Feature Selection and Dimension Reduction
(第 9 章 特徵挑選與維度縮減)

Chia-Yen Lee, Ph.D. (李家岩 博士)

Department of Information Management (資訊管理學系)
National Taiwan University (國立台灣大學)

目次

- 第一章 製造數據科學
- 第二章 製造系統分析與管理
- 第三章 數據科學基礎與模型評估
- 第四章 數據科學分析架構與系統運算決策
- 第五章 數據預處理與製造數據特性
- 第六章 線性分類器
- 第七章 無母數迴歸與分類
- 第八章 決策樹與集成學習
- **第九章 特徵挑選與維度縮減**
- 第十章 類神經網路與深度學習
- 第十一章 集群分析
- 第十二章 特徵工程、數據增強與數據平衡
- 第十三章 故障預測與健康管理
- 第十四章 可解釋人工智慧
- 第十五章 概念漂移
- 第十六章 元啟發式演算法
- 第十七章 強化學習

藍：老師課堂講授
綠：學生自學

- 附錄A 線性迴歸
- 附錄B 支持向量機
- 附錄C 統計製程管制與先進製程控制
- 附錄D 超參數最佳化

- 應用涵蓋

  產能規劃、瑕疵檢測、製程監控與診斷、機台保養、需求預測、 生產排程、電腦視覺、自動光學檢測、原料價格預測與採購等

特徵挑選

為什麼要選重要參數/因子?

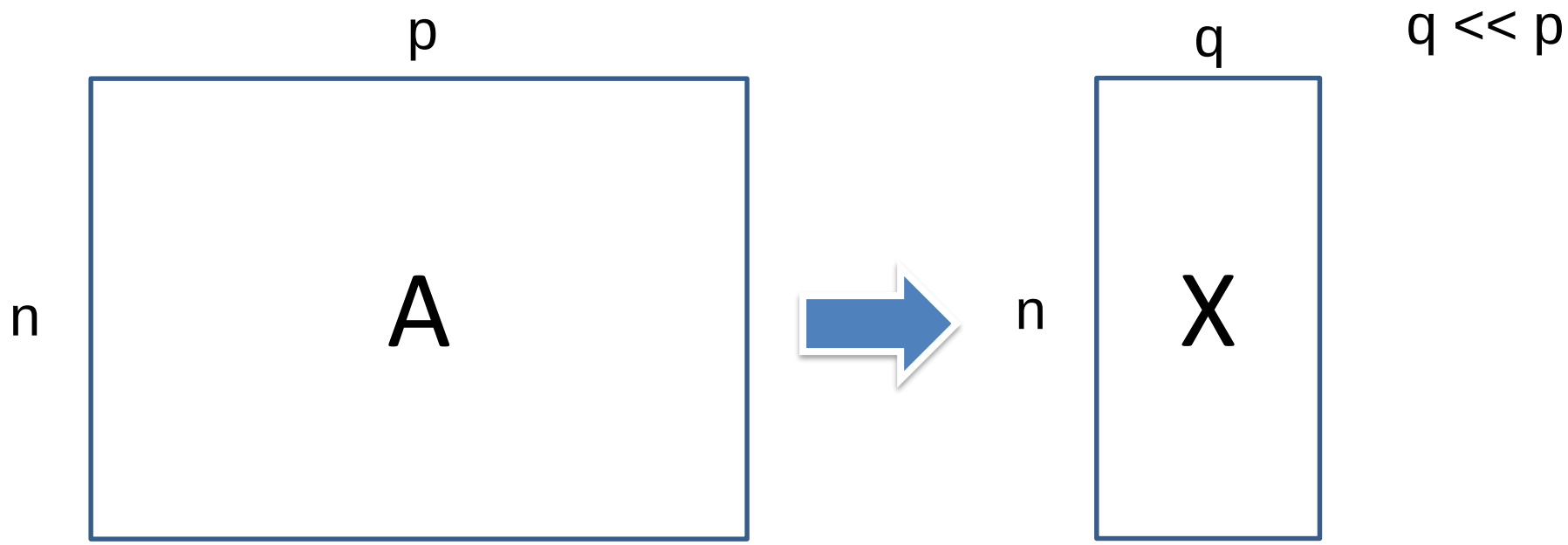
- 易於決策判斷
  - 看"____"就知道要不要決定出去旅遊
  - 看"____"就知道這衣服適不適合
  - 量"____"就知道身體健康狀況

製造現場篩選重要因子的目的

- Troubleshooting
  - 掌握影響機台品質變異的主要因子、上下游因子的交互作用
  - Engineering Process Control (EPC)
- 建立管理機制、簡單法則
  - 看Bottleneck就可推估現場WIP level
- 精度預測?
  - 提升預測準確度
  - On-line real-time prediction
- 即時線上監控Monitoring → 用較低成本/較少管制圖
  - 環境因子監控、機台參數監控

特徵挑選思維

- 「特徵挑選」（feature selection）
  - 是指從原始數據中找出重要特徵集合的方法，換言之，從原始的p個參數中，挑選出重要的q個，且 $q \ll p$ 。
  - Address “Curse of Dimensionality”
    - The number of observations required exponentially grows to estimate the function or model parameters.



Feature Selection

Objective (Guyon and Elisseeff, 2003)

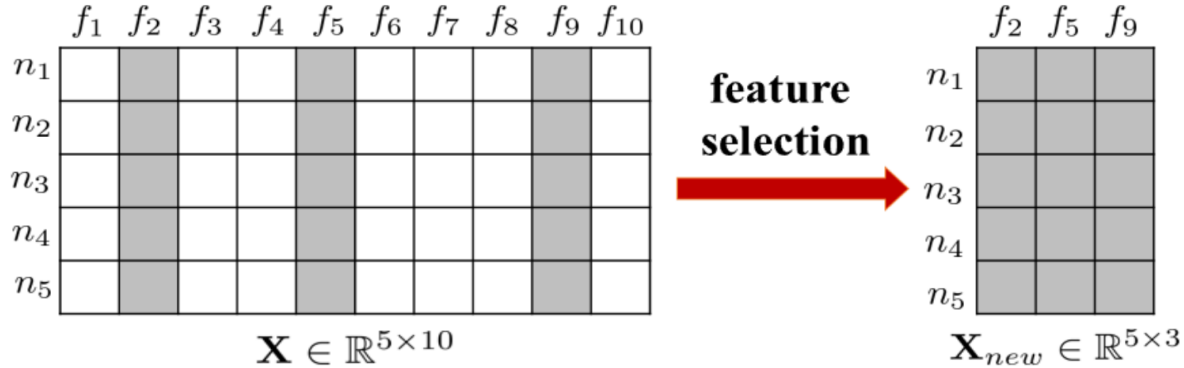
- improving the prediction performance of the predictors
- providing faster and more cost-effective predictors
- providing a better understanding of the underlying process that generated the data. (eg. for process monitoring)

Types

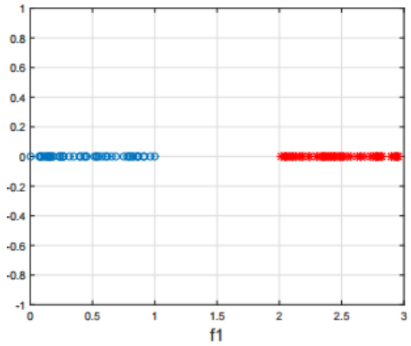
- Feature/Variable selection: select the "best" subset of the existing variables/features without a transformation.
  - Supervised Learning (監督式學習) with "Y" as label
  - Eg. stepwise regression, LASSO, random forest, etc.
- Dimension Reduction (feature extraction/ variable transformation): transforming the existing variables into a lower dimensional space
  - Unsupervised Learning (非監督式學習) with only "X"
  - Eg. independent component analysis (ICA), principal component analysis (PCA), etc.

Feasure/Variable Selection

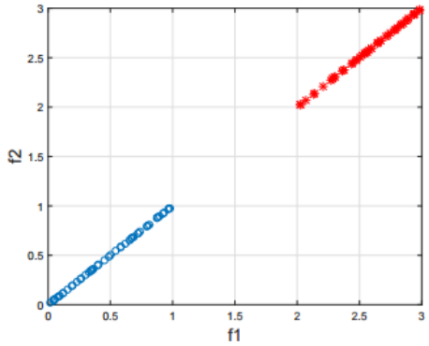
- Variable selection selects an "optimal" subset of features from the original high-dimensional feature set with a certain criterion



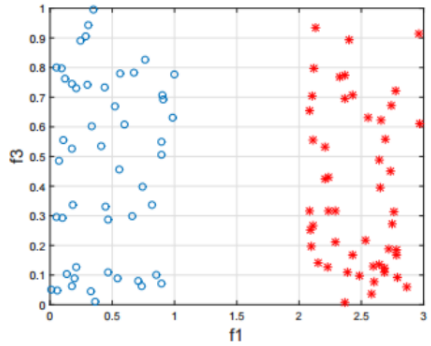
- Variable selection keeps relevant features for learning and removes redundant and irrelevant features
  - For example, for a binary classification task ($f1$ is relevant; $f2$ is redundant given $f1$; $f3$ is irrelevant)



(a) relevant feature f_1



(b) redundant feature f_2



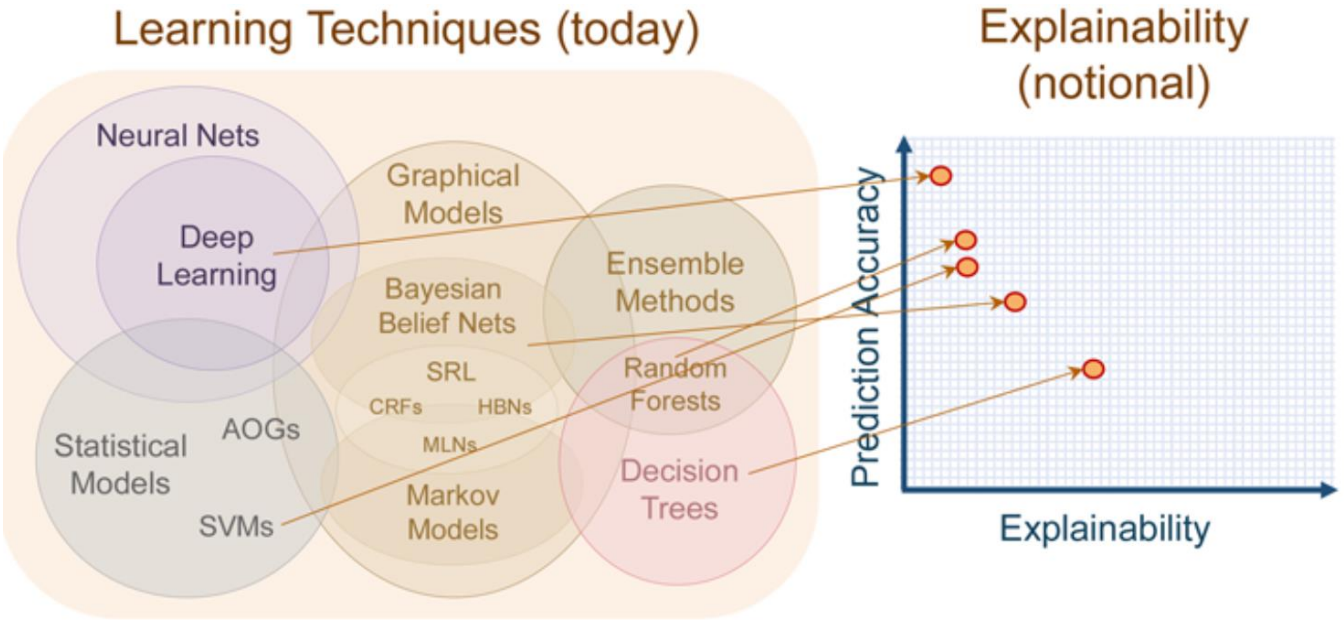
(c) irrelevant feature f_3

Dimension Reduction/Feature Extraction

- Project the original high-dimensional data into a new feature space of low dimensionality
- Given a set of n data instances with p features, obtain the low-dimensional representations:
  - $\mathbf{x}_i \in \mathbb{R}^p \rightarrow \mathbf{y}_i \in \mathbb{R}^k$ ($k \ll p$)
- The new feature space is usually a linear or a nonlinear combination of the previous feature space
  - Linear: PCA, ICA, LDA (Linear Discriminant Analysis)…
  - Nonlinear: ISOMAP (Isometric Mapping), LLE (Locally linear embedding) …
- The new features often do not have physical meanings

Variable Selection vs. Dimension Reduction

- Commonalities
  - Speed up the learning process and Reduce the storage requirements
  - Improve the learning performance and Build more generalized models
- Differences
  - Dimension reduction obtains new features while variable selection selects a subset of original ones
  - Variable selection maintains physical meanings and gives models better readability and interpretability



With variable selection, both the accuracy and interpretability of most learning algorithms can be enhanced!

特徵挑選思維

- 理論上特徵挑選目的
  - 可**避免過度配適**(overfitting)的發生，即是減少無關的資訊。
  - **避免多元共線性**(multi-collinearity)的發生，即是減少資訊重疊性。
  - **緩和維度詛咒**(curse of dimensionality)的問題，即是減少計算時間與資源的耗費。

- 實務上特徵挑選目的
  - 增強**預測準確度**
  - 簡化模型、提升模型的**可解釋性**、釐清**因果關係**、探索**物理意義**以了解問題的本質
  - 挑選後的少數重要特徵能提供**有效率以及節省成本的決策**

特徵挑選方法

- 特徵挑選的種類可分為
  - 過濾法(filter, feature ranking)
  - 包裝法(wrapper, subset selection)
  - 嵌入法(embedded)
  - 維度縮減法(dimension reduction)

特徵挑選方法：過濾法

- 過濾法(filter, feature ranking)
  - 透過直接探討各別自變數與應變數的相關性，可選擇不同的指標作為參考依據，這些指標包含**皮爾森相關係數、傑卡德相似係數、變異數分析**等，最後依照相關性指標高低排序後，挑選出高相關的自變數(或刪除低相關變數)

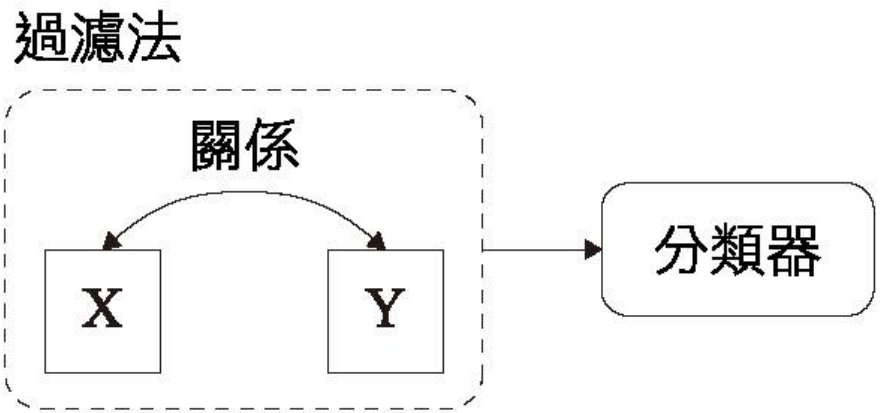


圖 9.1　過濾、包裝、嵌入法與維度縮減的比較

特徵挑選方法：過濾法

- 變數挑選的邏輯
  - 天下雨 → 地濕 (若A則B：A稱為B的充分條件，B稱為A的必要條件)
  - 地濕 → 不一定天下雨 (B不一定則A)
  - 地沒濕 → 天一定沒下雨 (非B則非A)

- 同理...
  - 因果 → 相關
  - 相關 → 不一定因果
  - 沒相關 → 絕對沒因果

- 結論
  - 當變數欄位太多時，可試著透過X跟Y的相關係數，來刪除低相關的變數欄位 (強假設：因子之間無交互作用)
  - 接著應該做 實驗設計(DOE)、Fused Lasso、Tree-based Method, etc. 確認交互作用影響

Lee, C.-Y., and Chien, C.-F., 2022. Pitfalls and protocols of data science in manufacturing practice. Journal of Intelligent Manufacturing, 33, 1189–1207.

特徵挑選方法：過濾法

表 9.1　相關性指標的類型

變數一＼變數二	連續	順序	類別
連續	皮爾森相關係數	斯皮爾曼等級相關	點二相關係數
順序	斯皮爾曼等級相關	斯皮爾曼等級相關 肯德爾等級相關係數	排序二相關係數
類別	點二相關係數	排序二相關係數	Phi 相關係數 卡方檢定 傑卡德相似係數

Similarity of Binary Vector

Similarity of Two Binary Vectors

- The similarity value (or correlation coefficient) is between 0 and 1. 1 indicates a total correlation and 0 indicates no correlation.

- If x1 and x2 are two binary vectors with n elements. There are four counters which could be built to estimate the frequency between two elements in the same position.

  - f_{00}: the counts with x1=0 and x2=0
  - f_{01}: the counts with x1=0 and x2=1
  - f_{10}: the counts with x1=1 and x2=0
  - f_{11}: the counts with x1=1 and x2=1

Example:
x1 = (0 1 0 1 0 1)
x2 = (1 1 1 0 0 0)

we can derive

$$f_{00} = 1$$
$$f_{01} = 2$$
$$f_{10} = 2$$
$$f_{11} = 1$$

Similarity of Binary Vector

- Simple Matching Coefficient, SMC (簡單配對係數)
  - $SMC = \frac{\text{coutns of both equal to 0 or equal to 1}}{\text{number of all elements in the vector}} = \frac{f_{00}+f_{11}}{f_{00}+f_{01}+f_{10}+f_{11}}$

- Jaccard Coefficient
  - $Jaccard = \frac{\text{coutns of both equal to 1}}{\text{number of all elements in the vector except both equal to 0}} = \frac{f_{11}}{f_{01}+f_{10}+f_{11}}$

Similarity of Binary Vector

SMC and Jaccard

- Example:
  - x1 = (1 0 0 0 0 0 0 0 0 0)
  - x2 = (0 0 0 0 0 0 1 0 0 1)
  - we can derive
    - $f_{00} = 7$
    - $f_{01} = 2$
    - $f_{10} = 1$
    - $f_{11} = 0$
  - SMC $= \frac{f_{00}+f_{11}}{f_{00}+f_{01}+f_{10}+f_{11}} = \frac{7+0}{7+2+1+0}$=0.7
  - $Jaccard = \frac{f_{11}}{f_{01}+f_{10}+f_{11}} = \frac{0}{2+1+0}$ =0
- Sparse matrix issue

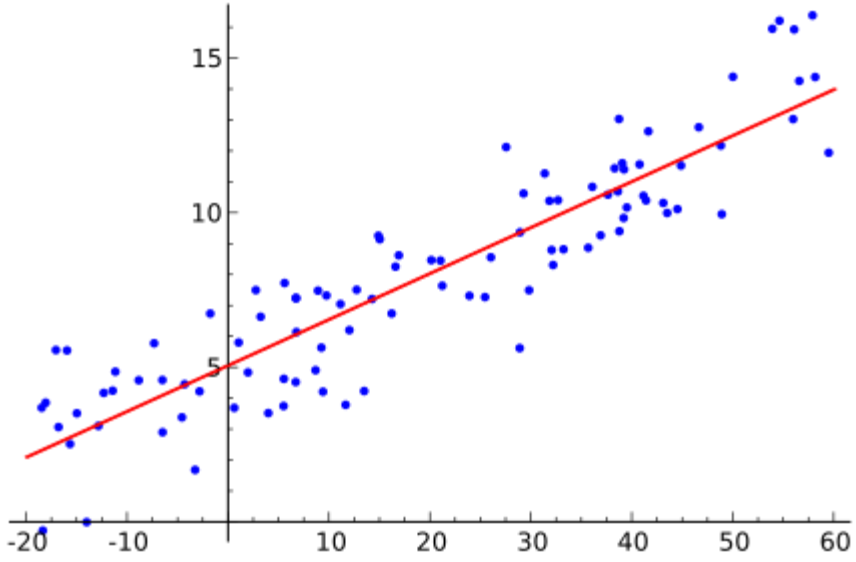
Linear Regression by OLS

- Linear Regression

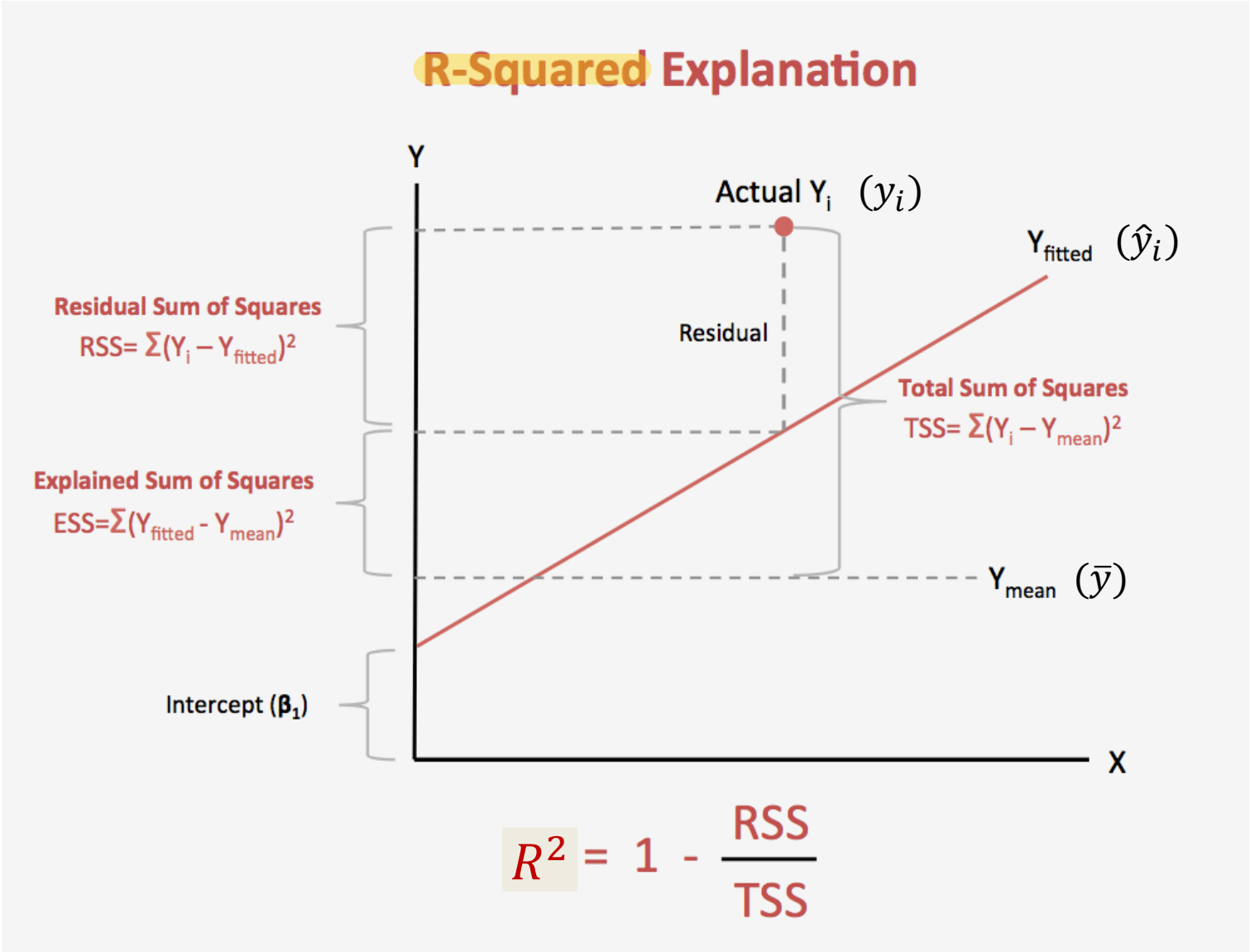
$$y_i = \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \varepsilon_i \quad \text{or} \quad y = X\beta + \varepsilon$$

- Ordinary Least Squares (OLS)

$$\text{Min} \sum_{i=1}^{n} (y_i - x_i^{\mathrm{T}} b)^2 = (y - Xb)^{\mathrm{T}} (y - Xb) \Rightarrow \hat{\beta} = (X^{\mathrm{T}} X)^{-1} X^{\mathrm{T}} y$$



Linear Regression by OLS



https://blog.csdn.net/weixin_40904071/article/details/95933283

誤差變數的必要條件

- 為了使這方法有效，以下三個與誤差變數(ε)之機率分配有關的假設必須被滿足。
  - 常態性：ε 的機率分配為常態。機率分配的平均數為 0；也就是，$E(\varepsilon) = 0$。
  - 同質性：ε 的標準差為 σ_ε，無論 x 的值為何， σ_ε是一個常數。
  - 獨立性：與任何特定 y 值相關的 ε 值與任何其他 y 值相關的 ε 值是獨立的。
    - $Cov(\varepsilon_i, \varepsilon_j) = 0$

