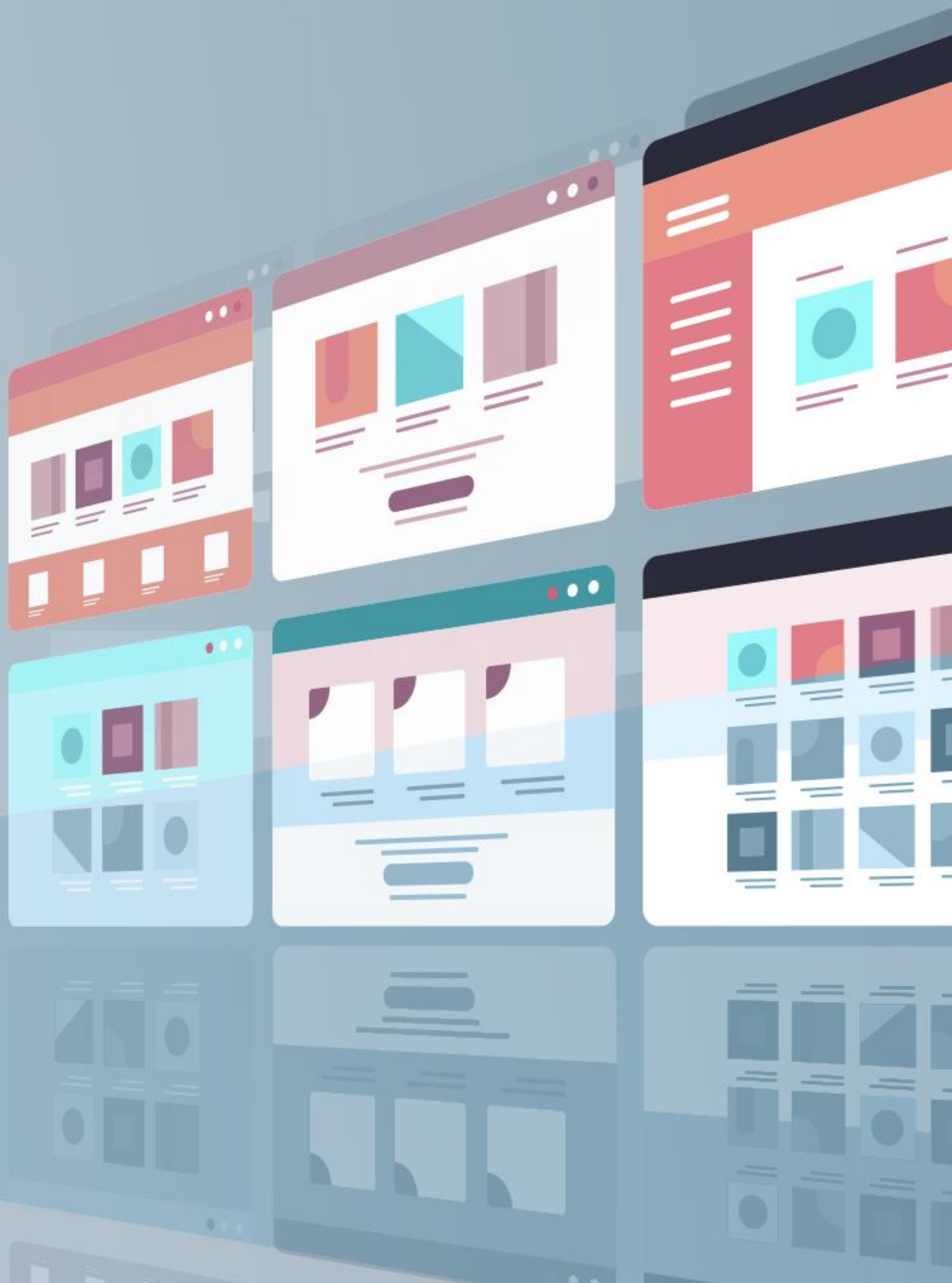
11

데이터분석방법론(1)

