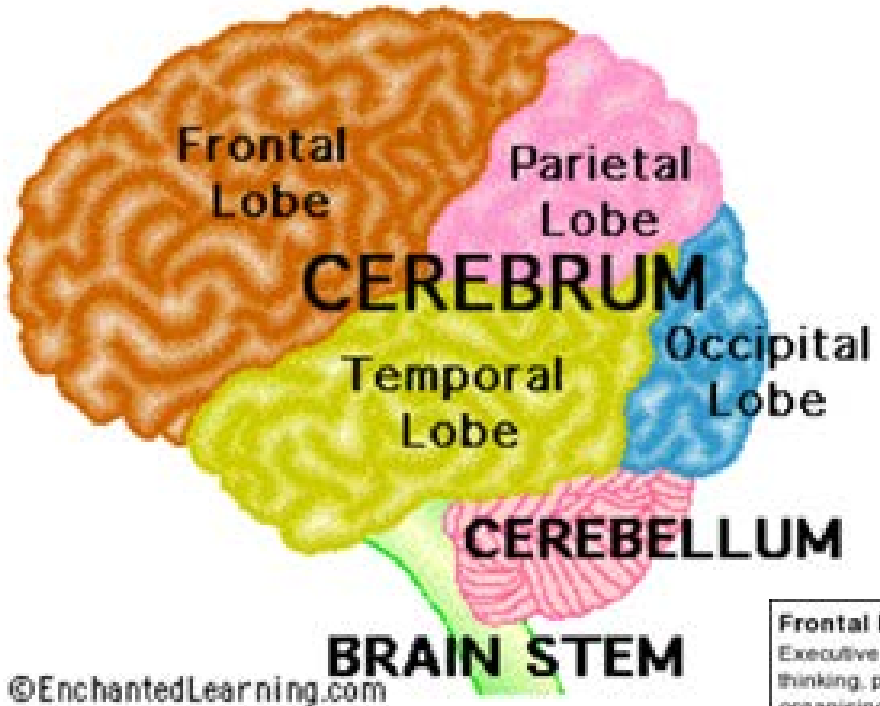
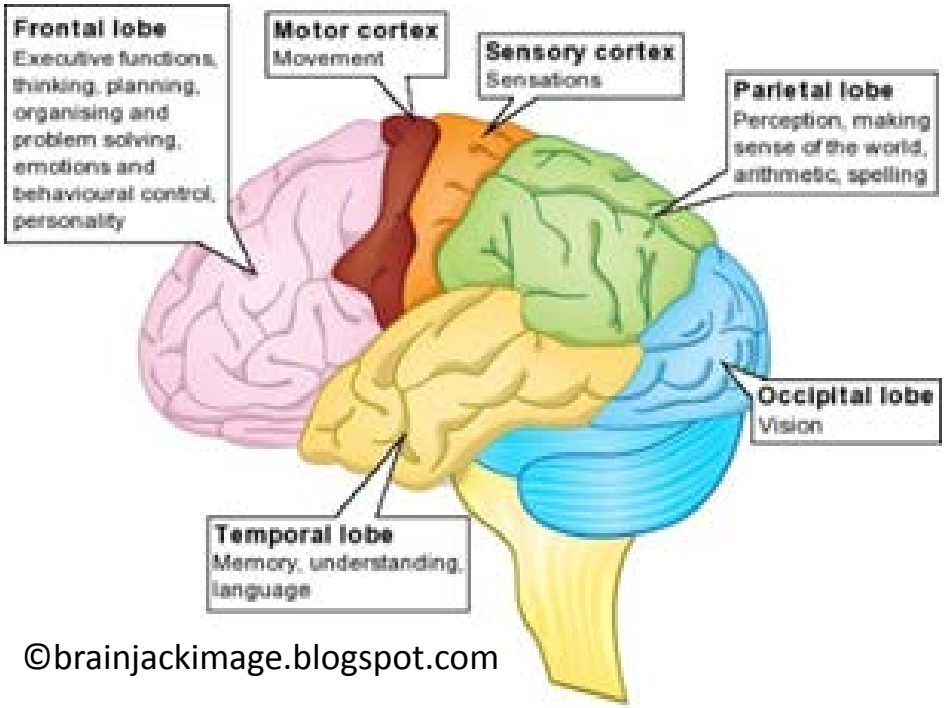
ニューラルネットワークのプログラム