OLS: Feature Ranked by p-value

- 鳶尾花資料

```
## Call:
## lm(formula = Sepal.Length ~ Sepal.Width + Petal.Length + Petal.Width,
##     data = iris)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82816 -0.21989  0.01875  0.19709  0.84570
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.85600    0.25078   7.401 9.85e-12 ***
## Sepal.Width   0.65084    0.06665   9.765  < 2e-16 ***
## Petal.Length  0.70913    0.05672  12.502  < 2e-16 ***
## Petal.Width  -0.55648    0.12755  -4.363 2.41e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3145 on 146 degrees of freedom
## Multiple R-squared:  0.8586, Adjusted R-squared:  0.8557
## F-statistic: 295.5 on 3 and 146 DF,  p-value: < 2.2e-16
```

花萼寬度

花瓣長度

特徵挑選方法：包裝法

- 包裝法(wrapper, subset selection)
  - **使用搜尋的演算法試圖找出最理想的特徵組合**，透過迭代的程序（iterative procedure），在每一次建模後，依照模型評估指標（AIC, BIC, R^2_{adj}）判斷，挑選合適的自變數，直到找到使得模型評估指標最佳的特徵組合

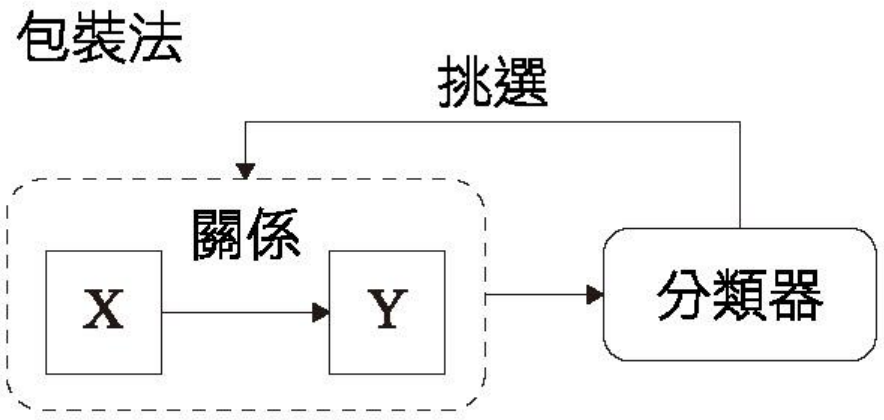


圖 9.1　過濾、包裝、嵌入法與維度縮減的比較

特徵挑選方法：包裝法

- 包裝法(wrapper, subset selection)
  - 最佳子集挑選(Best Subset)
    - 運算效率差
    - 高維度的特徵空間容易導致過度適配
  - 逐步挑選(Stepwise Selection)
    - Forward stepwise regression
    - Backward stepwise regression
    - Stepwise regression

Stepwise Selection (i.e. stepwise regression逐步迴歸)

- Starting with no variables in the model, then adding the variable (if any) that improves the model the most (with smallest p-value).
- Forward selection (y, x_1, x_2, x_3, x_4)

$$y = \alpha + \beta x_1 + \varepsilon$$
$$y = \alpha + \beta x_2 + \varepsilon$$
$$y = \alpha + \beta x_3 + \varepsilon$$
$$y = \alpha + \beta x_4 + \varepsilon$$

$$y = \alpha + \beta_2 x_2 + \beta x_1 + \varepsilon$$
$$y = \alpha + \beta_2 x_2 + \beta x_3 + \varepsilon$$
$$y = \alpha + \beta_2 x_2 + \beta x_4 + \varepsilon$$

$$y = \alpha + \beta_2 x_2 + \beta_4 x_4 + \beta x_1 + \varepsilon$$
$$y = \alpha + \beta_2 x_2 + \beta_4 x_4 + \beta x_3 + \varepsilon$$

統計顯著的變數
x_2 and x_4

特徵挑選方法：包裝法

最佳子集挑選 vs. 逐步挑選

- 相較於最佳子集挑選需考慮2^p種組合
- 向前逐步挑選每此迭代時需考慮$p-k$個模型(k為迭代次數，$k=0,\ldots,p-1$)，一共會配適$\sum_{k=0}^{p-1}(p-k)=1+p(p+1)/2$個模型
  - 舉例而言，若考慮一筆$p=20$的數據，最佳子集挑選需配適1,048,576個模型，而向前逐步挑選僅需配適211個模型
  - 向前逐步挑選可以被應用在高維度$p>n$的問題中
- 向前逐步挑選缺點
  - 在於模型的**順序相依性**(sequence dependence)，亦即模型每次選擇的變數是相依於前一個最佳的模型，倘若前幾次迭代中所放入的某個自變數，隨著新變數的放入使得模型效果越差時，無法將其剔除，一般來說會是部分**多元共線性**所造成的結果。

表 9.5　最佳子集與向前逐步比較

特徵個數	最佳子集	向前逐步
一個	電流	電流
兩個	電流、溫度	電流、溫度
三個	電流、溫度、壓力	電流、溫度、壓力
四個	加速度、溫度、壓力、酸鹼度	電流、溫度、壓力、酸鹼度

特徵挑選方法：包裝法

- 包裝法(wrapper, subset selection)

表 9.7　包裝法的比較

方法＼特性	使用時機		優點	缺點
	問題維度	運算資源		
最佳子集挑選	$p < n$	非常多	可得到全域最佳的特徵子集	運算負擔非常龐大，與維度呈指數成長
向前逐步挑選	皆可，常針對 $p > n$	較少	1. 運算效率高 2. 可處理高維度問題	1. 受嚴重的順序相依性影響 2. 當重要特徵過多時運算負擔也會增加
向後逐步挑選	$p < n$	少	1. 運算效率高 2. 受較小的順序相依性影響	不易處理高維度問題
逐步迴歸	皆可，常針對 $p > n$	多	1. 受較小的順序相依性影響 2. 可處理高維度問題	當重要特徵過多時運算負擔也會增加

特徵挑選方法：嵌入法

- 嵌入法(embedded)
  - 為監督式學習，使用具備自行特徵挑選能力的模型，包含了**藉由壓縮係數找出重要特徵的方法**（shrinkage method），以及**使用集成學習方法並計算各變數重要性**（feature importance）的方法，將特徵挑選機制嵌入於模型中

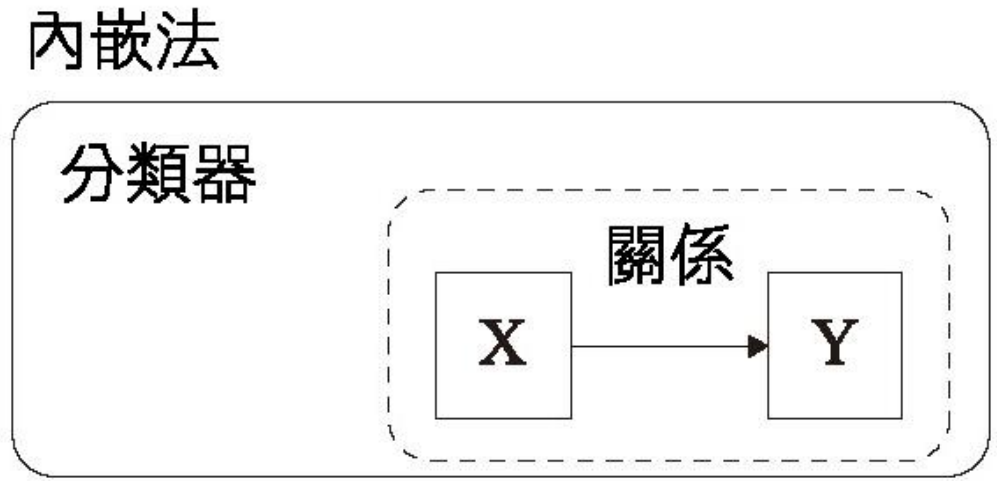


圖 9.1　過濾、包裝、嵌入法與維度縮減的比較

特徵挑選方法：嵌入法

嵌入法(embedded)– 正則化(regularization)

- Ridge Regression脊迴歸 (L2 penalty)
  - $\hat{\beta}^{ridge} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2$

    $\text{subject to } \sum_{j=1}^{p} \beta_j^2 \leq t$
- 避免**過度配適**與**多元共線性**
  - 當特徵不具資訊(例如流水號)或彼此之間高相關時，將導致迴歸係數估計的變異變得非常大，若我們對這些估計的係數加以限制，便能有效**降低變異**，但同時會部分犧牲估計的精準度，產生估計上的**偏誤**(bias)
  - 加了限制式後，此時迴歸係數估計就不再是不偏估計了→**偏估計量**
  - 根據**拉格朗日乘數法**(method of Lagrange multiplier)，若我們將限制轉成對於目標式的懲罰，最小化目標的同時也最小化懲罰項。
  - $\hat{\beta}^{ridge} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$
  - 其中λ是拉格朗日乘數，並稱$\lambda \sum_{j=1}^{p} \beta_j^2$是**懲罰項**(penalty term)或**正則項**(regularizer)。λ是一個可調整的超參數，代表懲罰項的權重。

特徵挑選方法：嵌入法

嵌入法(embedded)– 正則化(regularization)

- Ridge Regression脊迴歸 (L2 penalty)
  - 當$\lambda = 0$時，等同於原始的迴歸問題，當$\lambda \to \infty$時，所有的迴歸係數會被壓縮並逐漸趨近於零，因此正則化方法也被稱為「**壓縮方法**」(shrinkage method)。

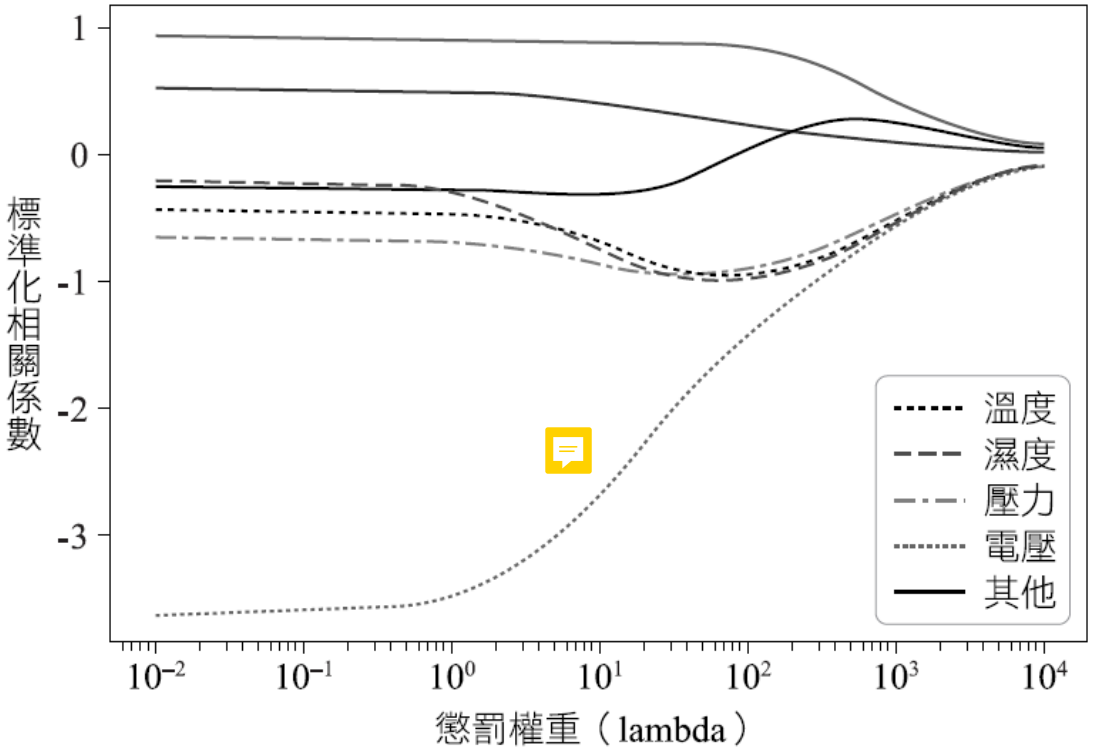


圖 9.3　脊迴歸的正則化

特徵挑選方法：嵌入法

嵌入法(embedded)– 正則化(regularization)

- Lasso Regression套索迴歸 (L1 penalty)
  - $\hat{\beta}^{lasso} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2$

    $\text{subject to } \sum_{j=1}^{p} |\beta_j| \leq t$
  - $\hat{\beta}^{lasso} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$

- 脊迴歸 vs. 套索迴歸
  - 都是藉由懲罰以增加偏誤降低變異的兩者究竟有何差異？
  - 脊迴歸隨懲罰增加將係數慢慢壓縮逼近於零但不等於零(除非$\lambda \to \infty$時所有係數才為零)
  - 套索迴歸則會隨著懲罰增加將部分係數直接壓縮為零，迴歸係數為零意指該自變數特徵沒有被挑選到
  - 因此套索迴歸更具備特徵挑選的能力，高相關特徵會被擇一挑選。

特徵挑選方法：嵌入法

- 嵌入法(embedded)– 正則化(regularization)
  - Lasso Regression套索迴歸 (L1 penalty)

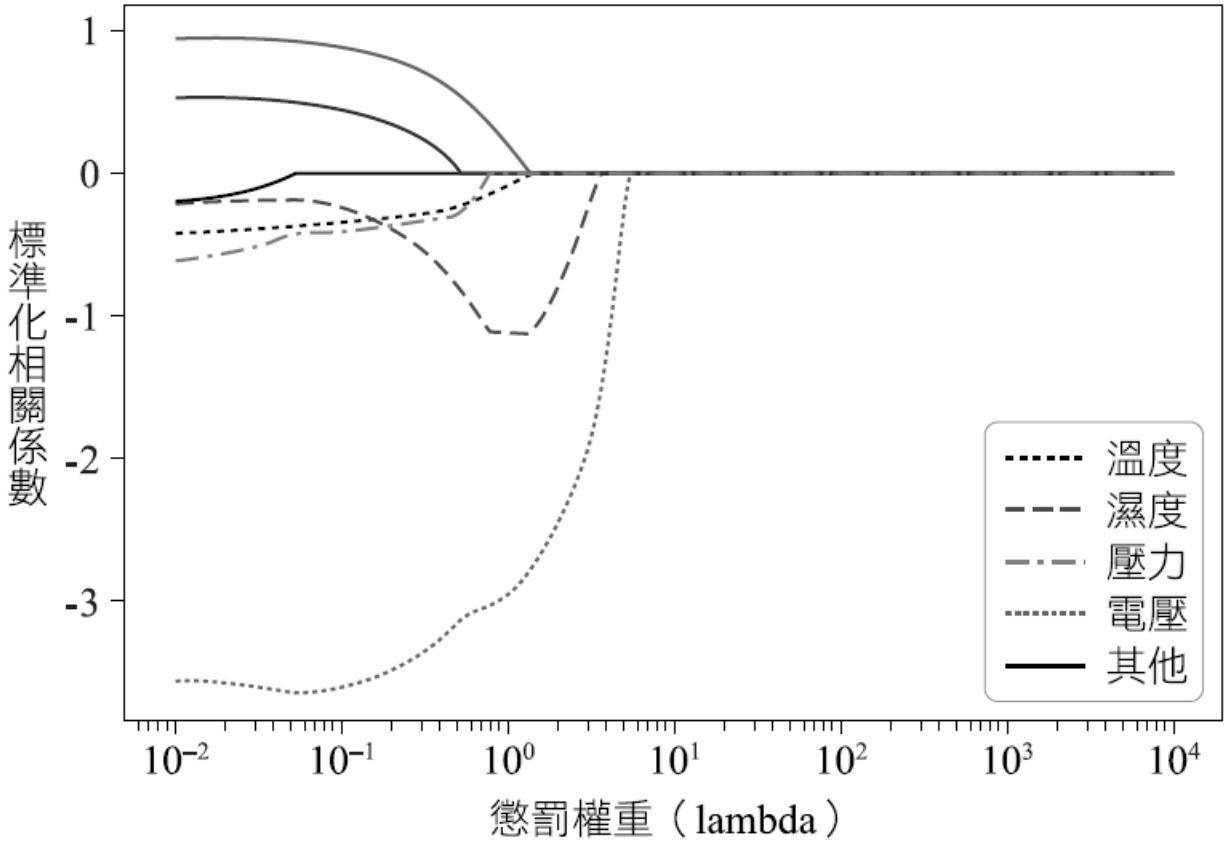


圖 9.4　套索迴歸的正則化

特徵挑選方法：嵌入法

脊迴歸 vs. 套索迴歸

- 為何套索迴歸能將迴歸係數壓縮到零呢？
  - 藉由**幾何觀點**來理解套索迴歸與脊迴歸的特性。圖中斜橢圓線為兩係數計算出訓練誤差的等高線圖，相同等高線上的係數有相同的誤差，其中中心點$\hat{\beta}$為最小平方法所估計出的**不偏估計值**(unbiased estimate)。
  - 灰色區域為我們給予的限制，左圖套索迴歸的限制為$|\beta_1| + |\beta_2| \leq t$，呈現一個45°的正方形；右圖脊迴歸的限制為$\beta_1^2 + \beta_2^2 \leq t$，呈現一個圓形。

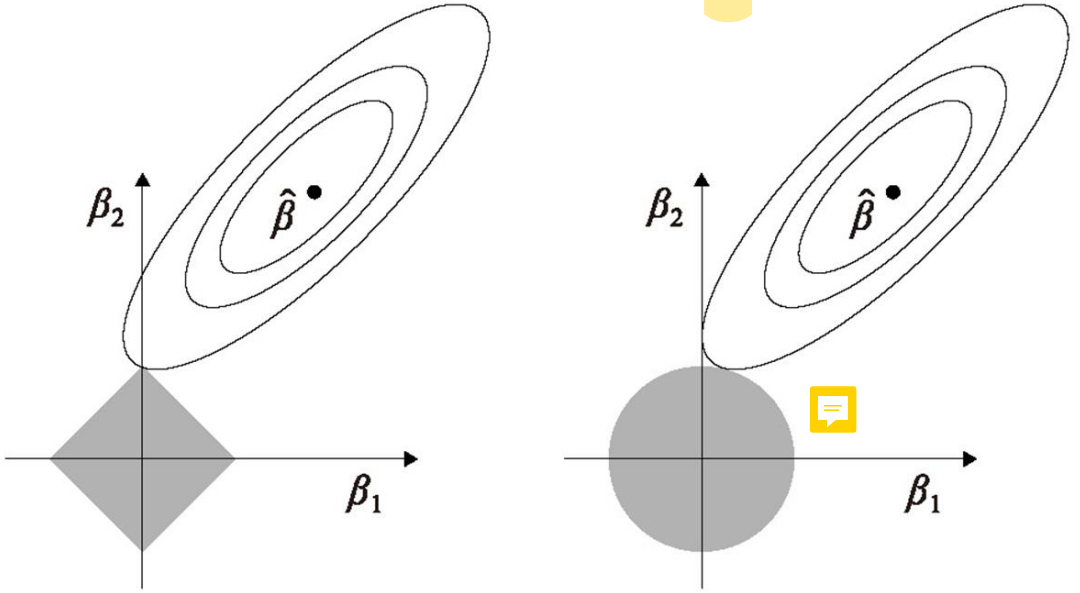


圖 9.5　套索迴歸與脊迴歸幾何示意圖（Hastie et al., 2009）

  - 若我們從不偏估計值$\hat{\beta}$出發，套索迴歸L_1懲罰會使得β_1與β_2的解必須落於正方形限制內，懲罰項λ變大意指t變小，也就是正方形的形狀可以是很寬鬆的大(與$\hat{\beta}$相交)又或是很嚴格的小(在原點上)。當正方形越小時，也就是我們給予的限制越強時，某一項係數(β_1或β_2)會先被壓縮到零(先觸碰到正方形的一角)；然而脊迴歸L_2懲罰則使得β_1與β_2遞減方式會以某種曲率的形式，因此隨著圓形的縮小直到原點所有係數才皆被壓縮至零。

特徵挑選方法：嵌入法

嵌入法(embedded)– 正則化(regularization)

表 9.8　最佳子集、脊迴歸與套索迴歸的特徵挑選示意公式

估計方法	公式
最佳子集（Best subset）（特徵數為M）	$\hat{\beta}_j \cdot I\left(\lvert\hat{\beta}_j\rvert \geq \lvert\hat{\beta}_{(M)}\rvert\right)$
脊迴歸（Ridge）	$\hat{\beta}_j/(1+\lambda)$
套索迴歸（Lasso）	$\text{sign}(\hat{\beta}_j)\left(\lvert\hat{\beta}_j\rvert - \lambda\right)_+$

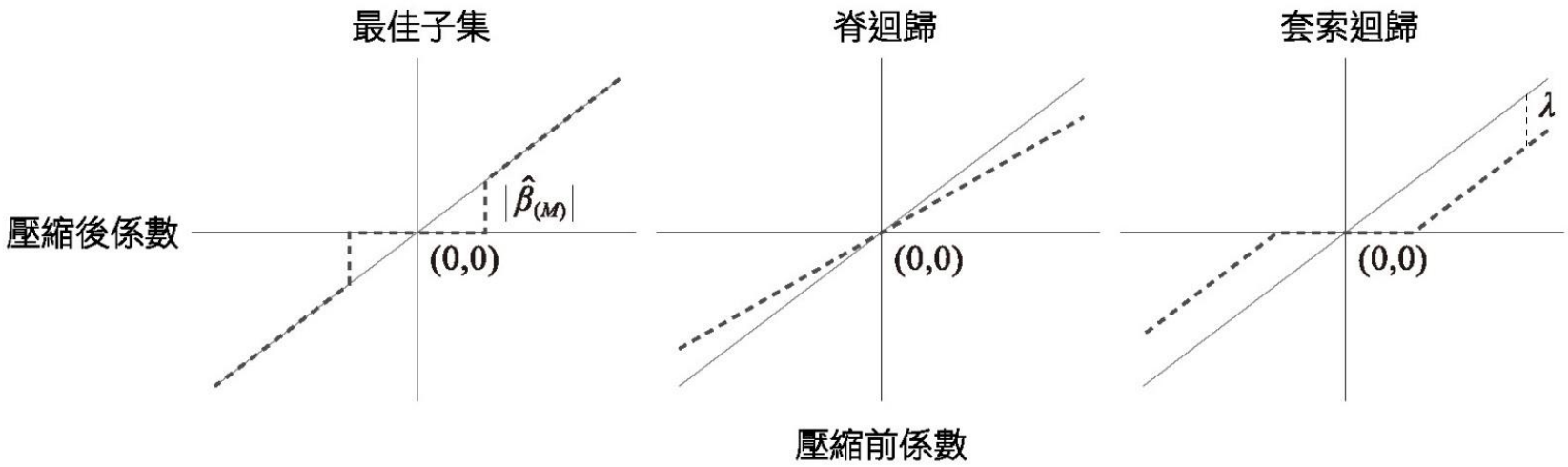


圖 9.6　最佳子集、脊迴歸與套索迴歸的特徵挑選示意圖（Hastie et al., 2015）

Elastic Net

- When p > n, lasso can select only n variables and it tends to select one variable from any set of highly correlated variables.
- When n > p, ridge regression tends to perform better given strongly correlated variables.(壓縮高相關特徵至一致係數大小的能力)
- Combine ridge and lasso

The elastic net extends lasso by adding an additional ℓ^2 penalty term giving

$$\min_{\beta \in \mathbb{R}^p} \left\{ \|y - X\beta\|_2^2 + \lambda_1 \|\beta\|_1 + \lambda_2 \|\beta\|_2^2 \right\},$$

which is equivalent to solving

$$\min_{\beta_0, \beta} \left\{ \|y - \beta_0 - X\beta\|_2^2 \right\} \text{ subject to } (1-\alpha)\|\beta\|_1 + \alpha\|\beta\|_2^2 \leq t,$$

$$\text{where } \alpha = \frac{\lambda_2}{\lambda_1 + \lambda_2}.$$

- Therefore, highly correlated covariates tend to have similar regression coefficients (grouping effect).

