주성분 분석

주성분분석

- 상관관계가 존재하는 변수 $\boldsymbol{x}^T = (x_1, \ldots, x_q)$ 내의 변동을 $\boldsymbol{x}$변수들의 선형결합으로 서로 상관되지 않는 새로운 변수들인 $\boldsymbol{y}^T = (y_1, \ldots, y_q)$에 의해 설명
- $\boldsymbol{y}^T = (y_1, \ldots, y_q)$: 주성분 (principal components)
- 희망사항: 처음 몇 개의 주성분이 원래 변수 내의 변동의 많은 부분을 설명

활용 예

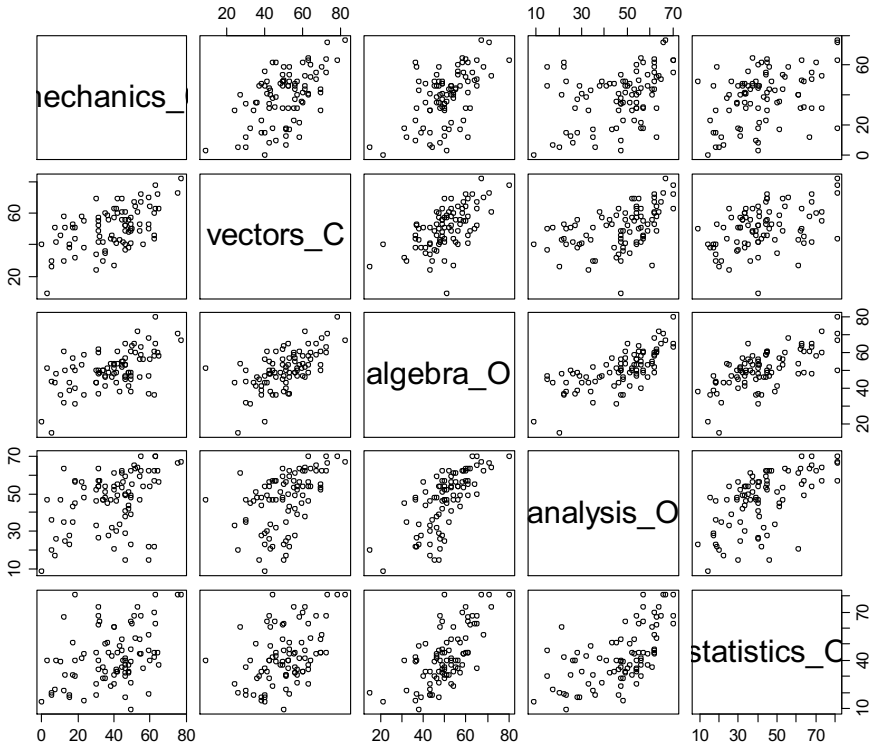
- 시험 수행의 전반적인 결과에 대한 정보를 나타내는 지수 작성
  - ✓순위 매기는 방법의 향상 위해 학생들의 점수를 최대한 펼치고 싶음
  - ✓점수 표준화, 가중치 고려
- 동물들의 형태학상 측정값을 사용해 동물 간의 구분을 위한 요소 찾음
  - ✓첫번째 주성분: 크기
  - ✓두번째 주성분: 모양 ➔관심사
- 그래프 표현 또는 다른 분석을 위한 입력변수로 활용

Example : Open/closed book

- Test score for mechanics, vectors, algebra, analysis, statistics

```
> head(data,10)
   mechanics_C vectors_C algebra_O analysis_O statistics_O
1           77        82        67          67           81
2           63        78        80          70           81
3           75        73        71          66           81
4           55        72        63          70           68
5           63        63        65          70           63
6           53        61        72          64           73
7           51        67        65          65           68
8           59        70        68          62           56
9           62        60        58          62           70
10          64        72        60          62           45
> str(data)
'data.frame': 88 obs. of  5 variables:
 $ mechanics_C : int  77 63 75 55 63 53 51 59 62 64 ...
 $ vectors_C   : int  82 78 73 72 63 61 67 70 60 72 ...
 $ algebra_O   : int  67 80 71 63 65 72 65 68 58 60 ...
 $ analysis_O  : int  67 70 66 70 70 64 65 62 62 62 ...
 $ statistics_O: int  81 81 81 68 63 73 68 56 70 45 ...

> cor(data)
             mechanics_C vectors_C algebra_O analysis_O statistics_O
mechanics_C    1.0000000 0.5534052 0.5467511  0.4093920    0.3890993
vectors_C      0.5534052 1.0000000 0.6096447  0.4850813    0.4364487
algebra_O      0.5467511 0.6096447 1.0000000  0.7108059    0.6647357
analysis_O     0.4093920 0.4850813 0.7108059  1.0000000    0.6071743
statistics_O   0.3890993 0.4364487 0.6647357  0.6071743    1.0000000
```

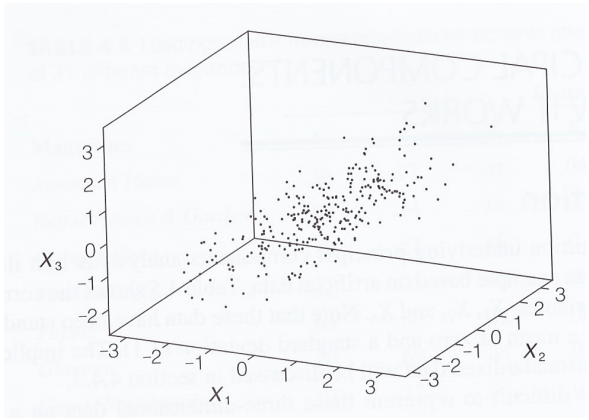


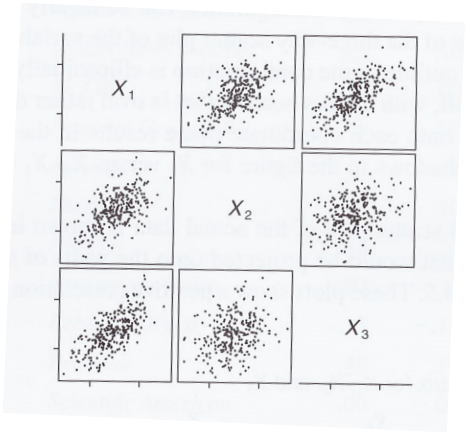
표본 주성분의 발견: 1st PC

- 표본분산이 모든 가능한 선형 결합 중에서 가장 큰 원래 변수들의 선형 결합

$$y_1 = a_{11}x_1 + a_{12}x_2 + \cdots + a_{1q}x_q$$

- $\boldsymbol{a_1^T a_1} = 1,$의 제약조건 하에서 y_1의 분산을 최대화 하는 $\boldsymbol{a_1^T} = (a_{11}, a_{12}, \ldots, a_{1q})$ 를 찾음





