제15강: Factor Analysis & PCA

금융 통계 및 시계열 분석

TRADE INFORMATIX

2014년 2월 28일

목차

팩터 모형 (Factor Model)

- 복수의 주식 수익률 시계열이 몇 개의 공통된 원인 (common factor) 시계열에 의존
- 팩터 모형 유형
  - Macroeconomic Factor Model
    - 거시경제 지표를 common factor로 사용
    - GDP 증가율, 이자율, 인플레이션, 실업률 등
  - Fundamental Factor Model
    - 개별 회사의 펀더멘탈 지표를 common factor로 사용
    - 회사 크기, book value, market value, 섹터 등
  - Stochastic Factor Model
    - 관측되지 않는 common factor를 통계적으로 산출
- 팩터 모형의 활용
  - 리스크 관리
  - 포트폴리오 최적화
  - 복수 시계열 예측

팩터 모형

❑ 선형 수익률 모형

$$r_{it} \quad = \quad \alpha_i + \beta_{i1} f_{1t} + \cdots \beta_{im} f_{mt} + e_{it}$$

- $t = 1, \ldots, T$: 시간 index
- $i = 1, \ldots, k$: 종목 index
- $j = 1, \ldots, m$: 팩터 index
- r_{it} : 수익률 시계열
- α_i: intercept
- f_{jt}: factor
- β_{it}: factor loading
- e_i: specific factor

팩터 모형의 행렬 표현 1

❑ cross-section 행렬 표현

$$\begin{aligned} r_t &= \alpha + \beta f_t + e_t \\ \begin{pmatrix} r_{1t} \\ r_{2t} \\ \vdots \\ r_{kt} \end{pmatrix} &= \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_k \end{pmatrix} + \begin{pmatrix} \beta_{11} & \beta_{12} & \cdots & \beta_{1m} \\ \beta_{21} & \beta_{22} & \cdots & \beta_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{k1} & \beta_{k1} & \cdots & \beta_{km} \end{pmatrix} \begin{pmatrix} f_{1t} \\ f_{2t} \\ \vdots \\ f_{mt} \end{pmatrix} + \begin{pmatrix} e_{1t} \\ e_{2t} \\ \vdots \\ e_{mt} \end{pmatrix} \end{aligned}$$

❑ time series 행렬 표현

$$\begin{aligned} R_i &= \alpha_i 1_T + F\beta_i^T + E_i \\ \begin{pmatrix} r_{i1} \\ r_{i2} \\ \vdots \\ r_{iT} \end{pmatrix} &= \alpha_i \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} + \begin{pmatrix} f_{11} & f_{21} & \cdots & f_{m1} \\ f_{12} & f_{22} & \cdots & f_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ f_{1T} & f_{2T} & \cdots & f_{mT} \end{pmatrix} \begin{pmatrix} \beta_{11} \\ \beta_{12} \\ \vdots \\ \beta_{1m} \end{pmatrix} + \begin{pmatrix} e_{i1} \\ e_{i2} \\ \vdots \\ e_{iT} \end{pmatrix} \end{aligned}$$

팩터 모형의 행렬 표현 2

❑ Augmented 행렬 표현

$$r_t = \xi g_t + e_t = \begin{pmatrix} \alpha & \beta \end{pmatrix} \begin{pmatrix} 1 \\ f_t \end{pmatrix} + e_t$$

$$\begin{aligned} R &= G\xi^T + E \\ \begin{pmatrix} r_1^T \\ r_2^T \\ \vdots \\ r_T^T \end{pmatrix} &= \begin{pmatrix} g_1^T \\ g_2^T \\ \vdots \\ g_T^T \end{pmatrix} \begin{pmatrix} \alpha^T \\ \beta^T \end{pmatrix} + \begin{pmatrix} e_1^T \\ e_2^T \\ \vdots \\ e_T^T \end{pmatrix} \\ &= \begin{pmatrix} 1 & f_1^T \\ 1 & f_2^T \\ \vdots & \\ 1 & f_T^T \end{pmatrix} \begin{pmatrix} \alpha^T \\ \beta^T \end{pmatrix} + \begin{pmatrix} e_1^T \\ e_2^T \\ \vdots \\ e_T^T \end{pmatrix} \end{aligned}$$

팩터 모형 가정

- Common Factor는 m차원 정상신호

$$
\begin{aligned}
E[f_t] &= \mu_f \\
\text{Cov}[f_t] &= \sigma_f
\end{aligned}
$$

- Specific Factor는 서로 독립인 k차원 White Noise

$$
\begin{aligned}
E[e_{it}] &= 0 \ \text{ for all } i \text{ and } t \\
\text{Cov}[e_{it}, e_{js}] &= D = \begin{cases} \sigma_i^2 & \text{if } i = j \text{ and } t = s \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
$$