偏誤修正方法一：Adaptive Lasso

Adaptive Lasso

- Two-stage procedure instead of just using L_1-penalty:
- $\hat{\beta}^{Ada} = \underset{\beta}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \hat{w}_j |\beta_j| \right\}$
  - Let $\gamma > 0$ and $\hat{w}_j = |\hat{\beta}_j|^{-\gamma} \geq 0$, where $\hat{\beta}_j$ is an initial estimator of the coefficients.
- Adaptive Lasso enjoys the oracle properties
  - (1) Consistency in variable selection; $\lim_{n\to\infty} \mathbb{P}(\hat{\mathcal{A}} = \mathcal{A}) = 1$, as $n \to \infty$
  - (2) Asymptotic normality $\sqrt{n}(\hat{\beta}_{\mathcal{A}} - \hat{\beta}_{\mathcal{A}}) \to_d \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, as $n \to \infty$
- Lasso can be used for the initial estimation.
- Adaptive Lasso gives a small penalty to β_j with initial estimates of large magnitude and then increase penalty over iteration.
- 因而此懲罰項為加權的L1 penalty。此懲罰對於係數趨於零的特徵有較大的懲罰（可能為共線性或不顯著的特徵）；反之，對於非零係數則有較小的懲罰，因而受到偏誤的影響較小
- Adaptive Elastic Net

偏誤修正方法一：Adaptive Elastic Net

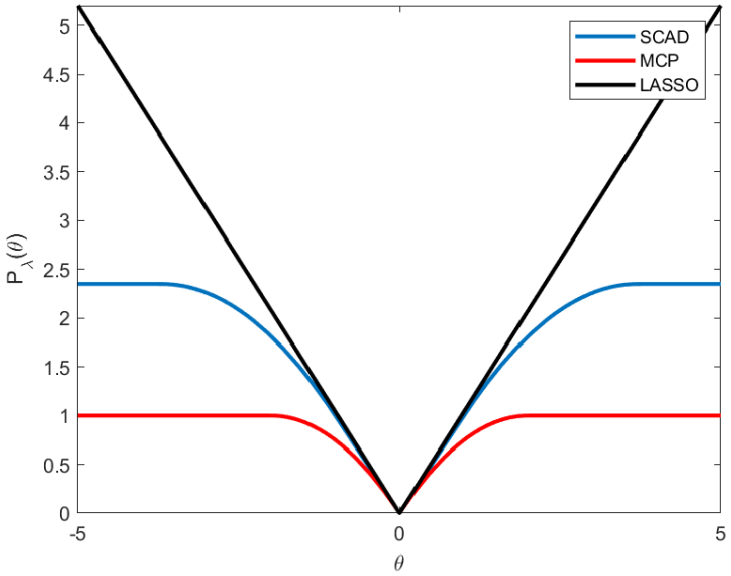
Adaptive Elastic Net

- 在實證上很難跟大家說明LASSO(L1)一定比較好，因為收斂速度或預防overfitting有時候ridge(L2)會有較好的表現

- Adaptive Elastic Net
  - http://users.stat.umn.edu/~zouxx019/Papers/aenet.pdf

偏誤修正方法二：Folded Concave Penalties(FCPs)

- Use Lasso still provides a “bias estimate”
  - Methods: folded concave penalties (FCPs)
    - smoothly clipped absolute deviation (SCAD) penalty
    - minimax concave penalty (MCP).
  - Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. ***Journal of the American Statistical Association***, 96(456):1348{1360.
  - Zhang, C. H. (2010). Nearly unbiased variable selection under minimax concave penalty. ***Annals of Statistics***, 38(2):894{942.



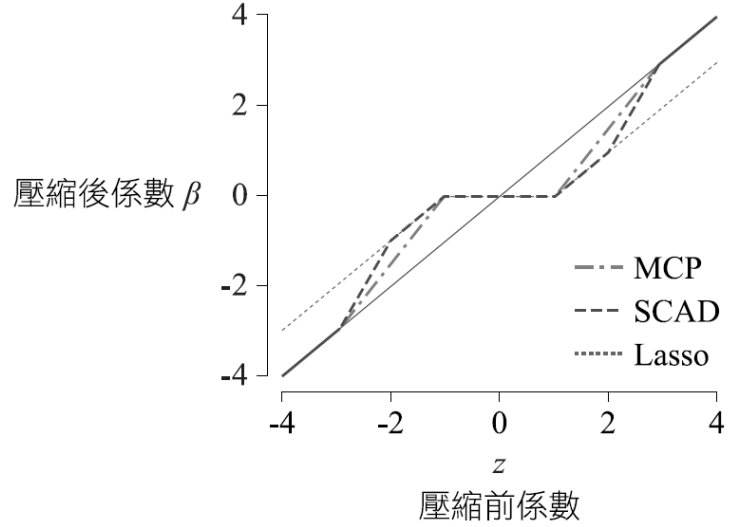


圖 9.7　折疊式凹性懲罰

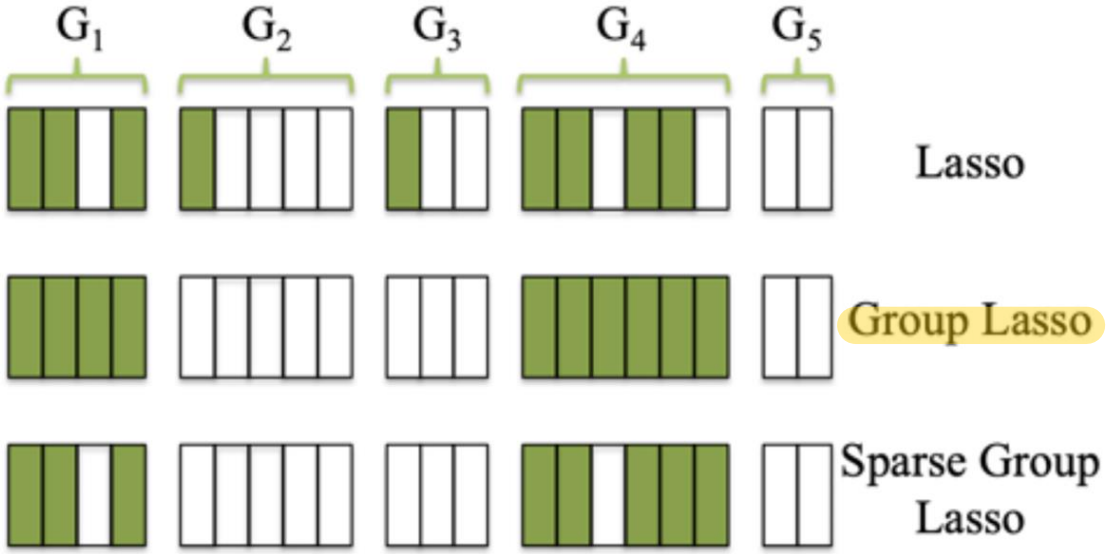
Extension: Group Lasso

Group Lasso

- Allow predefined groups of variables to jointly be selected or removed.
- Use: one-hot encoding for categorical variable coded as binary variables.
- $$\min_{\beta \in \mathbb{R}^p} \left\{ \left\| y - \sum_{j=1}^{J} X_j \beta_j \right\|_2^2 + \lambda \sum_{j=1}^{J} \|\beta_j\|_{K_j} \right\}, \quad \|z\|_{K_j} = (z^t K_j z)^{1/2}$$

  (J groups)

  - the penalty term is a sum over L_2-norms defined by the positive definite matrices K_j.



Bai et al. (2020)

Group lasso with overlap allows covariates to be shared across groups, e.g., if a gene were to occur in two pathways.

Bai, Y., Calhoun, V. D., and Wang, Y.-P. (2020). Integration of multi-task fMRI for cognitive study by structure-enforced collaborative regression. Conference of Biomedical Applications in Molecular, Structural, and Functional Imaging.

Extension: Fused Lasso

Fused Lasso

- spatial or temporal structure that must be considered, such as time series or image-based data.

$$\min_{\beta} \left\{ \frac{1}{N} \sum_{i=1}^{N} \left(y_i - x_i^t \beta \right)^2 \right\}$$

$$\text{subject to } \sum_{j=1}^{p} |\beta_j| \leq t_1 \text{ and } \sum_{j=2}^{p} |\beta_j - \beta_{j-1}| \leq t_2$$

- 2nd constraint penalizes large changes about the temporal or spatial structure, which forces the coefficients to vary smoothly to reflect the system's underlying logic.
- Clustered lasso is a generalization of fused lasso that identifies and groups relevant variables.

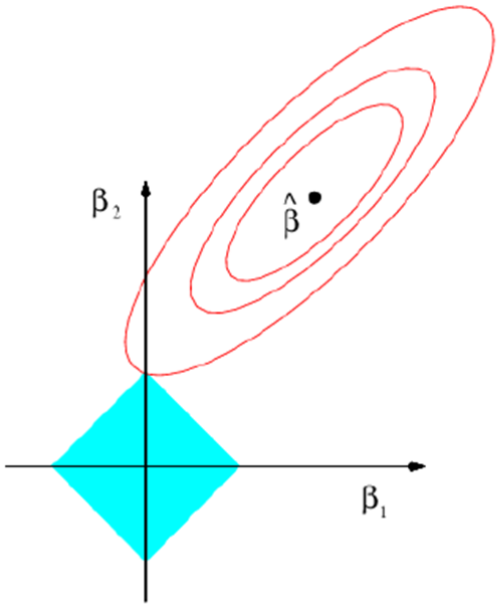
$$\sum_{i<j}^{p} |\beta_i - \beta_j| \leq t_2$$

- In contrast, variables can be clustered into highly correlated groups, and then a single representative variable can be extracted from each cluster.

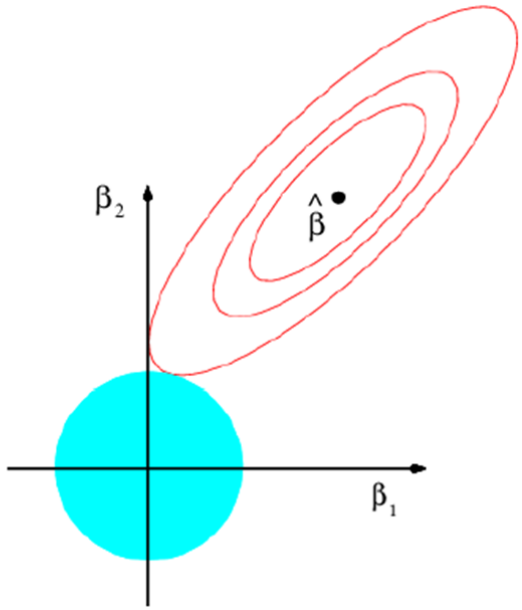
Extension: Variable Selection Property of Lasso

- Why is it that the Lasso, unlike ridge regression, results in coefficient estimates that are exactly equal to zero?

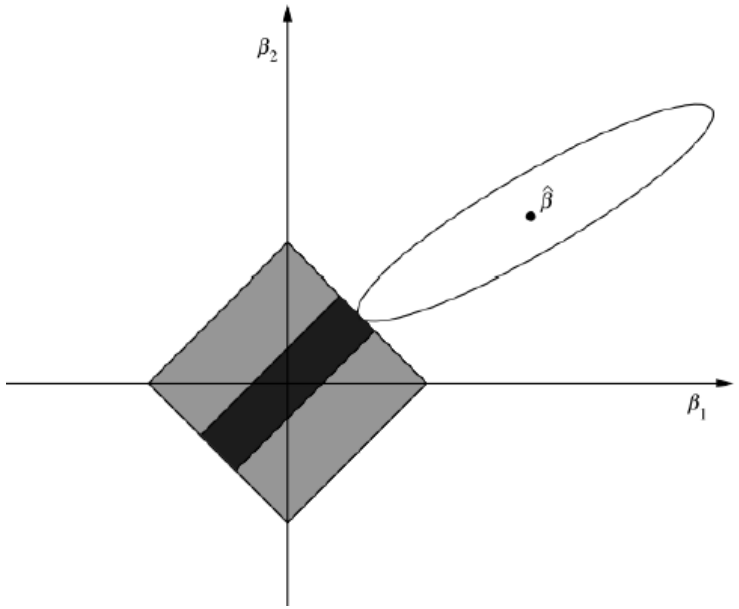
$$L(\lambda_1, \lambda_2, \beta) = |\mathbf{y} - \mathbf{X}\beta|^2 + \lambda_2|\beta|^2 + \lambda_1 \sum_{i=2}^{p} |\beta_j - \beta_{j-1}|$$



Lasso



Ridge



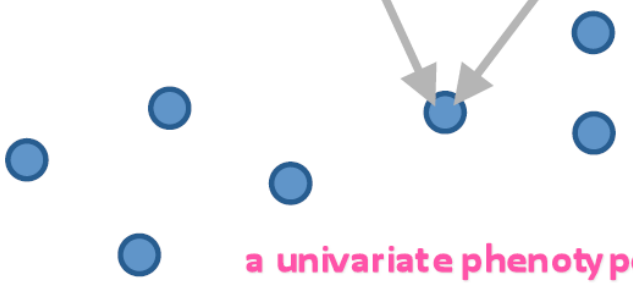
Fused Lasso

Application: Gene Association

Lasso

causal SNP

ACGTTTTACTGTACAATT

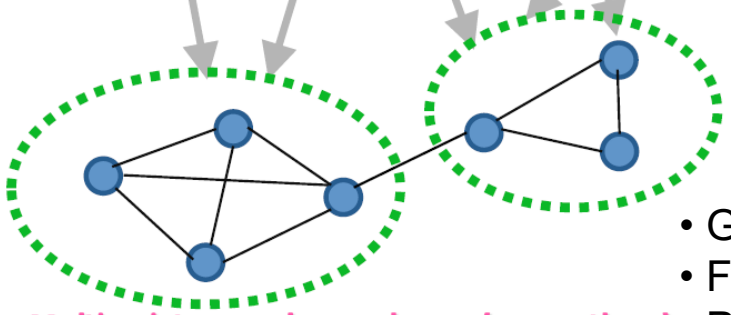


a univariate phenotype:
gene expression level

Gflasso (Kim & Xing, 2009)

Graph--guided Fused lasso

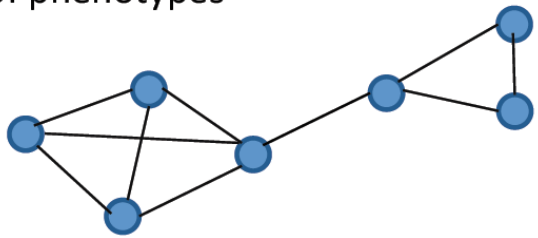
ACGTTTTACTGTACAATT



Multivariate complex syndrome (e.g., asthma)
age at onset, history of eczema
genome-wide expression profile

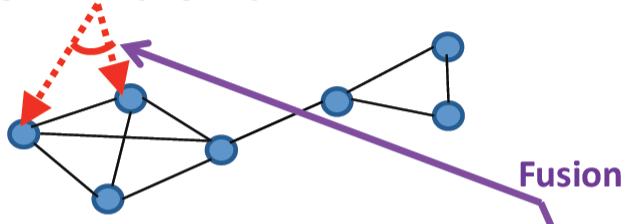
- Greater power
- Fewer false positives
- Phenome associations

Step 1: Thresholded correlation graph of phenotypes



Step 2: Graph-constrained fused lasso

ACGTTTTACTGTACAATT



Fusion

$$\hat{\mathbf{B}}^{\mathrm{GC}} = \operatorname{argmin} \sum_k (\mathbf{y}_k - \mathbf{X}\boldsymbol{\beta}_k)^T \cdot (\mathbf{y}_k - \mathbf{X}\boldsymbol{\beta}_k) + \lambda \sum_k \sum_j |\beta_{jk}| + \gamma \sum_{(m,l)\in E} \sum_j |\beta_{jm} - \operatorname{sign}(r_{ml})\beta_{jl}|$$

Lasso Penalty

Graph-constrained fusion penalty

Kim, Seyoung (2013). Advanced Topics in Computational Genomics, Spring 2013. https://www.cs.cmu.edu/~sssykim/teaching/s13/s13.html

Extension: Graph-constrained Fused Lasso

- Graph-constrained Fused Lasso
  - Quadratic programming formulation (convex optimization)
    - Graph-constrained fused lasso

$$\hat{\mathbf{B}}^{\mathrm{GC}} = \operatorname{argmin} \sum_k (\mathbf{y}_k - \mathbf{X}\boldsymbol{\beta}_k)^T \cdot (\mathbf{y}_k - \mathbf{X}\boldsymbol{\beta}_k)$$

$$\text{s. t.} \quad \sum_k \sum_j |\beta_{jk}| \leq s_1 \text{ and } \sum_{(m,l)\in E} \sum_j |\beta_{jm} - \operatorname{sign}(r_{ml})\beta_{jl}| \leq s_2$$

    - Graph-weighted fused lasso

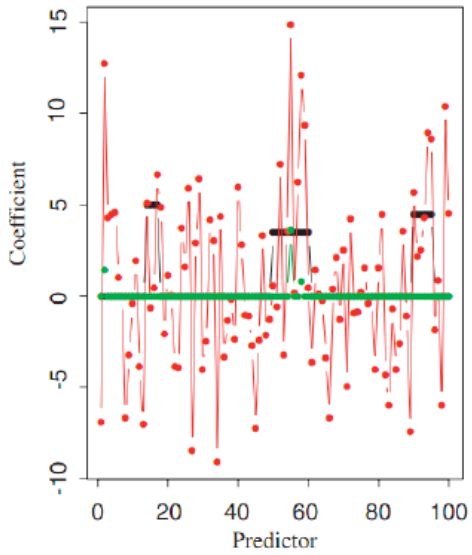
$$\hat{\mathbf{B}}^{\mathrm{GW}} = \operatorname{argmin} \sum_k (\mathbf{y}_k - \mathbf{X}\boldsymbol{\beta}_k)^T \cdot (\mathbf{y}_k - \mathbf{X}\boldsymbol{\beta}_k)$$

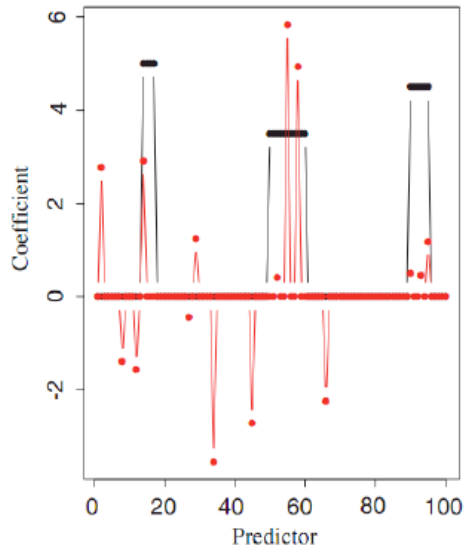
$$\text{s. t.} \quad \sum_k \sum_j |\beta_{jk}| \leq s_1 \text{ and } \sum_{(m,l)\in E} f(r_{ml}) \sum_j |\beta_{jm} - \operatorname{sign}(r_{ml})\beta_{jl}| \leq s_2$$

Kim, Seyoung (2013). Advanced Topics in Computational Genomics, Spring 2013. https://www.cs.cmu.edu/~sssykim/teaching/s13/s13.html

Application: Gene Association

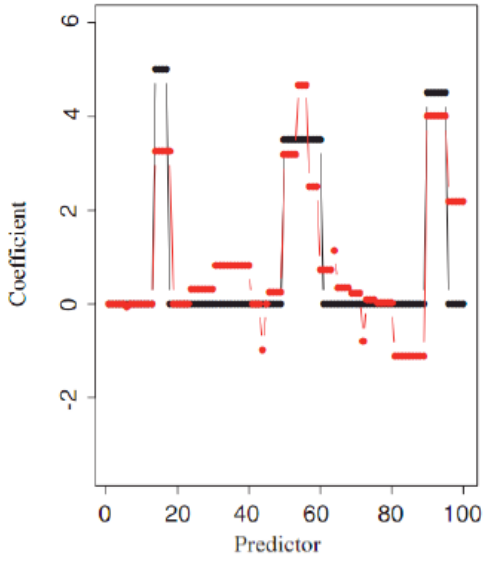
Standard regression

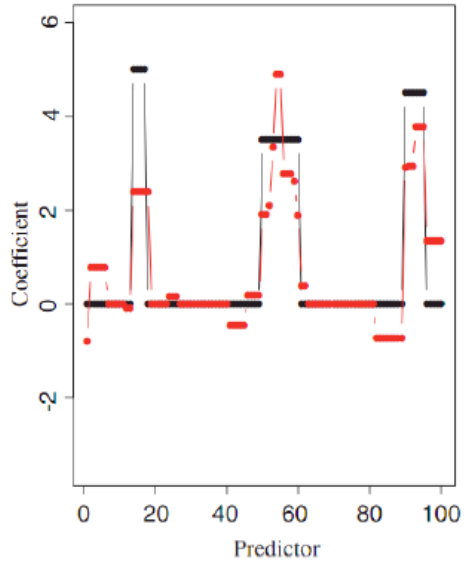




lasso

Fusion penalty only





Fused lasso

- Black line: true values
- Red line: estimated values

Kim, Seyoung (2013). Advanced Topics in Computational Genomics, Spring 2013. https://www.cs.cmu.edu/~sssykim/teaching/s13/s13.html

Application: Graph Denoising

Example: algorithms for the 2d fused

The 2d fused lasso or 2d total variation denoising prob

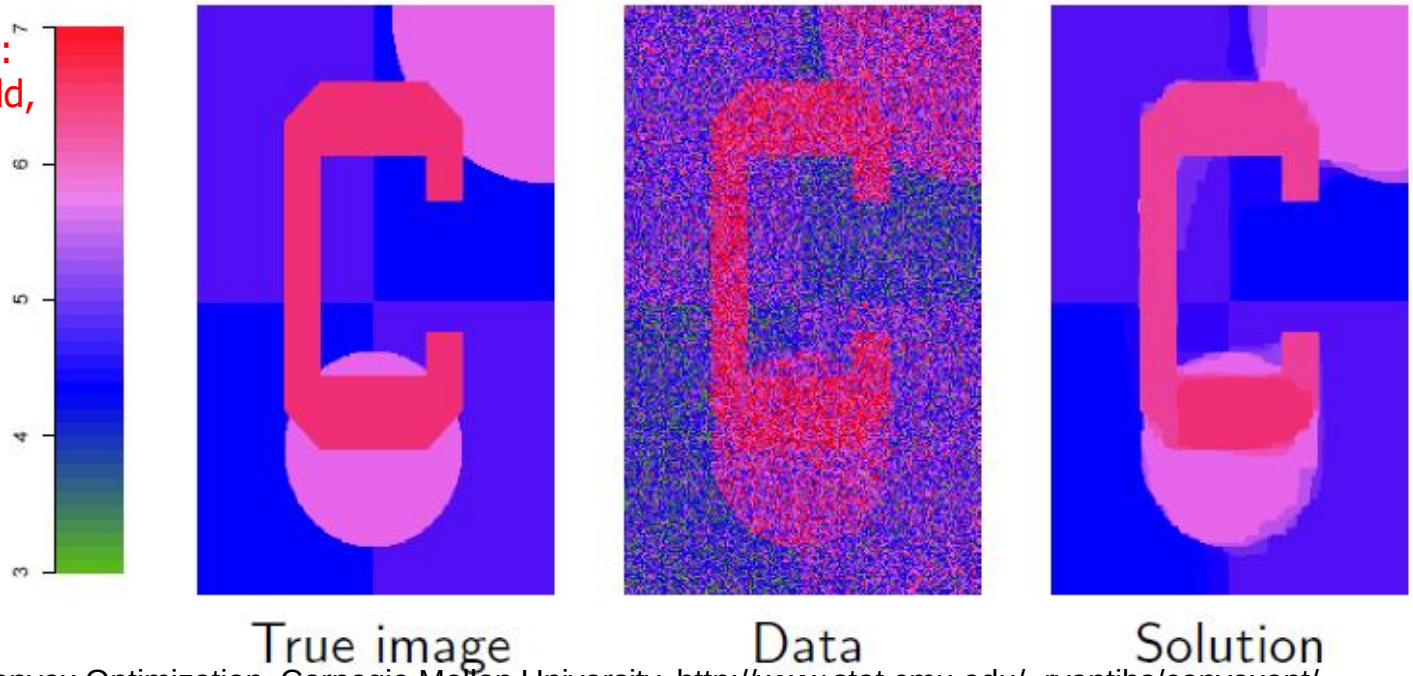
$$\min_{\theta} \frac{1}{2}\sum_{i=1}^{n}(y_i-\theta_i)^2+\lambda\sum_{(i,j)\in E}|\theta_i-\theta_j|$$

This fits a piecewise constant function over an image, $y_i,\ i=1,\ldots,n$ at pixels

Edge: adjacent points
Larger and larger lambda:
1. theta_i and theta_j are equal
2. the pixels in the same neighborhood are forced to the same color (color are closer and closer)
3. the function more and more piecewise constant. (finally just one constant value, i.e. all the same color. This one color will be the mean of yi)

Piecewise constant: over some threshold, then the color changed.

Lambda equal to zero. There is nothing about "denoising", just original dataset



Tibshirani, Ryan, Convex Optimization, Carnegie Mellon University. http://www.stat.cmu.edu/~ryantibs/convexopt/

Application: Control Chart Changepoint

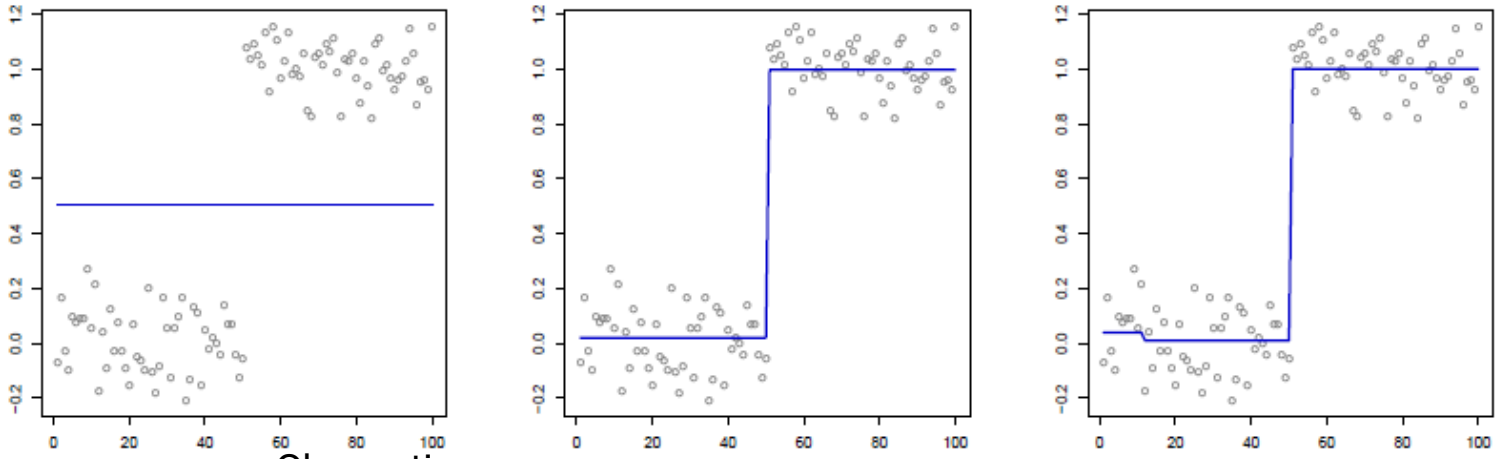
Example: testing changepoints from the 1d fused lasso

In the 1d fused lasso or 1d total variation denoising problem

$$\min_{\theta} \frac{1}{2}\sum_{i=1}^{n}(y_i-\theta_i)^2+\lambda\sum_{i=1}^{n-1}|\theta_i-\theta_{i+1}|$$

the parameter $\lambda \geq 0$ is called a tuning parameter. As λ decreases, we see more changepoints in the solution $\hat{\beta}$

value of the pixel in one "line"(1d)



$\lambda = 25$ $\lambda = 0.62$ $\lambda = 0.41$

too larger lambda, so it is a constant.