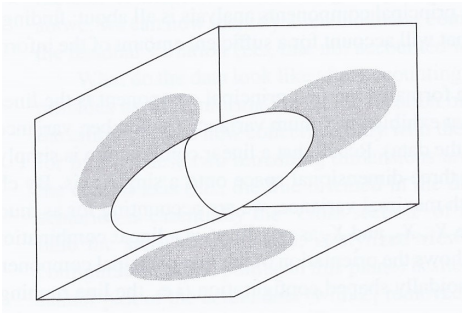


TABLE 4.5 Correlation matrix for X_1, X_2, and X_3

	X_1	X_2	X_3
X_1	1.000	0.562	0.704
X_2	0.562	1.000	0.304
X_3	0.704	0.304	1.000
	$\text{var}(X_1) = 1.00$	$\text{var}(X_2) = 1.00$	$\text{var}(X_3) = 1.00$

- S: $\boldsymbol{x}$의 $q \times q$ 표본공분산행렬
- Find $\mathbf{y_1} = \boldsymbol{a_1^T x}$ with the maximum variance

$$var(\boldsymbol{y_1}) = \boldsymbol{a_1^T S a_1}$$

만일 $\boldsymbol{x}$가 표준화 되었다면, $S = R,\ var(\boldsymbol{y}_1) = \boldsymbol{a}_1^T R \boldsymbol{a}_1$

- Maximize $\boldsymbol{a}' R \boldsymbol{a}$ subject to $\boldsymbol{a}' \boldsymbol{a} = 1$
  - ✓ Lagrange multiplier method

$$L = \boldsymbol{a}' R \boldsymbol{a} - \lambda(\boldsymbol{a}' \boldsymbol{a} - 1)$$

$$\frac{\partial L}{\partial \lambda} = 2R\boldsymbol{a} - 2\lambda \boldsymbol{a} = 0$$

$$R\boldsymbol{a} = \lambda \boldsymbol{a}$$

➔**Eigenvalue problem**

➔λ: $\boldsymbol{R}$ 의 고유값 (eigenvalue), $\boldsymbol{a}$: $\boldsymbol{R}$의 고유벡터 (eigenvector)

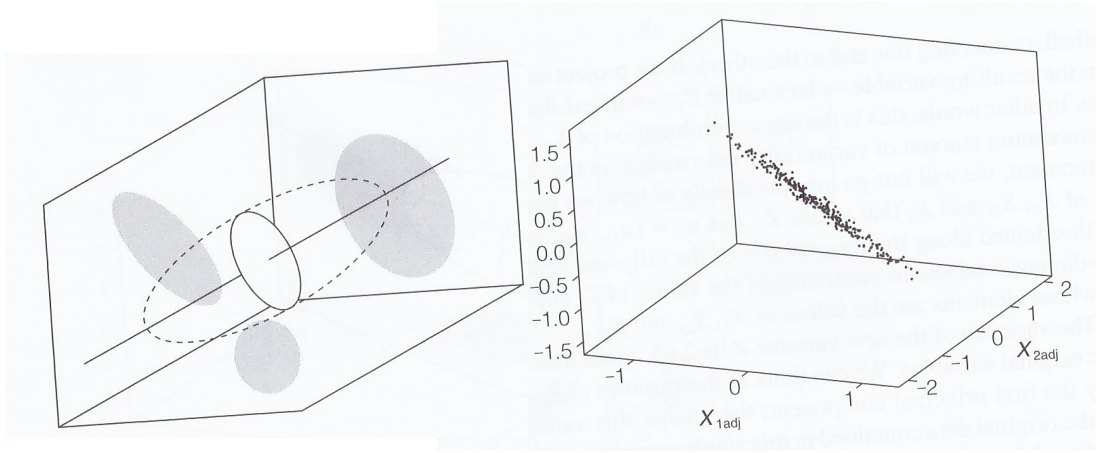
➔If R is full rank, there exist p of real number eigenvalues.

➔If R is positive definite, all the eigenvalues are positive.

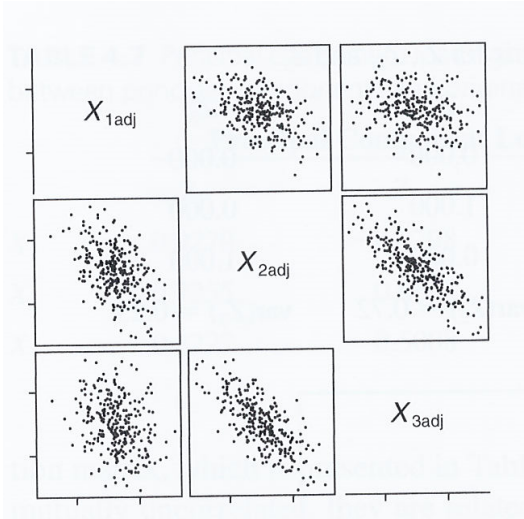
가장 큰 고유값에 대응하는 고유벡터=a_1

표본 주성분의 발견: 2nd PC

- $\boldsymbol{y_1}$의 정보를 제거한 후의 데이터는?
  - ✓1st PC에 수직인 평면에 데이터를 projection 시킴



- $\boldsymbol{y_1}$의 정보를 제거한 후의 scatter plot matrix



- 오른쪽 3d scatter plot에서 가장 큰 분산을 가지는 방향

$$\boldsymbol{a_2'} = (a_{21}, a_{22}, \ldots, a_{2q})$$

- 2nd PC: $\boldsymbol{y_2} = \boldsymbol{a_2'}X$

$$y_2 = a_{21}x_1 + a_{22}x_2 + \cdots + a_{2q}x_q$$
$$y_2 = \boldsymbol{a_2^T x}$$

