

# 線形回帰（2）

機械学習演習 プランナークラス

[masada@rikkyo.ac.jp](mailto:masada@rikkyo.ac.jp)

正則化 regularization

# Ridge回帰とLasso

[ESLII] Jerome H. Friedman, Robert Tibshirani, and Trevor Hastie.  
The Elements of Statistical Learning: Data Mining, Inference, and  
Prediction. Second Edition. Chapter 3.

# 変数を選択することの問題点

- 説明変数が多いとき、例えばESLII, Sec. 3.2.1のExample: Prostate Cancerのように検定の結果を使ってnon-significantな変数を削ったりする
  - 同書3.3節には、もっと良い変数選択の方法が書かれてある。
- しかし、変数を選択するというのは、離散的な手続き
  - 予測対象となるデータ集合によって、性能に段差がつくことがある
- そこで、shrinkage methodsと呼ばれる連続的な手続きを採る

$$(Z \text{ score}) = (\text{Coefficient}) / (\text{Std. Error})$$

による変数選択

**TABLE 3.2.** *Linear model fit to the prostate cancer data. The Z score is the coefficient divided by its standard error (3.12). Roughly a Z score larger than two in absolute value is significantly nonzero at the  $p = 0.05$  level.*

Term	Coefficient	Std. Error	Z Score
Intercept	2.46	0.09	27.60
lcavol	0.68	0.13	5.37
lweight	0.26	0.10	2.75
age	−0.14	0.10	−1.40
lbph	0.21	0.10	2.06
svi	0.31	0.12	2.47
lcp	−0.29	0.15	−1.87
gleason	−0.02	0.15	−0.15
pgg45	0.27	0.15	1.74

# 変数選択を連続的にする

- ある説明変数を使わない = その説明変数の係数をゼロにする

ON/OFFではなく、連続的にすると・・・



- ある説明変数を使わない = その説明変数の係数がゼロに近くなるようにする

# Ridge回帰

- 通常の最小二乗法とは、最小化すべき関数が少し違う

$$l(\mathbf{a}) = \sum_{i=1}^N \left( y_i - a_0 - \sum_{j=1}^d a_j x_{i,j} \right)^2 + \lambda \sum_{j=1}^d a_j^2$$

- 説明変数の係数（切片は含まない）の2乗和も同時に最小化
  - 係数が全体的に0のほうに近寄った値になる。
  - $\lambda$ でその強さをコントロールする。
  - $\lambda$ は交差検証などで決定する。

# Lasso

- 通常の最小二乗法とは、最小化すべき関数が少し違う

$$l(\mathbf{a}) = \frac{1}{2} \sum_{i=1}^N \left( y_i - a_0 - \sum_{j=1}^d a_j x_{i,j} \right)^2 + \lambda \sum_{j=1}^d |a_j|$$

