

機械学習 入門：画像判定

広島大学 AI・データイノベーション教育研究センター

滑川 裕介

目標

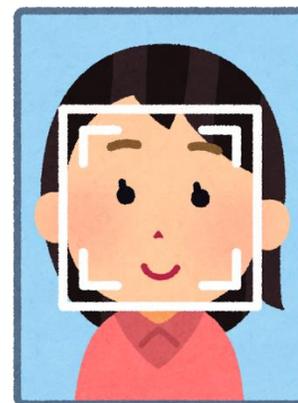
機械学習を用いた画像判定が実行できるようになる。

この授業で紹介すること

- 画像判定に関わる機械学習の概観
- Python による機械学習と画像判定

キーワード

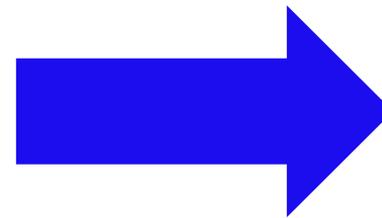
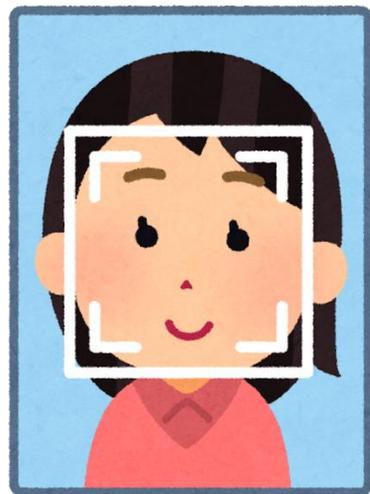
機械学習、画像判定、k 近傍法



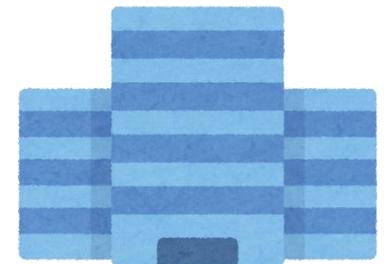
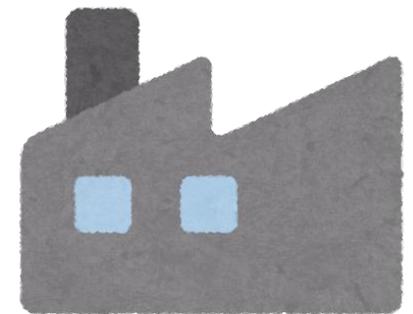
こんなことは、ありませんか？

顔認証や指紋認証を使用したことは、ありませんか？

これらの認証技術は機械学習により大幅に精度が向上しました。
このような画像判定は工場、病院などの現場で応用されています。



応用

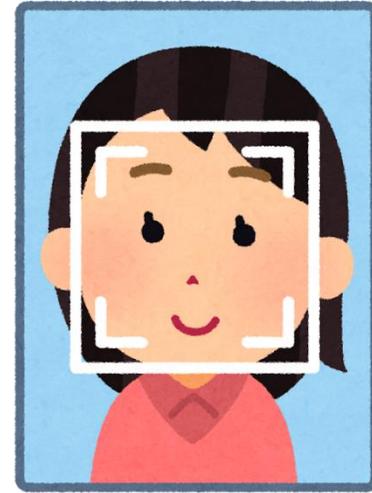


機械学習による画像認証

機械学習とは人工知能の一種であり、データの学習により自動的に性能を改善する手法です。機械学習を用いることで、データの分類・予測（回帰）ができます。たとえば、画像データの学習により、同一人物かどうか判別できます。

分類

予測



例題：画像から数字を読み取る

実際に画像データを学習させ、数値を判定してみましょう。



この授業では scikit-learn (サイキット ラーン) という Python の機械学習ライブラリを使用します。Google Colab などで次ページのプログラムを実行してみてください。