- 제약식: $\boldsymbol{a_2^T a_2} = 1, \boldsymbol{a_2^T a_1} = 0$
- 두 번째로 큰 고유값에 대응하는 고유벡터=$\boldsymbol{a_2}$

S 가 $q \times q$ 표본공분산 행렬이고 $(\lambda_1, \boldsymbol{a_1}), (\lambda_2, \boldsymbol{a_2}), \ldots, (\lambda_q, \boldsymbol{a_q})$가 S의 고유값, 고유벡터라고 하자. i 번째 주성분은 아래와 같다.

$$y_i = \boldsymbol{a_i' x} = a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{iq}x_q, \qquad i = 1,2,\ldots,q$$

단, $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_q \geq 0$ 이고 $\boldsymbol{x}$ 는 변수 $X_1, X_2, \ldots, X_q$의 관측치이다.

관측된 변수들의 집합을 서로 관련되지 않은 새로운 변수들의 집합으로 변환

Example : Open/closed book

```
> pca=prcomp(data, scale=T)
> summary(pca)
Importance of components:
                          PC1    PC2     PC3     PC4     PC5
Standard deviation     1.7835 0.8600 0.66706 0.62281 0.49658
Proportion of Variance 0.6362 0.1479 0.08899 0.07758 0.04932
Cumulative Proportion  0.6362 0.7841 0.87310 0.95068 1.00000
```

변수를 표준화 한 후 주성분분석 시행
= 상관계수 행렬 사용

```
> pca
Standard deviations:
[1] 1.7835302 0.8599836 0.6670571 0.6228101 0.4965788

Rotation:
                  PC1        PC2         PC3        PC4        PC5
mechanics_C -0.3996045 -0.6454583  0.62078249 -0.1457865 -0.1306722
vectors_C   -0.4314191 -0.4415053 -0.70500628  0.2981351 -0.1817479
algebra_O   -0.5032816  0.1290675 -0.03704901 -0.1085987  0.8466894
analysis_O  -0.4569938  0.3879057 -0.13618182 -0.6662561 -0.4221885
statistics_O -0.4382444  0.4704545  0.31253342  0.6589164 -0.2340223
```

주성분 점수의 계산

- 원래 변수 벡터 $\boldsymbol{x_i}$ 로 갖는 개체 i 에 대한 m개의 주성분점수

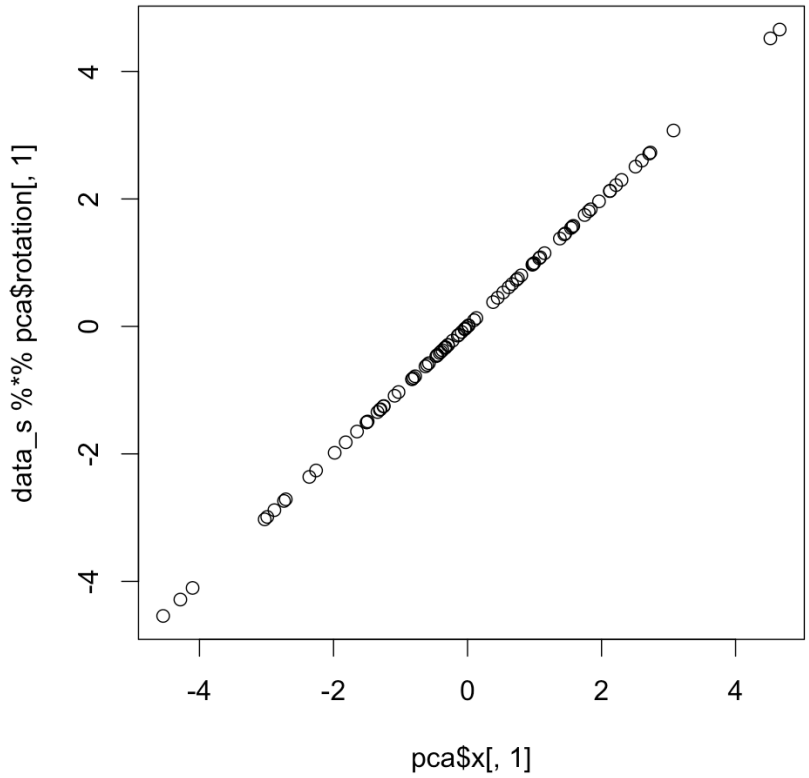
$$y_{i1} = \boldsymbol{a}_1^T \boldsymbol{x}_i$$
$$y_{i2} = \boldsymbol{a}_2^T \boldsymbol{x}_i$$
$$\vdots$$
$$y_{im} = \boldsymbol{a}_m^T \boldsymbol{x}_i$$