- Specific Factor와 Common Factor는 서로 독립

$$
\text{Cov}[f_{it}, e_{js}] = 0 \ \text{ for all } i, j, t, s
$$

팩터모형의 추정문제

- ❑ factor가 관측 가능한 경우에는 단순히 factor loading을 추정하는 문제
- ❑ factor가 관측 불가능한 경우에는 주어진 beta 값을 이용하여 추정하거나 통계적인 factor 추정
- ❑ Macroeconomic Factor Model
  - ▶ 거시경제지표를 factor로 사용
  - ▶ factor가 관측 가능하므로 단순 factor loading 추정문제
- ❑ Fundamental Factor Model
  - ▶ 펀더멘털지표는 팩터모형 가정에 맞지않음
  - ▶ 따라서 펀더멘털지표를 factor loading으로 가정하여 반대로 factor를 추정
  - ▶ 팩터추정 문제는 파라미터 추정이 아닌 state 시계열 추정문제
- ❑ PCA (Principal Component Analysis)
  - ▶ 다수의 전체 수익률 시계열의 움직임을 통계적으로 가장 잘 설명할 수 있는 소수의 시계열 common factor 추정
  - ▶ factor와 factor loading을 동시에 추정

Macroeconomic Factor Model

- 관측 가능한 거시경제지표를 Common Factor로 사용
- Multivariate Linear Regression 문제

$$\hat{\xi}^T = (G^T G)^{-1}(G^T R)$$

- Residual = Specific Factor

$$\hat{E} = R - G\hat{\xi}^T$$

- Residual Covariance matrix

$$\hat{D} = \text{diag}(\hat{\sigma}_1^2, \hat{\sigma}_2^2, \cdots, \hat{\sigma}_k^2)$$

- 개별 specific factor의 크기 $\hat{\sigma}_i^2$ 는 $\hat{E}^T \hat{E}$ 의 (i, i) 번째 대각 원소
- $\hat{E}^T \hat{E}$ 의 비대각원소 (off-diagonal element) 의 값이 0가 아니면 모형 가정에 오류

Market Model

- 대표적 Single Factor Macroeconomic Factor Model
- 시장 초과수익률을 개별 종목 초과수익률에 대한 common factor로 표현

$$r_{it} = \alpha_i + \beta_i r_{mt} + e_{it}$$

  - r_{it} : 개별종목의 초과수익률
  - r_{mt} : 시장의 초과수익률
  - α_i : 개별종목의 alpha
  - β_i : 개별종목의 beta

Market Model의 예

- S&P과 시총상위 13종목의 월간수익률 (1990.1-2003.12)
- 무위험이자율 : Treasury 3개월

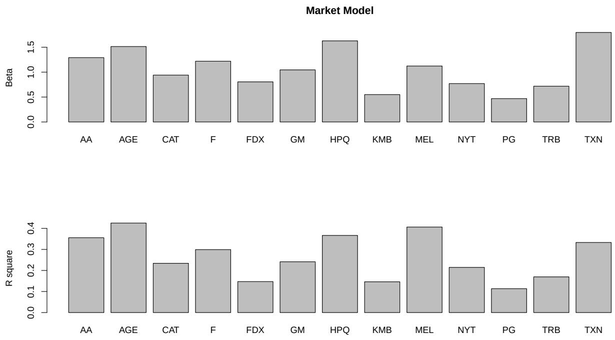
```
> library("FinTS")
> data(m.fac9003)
> xmtx <- cbind(rep(1,168), as.matrix(m.fac9003[,14]))
> colnames(xmtx) <- c("Alpha", "Beta")
> head(xmtx)

  Alpha  Beta
1     1 -7.52
2     1  0.21
3     1  1.77
4     1 -3.34
5     1  8.55
6     1 -1.53

> rtn <- as.matrix(m.fac9003[,1:13])
> head(rtn)

      AA    AGE    CAT     F    FDX    GM   HPQ    KMB    MEL    NYT    PG    TRB   TXN
1 -16.40 -12.17  -4.44 -0.06  -2.28 -2.12 -6.19 -11.01 -10.77  -6.30 -8.89 -13.04 -7.61
2   4.04   4.95   8.84  6.02  10.47  8.97 -4.01  -5.20   0.34  -4.62 -0.84  -0.37  4.97
3   0.12  13.08   0.17  2.06  10.84  1.57  5.67   3.21  -0.17  -0.66  5.41   2.36  2.69
4  -4.28 -11.06   0.25 -5.67  -2.44 -4.19 -5.29  -0.65  -2.20 -10.60  4.26  -7.98 -6.85
5   5.81  19.70   8.52  3.89 -16.17 10.94  8.81   8.83  11.85  11.59 16.35   8.82 22.88
6  -4.05  -1.44 -22.10 -5.79  -2.81 -2.70 -1.47   1.55  -7.76  -0.12  4.80  -0.64 -5.87
```

Market Model의 예

```
> xit.hat <- solve(t(xmtx) %*% xmtx, t(xmtx) %*% rtn)
> E.hat <- rtn - xmtx %*% xit.hat
> D.hat <- diag(crossprod(E.hat)/(168-2))
> sigma.hat <- sqrt(D.hat)
> r.square <- 1 - diag(t(E.hat)%*%E.hat)/diag(t(rtn)%*%rtn)
> t(rbind(xit.hat, sigma.hat, r.square))

        Alpha      Beta sigma.hat  r.square
AA  0.5491240 1.2915911  7.694054 0.3555623
AGE 0.7218061 1.5141359  7.807465 0.4252218
CAT 0.8393521 0.9406928  7.724468 0.2340053
F   0.4543643 1.2192453  8.240771 0.2990748
FDX 0.7995790 0.8051166  8.853854 0.1472167
GM  0.1982025 1.0457019  8.130114 0.2413721
HPQ 0.6835681 1.6279512  9.469272 0.3664564
KMB 0.5463020 0.5498052  6.070099 0.1462376
MEL 0.8849263 1.1228708  6.120035 0.4063423
NYT 0.4904120 0.7706495  6.590364 0.2146277
PG  0.8880914 0.4688034  6.458878 0.1133911
TRB 0.6512465 0.7178808  7.215148 0.1696505
TXN 1.4388867 1.7964117 11.473988 0.3329717
```