Analysis of Collinear Data

통계·데이터과학과 장영재 교수

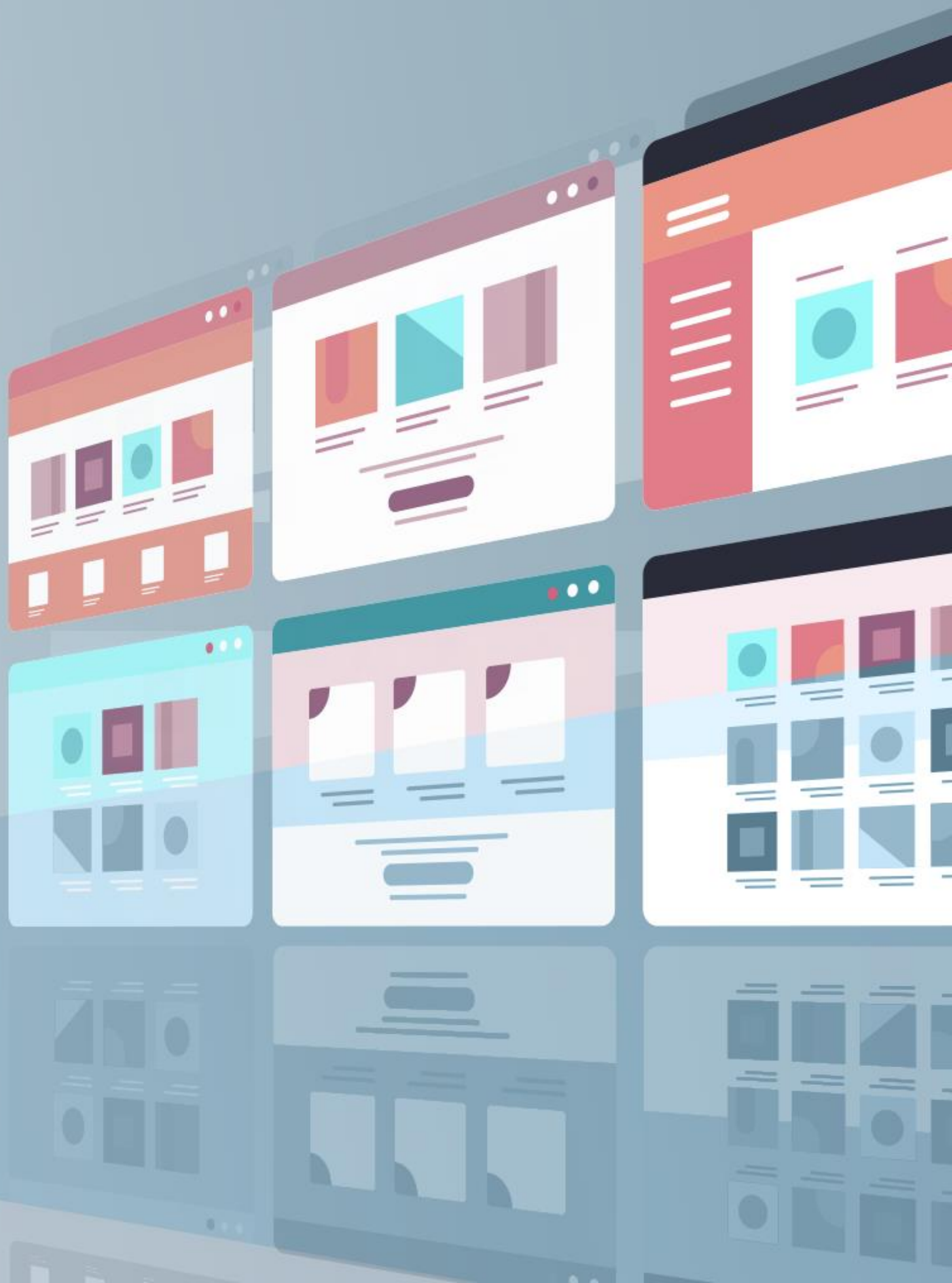


한국방송통신대학교 대학원

학습목차

01

Collinearity



1. Collinearity

- A perfect linear relationship among the regressors in a linear model implies that the least-squares coefficients are not uniquely defined.
- A strong, but less than perfect, linear relationship among the X's causes the least-squares coefficients to be unstable:
  - Coefficient standard errors are large, reflecting the imprecision of estimation of the β's; consequently confidence intervals for the β's are broad.

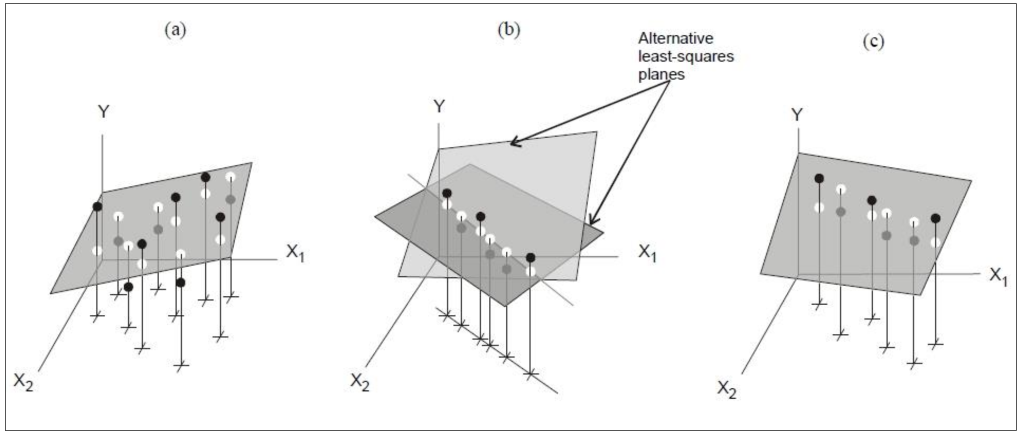


Figure 1. (a) Low correlation between X_1 and X_2 – regression plane well supported; (b) perfect correlation beten X_1 and X_2, showing two of the infinite number of least-squares planes; (c) high but not perfect correlation between X_1 and X_2 – regression plane not well supported.