2016年4月22日

1. Brief Introduction of the Brian
2. Brief Introduction of Four NN Algorithms

Brain Structure and Functions

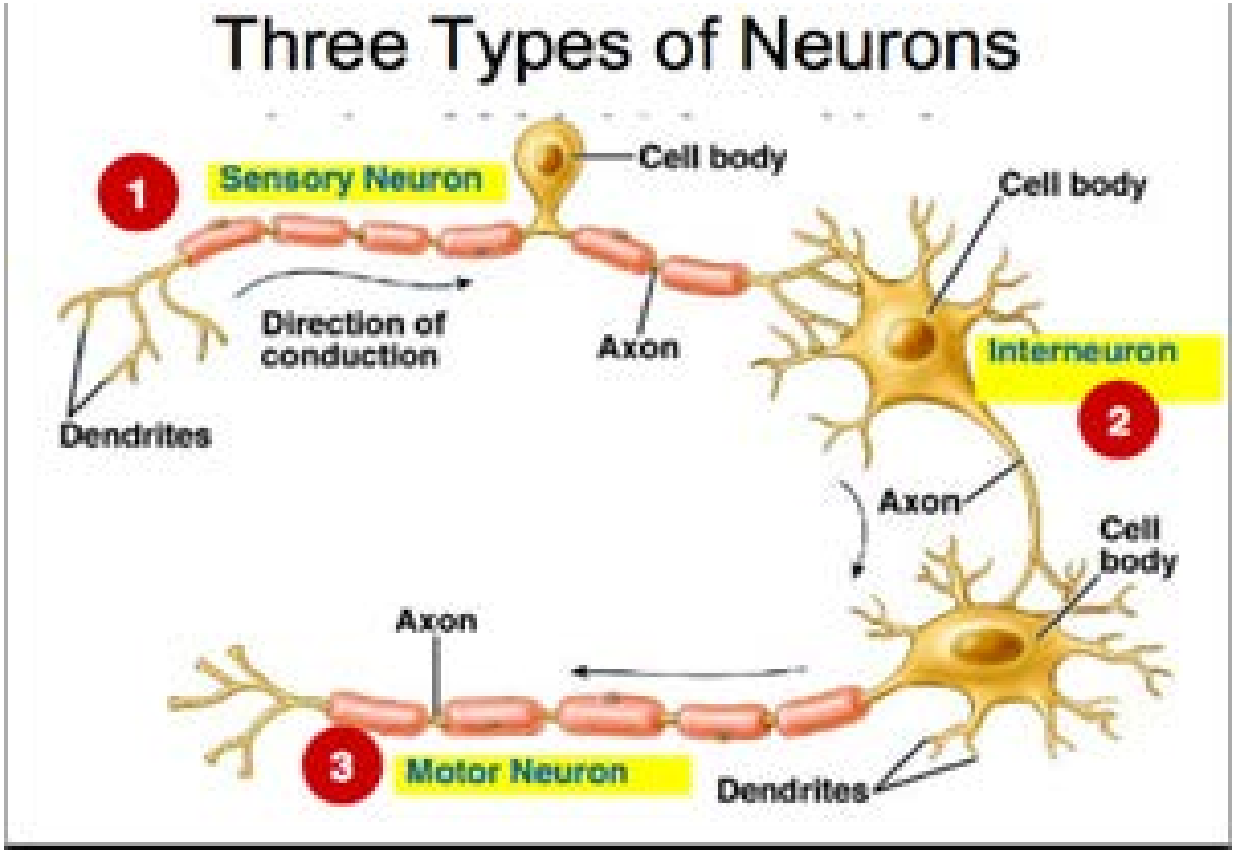

Frontal Lobe
Parietal Lobe
CEREBRUM
Occipital Lobe
Temporal Lobe
CEREBELLUM
BRAIN STEM
©EnchantedLearning.com


Frontal lobe
Executive functions, thinking, planning, organising and problem solving, emotions and behavioural control, personality
Motor cortex
Movement
Sensory cortex
Sensations
Parietal lobe
Perception, making sense of the world, arithmetic, spelling
Occipital lobe
Vision
Temporal lobe
Memory, understanding, language
©brainjackimage.blogspot.com

Neurons

There are three main types: sensory neurons, motor neurons, and interneurons.

