R 語言與機器學習 (五)

丘祐瑋
David Chiu

降低維度

降低維度

- 影響事情發展的因素是多元性的;但不同因素之間會互相影響(共線性)，或相重疊;進而影響到統計結果的真實性
  - 使用降低維度來降低訊息重疊
  - 使用降低維度來減少工作量
  - 找出一個互不相關的綜合指標來反映原本數據所含大部分的訊息

降低維度的應用

- 透過產生一有最大變異數的欄位線性組合，可用來降低原本問題的維度與複雜度
  - e.g. 濃縮用到的特徵，編纂成一個新指標
  - 產生經濟指標

- 去掉不必要的變量
  - e.g.減少工作量，還有避免誤用指標
  - 變量排序與篩選

降低維度的歷史

- Stone 於1974 年對美國1929 ~ 1938 年經濟數據的研究
  - 透過降低維度找到三個主成分(解釋度達97.4%)，以總結17個變數
    - F1 總收入
    - F2 總收入變化率
    - F3 經濟發展趨勢
  - 主成分保留了原始變量大部分的訊息
  - 主成分個數少於原變量的個數
  - 主成分之間互不相關
  - 每個主成分都是原始變數的線性組合

降低維度的方法

- 選擇特徵 (Feature Selection)
  - 從原有的特徵中挑選出最佳的部分特徵
  - 能夠簡化分類器的計算
  - 幫助瞭解分類問題的因果關係

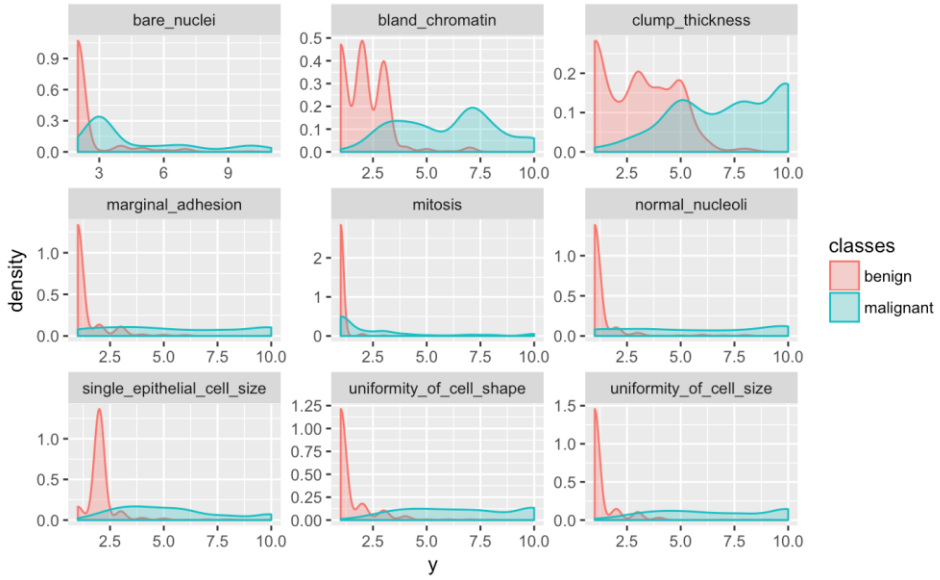
- 抽取特徵 (Feature Extraction)
  - 將資料群由高維度的空間中投影到低維度的空間
  - 找出一組基底向量（base）來進行線性座標轉換
  - 使得轉換後的座標，能夠符合某一些特性

選擇特徵

- 選擇特徵 (Feature Selection)
  - 從原有的特徵中挑選出最佳的部分特徵
  - 能夠簡化分類器的計算
  - 幫助瞭解分類問題的因果關係

檢視特徵

```
library(tidyr)
library(ggplot2)
gather(bc_data, x, y, clump_thickness:mitosis) %>%
  ggplot(aes(x = y, color = classes, fill = classes)) +
    geom_density(alpha = 0.3) +
    facet_wrap( ~ x, scales = "free", ncol = 3)
```



特徵排序

特徵排序 (Feature Ranking)

- 根據設定條件將特徵做排序，並從中選擇高於閥值的特徵

```
for each feature F_i
   wf_i = getFeatureWeight(F_i)
   add wf_i to weight_list
sort weight_list
choose top-k features
```