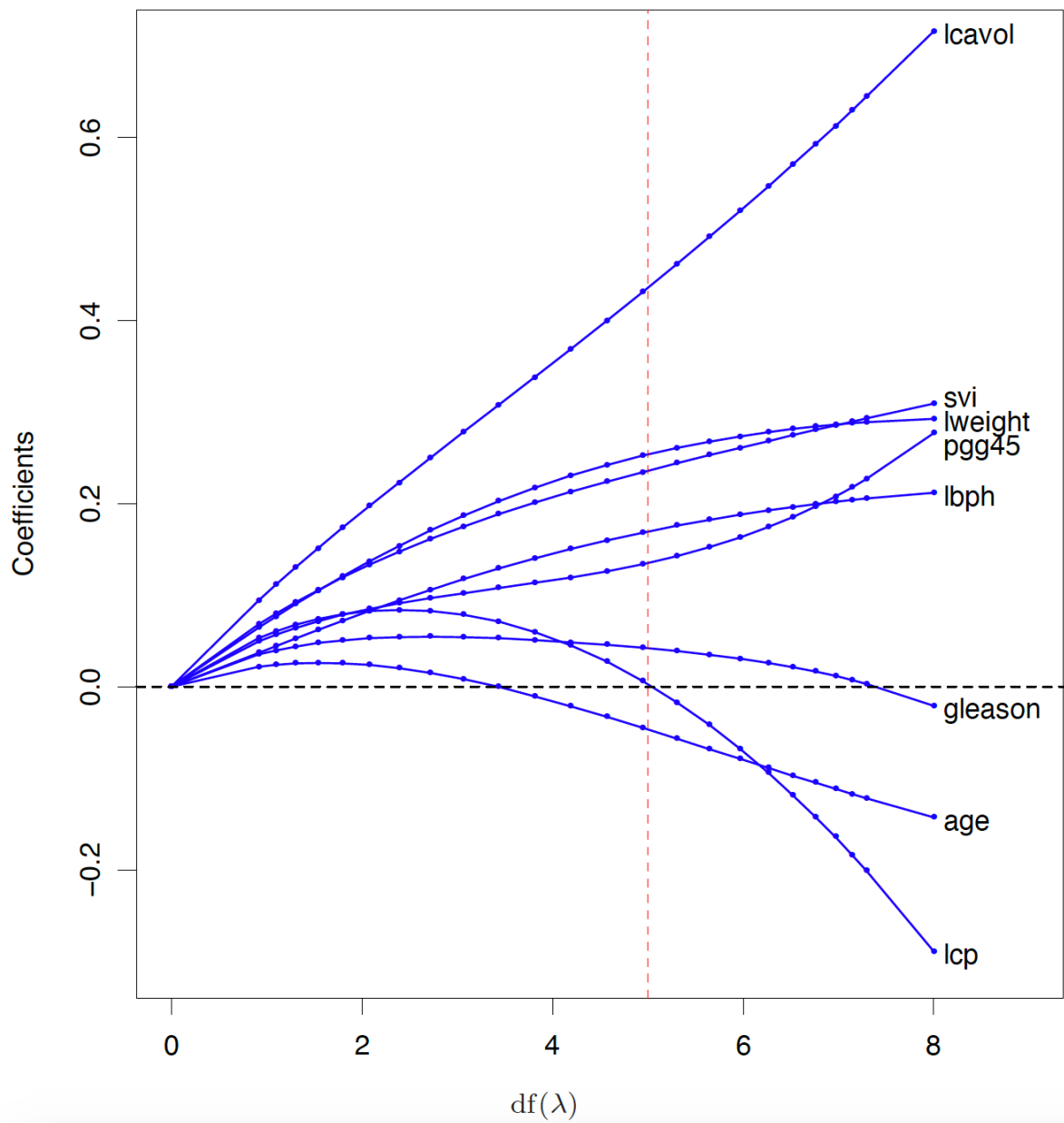
- 説明変数の係数（切片は含まない）の絶対値和も同時に最小化
  - 係数が全体的に0のほうに近寄った値になる。
  - $\lambda$  でその強さをコントロールする。
  - $\lambda$  は交差検証などで決定する。