- The sensory neurons respond to touch, sound, light and all other stimuli.
- The motor neurons receive signals from the brain and spinal cord.
- The interneurons connect neurons to other neurons within the same region of the brain, or spinal cord in neural networks.

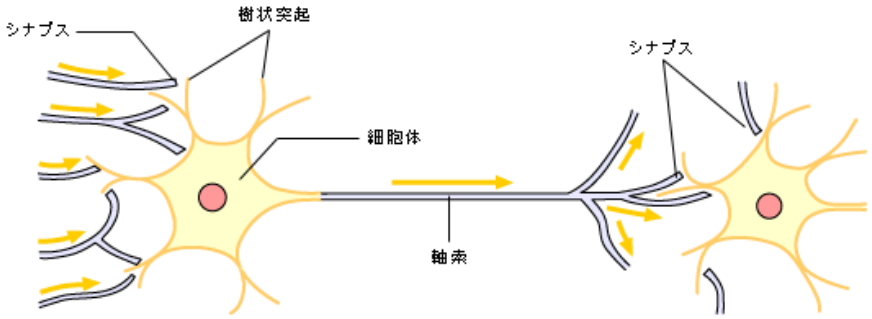


脳の構成

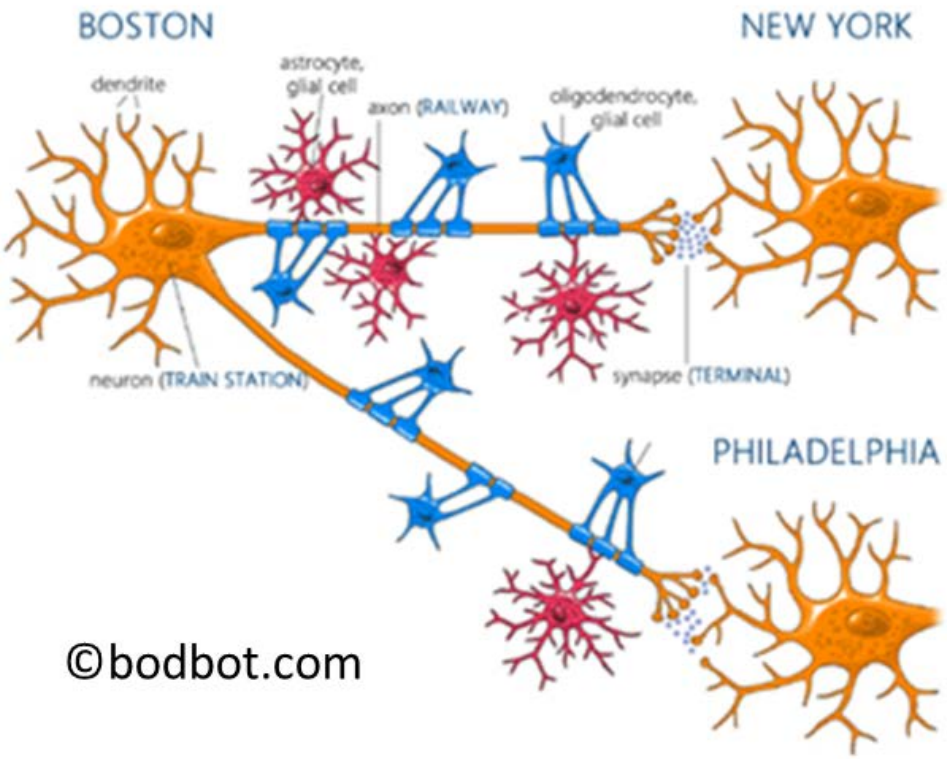
人間の脳には140億個のニューロン（神経細胞）が存在し、各ニューロンは8000個のシナプス（ニューロン同士の接合部）を持つ。ニューロンに入力となる刺激が入ると、活動電位が発生し、ほかのニューロンに情報を伝える。

