特徴抽出

カーネル主成分分析, 非負値行列因子分解, オートエンコーダ, t-SNE

中田和秀

東京科学大学 工学院 経営工学系

機械学習入門

https://www.nakatalab.iee.e.titech.ac.jp/text/nakata.html

概要

ここでは教師なし学習の一つである特徴抽出について説明する。特徴抽出によって得られた低次元ベクトルは、教師あり学習の特徴量として使ったり、可視化して特徴を理解することなどに利用できる。

目次：

記号の使い方：

- $A := B$ は、B で A を定義する、B を A に代入することを意味する
- $[n]$ は n までのインデックスの集合を表し $[n] := \{1, 2, \cdots, n\}$

特徴抽出

データ： $\{\boldsymbol{x}_d\}_{d\in[D]}$, $\boldsymbol{x}_d \in \mathbb{R}^n$

目的： n 次元のベクトルを P 次元 $(P \ll n)$ のベクトルで表現
- 無駄な情報（ノイズ？）を削り、本質的な情報を取り出すため
- 各データ点を理解するため

見方の違いによって様々な名前で呼ばれる。
- 特徴抽出： 重要な特徴を低次元のベクトルで表現する
- 次元圧縮： 高次元ベクトルを低次元ベクトルに変換する
- 埋め込み： 各データを低次元な線形空間の中へ埋め込む
- 非可逆データ圧縮： 元の情報を低次元なベクトルで出来るだけ保持する
- 可視化： 2 次元（3 次元）のベクトルに変換して図示する

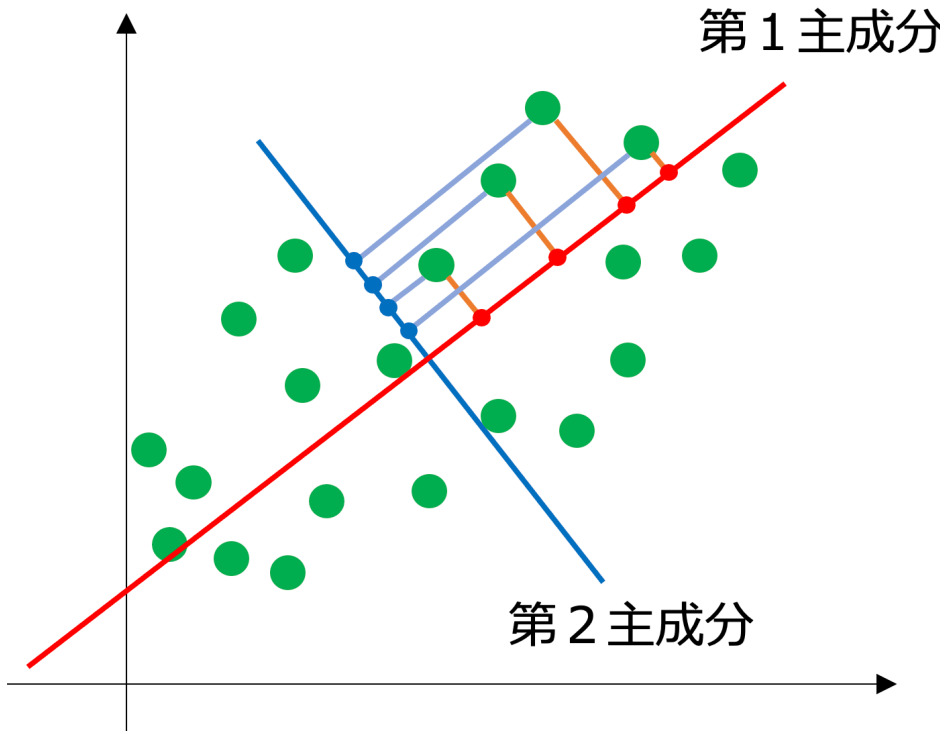
ここでは、以下の手法を紹介する
- カーネル主成分分析
- 非負値行列因子分解 (NMF)
- オートエンコーダー
- t-SNE

主成分分析

- $\boldsymbol{x} = \boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_D$ に対して線形変換をして、一つの実数で特徴付け

$$s := \boldsymbol{w}^T \boldsymbol{x} \in \mathbb{R}$$

- アイデア：新しい特徴量 s が各データ点の違いを表現するためには、
  s の分散が大きなほうがよい（情報量が大きい）。



分散共分散行列

$$\bar{\boldsymbol{x}} := \frac{1}{D}\sum_{d\in[D]} \boldsymbol{x}_d, \qquad \overline{\boldsymbol{x}_d} := \boldsymbol{x}_d - \bar{\boldsymbol{x}}, \qquad \boldsymbol{X}_0 := \begin{pmatrix} \overline{\boldsymbol{x}_1}^T \\ \overline{\boldsymbol{x}_2}^T \\ \vdots \\ \overline{\boldsymbol{x}_D}^T \end{pmatrix} \in \mathbb{R}^{D\times n}$$

とする。（標本）分散共分散行列は

$$\mathbb{V}[\boldsymbol{x}] := \frac{1}{D}\boldsymbol{X}_0^T\boldsymbol{X}_0 \quad \in \mathbb{R}^{n\times n}$$

$\boldsymbol{\Sigma} := \mathbb{V}[\boldsymbol{x}] = \frac{1}{D}\boldsymbol{X}_0^T\boldsymbol{X}_0$ とすると、

$$\mathbb{V}[s] = \mathbb{V}[\boldsymbol{w}^T\boldsymbol{x}] = \boldsymbol{w}^T\mathbb{V}[\boldsymbol{x}]\boldsymbol{w} = \boldsymbol{w}^T\boldsymbol{\Sigma}\boldsymbol{w}$$

第一主成分

次の最適化問題を解く

$$\begin{aligned} \max_{\boldsymbol{w}} \quad & \boldsymbol{w}^T \boldsymbol{\Sigma} \boldsymbol{w} \\ \text{s.t.} \quad & \boldsymbol{w}^T \boldsymbol{w} = 1. \end{aligned}$$

この問題の最適解は $\boldsymbol{\Sigma}$ の最大固有値に対応する固有ベクトル

→ 説明は次スライド

- 最適解 $\boldsymbol{w}_1^*$ を使って計算された特徴量 $s_1 := \boldsymbol{w}_1^{*T} \overline{\boldsymbol{x}}$ を第１主成分得点という。
- 全てのデータ $\{\boldsymbol{x}_d\}_{d \in [D]}$ に対する第１主成分得点を並べたベクトル

$$\boldsymbol{s}_1 := \begin{pmatrix} \boldsymbol{w}_1^{*T} \overline{\boldsymbol{x}_1} \\ \boldsymbol{w}_1^{*T} \overline{\boldsymbol{x}_2} \\ \vdots \\ \boldsymbol{w}_1^{*T} \overline{\boldsymbol{x}_D} \end{pmatrix} = \begin{pmatrix} \overline{\boldsymbol{x}_1}^T \boldsymbol{w}_1^* \\ \overline{\boldsymbol{x}_2}^T \boldsymbol{w}_1^* \\ \vdots \\ \overline{\boldsymbol{x}_D}^T \boldsymbol{w}_1^* \end{pmatrix} = \boldsymbol{X}_0 \boldsymbol{w}_1^* \in \mathbb{R}^D$$

説明

等式制約付き非線形最適化問題なのでラグランジュの未定乗数法で解く。
ラグランジュ関数は、ラグランジュ乗数 α を導入して

$$L(\boldsymbol{w}, \alpha) := \boldsymbol{w}^T \boldsymbol{\Sigma} \boldsymbol{w} + \alpha(1 - \boldsymbol{w}^T \boldsymbol{w})$$

となる。最適解であるための必要条件は、

$$\nabla_{\boldsymbol{w}} L(\boldsymbol{w}, \alpha) = 2\boldsymbol{\Sigma}\boldsymbol{w} - 2\alpha\boldsymbol{w} = \boldsymbol{0},$$
$$\nabla_{\alpha} L(\boldsymbol{w}, \alpha) = 1 - \boldsymbol{w}^T \boldsymbol{w} = 0$$

１つ目の式より、

$$\boldsymbol{\Sigma}\boldsymbol{w} = \alpha\boldsymbol{w}$$

つまり、最適解の候補 $\boldsymbol{w}$ は $\boldsymbol{\Sigma}$ の固有ベクトルであることを示している。固有値 λ に対する固有ベクトルを $\boldsymbol{w}$ とすると、目的関数値は

$$\boldsymbol{w}^T \boldsymbol{\Sigma} \boldsymbol{w} = \boldsymbol{w}^T (\lambda \boldsymbol{w}) = \lambda \boldsymbol{w}^T \boldsymbol{w} = \lambda$$

よって、最大固有値に対する固有ベクトルが最適解になる。

他の主成分

第二主成分

- 第一主成分の影響を取り除いた上で、分散が最大となる線形変換を考える
- 第二主成分 $\boldsymbol{w}_2^*$ は $\boldsymbol{\Sigma}$ の２番目に大きな固有値に対する固有ベクトル

  $\rightarrow$ 詳細は次ページ
- 第二主成分得点を並べたものは $\boldsymbol{s}_2 := \boldsymbol{X}_0 \boldsymbol{w}_2^* \in \mathbb{R}^D$

第 p 主成分 $(p \in [P])$

- 第 $p-1$ 主成分までの影響を取り除いた上、分散が最大となる線形変換を考える
- 第 p 主成分は $\boldsymbol{\Sigma}$ の p 番目に大きな固有値に対応する固有ベクトル $\boldsymbol{w}_p^*$
- 第 p 主成分得点を並べたものは $\boldsymbol{s}_p := \boldsymbol{X}_0 \boldsymbol{w}_p^* \in \mathbb{R}^D$

$\boldsymbol{x}_d$ の次元圧縮をして得られた新たな特徴ベクトルは

$$([\boldsymbol{s}_1]_d, [\boldsymbol{s}_2]_d, \cdots, [\boldsymbol{s}_P]_d)^T \in \mathbb{R}^P$$

第二主成分の計算

- $\overline{\boldsymbol{x}_d}$ に対して、既に特徴量化した成分を引いておく（直交射影）。

$$\overline{\boldsymbol{x}_d} - (\boldsymbol{w}_1^T \overline{\boldsymbol{x}_d}) \boldsymbol{w}_1$$

- ２番目の特徴量 $s_2 := \boldsymbol{w}_2^T \boldsymbol{x}$ を作るための重み $\boldsymbol{w}_2$ を考える。

$$\begin{aligned} \overline{\boldsymbol{x}_d} &\to \overline{\boldsymbol{x}_d} - (\boldsymbol{w}_1^T \overline{\boldsymbol{x}_d}) \boldsymbol{w}_1 \\ \boldsymbol{X}_0^T &\to \boldsymbol{X}_0^T - \boldsymbol{w}_1 \boldsymbol{w}_1^T \boldsymbol{X}_0^T \\ \boldsymbol{\Sigma} &\to \boldsymbol{\Sigma} - \lambda_1 \boldsymbol{w}_1 \boldsymbol{w}_1^T \end{aligned}$$