資料集

- 給予腫瘤切片的特徵，預測該腫瘤是惡性腫瘤(malignant)還良性腫瘤(benign)？

1. Sample code number
2. Clump Thickness（腫塊厚度）
3. Uniformity of Cell Size（細胞大小）
4. Uniformity of Cell Shape （細胞形狀）
5. Marginal Adhesion （邊緣粘度）
6. Single Epithelial Cell Size（單獨上皮細胞大小
7. Bare Nuclei （裸細胞核）
8. Bland Chromatin（淡染色質）
9. Normal Nucleoli（正常細胞核）
10. Mitoses （分裂激素）
11. Class

讀取資料集

```
url <- 'http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/breast-cancer-wisconsin.data'
bc_data <- read.csv(url, header = FALSE)
colnames(bc_data) <- c("sample_code_number",
                       "clump_thickness",
                       "uniformity_of_cell_size",
                       "uniformity_of_cell_shape",
                       "marginal_adhesion",
                       "single_epithelial_cell_size",
                       "bare_nuclei",
                       "bland_chromatin",
                       "normal_nucleoli",
                       "mitosis",
                       "classes")

bc_data$classes <- ifelse(bc_data$classes == "2", "benign",
                          ifelse(bc_data$classes == "4", "malignant", NA))
```

資料清理與轉換

- 去除空值資料

```
bc_data[bc_data == "?"] <- NA
sum(is.na(bc_data))
nrow(bc_data)

bc_data <- na.omit(bc_data)
sum(is.na(bc_data))
nrow(bc_data)
```

特徵排序實作

```
#install.packages("FSelector")

library(FSelector)
weights = random.forest.importance(classes~., bc_data,
importance.type = 1)
print(weights)
subset = cutoff.k(weights, 5)
f = as.simple.formula(subset, "classes")
```

使用caret 套件排序特徵

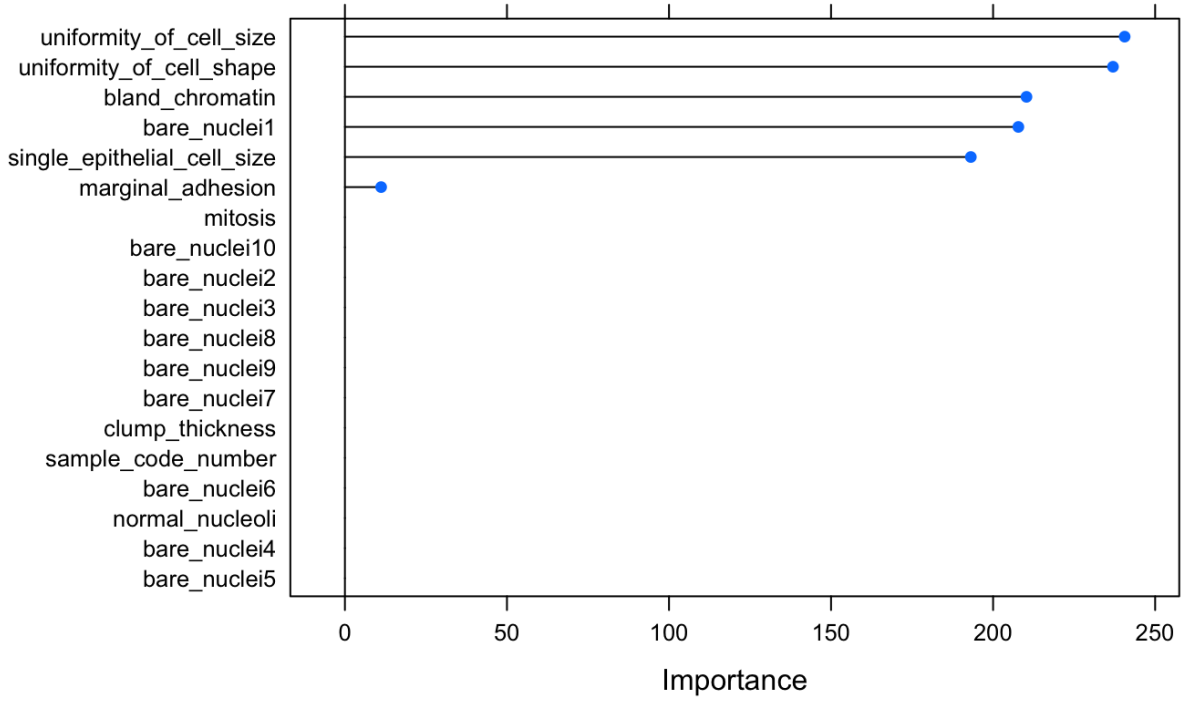
```
library(caret)
control <- trainControl(method="repeatedcv", number=10, repeats=3)

model <- train(classes~., data=bc_data, method="rpart", preProcess="scale",
trControl=control)

importance <- varImp(model, scale=FALSE)
importance
```

