

自己組織化マップ Self Organizing Map: SOM

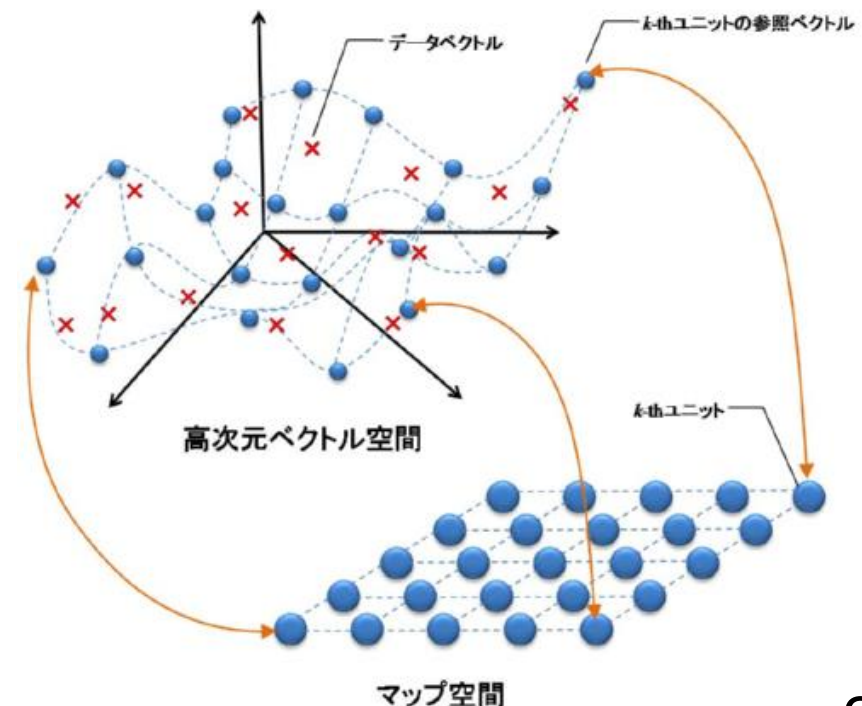
M1 Kinoshita Takahiro

概要

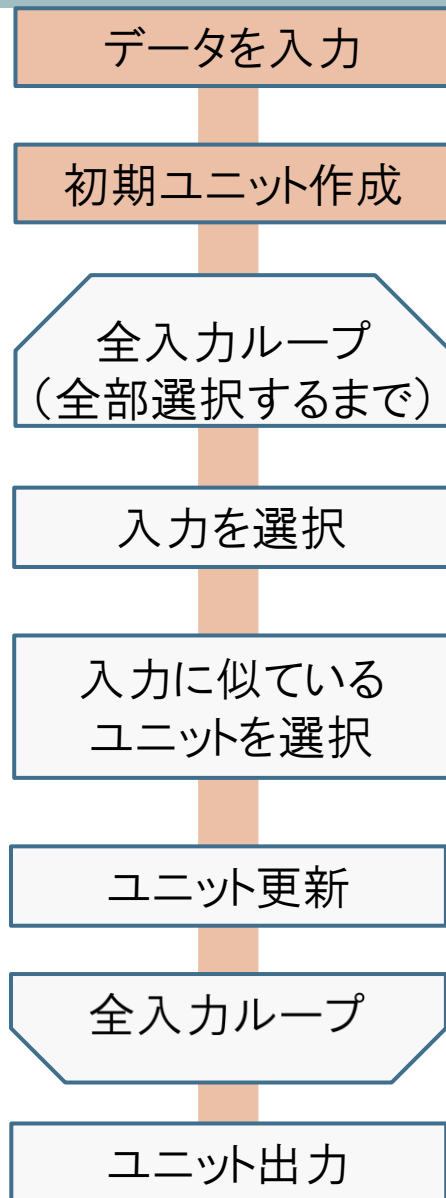
- 自己組織化マップ(SOM)とは何か
 - 教師なしで高次元な関係性を低次元で表現するNN
- SOMの学習アルゴリズム
- SOMのデモ
 - 色のSOM
 - 教師なしで色の分布が分かれていくことを実演
 - MNISTのSOM
 - 教師なしで数字の分布が分かれていくことを実演

自己組織化マップ(SOM)とは何か

- 教師なしで高次元データをデータ間の関係を保ったまま任意の次元へと写像できるNNの一種
 - 高次元空間での関係性 \Leftrightarrow 平面空間での関係性
 - 教師なしでデータ同士をわけするため自己組織化と呼ばれる
- ユニット自体は高次元データのコピーというわけではないので（高次元データ数 \neq ユニット数）生成モデルとも考えられる