2. Detecting Collinearity

◆ A perfect linear relationship among the X's

$$c_1X_{i1} + c_2X_{i2} + \cdots + c_kX_{ik} = c_0$$

where $c_1, c_2, \ldots, c_k$ are not all 0:

- When some predictors are linear combinations of others, then X′X is singular, and we have (exact) collinearity.
- The least-squares normal equations do not have a unique solution.
- The sampling variances of the regression coefficients are infinite.

2. Detecting Collinearity

- A less than perfect collinearity:
  - The sampling variance of the least-squares slope coefficient $\boldsymbol{B_j}$ is

$$V(B_j) = \frac{1}{1-R_j^2} \times \frac{\sigma_\varepsilon^2}{(n-1)S_j^2}$$

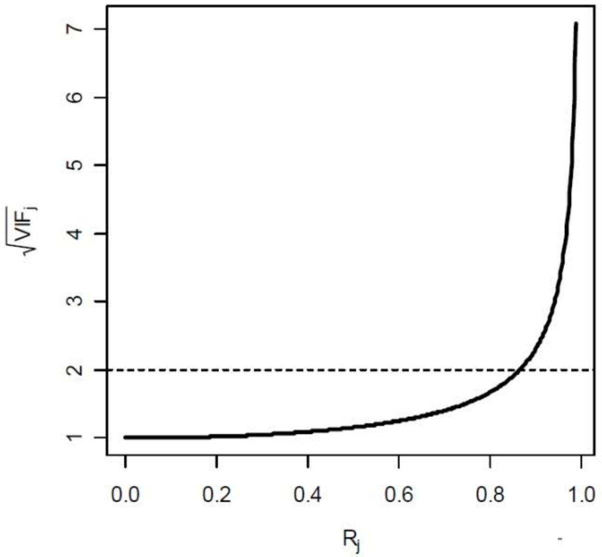
  where

  – R_j^2 is the squared multiple correlation for the regression of $\boldsymbol{X_j}$ on the other X's, and $S_j^2 = \sum(X_{ij} - \bar{X}_j)^2/(n-1)$ is the variance of $\boldsymbol{X_j}$.

  - The term $1/(1-R_j^2)$: the variance-inflation factor (VIF)
    : the impact of collinearity on the precision of $\boldsymbol{B_j}$.
  - The width of the confidence interval for β_j
    : proportional to the square root of the VIF
  - Collinearity leads to imprecise estimate of β.
  - If VIF_j > 5~10, we conclude that there is a collinearity.

2. Detecting Collinearity

- Figure 2 shows that the precision of estimation experiences a significant degradation when Rj approaches .9.




```
> rj= seq(0,1, length=100)
> vifj = 1/(1-rj^2)
> rvifj=sqrt(vifj)
> plot(rj, rvifj, ylab="sqrt(vifj)", type="l")
> abline(h=2, lty=2)
```

Figure 2. The square root of the variance-inflation factor as a function of the multiple correlation for the regression of X_j on the other X's.

3. R Example:

Data : Seat position, size and age of 38 drivers(Univ. of Michigan)

```
> library(faraway)
> data(seatpos)
> seatpos[c(1:10),]
```

```
> seatpos[c(1:10),]
   Age Weight HtShoes    Ht Seated  Arm Thigh  Leg hipcenter
1   46    180   187.2 184.9   95.2 36.1  45.3 41.3  -206.300
2   31    175   167.5 165.5   83.8 32.9  36.5 35.9  -178.210
3   23    100   153.6 152.2   82.9 26.0  36.6 31.0   -71.673
4   19    185   190.3 187.4   97.3 37.4  44.1 41.0  -257.720
5   23    159   178.0 174.1   93.9 29.5  40.1 36.9  -173.230
6   47    170   178.7 177.0   92.4 36.0  43.2 37.4  -185.150
7   30    137   165.7 164.6   87.7 32.5  35.6 36.2  -164.750
8   28    192   185.3 182.7   96.9 35.8  39.9 43.1  -270.920
9   23    150   167.6 165.0   91.4 29.4  35.5 33.4  -151.780
10  29    120   161.2 158.7   85.2 26.6  31.0 32.8  -113.880
```