- 만약 상관행렬에서 주성분을 추출했다면 $\boldsymbol{x_i}$ 는 표준화된 값을 사용

Example : Open/closed book

```
> pca$x
              PC1           PC2         PC3          PC4          PC5
 [1,] -4.28504073 -0.6741022511  0.12358906  0.793110838 -0.514380331
 [2,] -4.54198890  0.2133117636 -0.23177965  0.551606340  0.596189735
 [3,] -4.10268998 -0.2755753002  0.53043774  0.609686182 -0.027815951
 [4,] -3.02684645  0.1491620709 -0.37021544  0.159588968 -0.439504772
 [5,] -2.88208144  0.0440801386  0.29888613 -0.322573985 -0.147677948
 [6,] -2.98877530  0.6812619625  0.26287555  0.295033099  0.547544853
 [7,] -2.71217828  0.3583687948 -0.20520172  0.283510624 -0.038914647
 [8,] -2.73843132 -0.4067907451 -0.28235205 -0.069407143  0.346963064
 [9,] -2.36071222  0.0785121645  0.64884113  0.315622048 -0.523982277
[10,] -2.26000540 -1.0556024144 -0.38343190 -0.404011350 -0.206387128
[11,] -1.98261286 -0.0724899833 -0.22661127 -0.186590832 -0.156616432
[12,] -1.81749094 -0.5948908808 -0.44944249 -0.470326449 -0.136137591
[13,] -1.64927411  0.5564135524  0.67558513  0.174431347  0.476086223
[14,] -1.50490281 -1.1894605379  0.05181387 -0.632204545  0.030312094
[15,] -1.50521738  1.3661450580 -0.09034238  0.779205750  0.037688844
[16,] -1.34334188 -1.0332448634 -0.09966038 -0.343081323 -0.089335633
```

```
> data_s=scale(data)
> plot(pca$x[,1],data_s%*%pca$rotation[,1])
```



주성분의 성질

PC k 의 분산	$\text{var}(y_k) = \lambda_k,\ \ k = 1, 2, \ldots, q$
PC i 와 PC k 의 공분산	$\text{cov}(y_i, y_k) = 0\ ,\ i \neq k$
총분산	$\sum_{i=1}^{q} s_{ii} = \lambda_1 + \lambda_2 + \ \ldots + \lambda_q$
PC i 와 k 번째 원변수와의 공분산	$\text{cov}(y_i, x_k) = \lambda_i a_{ik}$
PC i 와 k 번째 원변수와의 상관계수	$\text{cor}(y_i, x_k) = r_{y_i, x_k} = \frac{a_{ik}\sqrt{\lambda_i}}{\sqrt{s_{kk}}}, \quad i, k = 1,2, \ldots, p$
PC i 가 설명하는 총분산의 비율	$\frac{\lambda_i}{\lambda_1 + \lambda_2 + \ldots + \lambda_p}$ ($= \frac{\lambda_i}{p}$, 상관계수 행렬을 사용한 경우)

Example : Open/closed book

```
> pca=prcomp(data, scale=T)
> summary(pca)
Importance of components:
                          PC1    PC2     PC3     PC4     PC5
Standard deviation     1.7835 0.8600 0.66706 0.62281 0.49658
Proportion of Variance 0.6362 0.1479 0.08899 0.07758 0.04932
Cumulative Proportion  0.6362 0.7841 0.87310 0.95068 1.00000

> pca
Standard deviations:
[1] 1.7835302 0.8599836 0.6670571 0.6228101 0.4965788

Rotation:
                   PC1        PC2         PC3        PC4        PC5
mechanics_C  -0.3996045 -0.6454583  0.62078249 -0.1457865 -0.1306722
vectors_C    -0.4314191 -0.4415053 -0.70500628  0.2981351 -0.1817479
algebra_O    -0.5032816  0.1290675 -0.03704901 -0.1085987  0.8466894
analysis_O   -0.4569938  0.3879057 -0.13618182 -0.6662561 -0.4221885
statistics_O -0.4382444  0.4704545  0.31253342  0.6589164 -0.2340223
```

- Standard deviations: $\sqrt{\lambda_i}$➔ sum of the squares=5 (if the correlation matrix was used)
- Rotation: 열이 고유벡터➔ sum of squares of each column are 1.

```
> colSums(pca$rotation^2)
PC1 PC2 PC3 PC4 PC5
  1   1   1   1   1
> sum(pca$sdev^2)
[1] 5
```

주성분 개수의 선택

Q: 얼마나 많은 주성분들이 주어진 데이터셋의 적절한 요약을 위해서 필요한가?

$m(< q)$: 선택된 주성분의 개수

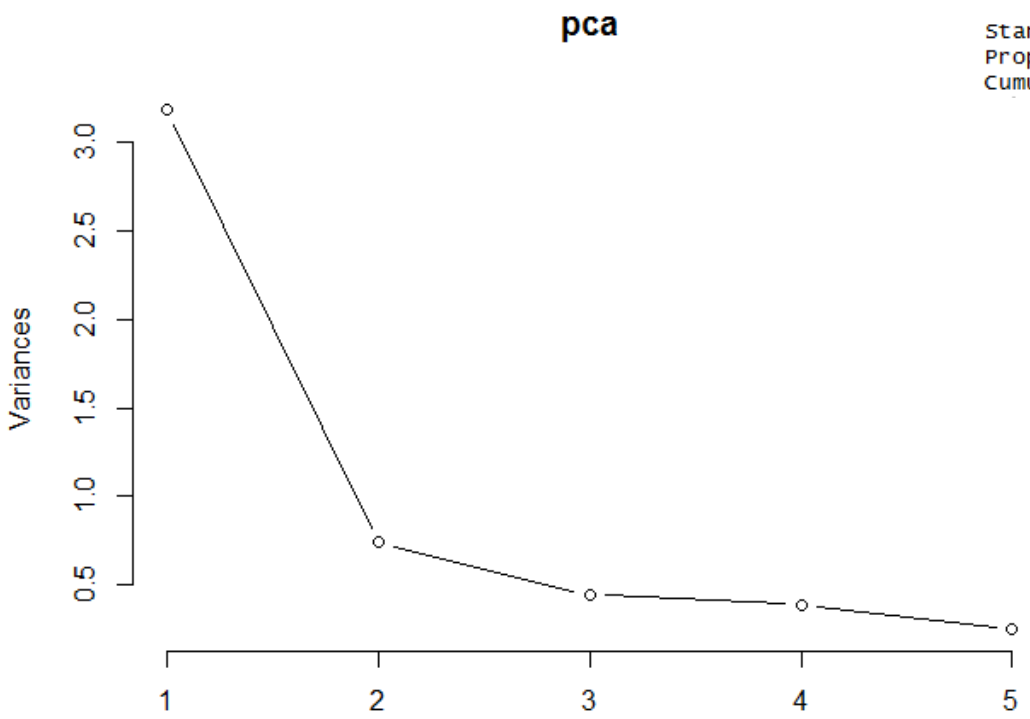
- 총분산 설명하는 비중$(\sum_{i=1}^{m} \lambda_i / \sum_{i=1}^{q} \lambda_i)$이 70%에서 90% 사이에서 선택