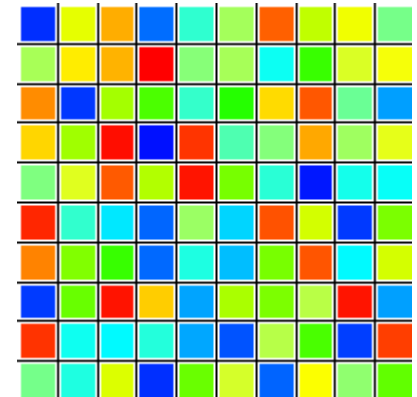
SOMの学習アルゴリズム



- 入力するデータ例: 色 (K, 3)
 - 3変数を持つデータ
- 初期ユニット ($N * N, 3$)
 - ランダムに初期化したベクトル
 - 学習後の出力に値する

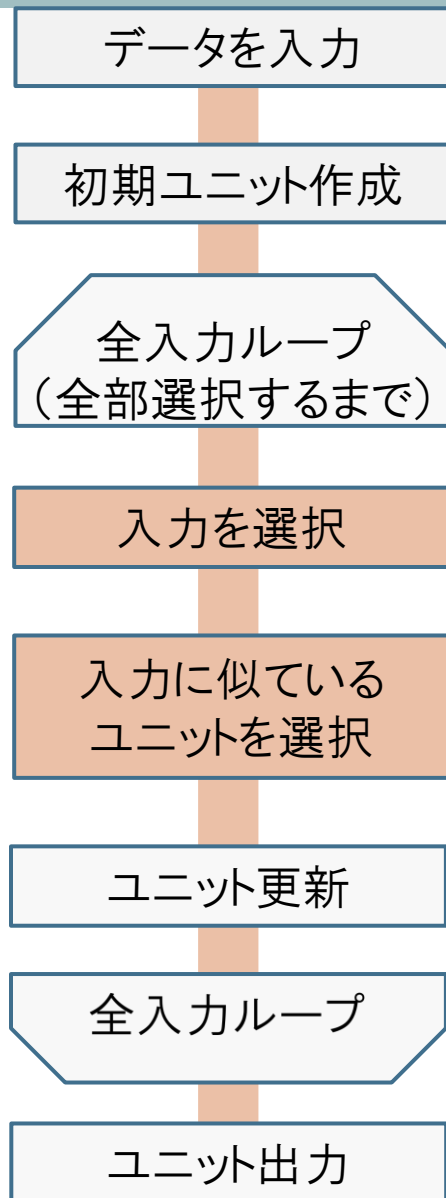


↑
入力データ