式変形は次スライド

最適化問題は次のようになる

$$\begin{aligned} \max_{\boldsymbol{w}} \quad & \boldsymbol{w}^T (\boldsymbol{\Sigma} - \lambda_1 \boldsymbol{w}_1 \boldsymbol{w}_1^T) \boldsymbol{w} \\ \text{s.t.} \quad & \boldsymbol{w}^T \boldsymbol{w} = 1. \end{aligned}$$

この問題の最適解は $\boldsymbol{\Sigma}$ の２番目に大きな固有値に対する固有ベクトルである。

式変形

平均

$$\frac{1}{D}\sum_{d\in[D]}\overline{\boldsymbol{x}_d}-(\boldsymbol{w}_1^T\overline{\boldsymbol{x}_d})\boldsymbol{w}_1=\frac{1}{D}\sum_{d\in[D]}\overline{\boldsymbol{x}_d}-\boldsymbol{w}_1\boldsymbol{w}_1^T\overline{\boldsymbol{x}_d}$$

$$=\frac{1}{D}(\boldsymbol{I}-\boldsymbol{w}_1\boldsymbol{w}_1^T)\sum_{d\in[D]}\overline{\boldsymbol{x}_d}=\boldsymbol{0}$$

分散共分散行列

$$\frac{1}{D}(\boldsymbol{X}_0^T-\boldsymbol{w}_1\boldsymbol{w}_1^T\boldsymbol{X}_0^T)(\boldsymbol{X}_0-\boldsymbol{X}_0\boldsymbol{w}_1\boldsymbol{w}_1^T)$$

$$=\frac{1}{D}\left(\boldsymbol{X}_0^T\boldsymbol{X}_0-\boldsymbol{w}_1\boldsymbol{w}_1^T\boldsymbol{X}_0^T\boldsymbol{X}_0-\boldsymbol{X}_0^T\boldsymbol{X}_0\boldsymbol{w}_1\boldsymbol{w}_1^T+\boldsymbol{w}_1\boldsymbol{w}_1^T\boldsymbol{X}_0^T\boldsymbol{X}_0\boldsymbol{w}_1\boldsymbol{w}_1^T\right)$$

$$=\frac{1}{D}\boldsymbol{X}_0^T\boldsymbol{X}_0-\lambda_1\boldsymbol{w}_1\boldsymbol{w}_1^T-\lambda_1\boldsymbol{w}_1\boldsymbol{w}_1^T+\lambda_1\boldsymbol{w}_1\boldsymbol{w}_1^T$$

$$=\frac{1}{D}\boldsymbol{X}_0^T\boldsymbol{X}_0-\lambda_1\boldsymbol{w}_1\boldsymbol{w}_1^T=\boldsymbol{\Sigma}-\lambda_1\boldsymbol{w}_1\boldsymbol{w}_1^T$$

固有値分解

$$\boldsymbol{\Sigma}=\sum_{i=1}^{n}\lambda_i\boldsymbol{w}_i\boldsymbol{w}_i^T\quad\text{のとき、}\quad\boldsymbol{\Sigma}-\lambda_1\boldsymbol{w}_1\boldsymbol{w}_1^T=\sum_{i=2}^{n}\lambda_i\boldsymbol{w}_i\boldsymbol{w}_i^T$$

寄与率

各特徴量の分散の和は固有値の和と一致する。

$$\sum_{i\in[n]} \mathbb{V}[x_i] = \sum_{i\in[n]} \Sigma_{ii} = \mathsf{Tr}(\boldsymbol{\Sigma}) = \sum_{i\in[n]} \lambda_i$$

分散全体に対する主成分得点 s_p の分散の割合を寄与率という。

寄与率： $c_p := \dfrac{\lambda_p}{\sum_{i\in[n]} \lambda_i} = \dfrac{\lambda_p}{\sum_{i\in[n]} \Sigma_{ii}} \qquad (p \in [P])$

累積寄与率： $r_p := \sum_{i\in[p]} c_i = \dfrac{\sum_{i\in[p]} \lambda_i}{\sum_{i\in[n]} \lambda_i} = \dfrac{\sum_{i\in[p]} \lambda_i}{\sum_{i\in[n]} \Sigma_{ii}} \qquad (p \in [P])$

$$0 \leq c_P \leq c_{P-1} \leq \cdots \leq c_1 = r_1 \leq r_2 \leq \cdots r_P \leq 1$$

累積寄与率によって、どの程度の情報を抽出できているか判断する

可視化

- 1 番目のスコアを x 軸、2 番目のスコアを y 軸にとると可視化が可能になる。

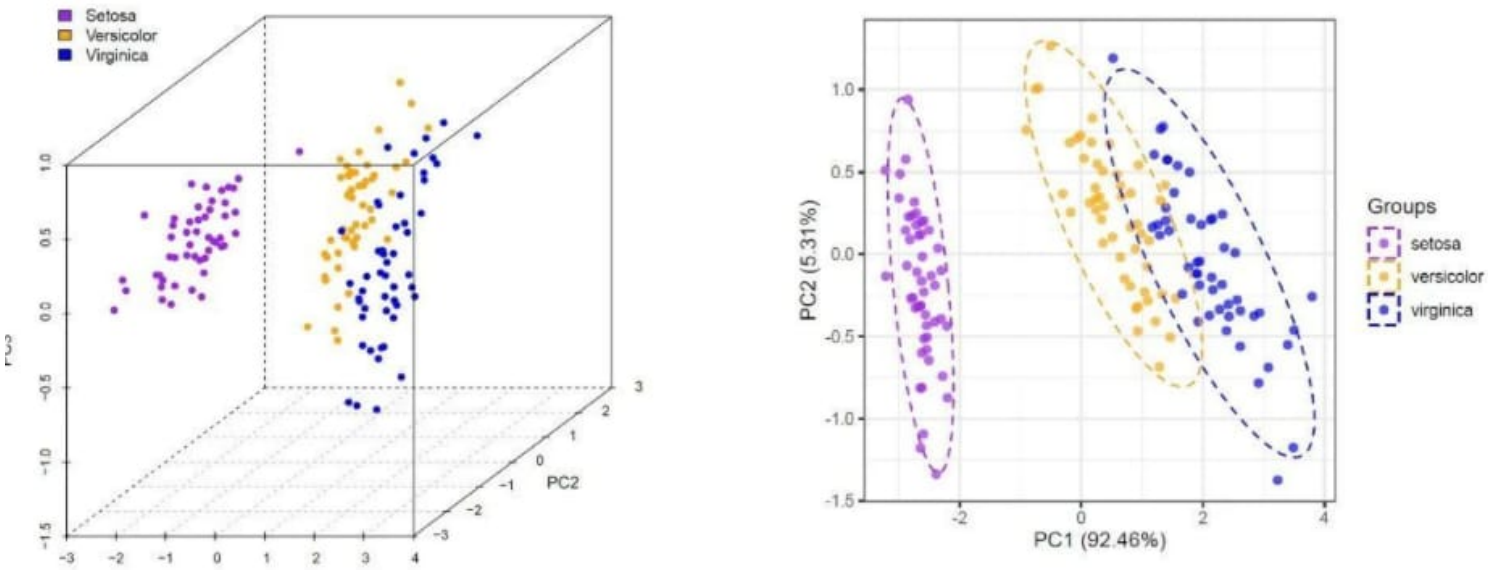


Figure: https://www.binarydevelop.com/article/rpca-28546

カーネル主成分分析

- 主成分分析では、線形変換によって新しい特徴量（主成分得点）を計算
- 非線形な変換を行いたい
- 入力空間を高次元の特徴空間に写像して、その特徴空間上で主成分分析を行う

$\boldsymbol{\phi} : \mathbb{R}^n \to V$ として、$\boldsymbol{x}_d$ の代わりに $\boldsymbol{\phi}(\boldsymbol{x}_d)$ を使う [1]。

$$\overline{\boldsymbol{\phi}_{\boldsymbol{x}}} := \frac{1}{D} \sum_{d \in [D]} \boldsymbol{\phi}(\boldsymbol{x}_d), \quad \overline{\boldsymbol{\phi}(\boldsymbol{x}_d)} := \boldsymbol{\phi}(\boldsymbol{x}_d) - \overline{\boldsymbol{\phi}_{\boldsymbol{x}}}, \quad \widetilde{\boldsymbol{X}}_0 := \begin{pmatrix} \overline{\boldsymbol{\phi}(\boldsymbol{x}_1)}^T \\ \overline{\boldsymbol{\phi}(\boldsymbol{x}_2)}^T \\ \vdots \\ \overline{\boldsymbol{\phi}(\boldsymbol{x}_D)}^T \end{pmatrix} \in \mathbb{R}^{D \times |V|}$$

分散共分散行列

$$\boldsymbol{\Sigma} := \frac{1}{D} \widetilde{\boldsymbol{X}}_0^T \widetilde{\boldsymbol{X}}_0 \in \mathbb{R}^{|V| \times |V|}$$

[1] 写像先の高次元空間 V の次元 $|V|$ は無限かもしれないが、有限のイメージで話を進める

カーネルトリック [2]

$\frac{1}{D}\widetilde{\boldsymbol{X}}_0^T\widetilde{\boldsymbol{X}}_0 \in \mathbb{R}^{|V|\times|V|}$ の固有値分解が必要

- $\boldsymbol{\phi}(\boldsymbol{x})$ を使わず、$\boldsymbol{\phi}(\boldsymbol{x}_i)^T\boldsymbol{\phi}(\boldsymbol{x}_j)$ に相当する $\mathrm{Ker}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ を使いたい
- つまり、$\boldsymbol{K} := \widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{X}}^T \in \mathbb{R}^{D\times D}$ を使って計算する

$$\widetilde{\boldsymbol{X}} := \begin{pmatrix} \boldsymbol{\phi}(\boldsymbol{x}_1)^T \\ \boldsymbol{\phi}(\boldsymbol{x}_2)^T \\ \vdots \\ \boldsymbol{\phi}(\boldsymbol{x}_D)^T \end{pmatrix} \in \mathbb{R}^{D\times|V|}$$

問題点

- $\widetilde{\boldsymbol{X}}$ でなく $\widetilde{\boldsymbol{X}}_0$ を使った計算が必要（平均を $\mathbf{0}$ にする平行移動がある）
- $\widetilde{\boldsymbol{X}}_0^T\widetilde{\boldsymbol{X}}_0 \in \mathbb{R}^{|V|\times|V|}$ だが、計算できるのはカーネル行列 $\boldsymbol{K} := \widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{X}}^T \in \mathbb{R}^{D\times D}$

[2] カーネルトリックの詳細については、カーネル法のスライドを参照されたい

修正カーネル行列

計算可能なカーネル行列 $\boldsymbol{K} := \widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{X}}^T$ から修正カーネル行列 $\boldsymbol{K}_0 := \widetilde{\boldsymbol{X}}_0\widetilde{\boldsymbol{X}}_0^T$ を作る。

$$\text{平均ベクトル} \qquad \overline{\boldsymbol{\phi}_{\boldsymbol{x}}} := \frac{1}{D}\sum_{d\in[D]}\boldsymbol{\phi}(\boldsymbol{x}_d) = \frac{1}{D}\widetilde{\boldsymbol{X}}^T\boldsymbol{e}$$