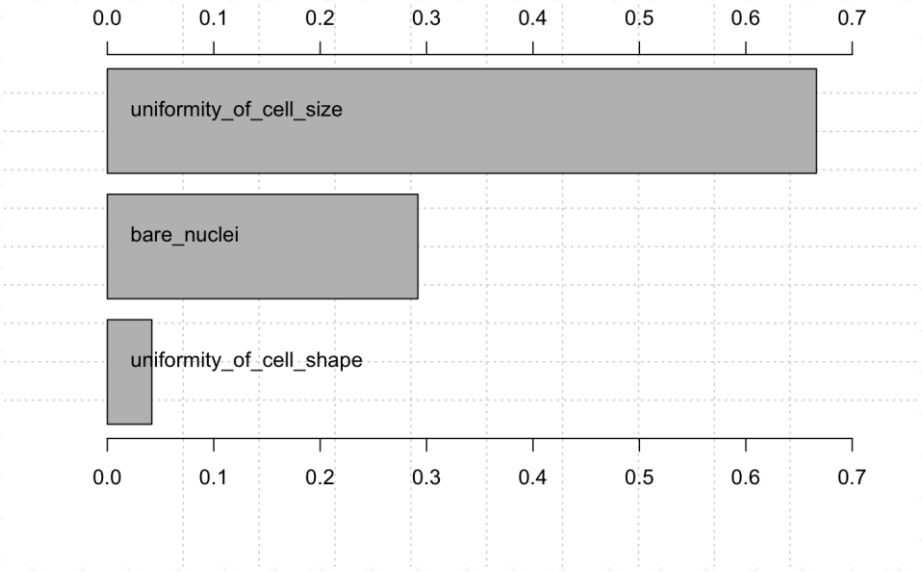
繪製特徵排序圖

plot(importance)



使用rminer

```
#install.packages("rminer")
library(rminer)
model<-fit(classes~.,bc_data,model="rpart")
VariableImportance=Importance(model,bc_data,method="sensv")
L<-list(runs=1,sen=t(VariableImportance$imp),sresponses=VariableImportance$sresponses)
mgraph(L,graph="IMP",leg=names(bc_data),col="gray",Grid=10)
```



