

## 14 機械学習アルゴリズムの研究

工藤一樹, 松岡大夢

指導教員 石館 勝好

### 1. はじめに

最近、ニュースで Pepper や自動車の自動運転の話題をよく目にし、強く興味を持った。調べてみると、これらには共通して機械学習が使われていることが分かった。

そこで本研究では、機械学習のアルゴリズムを勉強して、応用プログラムを作成したい。そして、その知識を仕事をするうえで活かしたい。

### 2. 研究の概要

当初は、機械学習のライブラリを用いて、画像認識のプログラムの作成を通して、アルゴリズムの勉強をする予定だった。しかし、ライブラリを使用していくだけでは機械学習のアルゴリズムの理解に繋がらず、まずはアルゴリズムの勉強を中心に行なうこととした。そして、アルゴリズムを理解したうえでプログラムの作成に取り掛かる。言語は Python を使用する。

### 3. 機械学習

#### 3.1 機械学習について

センサやデータベースなどから、ある程度の数のサンプルデータ集合を入力して解析を行い、そのデータから有用な規則、ルール、知識表現、判断基準などを抽出し、複雑なデータを分類・予測できるようにするものである。

機械学習は検索エンジン、医療診断、スパムメールの検出、金融市場の予測、DNA 配列の分類、音声認識や文字認識などのパターン認識、ゲーム戦略、ロボット、など幅広い分野で用いられている。

また、教師あり学習と教師なし学習の 2 つに分けられ、使用用途によって使い分けられる。

#### 3.2 教師あり学習

教師あり学習とは、事前にデータとラベルを与えておき、それをもとに学習する方法である。実用例としてはスパムメールフィルタなどがある。スパムメールとスパムではないメールをあらかじめ用意し、それぞれの共通の特徴やパターンを導き出し、それをもとに判断している。

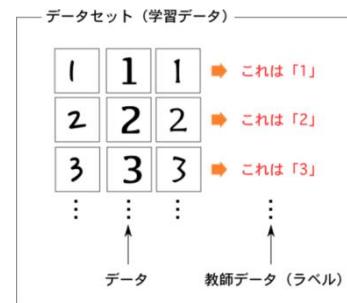


図 1 教師あり学習イメージ

#### 3.3 教師なし学習

教師なし学習とは事前にデータのみを与えておき、特徴や法則を見つけることが目的である。例としては、アンケート分析による顧客などのグループ分けがある。アンケートの回答内容の特徴を探し、その特徴をもとにいくつかのグループに分ける。ただし、データのみだと求めていた結果がないから、前提条件を与える必要がある。

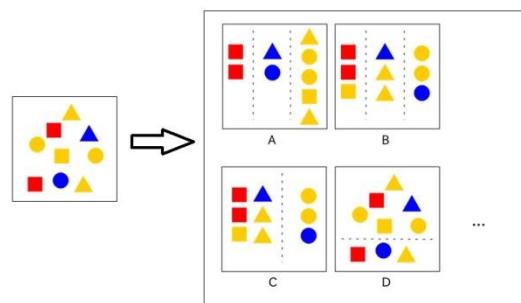


図 2 教師なし学習イメージ

#### 4. 機械学習ライブラリの利用

初めに私たちは、OpenCV に付属しているライブラリで分類器を作成した。分類器とは機械に大量のデータと規則を教えてやり、それをもとに分類する方法である。今回はそのアプリケーションでりんごとりんごではないものの分類に挑戦した。しかし、りんごが認識されなかつた。原因としては①画像枚数不足②画像の質が悪い(画質が悪い、余計なものが写っている)が考えられる。