であるので、

$$\widetilde{\boldsymbol{X}}_0 := \begin{pmatrix} \boldsymbol{\phi}(\boldsymbol{x}_1)^T - \overline{\boldsymbol{\phi}_{\boldsymbol{x}}}^T \\ \boldsymbol{\phi}(\boldsymbol{x}_2)^T - \overline{\boldsymbol{\phi}_{\boldsymbol{x}}}^T \\ \vdots \\ \boldsymbol{\phi}(\boldsymbol{x}_D)^T - \overline{\boldsymbol{\phi}_{\boldsymbol{x}}}^T \end{pmatrix} = \widetilde{\boldsymbol{X}} - \boldsymbol{e}\overline{\boldsymbol{\phi}_{\boldsymbol{x}}}^T = \widetilde{\boldsymbol{X}} - \frac{1}{D}\boldsymbol{e}\boldsymbol{e}^T\widetilde{\boldsymbol{X}}$$

という関係が成り立つ。このとき、

$$\boldsymbol{K}_0 = \boldsymbol{K} - \frac{1}{D}\boldsymbol{e}\boldsymbol{e}^T\boldsymbol{K} - \frac{1}{D}\boldsymbol{K}\boldsymbol{e}\boldsymbol{e}^T + \frac{\boldsymbol{e}^T\boldsymbol{K}\boldsymbol{e}}{D^2}\boldsymbol{e}\boldsymbol{e}^T$$

固有値の計算

$\widetilde{\boldsymbol{X}}_0^T \widetilde{\boldsymbol{X}}_0$ の固有値 λ と固有ベクトル $\boldsymbol{w} \in \mathbb{R}^{|V|}$ は次の関係を満たす。

$$\widetilde{\boldsymbol{X}}_0^T \widetilde{\boldsymbol{X}}_0 \boldsymbol{w} = \lambda \boldsymbol{w}$$

このとき、次の関係が成り立つ。

$$\exists \boldsymbol{a} \in \mathbb{R}^D, \quad \boldsymbol{w} = \widetilde{\boldsymbol{X}}_0^T \boldsymbol{a}$$

これを代入すると、

$$\widetilde{\boldsymbol{X}}_0^T \widetilde{\boldsymbol{X}}_0 \widetilde{\boldsymbol{X}}_0^T \boldsymbol{a} = \lambda \widetilde{\boldsymbol{X}}_0^T \boldsymbol{a}$$

左から $\widetilde{\boldsymbol{X}}_0$ をかけると

$$\widetilde{\boldsymbol{X}}_0 \widetilde{\boldsymbol{X}}_0^T \widetilde{\boldsymbol{X}}_0 \widetilde{\boldsymbol{X}}_0^T \boldsymbol{a} = \lambda \widetilde{\boldsymbol{X}}_0 \widetilde{\boldsymbol{X}}_0^T \boldsymbol{a}$$

これは、$\boldsymbol{K}_0^2 \boldsymbol{a} = \lambda \boldsymbol{K}_0 \boldsymbol{a}$ なので、次の関係を満たせば上の式も成り立つ。

$$\boldsymbol{K}_0 \boldsymbol{a} = \lambda \boldsymbol{a}$$

この固有値問題を解き、求まった $\boldsymbol{a}$ に対し $\widetilde{\boldsymbol{X}}_0^T \boldsymbol{a}$ が必要な固有ベクトル $\boldsymbol{w}$ となる。

主成分得点の計算

既に固有ベクトル $\boldsymbol{a}_p$ $(p \in [P])$ は計算されているとする。

第 p 主成分得点

$$s_p := \boldsymbol{w}_p^{*T} \overline{\boldsymbol{\phi}(\boldsymbol{x})} = \overline{\boldsymbol{\phi}(\boldsymbol{x})}^T \boldsymbol{w}_p^* = \overline{\boldsymbol{\phi}(\boldsymbol{x})}^T \widetilde{\boldsymbol{X}}_0^T \boldsymbol{a}_p$$

全データ $\{\boldsymbol{x}_d\}_{d \in [D]}$ に対する得点を縦に並べると

$$\begin{aligned}
\boldsymbol{s}_p :=& \begin{pmatrix} \overline{\boldsymbol{\phi}(\boldsymbol{x}_1)}^T \\ \overline{\boldsymbol{\phi}(\boldsymbol{x}_2)}^T \\ \vdots \\ \overline{\boldsymbol{\phi}(\boldsymbol{x}_D)}^T \end{pmatrix} \widetilde{\boldsymbol{X}}_0^T \boldsymbol{a}_p \\
=& \widetilde{\boldsymbol{X}}_0 \widetilde{\boldsymbol{X}}_0^T \boldsymbol{a}_p \\
=& \boldsymbol{K}_0 \boldsymbol{a}_p \\
=& \lambda_p \boldsymbol{a}_p \qquad\qquad \in \mathbb{R}^D
\end{aligned}$$

学習アルゴリズム

学習アルゴリズム

ステップ1 $K_{ij} := \mathrm{Ker}(\boldsymbol{x}_i, \boldsymbol{x}_j) \quad i, j \in [D]$

ステップ2 $\boldsymbol{K}_0 := \boldsymbol{K} - \frac{1}{D}\boldsymbol{e}\boldsymbol{e}^T\boldsymbol{K} - \frac{1}{D}\boldsymbol{K}\boldsymbol{e}\boldsymbol{e}^T + \frac{\boldsymbol{e}^T\boldsymbol{K}\boldsymbol{e}}{D^2}\boldsymbol{e}\boldsymbol{e}^T$

ステップ3 $\boldsymbol{K}_0$ の固有値が大きな p 個の固有ベクトルの計算。$\boldsymbol{a}_p \quad (p \in [P])$

ステップ4 $p \in [P]$ に対して、第 p 主成分得点の計算

$$\boldsymbol{s}_p := \lambda_p \boldsymbol{a}_p \quad \in \mathbb{R}^D$$

$\phi(\boldsymbol{x}_d)$ の次元圧縮をして得られた新たな特徴ベクトルは

$$([\boldsymbol{s}_1]_d, [\boldsymbol{s}_2]_d, \cdots, [\boldsymbol{s}_P]_d)^T \in \mathbb{R}^P$$

- $\{\boldsymbol{\phi}(\boldsymbol{x}_d)\}_{d\in[D]}$ に対する線形変換
- $\{\boldsymbol{x}_d\}_{d\in[D]}$ に対する非線形変換

計算例

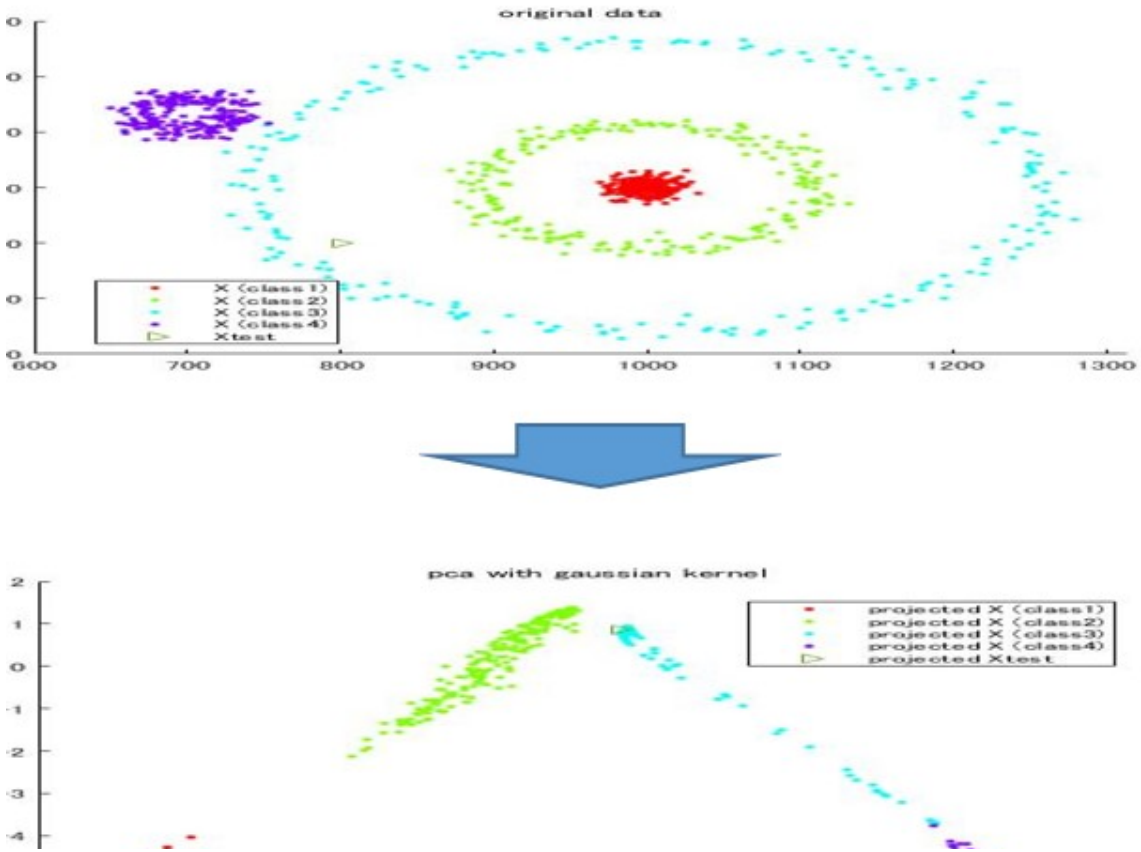


Figure: https://jp.mathworks.com/matlabcentral/fileexchange/71647-matlab-kernel-pca?s_tid=prof_contriblnk

カーネル主成分分析のまとめ

特徴

- データ数 D の大きさの行列の固有値分解が必要

  計算量：$O(D^3)$

  計算量のオーダーは減らすこともできるが、色々と難しい・・・
- データ数が増えると固有ベクトルの計算が困難
- カーネルを導入すると、特徴量（軸）の解釈が困難になる
- 主成分の数 P は累積寄与率をみて決めることも可能

非負値行列因子分解 (NMF)

データ　$\boldsymbol{T} \in \mathbb{R}^{I \times J}$

$T_{ij} \geq 0$：ユーザ i のアイテム j に対する評価値

目的

各ユーザと各アイテムの特徴量を構築したい

モデル：

- 評価値 T_{ij} は、K 個の要因の和で構成される
- ユーザやアイテムは、各要因に対する感度を持つ
  - $\boldsymbol{x}_i \in \mathbb{R}^K$：ユーザ i の各要因に対する感度　　$\boldsymbol{x}_i \geq \boldsymbol{0}$
  - $\boldsymbol{y}_j \in \mathbb{R}^K$：アイテム j の各要因に対する感度　　$\boldsymbol{y}_j \geq \boldsymbol{0}$
- $T_{ij} \simeq \boldsymbol{x}_i^T \boldsymbol{y}_j = \sum_{k \in [K]} [\boldsymbol{x}_i]_k [\boldsymbol{y}_j]_k$