- 평균 고유값 $\sum_{i=1}^{q} \lambda_i / q$보다 작은 고유값을 갖는 주성분 제거
  - ✓ 평균고유값=평균분산
  - ✓ 주성분이 상관행렬로 부터 추출된다면 평균분산=1이므로 1보다 작은 고유값 제거
  - ✓ 0.7보다 작은 고유값을 제거하는 것 제안하기도 함

- Scree plot 활용
  - ✓ λ_i 를 i에 대해 점찍은 그림
  - ✓ 팔꿈치에 대응하는 즉, 기울기가 가파른 상태에서 완만한 상태로 변화하는 점에 대응하는 값까지 보유

Example : Open/closed book

```
> plot(pca,type='l')
```




```
> pca=prcomp(data, scale=T)
> summary(pca)
Importance of components:
                          PC1    PC2     PC3     PC4     PC5
Standard deviation     1.7835 0.8600 0.66706 0.62281 0.49658
Proportion of Variance 0.6362 0.1479 0.08899 0.07758 0.04932
Cumulative Proportion  0.6362 0.7841 0.87310 0.95068 1.00000
```

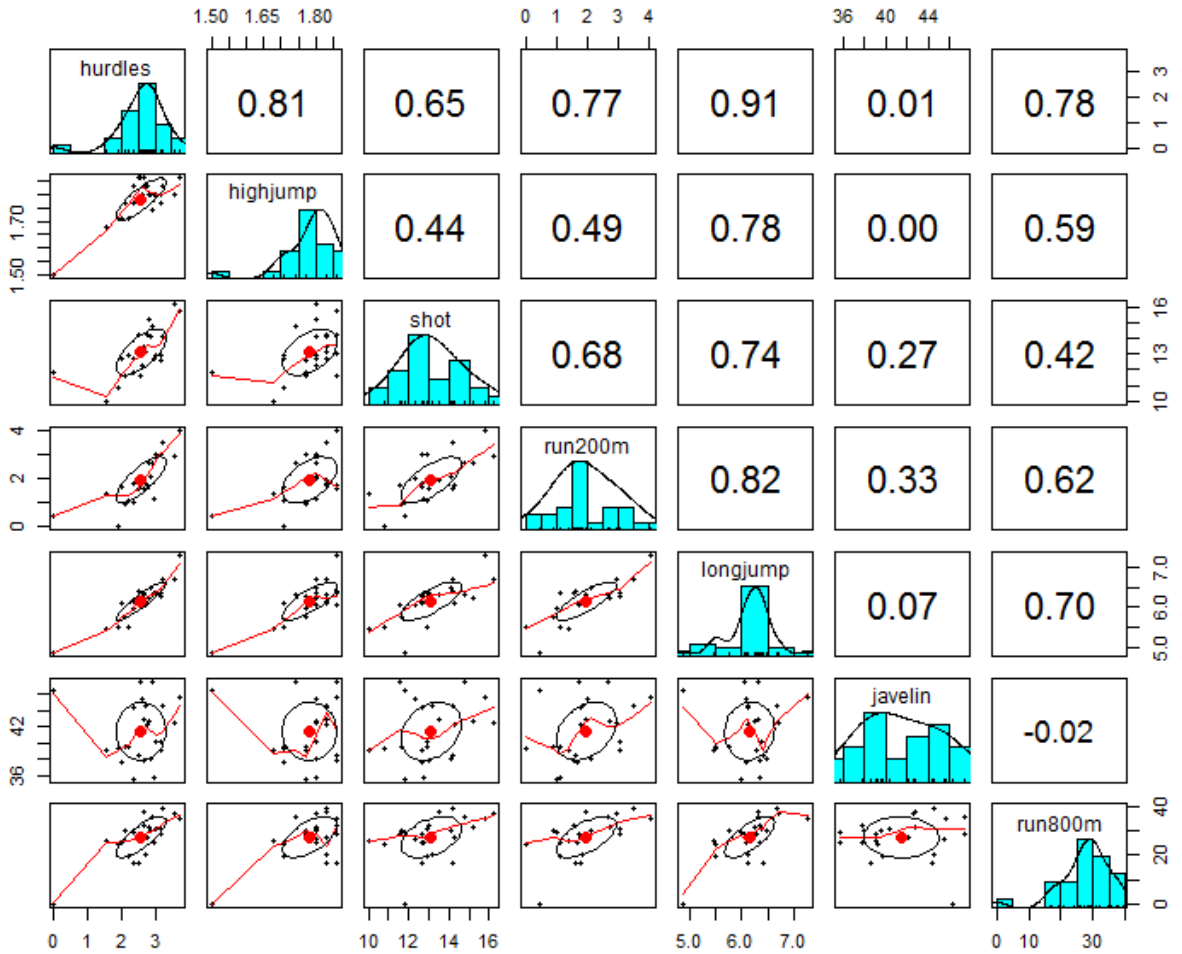
Example: 올림픽 7종 경기 결과

- 1988년 서울올림픽 25명 선수의 여자 7종경기 결과
- 100m 허들(hurdle), 투포환(shot), 높이뛰기(highjump), 200m 달리기(run200m), 멀리뛰기(longjump), 투창(javelin), 800m 달리기(run800m)
- 데이터의 구조를 탐색하고 득점 시스템에 의해 획득된 점수(score)와 어떻게 관련되는지 평가하려함

```
> heptathlon$hurdles <- with(heptathlon, max(hurdles)-hurdles)
> heptathlon$run200m <- with(heptathlon, max(run200m)-run200m)
> heptathlon$run800m <- with(heptathlon, max(run800m)-run800m)
> score <- which(colnames(heptathlon) == "score")
>
> round(cor(heptathlon[,-score]), 2)
         hurdles highjump shot run200m longjump javelin run800m
hurdles     1.00     0.81 0.65    0.77     0.91    0.01    0.78
highjump    0.81     1.00 0.44    0.49     0.78    0.00    0.59
shot        0.65     0.44 1.00    0.68     0.74    0.27    0.42
run200m     0.77     0.49 0.68    1.00     0.82    0.33    0.62
longjump    0.91     0.78 0.74    0.82     1.00    0.07    0.70
javelin     0.01     0.00 0.27    0.33     0.07    1.00   -0.02
run800m     0.78     0.59 0.42    0.62     0.70   -0.02    1.00
```