3. R Example:

```
> g = lm(hipcenter~ . , seatpos)
> summary(g)
```

```
> g = lm(hipcenter~ ., seatpos)
> summary(g)

Call:
lm(formula = hipcenter ~ ., data = seatpos)

Residuals:
    Min      1Q  Median      3Q     Max
-73.827 -22.833  -3.678  25.017  62.337

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 436.43213  166.57162   2.620   0.0138 *
Age           0.77572    0.57033   1.360   0.1843
Weight        0.02631    0.33097   0.080   0.9372
HtShoes      -2.69241    9.75304  -0.276   0.7845
Ht            0.60134   10.12987   0.059   0.9531
Seated        0.53375    3.76189   0.142   0.8882
Arm          -1.32807    3.90020  -0.341   0.7359
Thigh        -1.14312    2.66002  -0.430   0.6706
Leg          -6.43905    4.71386  -1.366   0.1824
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 37.72 on 29 degrees of freedom
Multiple R-squared: 0.6866,     Adjusted R-squared: 0.6001
F-statistic:  7.94 on 8 and 29 DF,  p-value: 1.306e-05
```

p-value for the F-statistics is very small, but none of the individual predictors is significant. This models shows the signs of collinearity.

3. R Example:

- Take a look at the pairwise correlations.

```
> round(cor(seatpos),3)
```

```
> round(cor(seatpos),3)
             Age Weight HtShoes     Ht Seated    Arm  Thigh    Leg hipcenter
Age        1.000  0.081  -0.079 -0.090 -0.170  0.360  0.091 -0.042     0.205
Weight     0.081  1.000   0.828  0.829  0.776  0.698  0.573  0.784    -0.640
HtShoes   -0.079  0.828   1.000  0.998  0.930  0.752  0.725  0.908    -0.797
Ht        -0.090  0.829   0.998  1.000  0.928  0.752  0.735  0.910    -0.799
Seated    -0.170  0.776   0.930  0.928  1.000  0.625  0.607  0.812    -0.731
Arm        0.360  0.698   0.752  0.752  0.625  1.000  0.671  0.754    -0.585
Thigh      0.091  0.573   0.725  0.735  0.607  0.671  1.000  0.650    -0.591
Leg       -0.042  0.784   0.908  0.910  0.812  0.754  0.650  1.000    -0.787
hipcenter  0.205 -0.640  -0.797 -0.799 -0.731 -0.585 -0.591 -0.787     1.000
```

: There are several large correlation between predictors.

- Take a look at the pairwise correlations.

```
> x = model.matrix(g)[,-1]
> vif(x)
      Age    Weight    HtShoes         Ht    Seated       Arm     Thigh
 1.997931  3.647030 307.429378 333.137832  8.951054  4.496368  2.762886
      Leg
 6.694291
```

: Much VIF in HtShoes and Ht.

3. R Example:

- One cure for collinearity

Examine the full correlation matrix and consider just the correlations of the length variables.

> round(cor(x[,3:8]), 2)

```
> round(cor(x[,3:8]),2)
        HtShoes   Ht Seated  Arm Thigh  Leg
HtShoes    1.00 1.00   0.93 0.75  0.72 0.91
Ht         1.00 1.00   0.93 0.75  0.73 0.91
Seated     0.93 0.93   1.00 0.63  0.61 0.81
Arm        0.75 0.75   0.63 1.00  0.67 0.75
Thigh      0.72 0.73   0.61 0.67  1.00 0.65
Leg        0.91 0.91   0.81 0.75  0.65 1.00
```

: These six variables are strongly correlated each other – any one of them might do a good job of representing the other.

3. R Example:

```
> g2 = lm(hipcenter~Age+Weight+Ht, seatpos)
> summary(g2)

Call:
lm(formula = hipcenter ~ Age + Weight + Ht, data = seatpos)

Residuals:
    Min      1Q  Median      3Q     Max 
-91.526 -23.005   2.164  24.950  53.982 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 528.297729 135.312947   3.904 0.000426 ***
Age           0.519504   0.408039   1.273 0.211593    
Weight        0.004271   0.311720   0.014 0.989149    
Ht           -4.211905   0.999056  -4.216 0.000174 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 36.49 on 34 degrees of freedom
Multiple R-squared: 0.6562,	Adjusted R-squared: 0.6258 
F-statistic: 21.63 on 3 and 34 DF,  p-value: 5.125e-08
```

: The fit is very similar to the original one in terms of R^2, but with much fewer predictors

