟なデータ保管。

データフレーム (DataFrame)

表形式データ・ラベル付き

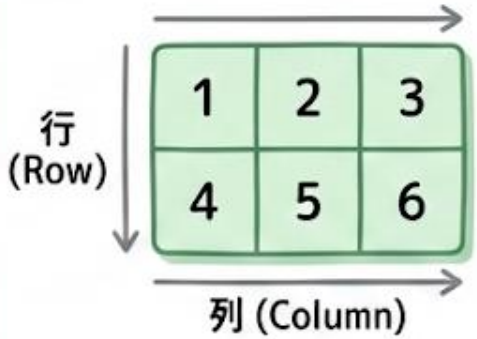
id	name	price
1	'apple'	100
2	'orange'	300
3	'orange'	200

```
import pandas as pd
d = [[1, 'apple', 100], ...]
df = pd.DataFrame(d,
columns=['id', 'name', 'price'])
```

データ分析に最適。

配列 (Array)

多次元データ・数値計算向け

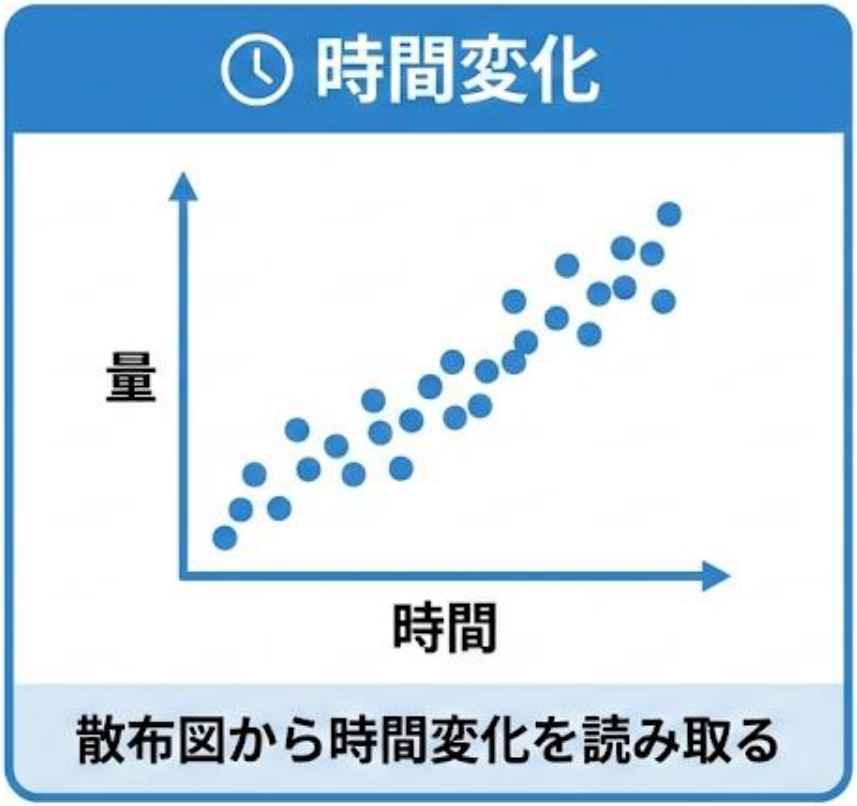


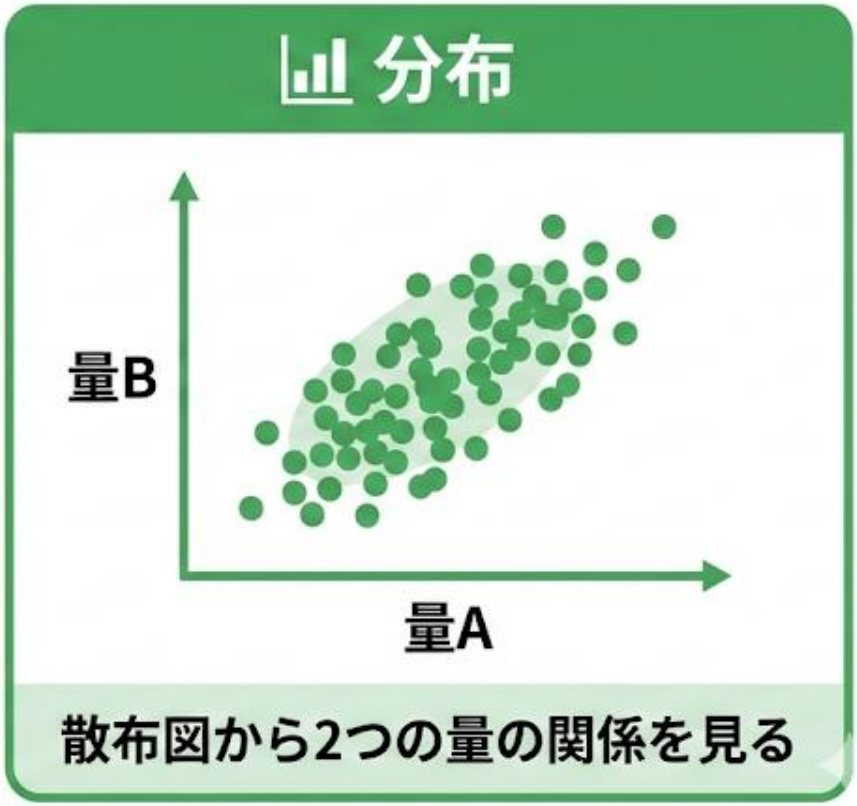

```
import numpy as np
a = np.array([[1, 2, 3],
              [4, 5, 6]])
```