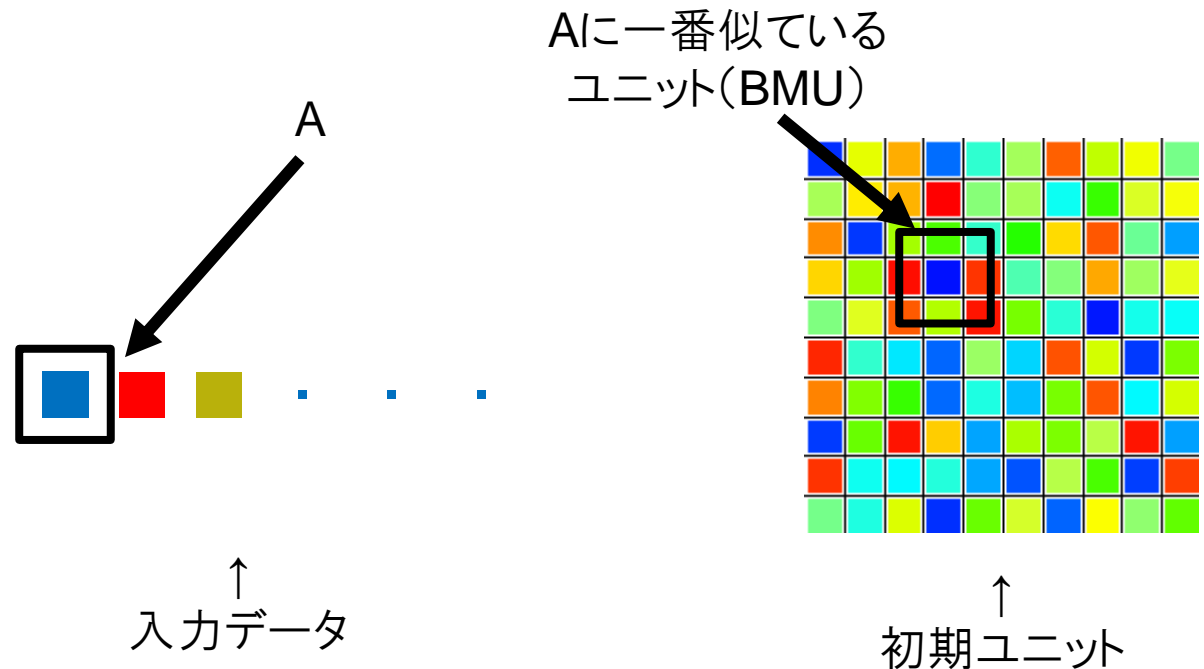


↑
初期ユニット

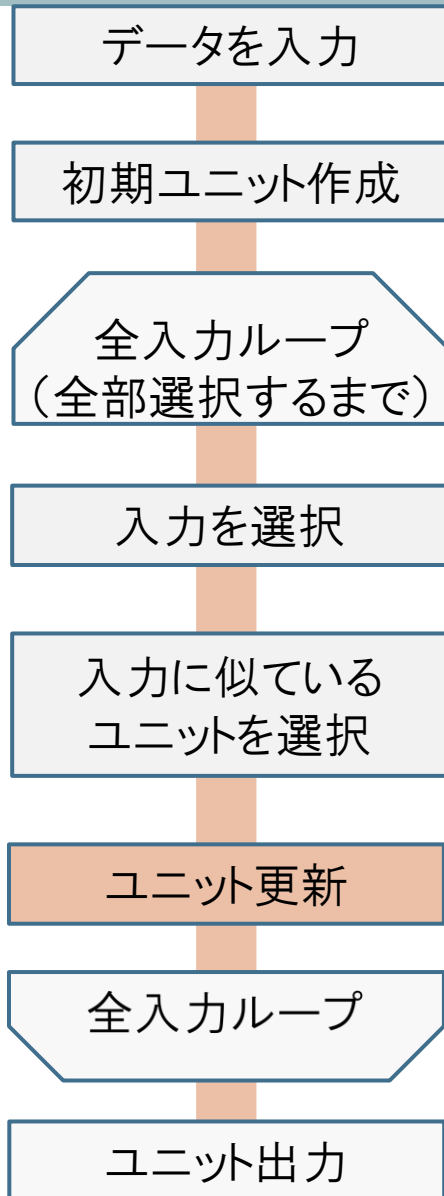
SOMの学習アルゴリズム



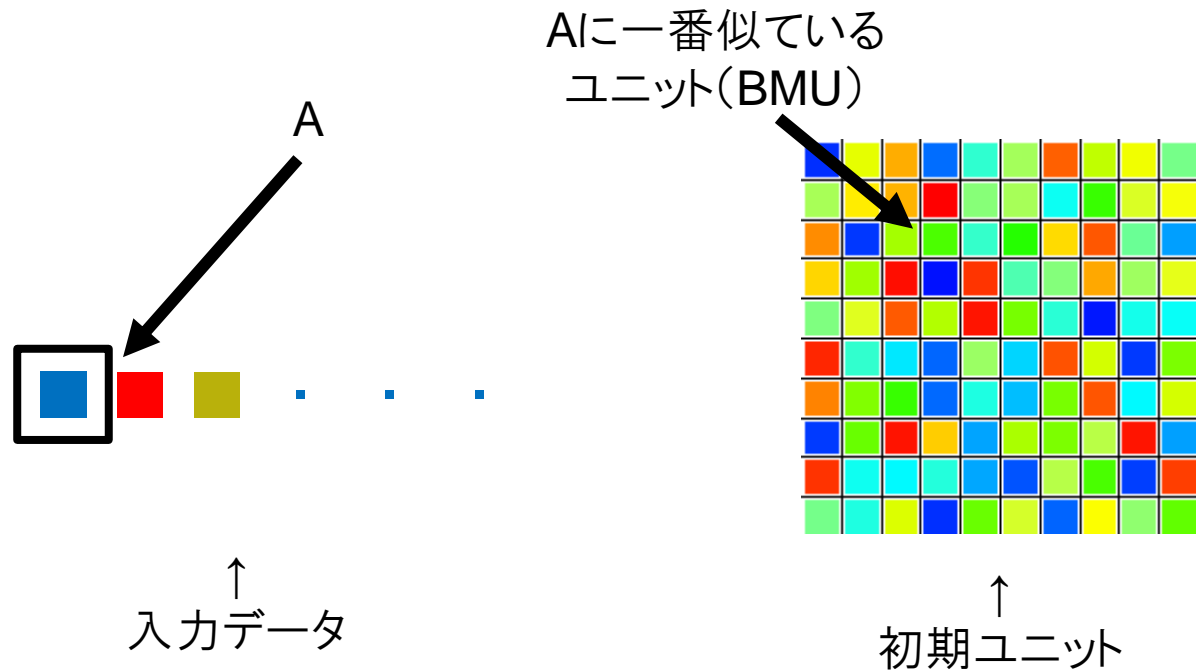
- 入力データから1つ選択(Aとする)
- Aと最も似ているユニットを探索(BMUとする)
 - 3次元におけるユークリッド距離で計算