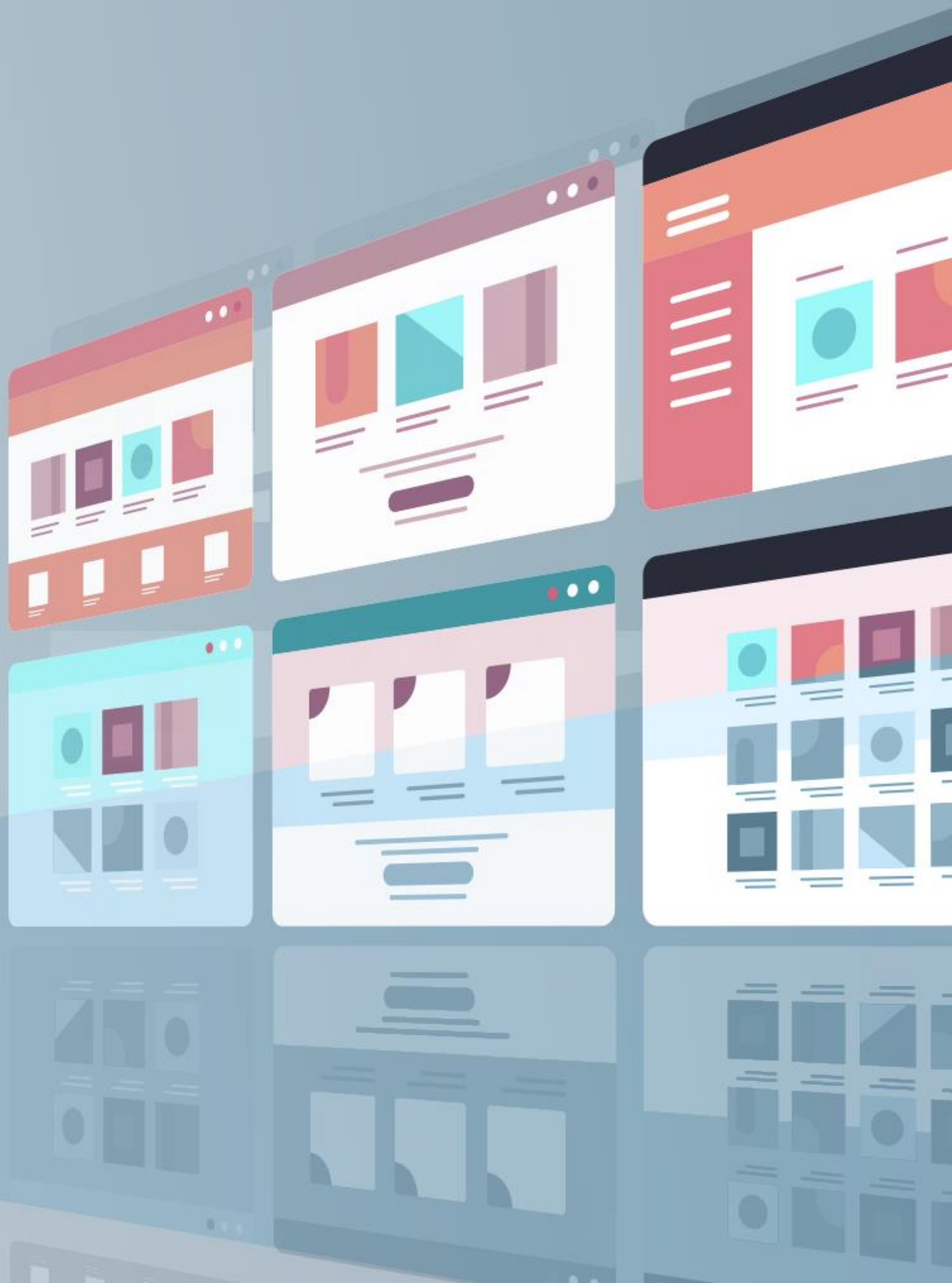
: To keep all variables, we can use ridge regression.

4. Coping With Collinearity

- When X_1 and X_2 are strongly collinear
  - The data contain little information about the impact of X1 on Y holding X2 constant, because there is little variation in X1 when X2 is fixed.
  - Of course, the same is true for X_2 fixing X_1.
- Strategies for dealing with collinear data
  - None magically extracts nonexistent information from the data.
  - Rather, the research problem is redefined, often subtly and implicitly.
  - Sometimes the redefinition is reasonable; usually it is not.
- The ideal solution to the problem of collinearity
  - To collect new data in such a manner that the problem is avoided,
    for example, by experimental manipulation of the X's
    : This solution is rarely practical.

02

Introduction to PCA



1. The model

- The regression model is

$$Y = X\beta + \varepsilon \quad (1)$$

- Y is an n×1 vector of observations on the response variable,
- $X = (X_{(1)}, \cdots, X_{(p)})$ is an n×p matrix of n observations on p explanatory variables
- β is a p×1 vector

- Assumption :

$$E(\varepsilon) = 0, E(\varepsilon'\varepsilon) = \sigma^2 I$$

X and Y have been centered and scaled so that X′X and X′Y are matrices of correlation coefficients.

2. Principal components

- There exists a matrix, C, satisfying

$$C'(X'X)C = \Lambda \quad \text{and} \quad C'C = CC' = I \qquad (2)$$

- Λ is a diagonal matrix with the ordered eigenvalues of $X'X$ on the diagonal.
- The eigenvalues are denoted by $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_p$.
- The columns of C are the normalized eigenvectors corresponding to $\lambda_1, \cdots, \lambda_p$.