特徵篩選

移除高相關性的變量

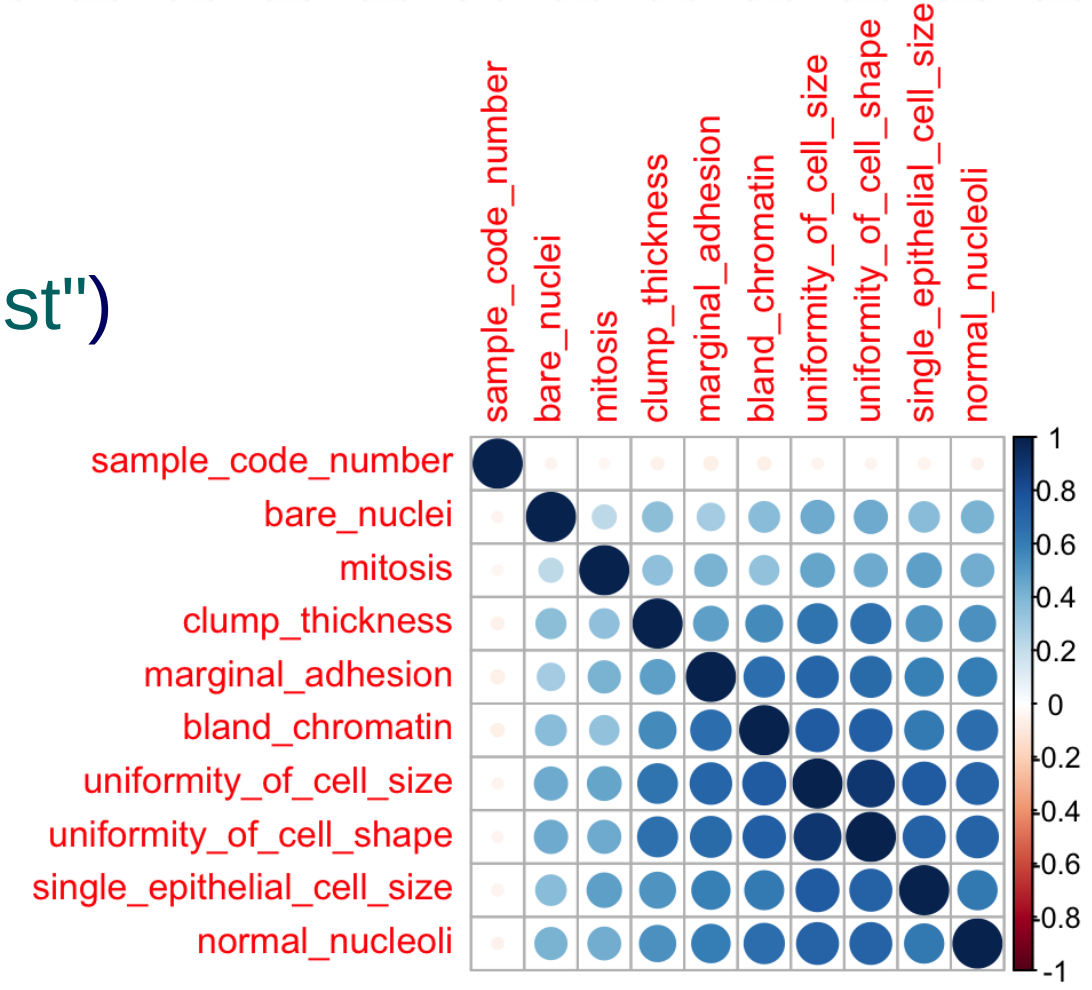
- 相關性高的變量會導致模型誤判，可以先找出相關性高的變量，再予以刪除，便可以提高模型辨識的準確度。

- 使用 cor

```
cor(bc_data[, -11])
```

檢視特徵相關性

```
library(corrplot)
corMatMy <- cor(bc_data[, -11])
corrplot(corMatMy, order = "hclust")
```



移除高相關性的變量

```
highlyCor <- colnames(bc_data[, -11])[findCorrelation(corMatMy,
cutoff = 0.7, verbose = TRUE)]
highlyCor
bc_data_cor <- bc_data[, which(!colnames(bc_data) %in%
highlyCor)]
```

分類演算法內建 (e.g. glm)

```
library(MASS)
model <- glm(classes~.,data=bc_data,family=binomial())
summary(model)
model.step <- stepAIC(model)
summary(model.step)
```