$\boldsymbol{x}_i$, $\boldsymbol{y}_j$ をユーザやアイテムの特徴量とする

最適化問題

2 乗誤差 $(T_{ij} - \boldsymbol{x}_i^T \boldsymbol{y}_j)^2$ が小さくなるように特徴ベクトル $\boldsymbol{x}_i, \boldsymbol{y}_j$ を決める

非負値行列因子分解

変数： $\boldsymbol{x}_i \in \mathbb{R}^K \ (i \in [I]), \quad \boldsymbol{y}_j \in \mathbb{R}^K \ (j \in [J])$

$$\begin{aligned} \min \quad & \sum_{i \in [I]} \sum_{j \in [J]} (T_{ij} - \boldsymbol{x}_i^T \boldsymbol{y}_j)^2 \\ \text{s.t.} \quad & \boldsymbol{x}_i \geq \boldsymbol{0} \ (i \in [I]), \\ & \boldsymbol{y}_j \geq \boldsymbol{0} \ (j \in [J]). \end{aligned}$$

$$\boldsymbol{X} := \begin{pmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_I^T \end{pmatrix} \in \mathbb{R}^{I \times K}, \quad \boldsymbol{Y} := \begin{pmatrix} \boldsymbol{y}_1 & \boldsymbol{y}_2 & \cdots & \boldsymbol{y}_J \end{pmatrix} \in \mathbb{R}^{K \times J}$$

とする

行列表現

非負値行列因子分解

変数： $\boldsymbol{X} \in \mathbb{R}^{I \times K}$, $\boldsymbol{Y} \in \mathbb{R}^{K \times J}$

$$\begin{aligned} \min \quad & \|\boldsymbol{T} - \boldsymbol{X}\boldsymbol{Y}\|_F^2 \\ \text{s.t.} \quad & \boldsymbol{X} \geq \boldsymbol{O},\ \boldsymbol{Y} \geq \boldsymbol{O}. \end{aligned}$$

フロベニウスノルム $\|\boldsymbol{A}\|_F := \sqrt{\mathrm{Tr}(\boldsymbol{A}^T\boldsymbol{A})} = \sqrt{\mathrm{Tr}(\boldsymbol{A}\boldsymbol{A}^T)} = \sqrt{\sum_{i\in[I]}\sum_{j\in[J]}(A_{ij})^2}$

$\boldsymbol{T} \simeq \boldsymbol{X}\boldsymbol{Y}$

- 行列を低ランク行列で近似
  $\mathrm{rank}(\boldsymbol{T}) = \min\{I, J\}, \quad \mathrm{rank}(\boldsymbol{X}\boldsymbol{Y}) = K \qquad (I, J \gg K)$
- 行列を２つの行列に分解

非負値行列因子分解（NMF：Nonnegative Matrix Factorization）と呼ぶ

行列因子分解 (MF)

まず、非負条件の無い行列因子分解は簡単であることをみておく。

行列因子分解

変数： $\boldsymbol{X} \in \mathbb{R}^{I \times K}$, $\boldsymbol{Y} \in \mathbb{R}^{K \times J}$

$$\min \quad \|\boldsymbol{T} - \boldsymbol{X}\boldsymbol{Y}\|_F^2$$

$\boldsymbol{T}$ の特異値分解：　　特異値は降順に $\boldsymbol{\Sigma}$ の左上から対角部分に並んでいるとする

$$\boldsymbol{T} = \boldsymbol{P}\boldsymbol{\Sigma}\boldsymbol{Q}^T$$

- $\boldsymbol{\Sigma}_K \in \mathbb{R}^{K \times K}$： $\boldsymbol{\Sigma}$ の左上 $K \times K$ を抜き出した対角行列
- $\boldsymbol{P}_K \in \mathbb{R}^{I \times K}$： $\boldsymbol{P}$ の左側 K 列を抜き出した行列
- $\boldsymbol{Q}_K \in \mathbb{R}^{J \times K}$： $\boldsymbol{Q}$ の左側 K 列を抜き出した行列

このとき、最適解 $\boldsymbol{X}^*, \boldsymbol{Y}^*$ は次のように表せる。

$$\boldsymbol{X}^* := \boldsymbol{P}_K \boldsymbol{\Sigma}_K^{1/2} \boldsymbol{H}, \quad \boldsymbol{Y}^* := \boldsymbol{H}^{-1} \boldsymbol{\Sigma}_K^{1/2} \boldsymbol{Q}_K^T$$

ただし、$\boldsymbol{H} \in \mathbb{R}^{K \times K}$ は任意の正則行列

説明

Eckart–Young の定理

$\boldsymbol{Z}$ を変数とする次の最適化問題

$$\begin{aligned} \min \quad & \|\boldsymbol{T} - \boldsymbol{Z}\|_F \\ \text{s.t.} \quad & \operatorname{rank}(\boldsymbol{Z}) \leq K. \end{aligned}$$

の最適解は $\boldsymbol{Z}^* = \boldsymbol{P}_K \boldsymbol{\Sigma}_K \boldsymbol{Q}_K^T$　　（$\boldsymbol{P}_K, \boldsymbol{\Sigma}_K, \boldsymbol{Q}_K$ は前ページで定義したもの）

証明は `https://en.wikipedia.org/wiki/Low-rank_approximation`

この定理より、$\boldsymbol{X}^*\boldsymbol{Y}^* = \boldsymbol{P}_K \boldsymbol{\Sigma}_K \boldsymbol{Q}_K^T$ と分かる。このとき、最適解 $\boldsymbol{X}^*, \boldsymbol{Y}^*$ は

$$\boldsymbol{X}^* := \boldsymbol{P}_K \boldsymbol{\Sigma}_K^{1/2} \boldsymbol{H}, \quad \boldsymbol{Y}^* := \boldsymbol{H}^{-1} \boldsymbol{\Sigma}_K^{1/2} \boldsymbol{Q}_K^T$$

と表せる。ただし、$\boldsymbol{H} \in \mathbb{R}^{K \times K}$ は任意の正則行列

- この形式で表された $\boldsymbol{X}^*$ と $\boldsymbol{Y}^*$ が最適解であるのは明らか。
- 全ての最適解がこの形式で表されることも、特異値を 0 と 0 以外のもので分ければ証明できる。

$\boldsymbol{H}$ の取り方に任意性があるため、最適解が一意に決まらないのが欠点

非負値行列因子分解

最適化問題

$$\min_{\boldsymbol{X},\boldsymbol{Y}} \ \|\boldsymbol{T}-\boldsymbol{X}\boldsymbol{Y}\|_F^2$$
$$\text{s.t. } \boldsymbol{X}\geq\boldsymbol{O},\ \boldsymbol{Y}\geq\boldsymbol{O}.$$

$\boldsymbol{X}$ と $\boldsymbol{Y}$ を交互に最適化することを考える。

$$\min_{\boldsymbol{X}} \ \|\boldsymbol{T}-\boldsymbol{X}\boldsymbol{Y}\|_F^2 \qquad \min_{\boldsymbol{Y}} \ \|\boldsymbol{T}-\boldsymbol{X}\boldsymbol{Y}\|_F^2$$
$$\text{s.t. } \boldsymbol{X}\geq\boldsymbol{O}. \qquad \text{s.t. } \boldsymbol{Y}\geq\boldsymbol{O}.$$

目的関数は $\boldsymbol{X},\boldsymbol{Y}$ それぞれの変数に関する凸２次関数であるが、非負制約があるため、最適解を求めることは難しい。

$\longrightarrow$ 補助関数法を使う。

補助関数 1

$$F(\boldsymbol{X}, \boldsymbol{Y}) := \|\boldsymbol{T} - \boldsymbol{X}\boldsymbol{Y}\|_F^2 = \sum_{i\in[I]} \sum_{j\in[J]} \left(T_{ij} - \sum_{k\in[K]} X_{ik} Y_{kj} \right)^2$$

$i \in [I],\ j \in [J]$ に対し、

$$\left(T_{ij} - \sum_{k\in[K]} X_{ik} Y_{kj} \right)^2 = \left(\sum_{k\in[K]} X_{ik} Y_{kj} \right)^2 - 2\, T_{ij} \sum_{k\in[K]} X_{ik} Y_{kj} + T_{ij}^2$$

$\left(\sum_{k\in[K]} X_{ik} Y_{kj} \right)^2$ は、補助関数法の $f_0 \left(\sum_{k\in[K]} f_k(\boldsymbol{X}, \boldsymbol{Y}) \right)$ というケース

- $f_0(x) := x^2$ は凸関数
- $f_k(\boldsymbol{X}, \boldsymbol{Y}) := X_{ik} Y_{kj} \geq 0$

補助関数 2

任意の $\sum_{k\in[K]} r_{ijk} = 1,\ r_{ijk} > 0$ に対し、

$$\left(\sum_{k\in[K]} X_{ik}Y_{kj}\right)^2 = \left(\sum_{k\in[K]} r_{ijk}\frac{X_{ik}Y_{kj}}{r_{ijk}}\right)^2 \le \sum_{k\in[K]} r_{ijk}\left(\frac{X_{ik}Y_{kj}}{r_{ijk}}\right)^2 = \sum_{k\in[K]} \frac{X_{ik}^2 Y_{kj}^2}{r_{ijk}}$$

そして、$r^*_{ijk} = \dfrac{X_{ik}Y_{kj}}{\sum_{k'\in[K]} X_{ik'}Y_{k'j}}$ のとき等号が成り立つ。

補助関数

$$G(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{R}) := \sum_{i\in[I]}\sum_{j\in[J]}\left(\sum_{k\in[K]}\frac{X_{ik}^2Y_{kj}^2}{r_{ijk}} - 2\,T_{ij}\sum_{k\in[K]} X_{ik}Y_{kj} + T_{ij}^2\right)$$

は、$F(\boldsymbol{X},\boldsymbol{Y})$ の補助関数

(r_{ijk} は 3 つの添字からなる変数だが、まとめて行列風の $\boldsymbol{R}$ という記号を使う)

補助関数法

このとき、補助関数法のステップ２で解く最適化問題は

$$\begin{aligned} \min_{\boldsymbol{X},\boldsymbol{Y}} \ & G(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{R}) \\ \text{s.t.} \ & \boldsymbol{X} \geq \boldsymbol{O},\ \boldsymbol{Y} \geq \boldsymbol{O}. \end{aligned}$$

である。これを $\boldsymbol{X}$ と $\boldsymbol{Y}$ の問題に分割して順に解く。

$$\begin{aligned} \min_{\boldsymbol{X}} \ & G(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{R}) \\ \text{s.t.} \ & \boldsymbol{X} \geq \boldsymbol{O}. \end{aligned} \qquad \begin{aligned} \min_{\boldsymbol{Y}} \ & G(\boldsymbol{X},\boldsymbol{Y},\boldsymbol{R}) \\ \text{s.t.} \ & \boldsymbol{Y} \geq \boldsymbol{O}. \end{aligned}$$

これらの問題の最適解は $\widehat{\boldsymbol{T}} := \boldsymbol{X}\boldsymbol{Y}$ としたとき、次のように計算できる[3]。

$$X^*_{ik} = X_{ik} \frac{\sum_{j\in[J]} T_{ij} Y_{kj}}{\sum_{j\in[J]} \widehat{T}_{ij} Y_{kj}} \qquad Y^*_{kj} = \frac{\sum_{i\in[I]} X_{ik} T_{ij}}{\sum_{i\in[I]} X_{ik} \widehat{T}_{ij}}$$

