コンピュータビジョン

担当: 井尻 敬

6/3更新

Contents

Contents

主成分分析

- データ群から最もばらつきの大きな軸を見つける
- データの次元圧縮に利用できる
- パターン認識，画像処理，そのほか様々な分野で使われる

主成分分析(Principal Component Analysis)

これなら分かる応用数学教室（p. 205）

『統計データから互いに無関係の因子を取り出して，観測値をそれらの因子の線形結合で説明することを主成分分析と呼び，取り出された因子を主成分と呼ぶ』

ディジタル画像処理（ p. 273）

『高次元特徴空間に分散する多数の学習用入力画像から，分布をよく表現できる低次元の特徴空間を求める手法』

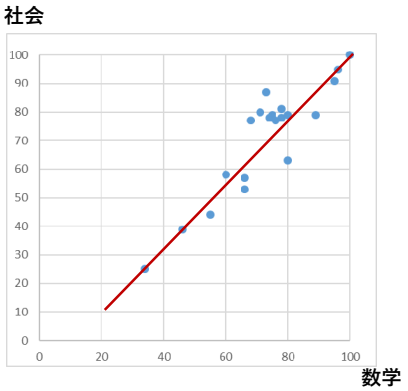
Wikipedia (2018/05/23)

『相関のある多数の変数から相関のない少数で全体のばらつきを最もよく表す主成分と呼ばれる変数を合成する多変量解析の一手法』

主成分分析

ある20人のテスト点数とその散布図 (横:数学 縦:社会)が下図の通り

数学	社会
80	63
95	91
100	100
66	53
34	25
89	79
96	95
78	78
55	44
60	58
46	39
80	79
73	87
68	77
75	79
76	77
66	57
78	81
74	78
71	80



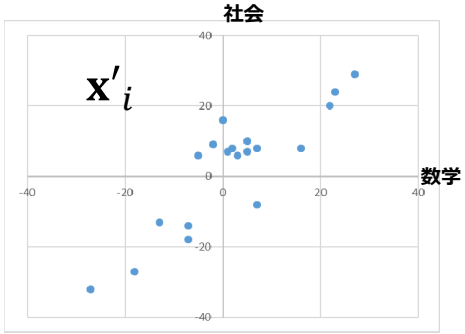
最もばらつきの大きな方向を考えてみる

※井尻が適当に作った 嘘 データ です

主成分分析

- 入力データ：$\mathbf{x}_i \in \boldsymbol{R}^2, i = 1,2,\dots,N$
- 平均が原点となるよう平行移動する

$$\mathbf{x'}_i = \mathbf{x}_i - \frac{\mathbf{1}}{N}\sum_i \mathbf{x}_i$$



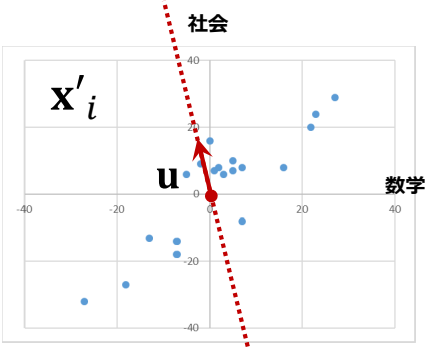
主成分分析

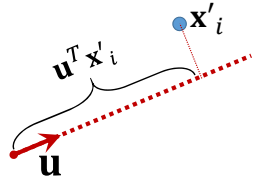
- ある単位ベクトル $\mathbf{u}$ を考える
- $\mathbf{u}$にデータ点を射影した距離の2乗平均は

$$\frac{1}{N}\sum_i (\mathbf{u}^T\mathbf{x'}_i)^2$$

これを最大化する $\mathbf{u}$ を探す！ ※計算法後述

→最もデータがばらつく方向が分かる





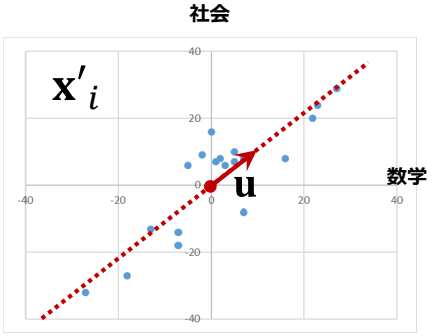
主成分分析

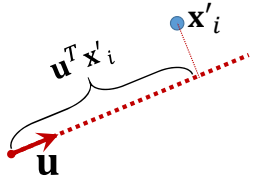
- ある単位ベクトル $\mathbf{u}$ を考える
- $\mathbf{u}$にデータ点を射影した距離の2乗平均は

$$\frac{1}{N}\sum_i (\mathbf{u}^T \mathbf{x}'_i)^2$$

これを最大化する $\mathbf{u}$ を探す！　※計算法後述

→最もデータがばらつく方向が分かる





主成分分析

例）右表のデータに対して，

$$\frac{1}{N}\sum_i (\mathbf{u}^T \mathbf{x}'_i)^2$$

を最大化する$\mathbf{u}$ を計算すると(方法は後述します)