Market Model의 예

```
> layout(matrix(1:2))
> barplot(xit.hat[2,], ylab="Beta", main="Market Model")
> barplot(r.square, ylab="R square")
```



Market Model의 활용

❑ Market Model을 이용한 Correlation 추정

$$\mathrm{Cov}[r_t] = \beta\Sigma\beta^T + D$$

❑ 포트폴리오 최적화

$$\min_{\omega} \sigma_p^2 = \omega^T \Sigma \omega \quad (\omega^T \mathbf{1} = 1)$$

$$\omega^* = \frac{\Sigma^{-1}\mathbf{1}}{\mathbf{1}^T \Sigma^{-1} \mathbf{1}}$$

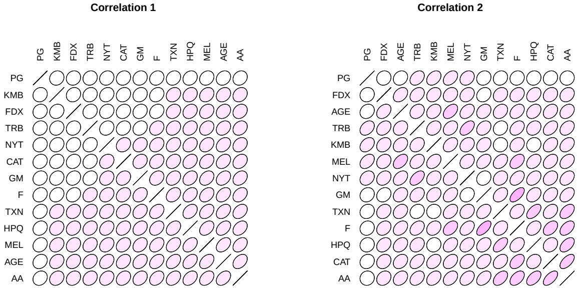
- ▶ ω : 포트폴리오 비중
- ▶ σ_p : 포트폴리오 변동성

Correlation Matrix 추정

```
> library(lattice)
> beta <- t(xit.hat[2,])
> cov.r <- var(xmtx[,2]) * t(beta) %*% beta + diag(D.hat)
> sd.r <- sqrt(diag(cov.r))
> corr.r <- cov.r/outer(sd.r,sd.r)
> print(corr.r, digits=3)

       AA   AGE   CAT     F   FDX    GM   HPQ   KMB   MEL   NYT    PG   TRB   TXN
AA  1.000 0.378 0.274 0.317 0.215 0.286 0.351 0.215 0.366 0.266 0.176 0.233 0.330
AGE 0.378 1.000 0.300 0.347 0.236 0.313 0.384 0.235 0.400 0.290 0.193 0.254 0.361
CAT 0.274 0.300 1.000 0.252 0.171 0.227 0.278 0.170 0.290 0.211 0.140 0.185 0.262
F   0.317 0.347 0.252 1.000 0.198 0.262 0.322 0.197 0.335 0.244 0.162 0.213 0.303
FDX 0.215 0.236 0.171 0.198 1.000 0.178 0.219 0.134 0.228 0.165 0.110 0.145 0.206
GM  0.286 0.313 0.227 0.262 0.178 1.000 0.290 0.178 0.303 0.220 0.146 0.192 0.273
HPQ 0.351 0.384 0.278 0.322 0.219 0.290 1.000 0.218 0.371 0.270 0.179 0.236 0.335
KMB 0.215 0.235 0.170 0.197 0.134 0.178 0.218 1.000 0.227 0.165 0.109 0.144 0.205
MEL 0.366 0.400 0.290 0.335 0.228 0.303 0.371 0.227 1.000 0.281 0.186 0.246 0.349
NYT 0.266 0.290 0.211 0.244 0.165 0.220 0.270 0.165 0.281 1.000 0.135 0.179 0.253
PG  0.176 0.193 0.140 0.162 0.110 0.146 0.179 0.109 0.186 0.135 1.000 0.119 0.168
TRB 0.233 0.254 0.185 0.213 0.145 0.192 0.236 0.144 0.246 0.179 0.119 1.000 0.222
TXN 0.330 0.361 0.262 0.303 0.206 0.273 0.335 0.205 0.349 0.253 0.168 0.222 1.000

> print(cor(rtn), digits=3)

        AA   AGE   CAT     F    FDX    GM    HPQ    KMB   MEL   NYT     PG   TRB    TXN
AA  1.0000 0.299 0.596 0.469 0.2183 0.389 0.5086 0.3338 0.354 0.358 0.0601 0.325 0.4579
AGE 0.2988 1.000 0.281 0.345 0.3373 0.254 0.3056 0.2747 0.438 0.358 0.1770 0.239 0.2536
CAT 0.5963 0.281 1.000 0.421 0.2107 0.344 0.2292 0.3183 0.351 0.269 0.1317 0.389 0.3308
F   0.4689 0.345 0.421 1.000 0.2511 0.614 0.3223 0.2388 0.400 0.387 0.1046 0.334 0.2904
FDX 0.2183 0.337 0.211 0.251 1.0000 0.196 0.2759 0.2456 0.219 0.247 0.0903 0.304 0.2053
GM  0.3885 0.254 0.344 0.614 0.1964 1.000 0.2991 0.2583 0.367 0.184 0.1388 0.280 0.3387
HPQ 0.5086 0.306 0.229 0.322 0.2759 0.299 1.0000 0.0844 0.333 0.337 0.0843 0.221 0.5548
KMB 0.3338 0.275 0.318 0.239 0.2456 0.258 0.0844 1.0000 0.348 0.240 0.3404 0.261 0.0638
MEL 0.3539 0.438 0.351 0.400 0.2191 0.367 0.3332 0.3484 1.000 0.252 0.3722 0.304 0.3441
NYT 0.3579 0.358 0.269 0.387 0.2473 0.184 0.3370 0.2397 0.252 1.000 0.2392 0.496 0.2212
PG  0.0601 0.177 0.132 0.105 0.0903 0.139 0.0843 0.3404 0.372 0.239 1.0000 0.336 0.1357
TRB 0.3251 0.239 0.389 0.334 0.3035 0.280 0.2212 0.2608 0.304 0.496 0.3362 1.000 0.1648
TXN 0.4579 0.254 0.331 0.290 0.2053 0.339 0.5548 0.0638 0.344 0.221 0.1357 0.165 1.0000
```