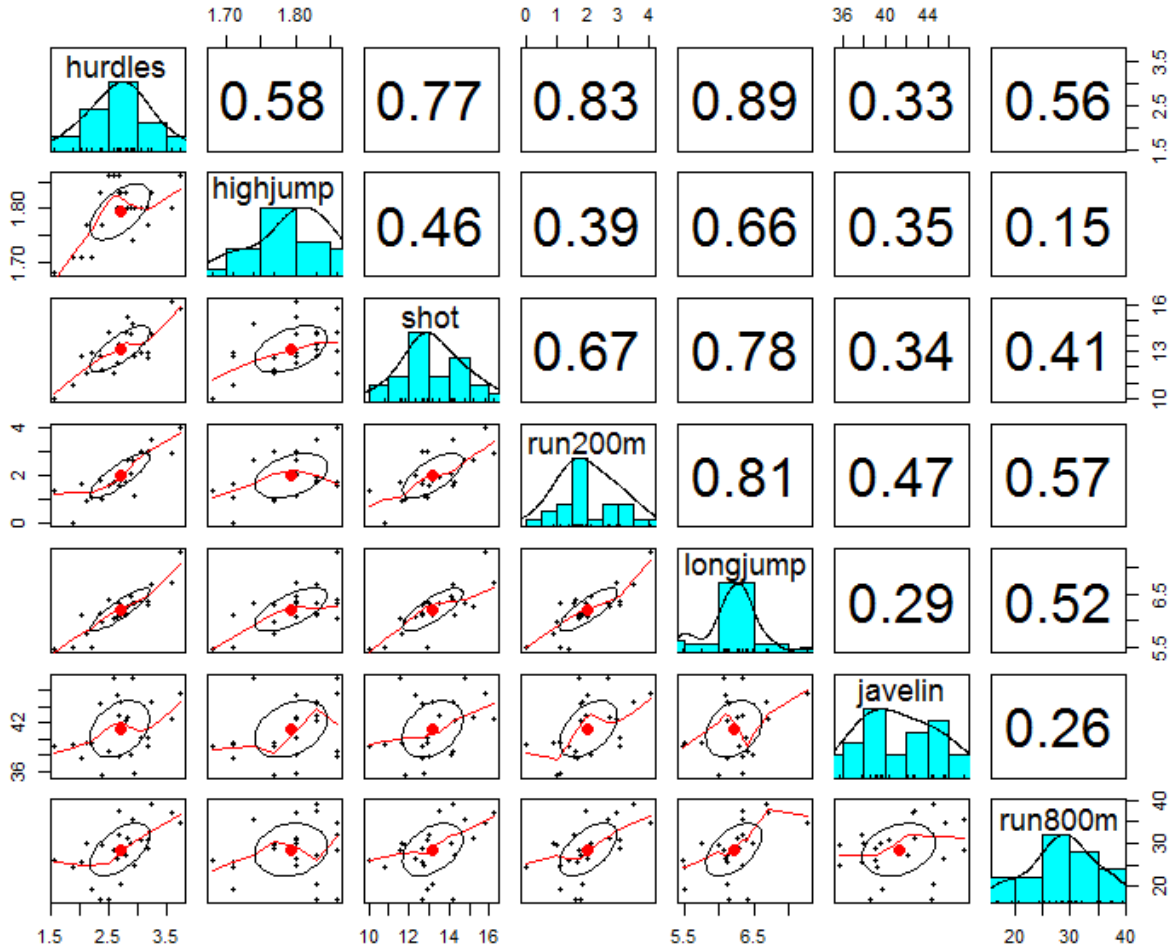
```
library(psych)
pairs.panels(heptathlon[,-score])
```

- 대부분 양의 상관관계
- 투창은 다른 경기 결과와 상관관계가 낮음
- 투창을 제외한 나머지는 대부분 심한 이상치 존재
  - ✓파푸아뉴기니(PNG) 선수



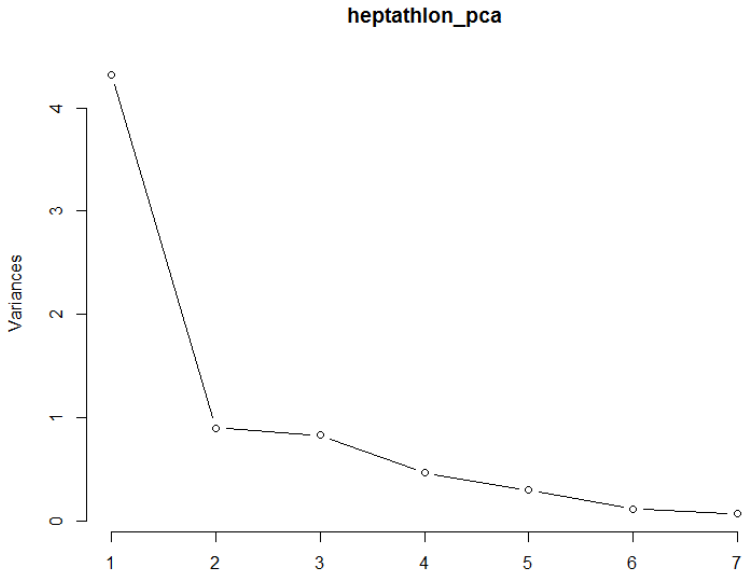
- 이상치 제거 후 산점도와 상관계수 행렬

```
> heptathlon2 <- heptathlon[-grep("PNG", rownames(heptathlon)),]
> round(cor(heptathlon2[,-score]), 2)
         hurdles highjump shot run200m longjump javelin run800m
hurdles     1.00     0.58 0.77    0.83     0.89    0.33    0.56
highjump    0.58     1.00 0.46    0.39     0.66    0.35    0.15
shot        0.77     0.46 1.00    0.67     0.78    0.34    0.41
run200m     0.83     0.39 0.67    1.00     0.81    0.47    0.57
longjump    0.89     0.66 0.78    0.81     1.00    0.29    0.52
javelin     0.33     0.35 0.34    0.47     0.29    1.00    0.26
run800m     0.56     0.15 0.41    0.57     0.52    0.26    1.00
> pairs.panels(heptathlon2[,-score])
```