記憶は、シナプス間の情報伝達の量に変化が生じることで形成されると考えられている。

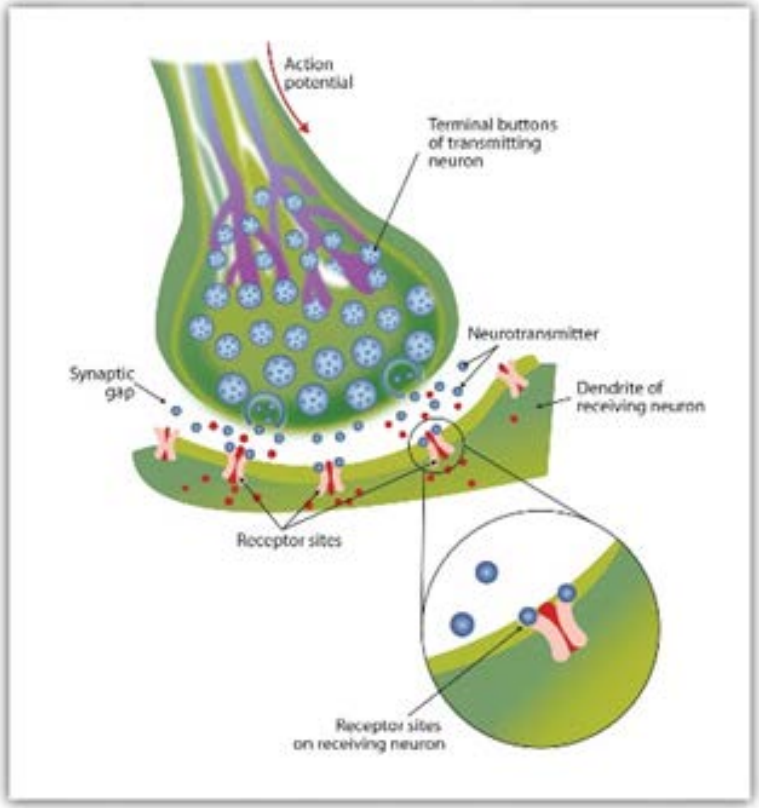
破壊されたニューロンの再生や、新たなシナプスの形成はないものとしている。



http://j-net21.smrj.go.jp/develop/digital/entry/001-20081119-10.html



©bodbot.com



©www.flatworldknowledge.com

Memory

The synapse plays a role in the formation of memory.

- As neurotransmitters activate receptors across the synaptic gap
- The connection between the two neurons is strengthened if both neurons are active, which is thought to result in the storage of information.
- This synaptic strengthening process is known as long-term potentiation (LTP), i.e. the form of neural plasticity occurred in the hippocampus.
- It is said that short-term memory is supported by transient patterns of neuronal communication, dependent on regions of the frontal lobe and the parietal lobe
- While long-term memory is maintained by more permanent changes in neural connections widely spread throughout the brain.
- The hippocampus is essential to the consolidation of information from short-term to long-term memory.

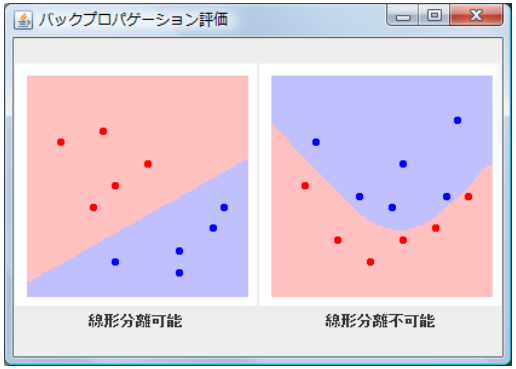
Brief Introduction of Four Neural Network Algorithms