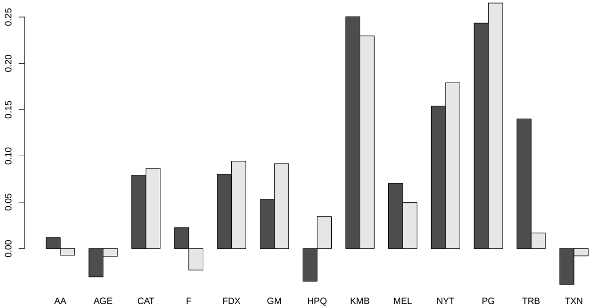
Correlation Matrix 추정

```
> library(ellipse)
> par(mfrow=c(1,2), oma=c(2,0,2,0))
> ord <- order(corr.r[1,])
> ordered.cor.r <- corr.r[ord, ord]
> plotcorr(ordered.cor.r, col=cm.colors(11)[5*ordered.cor.r + 6], main="Correlation 1")
> ord <- order(cor(rtn)[1,])
> ordered.cor.rtn <- cor(rtn)[ord, ord]
> plotcorr(ordered.cor.rtn, col=cm.colors(11)[5*ordered.cor.rtn + 6], main="Correlation 2")
```



포트폴리오 최적화

```
> w.gmin.model1 <- solve(cov.r) %*% rep(1, nrow(cov.r))
> w.gmin.model1 <- w.gmin.model1/sum(w.gmin.model1)
> w.gmin.model2 <- solve(var(rtn)) %*% rep(1, nrow(cov.r))
> w.gmin.model2 <- w.gmin.model2/sum(w.gmin.model2)
> w.gmin.models <- t(cbind(w.gmin.model1, w.gmin.model2))
> rownames(w.gmin.models) <- c("model 1", "model 2")
> print(w.gmin.models, digit=2)

              AA     AGE   CAT      F   FDX    GM    HPQ  KMB  MEL  NYT   PG   TRB    TXN
model 1  0.0117 -0.0306 0.079  0.022 0.080 0.053 -0.035 0.25 0.07 0.15 0.24 0.140 -0.039
model 2 -0.0073 -0.0085 0.087 -0.023 0.094 0.092  0.034 0.23 0.05 0.18 0.27 0.017 -0.008
```



Multifactor Model

- Chen, Roll, and Ross Model (1986)
  - stock return response for unexpected change of macroeconomic indicators
  - 경제지표를 Vector ARMA 모형 등으로 추정한 후
  - 추정치와 달라지는 부분 즉 잔차를 surprise factor로 설정
  - surprise factor에 대한 beta 계산

Multifactor Model 예제

- 2 factor 모형 예제
- 13 개 주식종목 초과수익률을 다음 팩터로 분석
  - CPI(Consumer Price Index)
  - Civilian employment numbers