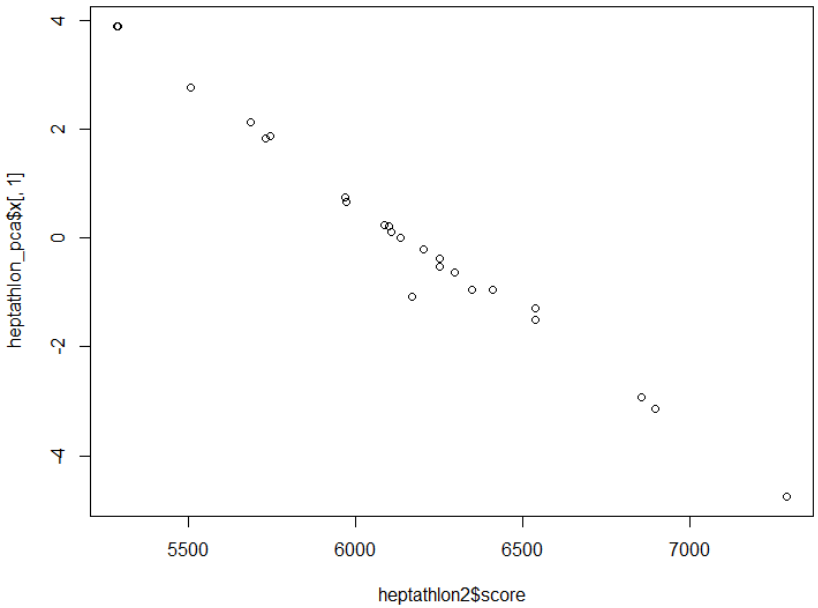
```
> heptathlon_pca <- prcomp(heptathlon2[, -score], scale = TRUE)
> print(heptathlon_pca)
Standard deviations:
[1] 2.08 0.95 0.91 0.68 0.55 0.34 0.26

Rotation:
            PC1    PC2   PC3    PC4    PC5    PC6    PC7
hurdles   -0.45  0.058 -0.17  0.048 -0.199  0.847 -0.070
highjump  -0.31 -0.651 -0.21 -0.557  0.071 -0.090  0.332
shot      -0.40 -0.022 -0.15  0.548  0.672 -0.099  0.229
run200m   -0.43  0.185  0.13  0.231 -0.618 -0.333  0.470
longjump  -0.45 -0.025 -0.27 -0.015 -0.122 -0.383 -0.749
javelin   -0.24 -0.326  0.88  0.060  0.079  0.072 -0.211
run800m   -0.30  0.657  0.19 -0.574  0.319 -0.052  0.077
> summary(heptathlon_pca)
Importance of components:
                          PC1   PC2   PC3    PC4    PC5    PC6     PC7
Standard deviation     2.079 0.948 0.911 0.6832 0.5462 0.3375 0.26204
Proportion of Variance 0.618 0.128 0.119 0.0667 0.0426 0.0163 0.00981
Cumulative Proportion  0.618 0.746 0.865 0.9313 0.9739 0.9902 1.00000
```



- 7개 변수를 2개 주성분점수로 축약할 때 총분산의 74.6% 설명가능
- "팔꿈치" 역시 주성분 개수 2일 때 존재

```
> cor(heptathlon2$score, heptathlon_pca$x[,1])
[1] -0.99
> plot(heptathlon2$score, heptathlon_pca$x[,1])
```




```
> heptathlon_pca$x
                        PC1    PC2    PC3    PC4    PC5    PC6    PC7
Joyner-Kersee (USA) -4.758 -0.140 -0.006  0.2934 -0.362 -0.271 -0.476
John (GDR)          -3.148  0.949 -0.244  0.5492  0.754  0.378 -0.052
Behmer (GDR)        -2.926  0.695  0.622 -0.5547 -0.190 -0.258  0.111
Sablovskaite (URS)  -1.288  0.179  0.251  0.6372  0.604 -0.216  0.531
Choubenkova (URS)   -1.503  0.962  1.781  0.7840  0.590  0.080 -0.301
Schulz (GDR)        -0.958  0.351  0.413 -1.1135  0.715 -0.254  0.038
Fleming (AUS)       -0.953  0.500 -0.265 -0.1402 -0.866  0.037  0.230
Greiner (USA)       -0.633  0.376 -1.140  0.1426  0.208 -0.142 -0.064
Lajbnerova (CZE)    -0.382 -0.712 -0.068  0.0872  0.677  0.250  0.356
Bouraga (URS)       -0.522  0.777 -0.481  0.2837 -1.188  0.399  0.197
Wijnsma (HOL)       -0.218 -0.234 -1.154 -1.2601  0.375 -0.203  0.175
Dimitrova (BUL)     -1.076  0.516 -0.312 -0.1270 -0.920  0.267  0.211
Scheider (SWI)       0.003 -1.447  1.583 -1.2544 -0.205  0.176 -0.039
Braun (FRG)          0.109 -1.636  0.470  0.3626 -0.147  0.261 -0.013
Ruotsalainen (FIN)   0.209 -0.689  1.152 -0.1129 -0.315  0.184 -0.141
Yuping (CHN)         0.233 -1.960 -1.541  0.5983  0.175 -0.502  0.050
Hagger (GB)          0.660 -0.088 -1.797 -0.1824 -0.051  0.551 -0.464
Brown (USA)          0.757 -2.043  0.452  0.4769 -0.382 -0.266 -0.111
Mulliner (GB)        1.881  0.915 -0.359  0.7996 -0.069 -0.733 -0.313
Hautenauve (BEL)     1.828  0.726 -1.049 -0.7118  0.141  0.069 -0.075
Kytola (FIN)         2.118  0.399  0.190 -0.7884  0.418 -0.034  0.121
Geremias (BRA)       2.771  0.035  0.170  1.3856  0.285  0.381  0.346
Hui-Ing (TAI)        3.901  1.202  0.944 -0.0024 -0.671 -0.528  0.094
Jeong-Mi (KOR)       3.897  0.367  0.391 -0.1523  0.425  0.373 -0.411
```