証明は２ページ後

[3] $\boldsymbol{R}$ は $\boldsymbol{X}, \boldsymbol{Y}$ で置き換えているので消えている

アルゴリズム

非負値行列因子分解のアルゴリズム

ステップ 0 初期解 $\boldsymbol{X}, \boldsymbol{Y} > \boldsymbol{O}$ を適当に決める。

$\boldsymbol{X}$ の最適化

ステップ 1 $\widehat{\boldsymbol{T}} := \boldsymbol{X}\boldsymbol{Y}$

ステップ 2 $\boldsymbol{A} := \boldsymbol{T}\boldsymbol{Y}^T, \quad \widehat{\boldsymbol{A}} := \widehat{\boldsymbol{T}}\boldsymbol{Y}^T$

ステップ 3 $X_{ik} := X_{ik} \dfrac{A_{ik}}{\widehat{A}_{ik}} \qquad (i \in [I],\ k \in [K])$

$\boldsymbol{Y}$ の最適化

ステップ 4 $\widehat{\boldsymbol{T}} := \boldsymbol{X}\boldsymbol{Y}$ （更新した $\boldsymbol{X}$ を使うので、ステップ 1 の $\widehat{\boldsymbol{T}}$ とは異なる）

ステップ 5 $\boldsymbol{B} := \boldsymbol{X}^T\boldsymbol{T}, \quad \widehat{\boldsymbol{B}} := \boldsymbol{X}^T\widehat{\boldsymbol{T}}$

ステップ 6 $Y_{kj} := Y_{kj} \dfrac{B_{kj}}{\widehat{B}_{kj}} \qquad (k \in [K],\ j \in [J])$

ステップ 7 終了条件を満たさなければ、ステップ 1 に戻る。

※ 局所最適解が求める。初期点を替えて何度か実行するとよい。

証明 1

$\boldsymbol{X}$ に関する最適化問題を考える。

非負条件 $\boldsymbol{X} \geq \boldsymbol{O}$ を無視すると、目的関数は $\boldsymbol{X}$ に関する凸 2 次関数。

$$
\begin{aligned}
G(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{R}) &:= \sum_{i,j} \left(\sum_k \frac{X_{ik}^2 Y_{kj}^2}{r_{ijk}} - 2T_{ij} \sum_k X_{ik} Y_{kj} + T_{ij}^2 \right) \\
&= \sum_{i,k} \left(\left(\sum_j \frac{Y_{kj}^2}{r_{ijk}} \right) X_{ik}^2 - 2 \left(\sum_j T_{ij} Y_{kj} \right) X_{ik} \right) + \sum_{i,j} T_{ij}^2
\end{aligned}
$$

X_{ik} ごとに、2 次関数の最小化を考えればよいので、最適解は次のようになる。

$$
X_{ik}^* = \frac{\sum_j T_{ij} Y_{kj}}{\sum_j \dfrac{Y_{kj}^2}{r_{ijk}}}
$$

$r_{ijk} := \dfrac{X_{ik} Y_{kj}}{\sum_{k'} X_{ik'} Y_{k'j}}$ を代入する [4]

[4] r_{ijk} は定数であり（$\boldsymbol{X}, \boldsymbol{Y}$ は現在の反復点）、最適化問題の変数ではないことに注意

証明２

$$X^*_{ik} = \frac{\sum_j T_{ij} Y_{kj}}{\sum_j \frac{Y_{kj}}{X_{ik}} \sum_{k'} X_{ik'} Y_{k'j}} = X_{ik} \frac{\sum_j T_{ij} Y_{kj}}{\sum_j Y_{kj} \sum_{k'} X_{ik'} Y_{k'j}}$$

$$= X_{ik} \frac{\sum_j T_{ij} Y_{kj}}{\sum_j \widehat{T}_{ij} Y_{kj}} \qquad (\text{ただし } \widehat{\boldsymbol{T}} := \boldsymbol{X}\boldsymbol{Y})$$

- $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{T} \geq \boldsymbol{O}$ のとき、$\boldsymbol{X}^* \geq 0$ となり、自動的に非負条件制約は満たされる。よって、これが最適解である。
- 現在の反復点 $\boldsymbol{X}, \boldsymbol{Y}$ からの予測値 $\widetilde{\boldsymbol{T}}$ と正解 $\boldsymbol{T}$ との比から次の反復点 $\boldsymbol{X}$ が作られる。

$\boldsymbol{Y}$ の最適化も同様に、$G(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{R})$ を Y_{kj} に関する２次関数をつくれば $\boldsymbol{Y}$ に関する最小解を求めることができる。

$$Y^*_{kj} := Y_{kj} \frac{\sum_i T_{ij} X_{ik}}{\sum_i X_{ik} \sum_{k'} X_{ik'} Y_{k'j}} = Y_{kj} \frac{\sum_i X_{ik} T_{ij}}{\sum_i X_{ik} \widetilde{T}_{ij}}$$

非負値行列因子分解のまとめ

特徴

- 行列の掛け算を繰り返す計算

  １反復当たりの計算量： $O(IJK)$

- 1 反復当たりの計算量はデータの 1 乗に比例

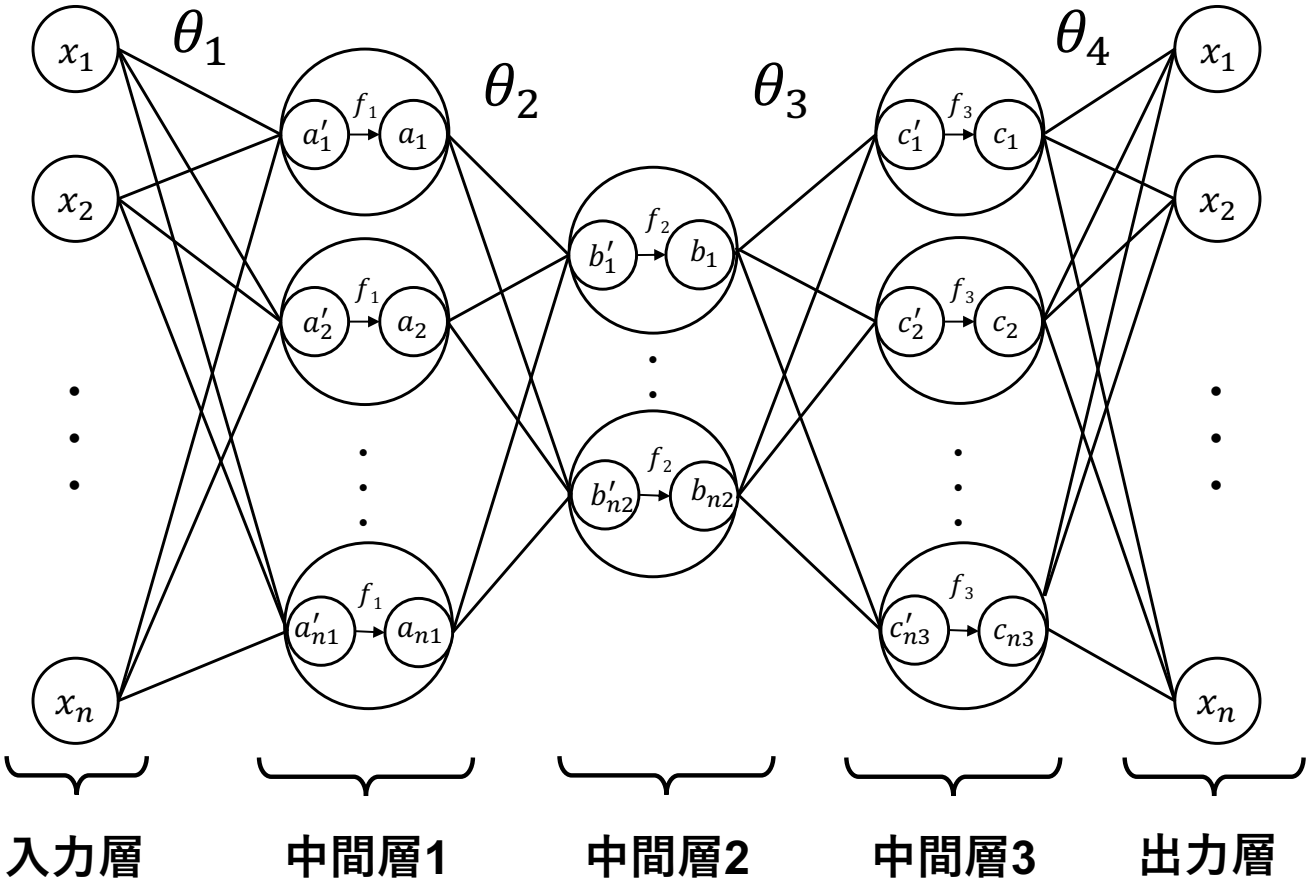
  ※ データ数を $D := IJ$ として K を固定すると $O(D)$

- 多くの手法よりも高速に計算できる。
- ユーザとアイテムの両側で特徴量化を行うことができる
- 非負の値を取るため解釈がしやすい。

オートエンコーダー（自己符号化）

ニューラルネットワークを使った特徴抽出

- 入力 $\boldsymbol{x}_d$ から出力 $\boldsymbol{x}_d$ を作るニューラルネットワーク
- 中間層の次元を小さくしておき、その情報を復元して出力 $\boldsymbol{x}_d$ を作る
- 中間層の情報は $\boldsymbol{x}_d$ の主要な情報となるので、これが特徴抽出となる



ベクトルの変換

$\boldsymbol{x} \in \mathbb{R}^n$ 入力層

$\downarrow$

$\boldsymbol{a}' \in \mathbb{R}^{n_1}$ 中間層１の入力

$\downarrow$

$\boldsymbol{a} \in \mathbb{R}^{n_1}$ 中間層１の出力

$\downarrow$

$\boldsymbol{b}' \in \mathbb{R}^{n_2}$ 中間層２の入力

$\downarrow$

$\boldsymbol{b} \in \mathbb{R}^{n_2}$ 中間層２の出力

$\downarrow$

$\boldsymbol{c}' \in \mathbb{R}^{n_3}$ 中間層３の入力

$\downarrow$

$\boldsymbol{c} \in \mathbb{R}^{n_3}$ 中間層３の出力

$\downarrow$

$\hat{\boldsymbol{x}} \in \mathbb{R}^n$ 出力層

$$\boldsymbol{a}' = f_{\theta_1}(\boldsymbol{x}) := \boldsymbol{M}_1\boldsymbol{x} + \boldsymbol{v}_1$$

$\boldsymbol{a} = g_1(\boldsymbol{a}')$ 活性化関数

$$\boldsymbol{b}' = f_{\theta_2}(\boldsymbol{a}) := \boldsymbol{M}_2\boldsymbol{a} + \boldsymbol{v}_2$$

$\boldsymbol{b} = g_2(\boldsymbol{b}')$ 活性化関数

$$\boldsymbol{c}' = f_{\theta_3}(\boldsymbol{b}) := \boldsymbol{M}_3\boldsymbol{b} + \boldsymbol{v}_3$$