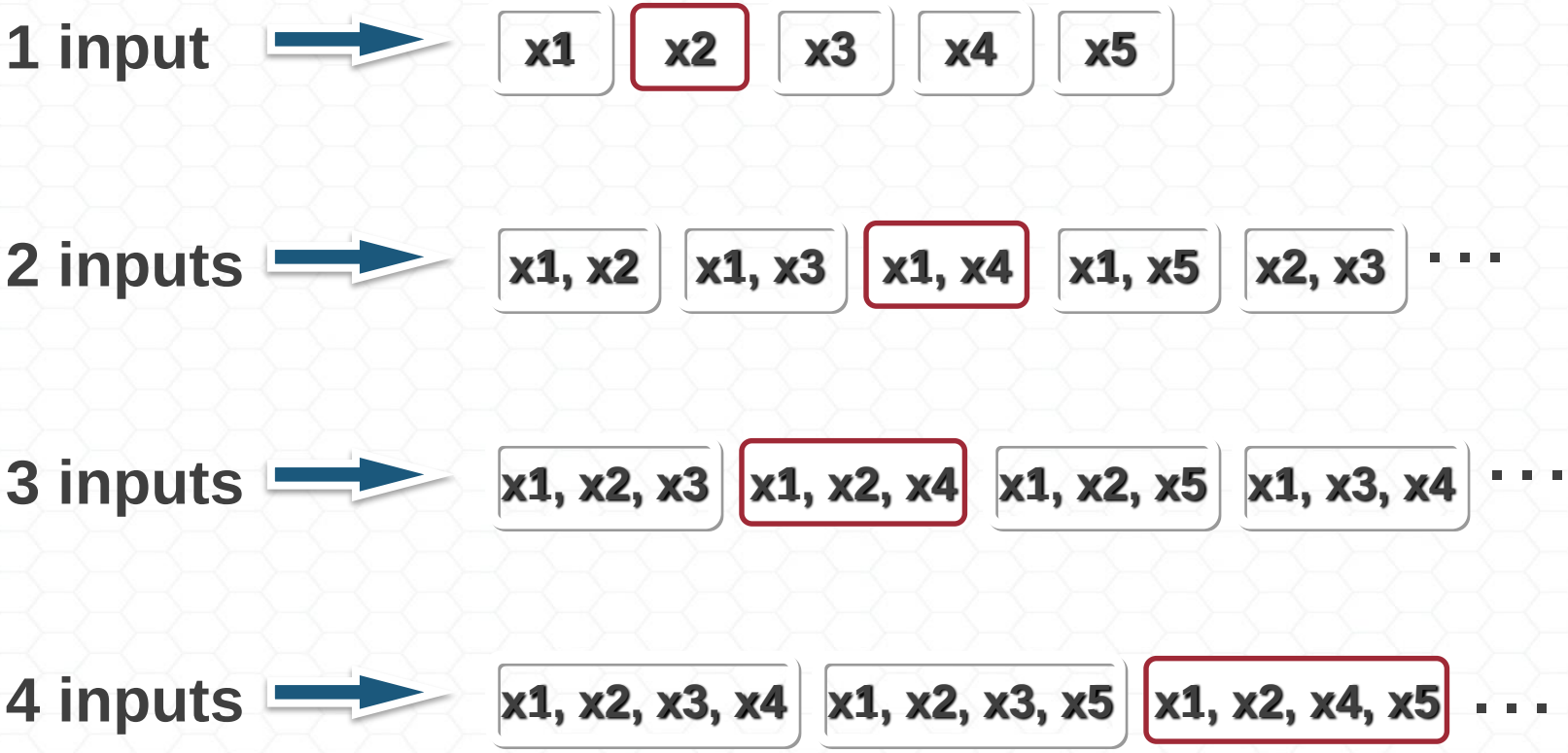
使用AIC 挑選變數

子集合選擇 (Subset Select)