Tibshirani, Ryan, Convex Optimization, Carnegie Mellon University. http://www.stat.cmu.edu/~ryantibs/convexopt/

Extension: Adaptive Lasso

Adaptive Lasso

- Simulated data: 1000 variables, 3 of which have true signal; “medium-sized” signal-to-noise ratio

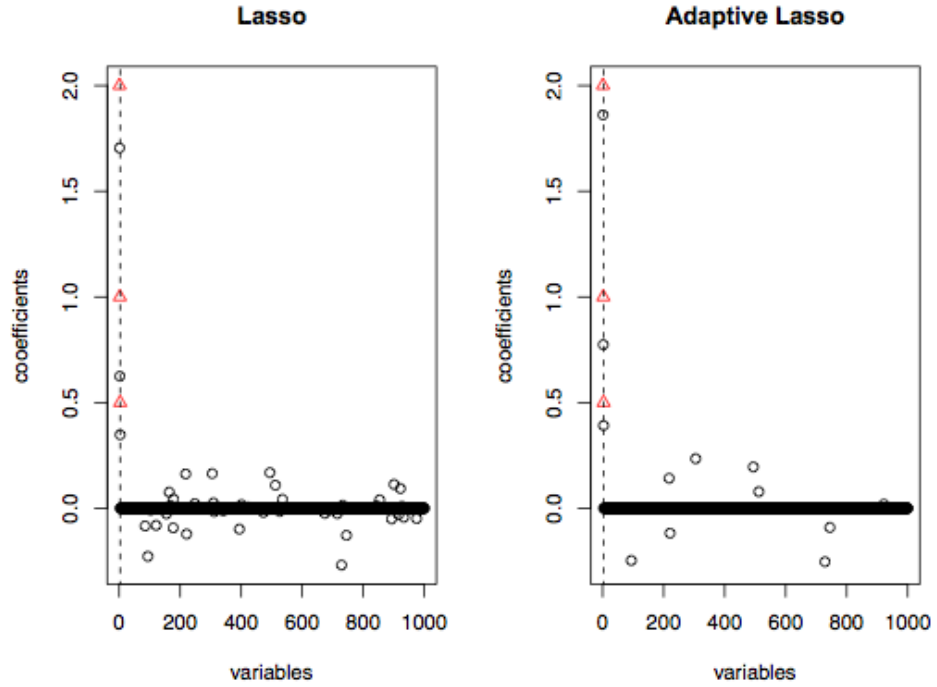


Fig. 2.4 Estimated regression coefficients in the linear model with $p = 1000$ and $n = 50$. Left: Lasso. Right: Adaptive Lasso with Lasso as initial estimator. The 3 true regression coefficients are indicated with triangles. Both methods used with tuning parameters selected from 10-fold cross-validation.

- Both selected the active set, but adaptive Lasso selects only 10 noise variables as opposed to 41 for the Lasso

Bühlmann, P., and van de Geer, S. (2011). *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer, Berlin, Heidelberg.

特徵挑選方法：嵌入法

嵌入法(embedded)– 集成學習

- 由於在包裝法與嵌入法中多數的方法皆以線性迴歸模型為主，特徵僅考慮與應變數間的**線性關係**。為此，我們可考慮用集成學習來建構**交互作用或非線性關係**，以提升特徵挑選的穩健性與重要性。
  - 基於預測結果：排除某個特徵後模型計算準確度降低多少
  - 基於分支結果：用某特徵作為節點時預測或分類誤差的減少
- 雖這兩項指標會受到特徵間的相關性或模型超參數選擇的影響，排序上有所差異，**但基本上不重要的特徵不會被挑選出來**，因此保守的做法可經由多次的以較低標準閥值的過濾法挑選特徵。

特徵挑選方法：投票法與工程驗證

投票法與工程驗證

表 9.10　投票法於特徵挑選

SVID	逐步迴歸（stepwise regression）	套索迴歸（lasso regression）	隨機森林（random forest）	梯度提升機（gradient boosting machine）	投票法（voting）
SVID_003	○	○	○	○	4
SVID_101		○	○	○	3
SVID_021		○	○	○	3
SVID_040	○		○	○	3
SVID_002	○	○		○	3
SVID_128	○		○		2
SVID_062	○	○			2
SVID_077		○		○	2
⋮	⋮				⋮

- 模型特徵挑選之後，應進行工程驗證以確認特徵有物理意義或因果關係
- 工程師藉由領域知識刪除物理意義極弱的特徵或保留因果關係確定的特徵，在數據分析與工程驗證來回數次後（約反覆3-4 輪），特徵子集會趨於收斂，最後保留的重要特徵就同時具備在預測重要性及物理意義。

- 維度縮減法(dimension reduction)
  - 為非監督式學習，此方法未使用與應變數的關聯，主要針對自變數透過參數的線性或非線性組合以建構較少的新變數代表其原始資訊量。

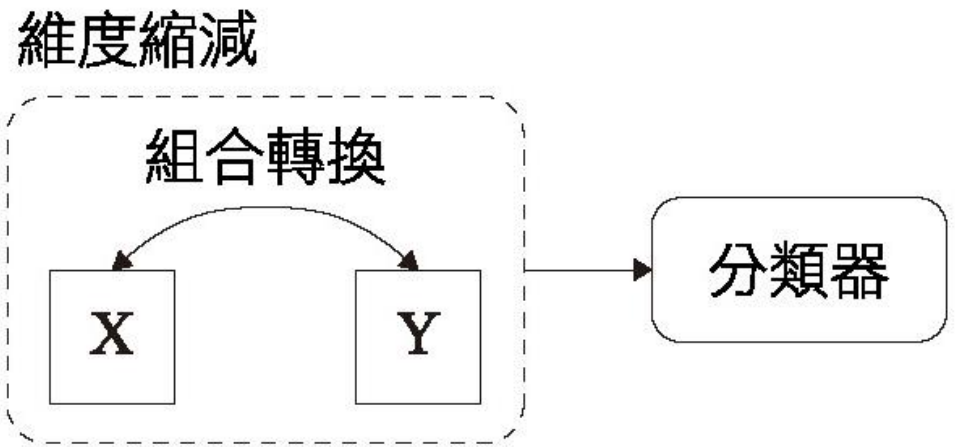


圖 9.1　過濾、包裝、嵌入法與維度縮減的比較

維度縮減：主成分分析

- Idea… (Unsupervised Learning)
  - 4 different kinds of linear combinations

(w1,w2)=(0.5,0.5)

	X1	X2	L1	L2	L3	L4
1	173.00	66.00	169.00	119.50	153.77	164.44
2	155.00	49.00	144.25	102.00	129.50	139.77
3	175.00	72.00	174.66	123.50	159.82	170.25
4	171.00	68.00	169.00	119.50	154.25	164.60
5	166.00	63.00	161.93	114.50	147.28	157.54
6	167.00	64.00	163.34	115.50	148.67	158.96
7	163.00	61.00	158.39	112.00	143.91	154.05
8	155.00	52.00	146.37	103.50	131.95	142.01
9	159.00	55.00	151.32	107.00	136.71	146.91
10	168.00	65.00	164.76	116.50	150.07	160.37
11	166.00	61.00	160.51	113.50	145.65	156.05
12	169.00	73.00	171.12	121.00	157.18	167.00
13	159.00	57.00	152.74	108.00	138.34	148.41
14	154.00	49.00	143.54	101.50	128.92	139.11
15	160.00	60.00	155.56	110.00	141.37	151.31
Mean	164.00	61.00	159.10	112.50	144.49	154.72
Var.	45.57	56.43	98.64	49.32	97.73	**98.96**
S.D.	6.75	7.51	9.93	7.02	9.89	9.95

陳順宇，2005。多變量分析，4版，華泰文化。

維度縮減：主成分分析

主成分分析 (principal component analysis, PCA)

- 主成分指的是原始特徵經由轉換後，形成彼此獨立且依照變異大小排序的新特徵，使用較少數的新特徵來代表大部分原始數據的資訊量
  - $Z_j = \phi_{j1}x_1 + \phi_{j2}x_2 + \cdots + \phi_{jp}x_p,\ \text{for}\ i = 1, \ldots, p$
- 其中第j個**主成分**(principal component, PC)是由所有特徵x_i(可事先標準化以避免特徵因單位不同而各自變異大小有差異)以 ϕ_{ji}為權重的線性加權組合，轉換後的主成分(新特徵)間彼此獨立且能代表所有原始特徵的資訊量。
- 而權重該如何決定？以第一主成分為例，為了使得轉換後的主成分資訊量(變異數)最大，求解的最佳化問題如公式所示。
  - $\Phi_1 = \underset{\phi_{11},\ldots,\phi_{1i}}{\text{argmax}} \left\{ \frac{1}{n} \sum_{k=1}^{n} \left(\sum_{i=1}^{p} \phi_{1i} x_{ik} \right)^2 \right\}$，其中樣本$k = 1, \ldots, n$
- 在自變數所構成的共變異矩陣(或用相關係數矩陣避免自變數間單位差異)下，權重Φ_1可推得為此共變異矩陣的第一個**特徵值**(eigenvalue)所對應的第一**特徵向量**(eigenvector)，其特徵值同時代表該主成分的**變異數**。在**幾何空間**中，藉由特徵向量進行轉換相當於對於原始空間的**轉軸**。

維度縮減：主成分分析

- 主成分分析 (principal component analysis)
  - 左圖為兩個高相關(這裡指線性相關)的原始特徵
  - 主成分分析後，空間中的x軸y軸逆時針的轉軸，以新的第一主成分(PC1)的視角會使得數據所展現的變異最大，也就是各樣本投影到PC1使得其投影點的變異最大。轉軸的角度是往高相關的方向，也就是說在尋找最大變異的同時，等同於在尋找有相似資訊量的高相關特徵，因而在主成分中高相關的特徵間也會有相近且較高的權重。
  - 在轉軸後，主成分之間會相互獨立，這也就是矩陣中的特徵向量彼此**正交**(orthogonal)。在觀念上，可以理解為將**高相關的資訊**轉換到新特徵的**變異**上。

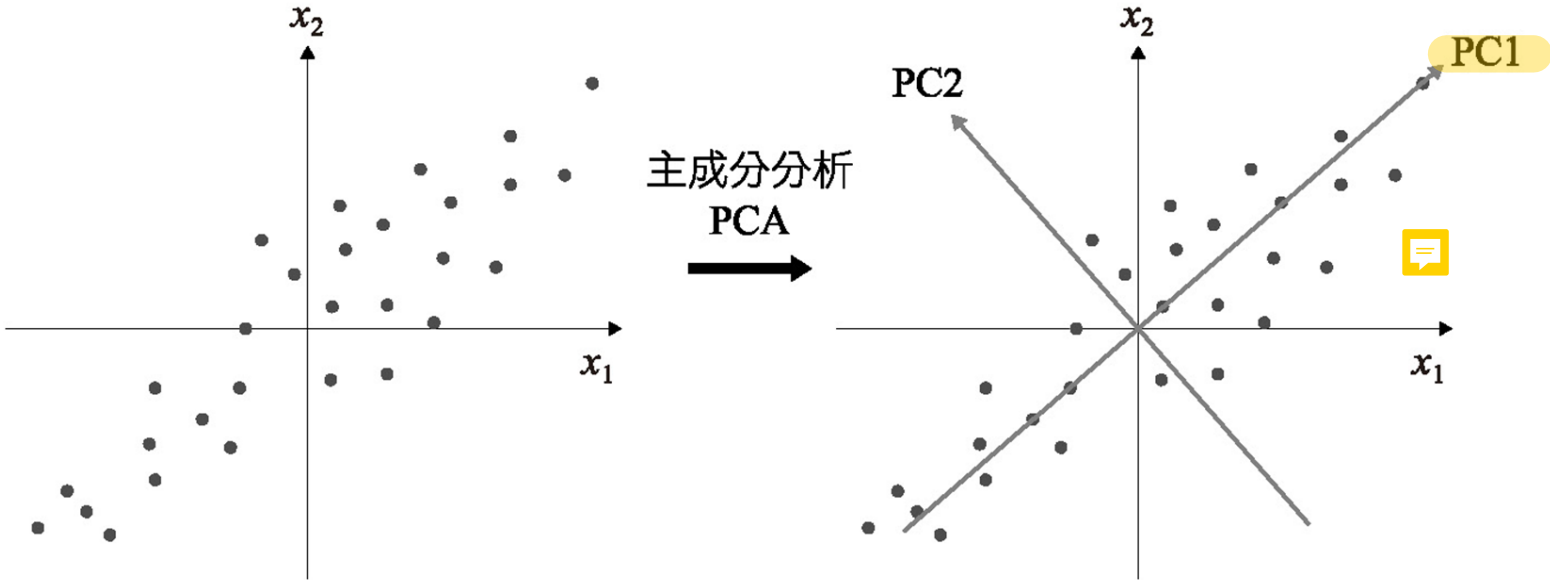


圖 9.8　主成分分析的轉軸

Principal Component Analysis (PCA)

- Three properties of PC
  - Representative
  - Independent
  - Concise
- Purpose
  - Assign a larger weight to important variable and less weight to unimportant variable in order to maximize variance of PC.
- Bivariate case
  - $Var(y) = Var(a_1 x_1 + a_2 x_2)$

$$= a_1{}^2 Var(x_1) + a_2{}^2 Var(x_2) + 2a_1\, a_2\, Cov(x_1, x_2)$$

$$= (a_1\ a_2)\boldsymbol{\Sigma}\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \boldsymbol{a'\Sigma a}$$

Why?

where $(a_1\ a_2)$ is vector of coefficient and

$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1{}^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2{}^2 \end{pmatrix}$ is a covariance matrix of (x_1, x_2)

Eigenvalue and Eigenvector

Eigenvalue and Eigenvector

- Let $\boldsymbol{A}$ be a square matrix, for one scalar λ if there exists a nonzero vector $\boldsymbol{a}$ such that $\boldsymbol{Aa} = \lambda\boldsymbol{a}$. Then, λ is eigenvalue and its corresponding vector $\boldsymbol{a}$ (or $\boldsymbol{e}$) is called eigenvector.

Example

- A matrix *A*

$$A = \begin{pmatrix} .52 & .36 \\ .36 & .73 \end{pmatrix}$$

- We can calculate eigenvalue by using

$$A - \lambda I = \begin{pmatrix} .52 - \lambda & .36 \\ .36 & .73 - \lambda \end{pmatrix}$$

- The eigenvalue is the root of $Det(A - \lambda I) = 0$.

- Example

$$Det(A-\lambda I)=\begin{vmatrix}.52-\lambda & .36\\ .36 & .73-\lambda\end{vmatrix}$$
$$=(.52-\lambda)(.73-\lambda)-.36^2$$
$$=\lambda^2-(.52+.73)\lambda+(.52\cdot.73-.36^2)$$
$$=\lambda^2-tr(A)\lambda+Det(A)$$

$$\begin{cases}\lambda_1+\lambda_2=tr(A)=.52+.73=1.25\\ \lambda_1\cdot\lambda_2=Det(A)=\begin{vmatrix}.52 & .36\\ .36 & .73\end{vmatrix}=.25\end{cases}$$

- Thus, λ_1=1 and λ_2=0.25

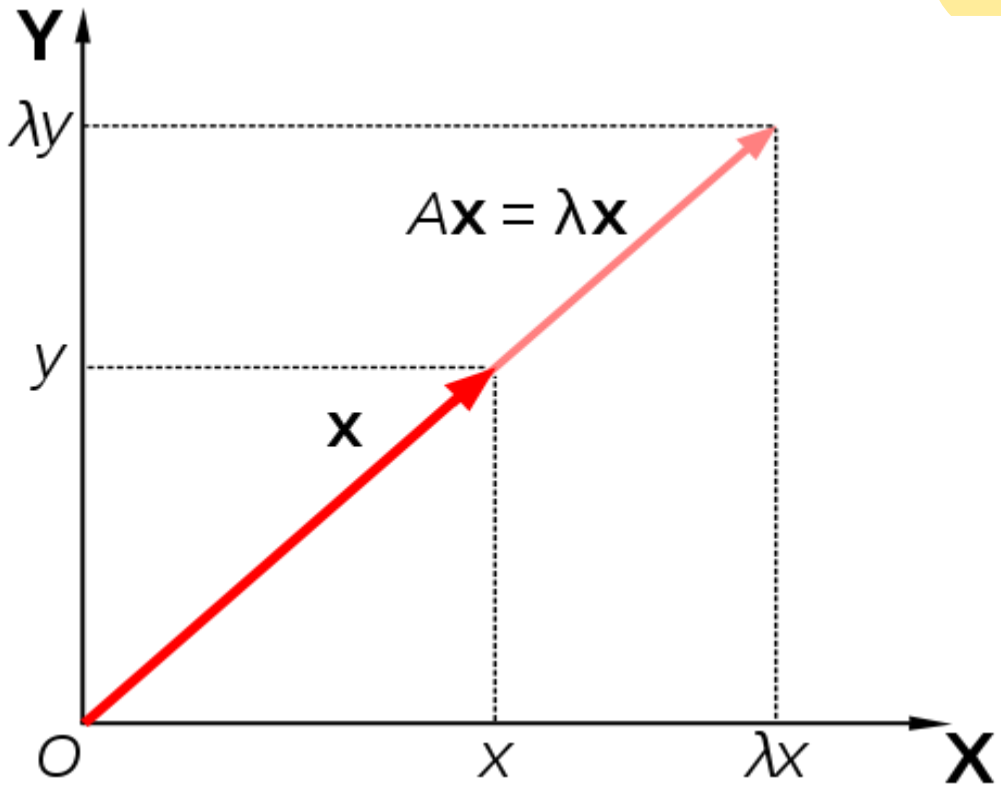
- Note that all eigenvalues are positive, then the matrix A is positive definite matrix.

Eigenvalue and Eigenvector

Geometric Illustration

- Matrix A acts by stretching the vector x, not changing its direction, so x is an eigenvector of A.

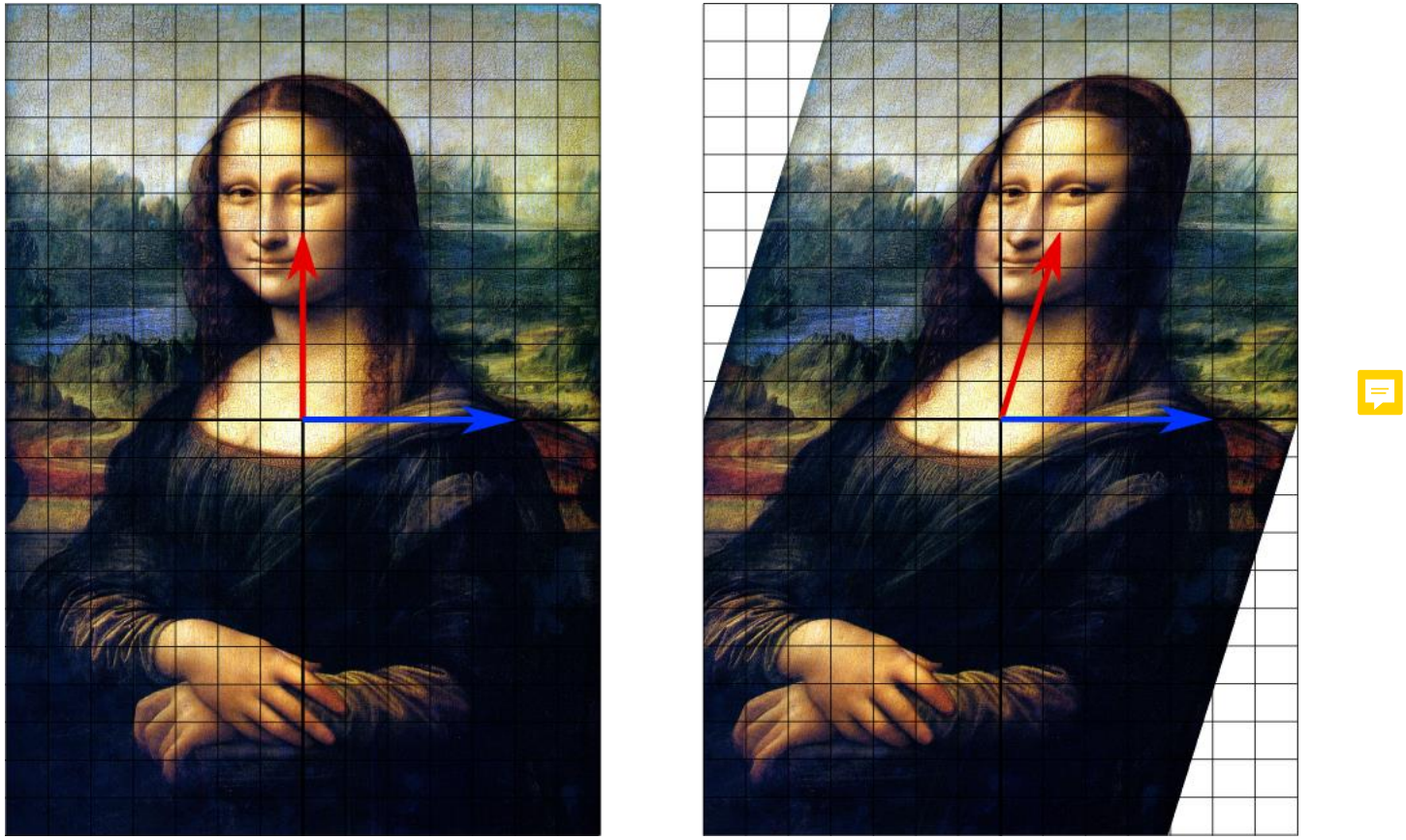
$$\mathbf{A}\boldsymbol{x} = \lambda\boldsymbol{x}$$



Wikipedia (2012)

Eigenvalue and Eigenvector

Geometric Illustration



Wikipedia (2012)

- In this shear mapping the red arrow changes direction but the blue arrow does not. The blue arrow is an eigenvector, and since its length is unchanged its eigenvalue is 1.

Principal Component Analysis (PCA)

Theorem

- Given a linear combination $y = a_1x_1 + a_2x_2 + \cdots + a_px_p$ and $\boldsymbol{a'a} = 1$, the solution of maximizing $Var(y) = \boldsymbol{a'\Sigma a}$ is the eigenvector corresponding to the maximal eigenvalue λ_1 of matrix $\boldsymbol{\Sigma}$, where $\boldsymbol{\Sigma}$ is a covariance matrix generated by random vector $x = \begin{pmatrix} x_1 \\ \vdots \\ x_p \end{pmatrix}$.