例題の Python コード

```
# ライブラリの読み込み
from sklearn.datasets import load_digits
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier

# 画像データの読み込みと前準備
data = load_digits()
data_reshape = data.images.reshape(len(data.images), -1)
x_train, x_test, y_train, y_test = train_test_split(data_reshape, data.target)

# 訓練データの学習
model = KNeighborsClassifier()
model.fit(x_train, y_train)

# テストデータを用いた判定
predicted = model.predict(x_test)

# 判定結果の図示
import matplotlib.pyplot as plt
fig, axes = plt.subplots(1, 6)
for ax, image, prediction in zip(axes, x_test, predicted):
    ax.imshow(image.reshape(8, 8), cmap=plt.cm.gray_r, interpolation='bicubic')
    ax.set_axis_off()
    ax.set_title('Predict %i' % prediction)
plt.show()
```

判定結果

おおむね正しく
判定できている

Predict 5 Predict 2



Predict 2 Predict 8



例題の解説 1 : scikit-learn ライブラリ

ライブラリの読み込み

```
from sklearn.datasets import load_digits
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
```

scikit-learn (サイキット ラーン) は Python の機械学習ライブラリです。

練習用データセット、データ整形ツール、機械学習部分を読み込んでいます。



scikit-learn

Machine Learning in Python

Getting Started

Release Highlights for 1.4

GitHub

- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

例題の解説 2 : 画像データ

```
# 画像データの読み込みと前準備
```

```
data = load_digits()
```

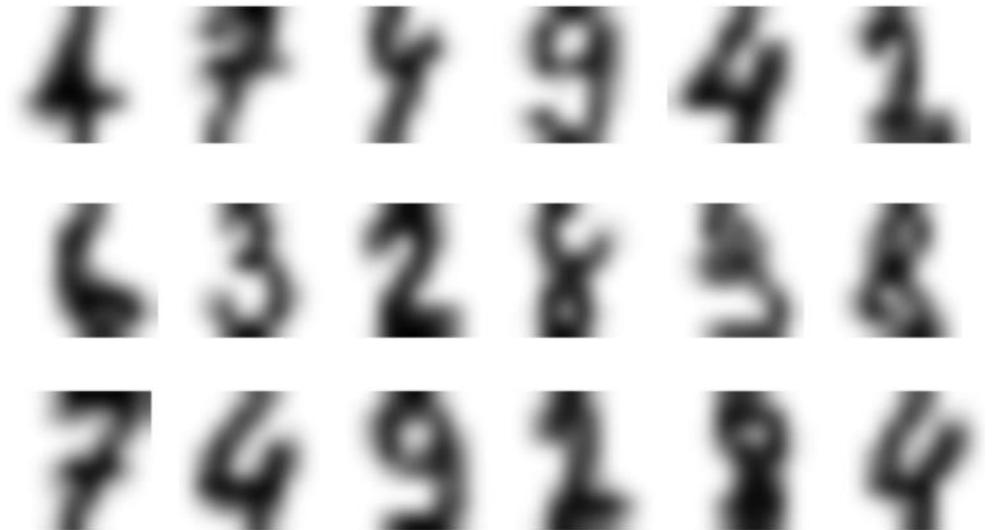
```
data_reshape = data.images.reshape(len(data.images), -1)
```

```
x_train, x_test, y_train, y_test = train_test_split(data_reshape, data.target)
```