高速な数値処理。

図解は Nano Banana Pro を用いて作成

散布図の使い分け：時間変化と分布



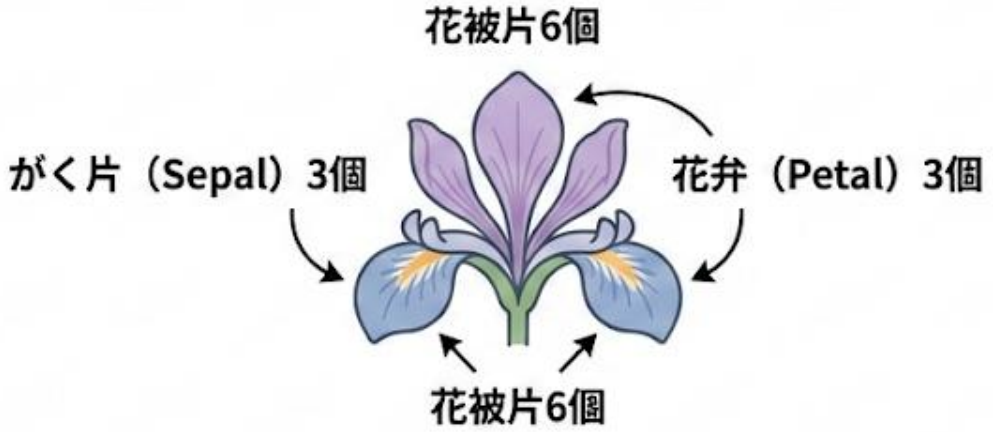


図解は Nano Banana Pro を用いて作成

Irisデータセット概要

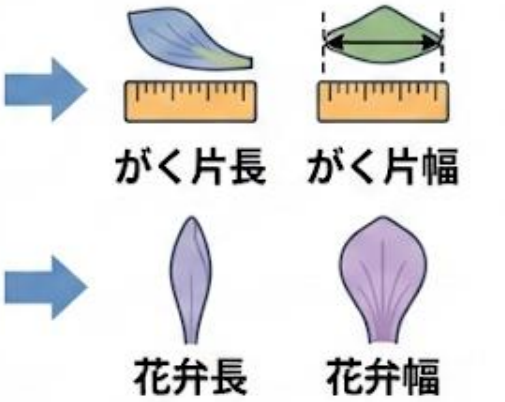
アヤメ属（Iris）

多年草



Irisデータセット

対象3種



がく片、花弁の幅・長さ計測

データ数：150
（50 × 3種）

1936年

作成者：Ronald Fisher

図解は Nano Banana Pro を用いて作成

Irisデータセット：特徴量と種類（ラベル）

`x`（特徴量データ：2次元配列）

がく片の長さ (cm)	がく片の幅 (cm)	花弁の長さ (cm)	花弁の幅 (cm)
5.1	3.5	1.4	0.2
4.9	3.0	1.4	0.2
6.4	3.2	4.5	1.5
5.1	3.0	1.6	0.3
4.7	3.0	4.6	1.5
⋮	⋮	⋮	⋮
6.4	3.2	4.5	1.5

各行＝1つのアヤメの測定値（4つの値）

`y`（ターゲットデータ：1次元配列）

0
0
1
2
0
1
2
⋮
...

0 = Setosa（種類A）

1 = Versicolor（種類B）

2 = Virginica（種類C）

各値＝xに対応する正解ラベル（アヤメの種類）

データセットの構成：特徴量（x）＋正解ラベル（y）

機械学習で分類を学習するためのペアデータ

図解は Nano Banana Pro を用いて作成

散布図作成の基本フロー（Python & Matplotlib）

1. 準備

```
!pip install japanize-matplotlib
```

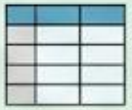
← 日本語化ライブラリのインストール

```
import matplotlib.pyplot as plt
import japanize_matplotlib
from sklearn.datasets import load_iris
```

🇯🇵 グラフの日本語表示（タイトル等）に必須

2. データ読込

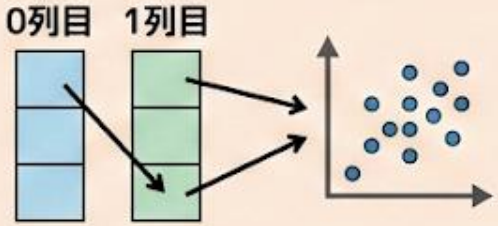
```
iris = load_iris()
x = iris.data
y = iris.target
```



Irisデータセットの読み込みと抽出

3. 描画命令

```
plt.scatter(x[:, 0], x[:, 1])
```



0列目と1列目のデータで散布図を作成

4. 装飾・表示

```
plt.xlabel(iris.feature_names[0])
plt.ylabel(iris.feature_names[1])
```