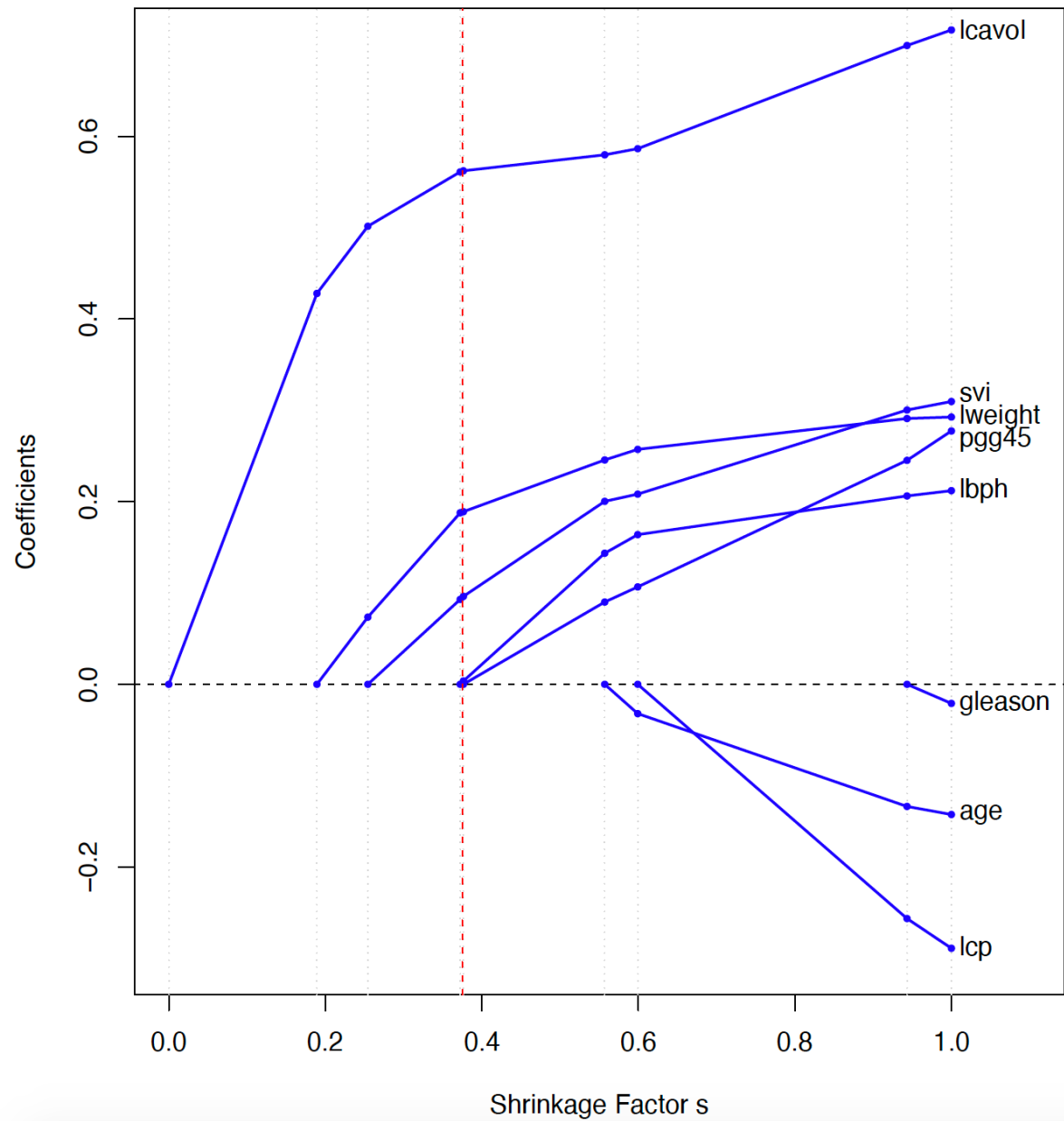
# Ridge回帰とLassoの違い

- $\lambda$  を大きくして係数をゼロに近づける項の効きを強くすると...

- Ridge回帰では、すべての係数が全体的にゼロに近寄る
- Lassoでは、係数がひとつずつ、ほぼゼロの値になっていく



Ridge回帰 [ESLII, p.65]



Lasso [ESLII, p.70]

# なぜ切片が正則化に含まれないのか(1/2)

- 例えば $y_i$ に一斉に1を足して、推定をやり直したとすると…
  - 通常の最小二乗法：切片の推定値だけが変化する
  - Ridge回帰やLasso：切片を正則化に含めると答え全体が変わる
- つまり、推定計算が $y_i$ の原点をどこに採るかに依存してしまう
- よって、切片は、通常、正則化には含ませない

## なぜ切片が正則化に含まれないのか(2/2)

- しかし、切片を含まない正則化を使った推定は、中心化されたデータを使うことで初めから切片を無視した正則化を使った推定と、全く同じ答えを与える
- また、前者の方法で得られる切片の推定値については、後者の方法で得られる他の係数の推定値を使って表現できる（下式）

$$\hat{a}_0 = \frac{1}{N} \sum_{i=1}^N y_i - \sum_{j=1}^d \hat{a}_j \left( \frac{1}{N} \sum_{i=1}^N x_{i,j} \right)$$