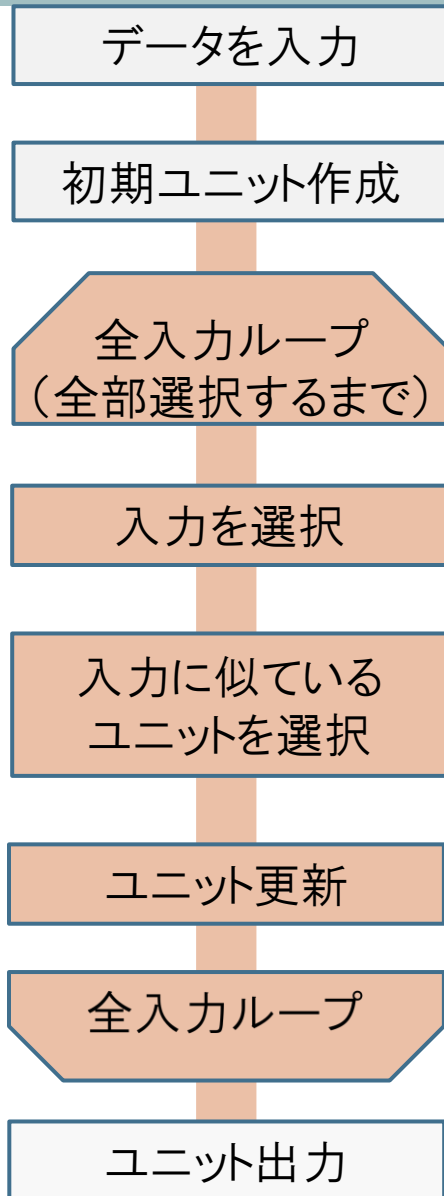
SOMの学習アルゴリズム



- BMUとその近傍ユニットを更新する
 - 近傍であればあるほど大きく更新する
 - 学習が進むほど近傍範囲は小さくなるように



SOMの学習アルゴリズム



- ユニット更新を全入力分行い学習は終了する

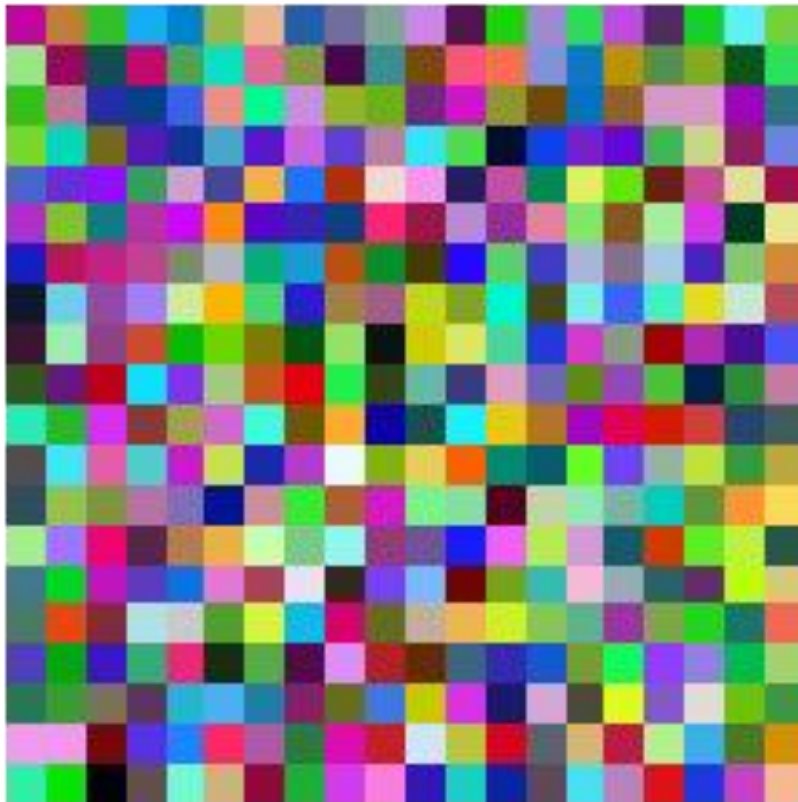
スタート

SOMのデモ(色)

