

## 実践演習12-5

小規模な映画評価データを使って、行列分解を行います。

ライブラリの読み込み

In [1]:

```
import numpy as np
from sklearn.decomposition import NMF
```

参考ページ(<http://www.guuxlabs.com/blog/2010/09/matrix-factorization-a-simple-tutorial-and-implementation-in-python/>)のデータ例を使用します。縦がユーザ、横が映画、数値が1-5の5段階評価で、0は評価なしです。

In [2]:

```
X = np.array([
    [5, 3, 0, 1],
    [4, 0, 0, 1],
    [1, 1, 0, 5],
    [1, 0, 0, 4],
    [0, 1, 5, 4]
])
```

ここでは、非負値行列分解NMF (<http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.NMF.html>)を使います。NMFはXを非負の行列WとHに分解します。引数n\_componentsは潜在変数の次元数です。

In [3]:

```
model = NMF(n_components = 2)
W = model.fit_transform(X)
H = model.components_
```

空所の値を予測してみます。似たようなユーザを参考に、埋められた値の妥当性を評価してみてください。

In [4]:

```
np.set_printoptions(suppress=True)
np.dot(W, H)
```

Out[4]:

```
array([[5.2558264, 1.99313836, 0.          , 1.45512772],
       [3.50429478, 1.32891458, 0.          , 0.9701988 ],
       [1.31294288, 0.94415991, 1.94956896, 3.94609389],
       [0.98129195, 0.72179987, 1.52759811, 3.0788454 ],
       [0.          , 0.65008935, 2.84003662, 5.21894555]])
```

In [5]:

```
W
```

Out[5]:

```
array([[0.          , 2.37453733],
       [0.          , 1.58321035],
       [1.35650216, 0.59317634],
       [1.06289656, 0.44333929],
       [1.97608595, 0.          ]])
```

In [6]:

```
H
```

Out[6]:

```
array([[0.          , 0.32897828, 1.43720298, 2.6410519 ],
       [2.21341073, 0.83937967, 0.          , 0.61280474]])
```

In [ ]: