村上　明
1/15/09
pp. 118-128

5. Clustering and classification

- 本書の残りは多変量解析を扱う
- データをグルーピングする（クラスタリングする）手法を 5.1 で、振り分ける手法を 5.2 で扱う

5.1 Clustering

5.1.1 Tables with measurements: principal components analysis

- affixProductivity というデータを用いる
  - テキスト内にどの程度それぞれの接辞が現れるかを数値化したもの。これにジャンル別差異があるかどうかを見たい→主成分分析
  - 44 テキスト×27 接辞
  - B：宗教関連テキスト　C：児童向け書籍　L：文学　O：その他
- 主成分分析（PCA：Principal components analysis)とは何か
  - p.119 の Figure 5.1 において
    - グレーになっているところがデータがある場所
    - 左上の figure を説明するには、xyz 軸全て必要
    - 右上だと軸をずらせば二軸でいい
    - 左下も同じ
    - 右下は 1 次元で良い
  - PCA はデータの位置を説明する次元を、軸を回転させることにより減らす
    - 左上の figure ではそれは不可能
    - 右上だとグレーのところは平面なので可能
    - 左下だとそれに加えて最も分散が大きいところを第一軸（主成分 1＝後ろ＆上に進む軸）とする
    - 主成分 2 は x 軸。主成分 3 は分散を説明しないので不必要
  - 実際のデータはもっと複雑だが、基本的な考え方は同じ：データの詰まった立方体を回転し、より少ない次元を分散説明力の大きい順に並べる
- 44 テキスト×27 変数なので、27 次元あると考えても良いが、この次元数を減らせないか
  - prcomp ( )関数を用いる。数値データの入ったマトリックスを入れる
    - 最後の三つのコラム（筆者名・ジャンル・筆者の生まれた年）は入れない

```
affixes.pr = prcomp(affixProductivity[ , 1:(ncol(affixProductivity)-3)])
```

  - names( )で構成要素を取得できる

```
names(affixes.pr)
```

  - sdev はそれぞれの主成分（27 個）の標準偏差

round(affixes.pr$sdev, 4)

- summary()でも標準偏差は出力される

summary(affixes.pr)

- proportion of variance = SD の二乗/Σ（SD の二乗）

props = round ((affixes.pr$sdev^2/sum(affixes.pr$sdev^2)), 3)

- 最初の次元が半分以上の分散を説明する
- 次元数の大雑把な指標：5%以上の分散を説明する次元
  - つまり Figure 5.2 (p.122)で黒く塗られている次元

barplot(props, col = as.numeric(props > 0.05), xlab = "principal components", ylab = "proportion of variance explained")
abline (h = 0.05)

- 似たグラフが以下で描ける

plot(affixes.pr)

- もう一つの大雑把な指標は、棒グラフを右から左に見て行き、最初に大きな差があるところ
  - やはりそれでも 3 次元
- 27 次元を 3 次元に減らしても元データの分散の 76.6%は説明できる

・ affixes.pr の x という要素の中に、三次元にした場合の座標が入っている

- 最初の三つだけを取り出す

affixes.pr$x[, 1:3]

- Figure5.3 が次元×次元の散布図
  - 描く手順は以下
  - lattice パッケージ内の splom ( )関数を用いる

```
library(lattice)
super.sym = trellis.par.get("superpose.symbol")

splom(data.frame(affixes.pr$x[,1:3]),
        groups = affixProductivity$Registers, panel = panel.superpose,
        key = list(
                title = "texts in productivity space",
                text = list (c("Religious", "Children", "Literary", "Other")),
                points = list (pch = super.sym$pch[1:4], col = super.sym$col[1:4])
        )
)
```

- rotation matrix について
  - 各主成分におけるそれぞれの接辞の負荷量
  - 主成分との相関を表す
  - 以下で出力できる

```
dim(affixes.pr$rotation)
affixes.pr$rotation[1:10, 1:3]
```

  - Figure 5.4(p.125)のような biplot を描いてみるのはデータ分析に有益
    - PC1 は ly 以外は loading が低い→ly の多寡を表している。Barrie には ly が高頻度で出現している
    - PC2 はもう少し広がりがあり、比較級・最上級が高頻度なテキストは上の方にある。-ation が多いテキストは下に。
  - biplot 関数を用いて Figure 5.4 を描く

```
biplot(affixes.pr, scale = 0, var.axes = F, col = c("darkgrey", "black"), cex = c(0.9, 1.2))
```

    - scale = 0 によりグラフ用に rescale することを防いでる
    - var.axes = F により矢印を表示しないようにしている
    - col でテキストと接辞の色を、cex でフォントサイズを指定している
    - 下と左の軸は主成分（？）を表し、上と右の軸は負荷量を表している
- PCA を行う際に気をつけるべきこと二点
  - 一点目：分布が左右対称であることが前提になっている
  - 二点目：列間の尺度を統一した方が良い。そうしなければ大きな範囲を持つ列が結果に強い影響を及ぼしてしまう。(-ly と-ation)
    - その場合は prcomp()に scale = TRUE を引数として入れると良い。共分散行列ではなく相関行列に基づく分析となる。

```
affixes.pr = prcomp(affixProductivity[ ,1:27], scale = T, center = T)
biplot(affixes.pr, var.axes = F, col = c("darkgrey", "black"), cex = c(0.6, 1), xlim = c(-0.42, 0.38))
```

    - native affix が右上に、non-native affix は左下に位置する傾向がある
    - 左下の non-native affix は硬い文章に高頻度である
    - native affix は子供向けの書籍により高頻度である

5.1.2 Tables with measurements: factor analysis

- 主成分分析を拡張させたのが探索的因子分析
  - PCA では分散を主成分で区切るので、ある主成分が説明する分散の割合は、「その分散/全ての主成分の分散の合計」で求めることができた
  - 一方、因子分析ではデータのノイズを考慮に入れ、誤差がモデルに加えられる
  - 軸を回転させることにより解釈が容易になる
  - 各変数が少数の因子に対してのみ因子負荷量が高い場合は解釈が容易である
- 先ほどのデータで因子分析を試みる

```
affixes.fac = factanal(affixProductivity[, 1:27], factors = 3)
```

  - 最尤法
  - 独自性（uniqueness）の後に因子負荷量が示されているが、0 に近いものは表示されない
  - 寄与率と検定結果がそのあとに続いている
  - Figure 5.5 (p.126)の下二つのグラフを描く

```
loadings = loadings(affixes.fac)
plot(loadings, type = "n", xlim = c(-0.4, 1))
text(loadings, rownames(loadings), cex = 0.8)
```

    - native affix と non-native affix の違いがよりよくわかる
    - non-native affix は右上にあり、naive affix は左下にある
    - nativeness が潜在変数だと言える
  - このままだと二因子によって説明されているが、promax 回転をすることにより一因子（Figure 5.5 の右下）で説明できるようになる

```
affixes.fac2 = factanal(affixProductivity[ ,1:27], factors = 3, rotation = "promax")
loadings2 = loadings(affixes.fac2)
plot(loadings2, type = "n", xlim = c(-0.4, 1))
text(loadings2, rownames(loadings))
abline(h = -0.1, col = "darkgrey")
```

  - non-native affix は下に、native affix は上にある
- 回転方法を決めるわかりやすい指標はない
  - varimax 回転は因子間に関連がないという前提で、結果の一般可能性を重視する場合に用いられる
  - promax 回転は因子間の相関を認め、データに合った因子モデルを得たい場合に用いられる