- Proof:

  Use Lagrange method, let $g(\boldsymbol{a}) = \boldsymbol{a'\Sigma a} - \lambda(\boldsymbol{a'a} - 1)$, where λ is a Lagrange multiplier. To maximize $g(\boldsymbol{a})$, we take derivative

  $g'(\boldsymbol{a}) = 2\boldsymbol{\Sigma a} - \boldsymbol{2}\lambda\boldsymbol{a} = 0$,

  i.e., $\boldsymbol{\Sigma a} = \lambda\boldsymbol{a}$

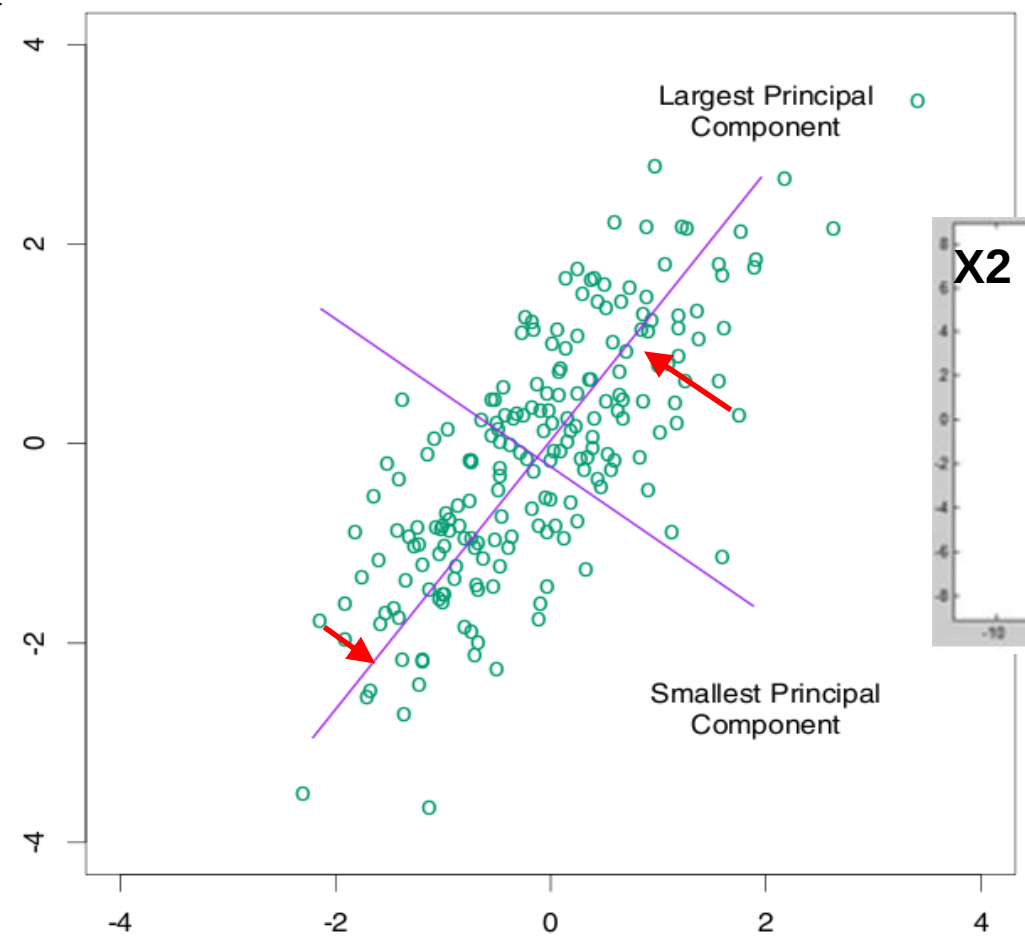
  Thus, $\boldsymbol{a}$ is eigenvalue corresponding to eigenvalue λ of matrix $\boldsymbol{\Sigma}$.

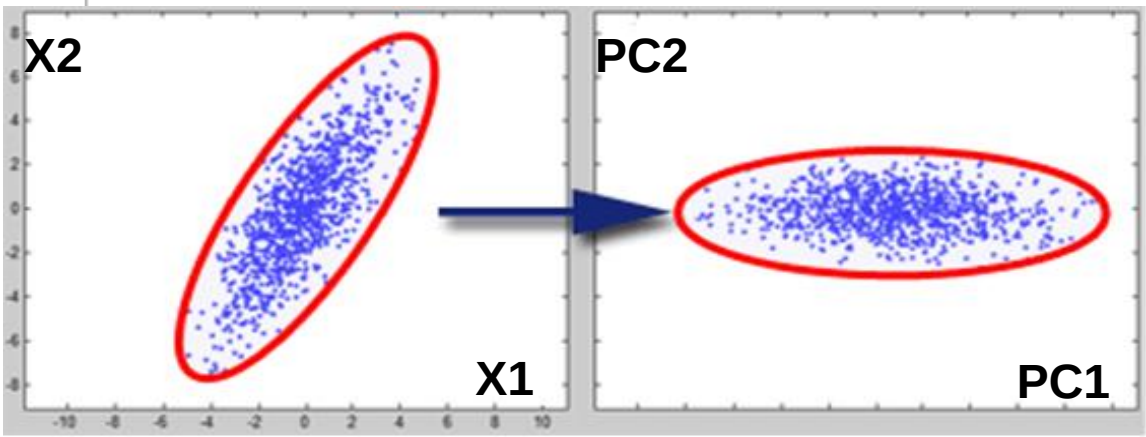
- Note: $\boldsymbol{\Sigma}$ is unknown generally, we usually use sample covariance $\mathbf{S}$.

PCA Illustration

Orthogonal Property

- Eigenvectors are orthogonal. Eg. $e_1 \perp e_2$
- PCs are orthogonal. Eg. $y_1 \perp y_2$ and the correlation coefficient is 0.





Suppose that $\mathbf{\Sigma}$ is the covariance matrix for the set of variables $\mathbf{x}$, and it has the eigenvalue/eigenvector pairs $\{\lambda_i, \mathbf{e}_i\}_{i=1}^{p}$, then:

1. The i^{th} PC is $y_i = \mathbf{e}_i^T \mathbf{x}$;
2. $\text{var}(y_i) = \text{var}(\mathbf{e}_i^T \mathbf{x}) = \mathbf{e}_i^T \mathbf{\Sigma} \mathbf{e}_i = \lambda_i$, this is the variance of the i^{th} PC;
3. $\text{cov}(y_i, y_k) = \mathbf{e}_i^T \mathbf{\Sigma} \mathbf{e}_k = 0$ for any $i \neq k$, i.e., y_i, y_k are uncorrelated.

it is easy to have the following

(1) $\mathbf{\Sigma}_y = \text{cov}(\mathbf{y}) = \begin{pmatrix} \lambda_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \lambda_p \end{pmatrix}$;

(2) $\text{tr}(\mathbf{\Sigma}_y) = \text{tr}(\mathbf{E}^T \mathbf{\Sigma}_x \mathbf{E}) = \text{tr}(\mathbf{\Sigma}_x \mathbf{E} \mathbf{E}^T) = \text{tr}(\mathbf{\Sigma}_x)$;

(3) Proportion of variances in the data due to the i^{th} PC:
$\frac{\text{var}(y_i)}{\sum_{i=1}^{p} \text{var}(y_i)} = \frac{\lambda_i}{\sum_{i=1}^{p} \lambda_i}$.

Example

- Example

Suppose $\mathbf{x} \sim N_2(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$ where

$$\boldsymbol{\mu}_x = \begin{pmatrix} 2 \\ 1 \end{pmatrix} \text{ and } \boldsymbol{\Sigma}_x = \begin{pmatrix} 0.25 & 0.45 \\ 0.45 & 1 \end{pmatrix}.$$

- Find the eigenvalues/eigenvectors of Σ_x
  - Use MATLAB function eig(.) but notice that the MATLAB function arrange the eigenvalues in ascending order.

$$\mathbf{E} = (\mathbf{e}_1 \ \mathbf{e}_2) = \begin{pmatrix} 0.42 & -0.91 \\ 0.91 & 0.42 \end{pmatrix} \text{ and}$$

$$\boldsymbol{\Lambda} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} = \begin{pmatrix} 1.21 & 0 \\ 0 & 0.039 \end{pmatrix}$$

Example

$y_1 = \mathbf{e}_1^T \mathbf{x}$ and $y_2 = \mathbf{e}_2^T \mathbf{x}$. But in order to make y_1, y_2 centered, we can include μ_1, μ_2 as

$$y_1 = \mathbf{e}_1^T(\mathbf{x} - \boldsymbol{\mu}_x) = 0.42(x_1 - \mu_1) + 0.91(x_2 - \mu_2)$$
$$y_2 = \mathbf{e}_2^T(\mathbf{x} - \boldsymbol{\mu}_x) = -0.91(x_1 - \mu_1) + 0.42(x_2 - \mu_2)$$

When observing $\mathbf{x} = \begin{pmatrix} 2.6 \\ 2.8 \end{pmatrix}$, its PCs are

$$y_1 = 0.42(2.6 - 2) + 0.91(2.8 - 1) = 1.89 \text{ and}$$
$$y_2 = -0.91(2.6 - 2) + 0.42(2.8 - 1) = 0.21$$

- Also, $\text{var}(y_1) = \lambda_1 = 1.21$ and $\text{var}(y_2) = \lambda_2 = 0.039$ (this is so regardless whether or not we include μ_x)
- $\frac{\text{var due to the } 1^{st} \text{ PC}}{\text{total variance}} = \frac{\lambda_1}{\lambda_1 + \lambda_2} = \frac{1.21}{1.25} = 96.8\%$

Proportion of Variance

- Proportion of Variance (主成分解釋變異比例)
  - $Var(y_i) = \lambda_i$
  - Sum of variance of PCs is the same as sum of variance of original variables

$$\sum_{i=1}^{p} Var(y_i) = \sum_{i=1}^{p} Var(x_i)$$

  - Proportion of variance of 1st PC y_1

$$\frac{Var(y_1)}{\sum_{i=1}^{p} Var(y_i)} = \frac{Var(y_1)}{\sum_{i=1}^{p} Var(x_i)} = \frac{\lambda_1}{\sum_{i=1}^{p} \lambda_i}$$

Loading of PC (主成分負荷)

- Loading of PC is correlation coefficient of the PC y_j and the variable x_i.

$$r_{y_j x_i} = \frac{a_{ji}\sqrt{\lambda_j}}{s_i}$$

where s_i is the standard deviation of x_i.

Communality (共通性, ${h_i}^2$)

- Definition: Proportion of variance of x_i can be explained by PCs
- Example: The proportion of variance of x_1 can be explained by 1st PC and 2nd PC is …
- ${h_1}^2 = {r_{y_1 x_1}}^2 + {r_{y_2 x_1}}^2$
- Factor Analysis

Remarks of Communality

- More number of PC be selected, the higher the communality ${h_i}^2$.
- When all PCs are selected, then ${h_i}^2 = 1$.
- Because of $\boldsymbol{a}'\boldsymbol{a} = 1$, i.e., $\sum_{i=1}^{p} {a_{ij}}^2 = 1$, then $|a_{ij}| \leq 1$.
- When only select 1st PC y_1, sum of communality of all variable x_i is

$$\sum_{i=1}^{p} {h_i}^2 = \lambda_1$$

- Similarly, if selecting y_1 & y_2, sum of communality of all variable x_i is

$$\sum_{i=1}^{p} {h_i}^2 = \lambda_1 + \lambda_2$$

PC needs to be defined and further interpreted.

- Eg. $y_1 = 0.9 \times height(x_1) + 0.8 \times weight(x_2) + 0.01 \times income(x_3)$
- We may call the PC y_1 as "Body shape" factor

Example

- Data
  - Height (X1) and Weight (X2)

	X1	X2
1	173	66
2	155	49
3	175	72
4	171	68
5	166	63
6	167	64
7	163	61
8	155	52
9	159	55
10	168	65
11	166	61
12	169	73
13	159	57
14	154	49
15	160	60

How to find a linear combination y1 of X1 and X2 to represent the shape of human body?

Example

Mean, Variance, and Standard Deviation

- 4 different kinds of linear combinations

	X1	X2	L1	L2	L3	L4
1	173.00	66.00	169.00	119.50	153.77	164.44
2	155.00	49.00	144.25	102.00	129.50	139.77
3	175.00	72.00	174.66	123.50	159.82	170.25
4	171.00	68.00	169.00	119.50	154.25	164.60
5	166.00	63.00	161.93	114.50	147.28	157.54
6	167.00	64.00	163.34	115.50	148.67	158.96
7	163.00	61.00	158.39	112.00	143.91	154.05
8	155.00	52.00	146.37	103.50	131.95	142.01
9	159.00	55.00	151.32	107.00	136.71	146.91
10	168.00	65.00	164.76	116.50	150.07	160.37
11	166.00	61.00	160.51	113.50	145.65	156.05
12	169.00	73.00	171.12	121.00	157.18	167.00
13	159.00	57.00	152.74	108.00	138.34	148.41
14	154.00	49.00	143.54	101.50	128.92	139.11
15	160.00	60.00	155.56	110.00	141.37	151.31
Mean	164.00	61.00	159.10	112.50	144.49	154.72
Var.	45.57	56.43	98.64	49.32	97.73	**98.96**
S.D.	6.75	7.51	9.93	7.02	9.89	9.95

Example

- How to find the optimal L4?
  - Under the $a_1^2 + a_2^2 = 1$, we find the a_1 and a_2 such that maximizing Var(y_1), where $y_1 = a_1 x_1 + a_2 x_2$
  - Var(y_1) = $a'\Sigma a$, where a is the eigenvector with respect to the maximal eigenvalue.
  - In practice, Σ is unknown, we usually use sample covariance matrix S instead of Σ.
- Calculation

$$S = \begin{pmatrix} 45.5714 & 47.6429 \\ 47.6429 & 56.4286 \end{pmatrix}$$

$$\det(S - \lambda I) = \begin{vmatrix} 45.5714 - \lambda & 47.6429 \\ 47.6429 & 56.4286 - \lambda \end{vmatrix}$$

$$\lambda_1 = 98.9511$$
$$\lambda_2 = 3.0489$$

Example

- Eigenvector and PC

$$S\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \lambda_1 \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \Rightarrow \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} 0.6659 \\ 0.7461 \end{pmatrix}$$

$$y_1 = 0.6659x_1 + 0.7461x_2$$

$$S\begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \lambda_2 \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \Rightarrow \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} 0.7461 \\ -0.6659 \end{pmatrix}$$

$$y_2 = 0.7461x_1 - 0.6659x_2$$

- Orthogonal

$$a'b = 0$$

Example

- Proportion of Variance
  - $\frac{\lambda_1}{\lambda_1+\lambda_2} = \frac{98.9511}{102} = 0.97$
  - $\frac{\lambda_1}{\lambda_1+\lambda_2} = \frac{3.0489}{102} = 0.03$
- Loading of PC (correlation coefficient)

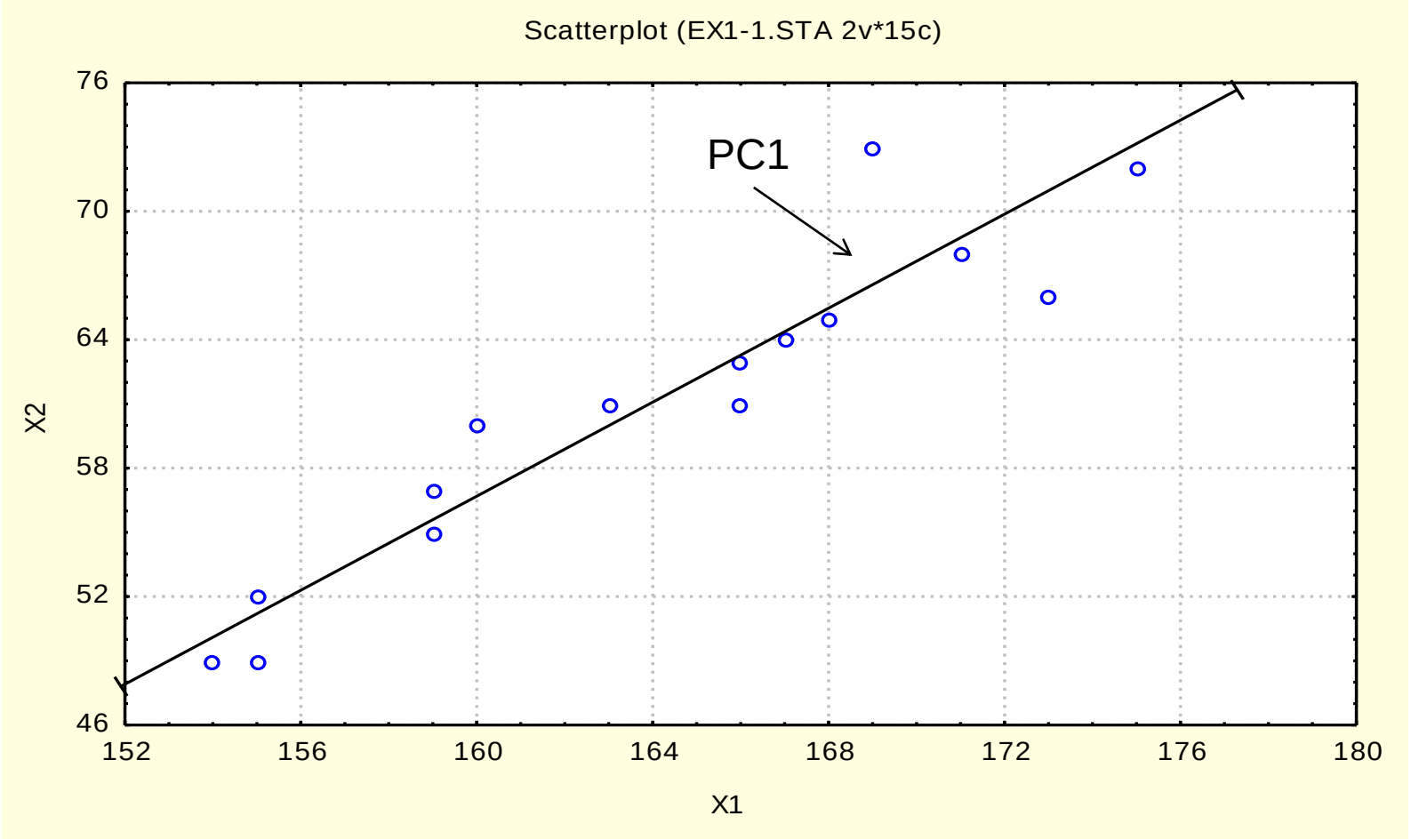
$$r_{y_1x_1} = \frac{a_{11}\sqrt{\lambda_1}}{s_1} = \frac{0.6659 \times \sqrt{98.9511}}{6.7507} = 0.9812$$

$$r_{y_1x_2} = \frac{a_{12}\sqrt{\lambda_1}}{s_2} = \frac{0.7461 \times \sqrt{98.9511}}{7.5119} = 0.9880$$

- Communality
  - PC1 for x_1: $h_1 = (0.9812)^2 = 0.96275$
  - PC1 for x_2: $h_2 = (0.9880)^2 = 0.97614$
  - PC1 and PC2 for x_1: $(r_{y_1x_1})^2 + (r_{y_2x_1})^2$

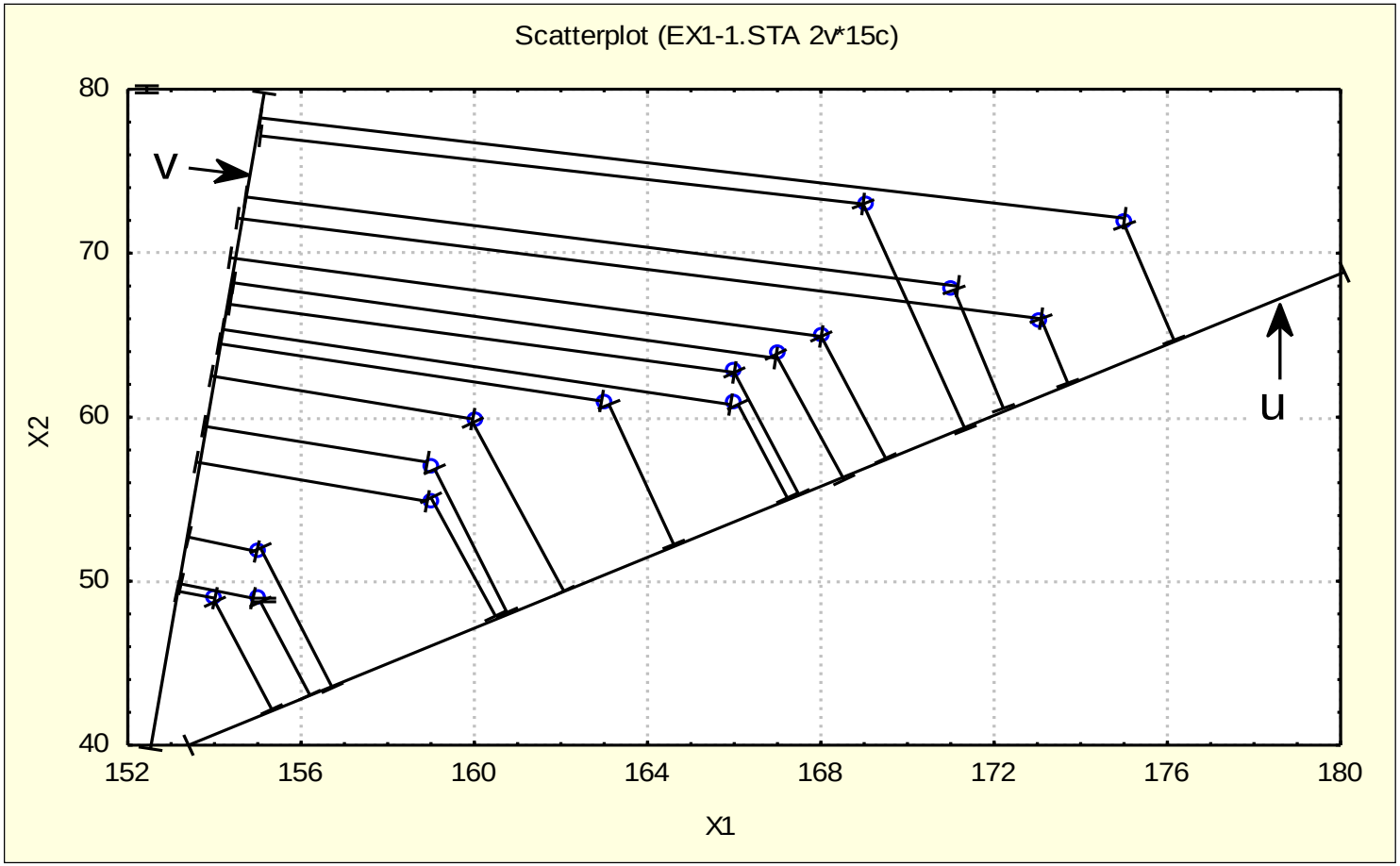
Example

- Data
  - Height and Weight



Principal Component Analysis (PCA)

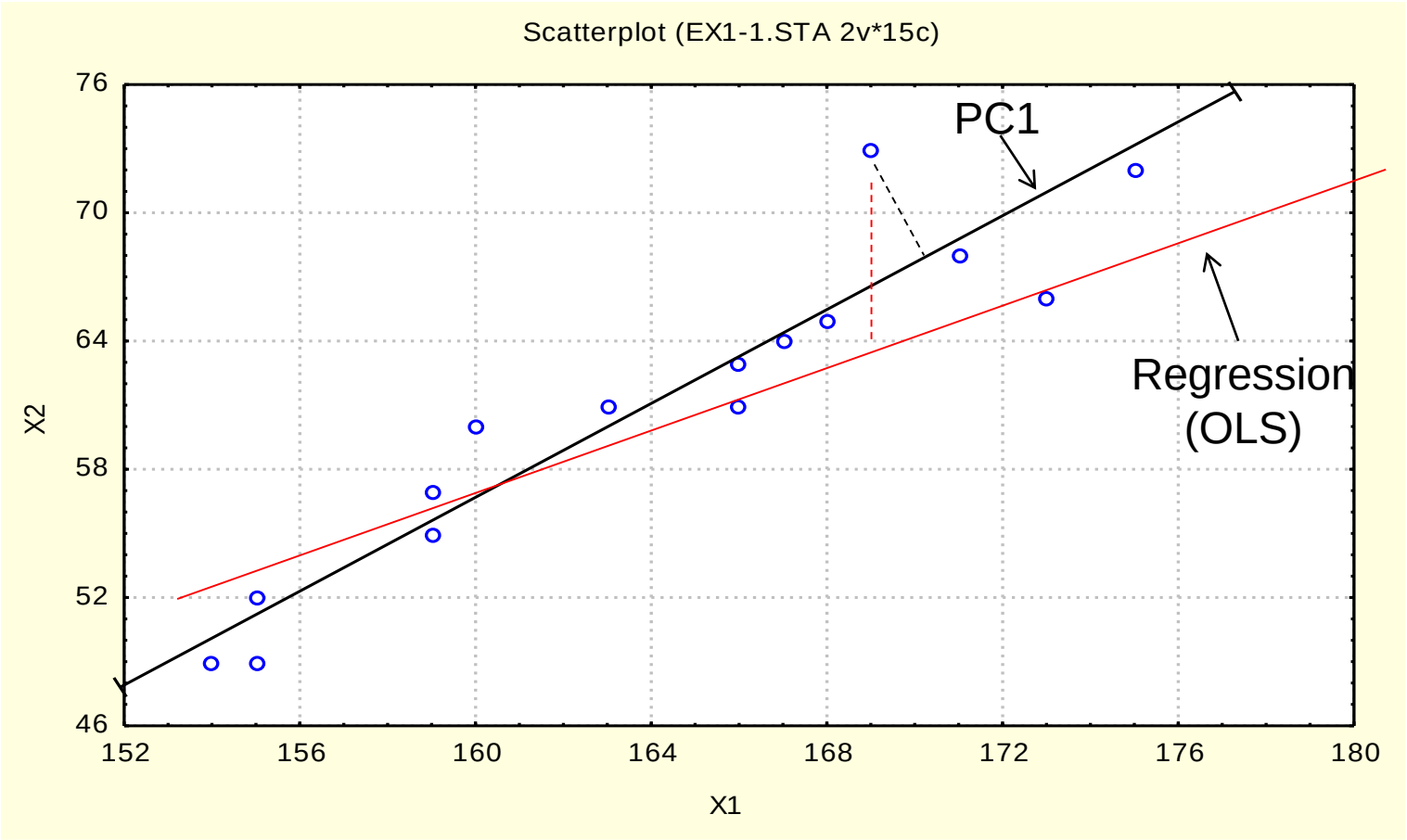
- Analog
  - Employ linear combination to maximize variance
  - Kind of: selecting a "angle" of taking picture for a group and distinguish each person clearly