It has to be emphasized that in practice, the bias parameter  $\theta_0$  is left out from the norm in the regularization term; penalization of the bias would make the procedure dependent on the origin chosen for  $y$ . Indeed, it is easily checked that adding a constant term to each one of the output values,  $y_n$ , in the cost function would not result in just a shift of the predictions by the same constant, if the bias term is included in the norm. Hence, usually, ridge regression is formulated as

$$\text{minimize } L(\boldsymbol{\theta}, \lambda) = \sum_{n=1}^N \left( y_n - \theta_0 - \sum_{i=1}^l \theta_i x_{ni} \right)^2 + \lambda \sum_{i=1}^l |\theta_i|^2. \quad (3.43)$$

It turns out (Problem 3.11) that minimizing Eq. (3.43) with respect to  $\theta_i$ ,  $i = 0, 1, \dots, l$ , is equivalent to minimizing Eq. (3.41) using *centered* data and neglecting the intercept. That is, one solves the task

$$\text{minimize } L(\boldsymbol{\theta}, \lambda) = \sum_{n=1}^N \left( (y_n - \bar{y}) - \sum_{i=1}^l \theta_i (x_{ni} - \bar{x}_i) \right)^2 + \lambda \sum_{i=1}^l |\theta_i|^2, \quad (3.44)$$

and the estimate of  $\theta_0$  in Eq. (3.43) is given in terms of the obtained estimates  $\hat{\theta}_i$ , i.e.,

$$\hat{\theta}_0 = \bar{y} - \sum_{i=1}^l \hat{\theta}_i \bar{x}_i,$$

欠測データ

# 欠測データの 統計解析

阿部貴行 [著]

統計解析  
スタンダード

国友直人  
竹村彰通  
岩崎 学  
[編集]

朝倉書店

調査観察データ  
解析の実際

1

# 欠測データの 統計科学

医学と社会科学への応用

高井啓二  
星野崇宏  
野間久史

岩波書店

高橋将宜・渡辺美智子 著

# 欠測データ処理

Rによる単一代入法と多重代入法

共立出版

統計学 5  
One Point

# 参考資料

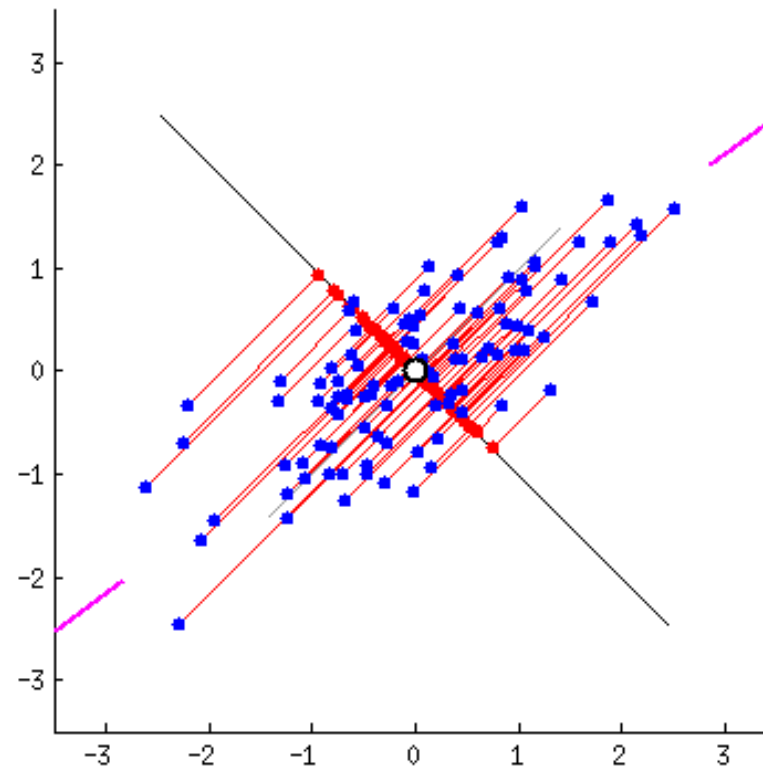
- 「諸外国の公的統計における欠測値の対処法」
  - <http://toukeigaku.sakura.ne.jp/jp/Toukeigaku/journal/112toukeigaku/takahashi112.pdf>
- 「欠測値の補完に係る主な方法」
  - [https://www.soumu.go.jp/main\\_content/000741245.pdf](https://www.soumu.go.jp/main_content/000741245.pdf)
    - [https://www.soumu.go.jp/main\\_sosiki/singi/toukei/hyokabunkakai/kaigi/02shingi05\\_02000472.html](https://www.soumu.go.jp/main_sosiki/singi/toukei/hyokabunkakai/kaigi/02shingi05_02000472.html)