$\boldsymbol{c} = g_3(\boldsymbol{c}')$ 活性化関数

$$\hat{\boldsymbol{x}} = f_{\theta_4}(\boldsymbol{c}) := \boldsymbol{M}_4\boldsymbol{c} + \boldsymbol{v}_4$$

自己符号化関数と誤差関数

自己符号化関数

自己符号化関数 $f : \mathbb{R}^n \to \mathbb{R}^n$

$$f(\boldsymbol{x}) = f_{\theta_4}(g_3(f_{\theta_3}(g_2(f_{\theta_2}(g_1(f_{\theta_1}(\boldsymbol{x})))))))$$

パラメタは $\boldsymbol{\theta} := \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3, \boldsymbol{\theta}_4\}$

データ $\{\boldsymbol{x}_d\}_{d \in [D]}$ に対する予測誤差を定義する

誤差関数

$\boldsymbol{x}$：目標、$\hat{\boldsymbol{x}} := f(\boldsymbol{x})$：予測

$$L(\boldsymbol{x}, \hat{\boldsymbol{x}}) := \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^2$$

学習

学習データ：$\{\boldsymbol{x}_d\}_{d\in[D]}$

学習

パラメタ $\boldsymbol{\theta} := \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3, \boldsymbol{\theta}_4\}$

$$\min_{\boldsymbol{\theta}} \frac{1}{D} \sum_{d\in[D]} L(\boldsymbol{x}_d, f_{\theta_4}(g_3(f_{\theta_3}(g_2(f_{\theta_2}(g_1(f_{\theta_1}(\boldsymbol{x}_d)))))))))$$

$F_d(\boldsymbol{\theta}) := L(\boldsymbol{x}_d, f_{\theta_4}(g_3(f_{\theta_3}(g_2(f_{\theta_2}(g_1(f_{\theta_1}(\boldsymbol{x}_d)))))))))$ とすると、

$$\min_{\boldsymbol{\theta}} \frac{1}{D} \sum_{d\in[D]} F_d(\boldsymbol{\theta})$$

学習アルゴリズム

確率的勾配降下法やその改良アルゴリズムで最適なパラメタ $\boldsymbol{\theta}$ を求める。

誤差逆伝播 1,2

$$F_d \overset{L}{\leftarrow} \hat{\boldsymbol{x}} \overset{f_{\theta_4}}{\leftarrow} \boldsymbol{c} \overset{g_3}{\leftarrow} \boldsymbol{c}' \overset{f_{\theta_3}}{\leftarrow} \boldsymbol{b} \overset{g_2}{\leftarrow} \boldsymbol{b}' \overset{f_{\theta_2}}{\leftarrow} \boldsymbol{a} \overset{g_1}{\leftarrow} \boldsymbol{a}' \overset{f_{\theta_1}}{\leftarrow} \boldsymbol{x}_d$$

連鎖律

$$\frac{\partial F_d}{\partial \boldsymbol{\theta}_1} = \frac{\partial F_d}{\partial \hat{\boldsymbol{x}}} \frac{\partial \hat{\boldsymbol{x}}}{\partial \boldsymbol{c}} \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{c}'} \frac{\partial \boldsymbol{c}'}{\partial \boldsymbol{b}} \frac{\partial \boldsymbol{b}}{\partial \boldsymbol{b}'} \frac{\partial \boldsymbol{b}'}{\partial \boldsymbol{a}} \frac{\partial \boldsymbol{a}}{\partial \boldsymbol{a}'} \frac{\partial \boldsymbol{a}'}{\partial \boldsymbol{\theta}_1}$$

$$\frac{\partial F_d}{\partial \boldsymbol{\theta}_2} = \frac{\partial F_d}{\partial \hat{\boldsymbol{x}}} \frac{\partial \hat{\boldsymbol{x}}}{\partial \boldsymbol{c}} \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{c}'} \frac{\partial \boldsymbol{c}'}{\partial \boldsymbol{b}} \frac{\partial \boldsymbol{b}}{\partial \boldsymbol{b}'} \frac{\partial \boldsymbol{b}'}{\partial \boldsymbol{\theta}_2}$$

$$\frac{\partial F_d}{\partial \boldsymbol{\theta}_3} = \frac{\partial F_d}{\partial \hat{\boldsymbol{x}}} \frac{\partial \hat{\boldsymbol{x}}}{\partial \boldsymbol{c}} \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{c}'} \frac{\partial \boldsymbol{c}'}{\partial \boldsymbol{\theta}_3}$$

$$\frac{\partial F_d}{\partial \boldsymbol{\theta}_4} = \frac{\partial F_d}{\partial \hat{\boldsymbol{x}}} \frac{\partial \hat{\boldsymbol{x}}}{\partial \boldsymbol{\theta}_4}$$

各パーツの微分は簡単に計算できる

誤差逆伝播 3

$$\frac{\partial F_d}{\partial \boldsymbol{\theta}_1} = \frac{\partial F_d}{\partial \hat{\boldsymbol{x}}} \frac{\partial \hat{\boldsymbol{x}}}{\partial \boldsymbol{c}} \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{c}'} \frac{\partial \boldsymbol{c}'}{\partial \boldsymbol{b}} \frac{\partial \boldsymbol{b}}{\partial \boldsymbol{b}'} \frac{\partial \boldsymbol{b}'}{\partial \boldsymbol{a}} \frac{\partial \boldsymbol{a}}{\partial \boldsymbol{a}'} \frac{\partial \boldsymbol{a}'}{\partial \boldsymbol{\theta}_1}$$

ノード数を全て n としてバイアス項を除いたとき、

$$\frac{\partial F_d}{\partial \hat{\boldsymbol{x}}} \in \mathbb{R}^{1\times n}, \qquad \frac{\partial \hat{\boldsymbol{x}}}{\partial \boldsymbol{c}}, \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{c}'}, \frac{\partial \boldsymbol{c}'}{\partial \boldsymbol{b}}, \frac{\partial \boldsymbol{b}}{\partial \boldsymbol{b}'}, \frac{\partial \boldsymbol{b}'}{\partial \boldsymbol{a}}, \frac{\partial \boldsymbol{a}}{\partial \boldsymbol{a}'} \in \mathbb{R}^{n\times n}, \qquad \frac{\partial \boldsymbol{a}'}{\partial \boldsymbol{\theta}_1} \in \mathbb{R}^{n\times n^2}$$

計算量

- 入力の方から計算（連鎖律の右から計算）
  $n \times n$ のベクトルと $n \times n^2$ の行列の積： $O(n^4)$
- 出力の方から計算（連鎖律の左から計算）
  $1 \times n$ のベクトルと $n \times n$ の行列の積： $O(n^2)$

  ※　最後の $\dfrac{\partial \boldsymbol{a}'}{\partial \boldsymbol{\theta}_1}$ の掛け算は疎性を使うと簡単に計算可能

誤差逆伝播法 4

行列 $\dfrac{\partial \boldsymbol{a}'}{\partial \boldsymbol{\theta}_1}$ の疎性（非ゼロ要素の割合）

行と列の意味

- $\boldsymbol{\theta}_1$ は $n \times n$ の行列を適当な順で並び替えたベクトル
- $\boldsymbol{a}'$ は中間層１の入力となるベクトル

$\boldsymbol{\theta}_1$ の一つの成分に対して、影響を与える $\boldsymbol{a}'$ は一つの成分

$\Longrightarrow$ $\dfrac{\partial \boldsymbol{a}'}{\partial \boldsymbol{\theta}_1}$ の各列に非ゼロ要素は一つ

$\Longrightarrow$ $\boldsymbol{v} \in \mathbb{R}^n$ に対し、$\boldsymbol{v}^T \dfrac{\partial \boldsymbol{a}'}{\partial \boldsymbol{\theta}_1}$ の計算に必要な計算量は $O(n^2)$

（非ゼロの部分だけ計算すればよい）

疎性があるため行列とベクトルの掛け算は簡単

誤差逆伝播法 5

$$\frac{\partial F_d}{\partial \boldsymbol{\theta}_1} = \frac{\partial F_d}{\partial \hat{\boldsymbol{x}}} \frac{\partial \hat{\boldsymbol{x}}}{\partial \boldsymbol{c}} \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{c}'} \frac{\partial \boldsymbol{c}'}{\partial \boldsymbol{b}} \frac{\partial \boldsymbol{b}}{\partial \boldsymbol{b}'} \frac{\partial \boldsymbol{b}'}{\partial \boldsymbol{a}} \frac{\partial \boldsymbol{a}}{\partial \boldsymbol{a}'} \frac{\partial \boldsymbol{a}'}{\partial \boldsymbol{\theta}_1}$$

$$\frac{\partial F_d}{\partial \boldsymbol{\theta}_2} = \frac{\partial F_d}{\partial \hat{\boldsymbol{x}}} \frac{\partial \hat{\boldsymbol{x}}}{\partial \boldsymbol{c}} \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{c}'} \frac{\partial \boldsymbol{c}'}{\partial \boldsymbol{b}} \frac{\partial \boldsymbol{b}}{\partial \boldsymbol{b}'} \frac{\partial \boldsymbol{b}'}{\partial \boldsymbol{\theta}_2}$$

$$\frac{\partial F_d}{\partial \boldsymbol{\theta}_3} = \frac{\partial F_d}{\partial \hat{\boldsymbol{x}}} \frac{\partial \hat{\boldsymbol{x}}}{\partial \boldsymbol{c}} \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{c}'} \frac{\partial \boldsymbol{c}'}{\partial \boldsymbol{\theta}_3}$$

$$\frac{\partial F_d}{\partial \boldsymbol{\theta}_4} = \frac{\partial F_d}{\partial \hat{\boldsymbol{x}}} \frac{\partial \hat{\boldsymbol{x}}}{\partial \boldsymbol{\theta}_4}$$

出力の方から順に計算すると、同じ計算が出てくるため省略できる

勾配の計算に必要な計算量　（s：中間層の層数、n：各層の（平均）ノード数）

- 入力の方から順に計算（連鎖律の右から計算）： $O(s^2 n^4)$
- 出力の方から順に計算（連鎖律の左から計算）： $O(sn^2)$

出力の方から計算する手法を誤差逆伝播法（バックプロパゲーション）という。

パラメタの数 $O(sn^2)$ 個に比例した計算量で勾配の計算が可能で高速。

オートエンコーダーのまとめ

特徴

- データ数よりも、中間層の層数やノード数の方が計算時間に影響

  計算量はパラメタ数と反復回数に比例
- パラメタ数を増やすと勾配の計算が困難
- 中間層の数やノード数などのハイパーパラメータを適切に決める必要がある
- 深層学習のフレームワークで計算できるため、ソフトウェアが充実

t-SNE

高次元のベクトル：$\{\boldsymbol{x}_d\}_{d\in[D]}$　　$\boldsymbol{x}_d \in \mathbb{R}^n$

$$\Downarrow$$

低次元のベクトル：$\{\boldsymbol{y}_d\}_{d\in[D]}$　　$\boldsymbol{y}_d \in \mathbb{R}^2$

※ 特徴量の可視化を想定しているため $\boldsymbol{y}_d \in \mathbb{R}^2$ を考える。
　アルゴリズムとしては任意の次元のベクトルを構築することが可能。