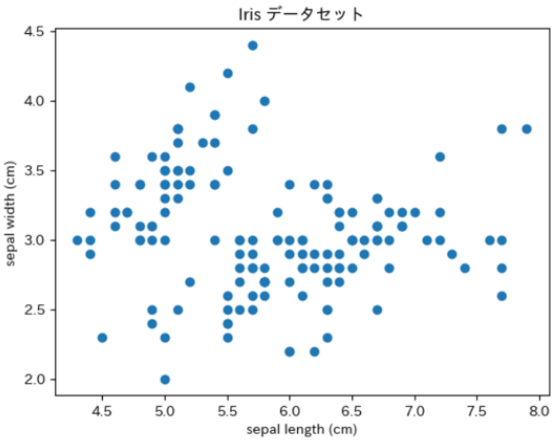
軸ラベルの設定（英語）

```
plt.title('Iris データセット')
```

タイトルの設定（日本語）

```
plt.show()
```

グラフの表示



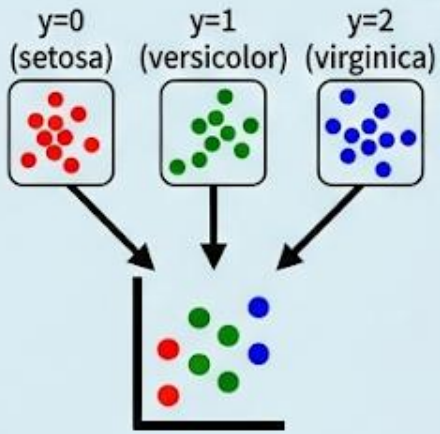
生成された散布図

図解は Nano Banana Pro を用いて作成

散布図作成の応用（アヤメの種類別色分け）

1. 種類別の描画ループ

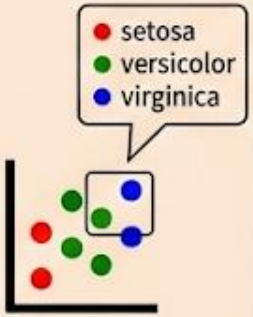
```
colors = ['red', 'green', 'blue']
for i in range(3):
    plt.scatter(x[y==i, 0], x[y==i, 1],
            c=colors[i],
            label=iris.target_names[i])
```



ターゲット(y)の値ごとにデータを抽出し、色とラベルを指定してプロット

2. 凡例の追加

```
plt.legend()
```

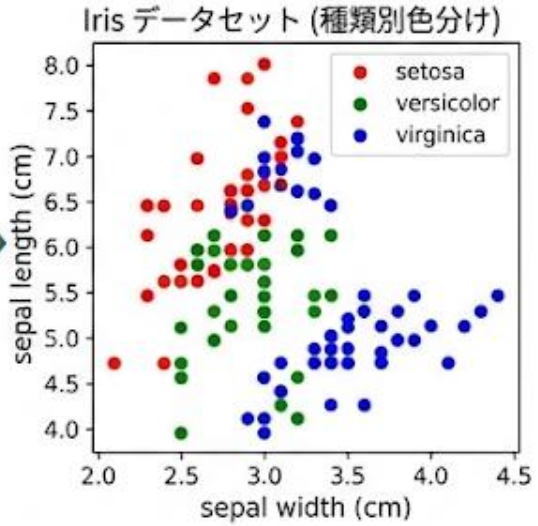


色と種類の対応を示す凡例を表示

3. 装飾・表示 ※前回と同じ

```
plt.xlabel(...)
plt.ylabel(...)
plt.title(...)
plt.show()
```