# 主成分分析

dimensionality reductionの一手法

# PCAのイメージ



# PCAによる次元削減のイメージ

1. データをあらかじめ中心化しておく（平均を引いておく）
  - スケーリングもしておく（標準偏差で割っておく）
2. 原点を通る直線のうちデータに一番「近い」ものを見つける
3. その直線に垂直な平面へ、データを射影する
4. 2.に戻って、次元がひとつ落ちた空間で同じことを繰り返す

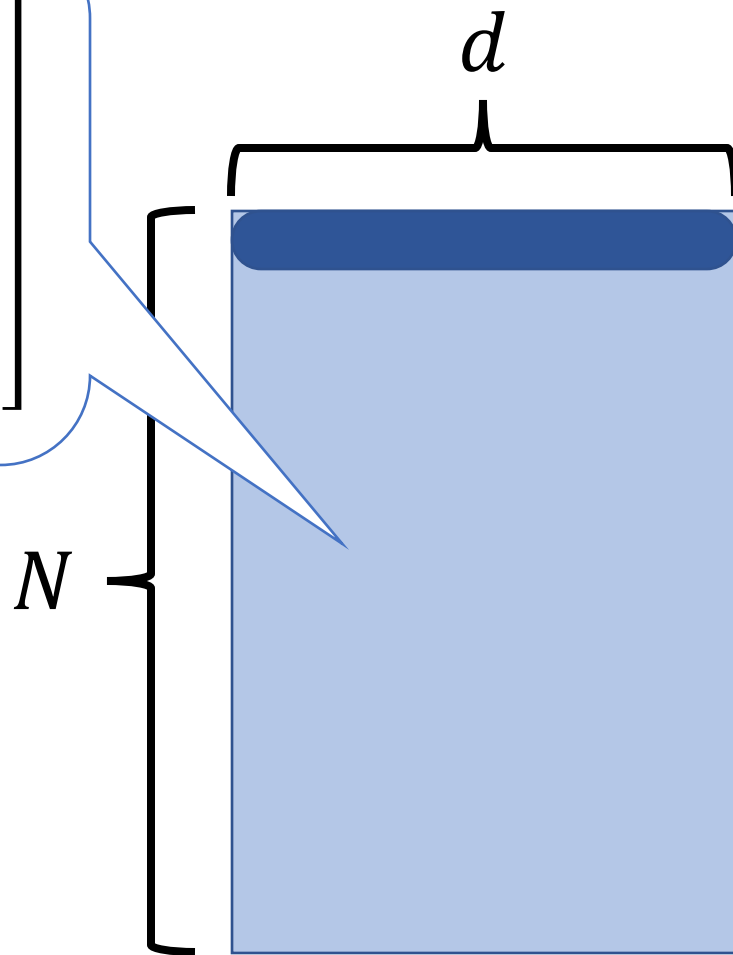
# PCAの雰囲気

- データがどの方向に大きく散らばっているかを知ることは有益
- そこで . . .
- データがより大きく散らばっている向きを順番に見つけていく
  - 後から見つけた向きは、先に見つけた向きに直交しているようにする

# PCAで計画行列をless noisyにする

$$\begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,d} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{i,1} & x_{i,2} & \cdots & x_{i,d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N,1} & x_{N,2} & \cdots & x_{N,d} \end{bmatrix}$$