ニューラルネットのアルゴリズム

様々なものが存在するが、4つを紹介予定

誤差逆伝播法

- 多層ネットワーク



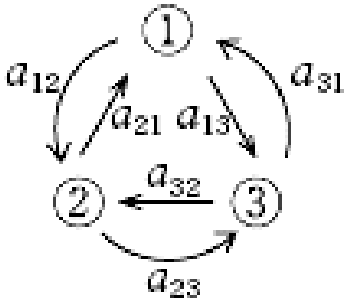
http://d.hatena.ne.jp/nowokay/20080701/1214915017

自己組織化写像

- 自動分類

ヘッブの法則

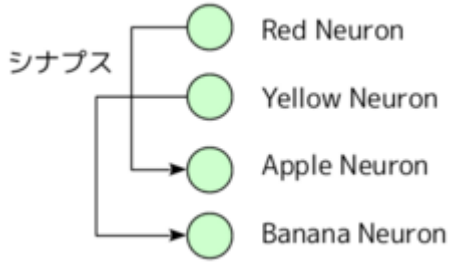
- 人間の脳活動



http://gaya.jp/spiking_neuron/matrix.htm

ホップフィールド

- 連想記憶



http://enakai00.hatenablog.com/entry/20150108/1420719651

プログラムの紹介

- ヘッブの法則（hebb.py）
- ホップフィールドネットワーク(hopfield.py)
- 自己組織化写像（som.py）
- 誤差逆伝搬法(BackPropagation.py) ➡ 起動しない様なので、保留

ヘッブの法則（hebb.py）

- 人間の脳の動きに近い。
  何度も同じことを反復学習することで、定着する。望んでいるか望んでいないかにも関わらず、繰り返した行動が習慣化する。
  という脳の働きをモデル化したプログラム。

ヘッブの法則（hebb.py）

- 実行結果

```
[1, 1, -1]を覚えました。
[1, 1, -1]を覚えました。
[1, 1, -1]を覚えました。
[1, 1, -1]から、[ 1  1 -1]を思い出しました！
[1, 1, 0]から、[ 1  1 -1]を思い出しました！
[1, -1, 1]を覚えました。
[1, -1, 1]を覚えました。
[1, 1, -1]から、[ 1  1 -1]を思い出しました！
[1, -1, 1]から、[-1 -1  1]を思い出しました！
```

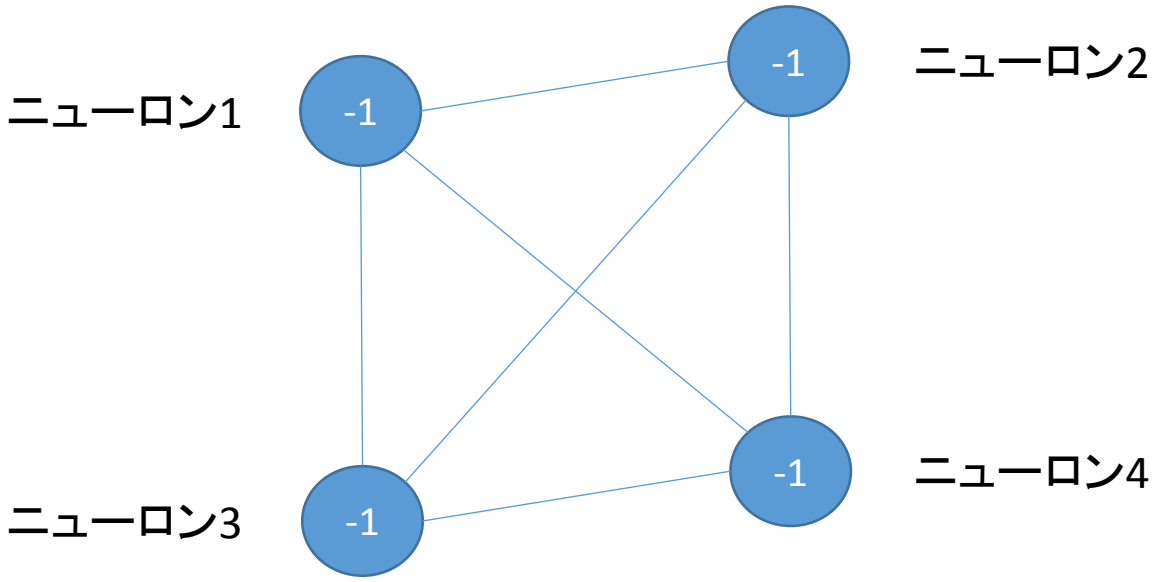
ヘッブの法則（hebb.py）

```
[1, 1, -1]を覚えました。
[1, 1, -1]を覚えました。
[1, 1, -1]を覚えました。
[1, 1, -1]から、[ 1  1 -1]を思い出しました！
[1, 1, 0]から、[ 1  1 -1]を思い出しました！
[1, -1, 1]を覚えました。
[1, -1, 1]を覚えました。
[1, 1, -1]から、[ 1  1 -1]を思い出しました！
[1, -1, 1]から、[-1 -1  1]を思い出しました！
```