軸ラベル、タイトルの設定とグラフ表示



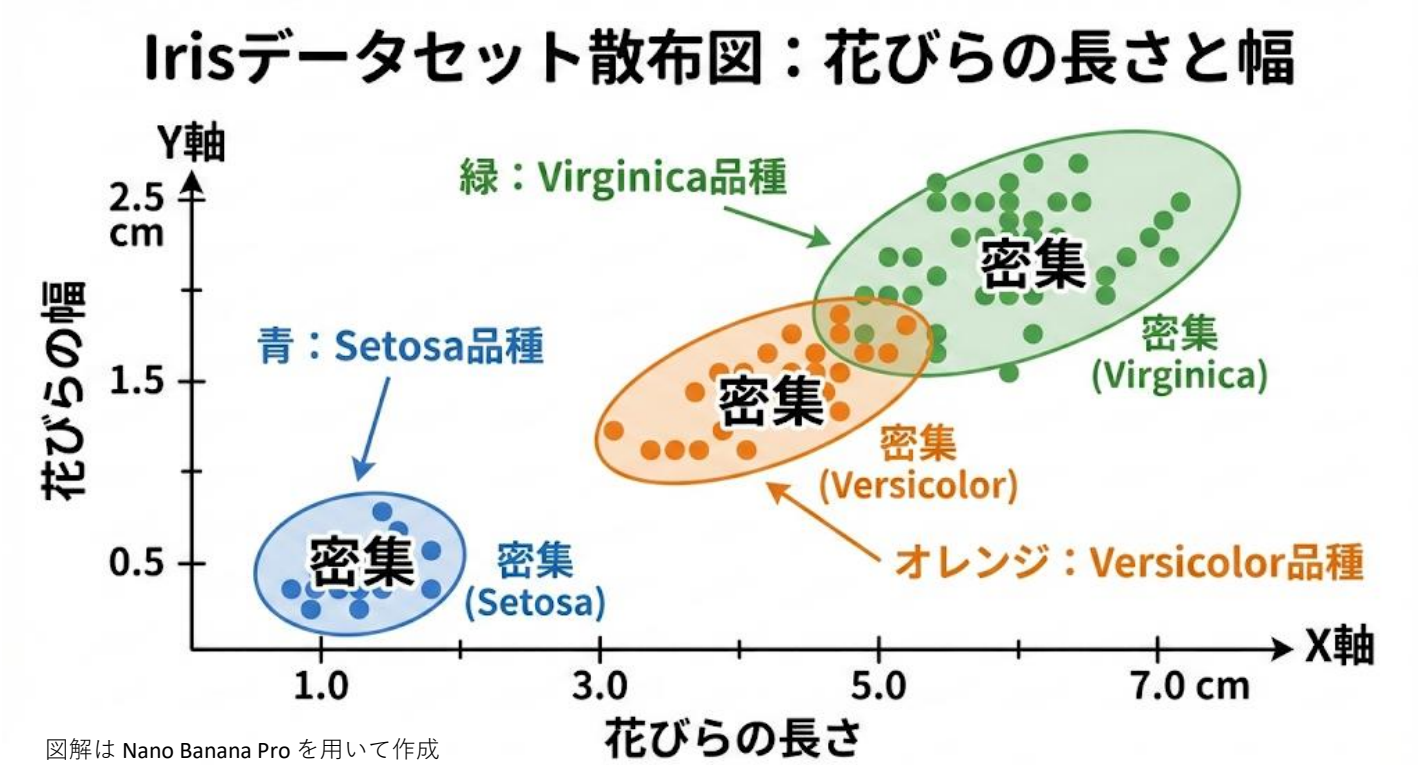
完成した色分け散布図

図解は Nano Banana Pro を用いて作成

分布から読み取れること

散布図から以下を読み取れる。

- 幅と長さの間に**関係がある（相関）**
- **花の種類**ごとに幅と長さの分布が異なる
- 同じ種類のデータは**密集する**傾向がある



図解は Nano Banana Pro を用いて作成

複数データのグラフ化（plt.plotの重ね書き）

同じキャンバスに層（レイヤー）を重ねるイメージ

手順：コードの実行

</>

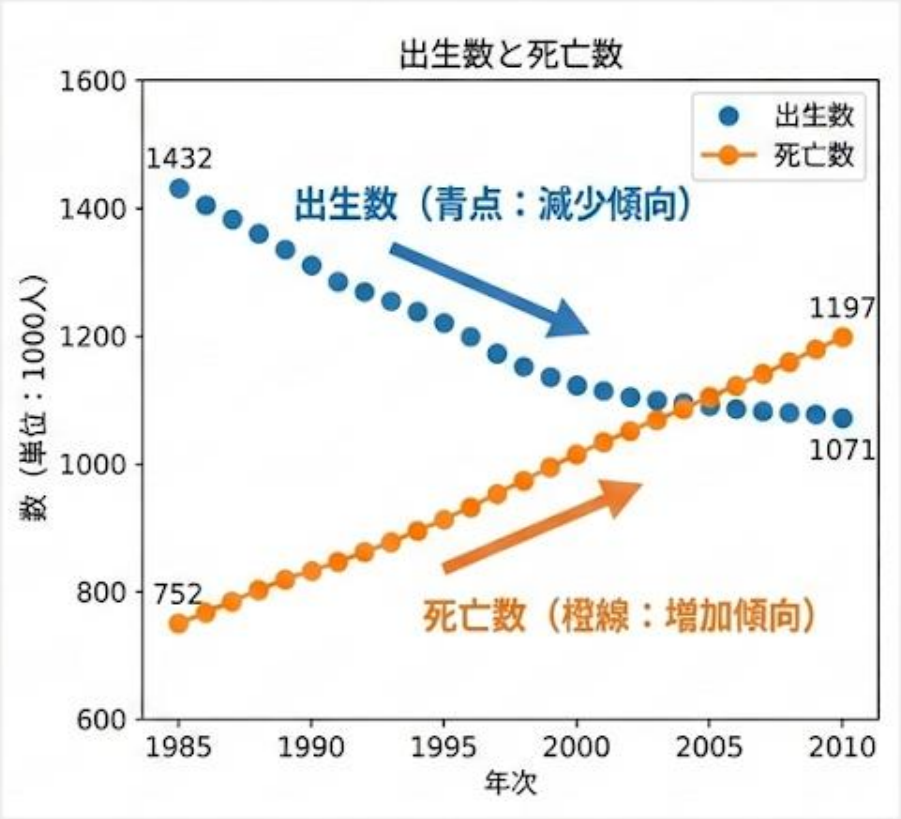
1. plt.plot(x, y1, 'o', label='出生数')

→ データ1（青点）を描画

2. plt.plot(x, y2, 'o-', label='死亡数')

→ データ2（橙線）を重ねて描画

結果：完成したグラフ



まとめ：plt.plotを複数回呼ぶことで、異なるデータを同一グラフ上に視覚化できる。

図解は Nano Banana Pro を用いて作成

平均の概念

合計（Sum）÷ 個数（Count）＝ 平均（Average）

例1：単一の値

120 ÷ 4 = 30

例2：複数の値の組

(120, 40) ÷ 4 = (30, 10)