練習用データセットライブラリから、実際に使う画像データを読み込んでいます。

ただし、そのままでは使いづらいため、データの変形および分割をしています。

画像データ例



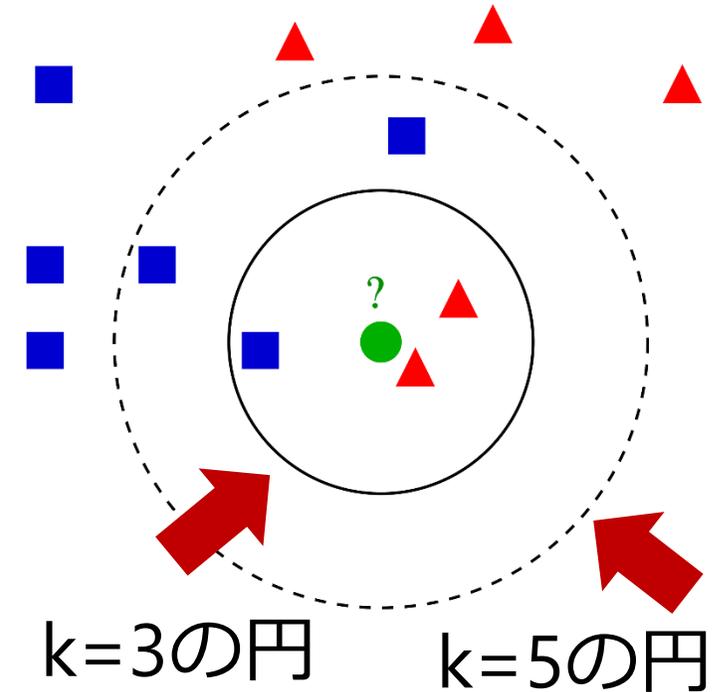
例題の解説 3 : k 近傍法

訓練データの学習

```
model = KNeighborsClassifier()  
model.fit(x_train, y_train)
```

例題では k 近傍法 (k-nearest neighbor 法) を学習・判定に用いました。

他にも、サポートベクターマシン法、確率的勾配降下法など、さまざまな手法があります。



Wikipedia (2024)

k 近傍法 [Fix and Hodges (1951)] :
入力と学習データとの距離 (類似度) を計算し上位 k 番目
までを選び、選んだデータの多数決を出力とする方法。

例題の解説 4 : matplotlib による図示

```
# 判定結果の図示
import matplotlib.pyplot as plt
fig, axes = plt.subplots(1, 6)
for ax, image, prediction in zip(axes, x_test, predicted):
    ax.imshow(image.reshape(8, 8), cmap=plt.cm.gray_r, interpolation='bicubic')
    ax.set_axis_off()
    ax.set_title('Predict %i' % prediction)
plt.show()
```

機械学習で判定した結果を matplotlib ライブラリを用いて図示しています。

matplotlib

出力結果

Predict 5 Predict 2



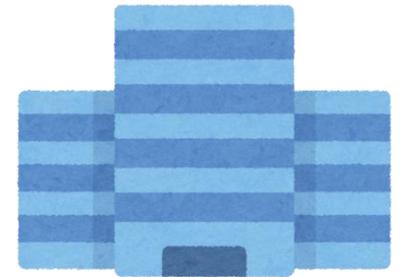
Predict 2 Predict 8



機械学習による画像判定の応用例

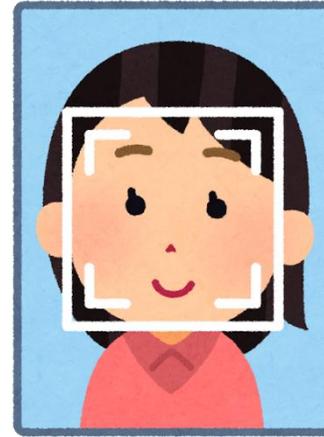
ここまで説明してきた機械学習による画像判定は、すでにさまざまな現場で使用されています。

- セルフレジ：売店、食堂で使用
- 良品・不良品の判別：工場で使用
- パスポートの顔認証：空港で使用
- 病気の判定（画像診断）：病院で使用
- 動物の種類判定：害獣駆除用の罠で使用



まとめ

機械学習を用いた画像判定について解説しました。



Python 機械学習ライブラリ scikit-learn を使用することで、比較的簡単に機械学習を用いた画像判定ができます。学習データ数の制限やアルゴリズムの限界のため常に正しく画像判定できるわけではありませんが、おおむね正しく判定できていたかと思います。

今回説明した機械学習による画像判定は、さまざまな現場で使用されており、今後も幅広い分野で応用されると考えられます。