PCA vs. OLS

PCA vs. Linear Regression (OLS)

- PCA: maximize variance (unsupervised)
  - Data project to PC axis
- OLS: minimize error (or residual) (supervised)



主成分分析

- 在主成分分析中，通常會依照主成分的特徵值(變異數)大小決定要保留多少個主成分，通常建議方式如下
  - 特徵值陡坡圖：將特徵值的變異數由高到低排序
  - 累積變異圖：將特徵值的變異數由高到低排序計算
  - 凱薩原則：挑選所有特徵值大於 1 的主成分（Kaiser, 1960）

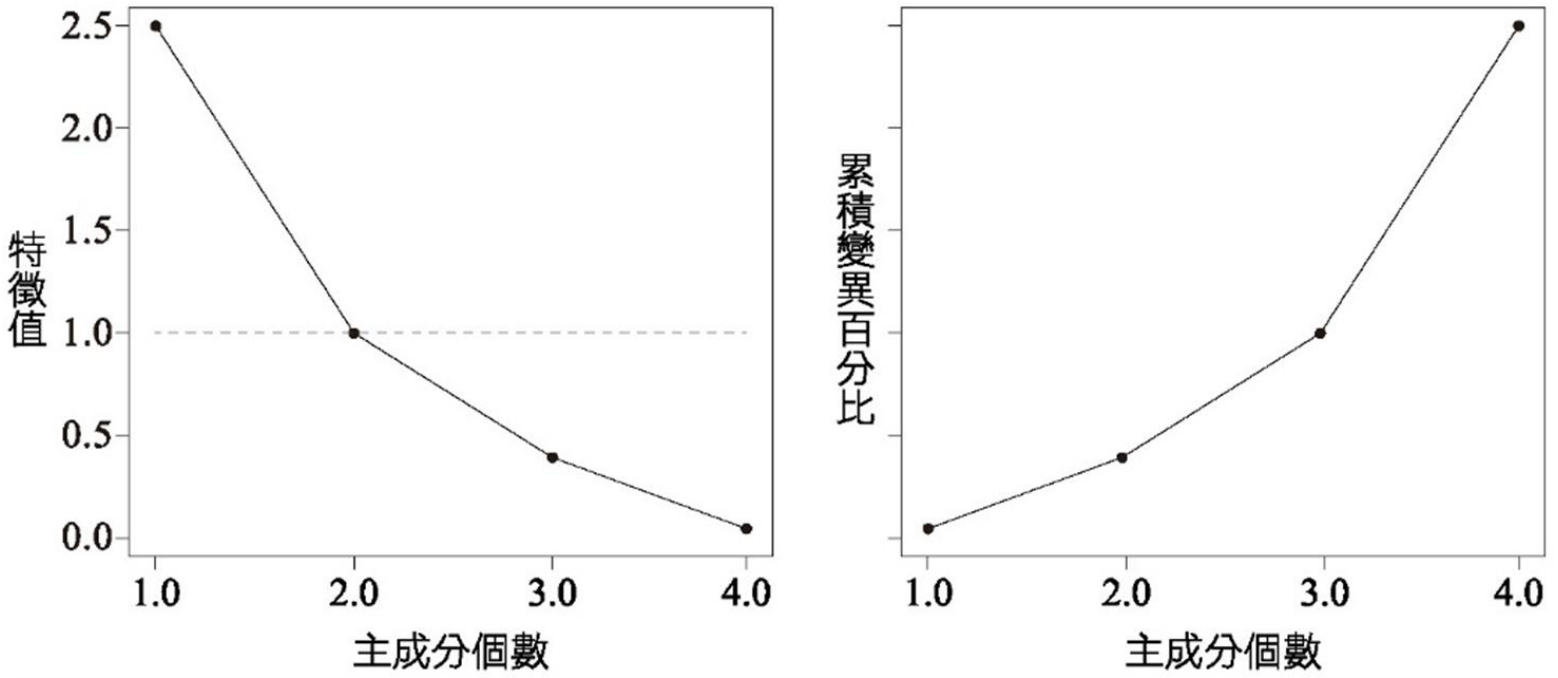
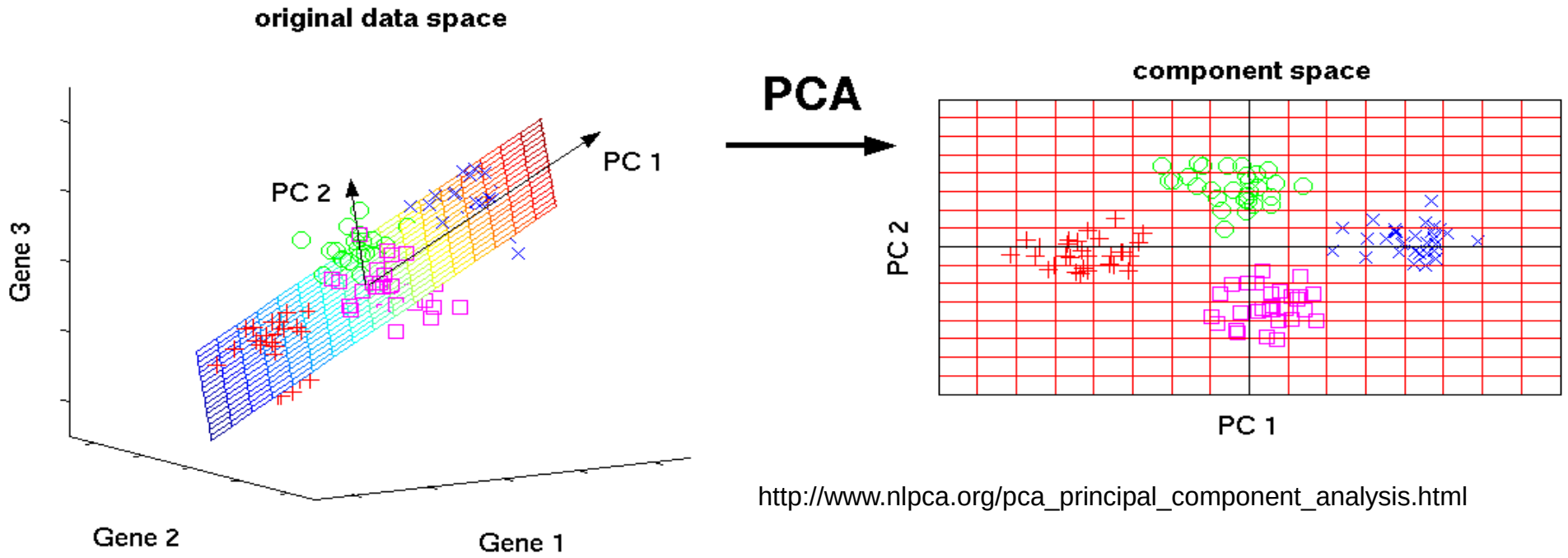


圖 9.10　特徵值陡坡圖與累積變異圖

PCA and Bioinformatics

- PCA is a standard technique for visualizing high dimensional data and for data pre-processing. PCA reduces the dimensionality (the number of variables) of a data set by maximizing variance.



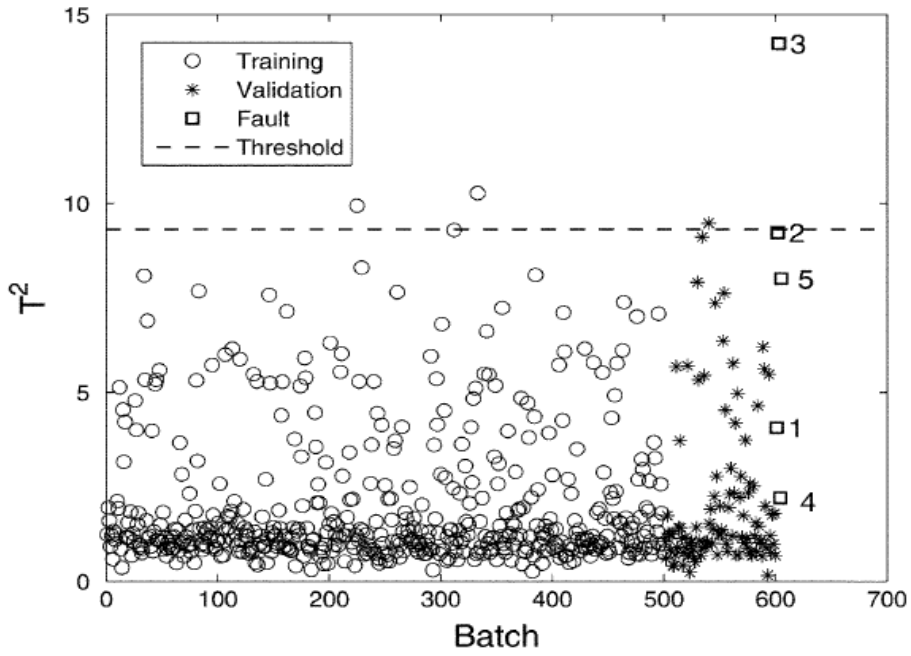
http://www.nlpca.org/pca_principal_component_analysis.html

- Illustrated are three-dimensional gene expression data which are mainly located within a two-dimensional subspace.
- Such two-dimensional visualization of the samples allow us to draw qualitative conclusions about the separability of experimental conditions (marked by different colors).

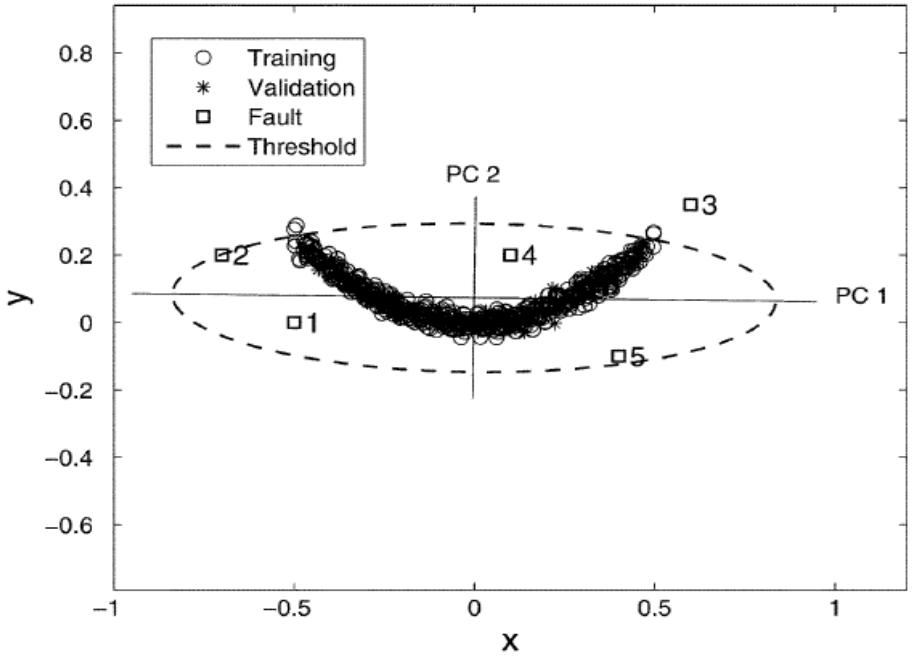
Applications

Fault Detection

- The Hotelling's T^2 is a measure of the variation in principal component space.



(a)



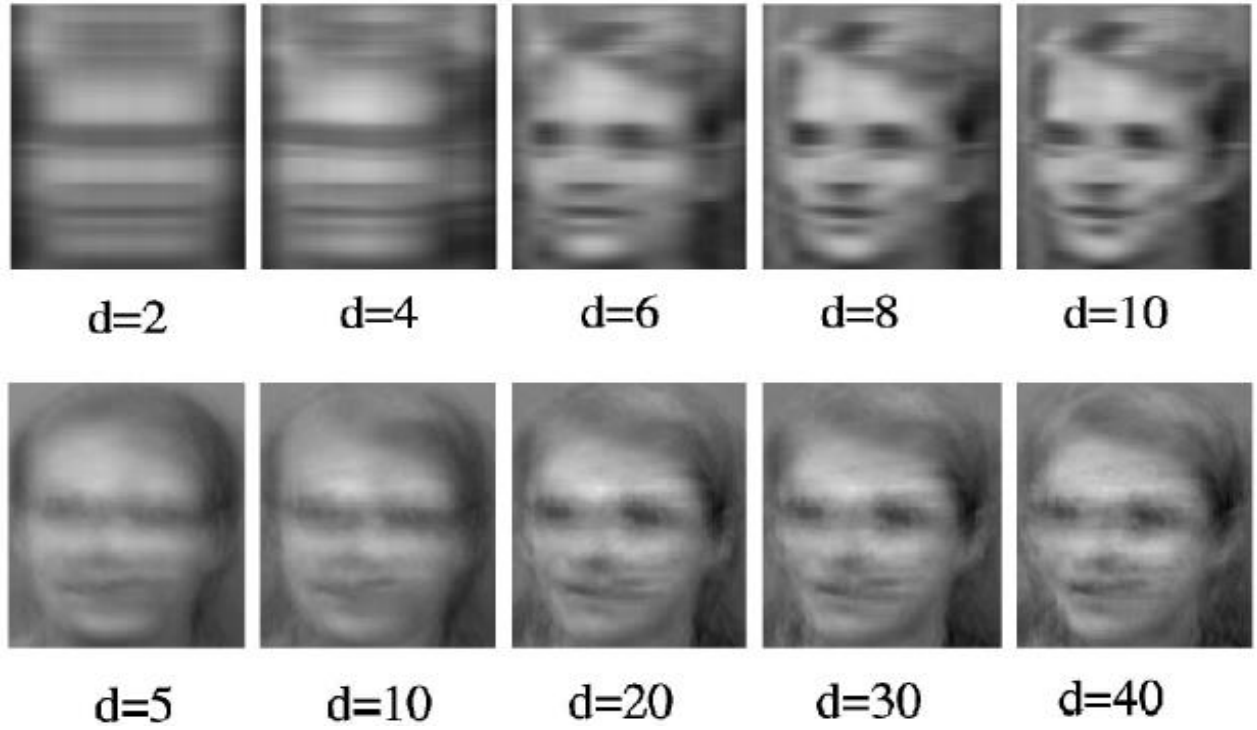
(b)

- Fault detection results based on two PCs: (a) T^2 chart and (b) PC's (solid lines) and T threshold (dashed ellipse) in original variable space.

He, Q. J. and J. Wang, 2007, Fault Detection Using the k-Nearest Neighbor Rule for Semiconductor Manufacturing Processes, IEEE Transactions on Semiconductor Manufacturing, 20 (4), 345-354.

Face Representation and Recognition

- Two-dimensional PCA (2DPCA) is based on 2D image matrices rather than 1D vectors so the image matrix does not need to be transformed into a vector prior to feature extraction.



Some reconstructed images based on 2DPCA (upper) and PCA (lower)

Yang, J., D. Zhang, A. F. Frangi, and J.-Y. Yang, 2004, Two-dimentional PCA: A new approach to appearance-based face representation and recognition IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(1), 131-137.

PCA Concluding Remarks

Concluding Remarks

- Generally, using correlation matrix for PCA due the unit elimination.
- No dimension transformation (or pivot transformation) again after PCA
- The PCs are selected by Cattell (1966) Scree test or the Kaiser (1960)'s eigenvalue-greater-than-one rule.
- In empirical study, it is a satisfied result if selecting 5 to 6 PCs achieving 70% variance interpretation.
- Each PCs shows maximal variance and are mutually orthogonal.

Principal Component Regression (PCR)

- Let $Z_1, Z_2, \ldots, Z_M$ represent $M < p$ *linear combinations* of our original p predictors. That is,

$$Z_m = \sum_{j=1}^{p} \phi_{mj} X_j \tag{1}$$

for some constants $\phi_{m1}, \ldots, \phi_{mp}$.

- We can then fit the linear regression model,

$$y_i = \theta_0 + \sum_{m=1}^{M} \theta_m z_{im} + \epsilon_i, \quad i = 1, \ldots, n, \tag{2}$$

using ordinary least squares.

- Note that in model (2), the regression coefficients are given by $\theta_0, \theta_1, \ldots, \theta_M$. If the constants $\phi_{m1}, \ldots, \phi_{mp}$ are chosen wisely, then such dimension reduction approaches can often outperform OLS regression.

Principal Component Regression (PCR)

Principal Component Regression (PCR)

- PCR identifies linear combinations, or *directions*, that best represent the predictors $X_1, \ldots, X_p$.
- These directions are identified in an *unsupervised* way, since the response Y is not used to help determine the principal component directions.
- That is, the response does not *supervise* the identification of the principal components.
- Consequently, PCR suffers from a potentially serious drawback: there is no guarantee that the directions that best explain the predictors will also be the best directions to use for predicting the response.

Partial Least Squares (PLS)

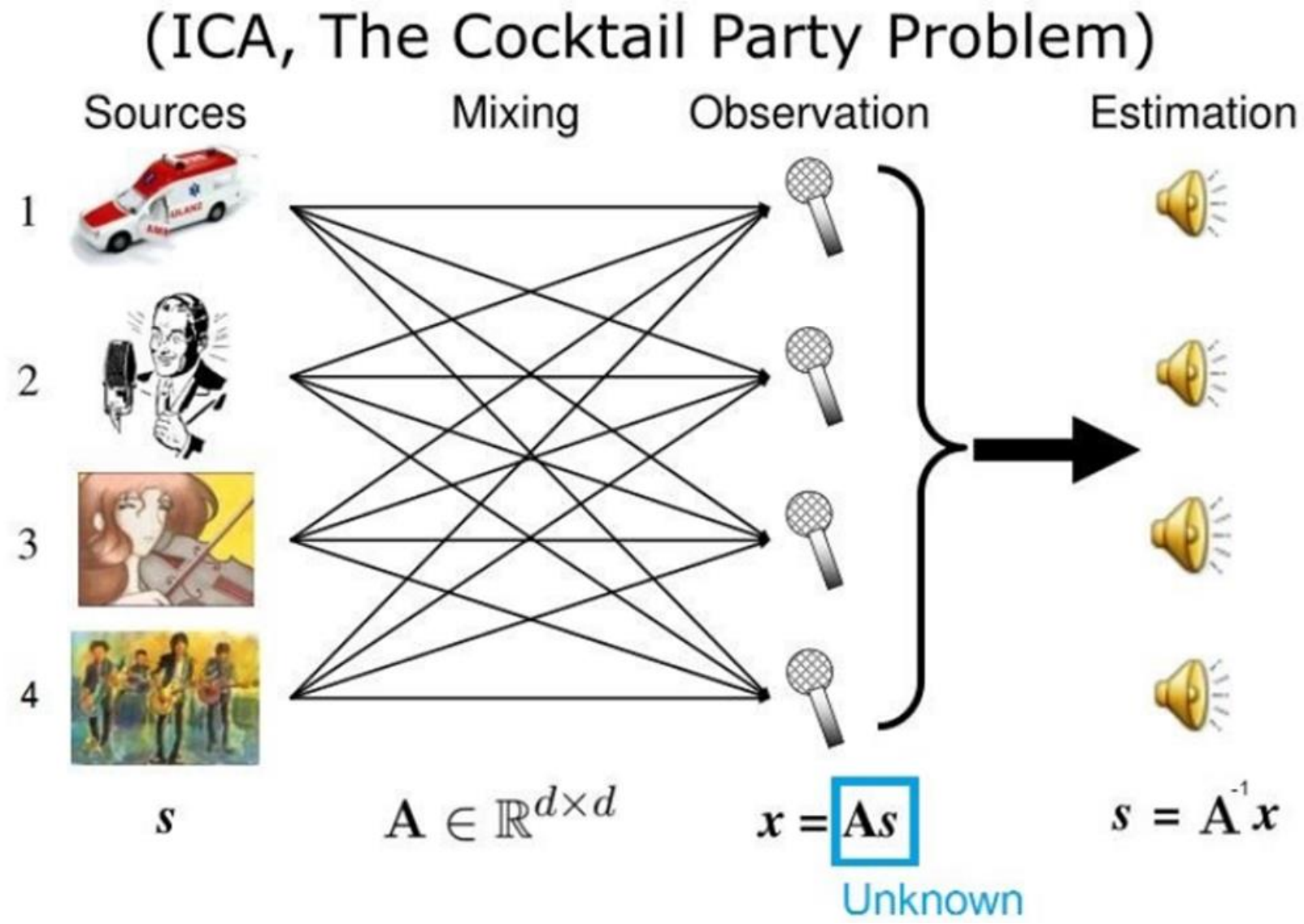
- Like PCR, PLS is a dimension reduction method, which first identifies a new set of features $Z_1, \ldots, Z_M$ that are linear combinations of the original features, and then fits a linear model via OLS using these M new features.
- But unlike PCR, PLS identifies these new features in a supervised way – that is, it makes use of the response Y in order to identify new features that not only approximate the old features well, but also that *are related to the response*.
- Roughly speaking, the PLS approach attempts to find directions that help explain both the response and the predictors.

Partial Least Squares (PLS)

- After standardizing the p predictors, PLS computes the first direction Z_1 by setting each ϕ_{1j} in (1) equal to the coefficient from the simple linear regression of Y onto X_j.
- One can show that this coefficient is proportional to the correlation between Y and X_j.
- Hence, in computing $Z_1 = \sum_{j=1}^{p} \phi_{1j} X_j$, PLS places the highest weight on the variables that are most strongly related to the response.
- Subsequent directions are found by taking residuals and then repeating the above prescription.