[1, 1, -1]を、まず3回覚える。そうすると、[1,1,-1] を見ると、[1,1,-1]をすぐに思い出すことができる。次に、この状態のまま[1,1,0]を見ると、これまでに学習したものは[1,1,-1]だけのため、これが思い出されてしまう。
この後に、新たに[1,-1,1]を2回覚える。その状態で、また[1,1,-1]を見ると、[1,1,-1]を思い出すことができた。これは、[1,1,-1]を3回覚えて習慣化したため。
さらに、[1,-1,1]を見せると2回覚えているはずだが思い出せない。学習が足りないのかもしれない・・・というプログラム。

ホップフィールドネットワーク(hopfield.py)

- John Hopfieldが提案した相互結合型ネットワーク
  相互結合型ネットワークは入力層、出力層がなく、それぞれのユニットは自分以外の全てのユニットと結合している。
- ニューロン4個の場合



ホップフィールドネットワーク(hopfield.py)

- このプログラムでは、「リンゴ」、「バナナ」、「赤」、「黄色」に反応するニューロンを例としての学習を行う。各ニューロンの初期状態は無刺激で-1となっていて、ニューロン間の結びつきは全て0である。

```
初期状態
Apple    -1
Banana   -1
Red      -1
Yellow   -1
Name: Activation, dtype: int64
output  Apple  Banana  Red  Yellow
input
Apple       0       0    0       0
Banana      0       0    0       0
Red         0       0    0       0
Yellow      0       0    0       0
```

ホップフィールドネットワーク(hopfield.py)

- 「赤いリンゴ」、「黄色いバナナ」の順に見た後のニューロン間の結合具合。「リンゴ」と「赤い」のニューロンの結びつき、「バナナ」と「黄色い」のニューロンの結びつきが強くなる。逆に、「赤」と「バナナ」や「黄色」と「リンゴ」は結びつきが弱まる。

赤いリンゴを見せる

output / input	Apple	Banana	Red	Yellow
Apple	0	-1	1	-1
Banana	-1	0	-1	1
Red	1	-1	0	-1
Yellow	-1	1	-1	0

黄色いバナナを見せる

output / input	Apple	Banana	Red	Yellow
Apple	0	-2	2	-2
Banana	-2	0	-2	2
Red	2	-2	0	-2
Yellow	-2	2	-2	0

ホップフィールドネットワーク(hopfield.py)

- この学習状態で「赤い」のニューロンを刺激する（-1から1にする）と、それに反応して「リンゴ」が1になって反応していることがわかる。
  つまり、「赤」から「リンゴ」を連想することができた。

```
赤を刺激
Apple    -1
Banana   -1
Red       1
Yellow   -1
Name: Activation, dtype: int64
赤といえば？
Apple     1
Banana   -1
Red       1
Yellow   -1
Name: Activation, dtype: int64
```

ホップフィールドネットワーク(hopfield.py)