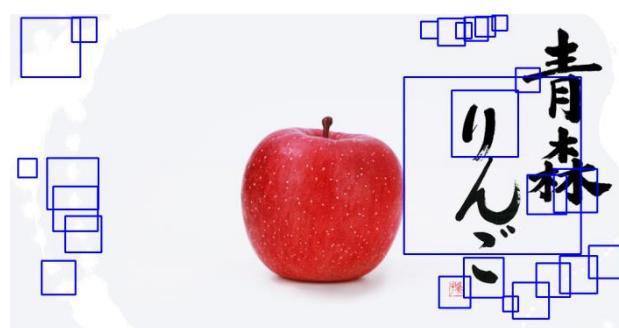


図 3 リンゴの認識失敗例

次に、分類器を作成することは難易度が高いということが分かったので、とりあえず OpenCV に最初から入っている顔を認識する分類器を使ってプログラムの作成を試みた。結果は以下のとおりである。



図 4 顔認識の成功例

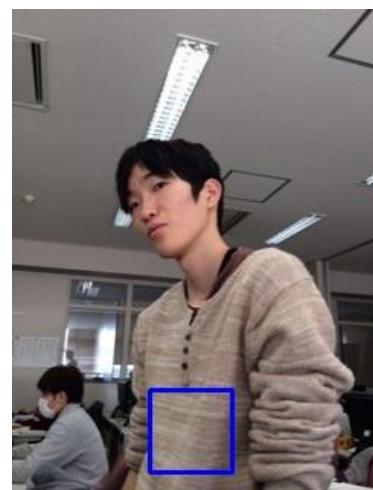


図 5 顔認識の失敗例

図 4、図 5 で示すように、成功した例もあつたが、ほとんどが顔を認識してくれなかつた。私たちは、分類器を改善すれば認識率が上がると考え、アルゴリズムの改善を試みた。しかし、OpenCV が用意しているアプリケーションはブラックボックスでどういうことをしているか分からなかつた。そこで、OpenCV を使った画像認識プログラムを作成するのは諦め、基礎的な機械学習のアルゴリズムについての学習から始めることにした。

#### 5. k 平均法

##### 5.1 アルゴリズム

k 平均法とは、教師なし学習に分類される機械学習アルゴリズムである。異なる性質のものが混ざり合つた集団(データ)から、互いに似た性質をもつものを集め、グループを作る方法である。

##### 5.2 手順

###### ① 初期状態

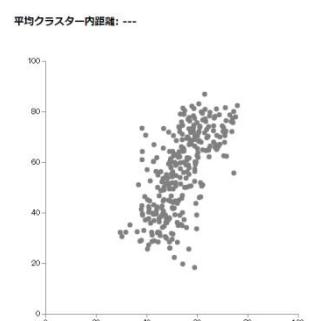


図 6 初期状態

- ② データの重心点をランダム  $k$  個決める  
(この図は 2 個)