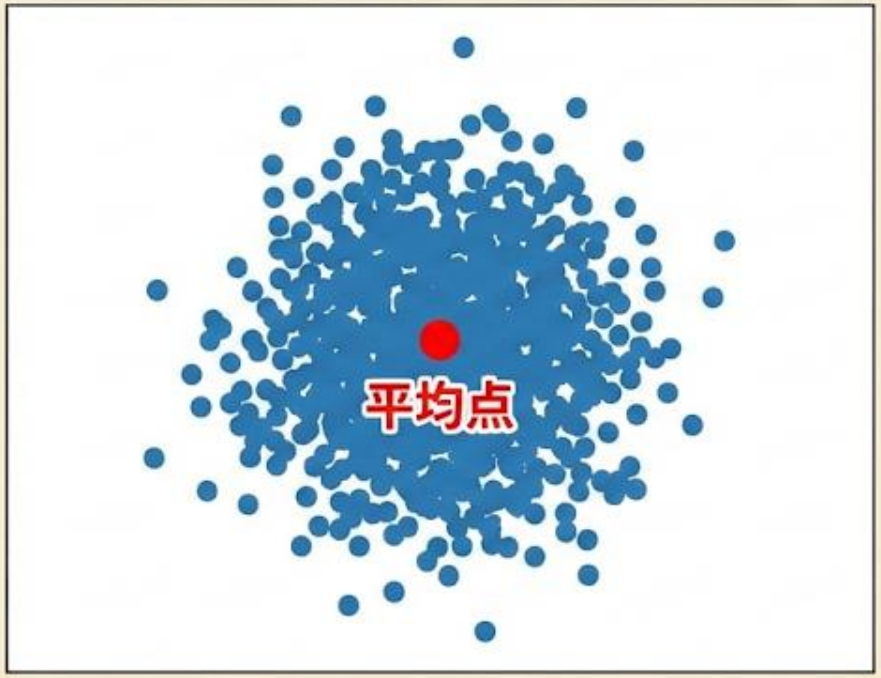
代表値として利用

複数計測で誤差軽減

図解は Nano Banana Pro を用いて作成

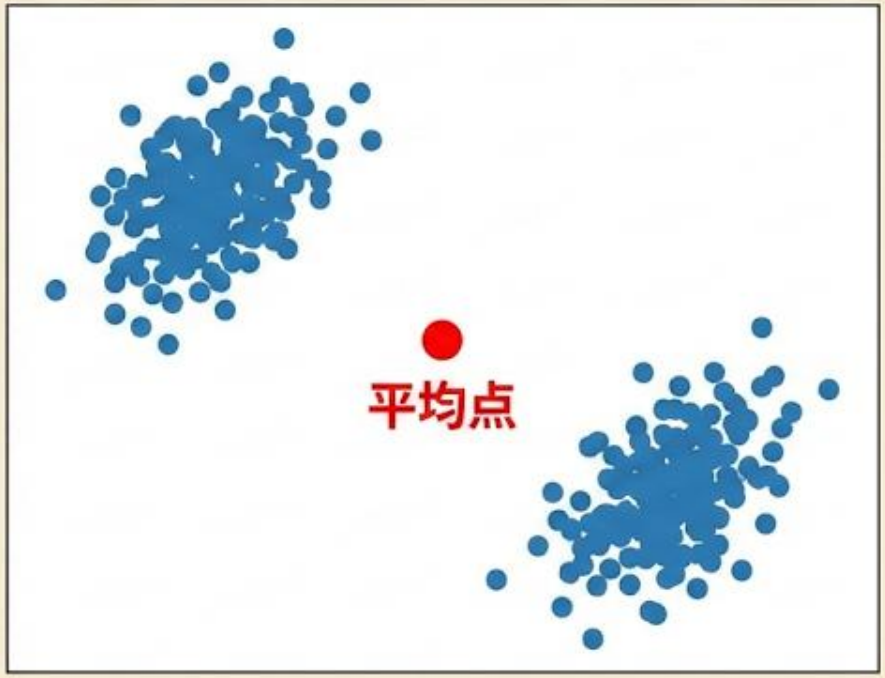
平均を使うときの注意点

平均が代表として適切な例



データが中心に集中しており、平均点は全体の傾向をよく表している。

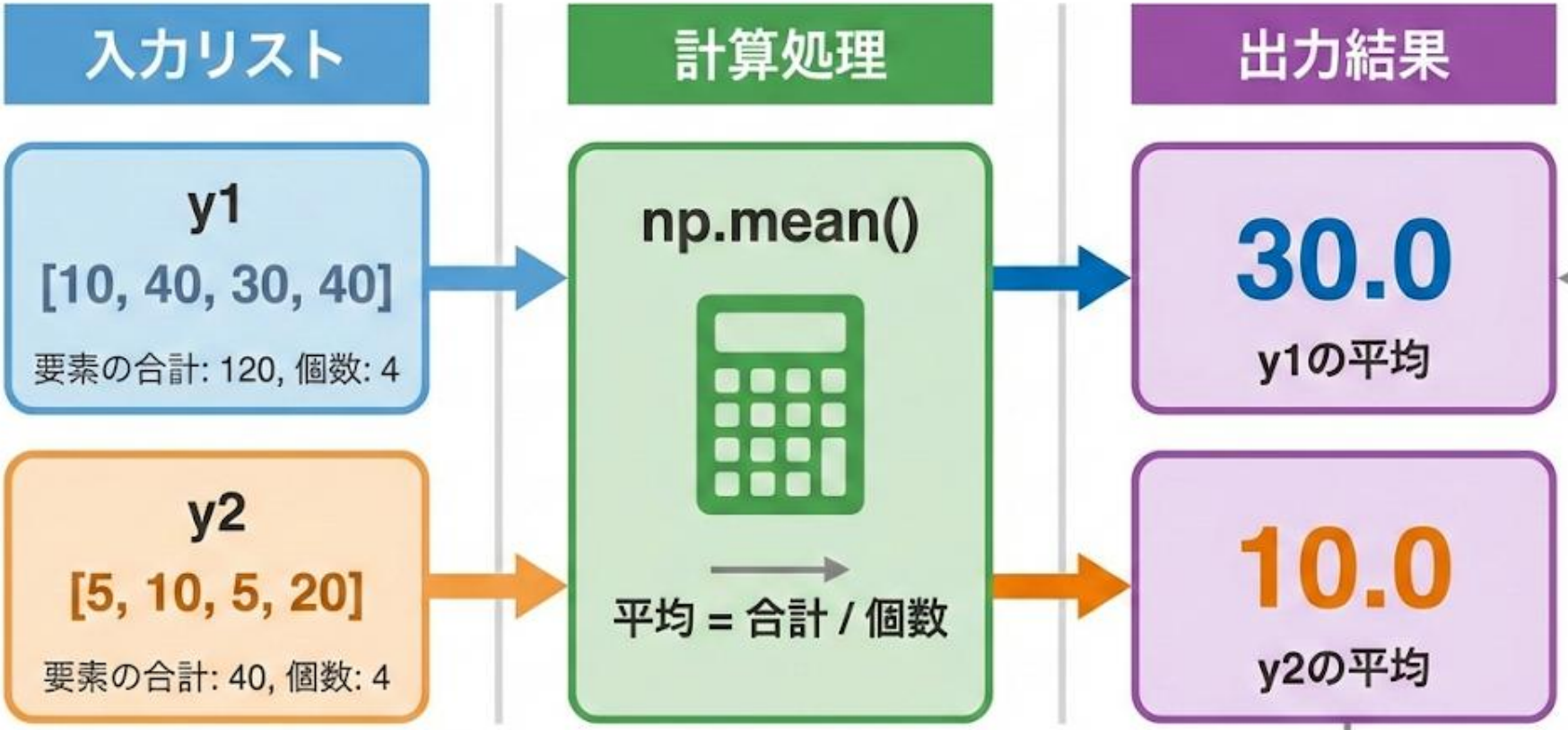
平均が代表として不適切な例



