

Pythonではじめる教師なし学習

11章 深層信念ネットワークを用いた特徴量検出

1116 17 9036

山口真哉

やること

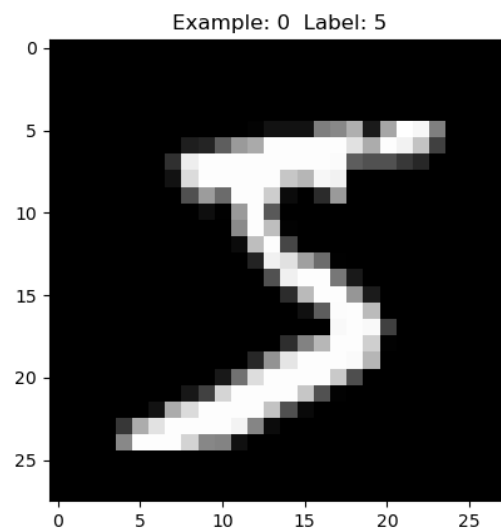
- 深層信念ネットワーク (DBN)
- DBNを用いて画像を大量に生成し, 分類問題を解く

モチベーション

- ラベル付きのデータはあるが、数が少ない場合がある.
- そのような場合に似たようなデータを生成して分類する時にかさ増ししたい.

データの準備

- mnist を使う.
- train size : 50000, vaild size : 10000, test size : 10000 に分割する.
- 状況を再現するためにtrainは5000番目までのみ使用する



Case 1 (LightGBM only)

- LightGBMを使用し, 10種類の分類問題を解く
- ハイパーパラメータは右の通り
- trainデータの数5000である.

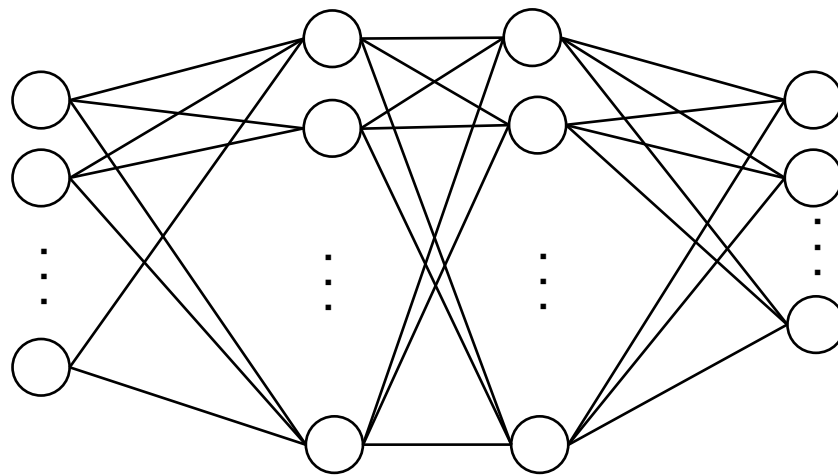
~結果~

- 正解率は94.37%になった.
- (ちなみにtrainデータを削らないと97.97%になった.)

```
params_lightGB = {  
    'task': 'train',  
    'num_class': 10,  
    'boosting': 'gbdt',  
    'objective': 'multiclass',  
    'metric': 'multi_logloss',  
    'metric_freq': 50,  
    'is_training_metric': False,  
    'max_depth': 4,  
    'num_leaves': 31,  
    'learning_rate': 0.1,  
    'feature_fraction': 1.0,  
    'bagging_fraction': 1.0,  
    'bagging_freq': 0,  
    'bagging_seed': 2018,  
    'verbose': -1,  
}
```

Case 2 (DBN)

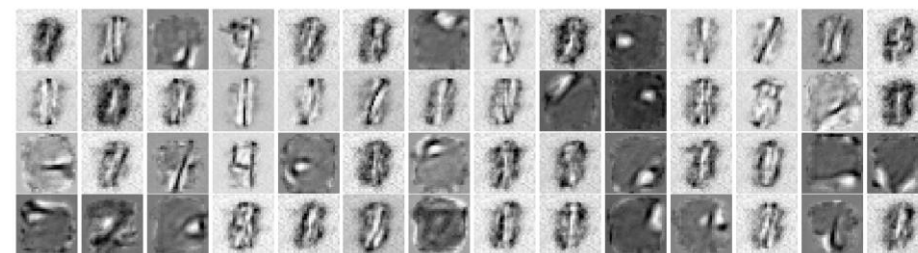
- DBN ... RBMを重ねたもの.
- 今回は1-3層で両方向的な学習をして,
4層目は構成したRBMの重みとバイアスを再構成誤差が最小になるように更新する.
- この学習を20回して, 学習してできた画像たちをtrain2データとする.
- LightGBMを使って, train2データを学習する.



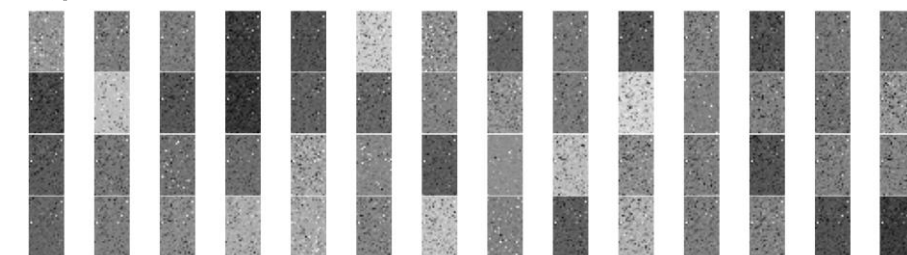
Case 2 (DBN)

- 右の画像は各層が学習した特徴量
- 層が深くなるにつれ抽象的な特徴を学習している.