- 次に、存在するかは分からないが、引き続き「黄色いリンゴ」を見せた場合の学習状態はこのようになる。

output	Apple	Banana	Red	Yellow
input				
Apple	0	-2	2	-2
Banana	-2	0	-2	2
Red	2	-2	0	-2
Yellow	-2	2	-2	0

（赤いリンゴ、黄色いバナナを見た時の状態）

黄色いリンゴを見せる

output	Apple	Banana	Red	Yellow
input				
Apple	0	-3	1	-1
Banana	-3	0	-1	1
Red	1	-1	0	-3
Yellow	-1	1	-3	0

ホップフィールドネットワーク(hopfield.py)

- 「リンゴ」が刺激された状態になっているため、無刺激状態にした上で、また「赤」のみ刺激状態にすると、先ほど「黄色いリンゴ」を見たが、それでも「リンゴ」を連想する結果となる。

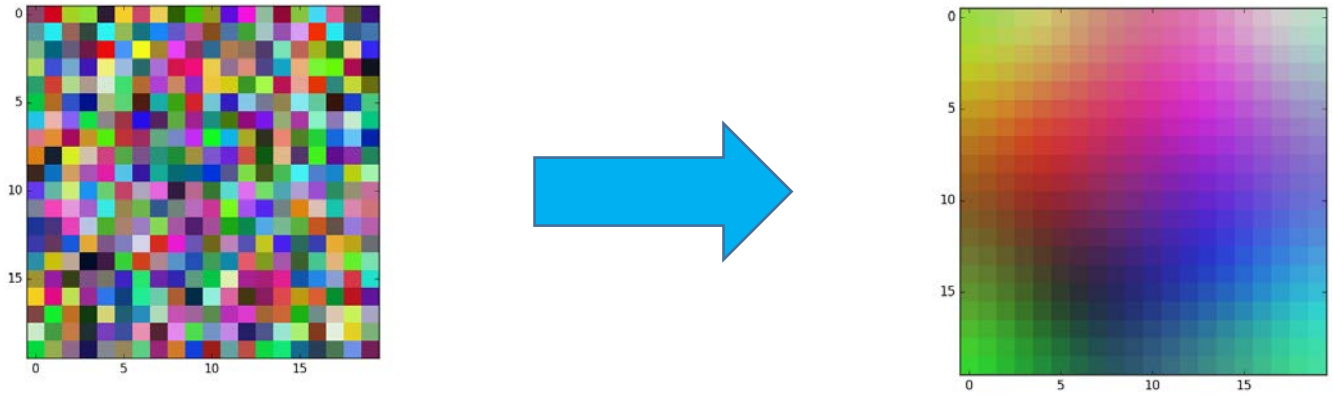
```
赤を刺激、リンゴを刺激していない状態に
Apple    -1
Banana   -1
Red       1
Yellow   -1
Name: Activation, dtype: int64
赤といえば？
Apple     1
Banana   -1
Red       1
Yellow   -1
Name: Activation, dtype: int64
```

自己組織化写像（som.py）

- R、G、Bの3次元で表される「色」等、ベクトル表記される多次元のデータを、マップ上の座標に近いもの同士で集めることが可能。
- 勝手に学習して分類してくれる。
- 色の場合の学習をサンプルプログラムとして紹介。

自己組織化写像（som.py）

- 実行終了まで少しかかる（final.pngが作られるまで待つ）。
- 最初はランダムだった配色が、回数を重ねるごとに少しずつランダム性が失われ、マス目がくっきりする。
- 似たような色が近くに並ぶが、これはコンピュータが元々類似色を知っていたわけではない



自己組織化写像（som.py）

- ある色を選択し、その色と一番近いものを図の中から探す。
  一番近かった色のマスとその周囲1マスに選択した色を混ぜる。
  この工程を繰り返すことで、図中の色の分布が似たような色で固まるようになる。
- デモ
  http://gaya.jp/spiking_neuron/som.htm

今後

- 各アルゴリズムの具体的な解説は、次回以降順次行っていく。
- 次回から2、3回はヘップの法則についての内容の予定。