t-SNEとUMAP

参考情報

- 書籍）Python による異常検知

座標変換によるマッピングである主成分分析は情報ロスが発生

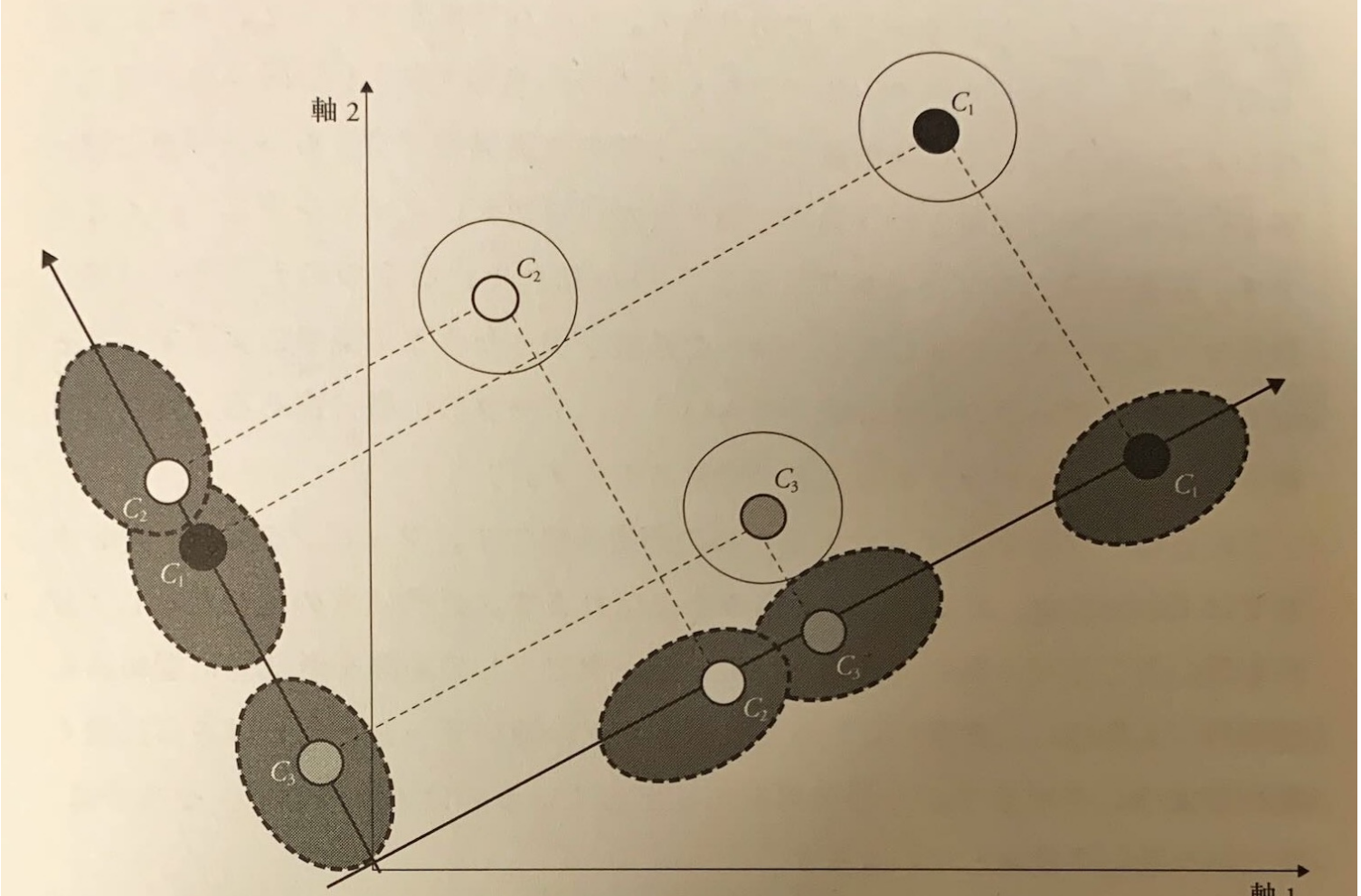
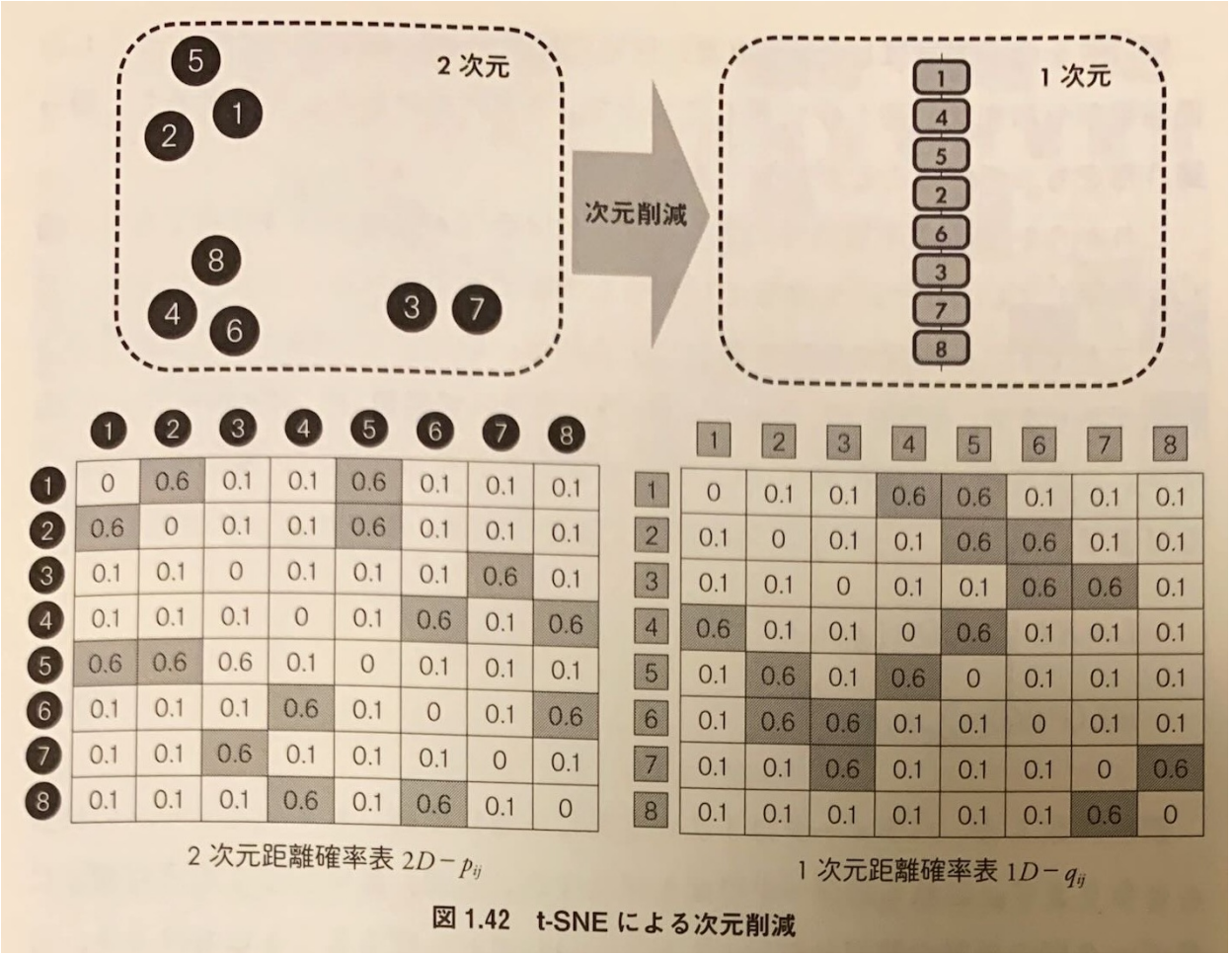


