

# チャーノフの顔グラフ

青木繁伸

2020年3月17日

## 1 目的

チャーノフの顔グラフを描く。

顔の造作を表すのは 18 個の要素で、実際のデータに対応づけることにより、データの大小が顔の表情に表れ、似たデータは似た顔になる。

## 2 使用法

```
import sys
sys.path.append("statlib")
from plot import face_graph
face_graph(x, size=480)
```

データの対応づけのための関数

```
import sys
sys.path.append("statlib")
from plot import face_data
face_data(d, pos)
```

### 2.1 引数

<code>x</code>	顔の要素 18 個の値を持つベクトル
<code>size</code>	顔の大きさ
<code>d</code>	データ行列
<code>pos</code>	データ行列の変数を、チャーノフの顔グラフを描くために必要な 18 個のパラメータと対応づける情報

各要素は、データ行列の何列目に対応するかを表す。同じ変数が重複してもよい。 $i$  番目の要素は、第  $i$  パラメータに割り当てる変数の番号とする。変数を割り当てないパラメータには 0 を指定しておく。

## 2.2 戻り値の名前

face\_data の戻り値は、元のデータの行数 × 18 列のデータ行列

## 2.3 顔の要素 18 個

18 個の要素について、割り当てられる値によりどのような顔になるかを以下の図に示す。各列において、順に最小値、通常値、最大値の場合を示す。

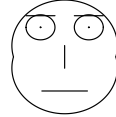
$x[0] = 0.2$



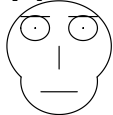
$x[0] = 0.5$



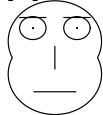
$x[0] = 0.8$



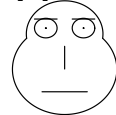
$x[1] = 0.2$



$x[1] = 0.5$



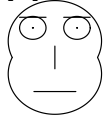
$x[1] = 0.8$



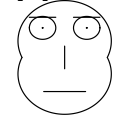
$x[2] = 0.2$



$x[2] = 1.0$

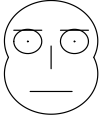


$x[2] = 1.0$

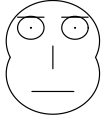




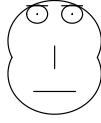
$x[9] = 0.1$



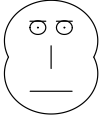
$x[9] = 0.4$



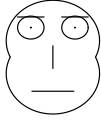
$x[9] = 0.7$



$x[10] = 0.1$



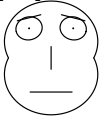
$x[10] = 0.5$



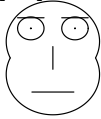
$x[10] = 0.9$



$x[11] = 0.3$



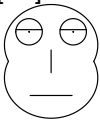
$x[11] = 0.5$



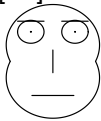
$x[11] = 0.7$



$x[12] = 0.1$



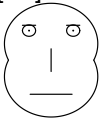
$x[12] = 0.5$



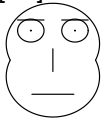
$x[12] = 0.9$



$x[13] = 0.3$



$x[13] = 0.6$



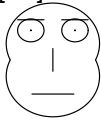
$x[13] = 0.9$



$x[14] = 0.1$



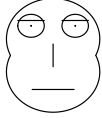
$x[14] = 0.5$



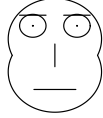
$x[14] = 0.9$



x[15] = 0.1



x[15] = 0.5



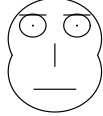
x[15] = 0.9



x[16] = 0.1



x[16] = 1.0



x[16] = 1.0



x[17] = 0.1



x[17] = 0.5



x[17] = 0.9



### 3 使用例

例として、サンプルサイズ 20 の 15 変数データを作る。平均値 50, 標準偏差 10 の正規乱数。

```
import numpy as np
np.random.seed(123)
a = np.random.normal(50, 10, (20, 15))
```