$$\mathbf{u} = (0.63, 0.77)$$

が得られた．この方向$\mathbf{u}$を**第一主成分**と呼ぶ

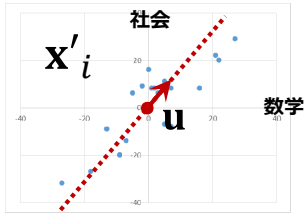
各データを$\mathbf{u}$ に射影する

(数学, 社会) の点が (80, 63)なら，

(数学, 社会) の平均値は(73, 71)なので

射影値 = (80-73)*0.63 +(63-71)*0.77

=-1.8

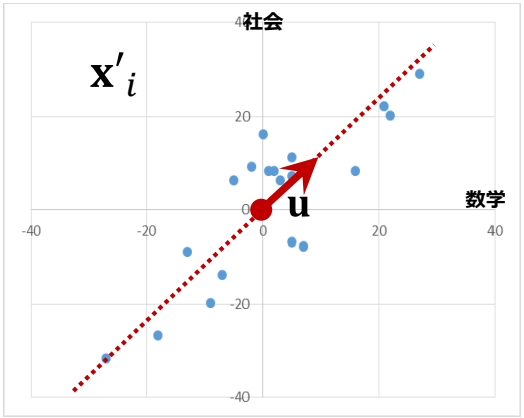


数学	社会	第1主成分得点
80	63	−1.8
95	91	29.3
100	100	39.3
66	53	−18.3
34	25	−60.0
89	79	16.2
96	95	33.0
78	78	8.5

この射影値を**第一主成分得点**と呼ぶ

この例では『学力』に対応すると考えられるかも

主成分分析 – 小休止



最もばらつきの大きい方向（第一主成分）を発見しその方向にデータを射影して第一主成分得点を取得した…

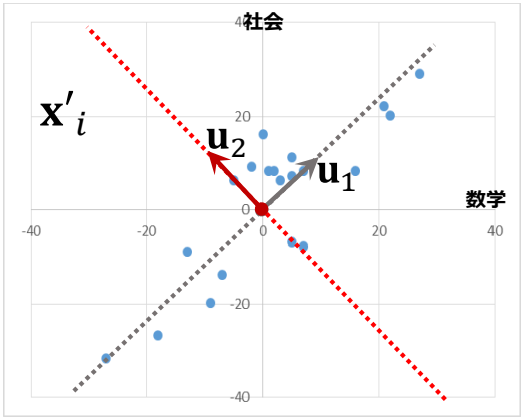
残ってる主な疑問

→ uと直交する方向にもデータはばらついているけど無視していいの？

→ 射影によってデータが失われたのでは？

→ ばらつき方向uはどうやって計算するの？

主成分分析 - 第n主成分



データ点のばらつきが最も大きい方向を**第1主成分**，その方向への射影を**第1主成分得点**と呼ぶ

第1主成分と直交し，かつ，データ点のばらつきが最も大きい方向を**第2主成分**とよび，その方向への射影を**第2主成分得点**と呼ぶ

同様に**第n主成分・第n主成分得点**が定義される

※主成分は，主成分ベクトルや負荷量ベクトルなどとも呼ばれる

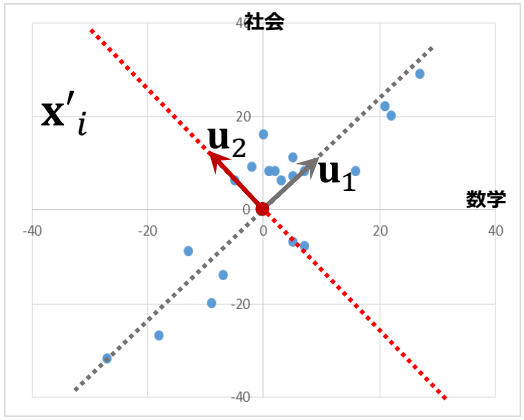
例）左図では・・・

第1主成分得点($\mathbf{u}_1$への射影)は『学力』を表現

第2主成分得点($\mathbf{u}_2$への射影)は『文系指向』を表現

しているように考えられるかも知れない（意味づけは解析者が実施）

主成分分析 - 第n主成分

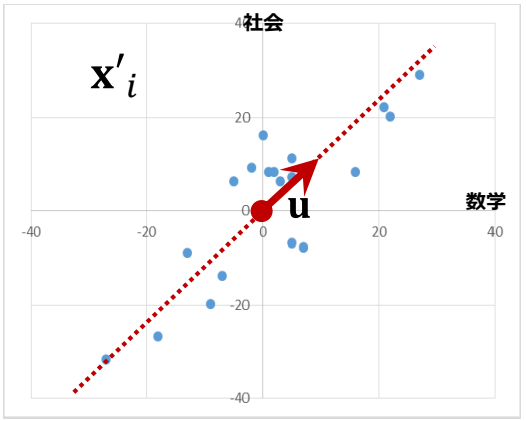


数学	社会	第1主成分得点	第2主成分得点
80	63	−1.8	−10.4
95	91	29.3	−4.3
100	100	39.3	−2.5
66	53	−18.3	−6.0
34	25	−60.0	1.1
89	79	16.2	−7.3
96	95	33.0	−2.6
78	78	8.5	0.6
55	44	−32.1	−3.2
60	58	−18.2	1.8
46	39	−41.7	0.6
80	79	10.6	−0.4
73	87	12.3	10.1
68	77	1.5	7.6
75	79	7.4	3.5
76	77	6.5	1.5
66	57	−15.2	−3.4
78	81	10.9	2.5
74	78	6.0	3.6
71	80	5.7	7.2