- 入力: `np.random.rand(10000, 3)`
- ユニット: (20*20, 3)



initialize



iter:000009950




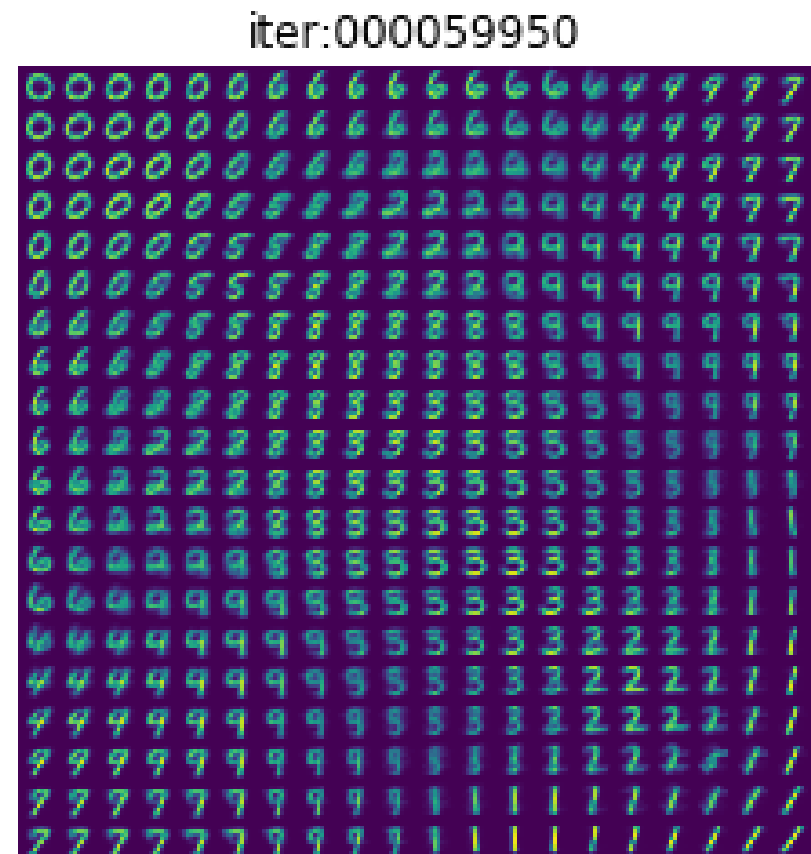
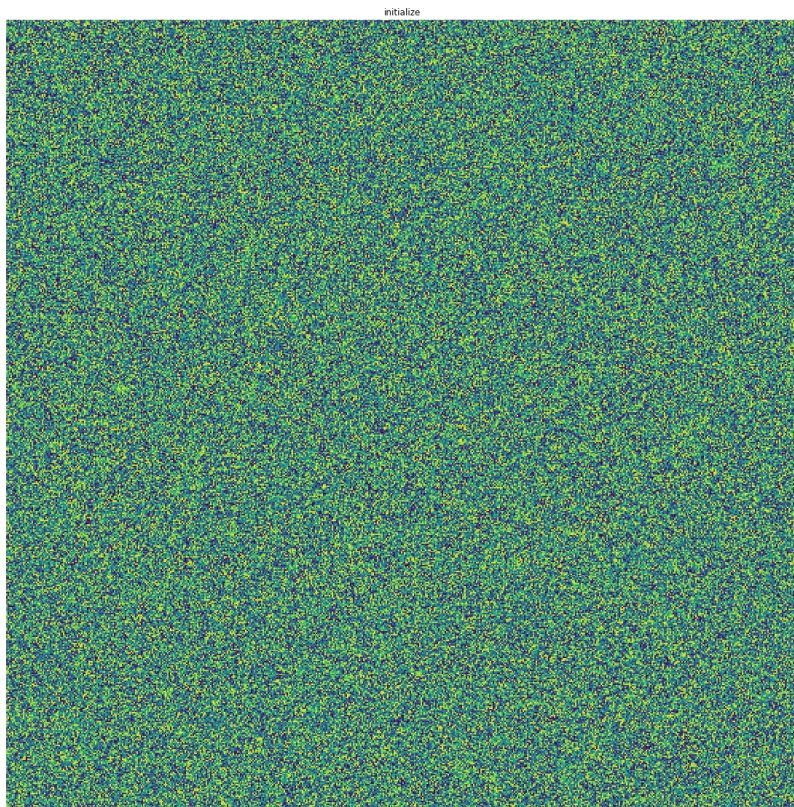
SOMのデモ(色)

