

# レッスン3… TensorFlow 初体験

渡邊 輝

ここではTensorFlowを初めて触るという方のために、簡単なニューラル・ネットワークを実装し、分類モデルを構築してみます。分類する対象は手書き数字で「0」から「9」までの画像データです。画像データはピクセルごとの数字の集合として表現される非構造化データですが、そのような複雑な数字の集合から画像の意味する数字を判別する分類モデルを構築します。

手書きの数字は同じ数字でもそれぞれの数字が独自の形態を持っています。また、書き手によって書き方、癖が異なるため、1つの数字に対するデータでもピクセル・レベルの数字データとして見ると、大きなばらつきを持った集合となります。このようなばらつきを持ったデータに対して、従来のプログラミング的なアプローチで人間が認識する意味（この場合数字0～9のいずれか）に分類しようとする、ほぼ無限に繰り返されるIF文のようなプログラムとなり、複雑にもかかわらず分類精度のあまり高くないものになってしまいます。

ニューラル・ネットワークのような機械学習モデルにデータの特徴（ばらつきの傾向）を学習させることで、そのようなプログラミングをすることなく高い精度での判別が可能となります。

## ● Googleのディープ・ラーニング・トレーニング用データで

利用するデータはMNIST<sup>注1</sup>という画像データです（図1）。機械学習分類モデルのベンチマークとしてよく用いられるデータセットで、トレーニング・データとして60000個、テスト・データとして10000個の手書き数字画像データとその正解ラベルを持っています。

トレーニング・データを利用してニューラル・ネットワークのモデル・トレーニングを行い、テスト・データを利用してその精度を確認します。機械学習モデルは自身が出力する判別結果（0～9の予測結果）と正解ラベルを比較し、その誤りを是正（＝トレーニングまたは学習）することで判別精度を高めます。なお、



図1 ディープ・ラーニング・トレーニング用の画像データ MNIST (<http://yann.lecun.com/exdb/mnist/>) 機械学習分類モデルのベンチマークとしてよく用いられる

本稿の内容はTensorFlow公式ページ<sup>注2</sup>で紹介されているものの意識+追記になります。

## ステップ1 データのダウンロード

### ● 入手先

前章までで解説した開発環境Jupyter Notebookを起動します。TensorFlowのチュートリアル用に用意されたinput\_dataという関数を使ってMNISTデータを読み込みます。初めてコードを実行する場合はデータのダウンロードから実行されます。

```
from tensorflow.examples.tutorials.mnist import input_data
mnist = input_data.read_data_sets("MNIST_data/", one_hot=True)
```

注2: <https://www.tensorflow.org/versions/r0.11/tutorials/mnist/beginners/index.html>

注1: <http://yann.lecun.com/exdb/mnist/>