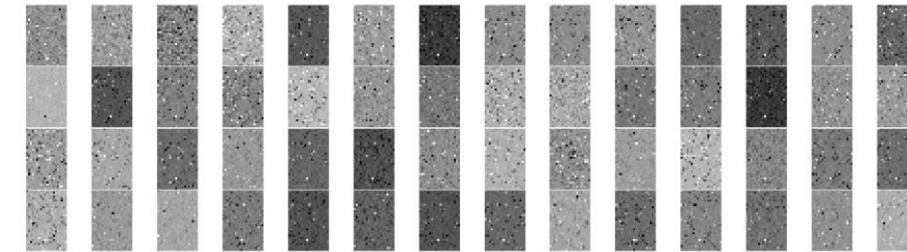
0層目



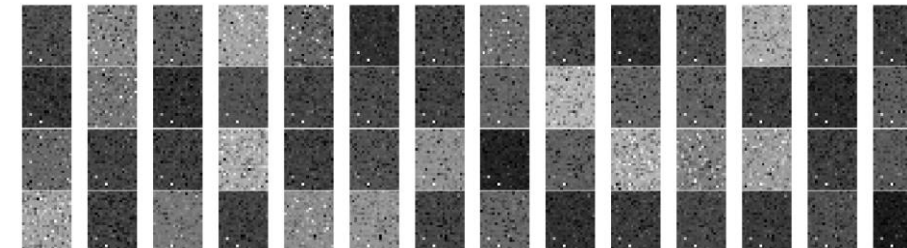
1層目



2層目

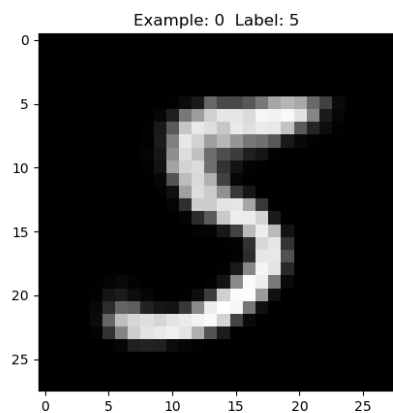


3層目

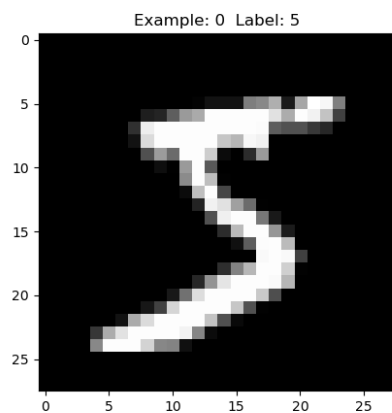


Case 2 (生成された画像)

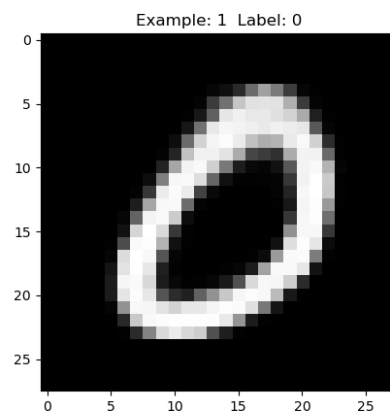
- うまく画像を生成できていることがわかる.
- 一番下の画像は同じ画像から生成した画像だが異なる画像になっている.



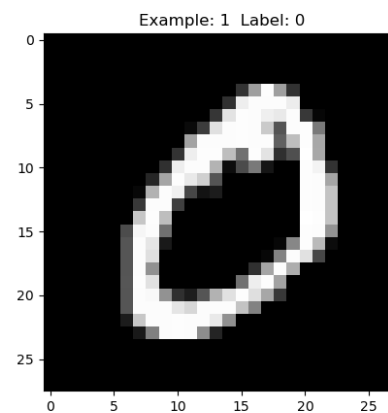
train後



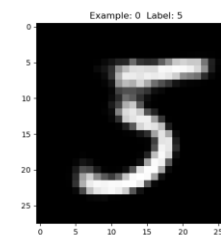
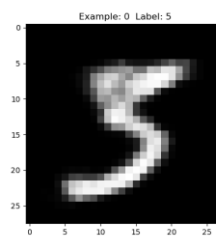
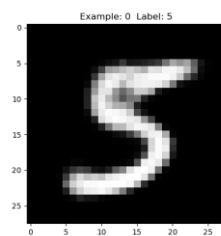
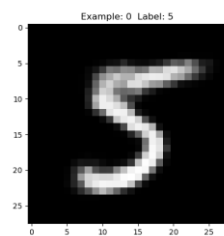
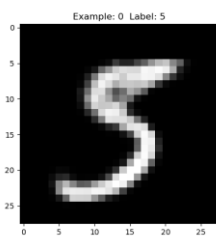
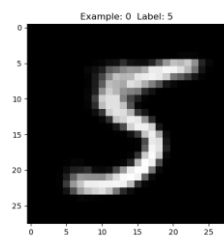
mnist



train後



mnist



Case 2 (結果)

- 正解率は95.51%になった.
- LightGBMのみの94.37%より良い結果となった.

まとめ

- RBMを積み重ねたDBNを勉強した.
- 画像を生成することで教師ありの手助けができる.
- mnistの例だとデータをかさ増しすることでうまくいった.