$\mathbf{u}_1 = (0.63, 0.77)$
$\mathbf{u}_2 = (-0.77, 0.63)$

平均：(73, 71)

主成分分析 – 小休止



最もばらつきの大きい方向（主成分）を発見しその方向にデータを射影して主成分得点を取得した…

残ってる主な疑問

→ uと直交する方向にもデータはばらついているけど無視していいの？ → **場合による（n次元データには第n主成分まで存在する）**

→ 射影によってデータが失われたのでは？

→ ばらつき方向uはどうやって計算するの？

主成分分析 –
第1主成分の計算

入力点群：$\hat{\mathbf{x}}_i \in R^d, i = 1,2,\dots,N$

平均値：$\mathbf{m} = \frac{1}{N}\sum_i \hat{\mathbf{x}}_i$

平行移動：$\mathbf{x}_i = \hat{\mathbf{x}}_i - \mathbf{m}$

以下の最大値問題を求めたい

$$\underset{||\mathbf{u}||=1}{\operatorname{argmax}} \frac{1}{N}\sum_i (\mathbf{u}^T\mathbf{x}_i)^2$$

主成分分析 –
第1主成分の計算

入力点群：$\hat{\mathbf{x}}_i \in R^d, i = 1,2,\dots,N$

平均値：$\mathbf{m} = \frac{1}{N}\sum_i \hat{\mathbf{x}}_i$

平行移動：$\mathbf{x}_i = \hat{\mathbf{x}}_i - \mathbf{m}$

以下の最大値問題を求めたい

$$\underset{||\mathbf{u}||=1}{\operatorname{argmax}} \frac{1}{N}\sum_i (\mathbf{u}^T\mathbf{x}_i)^2$$

準備：

分散共分散行列 $\mathbf{A} = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i{}^T \in R^{d\times d}$ を考える

- 行列$\mathbf{A}$は対称行列(異なる固有値に対する固有ベクトルが直行)
- 行列$\mathbf{A}$は半正定置性を持つ（固有値 ≥ 0）
- $\mathbf{A}$の固有値を$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \geq 0$とし，長さ1で互いに直交する固有ベクトルを$\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d$とすと

$$\boldsymbol{V}^T\mathbf{A}\mathbf{V} = \operatorname{diag}(\lambda_1, \lambda_2, \dots, \lambda_d)$$

$$\mathbf{V} = (\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_d)$$

と対角化できる．

主成分分析 – 第1主成分の計算

入力点群：$\hat{\mathbf{x}}_i \in R^d, i = 1,2,\dots,N$

平均値：$\mathbf{m} = \frac{1}{N}\sum_i \hat{\mathbf{x}}_i$

平行移動：$\mathbf{x}_i = \hat{\mathbf{x}}_i - \mathbf{m}$

以下の最大値問題を求めたい

$$\underset{||\mathbf{u}||=1}{\operatorname{argmax}} \frac{1}{N}\sum_i (\mathbf{u}^T\mathbf{x}_i)^2$$

$$\begin{aligned}
\frac{1}{N}\sum_i (\mathbf{u}^T\mathbf{x}_i)^2 &= \frac{1}{N}\sum_i \mathbf{u}^T\mathbf{x}_i\mathbf{x}_i{}^T\mathbf{u} && \text{※コスト関数を変形する}\\
&= \mathbf{u}^T\left(\frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i{}^T\right)\mathbf{u}\\
&= \mathbf{u}^T\boldsymbol{A}\mathbf{u} && \text{※}\mathbf{A} = \sum_i \mathbf{x}_i\mathbf{x}_i{}^T \text{ 分散共分散行列}\\
&= (\mathbf{V}\mathbf{V}^{\mathrm{T}}\mathbf{u})^T\boldsymbol{A}(\mathbf{V}\mathbf{V}^{\mathrm{T}}\mathbf{u})\\
&= (\mathbf{V}^T\mathbf{u})^T\boldsymbol{V}^T\mathbf{A}\mathbf{V}(\mathbf{V}^T\mathbf{u})\\
&= (\mathbf{V}^T\mathbf{u})^T\mathrm{diag}(\lambda_1,\lambda_2,\dots,\lambda_{\mathrm{d}})(\mathbf{V}^T\mathbf{u})\\
&\le (\mathbf{V}^T\mathbf{u})^T\mathrm{diag}(\lambda_1,\lambda_1,\dots,\lambda_1)(\mathbf{V}^T\mathbf{u})\\
&= \lambda_1(\mathbf{V}^T\mathbf{u})^T(\mathbf{V}^T\mathbf{u})\\
&= \lambda_1
\end{aligned}$$