維度縮減：獨立成分分析

- Independent Component Analysis (ICA) 獨立成分分析



http://rpubs.com/skydome20/R-Note17-ICA

維度縮減：獨立成分分析

獨立成分分析(independent component analysis, ICA)

- 是將特徵（常見為多個混雜的信號）分離成獨立且非高斯特徵的線性組合，屬於盲信號分離（blindsource separation）中的一個技巧，同時達到維度縮減的效果。混合信號（X）與原始信號（S）的關係如下

  - $$X_{m\times t}=\begin{bmatrix} | & | & | \\ x^{(1)} & \dots & x^{(t)} \\ | & | & | \end{bmatrix}=A_{m\times n}S_{n\times t}=\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix}\begin{bmatrix} | & | & | \\ s^{(1)} & \dots & s^{(t)} \\ | & | & | \end{bmatrix}$$

- 獨立成分分析可協助我們找出潛在真正影響機器運作的獨立訊號。
- 在權重的計算上，相對於主成分分析是將變異最大化，獨立成分分析則可透過(1)最大化非高斯的特性；(2)最小化相互資訊（mutual information）兩種目標進行信號分離，來得到權重$A_{m\times n}$。例如最大化非高斯特性的其中一種分法是最大化原始信號的峰態係數（kurtosis）如公式所示。

  - $$A=\underset{A}{\operatorname{argmax}}\left\{\frac{E[(s-\bar{s})^4]}{(E[(s-\bar{s})^2])^2}-3\right\}$$

維度縮減：獨立成分分析

PCA vs. ICA

- 此二方法不同的點在於主成分分析是試著找出數據**最大變異**的有代表性成分，其權重(特徵向量)彼此正交；而獨立成分分析則是試著找出最大化非高斯特性使得**彼此獨立**的共同成分，其權重非彼此正交。
- 此外數據轉換方向是相反的，主成分分析有將**多個原有特徵進行合併與命名**；而獨立成分分析則是將**多個收集特徵進行分離找出共同的因子**。

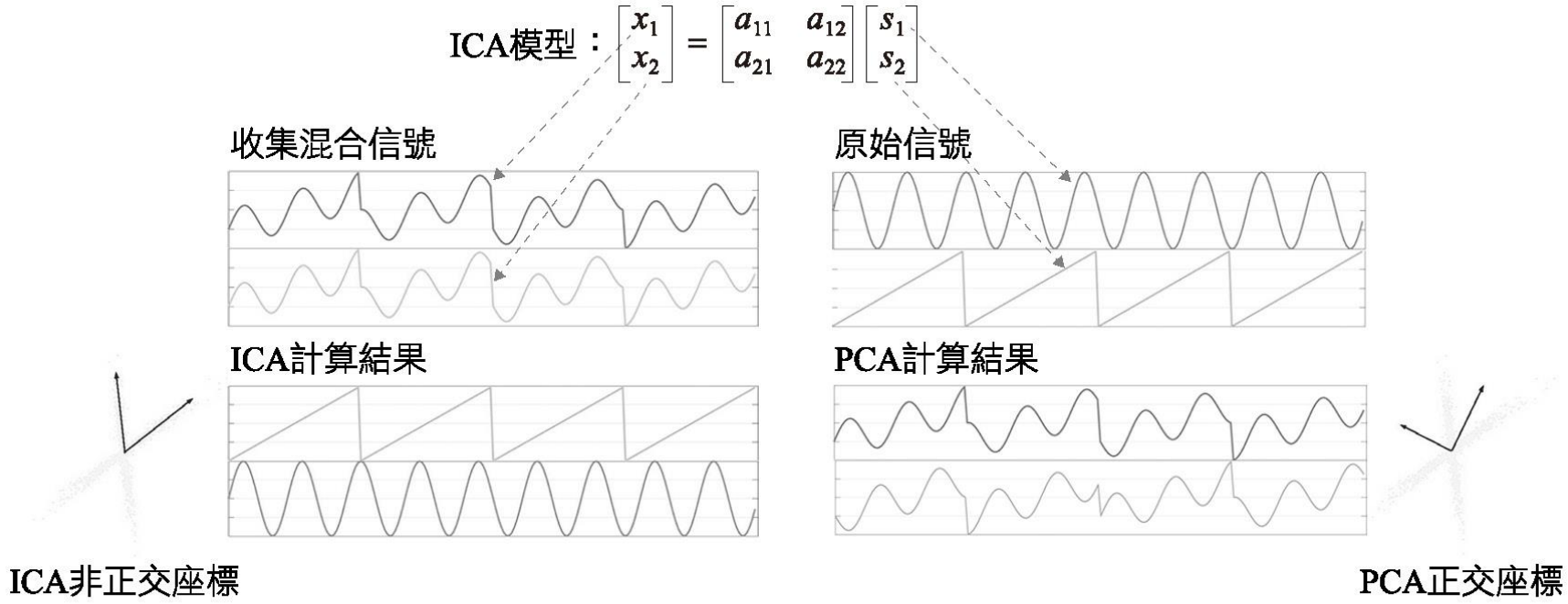
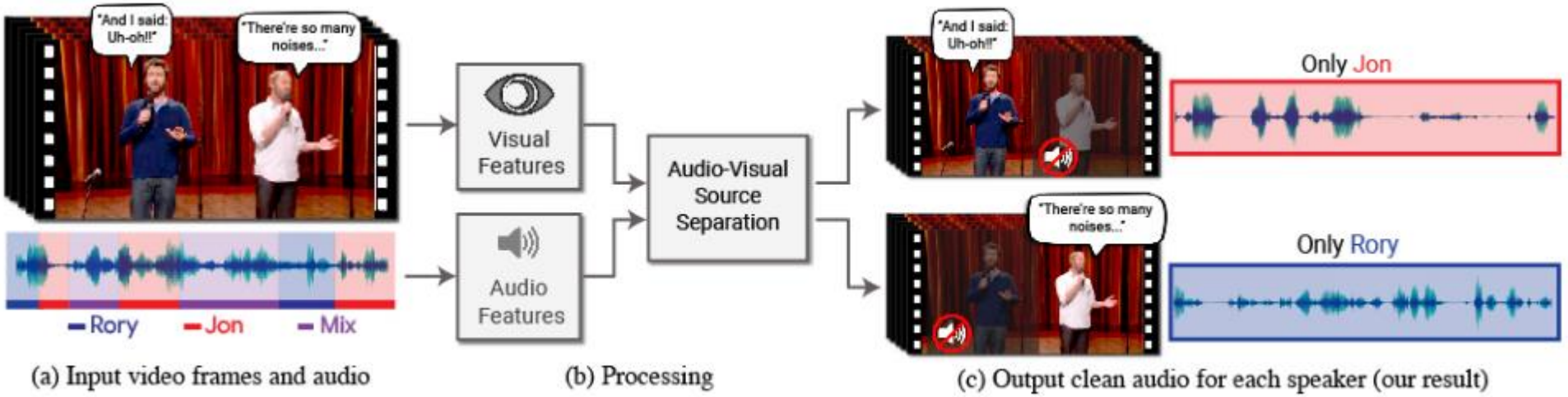


圖 9.12　主成分分析與獨立成分分析

Synthetic Cocktail Parties (Video Time)

https://www.youtube.com/watch?v=NzZDnRni-8A&feature=youtu.be



https://looking-to-listen.github.io/


Input video (two people speaking together)
JOHN
ALL
RORY
Video source: Team Coco, https://www.youtube.com/watch?v=UT7h4nRcWjU

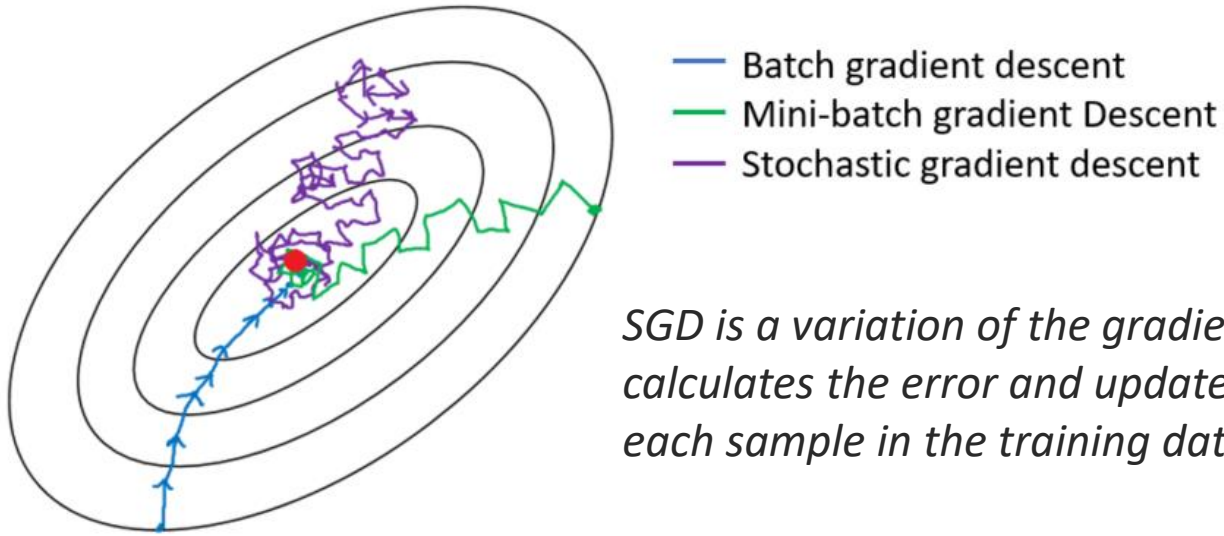
- 流形學習(manifold learning)
  - 廣義來說包含了數據科學中與幾何有關的理論。其中一個主要的分支為**低維嵌入**，包含了**多尺度轉換（multi-dimensional scaling, MDS）**、**等度量映射（ isometric mapping, isomap）**、**局部線性嵌入（locally linear embedding, LLE）**、**拉普拉斯特徵映射（Laplacian eigenmap, LE）**以及**隨機鄰近嵌入法（stochastic neighbor embedding, SNE）**等方法。這些方法的目的包含了維度縮減、視覺化以及分群，是使我們快速理解高維數據分布特性的一類方法。

t分配隨機鄰近嵌入法(t-distributed SNE, t-SNE)

- 將樣本間的相似度以**機率密度**表示，由原始高維數據嵌入到低維數據中
  - 其中原始空間的相似度是以高斯分配的機率密度表示，而嵌入空間則以t分配的機率密度(長尾的t分配較能放大原始高維空間距離相對較大的樣本，並且受離群值的影響較小)。
- 在高維特徵空間中，樣本點之間的相似度以高斯分配的條件機率表示如
  - $p_{j|i} = \frac{\exp\left(-\|x_i - x_j\|^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} \exp -\|x_i - x_k\|^2 / 2\sigma_i^2}$
  - 其中σ_i是以x_i為高斯分配的標準差，而每個樣本x_i對應其自己的標準差σ_i。
- 但由於樣本點的密度通常是不平均的，因此使用**困惑度**(perplexity)以二分法搜尋最佳的標準差σ_i，如公式(11.20)所示，
  - $Perp(P_i) = 2^{-\sum_j p_{j|i} \log p_{j|i}}$
  - 其中指數項為熵(entropy)，將其視為平滑過後的有效鄰居數。在低維特徵空間中，樣本點之間的相似度以t分配的條件機率表示如公式
  - $q_{ij} = \frac{\left(1 + \|y_i - y_j\|^2\right)^{-1}}{\sum_{k \neq i} (1 + \|y_i - y_k\|^2)^{-1}}$

t分配隨機鄰近嵌入法(t-distributed SNE, t-SNE)

- 最後我們的目標是盡可能使高維與低維中的條件機率分布越相似越好，因此最小化的價值函數(cost function)為高維與低維的「**KL散度**」(KL-divergence, KLD) (請參閱章節「**類神經網路與深度學習**」的介紹)。
  - $C = \sum_i D_{KL}(P_i||Q_i)$
- 通常採用**隨機梯度下降法**(stochastic gradient descent, SGD)求解上式
  - $\frac{\rho C}{\rho y_i} = 4\sum_i (p_{ij} - q_{ij})(y_i - y_j)\left(1 + \|y_i - y_j\|^2\right)^{-1}$



SGD is a variation of the gradient descent that calculates the error and updates the model for each sample in the training datasets.

https://sweta-nit.medium.com/batch-mini-batch-and-stochastic-gradient-descent-e9bc4cacd461

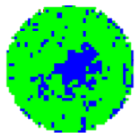
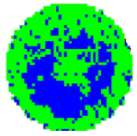
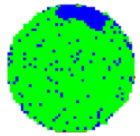
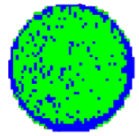
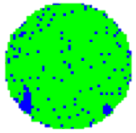
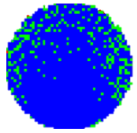
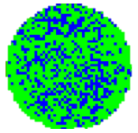
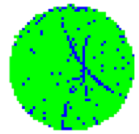
t分配隨機鄰近嵌入法(t-distributed SNE, t-SNE)

- 開放晶圓圖數據（WM-811K）為例，以t-SNE對該數據視覺化。實際上能將高維度的影像轉換到二維度空間中，並可將不同缺陷類型的晶圓圖進行分群。此外，雖然t-SNE能使特徵的特性更清晰，但我們也需留意**隨機性**與**解釋性**，其每一次結果的隨機性跟隨機梯度下降法有關。

WM-811K缺陷模式				
缺陷名稱	Center	Donut	Edge-Loc	Edge-Ring
缺陷示意圖				
缺陷名稱	Loc	Near-full	Random	Scratch
缺陷示意圖				

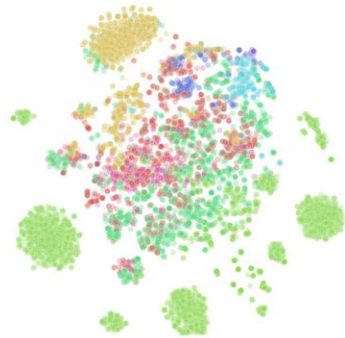
圖 9.13　晶圓圖（WM-811K）數據集 t-SNE 視覺化

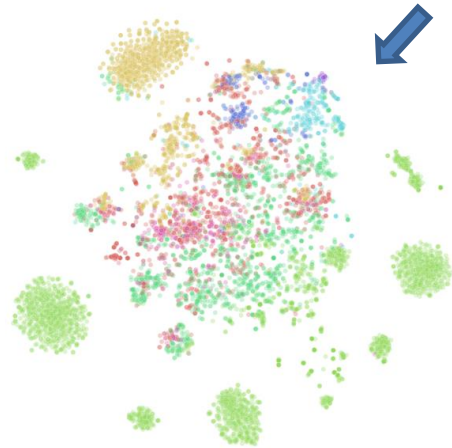
流形學習與視覺化

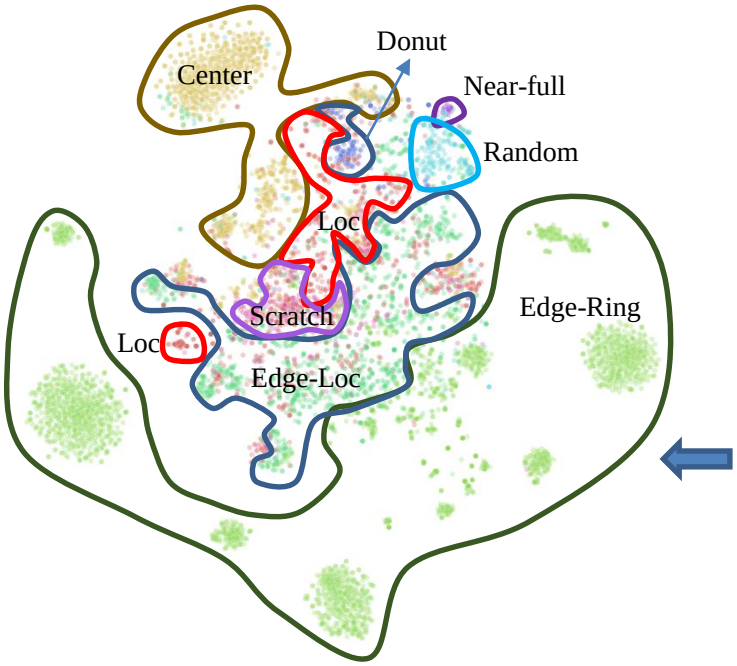
WM-811K 缺陷模式				
缺陷名稱	Center	Donut	Edge-Loc	Edge-Ring
缺陷示意圖				
數量(片)	4294	555	5189	9680
缺陷模式占比(%)	16.83%	2.17%	20.33%	37.93%
缺陷名稱	Loc	Near-full	Random	Scratch
缺陷示意圖				
數量(片)	3593	149	866	1193
缺陷模式占比(%)	14.08%	0.58%	3.39%	4.67%

Do PCA, then t-SNE.

- Loc
- Center
- Edge-Ring
- Edge-Loc
- Random
- Donut
- Near-full
- Scratch







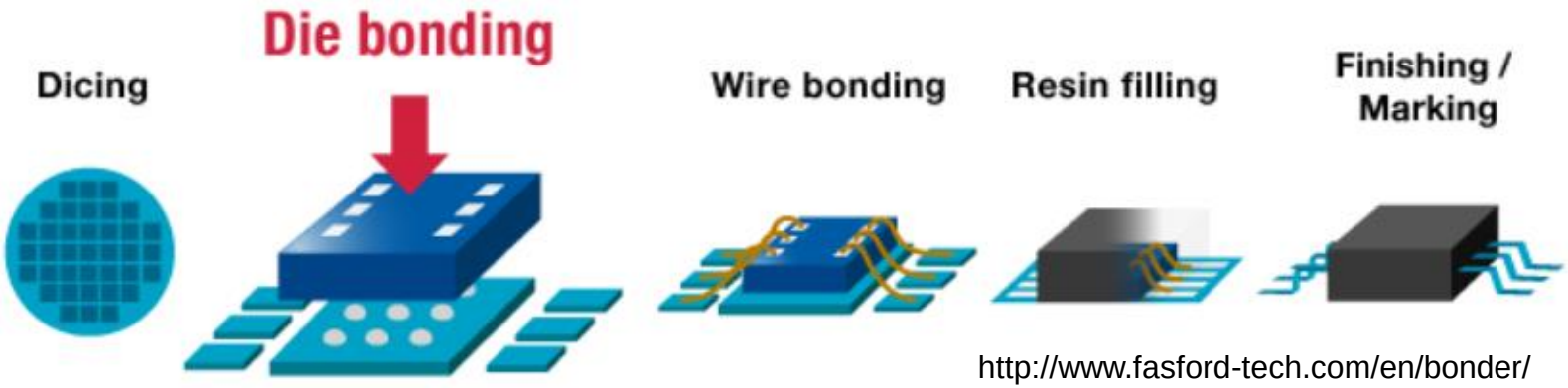
實證研究：半導體封裝脫層問題

- Semiconductor Technology at tsmc, 2011
  - https://www.youtube.com/watch?v=4Q_n4vdyZzc

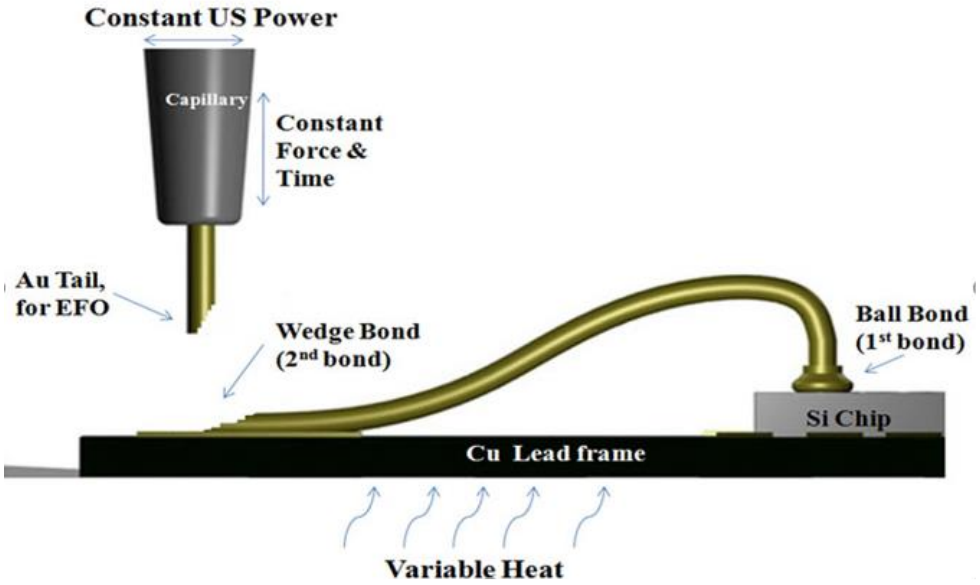
實證研究：半導體封裝脫層問題

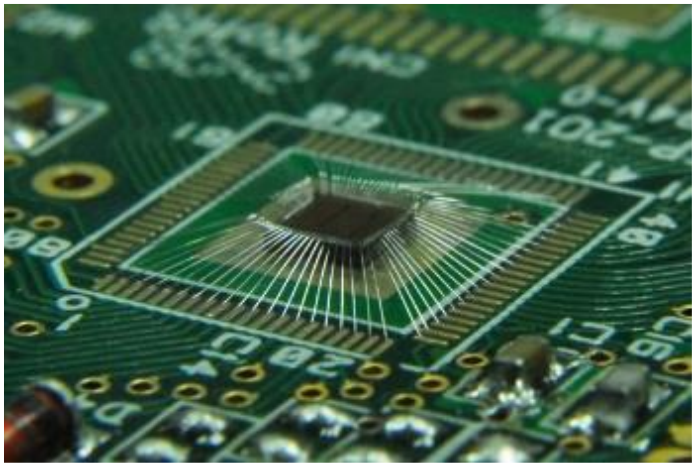
半導體封裝脫層問題(Delamination)

- 半導體封裝是將晶圓轉換為晶片的複雜製程，涉及數以千計的製程參數



http://www.fasford-tech.com/en/bonder/





實證研究：半導體封裝脫層問題

- **脫層**(Delamination)是導致產品產生缺陷的原因之一
  - 通常發生於**晶粒**（die）、**環氧基樹脂**（epoxy）、**環氧樹脂模塑膠**（epoxy molding compound, EMC）、**基板**（substrate）以及**導線架**（lead frame）等之間
  - 如圖所示（黑色區域為脫層的示意），其缺陷導致電阻的增加以及可靠度的下降。然而，工程技術與經驗進行故障排除（troubleshooting）通常需要**逐案**（case-by-case）分析，相當費時與費力，並且其中更涉及了高維度的製程參數、非線性的關係以及數據不平衡說明了問題的複雜度實際上相當高。

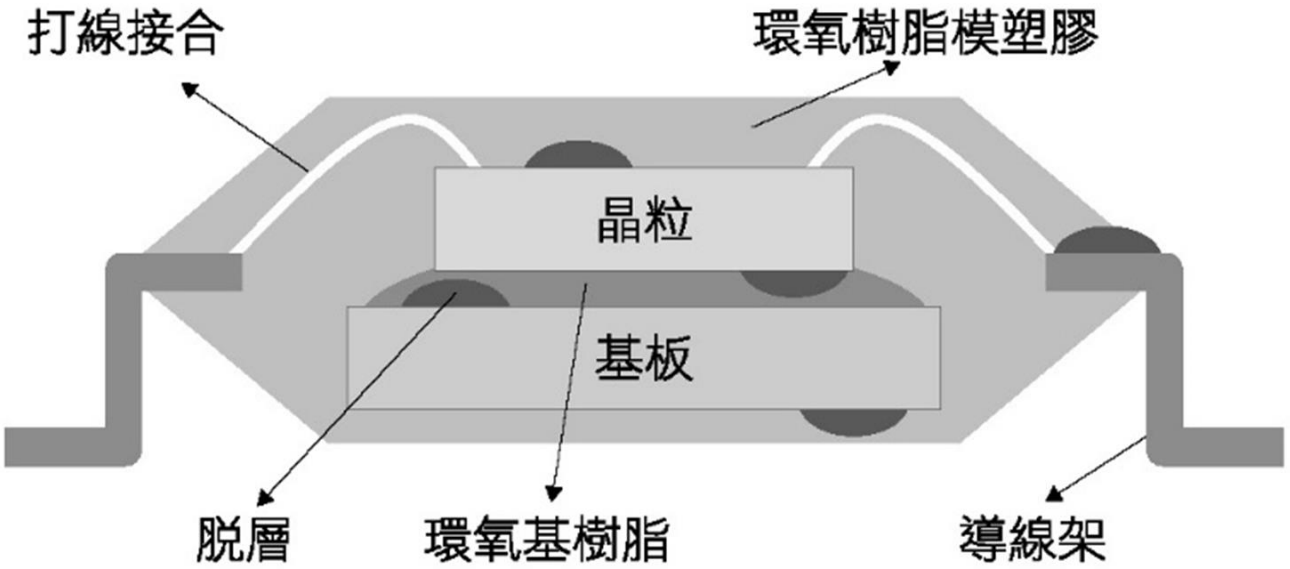


圖 9.14　界面脫層問題潛在發生的位置

實證研究：半導體封裝脫層問題

數據科學分析架構

- 首先，**數據預處理**將數據結構化，並由**特徵挑選**以**逐步迴歸**與**套索迴歸**找出造成脫層問題潛在的重要特徵，再以工程知識驗證特徵間的相關性
- 在**模型建構階段**以**類神經網路、支援向量機、梯度提升機**等模型進行預測與評估。此外，由於脫層的缺陷發生為少數事件，因此應留意**數據不平衡**的情形。