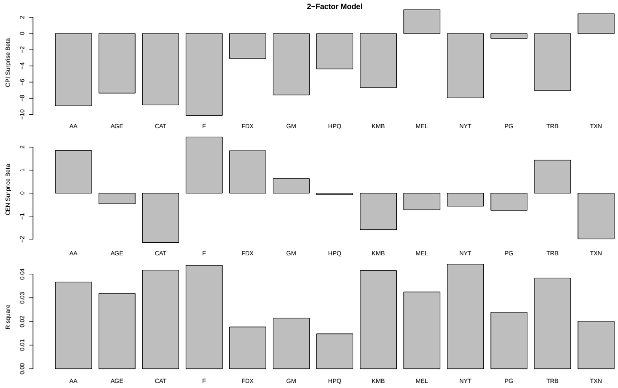
```
> library(vars)
> data(m.cpice16.dp7503)
> y1 <- as.data.frame(m.cpice16.dp7503)
> var3.fit <- VAR(y1, 3)
> res <- residuals(var3.fit)
> rownames(res) <- rownames(y1[-(1:3),])
> res <- res[178:333,]
> xmtx <- as.matrix(cbind(rep(1,156), res))
> colnames(xmtx)[1] <- "alpha"
> rtn <- m.fac9003[1:156,1:13]
> rownames.rtn <- as.character(time(rtn))
> rtn <- as.matrix(rtn)
> rownames(rtn) <- rownames.rtn
> head(rtn)

             AA    AGE    CAT     F    FDX    GM   HPQ    KMB    MEL    NYT    PG    TRB   TXN
Jan 1990 -16.40 -12.17  -4.44 -0.06  -2.28 -2.12 -6.19 -11.01 -10.77  -6.30 -8.89 -13.04 -7.61
Feb 1990   4.04   4.95   8.84  6.02  10.47  8.97 -4.01  -5.20   0.34  -4.62 -0.84  -0.37  4.97
Mar 1990   0.12  13.08   0.17  2.06  10.84  1.57  5.67   3.21  -0.17  -0.66  5.41   2.36  2.69
Apr 1990  -4.28 -11.06   0.25 -5.67  -2.44 -4.19 -5.29  -0.65  -2.20 -10.60  4.26  -7.98 -6.85
May 1990   5.81  19.70   8.52  3.89 -16.17 10.94  8.81   8.83  11.85  11.59 16.35   8.82 22.88
Jun 1990  -4.05  -1.44 -22.10 -5.79  -2.81 -2.70 -1.47   1.55  -7.76  -0.12  4.80  -0.64 -5.87

> head(xmtx)

         alpha         CPI         CE16
Jan 1990     1 -0.02941177 -0.291532372
Feb 1990     1 -0.23163886 -0.214583617
Mar 1990     1 -0.15383800 -0.112237079
Apr 1990     1  0.09807951  0.393685018
May 1990     1  0.16363304 -0.619199508
Jun 1990     1  0.01597111 -0.006387749
```

Multifactor Model 예제

```
> xit.hat <- solve(t(xmtx) %*% xmtx, t(xmtx) %*% rtn)
> E.hat <- rtn - xmtx %*% xit.hat
> D.hat <- diag(crossprod(E.hat)/(156-3))
> sigma.hat <- sqrt(D.hat)
> r.square <- 1 - diag(t(E.hat)%*%E.hat)/diag(t(rtn)%*%rtn)
> t(rbind(xit.hat, sigma.hat, r.square))

        alpha         CPI       CE16 sigma.hat   r.square
AA  0.5966080  -8.9290832  1.8510115  9.302656 0.03664830
AGE 1.1568808  -7.3643644 -0.4619642 10.305346 0.03182555
CAT 0.6175993  -8.8240302 -2.1477888  8.696768 0.04168748
F   0.3940300 -10.1166970  2.4416771  9.237652 0.04368141
FDX 1.0325726  -3.0863542  1.8442246  9.695801 0.01767278
GM  0.2058857  -7.5916051  0.6317389  9.327751 0.02141159
HPQ 1.1408887  -4.3739731 -0.0668382 12.063654 0.01477070
KMB 0.4547482  -6.6773375 -1.5864102  6.555536 0.04146649
MEL 1.3510436   2.9424026 -0.7236635  7.709058 0.03246025
NYT 0.6055017  -7.9504380 -0.5651737  7.434066 0.04419470
PG  1.0245551  -0.5932854 -0.7450971  6.964129 0.02387033
TRB 0.7808319  -7.0472921  1.4353925  7.982408 0.03833309
TXN 1.8917119   2.4457874 -1.9894041 14.024774 0.02009772
```

Multifactor Model 예제

```
> par(mar=c(2,6,1,1))
> layout(matrix(1:3))
> barplot(xit.hat[2,], ylab="CPI Surprise Beta", main="2-Factor Model")
> barplot(xit.hat[3,], ylab="CEN Surprice Beta")
> barplot(r.square, ylab="R square")
```



Fundamental Factor Model

- ❑ 개별 종목의 fundamental 지표를 사용하여 수익률을 설명
- ❑ 문제
  - ▶ fundamental 지표는 Factor Model의 기본적인 가정을 따르지 않음
- ❑ 해결방법 1 : Fama-French 방법
  - ▶ Factor Hedge Portfolio를 수립하여 그 수익률을 팩터로 사용
  - ▶ Factor에 따른 종목 순위를 계산하여 상위 30% 종목 Long, 하위 30% 종목 Short하여 Factor Hedge Portfolio 수립
- ❑ 해결방법 2 : BARRA 방법
  - ▶ fundamental 지표를 factor가 아닌 beta로 사용하고 factor 시계열을 추정

$$r_t = \beta f_t + e_t$$

  - ▶ 추정된 factor는 Factor Mimicking Portfolio의 수익률이 됨

BARRA Industry Factor Model

❑ 종목이 속한 industry에 의한 수익률 factor 계산

$$r_t = \beta_1 f_{t1} + \beta_2 f_{t2} + \cdots + \beta_m f_{tm} + e_t$$

$$\beta_{ij} = \begin{cases} 1 & \text{종목 } i\text{가 industry } j\text{에 속한 경우} \\ 0 & \text{종목 } i\text{가 industry } j\text{에 속하지 않은 경우} \end{cases}$$