- C may be used to calculate a new set of explanatory variables, namely C may be used to calculate a new set of explanatory variables, namely

$$\left(W_{(1)}, W_{(2)}, \cdots, W_{(p)}\right) = W = XC = \left(W_{(1)}, \cdots, W_{(p)}\right)C \quad (3)$$

That are linear functions of the original explanatory variables. The W's are referred to as principal components.

3. The model in terms of principal components

- The regression model of Equation (1) can be restated in terms of the principal components as

$$Y = W\alpha + u \qquad (4)$$

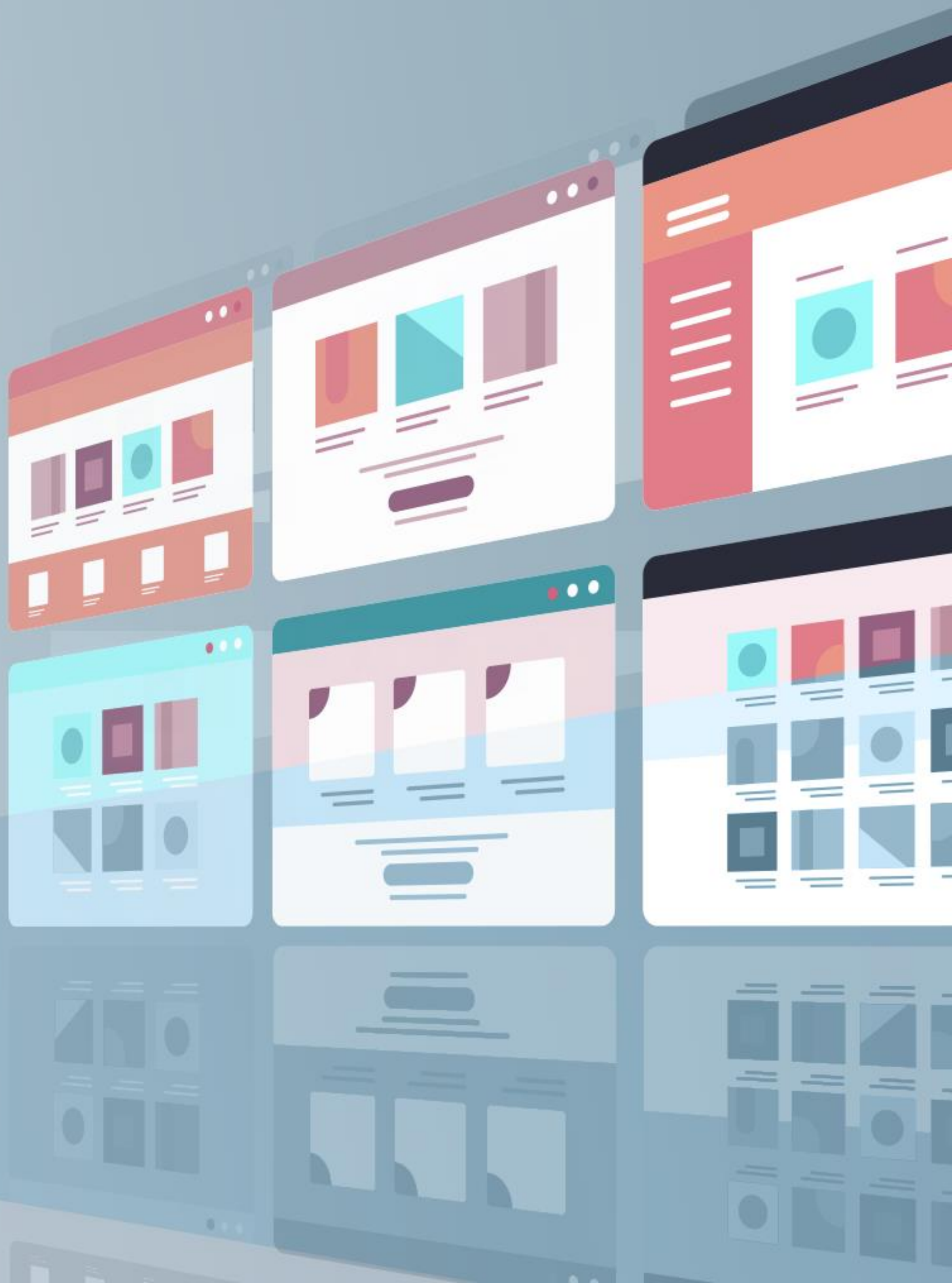
where W=XC and α=$C'\beta$.

- $W'_{(i)}W_{(j)} = 0$ for i≠j and $W'_{(i)}W_{(i)} = \lambda_i$. The $\lambda's$ may be viewed as sample variances of the principal components

- When $\lambda_i = 0$, an exact linear dependence exists among the explanatory variables, and when λ_i is small (approximately equal to zero) there is an approximate linear relationship among the explanatory variables.