データが二極化しており、平均点はどちらの集団にも属さず、実態を表していない。

図解は Nano Banana Pro を用いて作成

Python: NumPyによる平均算出

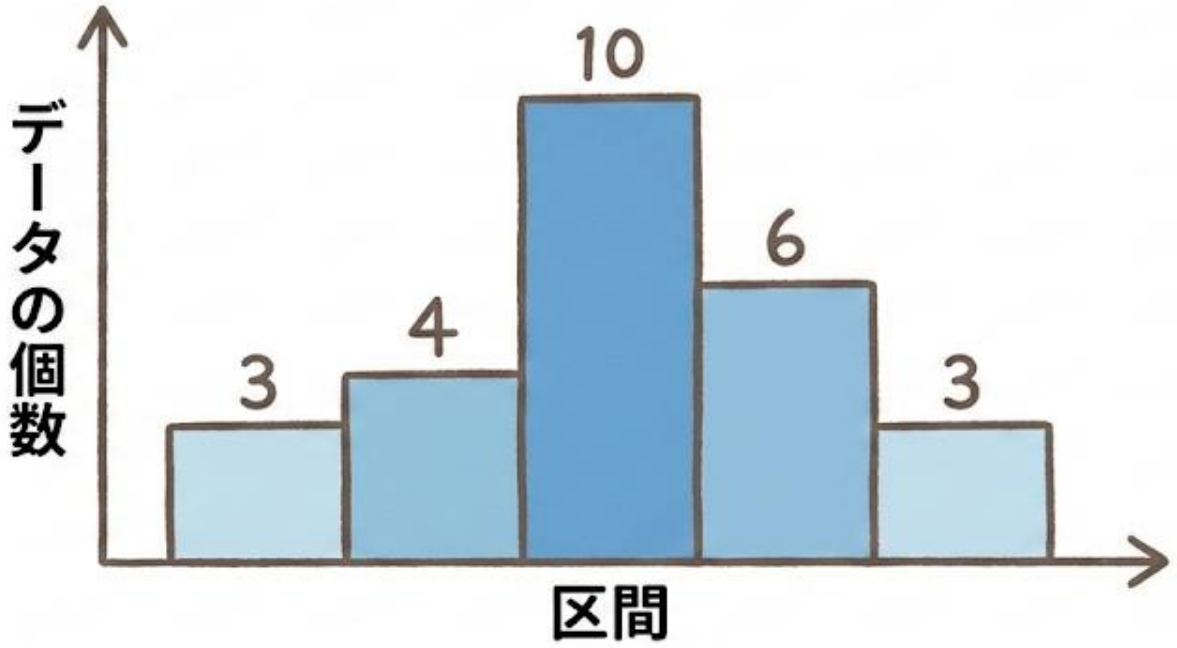



```
import numpy as np
y1 = [10, 40, 30, 40]
y2 = [5, 10, 5, 20]
print(np.mean(y1))  # 結果：30.0
print(np.mean(y2))  # 結果：10.0
```