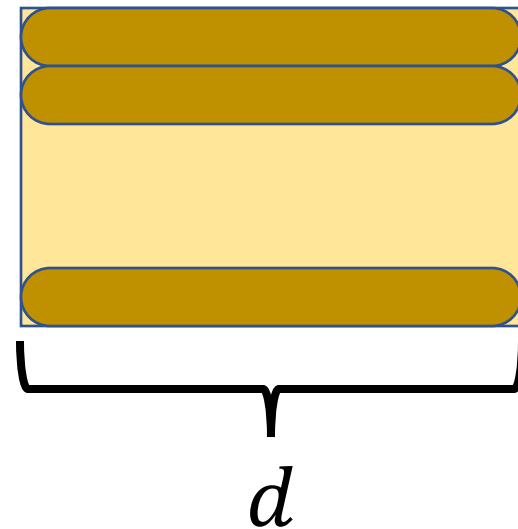
左辺は  
元々の  
計画行列



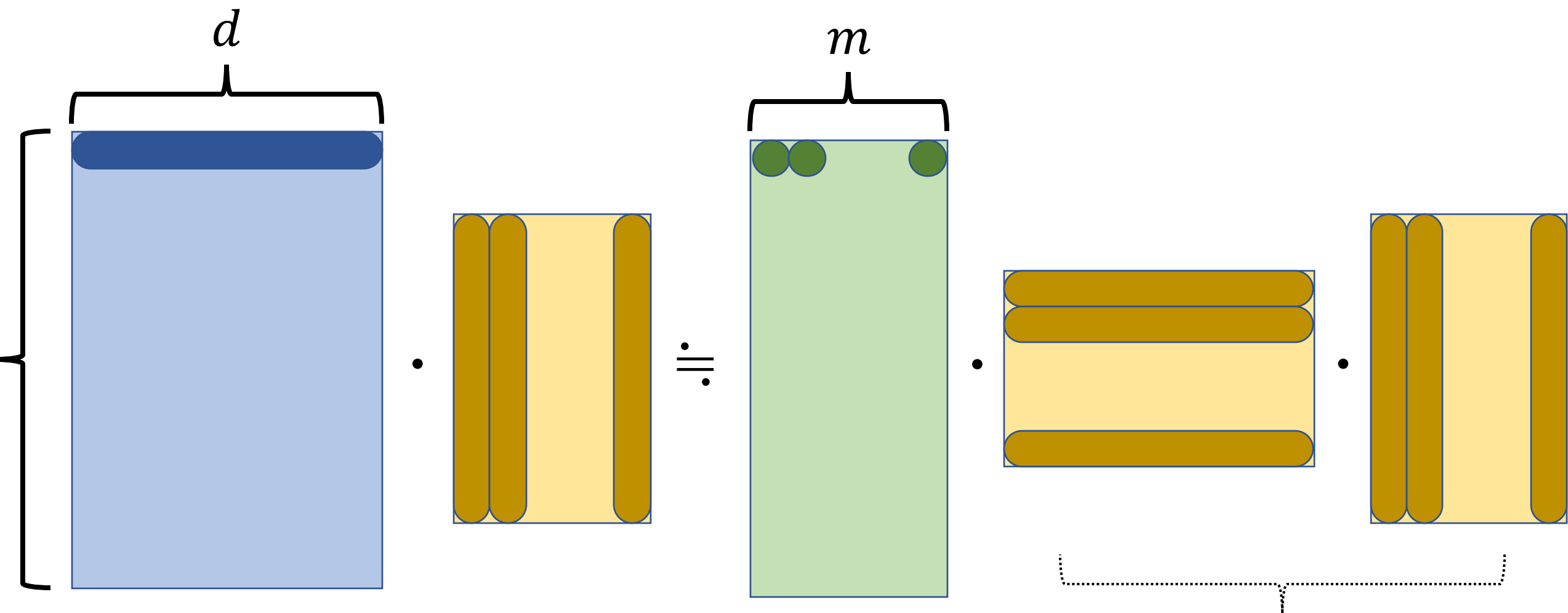
$\hat{=}$



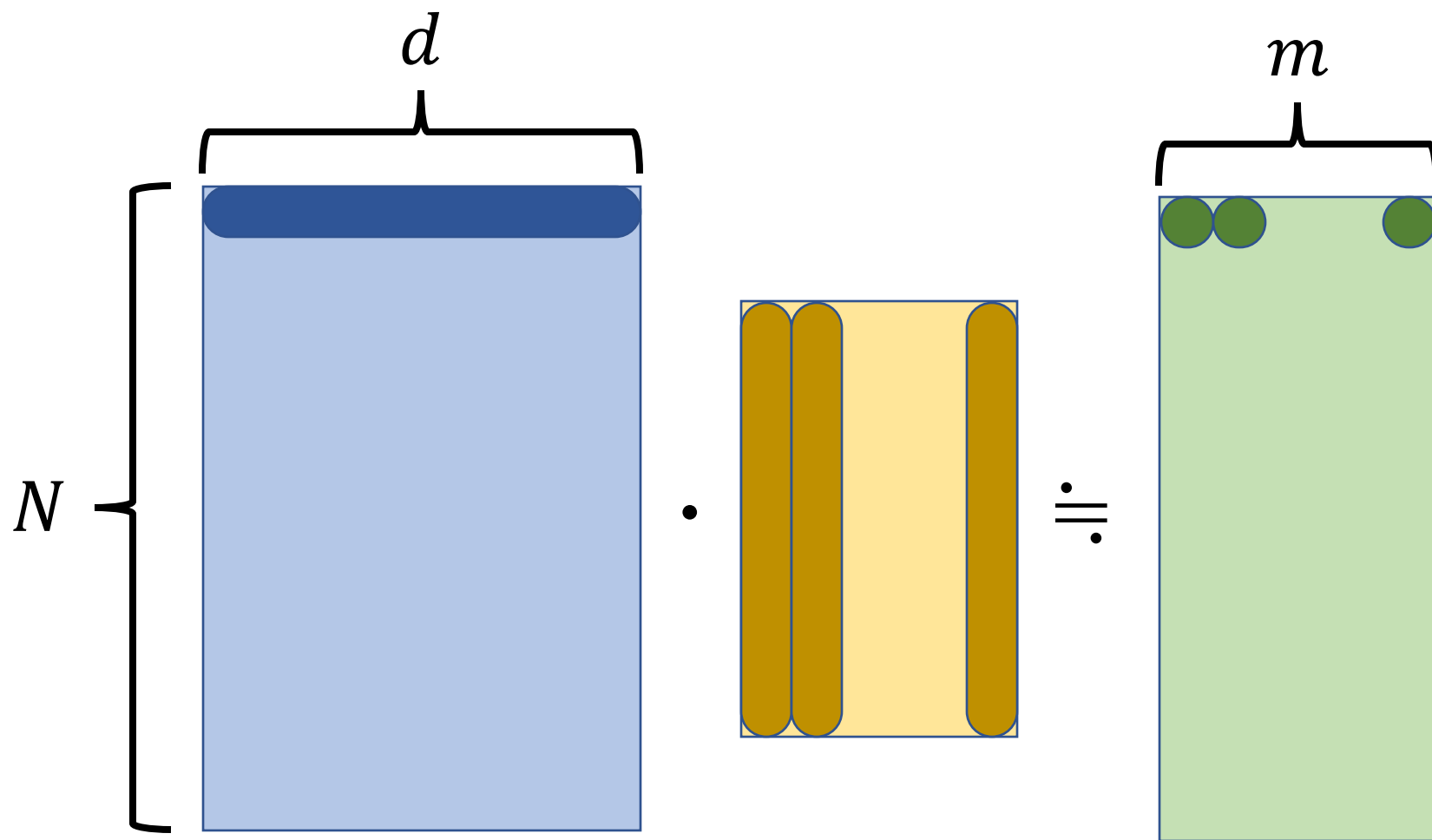
$\cdot$



右辺のほうが  
less noisyに  
なっている



PCAの場合、  
ここが単位行列になる。



- 元の $d$ 次元空間のなかに、 $m$ 本の直交する座標軸を取り...
- その軸を使って、元のデータ点の座標値を決め直す