アイデア：
ベクトルの近傍が保存されることを目指し、高次元空間上の２つのベクトル間の類似度と対応する低次元空間上の２つのベクトルの類似度が変わらないように変換。

論点

- 高次元空間と低次元空間での類似度の定義
- 「高次元空間での類似性」と「低次元空間での類似性」間の近さを示す距離

学習モデル

- P_{ij}：$\boldsymbol{x}_i$ と $\boldsymbol{x}_j$ の類似度
- Q_{ij}：$\boldsymbol{y}_i$ と $\boldsymbol{y}_j$ の類似度
- $f(\boldsymbol{P}, \boldsymbol{Q})$：類似度行列 $\boldsymbol{P}, \boldsymbol{Q} \in \mathbb{R}^{D \times D}$ 間の距離

学習モデル

$\boldsymbol{y}_d$ $(d \in [D])$ を変数とする制約なし最適化：

$$\min_{\{\boldsymbol{y}_d\}_{d \in [D]}} \quad f(\boldsymbol{P}, \boldsymbol{Q})$$

※ $\boldsymbol{Q}$ は $\boldsymbol{y}_d$ から計算される行列

次の順に説明を行う。

- SNE(Stochastic Neighbor Embedding)
- t-SNE(t-Distributed Stocahsic Neighbor Embbeddig)

以下では、$\sum$ で使う添字 $i, j, k, l \in [D]$ に対し、$[D]$ は省略することにする。

例：$\displaystyle\sum_{i(i \neq j)}$ は $\displaystyle\sum_{i \in [D]-\{j\}}$ の意味, $\displaystyle\sum_{i,j(i \neq j)}$ は $\displaystyle\sum_{i \in [D]} \sum_{j \in [D]-\{i\}}$ の意味

SNE

類似度：正規分布を利用した条件付き確率

$$P_{ij}: \qquad p_{j|i} := \frac{\exp\{-\|\boldsymbol{x}_j - \boldsymbol{x}_i\|^2/2\sigma^2\}}{\sum_{k(k\neq i)} \exp\{-\|\boldsymbol{x}_k - \boldsymbol{x}_i\|^2/2\sigma^2\}} \qquad i,j \in [D] \quad (i \neq j)$$

$$Q_{ij}: \qquad q_{j|i} := \frac{\exp\{-\|\boldsymbol{y}_j - \boldsymbol{y}_i\|^2\}}{\sum_{k(k\neq i)} \exp\{-\|\boldsymbol{y}_k - \boldsymbol{y}_i\|^2\}} \qquad i,j \in [D] \quad (i \neq j)$$

- データのばらつきを表す σ は、データから適切に決める。
- $\boldsymbol{p}_i := \{p_{j|i}\}_{j\in[D]-\{i\}}$, $\boldsymbol{q}_i := \{q_{j|i}\}_{j\in[D]-\{i\}}$ は確率分布

類似度間の距離：KL ダイバージェンス [5] の和

$$C := \sum_i KL(\boldsymbol{p}_i||\boldsymbol{q}_i) = \sum_i \sum_{j(j\neq i)} p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}$$

[5] KL ダイバージャンスは確率分布間の距離の一つ

学習アルゴリズム

勾配：　　計算過程は次スライド

$$\nabla_{\boldsymbol{y}_i} C = 2 \sum_{j(j \neq i)} (p_{j|i} + p_{i|j} - q_{j|i} - q_{i|j})(\boldsymbol{y}_i - \boldsymbol{y}_j)$$

最急降下法にモメンタム項を加えた反復解法で計算をする。

学習アルゴリズム

ステップ０　$\boldsymbol{x}_i$ と $\boldsymbol{x}_j$ の類似度 $p_{j|i}$ の計算　　$i, j \in [D],\ (i \neq j)$

$t = 0, \quad \boldsymbol{y}_i^t \in \mathbb{R}^2\ (i \in [D])$ の初期値を設定

ステップ１　$\boldsymbol{y}_i^t$ と $\boldsymbol{y}_j^t$ の類似度 $q_{j|i}$ の計算　　$i, j \in [D],\ (i \neq j)$

ステップ２　勾配 $\nabla_{\boldsymbol{y}_i} C$ の計算　　$i \in [D]$

ステップ３　$\boldsymbol{y}_i^t$ の更新（α, β^t はステップサイズ）

$$\boldsymbol{y}_i^{t+1} := \boldsymbol{y}_i^t - \alpha \nabla_{\boldsymbol{y}_i} C + \beta^t (\boldsymbol{y}_i^t - \boldsymbol{y}_i^{t-1}) \qquad i \in [D]$$

ステップ４　$t := t + 1$ として、ステップ１へ。

※ $\beta^t = 0$ とすると最急降下法

式変形 1

$$
\begin{aligned}
C &:= \sum_{i,j(i\neq j)} p_{j|i} \log \frac{p_{j|i}}{q_{j|i}} \\
&= \sum_{i,j(i\neq j)} p_{j|i} (\log p_{j|i} - \log q_{j|i}) \\
&= \sum_{i,j(i\neq j)} p_{j|i} \left(\log p_{j|i} \ + \ \|\boldsymbol{y}_j - \boldsymbol{y}_i\|^2 \ + \ \log \sum_{k(k\neq i)} \exp\{-\|\boldsymbol{y}_k - \boldsymbol{y}_i\|^2\} \right) \\
&= \sum_{i,j(i\neq j)} p_{j|i} \log p_{j|i} \ + \ \sum_{i,j(i\neq j)} p_{j|i} \|\boldsymbol{y}_j - \boldsymbol{y}_i\|^2 \\
&\quad + \ \sum_{i} \log \sum_{k(k\neq i)} \exp\{-\|\boldsymbol{y}_k - \boldsymbol{y}_i\|^2\} \qquad (\because \sum_{j(j\neq i)} p_{j|i} = 1)
\end{aligned}
$$

式変形２

$\boldsymbol{y}_s$ での微分 $\dfrac{\partial C}{\partial \boldsymbol{y}_s}$ を考える。C の中で $\boldsymbol{y}_s$ に依存する部分は次の４つ。

- $C_1 := \displaystyle\sum_{j(j\neq s)} p_{j|s}\ \|\boldsymbol{y}_j - \boldsymbol{y}_s\|^2$ $\cdots$ 2 項目の $j = s$ の場合
- $C_2 := \displaystyle\sum_{i(i\neq s)} p_{s|i}\ \|\boldsymbol{y}_s - \boldsymbol{y}_i\|^2$ $\cdots$ 2 項目の $i = s$ の場合
- $C_3 := \log \displaystyle\sum_{k(k\neq s)} \exp\{-\|\boldsymbol{y}_k - \boldsymbol{y}_s\|^2\}$ $\cdots$ 3 項目の $i = s$ の場合
- $C_4 := \displaystyle\sum_{i(i\neq s)} \log \sum_{k(k\neq i)} \exp\{-\|\boldsymbol{y}_k - \boldsymbol{y}_i\|^2\}$ $\cdots$ 3 項目の $i \neq s$ の場合

それぞれを $\boldsymbol{y}_s$ で微分する。

$$\frac{\partial C_1}{\partial \boldsymbol{y}_s} = 2 \sum_{j(j\neq s)} p_{j|s}\ (\boldsymbol{y}_s - \boldsymbol{y}_j)^T, \qquad \frac{\partial C_2}{\partial \boldsymbol{y}_s} = 2 \sum_{i(i\neq s)} p_{s|i}\ (\boldsymbol{y}_s - \boldsymbol{y}_i)^T$$

式変形３

$$\frac{\partial C_3}{\partial \boldsymbol{y}_s} = \frac{1}{\sum_{k(k\neq s)} \exp\{-\|\boldsymbol{y}_k - \boldsymbol{y}_s\|^2\}} \sum_{k(k\neq s)} \exp\{-\|\boldsymbol{y}_k - \boldsymbol{y}_s\|^2\}\ (-2)\ (\boldsymbol{y}_s - \boldsymbol{y}_k)^T$$

$$= -2 \sum_{k(k\neq s)} q_{k|s}\ (\boldsymbol{y}_s - \boldsymbol{y}_k)^T$$

$$\frac{\partial C_4}{\partial \boldsymbol{y}_s} = \sum_{i(i\neq s)} \frac{1}{\sum_{k(k\neq i)} \exp\{-\|\boldsymbol{y}_k - \boldsymbol{y}_i\|^2\}} \exp\{-\|\boldsymbol{y}_s - \boldsymbol{y}_i\|^2\}(-2)(\boldsymbol{y}_s - \boldsymbol{y}_i)^T$$

$$= -2 \sum_{i(i\neq s)} q_{s|i}\ (\boldsymbol{y}_s - \boldsymbol{y}_i)^T$$

よって、

$$\nabla_{\boldsymbol{y}_s} C = \left(\frac{\partial C}{\partial \boldsymbol{y}_s}\right)^T = \left(\frac{\partial C_1}{\partial \boldsymbol{y}_s} + \frac{\partial C_2}{\partial \boldsymbol{y}_s} + \frac{\partial C_3}{\partial \boldsymbol{y}_s} + \frac{\partial C_4}{\partial \boldsymbol{y}_s}\right)^T$$

$$= 2 \sum_{j(j\neq s)} (p_{j|s} + p_{s|j} - q_{j|s} - q_{s|j})(\boldsymbol{y}_s - \boldsymbol{y}_j)$$

t-SNE

t-SNE(t-Distributed Stocahsic Neighbor Embbeddig) では、

(1) 低次元空間での確率分布を正規分布でなく t 分布にする

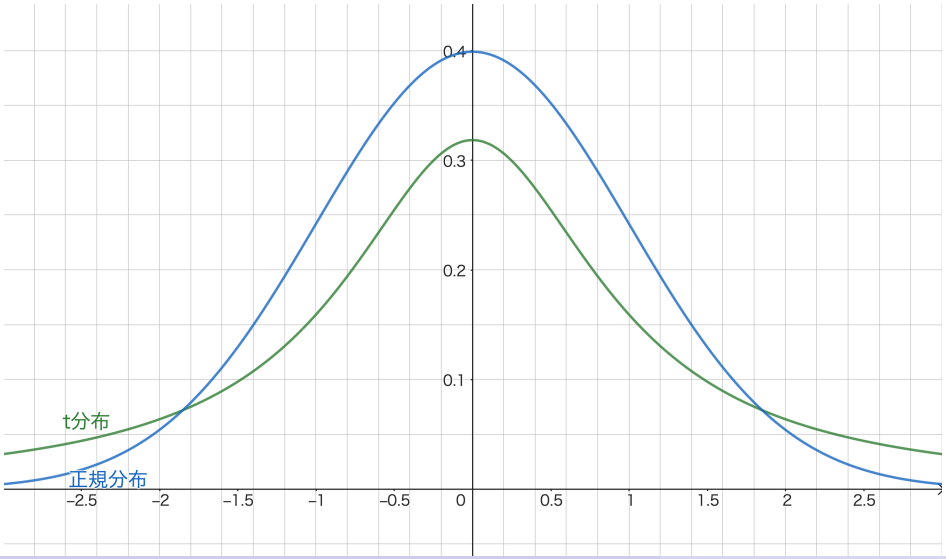
(2) 同時確率を考えることにより対称化を行う

(1) t 分布の導入

高次元のベクトルを低次元に埋め込むと crowding という問題が生じる。
正規分布よりも裾が厚い t 分布を使うことにより、類似していないデータを遠くに置くことを許容できるようになる。