03

Example of PCA



1. R Example of PCA

- Data

예) heptathlon in R "HSAUR" Package.

Records of athletes in 1988 Seoul Olympics in

hurdles(110m), highjump(high jump), shot(투포환),

run200m(200m), longjump(long jump), javelin(창던지기),

run800m(800m), score

2. Descriptive statistics

```
> library(HSAUR)
> data(heptathlon)
> head(heptathlon)
                      hurdles highjump  shot run200m longjump javelin run800m score
Joyner-Kersee (USA)     12.69     1.86 15.80   22.56     7.27   45.66  128.51  7291
John (GDR)              12.85     1.80 16.23   23.65     6.71   42.56  126.12  6897
Behmer (GDR)            13.20     1.83 14.20   23.10     6.68   44.54  124.20  6858
Sablovskaite (URS)      13.61     1.80 15.23   23.92     6.25   42.78  132.24  6540
Choubenkova (URS)       13.51     1.74 14.76   23.93     6.32   47.46  127.90  6540
Schulz (GDR)            13.75     1.83 13.50   24.65     6.33   42.82  125.79  6411
> summary(heptathlon)
    hurdles          highjump          shot          run200m         longjump    
 Min.   :12.69   Min.   :1.500   Min.   :10.00   Min.   :22.56   Min.   :4.880  
 1st Qu.:13.47   1st Qu.:1.770   1st Qu.:12.32   1st Qu.:23.92   1st Qu.:6.050  
 Median :13.75   Median :1.800   Median :12.88   Median :24.83   Median :6.250  
 Mean   :13.84   Mean   :1.782   Mean   :13.12   Mean   :24.65   Mean   :6.152  
 3rd Qu.:14.07   3rd Qu.:1.830   3rd Qu.:14.20   3rd Qu.:25.23   3rd Qu.:6.370  
 Max.   :16.42   Max.   :1.860   Max.   :16.23   Max.   :26.61   Max.   :7.270  
    javelin         run800m          score     
 Min.   :35.68   Min.   :124.2   Min.   :4566  
 1st Qu.:39.06   1st Qu.:132.2   1st Qu.:5746  
 Median :40.28   Median :134.7   Median :6137  
 Mean   :41.48   Mean   :136.1   Mean   :6091  
 3rd Qu.:44.54   3rd Qu.:138.5   3rd Qu.:6351  
 Max.   :47.50   Max.   :163.4   Max.   :7291  
> 
```

3. Need to transform the data

◆ Transformation

For hurdles, run200m, and run800m, transform the data because the smaller the value, the better the score.

```
> heptathlon$hurdles = max(heptathlon$hurdles) - heptathlon$hurdles
> heptathlon$run200m = max(heptathlon$run200m) - heptathlon$run200m
> heptathlon$run800m = max(heptathlon$run800m) - heptathlon$run800m
> heptathlon
```

	hurdles	highjump	shot	run200m	longjump	javelin	run800m	score
Joyner-Kersee (USA)	3.73	1.86	15.80	4.05	7.27	45.66	34.92	7291
John (GDR)	3.57	1.80	16.23	2.96	6.71	42.56	37.31	6897
Behmer (GDR)	3.22	1.83	14.20	3.51	6.68	44.54	39.23	6858
Sablovskaite (URS)	2.81	1.80	15.23	2.69	6.25	42.78	31.19	6540
Choubenkova (URS)	2.91	1.74	14.76	2.68	6.32	47.46	35.53	6540
Schulz (GDR)	2.67	1.83	13.50	1.96	6.33	42.82	37.64	6411
Fleming (AUS)	3.04	1.80	12.88	3.02	6.37	40.28	30.89	6351
Greiner (USA)	2.87	1.80	14.13	2.13	6.47	38.00	29.78	6297
Lajbnerova (CZE)	2.79	1.83	14.28	1.75	6.11	42.20	27.38	6252
Bouraga (URS)	3.17	1.77	12.62	3.02	6.28	39.06	28.69	6252
Wijnsma (HOL)	2.67	1.86	13.01	1.58	6.34	37.86	31.94	6205
Dimitrova (BUL)	3.18	1.80	12.88	3.02	6.37	40.28	30.89	6171
Scheider (SWI)	2.57	1.86	11.58	1.74	6.05	47.50	28.50	6137
Braun (FRG)	2.71	1.83	13.16	1.83	6.12	44.58	20.61	6109
Ruotsalainen (FIN)	2.63	1.80	12.32	2.00	6.08	45.44	26.37	6101
Yuping (CHN)	2.49	1.86	14.21	1.61	6.40	38.60	16.76	6087
Hagger (GB)	2.95	1.80	12.75	1.14	6.34	35.76	24.95	5975
Brown (USA)	2.35	1.83	12.69	1.78	6.13	44.34	17.00	5972
Mulliner (GB)	2.03	1.71	12.68	1.69	6.10	37.76	25.41	5746
Hautenauve (BEL)	2.38	1.77	11.81	1.00	5.99	35.68	29.53	5734
Kytola (FIN)	2.11	1.77	11.66	0.92	5.75	39.48	30.08	5686
Geremias (BRA)	2.19	1.71	12.95	1.11	5.50	39.64	19.41	5508