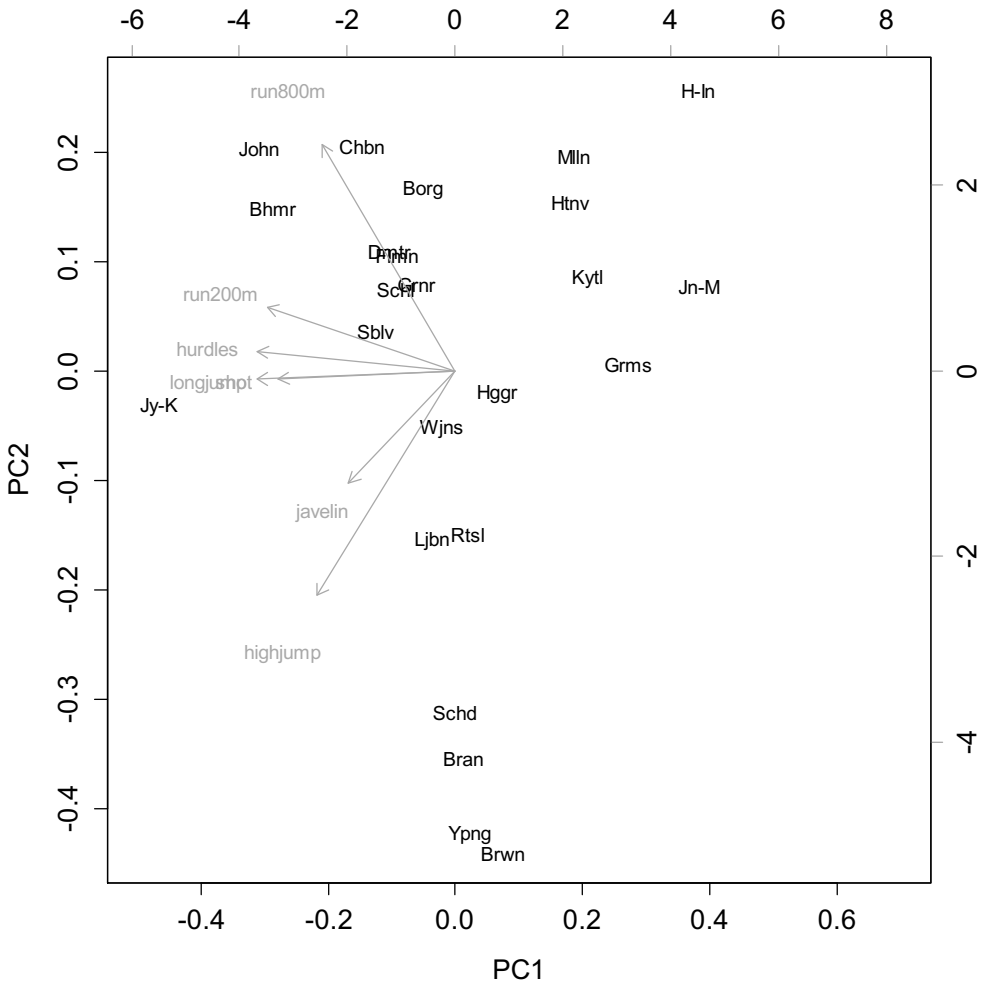
표준점수 시스템에 의한 점수와 첫번째 주성분점수와의 상관계수는 -0.99

행렬도

- 원변수와 주성분 점수와의 관계를 그래프로 표현
- PC1과 PC2의 산점도
  - ✓관측치 번호 혹은 행이름을 표시
- 화살표
  - ✓원변수와 PC간의 상관계수를 표현
  - ✓PC와 평행할 수록 해당 PC에 큰 영향
  - ✓벡터의 길이가 원변수의 분산을 표현

```
> tmp <- heptathlon[, -score]
> rownames(tmp) <- abbreviate(gsub(" \\(.*", "", rownames(tmp)))
> biplot(prcomp(tmp, scale = TRUE), col = c("black", "darkgray"), xlim = + c(-0.5, 0.7), cex = 0.7)
```

- Joyner-Kersee는 hurdle, longjump, shot, run200에서 좋은 성적을 거둠
- run200m, hurdles, longjump, shot은 상관관계가 높음
- javeli과 highjump는 상관관계가 높음
- run800m은 다른 종목들과 비교적 상관관계가 적음
- PC1은 경쟁자들의 전체적인 점수에 의해서 분리
- PC2는 선수들이 각자 잘한 종목을 구분



주성분 점수의 그래프 활용

- 변수의 개수를 줄여 축약된 정보를 한 눈에 볼 수 있게 요약
- 정규성 검정
  - ✓주성분의 Q-Q plot
  - ✓처음 몇 개의 주성분 간의 산점도 확인
- 이상치 탐색
  - ✓주성분의 Boxplot
  - ✓처음 몇 개의 주성분 간의 산점도 확인