❑ Factor Mimicking Portfolio

- ▶ 해당 팩터에 대한 포트폴리오의 exposure합이 1이 되면서 specific error에 의한 포트폴리오 변동성을 최소화하는 포트폴리오
- ▶ industry 팩터의 경우, 그 industry에 속한 종목으로만 포트폴리오를 구성

$$\min_{\omega}\left(\frac{1}{2}\omega^T D \omega\right)$$

$$\omega^T \beta = 1$$

- ▶ Factor Mimicking Portfolio의 해 (solution): Weighted OLS

$$\omega^T = (\beta^T D^{-1} \beta)^{-1} (\beta^T D^{-1})$$

BARRA Industry Factor Model 추정

- factor realization은 Factor Mimicking Portfolio의 수익률

$$f = Hr_t = (\beta^T D^{-1} \beta)^{-1} (\beta^T D^{-1}) r_t$$

- Multi-Step 추정법 사용

1. 단순 OLS 추정으로 factor realization 구함

$$\hat{f}_{t,o} = (\beta^T \beta)^{-1} (\beta^T) r_t$$

2. 단순 OLS 추정factor realization의 residual로 D 추정

$$\begin{aligned} \hat{D}_o &= \text{diag}\left(\frac{1}{T-1} \sum_{t=1}^{T} \hat{e}_{t,o} \hat{e}_{t,o}^T \right) \\ \hat{e}_{t,o} &= r_t - \beta \hat{f}_{t,o} \end{aligned}$$

3. 추정된 $\hat{D}_o$를 이용하여 Weighted OLS 추정치 계산

$$\hat{f}_{t,g} = (\beta^T \hat{D}_o^{-1} \beta)^{-1} (\beta^T \hat{D}_o^{-1}) r_t$$

4. 다시 residual을 이용하여 specific error 추정

$$\begin{aligned} \hat{D}_g &= \text{diag}\left(\frac{1}{T-1} \sum_{t=1}^{T} \hat{e}_{t,g} \hat{e}_{t,g}^T \right) \\ \hat{e}_{t,g} &= r_t - \beta \hat{f}_{t,g} \end{aligned}$$

BARRA Industry Factor Model 추정 예제

- ❏ 미국 주식 16종목의 월 수익률과 이자율
- ❏ 1978-01-01 - 1987-12-01

```
> library(fEcofin)
> library(PerformanceAnalytics)
> library(zoo)
> data(berndtInvest)
> berndt.df = berndtInvest[, -1]
> rownames(berndt.df) = as.character(berndtInvest[, 1])
> returns.mat = as.matrix(berndt.df[, c(-10, -17)])
> asset.names = colnames(returns.mat)
> head(returns.mat)

           CITCRP  CONED CONTIL DATGEN    DEC  DELTA GENMIL GERBER    IBM  MOBIL  PANAM   PSNH
1978-01-01 -0.115 -0.079 -0.129 -0.084 -0.100 -0.028 -0.099 -0.048 -0.029 -0.046  0.025 -0.008
1978-02-01 -0.019 -0.003  0.037 -0.097 -0.063 -0.033  0.018  0.160 -0.043 -0.017 -0.073 -0.025
1978-03-01  0.059  0.022  0.003  0.063  0.010  0.070 -0.023 -0.036 -0.063  0.049  0.184  0.026
1978-04-01  0.127 -0.005  0.180  0.179  0.165  0.150  0.046  0.004  0.130  0.077  0.089 -0.008
1978-05-01  0.005 -0.014  0.061  0.052  0.038 -0.031  0.063  0.046 -0.018 -0.011  0.082  0.019
1978-06-01  0.007  0.034 -0.059 -0.023 -0.021  0.023  0.008  0.028 -0.004 -0.043  0.019  0.032
            TANDY TEXACO  WEYER
1978-01-01 -0.075 -0.054 -0.116
1978-02-01 -0.004 -0.010 -0.135
1978-03-01  0.124  0.015  0.084
1978-04-01  0.055  0.000  0.144
1978-05-01  0.176 -0.029 -0.031
1978-06-01 -0.014 -0.025  0.005
```

BARRA Industry Factor Model 추정 예제

❑ industry beta 생성

```
> n.stocks = ncol(returns.mat)
> tech.dum = oil.dum = other.dum = matrix(0,n.stocks,1)
> rownames(tech.dum) = rownames(oil.dum) = rownames(other.dum) = asset.names
> tech.dum[c(4,5,9,13),] = 1
> oil.dum[c(3,6,10,11,14),] = 1
> other.dum = 1 - tech.dum - oil.dum
> B.mat = cbind(tech.dum,oil.dum,other.dum)
> colnames(B.mat) = c("TECH","OIL","OTHER")
> B.mat

       TECH OIL OTHER
CITCRP    0   0     1
CONED     0   0     1
CONTIL    0   1     0
DATGEN    1   0     0
DEC       1   0     0
DELTA     0   1     0
GENMIL    0   0     1
GERBER    0   0     1
IBM       1   0     0
MOBIL     0   1     0
PANAM     0   1     0
PSNH      0   0     1
TANDY     1   0     0
TEXACO    0   1     0
WEYER     0   0     1
```

