

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 7 月 24 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/Lec7.zip`
step2: `unzip Lec7.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
 - * CNN2.py
 - * report.tex
 - * task.pdf
- Lec7 へ移動
step1: `cd ./Lec7`

2 課題

keras を用いて、手書き文字 (28×28 ピクセル) の多値分類を畳み込み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、CNN2.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. 以下の点に注意し、keras を用いて畳み込みニューラルネットワークのアルゴリズムを完成させよ。ただし、CNN2.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。
- ネットワークは Sequential, Model のいずれを用いても良い。
- 少なくとも 1 回ずつ畳み込み、プーリング、全結合を行うこと。なお、活性化関数は適当なものを用いるものとする。
- compile の引数 metrics には、accuracy を指定すること。つまり、該当箇所は

```
compiles = ["accuracy"]
```

とする。

- 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。
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。
- 設定した中間層のユニット数や、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 (誤差関数や分類精度の推移や 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。
- Lec7 の項目に、CNN2.py, レポートをアップロードする。

7/30(火) の 17:00 (次回の授業前日) を提出期限とします。