等号成立は$\mathbf{V}^T\mathbf{u}$=(1,0,0,⋯,0)のとき，つまり$\mathbf{u}$= $\mathbf{v}_1$のとき最大値となる．

→ 分散共分散行列の1番目の固有ベクトルが第一主成分！

主成分分析 – 第2主成分の計算

入力点群：$\hat{\mathbf{x}}_i \in R^d, i = 1,2,\dots,N$

平均値：$\mathbf{m} = \frac{1}{N}\sum_i \hat{\mathbf{x}}_i$

平行移動：$\mathbf{x}_i = \hat{\mathbf{x}}_i - \mathbf{m}$

以下の最大値問題を求めたい

$$\underset{||\mathbf{u}||=1,\ \mathbf{u}^T\mathbf{v}_1=0}{\operatorname{argmax}} \sum_i (\mathbf{u}^T\mathbf{x}_i)^2$$

長さ1で、$\mathbf{v}_1$と直行するベクトル$\mathbf{u}$を探索する

先と同様に変形する

$$\begin{aligned}
\textstyle\sum_i(\mathbf{u}^T\mathbf{x}_i)^2 &= \mathbf{u}^T(\textstyle\sum_i \mathbf{x}_i\mathbf{x}_i{}^T)\mathbf{u}\\
&= (\mathbf{V}\mathbf{V}^{\mathrm{T}}\mathbf{u})^T\boldsymbol{A}(\mathbf{V}\mathbf{V}^{\mathrm{T}}\mathbf{u})\\
&= (\mathbf{V}^T\mathbf{u})^T\mathrm{diag}(\lambda_1,\lambda_2,\dots,\lambda_{\mathrm{d}})(\mathbf{V}^T\mathbf{u})
\end{aligned}$$

ここで条件$\mathbf{u}^T\mathbf{v}_1$=0 より$\mathbf{V}^T\mathbf{u} = (0,u_2,u_3,\dots)^T$であることに注意して…

$$\begin{aligned}
&= (\mathbf{V}^T\mathbf{u})^T\mathrm{diag}(0,\lambda_2,\dots,\lambda_{\mathrm{d}})(\mathbf{V}^T\mathbf{u})\\
&\le (\mathbf{V}^T\mathbf{u})^T\mathrm{diag}(0,\lambda_2,\dots,\lambda_2)(\mathbf{V}^T\mathbf{u})\\
&= \lambda_2
\end{aligned}$$

等号成立は$\mathbf{V}^T\mathbf{u}$=(0,1,0,⋯,0)のとき，つまり$\mathbf{u}$= $\mathbf{v}_2$のとき最大値となる．

→ 分散共分散行列の2番目の固有ベクトルが第2主成分！

主成分分析 – 第n主成分の計算

入力点群：$\hat{\mathbf{x}}_i \in R^d, i = 1,2,\dots,N$

平均値：$\mathbf{m} = \frac{1}{N}\sum_i \hat{\mathbf{x}}_i$

平行移動：$\mathbf{x}_i = \hat{\mathbf{x}}_i - \mathbf{m}$

以下の最大値問題を求めたい

$$\underset{\mathbf{u}=1}{\operatorname{argmax}} \sum_i (\mathbf{u}^T\mathbf{x}_i)^2$$

ただし$\mathbf{u}^T\mathbf{v}_1$=$\mathbf{u}^T\mathbf{v}_2 = \cdots = \mathbf{u}^T\mathbf{v}_{\mathrm{n}-1}$=0