BARRA Industry Factor Model 추정 예제

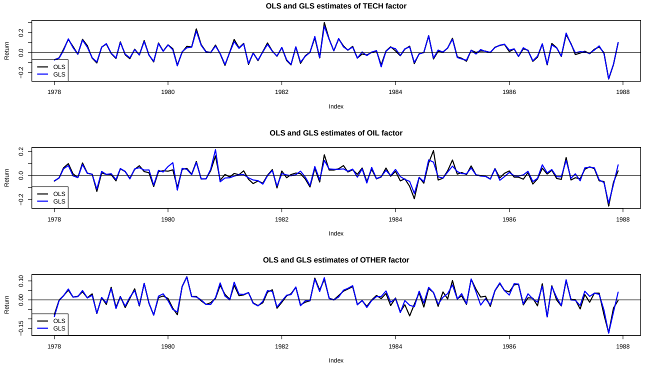
❑ factor realization 추정

```
> returns.mat = t(returns.mat)
> F.hat = solve(crossprod(B.mat))%*%t(B.mat)%*%returns.mat
> F.hat.zoo = zoo(t(F.hat), as.Date(colnames(F.hat)))
> E.hat = returns.mat - B.mat%*%F.hat
> diagD.hat = apply(E.hat, 1, var)
> Dinv.hat = diag(diagD.hat^(-1))
> H.hat = solve(t(B.mat)%*%Dinv.hat%*%B.mat)%*%t(B.mat)%*%Dinv.hat
> colnames(H.hat) = asset.names
> F.hat.gls = H.hat%*%returns.mat
> F.hat.gls.zoo = zoo(t(F.hat.gls), as.Date(colnames(F.hat.gls)))
> returns.mat = t(returns.mat)
> t(H.hat)

            TECH        OIL      OTHER
CITCRP 0.0000000 0.00000000 0.19917745
CONED  0.0000000 0.00000000 0.22023721
CONTIL 0.0000000 0.09610815 0.00000000
DATGEN 0.2196573 0.00000000 0.00000000
DEC    0.3187562 0.00000000 0.00000000
DELTA  0.0000000 0.22326150 0.00000000
GENMIL 0.0000000 0.00000000 0.22967370
GERBER 0.0000000 0.00000000 0.12696982
IBM    0.2810286 0.00000000 0.00000000
MOBIL  0.0000000 0.28645186 0.00000000
PANAM  0.0000000 0.11856526 0.00000000
PSNH   0.0000000 0.00000000 0.06682885
TANDY  0.1805580 0.00000000 0.00000000
TEXACO 0.0000000 0.27561323 0.00000000
WEYER  0.0000000 0.00000000 0.15711298

> colSums(t(H.hat))

 TECH   OIL OTHER
    1     1     1
```

BARRA Industry Factor Model 추정 예제

❏ factor realization 추정 결과

```
> par(mfrow=c(3,1))
> plot(merge(F.hat.zoo[,1], F.hat.gls.zoo[,1]), plot.type="single",
+      main = "OLS and GLS estimates of TECH factor", col=c("black", "blue"), lwd=2, ylab="Return")
> legend(x = "bottomleft", legend=c("OLS", "GLS"), col=c("black", "blue"), lwd=2); abline(h=0)
> plot(merge(F.hat.zoo[,2], F.hat.gls.zoo[,2]), plot.type="single",
+      main = "OLS and GLS estimates of OIL factor", col=c("black", "blue"), lwd=2, ylab="Return")
> legend(x = "bottomleft", legend=c("OLS", "GLS"), col=c("black", "blue"), lwd=2); abline(h=0)
> plot(merge(F.hat.zoo[,3], F.hat.gls.zoo[,3]), plot.type="single",
+      main = "OLS and GLS estimates of OTHER factor", col=c("black", "blue"), lwd=2, ylab="Return")
> legend(x = "bottomleft", legend=c("OLS", "GLS"), col=c("black", "blue"), lwd=2); abline(h=0)
> par(mfrow=c(1,1))
```



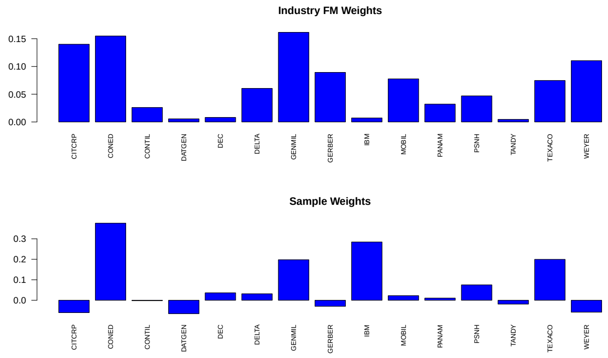
❑ Covariance Matrix 추정

```
> cov.ind = B.mat%*%var(t(F.hat.gls))%*%t(B.mat) + diag(diagD.hat)
> cor.ind = cov2cor(cov.ind)
> rownames(cor.ind) = colnames(cor.ind)
> r.square.ind = 1 - diagD.hat/diag(cov.ind)
> ind.fm.vals = cbind(B.mat, sqrt(diag(cov.ind)), sqrt(diagD.hat), r.square.ind)
> colnames(ind.fm.vals) = c(colnames(B.mat), "fm.sd", "residual.sd", "r.square")
> ind.fm.vals

       TECH OIL OTHER      fm.sd residual.sd  r.square
CITCRP    0   0     1 0.07290893  0.05468251 0.4374833
CONED     0   0     1 0.07092096  0.05200238 0.4623531
CONTIL    0   1     0 0.13258082  0.11807260 0.2068837
DATGEN    1   0     0 0.10645589  0.07189338 0.5439228
DEC       1   0     0 0.09862028  0.05968049 0.6337883
DELTA     0   1     0 0.09817243  0.07746800 0.3773190
GENMIL    0   0     1 0.07013326  0.05092287 0.4727972
GERBER    0   0     1 0.08376290  0.06848860 0.3314512
IBM       1   0     0 0.10101545  0.06356037 0.6040893
MOBIL     0   1     0 0.09118092  0.06839171 0.4374010
PANAM     0   1     0 0.12221752  0.10630422 0.2434562
PSNH      0   0     1 0.10600704  0.09440316 0.2069444
TANDY     1   0     0 0.11158904  0.07929637 0.4950324
TEXACO    0   1     0 0.09218407  0.06972351 0.4279332
WEYER     0   0     1 0.07820668  0.06156906 0.3802203
```