図解は Nano Banana Pro を用いて作成

ヒストグラム

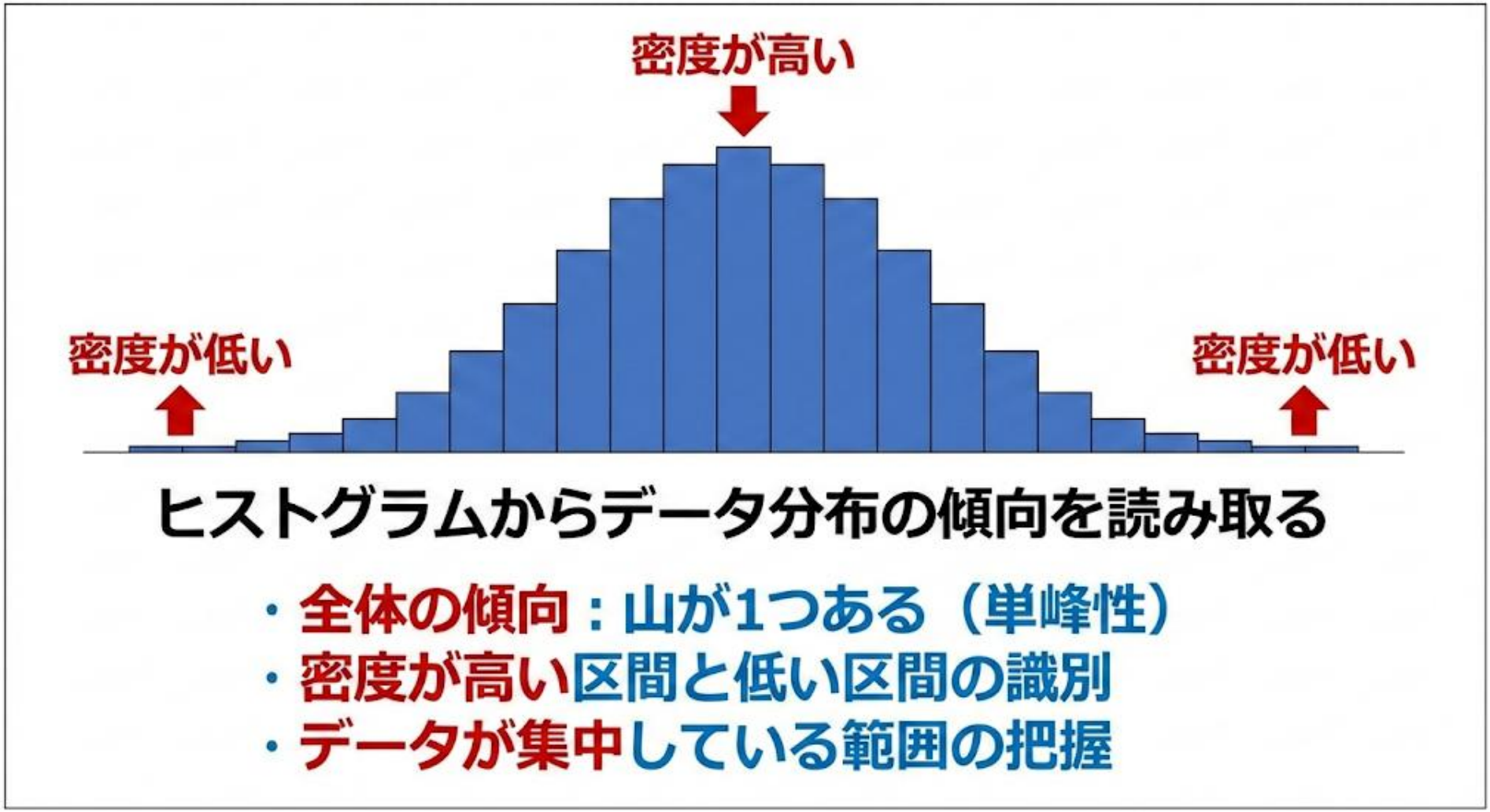
区間ごとにデータを数え上げたもの



図解は Nano Banana Pro を用いて作成

ヒストグラムから読み取れること

ヒストグラムからデータ分布の傾向を読み取れる。



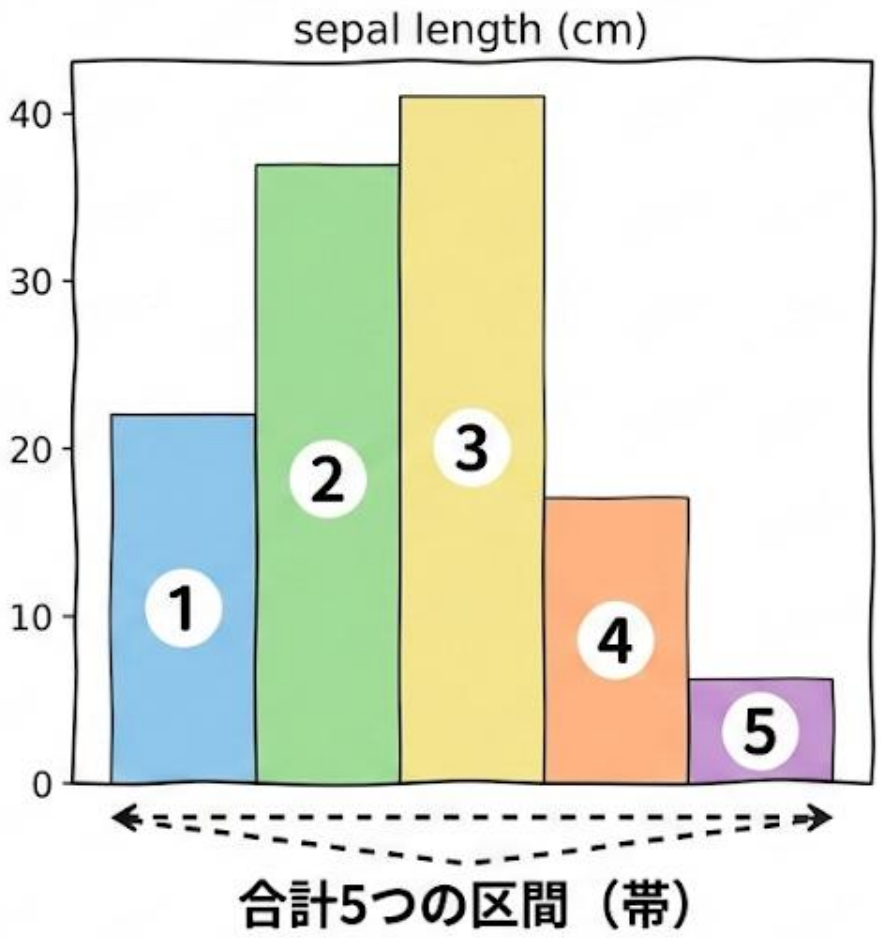
図解は Nano Banana Pro を用いて作成

Pythonヒストグラム作成：区間数の指定（bins）

```python
import matplotlib.pyplot as plt
import japanize_matplotlib
from sklearn.datasets import load_iris

iris = load_iris()
x = iris.data
y = iris.target

plt.hist(x[:, 0], bins=5)  # bins=5で区間数を5に設定
plt.title(iris.feature_names[0])
plt.show()
```