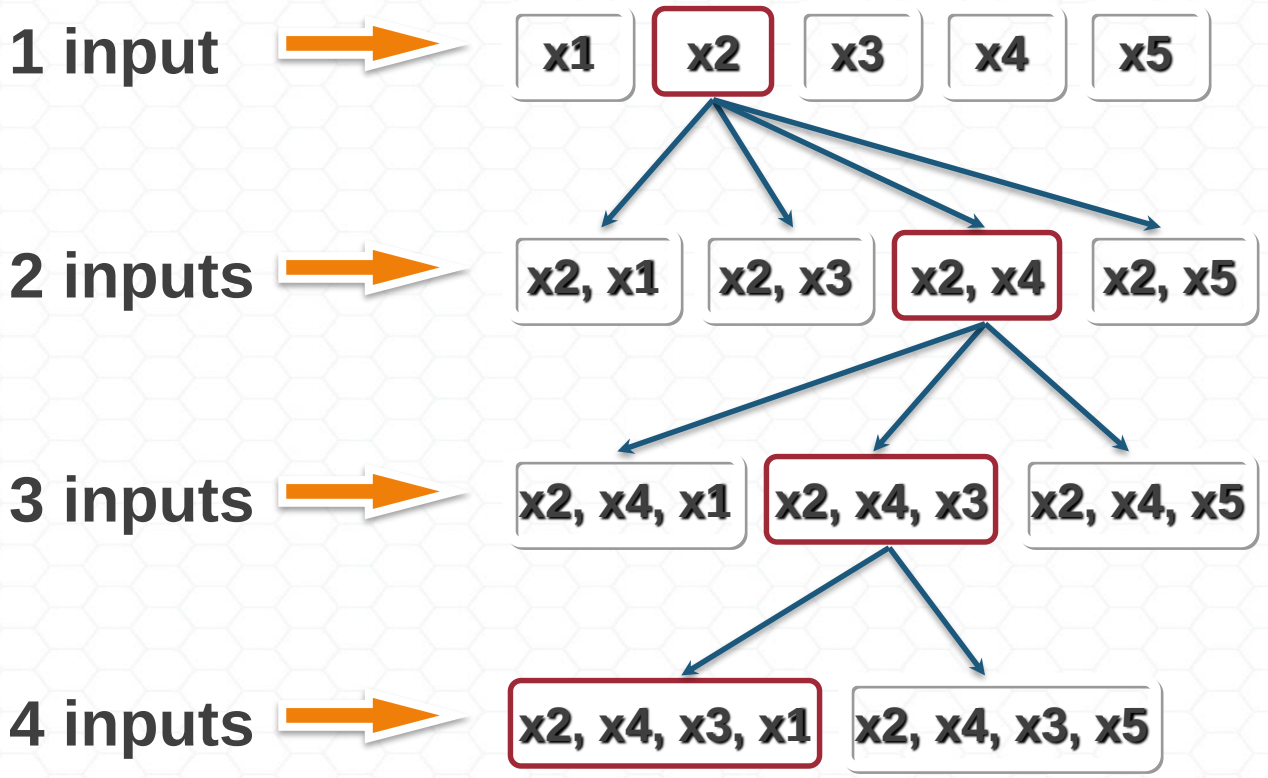
- 從既有特徵中選擇最好的子集合
  - 暴力法
  - 最近法 (Heuristic)
    - 第一個挑選的特徵必定是辨識率最高的特徵。
    - 下一個挑選的特徵必定是和原本已選取的特徵合併後，辨識率最高的一個。
    - 重複步驟 ，直至挑選出全部的特徵。

```
S = all subsets
for each subset s in S
   evaluate(s)
return (the best subset)
```

暴力法


1 input
x1
x2
x3
x4
x5
2 inputs
x1, x2
x1, x3
x1, x4
x1, x5
x2, x3
· · ·
3 inputs
x1, x2, x3
x1, x2, x4
x1, x2, x5
x1, x3, x4
· · ·
4 inputs
x1, x2, x3, x4
x1, x2, x3, x5
x1, x2, x4, x5
· · ·

最近法 (Forward Selection)



建立評估器 (Evaluator)

```
evaluator <- function(subset) {
 k <- 5
 set.seed(42)
 ind <- sample(5, nrow(bc_data), replace = TRUE)
 results <- sapply(1:k, function(i) {
   train <- bc_data[ind ==i,]
   test  <- bc_data[ind !=i,]
   tree  <- rpart(as.simple.formula(subset, "classes"), bc_data)
   error.rate <- sum(test$churn != predict(tree, test, type="class")) / nrow(test)
   return(1 - error.rate)
 })
 return(mean(results))
}
```

找出辨識率最高的子集合

```
attr.subset <- hill.climbing.search(names(bc_data)[!names(bc_data) %in% "classes"], evaluator)
f<-as.simple.formula(attr.subset, "classes")
print(f)
```

Recursive Feature Elimination (RFE)

```
library(caret)
set.seed(42)
results_rfe <- rfe(x = bc_data[, -11],
                   y = bc_data$classes,
                   sizes = c(1:9),
                   rfeControl = rfeControl(functions = rfFuncs, method = "cv", number = 10))
```