図 1.41　主成分分析の欠点

座標ではなくサンプル間の距離情報を維持してマッピング

- 座標ではなくサンプル間の距離（スカラ）を考える。t分布を使うのでt-SNE
- 圧縮前の次元と圧縮後の次元での距離確率表を求める（下図。2次元から1次元）
- 下記の1次元はランダムに配置した際の距離確率。左と右の距離確率表を近づける



	1	2	3	4	5	6	7	8
1	0	0.6	0.1	0.1	0.6	0.1	0.1	0.1
2	0.6	0	0.1	0.1	0.6	0.1	0.1	0.1
3	0.1	0.1	0	0.1	0.1	0.1	0.6	0.1
4	0.1	0.1	0.1	0	0.1	0.6	0.1	0.6
5	0.6	0.6	0.6	0.1	0	0.1	0.1	0.1
6	0.1	0.1	0.1	0.6	0.1	0	0.1	0.6
7	0.1	0.1	0.6	0.1	0.1	0.1	0	0.1
8	0.1	0.1	0.1	0.6	0.1	0.6	0.1	0

2 次元距離確率表 $2D-p_{ij}$

	1	2	3	4	5	6	7	8
1	0	0.1	0.1	0.6	0.6	0.1	0.1	0.1
2	0.1	0	0.1	0.1	0.6	0.6	0.1	0.1
3	0.1	0.1	0	0.1	0.1	0.6	0.6	0.1
4	0.6	0.1	0.1	0	0.6	0.1	0.1	0.1
5	0.1	0.6	0.1	0.6	0	0.1	0.1	0.1
6	0.1	0.6	0.6	0.1	0.1	0	0.1	0.1
7	0.1	0.1	0.6	0.1	0.1	0.1	0	0.6
8	0.1	0.1	0.1	0.1	0.1	0.1	0.6	0

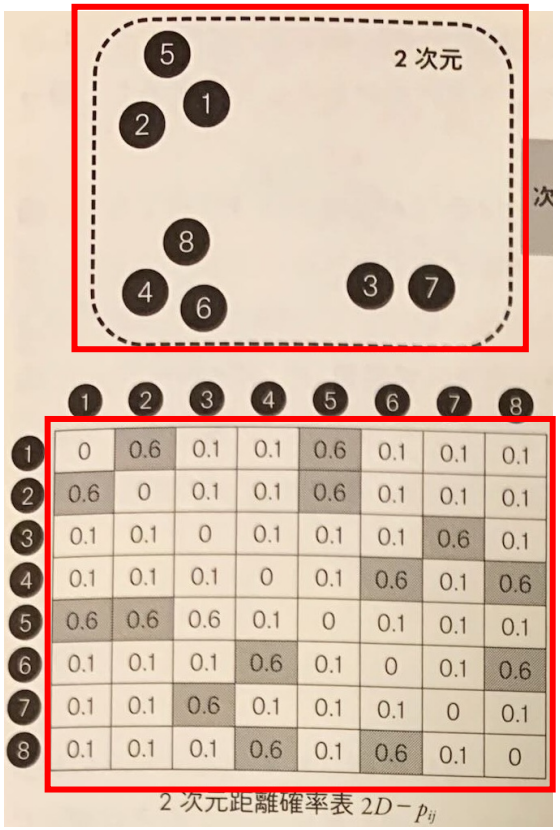
1 次元距離確率表 $1D-q_{ij}$

図 1.42　t-SNE による次元削減

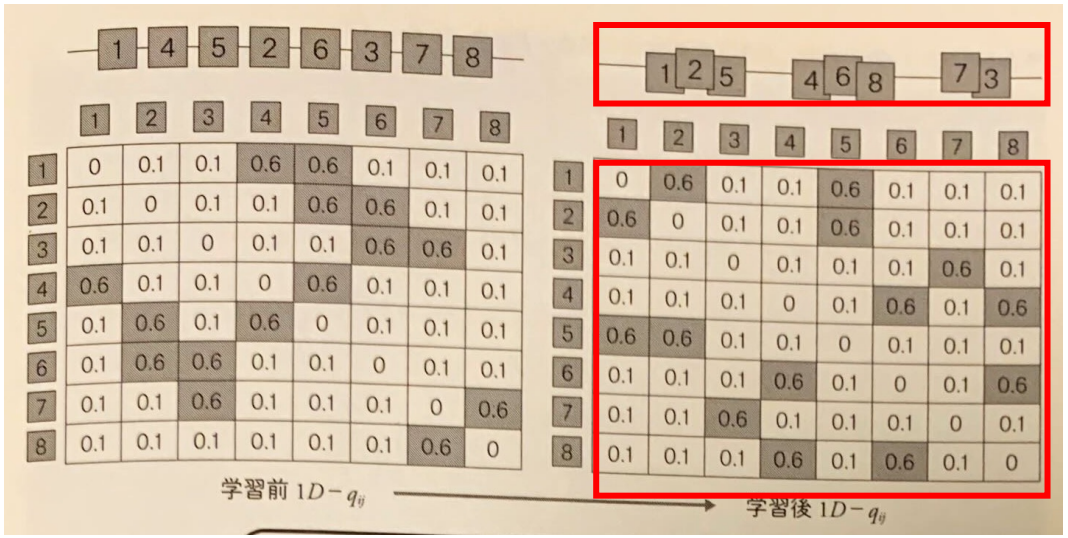
圧縮前と圧縮後の次元での距離確率表を近づける

- 分布を近づける＝KL（カルバック・ライブラリー）を最小にするように、圧縮後の次元の距離確率表を再配置する
- 主成分分析では情報ロスが起きる分布でも、t-SNEだと情報ロスなしでマッピングできる

圧縮前



圧縮後



UMAPの特徴

- 参考）https://qiita.com/odanny/items/06ab88353bcee7bf6aa7

- t-SNEよりも高速・高性能に次元削減・可視化する手法である。 UMAPは埋め込み次元数によらず、実行時間がほとんど一定である。t-SNE のように埋め込み次元が増えても指数関数的に実行時間が増える
- データの局所構造に基づいて構築したグラフを、大域構造を表すように配置していく