「bins=5」で区間数（帯の数）を5に指定。

データ：アヤメの「がく片の長さ」(x[:, 0])

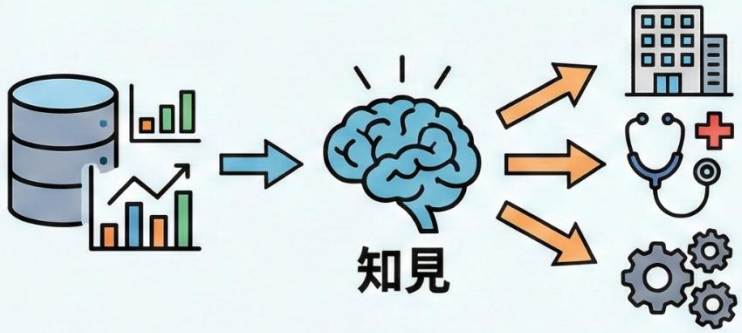
結果：ヒストグラムが5つの区間に分割されて描画される。

図解は Nano Banana Pro を用いて作成

全体まとめ

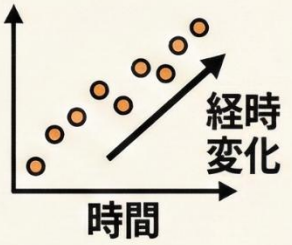
データサイエンス

データから正しい知見を導く学問
ビジネス・医療・工学など幅広く活用

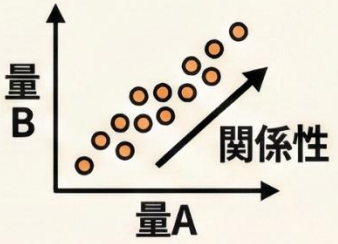


散布図の用途

時間変化を見る：
横軸に時間をとり
経時変化を読み取る



関係性を見る：
2つの量の関係を
読み取る



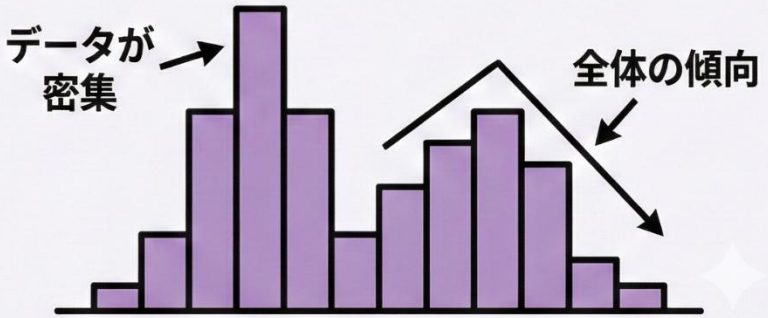
平均の使いどころ

○ データが1つの山に集中 → 平均が有効
× データが2つ以上に分離 → 平均は不適切

使用前に分布の形を確認すること

ヒストグラムの見方

棒が高い区間にデータが密集している
山の数と形から全体の傾向がわかる



図解は Nano Banana Pro を用いて作成