使用 Caret 內建的 RFE

抽取特徵

抽取特徵

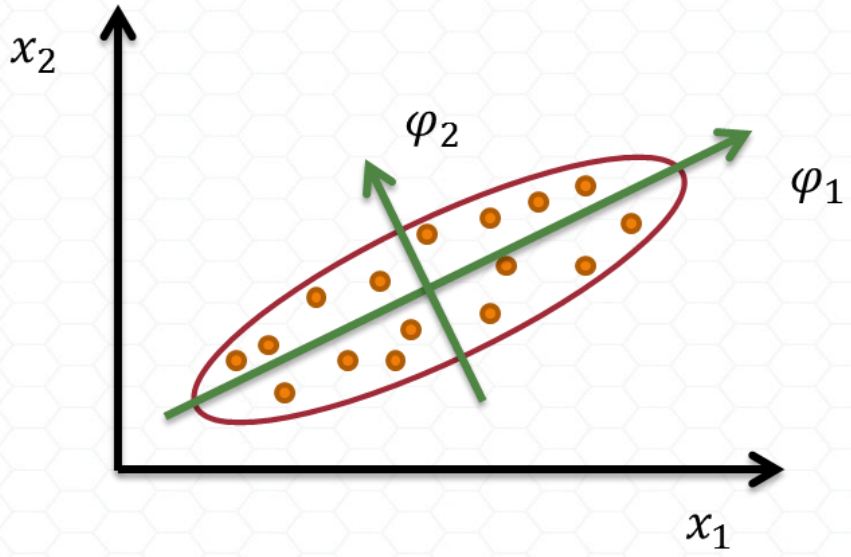
- 抽取特徵 (Feature Extraction)
  - 將資料群由高維度的空間中投影到低維度的空間
  - 找出一組基底向量（base）來進行線性座標轉換
  - 使得轉換後的座標，能夠符合某一些特性

主成分分析模型

- X1…Xp為p維向量，主成分分析可將p 個觀測量透過縣性組合轉換為p個新指標
  - $F_1 = u_{11}X_1 + u_{12}X_2 + \ldots u_{1p}X_p$
  - $F_2 = u_{21}X_1 + u_{22}X_2 + \ldots u_{2p}X_p$
  - $F_p = u_{p1}X_1 + u_{p2}X_2 + \ldots u_{pp}X_p$
- 滿足條件如下
  - 主成分係數平方和為1${u_{i1}}^2 + {u_{i2}}^2 + \cdots {u_{ip}}^2 = 1$
  - 主成分之間相互獨立 cov(Fi,Fj) = 0, $i \neq j$
  - 主成分的方差依重要性遞減
    - $Var(F_1) \geq Var(F_2) \ldots \geq Var(F_p)$

主成分分析目的

- 簡化變量
  - 主成分的個數小於原始變量的個數
  - 主成分盡可能反映原來變量的訊息



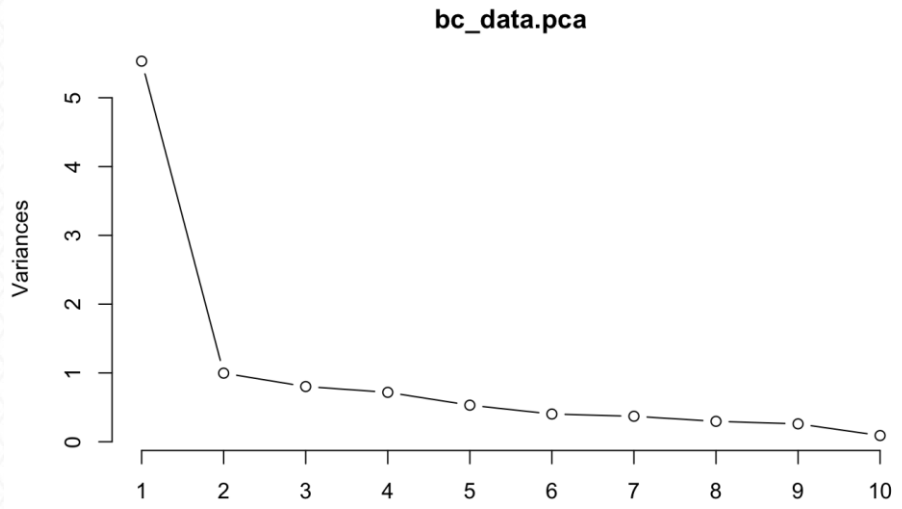
主成分分析

```
bc_data.pca <-prcomp(bc_data[, -11], center=TRUE, scale=TRUE)
summary(bc_data.pca)
predict(bc_data.pca, newdata=head(bc_data, 1))
```