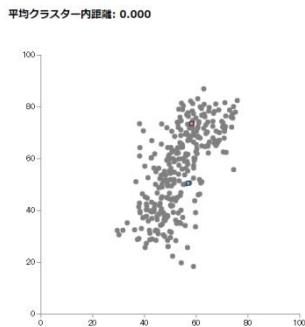


図 7 重心点決定

- ③ 各データを重心点に近いものに分類する

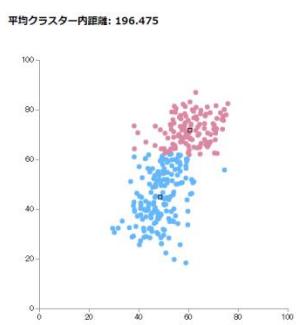


図 8 データ分類

- ④ クラスタ内でのデータの平均値を計算し、重心点を再設定する

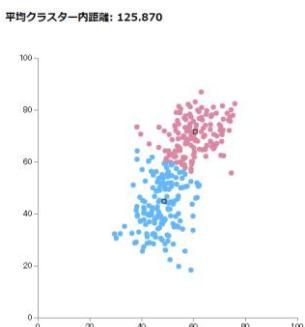


図 9 再計算

- ⑤ 各データを重心点に近いものに再分類する

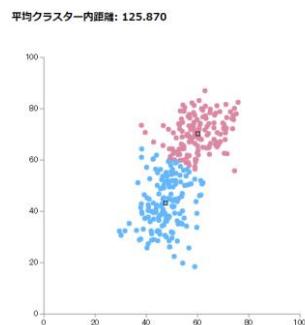


図 10 再分類

- ⑥ ④, ⑤を何度も繰り返し、変化がなくなったら終了

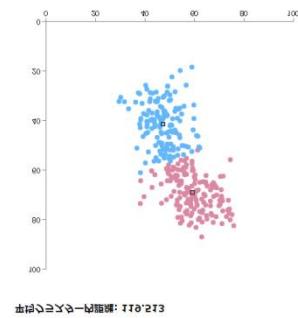


図 11 計算終了

### 5.3 応用例

参考書[1]には画像の減色処理に  $k$  平均法を応用したプログラムが紹介されている。

- (1) ランダムに  $k$  個の代表色を選択.
- (2) 1 ピクセルごとに  $k$  個の代表色から一番近いものを選んで、全ピクセルを  $k$  個に分類.
- (3) それぞれのクラスタの平均値を新たな代表色とする.
- (4) 新しい代表色を用いて、再度(2)の処理を行う.
- (5) 変化がなくなったら終了.

これを図 12 を対象に実行したところ、図 13 のように色の種類が  $k$  の値の数(今回は 2)だけになった。



図 12 元画像

図 13  $k=2$  の時の実行結果

### 5.4 $k$ 平均法まとめ

$k$  平均法については大まかな部分は理解できた。ただ、この  $k$  平均法を画像認識のプログラムに応用するためには、どのような数式が必要なのか、結果の誤差など考慮する必要があると感じた。

## 6. EM アルゴリズム

EM アルゴリズムとは最尤推定を基本としたアルゴリズムである。(最尤推定とは観測データに対して最もらしいデータの発生源の確率を求めるこことである。)

手順としては、期待値ステップと最大化ステップを交互に繰り返すことで計算を進めていく。期待値ステップでは推定されている分布に基づいてモデルの尤度の期待値を計算する。最大化ステップでは期待値ステップで求まった期待値を最大化するようなパラメータを求める。

例として、当たりが出る確率が異なる 2 つのくじを用意し、数回くじを引いた結果からどちらのくじから引いたのかを求めたいときに EM アルゴリズムが使われる。

この EM アルゴリズムは機械学習への応用ができる複数の画像を分類したり、複数の画像からその平均的な画像を生成したりすることができる。

### 6.1 応用例

参考書[1]では手書き文字を分類するプログラムが紹介されている。



図 14 画像が生成されていく様子

図 14 は EM アルゴリズムの手続きによって、一番左の画像生成器で用意したランダムな画像が数字の形になっていく様子である。



図 15 分類結果

図 15 は図 14 で生成された画像をもとに複数の手書き文字を分類した結果である。よく見てみると「6」のグループに「0」の画像が含まれている。

これは「0」の形が細長いため「6」の方が形が近いと判断されたと考えられる。

### 6.2 EM アルゴリズムまとめ

EM アルゴリズムがどういうものなのかはイメージすることができたが数式やプログラムがまだしっかり理解できていないのでこのサンプルプログラムをいじったりして理解を深めていきたいと思う。

### 7. 終わりに

2 つのアルゴリズムの勉強が数学的知識不足で数式などが分からず理解するのに時間がかかってしまった。今後は、数式も理解した上で、プログラムを作成する。

### 8. 参考文献

#### [1]参考書

「機械学習理論入門」 中井悦司 著

#### [2]機械学習について

<http://kazoo04.hatenablog.com/archive/2015>

#### [3]k 平均法

<http://enakai00.hatenablog.com/entry/2015/04/14/181305>

#### [4]EM アルゴリズム

<http://d.hatena.ne.jp/kenta1984/20100206/1265448744>

#### [5]最尤推定法

<http://www.slideshare.net/iranainanimosuteteshimaou/ss-55173144>