

自作AIステップ3…挑戦! 画像ディープ・ラーニング

ご購入はこちら

牧野 浩二, 西崎 博光

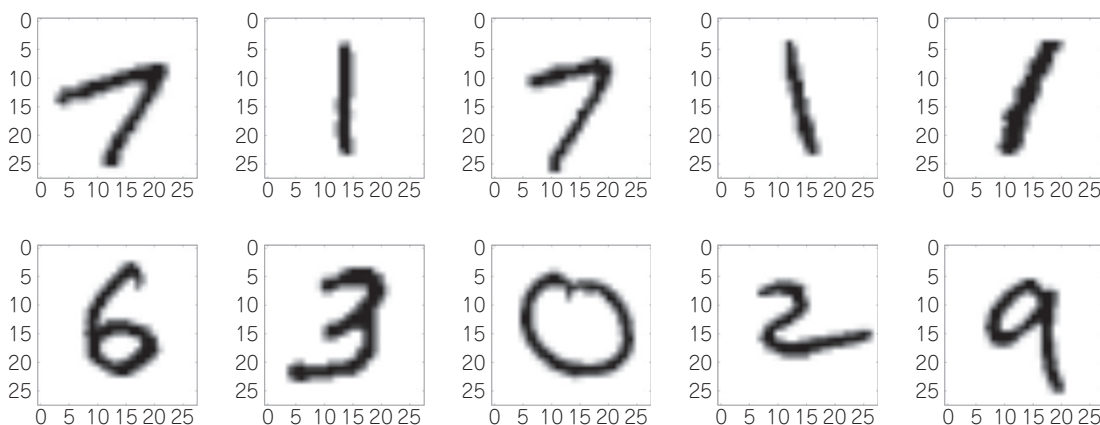


図1 (1) 手書き文字データMNISTの例

EMニスト
MNISTは、ディープ・ラーニングのアルゴリズムを試してみるために用意された「手書き文字のデータ・セット」です。手書き文字を分類することはディープ・ラーニングが出てくる前は難しい問題でした。

MNISTは、Chainerに限らずTensorFlowなど有名なディープ・ラーニングでも実行できるため、これらフレームワークの性能評価にも使われています。

難しそうに見えて実は簡単に書くことができるので、「なんだかものすごいディープ・ラーニング」を、体験しながら学ぶ「はじめの一步」にちょうどよいプログラムとなっています。

また、この手書き文字の認識は画像認識ですので、

少しプログラムを変えるだけで、この後に出てくる3大ディープ・ラーニングの1つである畳み込みニューラル・ネットワークでも実現できます。ディープ・ニューラル・ネットワークの基本を学びながら畳み込みニューラル・ネットワークへの拡張の布石となります。

図1のような手書き文字のデータがウェブで公開されていて、Chainer以外のいろいろなフレームワークでも利用されています。ディープ・ラーニングの性能や書き方を比較する際に広く用いられています。

Chainerのサンプル・プログラムの中にも、MNISTを学習するプログラムがあります。本稿では、分かりやすく改造したプログラムを使います。

実験

● できること…手書き文字の認識

手書き文字の画像データを入力し、文字を認識します。実行結果を図2に示します。

10個の画像を分類した結果を表1に示します。3と6はうまく分類できませんでした。

また、自分で作成した画像でも分類できます。詳細は後に示しますが、縦横の画素数が同じ(正方形の)画像に文字を書いた画像を入力として使えます。

```
$ python eval_myimage_gnn.py -i tegaki/
  tegakil.png -m result_DNN/MLP.model

DNN.model
自分の手書き文字を学習したモデルで評価してみるプログラム
# 入力画像ファイル: tegakil.png
# 学習済みモデルファイル: result/MLP.model

判定結果は1です。
```



(a) 入力画像

(b) 実行結果

図2 手書き文字を分類する