先と同様に計算すると…

$\mathbf{u}$= $\mathbf{v}_{\mathrm{n}}$のときに最大値を取ることが分かる．

つまり…

第n主成分は，分散共分散行列

$$\mathbf{A} = \frac{1}{N}\sum_i \mathbf{x}_i\mathbf{x}_i{}^T$$

のn番目に大きな固有値に対する固有ベクトルと等しくなる．

練習問題

2次元データ（1，3），（3，1），（5，7），（7，5）に付いて、以下の手順でPCAを計算せよ

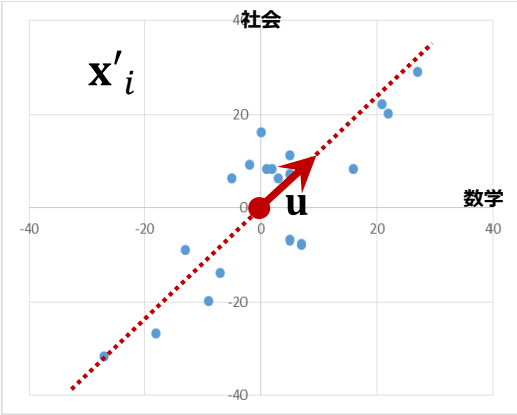
1．平均ベクトルを計算せよ

2．平均ベクトルを用いて平均が原点になるようデータを平行移動せよ（センタリング）

3．データの分散共分散行列 Z を計算せよ

4．分散共分散行列Zの固有値固有ベクトルを求め、第一主成分を計算せよ

主成分分析 – 小休止



最もばらつきの大きい方向（主成分）
を発見しその方向にデータを射影して
主成分得点を取得した…

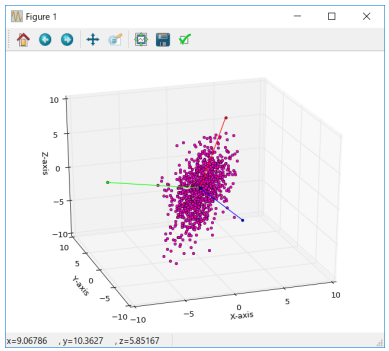
残ってる主な疑問

- uと直交する方向にもデータはばらついているけど無視していいの？ →場合による（n次元データには第n主成分まで存在する）
- 射影によってデータが失われたのでは？
- ばらつき方向uはどうやって計算するの？ → 分散共分散行列の固有ベクトルを求めればok

PCA_PLOT_3D.py

主成分分析 - 次元圧縮への応用

例）
3次元データ点群が下図の通り分布している
分布にはあまり偏りがないため，すべての主成分得点の数値が比較的大きな値に



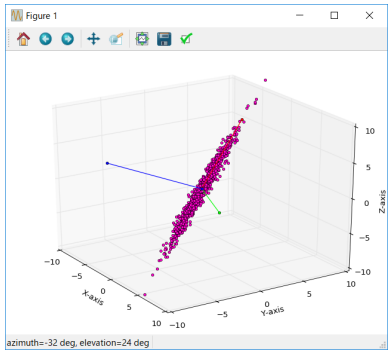
	points		
x	y	z	
0.86	-2.00	4.57	
0.86	0.27	2.78	
-1.19	0.73	-4.73	
3.22	1.17	4.63	
0.33	-1.07	-3.13	
0.03	0.49	3.68	
2.36	0.51	-1.73	
-2.16	-0.07	-0.87	
0.42	1.27	0.90	
0.15	-1.02	-1.12	
0.95	-0.20	0.01	
2.26	-0.23	0.81	
0.86	0.23	1.87	
-2.28	-0.47	-3.74	
0.67	-0.14	0.08	
0.42	0.58	-0.15	

	pca		
	1	2	3
	-4.74	-0.42	-1.81
	-2.94	0.12	0.40
	4.85	0.03	0.67
	-5.31	1.98	1.28
	2.88	1.02	-1.12
	-3.60	-0.90	0.68
	1.05	2.70	0.43
	1.33	-1.91	0.04
	-0.99	0.20	1.35
	0.98	0.35	-1.00
	-0.30	0.88	-0.17
	-1.40	1.94	-0.21
	-2.06	0.35	0.33
	4.13	-1.32	-0.45
	-0.29	0.59	-0.10
	0.01	0.44	0.63

PCA_PLOT_3D.py

主成分分析 - 次元圧縮への応用

例）3次元データ点群が下図の平面上に通り分布している
データ点は平面に乗っているため，第1主成分の寄与が大きく
第3主成分は寄与しない偏った分布



	points		
x	y	z	
1.30	-2.07	-2.85	
0.61	0.36	1.33	
-0.65	-0.33	-1.31	
-1.61	-0.71	-3.04	
-0.32	-2.74	-5.81	
1.04	2.45	5.94	
-0.49	-1.58	-3.66	
-1.85	-0.36	-2.57	
-0.74	-0.73	-2.20	
0.02	2.57	5.16	
0.27	1.55	3.36	
-0.57	-2.86	-6.29	
-0.59	-0.42	-1.42	
-1.15	0.27	-0.61	
-1.62	2.08	2.53	
-0.01	1.02	2.02	
0.73	-2.72	-4.70	

	pca		
	1	2	3
	-3.32	1.58	0.00
	1.45	0.52	0.00
	-1.30	-0.69	0.00
	-3.08	-1.63	0.00
	-6.37	-0.01	0.00
	6.52	0.67	0.00
	-3.95	-0.34	0.00
	-2.53	-1.92	0.00
	-2.29	-0.73	0.00
	5.81	-0.40	0.00
	3.77	0.00	0.00
	-6.87	-0.24	0.00
	-1.44	-0.61	0.00
	-0.44	-1.29	0.00
	2.31	-0.22	0.00
	-5.30	1.09	0.00

主成分分析 - 次元圧縮への応用

n次元データの次元を圧縮することを考える

- ***k***次元まで圧縮する
- 情報量の欠落を抑えられるいい感じの『***k***』を選択したい
  （平面に縮退しているような軸は削除しつつも，分散の大きな軸は利用したい）

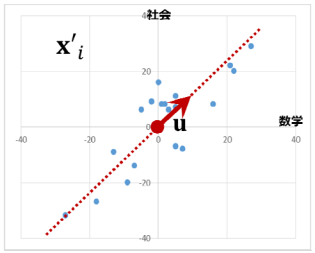
→ **寄与率**を利用する

$$\text{寄与率} = \frac{k\text{番目の方向までの分散}}{\text{全方向の分散}} = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{N} \lambda_i}$$