4. Principal Component Analysis

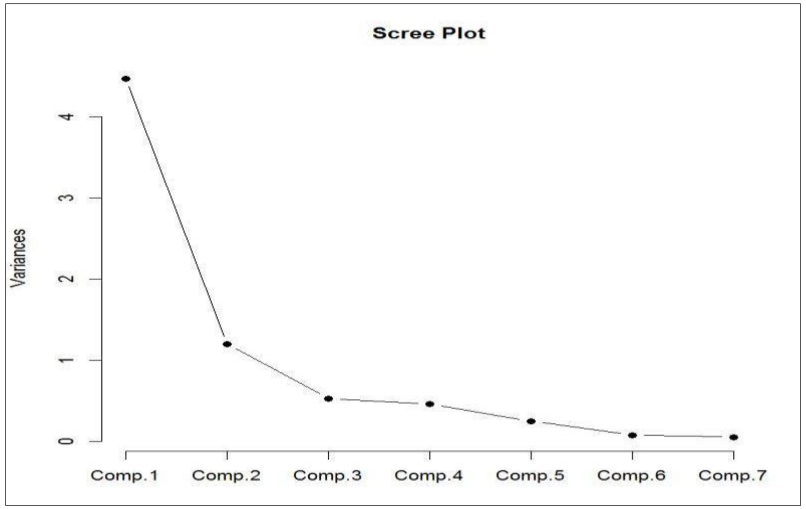
```
> library(stats)
> hep.data = heptathlon[,-8]
> heptathlon.pca = princomp(hep.data, cor=T, scores=T)
> names(heptathlon.pca)
[1] "sdev"     "loadings" "center"   "scale"    "n.obs"    "scores"   "call"
> heptathlon.pca
Call:
princomp(x = hep.data, cor = T, scores = T)

Standard deviations:
   Comp.1    Comp.2    Comp.3    Comp.4    Comp.5    Comp.6    Comp.7 
2.1119364 1.0928497 0.7218131 0.6761411 0.4952441 0.2701029 0.2213617 

 7  variables and  25 observations.
> |
```

```
> summary(heptathlon.pca)
Importance of components:
                          Comp.1    Comp.2     Comp.3     Comp.4     Comp.5     Comp.6
Standard deviation     2.1119364 1.0928497 0.72181309 0.67614113 0.49524412 0.27010291
Proportion of Variance 0.6371822 0.1706172 0.07443059 0.06530955 0.03503811 0.01042223
Cumulative Proportion  0.6371822 0.8077994 0.88222998 0.94753952 0.98257763 0.99299986
                            Comp.7
Standard deviation     0.221361710
Proportion of Variance 0.007000144
Cumulative Proportion  1.000000000
> eig.val = heptathlon.pca$sdev^2
> eig.val
    Comp.1     Comp.2     Comp.3     Comp.4     Comp.5     Comp.6     Comp.7 
4.46027516 1.19432056 0.52101413 0.45716683 0.24526674 0.07295558 0.04900101 
> |
```

4. Principal Component Analysis

◆ Scree plot and coefficients of principal components

```
> screeplot(heptathlon.pca, type="lines", pch=19, main="Scree Plot")
> heptathlon.pca$loadings[,1:2]
             Comp.1      Comp.2
hurdles  -0.4528710  0.15792058
highjump -0.3771992  0.24807386
shot     -0.3630725 -0.28940743
run200m  -0.4078950 -0.26038545
longjump -0.4562318  0.05587394
javelin  -0.0754090 -0.84169212
run800m  -0.3749594  0.22448984
> |
```



A scree plot is drawn in order of the size of the eigenvalues of the principal components using the screeplot() function. It can be seen that there are two principal components with eigenvalues greater than 1.

4. Principal Component Analysis

◆ First and Second Principal components

$$PC_1 = -0.453 \times hurdles - 0.377 \times highjump - 0.363 shot + \ldots - 0.075 \times javelin - 0.375 \times run800m$$

$$PC_2 = 0.158 \times hurdles + 0.248 \times highjump - 0.289 shot \ldots - 0.842 \times javelin + 0.224 \times run800m$$

- Considering that the absolute values of all variables except javelin(창던지기) have large absolute values, the first principal component can be said to be a component that represents the overall level of physical strength.

- The second principal component can be identified as a component closely related to javelin, given that the coefficient of javelin(창던지기) has a relatively large absolute value compared to other variables.

다음시간 안내

12

Selection of Variables in Regression Equation