- 入力: `np.random.rand(10000, 3)`
- ユニット: (20*20, 3)




SOMのデモ(MNIST)

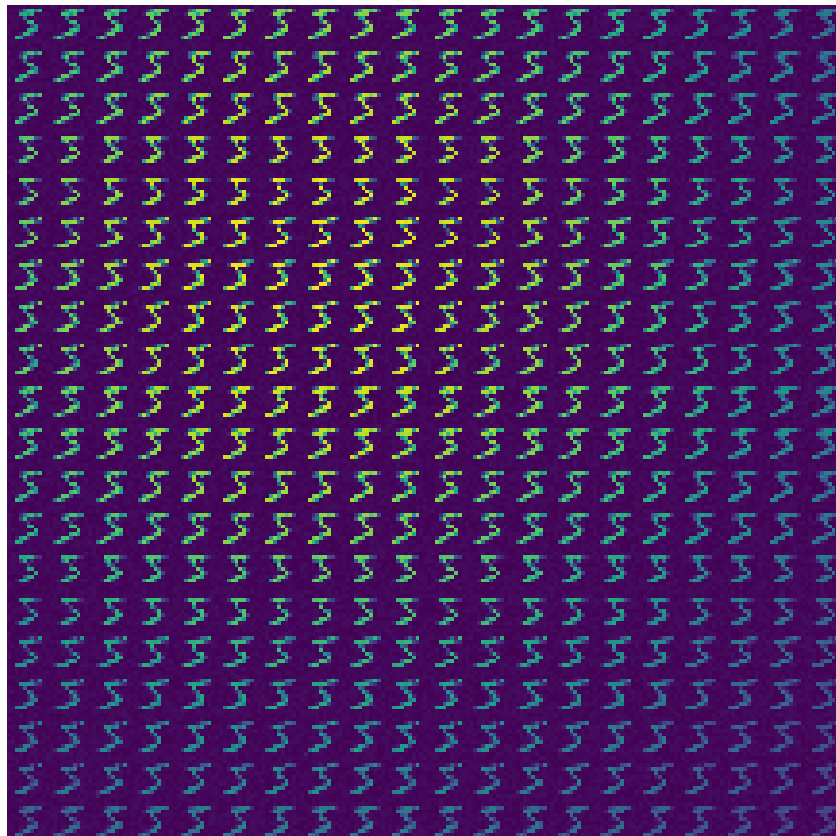
- 入力: keras.datasets.mnist (60000, 28*28) 
- ユニット: (20*20, 28*28)



SOMのデモ(MNIST)

- 入力: keras.datasets.mnist (60000, 28*28) 
- ユニット: (20*20, 28*28)

iter:0000000000



まとめ

- 自己組織化マップ(SOM)とは何か
 - 教師なしで高次元な関係性を低次元で表現するNN
- SOMの学習アルゴリズム
 - ユニットを入力に近づけていく処理を繰り返し行う
- SOMのデモ
 - 色のSOM
 - 教師なしで色の分布が分かれていくことを実演
 - MNISTのSOM
 - 教師なしで数字の分布が分かれていくことを実演

参考資料

- 自己組織化特徴マップ(SOM) - 静岡理科大学
 - <http://www.sist.ac.jp/~kanakubo/research/neuro/selforganizingmap.html>
- SOM tutorial part 1
 - <http://www.ai-junkie.com/ann/som/som1.html>
- 子供でもわかる「自己組織化マップ」
 - http://gaya.jp/spiking_neuron/som.htm
- Kohonenの自己組織化マップをpythonで実装 – technocrat
 - <http://technocrat.hatenablog.com/entry/2015/02/12/014557>
- Pythonで自己組織化マップ NumPy版 - Qiita
 - https://qiita.com/T_Shinaji/items/609fe9aabd99c287b389