※第k主成分方向の分散はλ_kとなる
例)寄与率が 0.8 以上になる最小のkを選択する

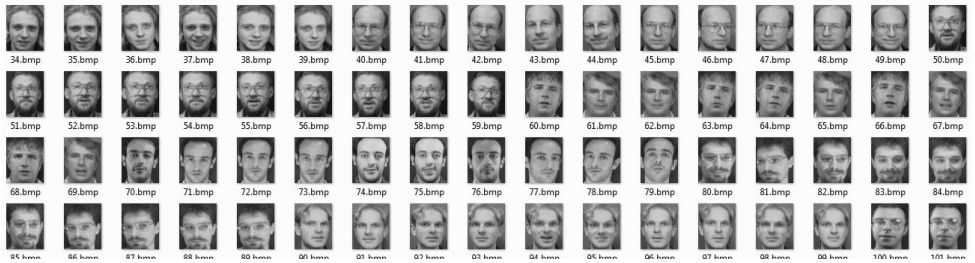
主成分分析 – まとめ

- 高次元空間に分布するデータよりばらつきの大きな軸を探索する手法
  - ばらつきの最も大きな軸を第1主成分、その軸への射影を第1主成分得点と呼ぶ
  - ばらつきのk番目に大きな軸を第k主成分、その軸への射影を第k主成分得点と呼ぶ
- 計算方法
  - 平均が原点に来るようにデータ点群全体を平行移動
  - 分散共分散行列を計算
  - 分散共分散行列の固有ベクトルが主成分を示す
- 特徴ベクトルの次元圧縮
  - 特徴ベクトル群から寄与率の高い主成分のみ抽出し，低次元化してから計算（識別など）を行なう．
  - 情報量をあまり落とさずに，計算量・メモリ量などの削減が可能



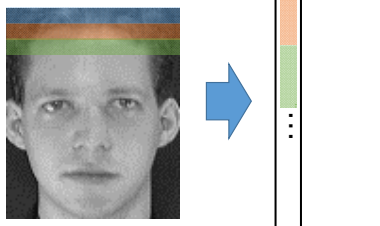
PCAによる画像の次元圧縮

- 例として顔データのPCA圧縮をしてみる
- AT&Tデータセットを利用 https://git-disl.github.io/GTDLBench/datasets/att_face_dataset/
- 40人 * 10枚 = 400枚の写真群 （PCAするには少し小さいが。。。）
- 解像度: 92 x 112



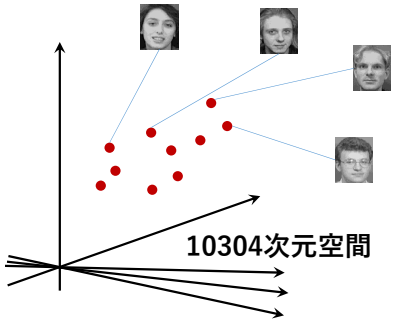
PCAによる画像の次元圧縮

- 92 x 112 pixelの写真を，10304次元ベクトルに変換



92 x 112

10304次元



※ 1枚の画像は 10304次元空間の点になる
※『人の顔』のように特定グループの写真群は，高次元空間の部分空間に含まれる（超平面に乗る）ことが多い

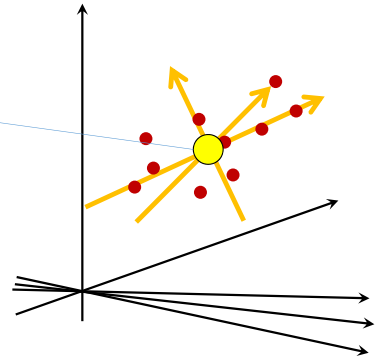
PCAによる画像の次元圧縮

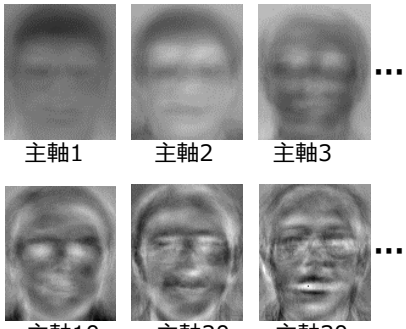
- 分散共分散行列は10304 x 10304に
- 400個の固有値・固有ベクトルが取得できる

※$\sum_i (\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^T$のrankは最大で$N$=400なので次元数分の軸は得られない

