

知能プログラミング演習 I 演習課題

梅津 佑太

umezu.yuta@nitech.ac.jp

2019 年 6 月 26 日

1 準備

- まだ演習用のフォルダを作っていない人は DLL のフォルダを作成
 - ホームディレクトリに演習用のディレクトリを作成

```
step1: mkdir -p DLL
```
- 作業ディレクトリ DLL に移動

```
step1: cd ./DLL
```
- 今日の課題を DLL にダウンロードして展開

```
step1: wget http://www-als.ics.nitech.ac.jp/~umezu/DLL19/Lec3.zip
```

```
step2: unzip Lec3.zip
```

 - 展開したフォルダの中に、以下のものがすべて入っていることを確認
 - * DNN.py
 - * report.tex
 - * task.pdf
- Lec3 へ移動

```
step1: cd ./Lec3
```

2 課題

Lec2 と同じ手書き文字 (28×28 ピクセル) の多値分類を深層ニューラルネットワークで実装する。以下のプログラムを作成せよ。ただし, DNN.py にコードを保存し, プログラムの出力結果はレポートとしてまとめること。

1. 必要なら, 前回までのプログラム `perceptron.py`, `NN.py` を参考にする。ことで, 深層ニューラルネットワークのアルゴリズムを完成させよ。また, 以下の点を考慮してプログラムを作成すること。ただし, DNN.py にコードを保存すること。
 - 分類するクラス数を 3 以上で指定する。
 - 活性化関数はシグモイド関数, ReLU, ハイパボリックタンジェントのうち, 少なくとも二つ用いること。
 - 最後の中間層から出力層への活性化関数はソフトマックス関数を用いることとし, 誤差関数はクロスエントロピーとする。
 - 中間層のユニット数は自由に定義して良いが, 少なくとも中間層が 2 層以上のネットワークを構成すること。
 - 訓練誤差, テスト誤差および confusion matrix を図として保存すること。
2. 解析結果のレポートを TeX で作成し, pdf ファイルに変換せよ。必要なら, `report.tex` を用いて良い。ただし, 以下のことに留意すること。
 - 設定した中間層の数や, 中間層ごとのユニット数, 各層で用いた活性化関数などの実験設定を正確に記述すること。
 - 必要なら解析結果の図 (誤差関数の推移や confusion matrix など) も用いること。
 - 解析結果に対する考察を述べること。

3 課題の提出

Moodle を使ってファイルを提出してください。提出方法は以下の通りです。

- Moodle にログインし、知能プログラミング演習のページへ移動。
- Lec3 の項目に、DNN.py, レポートをアップロードする。

7/2(火) の 17:00 (次回の授業前日) を提出期限とします。