自由度１の t 分布：

$$f(x) := \frac{1}{\pi(1+x^2)}$$



t 分布の導入

類似度：t 分布による条件付き確率

$$Q_{ij}: \qquad q_{j|i} := \frac{(1+\|\boldsymbol{y}_j - \boldsymbol{y}_i\|^2)^{-1}}{\sum_{k(k\neq i)} (1+\|\boldsymbol{y}_k - \boldsymbol{y}_i\|^2)^{-1}}$$

この場合の勾配は次のようになる。

$$\nabla_{\boldsymbol{y}_i} C = 2 \sum_{j(j\neq i)} \frac{p_{j|i} + p_{i|j} - q_{j|i} - q_{i|j}}{1+\|\boldsymbol{y}_j - \boldsymbol{y}_i\|^2} (\boldsymbol{y}_i - \boldsymbol{y}_j)$$

導出過程は省略するが、計算の手順は SNE や t-SNE の場合と同じ。

(2) 同時確率による対称化

条件付き確率 $p_{j|i}$ の代わりに同時確率 p_{ij} を考える

$$P_{ij}: \qquad p_{ij} := \frac{\exp\{-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2/2\sigma^2\}}{\sum_{k,l(k\neq l)} \exp\{-\|\boldsymbol{x}_k - \boldsymbol{x}_l\|^2/2\sigma^2\}} \qquad i,j \in [D],\ (i \neq j)$$

$$Q_{ij}: \qquad q_{ij} := \frac{(1+\|\boldsymbol{y}_i - \boldsymbol{y}_j\|^2)^{-1}}{\sum_{k,l(k\neq l)} (1+\|\boldsymbol{y}_k - \boldsymbol{y}_l\|^2)^{-1}} \qquad i,j \in [D],\ (i \neq j)$$

- $p_{ij} = p_{ji},\ q_{ij} = q_{ji}$ となり類似度に対称性がある。
- $\boldsymbol{P}, \boldsymbol{Q}$ は確率分布

類似度間の距離：KL ダイバージェンス

$$C := KL(\boldsymbol{P}||\boldsymbol{Q}) = \sum_{i,j(i\neq j)} p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

工夫を加えた対称化

$\boldsymbol{x}_i$ が他の点から遠く外れていた場合、同時確率

$$p_{ij} := \frac{\exp\{-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2/2\sigma^2\}}{\sum_{k,l(k\neq l)} \exp\{-\|\boldsymbol{x}_k - \boldsymbol{x}_l\|^2/2\sigma^2\}}$$

は小さくなり過ぎる。そのため、代わりに次の確率を使うとよい（らしい）。

$$p_{j|i} := \frac{\exp\{-\|\boldsymbol{x}_j - \boldsymbol{x}_i\|^2/2\sigma^2\}}{\sum_{k(k\neq i)} \exp\{-\|\boldsymbol{x}_k - \boldsymbol{x}_i\|^2/2\sigma^2\}} \quad \text{条件付き確率}$$

$$p_{ij} := \frac{p_{j|i} + p_{i|j}}{2D} \quad \text{対称化}$$

- $p_{ij} = p_{ji}$ であるため、類似度に対称性がある。
- $\boldsymbol{P}$ を確率分布にするため D で割っている。
- q_{ij} は変えない（変えると勾配の計算が困難）。

勾配

この場合の勾配は次のようになる。

$$\nabla_{\boldsymbol{y}_i} C = 4 \sum_{j(j \neq i)} \frac{p_{ij} - q_{ij}}{1 + \|\boldsymbol{y}_i - \boldsymbol{y}_j\|^2} (\boldsymbol{y}_i - \boldsymbol{y}_j)$$

目的関数：

$$\begin{aligned}
C &:= \sum_{i,j(i \neq j)} p_{ij} \log \frac{p_{ij}}{q_{ij}} \\
&= \sum_{i,j(i \neq j)} p_{ij} \left(\log p_{ij} + \log(1 + \|\boldsymbol{y}_i - \boldsymbol{y}_j\|^2) + \log \sum_{k,l(k \neq l)} (1 + \|\boldsymbol{y}_k - \boldsymbol{y}_l\|^2)^{-1} \right) \\
&= \sum_{i,j(i \neq j)} p_{ij} \log p_{ij} \quad + \quad \sum_{i,j(i \neq j)} p_{ij} \log(1 + \|\boldsymbol{y}_i - \boldsymbol{y}_j\|^2) \\
&\qquad + \quad \log \sum_{k,l(k \neq l)} (1 + \|\boldsymbol{y}_k - \boldsymbol{y}_l\|^2)^{-1} \qquad\qquad (\because \sum_{i,j(i \neq j)} p_{ij} = 1)
\end{aligned}$$

式変形 1

$\boldsymbol{y}_s$ で微分することを考える。$\boldsymbol{y}_s$ に依存する部分は次の 3 つ。

- $C_1 := \sum_{j(j \neq s)} p_{sj} \log(1 + \|\boldsymbol{y}_s - \boldsymbol{y}_j\|^2)$ $\cdots$ 2 項目の $i = s$ の場合
- $C_2 := \sum_{i(i \neq s)} p_{is} \log(1 + \|\boldsymbol{y}_i - \boldsymbol{y}_s\|^2)$ $\cdots$ 2 項目の $j = s$ の場合
- $C_3 := \log \sum_{k,l(k \neq l)} (1 + \|\boldsymbol{y}_k - \boldsymbol{y}_l\|^2)^{-1}$ $\cdots$ 3 項目

それぞれを微分する。

$$\frac{\partial C_1}{\partial \boldsymbol{y}_s} = \sum_{j(j \neq s)} p_{sj} \frac{1}{1 + \|\boldsymbol{y}_s - \boldsymbol{y}_j\|^2} \, 2(\boldsymbol{y}_s - \boldsymbol{y}_j)^T$$

$$\frac{\partial C_2}{\partial \boldsymbol{y}_s} = \sum_{i(i \neq s)} p_{is} \frac{1}{1 + \|\boldsymbol{y}_i - \boldsymbol{y}_s\|^2} \, 2(\boldsymbol{y}_s - \boldsymbol{y}_i)^T$$

式変形 2

$$
\begin{aligned}
\frac{\partial C_3}{\partial \boldsymbol{y}_s} &= \frac{1}{\sum_{k,l(k\neq l)} (1+\|\boldsymbol{y}_k-\boldsymbol{y}_l\|^2)^{-1}} \frac{\partial}{\partial \boldsymbol{y}_s} \sum_{k,l(k\neq l)} (1+\|\boldsymbol{y}_k-\boldsymbol{y}_l\|^2)^{-1} \\
&= \frac{1}{\sum_{k,l(k\neq l)} (1+\|\boldsymbol{y}_k-\boldsymbol{y}_l\|^2)^{-1}} \; 2 \; \frac{\partial}{\partial \boldsymbol{y}_s} \sum_{l(l\neq s)} (1+\|\boldsymbol{y}_s-\boldsymbol{y}_l\|^2)^{-1} \\
&= \frac{1}{\sum_{k,l(k\neq l)} (1+\|\boldsymbol{y}_k-\boldsymbol{y}_l\|^2)^{-1}} \; 2 \sum_{l(l\neq s)} -(1+\|\boldsymbol{y}_s-\boldsymbol{y}_l\|^2)^{-2} \; 2(\boldsymbol{y}_s-\boldsymbol{y}_l)^T \\
&= -4 \sum_{l(l\neq t)} \frac{q_{sl}}{1+\|\boldsymbol{y}_s-\boldsymbol{y}_l\|^2} (\boldsymbol{y}_s-\boldsymbol{y}_l)^T
\end{aligned}
$$

よって、

$$
\begin{aligned}
\nabla_{\boldsymbol{y}_s} C = \left(\frac{\partial C}{\partial \boldsymbol{y}_s}\right)^T &= \left(\frac{\partial C_1}{\partial \boldsymbol{y}_s} + \frac{\partial C_2}{\partial \boldsymbol{y}_s} + \frac{\partial C_3}{\partial \boldsymbol{y}_s}\right)^T \\
&= 4 \sum_{j(j\neq s)} \frac{p_{sj}-q_{sj}}{1+\|\boldsymbol{y}_s-\boldsymbol{y}_j\|^2} (\boldsymbol{y}_s-\boldsymbol{y}_j)
\end{aligned}
$$

学習アルゴリズム

ステップ０ $$p_{j|i} := \frac{\exp\{-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2/2\sigma^2\}}{\sum_{k(k\neq i)} \exp\{-\|\boldsymbol{x}_i - \boldsymbol{x}_k\|^2/2\sigma^2\}} \qquad i,j \in [D],\ (i \neq j)$$

$$p_{ij} := \frac{p_{j|i} + p_{i|j}}{2D} \qquad i,j \in [D],\ (i \neq j)$$

$t = 0,\quad \boldsymbol{y}_i^t \in \mathbb{R}^2\ (i \in [D])$ の初期値を設定

ステップ１ 類似度の計算

$$q_{ij} := \frac{(1 + \|\boldsymbol{y}_i^t - \boldsymbol{y}_j^t\|^2)^{-1}}{\sum_{k,l(k\neq l)} (1 + \|\boldsymbol{y}_k^t - \boldsymbol{y}_l^t\|^2)^{-1}} \qquad i,j \in [D],\ (i \neq j)$$

ステップ２ 勾配の計算

$$\nabla_{\boldsymbol{y}_i} C = 4 \sum_{j(j\neq i)} \frac{p_{ij} - q_{ij}}{1 + \|\boldsymbol{y}_i^t - \boldsymbol{y}_j^t\|^2} (\boldsymbol{y}_i^t - \boldsymbol{y}_j^t) \qquad i \in [D]$$

ステップ３ $\boldsymbol{y}_i^t$ の更新（α, β^t はステップサイズ）

$$\boldsymbol{y}_i^{t+1} := \boldsymbol{y}_i^t - \alpha \nabla_{\boldsymbol{y}_i} C + \beta^t (\boldsymbol{y}_i^t - \boldsymbol{y}_i^{t-1}) \qquad i \in [D]$$

ステップ４ $t := t + 1$ として、ステップ１へ。

t-SNE のまとめ

特徴

- 人間の直感に合うような可視化となることが多い。

  もちろん、必ず良いものが出来るということではない。
- 非線形な変換を伴った可視化ツールとしてよく使われる。

  データの特徴抽出法としても時々使われる。
- １反復あたりの計算量はデータ数の 2 乗に比例

  計算量：$O(D^2)$

  ※ $\boldsymbol{y} \in \mathbb{R}^k$ のとき、$O(D^2 k)$ となる。
- データ量が増えると計算が困難になるが、カーネル主成分分析ほどではない