- 主成分ベクトルは10304次元で、画像(サイズ92x112)として可視化できる

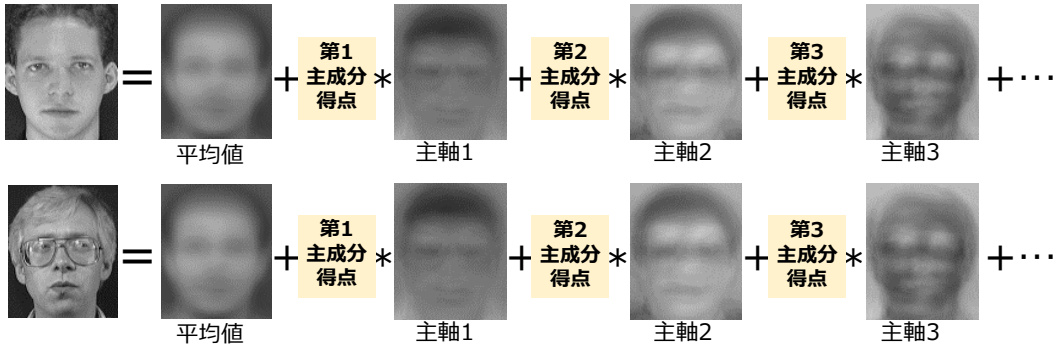
平均値





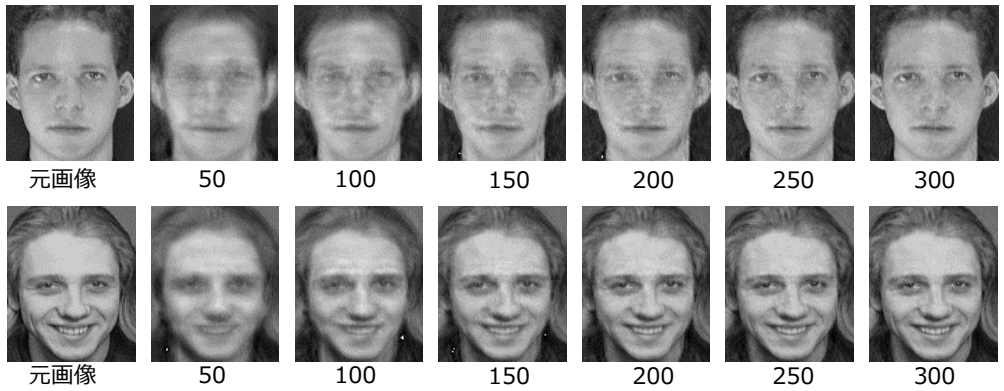
PCAによる画像の次元圧縮

- 元画像は，平均値 + Σ 主成分x主成分得点 の形で表現できる
- 後半の主成分は寄与が少ない(はず)ので，切り捨てても影響が少ない（のでは？）



PCAによる画像の次元圧縮

- 実際に50個，100個，…，300個の主成分を利用して再構築してみた



顔の向きもそろっているデータを利用するともっと速く収束すると思う

練習問題

主成分分析（PCA）に関する説明として正しいものをすべて選びなさい。

A. PCAはデータのばらつきが最も大きくなる方向を見つける手法である
B. PCAは、各データのクラスIDを利用して主成分軸を計算する
C. PCAにより求まる各主成分軸は互いに直交する
D. 第I主成分は、分散共分散行列の最小の固有値に対応する固有ベクトルである

次のうち、主成分分析（PCA）が実際に役立つ場面として適切なものをすべて選びなさい。

A. 高次元データ（センサデータなど）の次元を圧縮し、処理を高速化したいとき
B. 高次元データを、2次元に落として視覚的に理解したいとき。
C. 高次元データにおいてどの変数がデータのばらつきに寄与しているかを調べたいとき
D. センサデータ等の工学的データに適するが、人口・年齢・収入のような社会統計データには利用すべきでない
E. 人口・年齢・収入のような社会統計データに適するが、センサデータ等の工学的データには利用すべきでない
F. PCAは、高次元データがクラスID（ラベル）を持ってさえいればどんな場合にも適用可能である

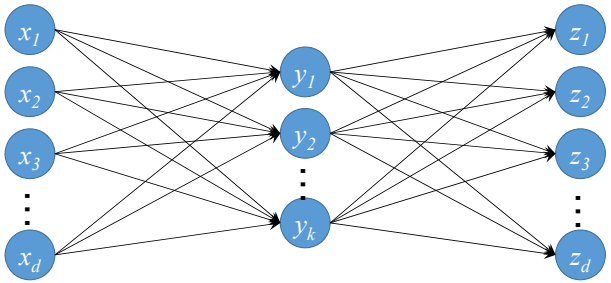
オートエンコーダ
自己符号化器

参考資料

- **深層学習**
- **(機械学習プロフェッショナルシリーズ)** 単行本
- 岡谷 貴之

オートエンコーダー（自己符号化器）とは

- ニューラルネットの一種で、データをよく表す特徴の獲得を目指す
- 目的出力を伴わない訓練データを利用した**教師なし学習**
- 下図のようなニューラルネットワークを考える



