

課題 1 : 前処理プログラムの実装

ロボット動作学習に必要なデータセットの作成、共通処理のプログラムを実装。

フォルダ構成

dplmライブラリは以下のように構成される。

- layer : 自作 layer の保存場所 (e.g. MTRNN)
- model : 自作 model の保存場所 (e.g. CAEやCAE-RNN、CNNRNN)
- test : 自作関数、layer、model の検証用プログラム
- trainer : RNN 学習用 trainer class とラッパー
- tutorial : 課題一式
- utils : 前処理などのプログラム

課題

`dplm/utils/data.py` 内の以下4つの関数を順にコメントアウトしながら作成してください。なお、pythonのfor文は、実行速度が非常に遅いため、極力行列演算で対応すること。また、引数と戻り値の仕様は守ってください。

- normalization : ある配列をユーザが指定する値に正規化
- getLissajous : ユーザが指定した位相・周期のリサージュ曲線のXY波形を出力
- getLissajousMovie : リサージュ図形上を移動する円形動画とそのXY波形を出力
- deprocess_img : 任意の範囲を持つ配列をRGB画像(0-255)に正規化。戻り値の配列が、0または255の範囲外にならないようにクロップ処理を追加すること。

検証: データの読み込み

`dplm/test/utils/data_test.py`を使って、作成した各関数をテストする。はじめにテストプログラムが保存されているフォルダへ移動。

```
$ cd ~/work/dplm/test/utils/
```

以下を実行し、エラーなく実行されるか確認。1行目でエラーが発生した場合、「0_setup.md : 4. Windows-WSL間のフォルダパスの作成」がうまくいっていない可能性あり。

```
$ python3 data_test.py
[INFO] MNIST dataset の読み込みとデータの確認
[Org_data] shape: (60000, 28, 28)
[Org_data] min=0, max=255
```

検証1: normalization

data_test.pyの20行目から32行目をコメントアウトして実行。以下のように表示されればOK。

```
$ pythond data_test.py
[INFO] 課題1-1 normalization関数のテスト
[Norm_data] shape: (60000, 28, 28)
[Norm_data] min=0.1, max=0.9
Org_dataとshapeが変わらず、でもデータの最大・最小値が指定値通りであればOK。
```

検証2: getLissajous

data_test.pyの38行目から61行目をコメントアウトして実行。matplotlibが出力した図が、[この図](#)と同じであればよい。

```
$ pythond data_test.py
[INFO] 課題1-2 getLissajous関数のテスト
以下リンクの図と同じになればOK。
http://www.ne.jp/asahi/tokyo/nkgw/www_2/gakusyu/rikigaku/Lissajous/fig-1.gif
```

検証3: getLissajousMovie

data_test.pyの67行目から82行目をコメントアウトして実行。以下のように表示されればOK。

```
[INFO] 課題1-3 getLissajousMovie関数のテスト
img size: (120, 64, 64, 3)
img min=-0.8999999761581421, max=0.8999999761581421
seq size: (120, 2)
```

検証4: deprocess_img

data_test.pyの88行目から112行目をコメントアウトして実行。フォルダ (`dplm/test/output`) にリサージュ曲線の画像 (01_data_test_Lissajous.png) と動画 (01_data_test_LissajousMovie.mp4) が保存されていればOK。

```
[INFO] 課題1-4 deprocess_img関数のテスト
Before
img min=-0.8999999761581421, max=0.8999999761581421
After
img min=0, max=255
```