face\_plot 用のデータ行列を作る。

通常は、既存のデータ行列を face\_data で変換する。

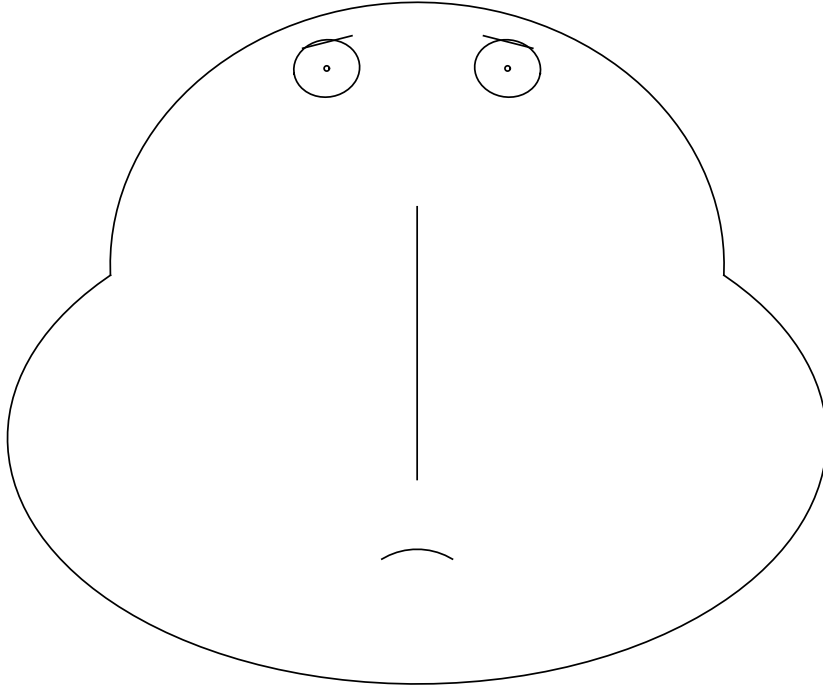
```
# 1 番目の変数をパラメータ 1 に, 2 番目の変数をパラメータ 2 に, パラメータ 3
  には変数は割り当てない, パラメータ 4 には変数 3 ...
pos = [1, 2, 15, 3, 6, 6, 8, 7, 3, 9, 10, 11, 0, 4, 12, 0, 13, 14]

import sys
sys.path.append("statlib")
from plot import face_data

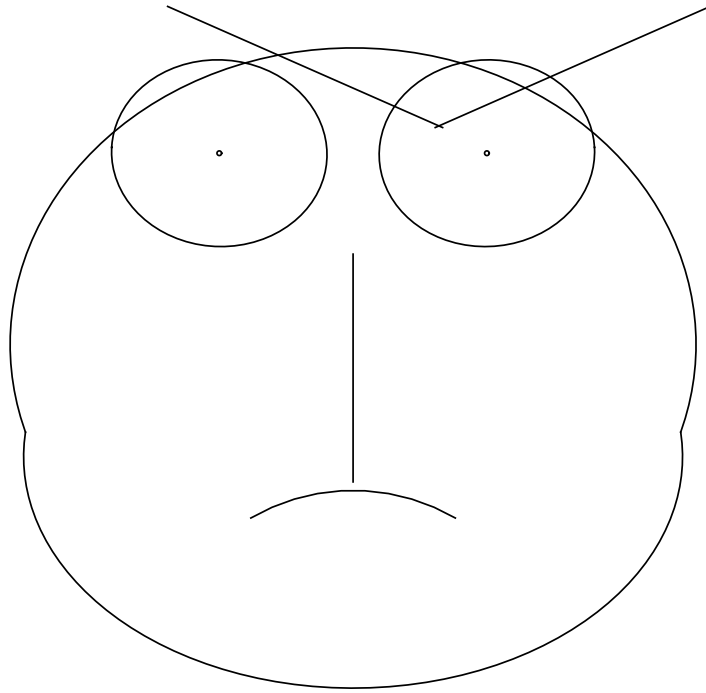
x = face_data(a, pos)
```

```
import sys
sys.path.append("statlib")
from plot import face_graph
```

```
a = face_graph(x[0, :])
```



```
a = face_graph(x[6, :])
```



```
a = face_graph(x[8, :])
```