入力 $\mathbf{x} \in R^d$
d次元ベクトルを入力

中間層
k次元ベクトルを出力

出力層 $\mathbf{z}$
d次元ベクトルを出力

オートエンコーダの概要

N個のデータ $\mathbf{x_i} \in R^d$

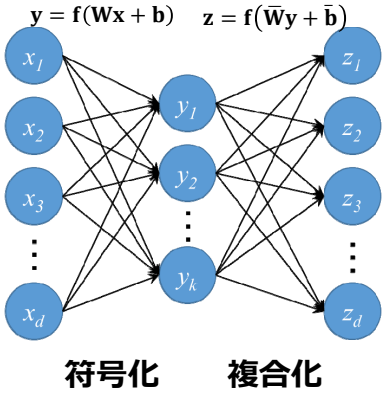
各データ$\mathbf{x_i}$に対する出力$\mathbf{z_i}$が、なるべく$\mathbf{x_i}$に等しくなるよう重みを学習する

※入力データと似たデータを出力できるように学習する

※中間層の次元がdより小さい場合，$\mathbf{x_i} = \mathbf{z_i}$を必ず満たすことは不可能

- 全データに対して，入力と近い出力が得られるような学習が行えたら…

→ 元データ$\mathbf{x_i}$の情報をあまり落とさずに次元削減ができたことになる

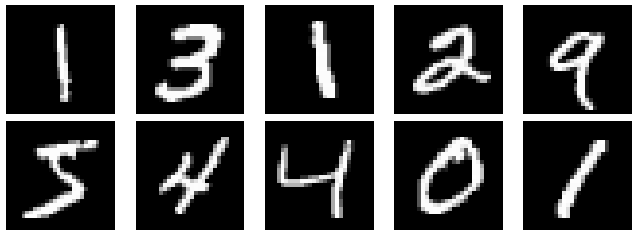


自己符号化器の例

MNIST : URL: http://yann.lecun.com/exdb/mnist/

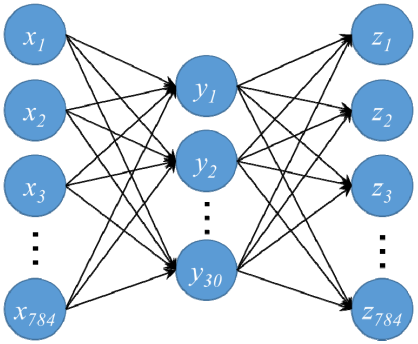
- パターン認識の勉強によく利用される**手書き数字画像**データセット
- 数字は画像の中心に配置され，数字のサイズは正規化されている
- 各画像のサイズは 28x28
- データ数：トレーニング用：60000文字 / テスト用：10000文字

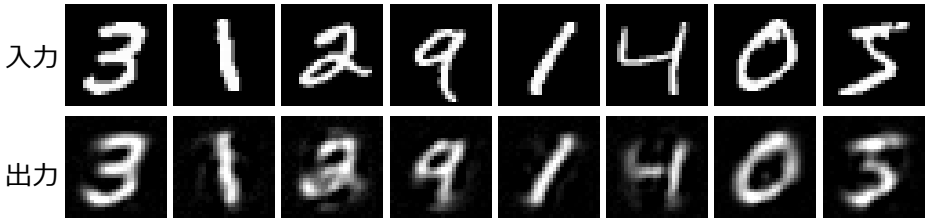
例)



自己符号化器の例

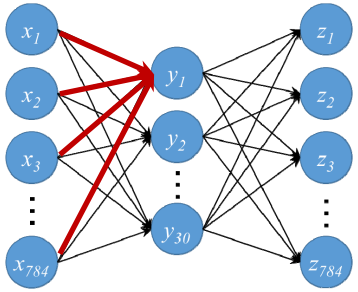
- **MNIST** を自己符号化器で符号化
  - データの次元 : 784 = 28x28
  - 中間層の次元 : 30
  - 訓練データ数 : 60000
  - 活性化関数 :恒等関数
  - epochs=50, batch_size=20



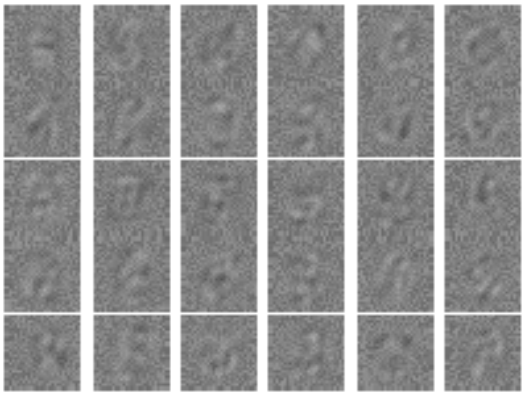


自己符号化器の例

- 自己符号化器の学習後の興味は『戻せたかどうか？』では無くて
  学習された重み係数（特徴量）



赤矢印部分の重みは784次元
これを画像に直したのが右図
※これが784次元の点群を30次元に圧縮するための軸を表す



まとめ

- オートエンコーダ（自己符号化器）とは…
  - 入力データになるべく似たデータを出力するニューラルネット
  - 目的出力を伴わない入力だけの訓練データを利用した教師なし学習
  - データをよく表す特徴の獲得を目指す
  - バイアス項 b=0，活性化関数を恒等写像とした場合主成分分析と実質的に同じ

- 応用例
  - 次元圧縮
  - 深層学習の前処理に利用