碎石圖 (scree plot)

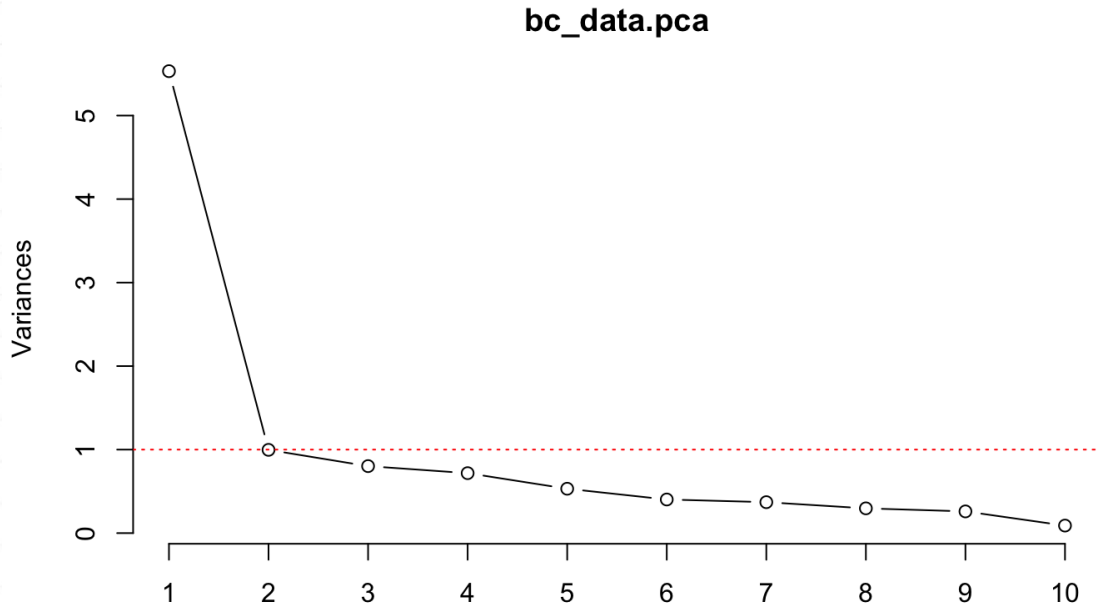
- 碎石圖 (scree plot)

```
screeplot(bc_data.pca, type="barplot")
screeplot(bc_data.pca, type="line")
```



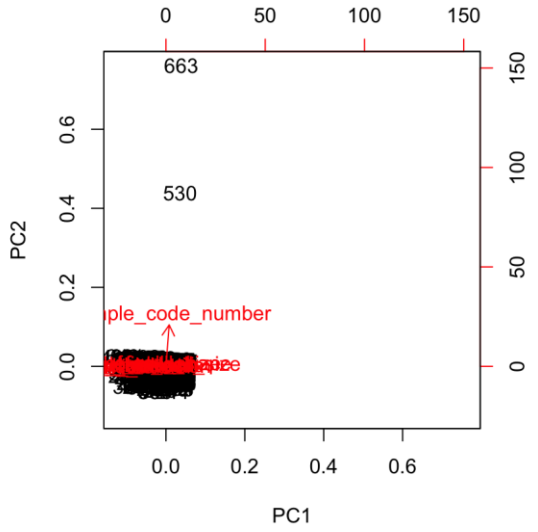
判斷主成分個數

```
bc_data.pca$sdev
bc_data.pca$sdev ^ 2
which(bc_data.pca$sdev ^ 2> 1)
screeplot(bc_data.pca, type="line")
abline(h=1, col="red", lty= 3)
```



PCA 雙邊圖

```
plot(bc_data.pca$x[,1], bc_data.pca$x[,2], xlim=c(-4,4))
text(bc_data.pca$x[,1], bc_data.pca$x[,2],
rownames(bc_data.pca$x), cex=0.7, pos=4, col="red")
biplot(bc_data.pca)
```



THANK YOU