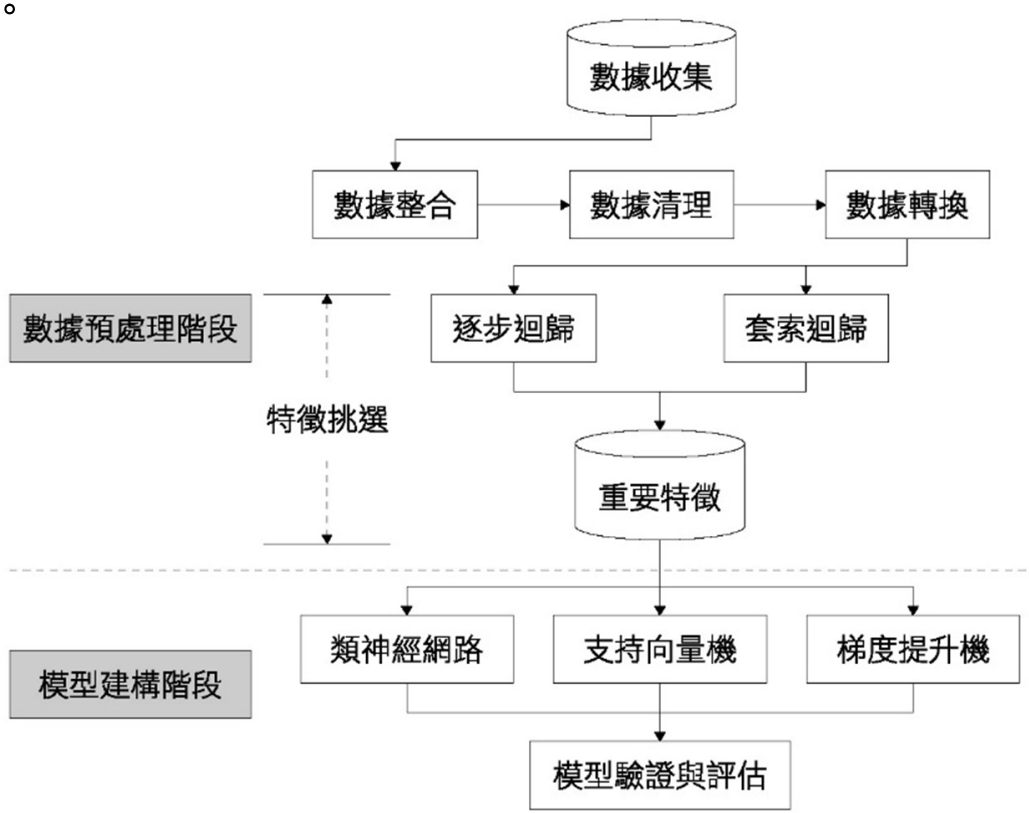
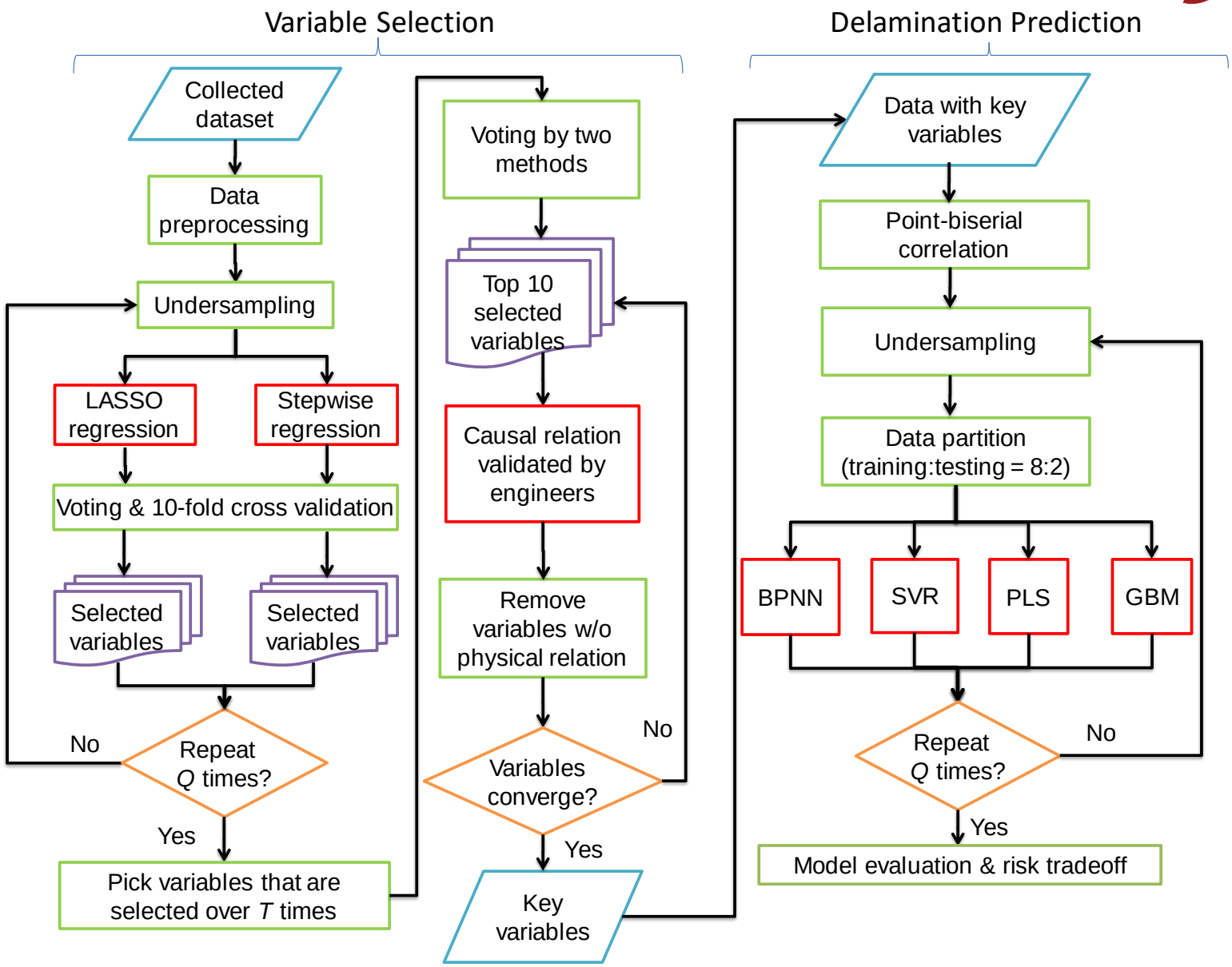


圖 9.15　脫層問題的數據科學分析架構

實證研究：半導體封裝脫層問題



Hung, S.-Y., Lee, C.-Y., and Lin, Y.-L., 2020. Data Science for Delamination Prognosis and Online Batch Learning in Semiconductor Assembly Process. IEEE Transactions on Components, Packaging and Manufacturing Technology, 10 (2), 314-324.

實證研究：半導體封裝脫層問題

原始數據	
資料來源	製造執行系統(MES) 與 全面控制方法(TCM)
資料筆數	約 7 百萬
特徵	途程控制、製程參數、機台參數與原物料等約 80 個特徵

數據預處理	
資料來源	製造執行系統(MES) 與 全面控制方法(TCM)
資料筆數	約 3 萬筆 (排除表單串接造成不完整與遺漏的數據)
特徵	經啞變數轉換後，轉成約 1000 個特徵

批號編號	製程	機台
Lot001	製程 1	A1
Lot001	製程 2	B1
Lot001	製程 3	C2
Lot002	製程 1	A2
Lot002	製程 2	B2
Lot002	製程 3	C1

數據整合

批號編號	製程 1_機台	製程 2_機台	製程 3_機台
Lot001	A1	B1	C2
Lot002	A2	B2	C1

圖 9.16　數據轉換

Class Imbalance Problem

Class Imbalance

- Random sampling deals with the issue. We focus on undersampling which samples a subset of the majority class.
- Undersampling (downsampling)
  - samples a subset of the majority class. Due to ignoring many majority class examples, we generally sample several times (resampling).
- Others: oversampling, cost-sensitive, ensemble, autoencoder, GAN,…

Example

- For Y label, PASS vs. FAIL = 1000 : 50
- Samples 50 PASS at a time for model training
- # of replications: 20 times
- Rank the variables by the “voting”
- Hint: 1:1 can be properly extended to 5:1

SVID	Voting by Undersampling
SVID_003	19
SVID_101	18
SVID_021	18
SVID_040	18
SVID_002	17
SVID_128	17
SVID_062	17
SVID_077	17
⋮	⋮

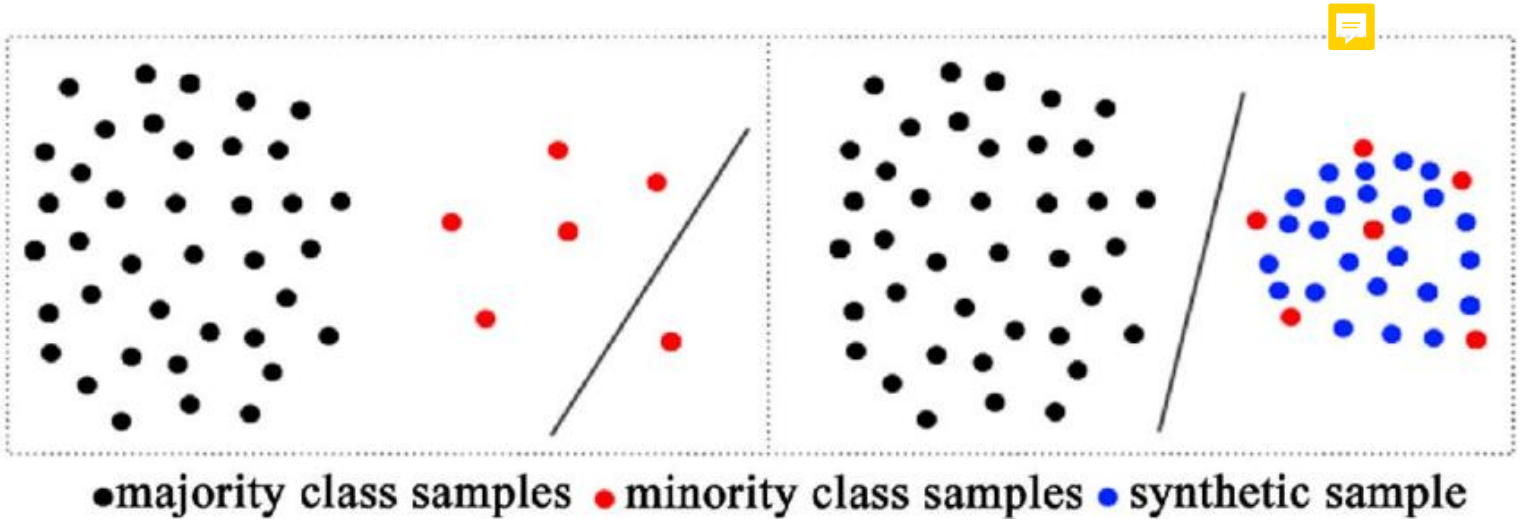
Pros and Cons

- Improve running time and storage problem
- Neglect potential useful information

Class Imbalance Problem

Class Imbalance Solutions

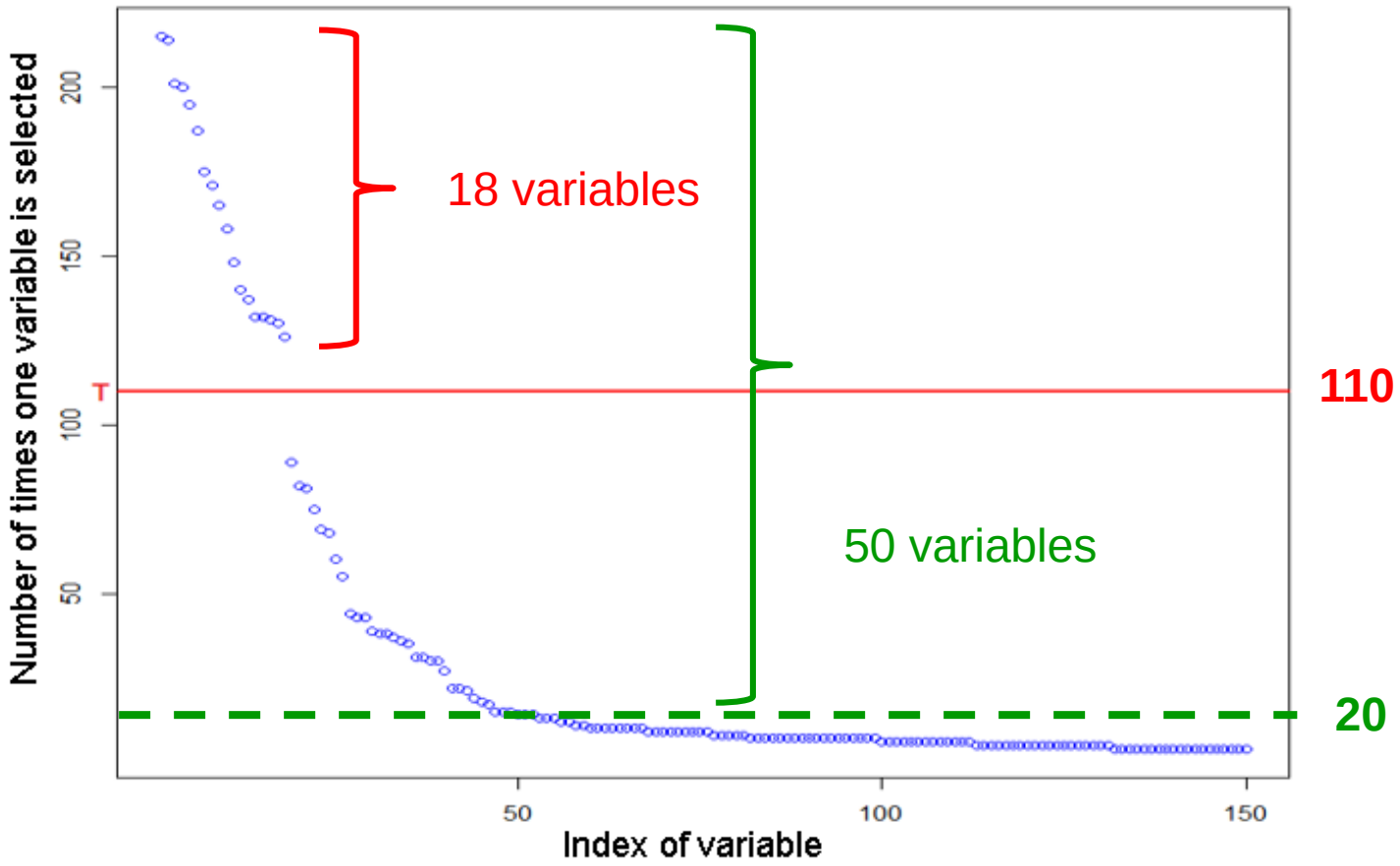
- Data Imbalance
  - Synthetic Data Generation
    - The Synthetic Minority Over-sampling TEchnique (SMOTE)
    - It uses bootstrapping and k-nearest neighbors to generate artificial data.
  - Pass:Fail: 1463:104 → Pass:Fail: 1463:607



Dang et al. (2013).A novel over-sampling method and its application to miRNA prediction. Journal of Biomedical Science and Engineering, 6 (2A), 236-248.
Hung (2018). https://rpubs.com/jeff_datascience/Semiconductor_Manufacturing

實證研究：半導體封裝脫層問題

Replications: $Q = \frac{\#\ of\ good\ products}{\#\ of\ bad\ products} = \frac{30000}{150} = 200$

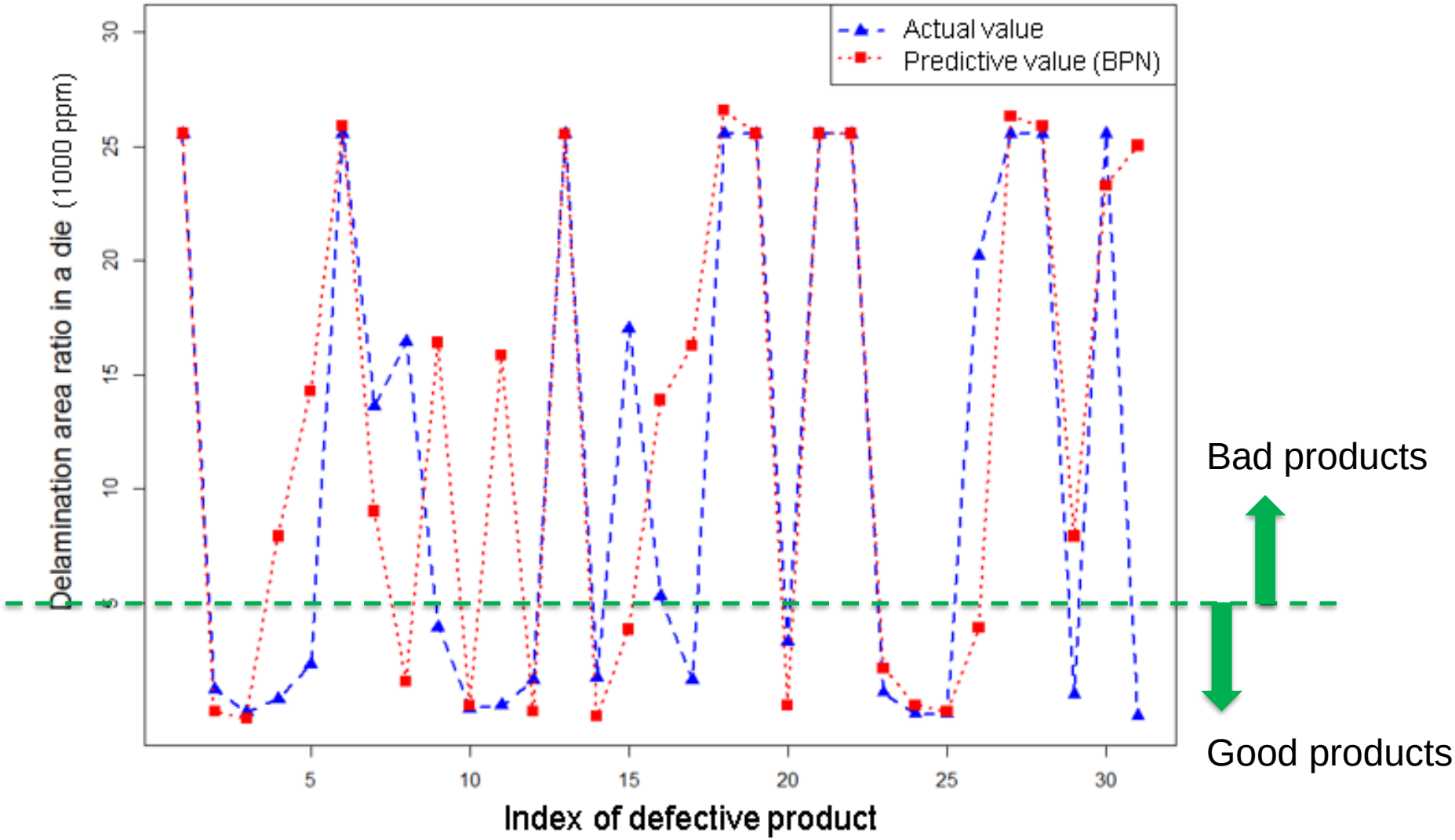


The scree plot of variable selecting frequency by Lasso

Hung, S.-Y., Lee, C.-Y., and Lin, Y.-L., 2020. Data Science for Delamination Prognosis and Online Batch Learning in Semiconductor Assembly Process. IEEE Transactions on Components, Packaging and Manufacturing Technology, 10 (2), 314-324.

實證研究：半導體封裝脫層問題

- Delamination area prediction



After adding a threshold, it become a "Classification Problem"

實證研究：半導體封裝脫層問題

	Avg. MSE		Testing(Avg.)			
	Training	Testing	Accuracy	TPR	TNR	AUC
BPN	8.33	37.13	71.16 %	0.927	0.462	0.695
SVR	17.03	37.39	59.57 %	0.902	0.204	0.588
PLS	42.06	53.13	57.98 %	0.896	0.273	0.585
GBM	8.29	37.07	78.82 %	0.685	0.908	0.796

(Prediction result)

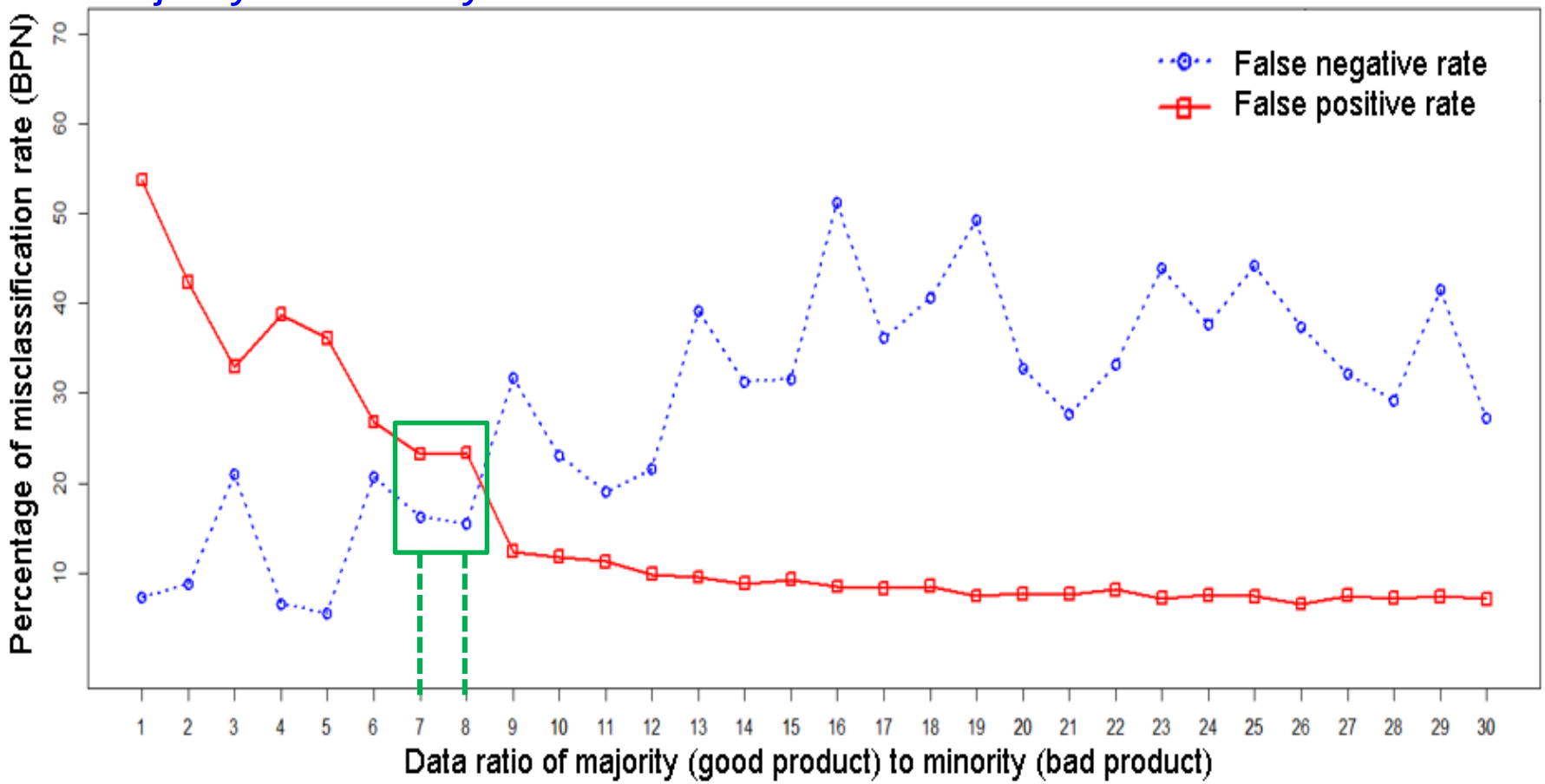
(Classification result)

		BPN Prediction		GBM Prediction	
		Bad(P)	Good(N)	Bad(P)	Good(N)
REAL	Bad(P)	29	2	22	9
	Good(N)	13	14	2	25

We may prefer using BPN due to a better TPR!

Tradeoff between False Positive and False Negative Risks

- Undersampling
  - Generally, half-half, i.e. PASS:FAIL=146:146
  - However, it may not be a best way and affect the two types of risks.
  - We investigate the "average misclassification rates under different majority-to-minority ratios"



實證研究：半導體封裝脫層問題

半導體封裝脫層問題(Delamination)

- **特徵挑選**是故障排除中最核心的流程。同時，為解決**數據不平衡**的問 題，採用**下抽樣（under-sampling）**，對多數群進行隨機抽樣，建立一新的數據子集其良品與不良品的比例接近（參閱後續章節「**特徵工程、數據增強與數據平衡**」），進行建模挑選重要參數，並重複以下抽樣多次建立不同的數據子集與建模，進行投票法。

- 最後，再將挑選出的重要特徵進行工程驗證確認在領域知識上的物理意義與因果關係。此案例中挑選出了像是**環氧類型**（epoxy type）、**設備類型**（device type）、**封裝方法**（package methods）等有意義的特徵，此外還找出一個曾未注意過的重要特徵，其交互作用顯著影響著**電漿清洗**（plasma cleaning）製程的品質。

結語

- 特徵挑選
  - 在數據科學中扮演著關鍵性的分析步驟之一。
  - 從方法本質上避免**過度配適**、**多元共線性**與解決**維度詛咒**等問題，以提升預測模型的穩健性與準確度。
  - 在特徵挑選中，我們將這些方法歸納成**過濾**、**包裝**、**嵌入**以及**維度縮減**法，這些方法整理出其假設、使用時機與優劣。
  - 然而，我們也留意在特徵挑選過程中，數據科學多協助確認特徵間的**相關性**，其物理特性與因果關聯仍需領域知識確認與驗證

- 因果關係三個必要條件(necessary condition)
  - **相關性**
  - **順序性**(sequence) (意指時間先後)
  - **連貫性**(coherence) (意指因果發生以**近因**為主而非遠因)

表 9.12　投票法於特徵挑選

	方法	假設與使用時機	優點	缺點
過濾法	相關性篩選	特徵間低度線性相關性	1. 運算效率非常快 2. 輔助我們初步判斷數據合理性	沒有考慮特徵之間的相關性以及交互作用
包裝法	最佳子集挑選	特徵維度不高	可得到全域最佳的特徵子集	運算負擔非常龐大，與維度呈指數成長
	向前或向後逐步挑選	特徵維度不可太高	運算效率高	1. 受順序相依性影響 2. 當重要特徵過多時運算負擔也會變得龐大
	逐步迴歸	特徵維度不可太高	受較小的順序相依性影響	當重要特徵過多時運算負擔也會變得龐大
嵌入法	套索迴歸	特徵維度高且特徵間高度線性相關	1. 運算效率非常高 2. 對偏誤與變異有較好的平衡	1. 需調整一個超參數 2. 高相關特徵會擇一挑選
	彈性網路	特徵維度高且特徵間高度線性相關	1. 具備套索迴歸的優點 2. 能同時保留重要且高相關的特徵	需調整兩個超參數
	集成學習	特徵與應變數有非線性或交互作用關係	能找出具有非線性與交互作用的重要特徵	1. 集成演算法需調整的超參數較多 2. 無法確定特徵是如何影響預測結果
維度縮減法	主成分分析	特徵間高度線性相關（物理特性相似）	對於高維度數據能將其縮減，並同時保留資訊量	1. 新特徵的解釋不易 2. 不易與其他特徵挑選方法一起使用
	獨立成分分析	特徵來源數量已知且彼此獨立	還原後的特徵具備物理意義與分析價值	1. 實務上較難符合獨立假設 2. 需知信號源的數量

Thanks for your attention

NTU Dept. of Information Management
name: 李家岩 (FB: Chia-Yen Lee)
phone: 886-2-33661206
email: chiayenlee@ntu.edu.tw
web: https://polab.im.ntu.edu.tw/