BARRA Industry Factor Model 추정 예제

❑ 포트폴리오 최적화

```
> w.gmin.ind = solve(cov.ind)%*%rep(1,nrow(cov.ind))
> w.gmin.ind = w.gmin.ind/sum(w.gmin.ind)
> w.gmin.sample = solve(var(returns.mat))%*%rep(1,nrow(cov.ind))
> w.gmin.sample = w.gmin.sample/sum(w.gmin.sample)
> colnames(w.gmin.sample) = "sample"
> par(mfrow=c(2,1))
> barplot(t(w.gmin.ind), horiz=F, main="Industry FM Weights", col="blue", cex.names = 0.75, las=2)
> barplot(t(w.gmin.sample), horiz=F, main="Sample Weights", col="blue", cex.names = 0.75, las=2)
> par(mfrow=c(1,1))
```



PCA(Principal Component Analysis)

- Orthogonal Transaform
  - 선형변환에 의한 포트폴리오들의 수익률 시계열이 서로 독립

$$
y_i = \omega_i^T r = \sum_{j=1}^{k} w_{ij} r_j
$$

- solution
  - 수익률 Covariance Matrix의 eigenvector가 포트폴리오 비중
  - 수익률 Covariance Matrix의 eigenvalue는 variance λ_i

$$
\begin{aligned}
\mathrm{Var}(y_i) &= \lambda_i \\
\mathrm{Cov}(y_i, y_j) &= 0 \text{ if } i \neq
\end{aligned}
$$

- Proportion of Variance
  - 전체 Covariance Matrix 중 해당 component로 설명되는 부분

$$
\frac{\mathrm{Var}(y_i)}{\sum_{i=1}^{k} \mathrm{Var}(r_i)} = \frac{\lambda_i}{\sum_{i=1}^{k} \lambda_i} =
$$

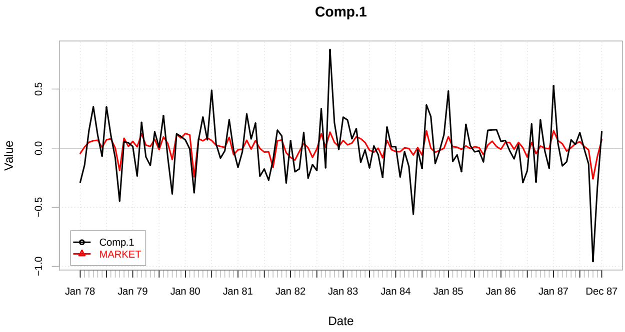
R의 PCA 명령

- ❑ **princomp(x)**
  - ▶ x : 수익률 시계열 matrix/dataframe

```
> (pc.fit = princomp(returns.mat))

Call:
princomp(x = returns.mat)

Standard deviations:
    Comp.1     Comp.2     Comp.3     Comp.4     Comp.5     Comp.6     Comp.7     Comp.8     Comp.9 
0.22817560 0.14078550 0.12639456 0.10444062 0.09741253 0.09042932 0.08123290 0.07731405 0.06790621 
   Comp.10    Comp.11    Comp.12    Comp.13    Comp.14    Comp.15 
0.05633979 0.05352967 0.04703070 0.04529347 0.04033415 0.03722748 

 15  variables and  120 observations.

> summary(pc.fit)

Importance of components:
                          Comp.1    Comp.2    Comp.3     Comp.4     Comp.5     Comp.6     Comp.7
Standard deviation     0.2281756 0.1407855 0.1263946 0.10444062 0.09741253 0.09042932 0.08123290
Proportion of Variance 0.3543271 0.1348906 0.1087233 0.07423434 0.06457965 0.05565248 0.04490864
Cumulative Proportion  0.3543271 0.4892178 0.5979411 0.67217543 0.73675507 0.79240755 0.83731619
                           Comp.8     Comp.9    Comp.10    Comp.11   Comp.12    Comp.13    Comp.14
Standard deviation     0.07731405 0.06790621 0.05633979 0.05352967 0.0470307 0.04529347 0.04033415
Proportion of Variance 0.04068018 0.03138232 0.02160212 0.01950092 0.0150532 0.01396166 0.01107164
Cumulative Proportion  0.87799637 0.90937869 0.93098081 0.95048173 0.9655349 0.97949659 0.99056823
                           Comp.15
Standard deviation     0.037227484
Proportion of Variance 0.009431773
Cumulative Proportion  1.000000000
```

PCA Component Plot

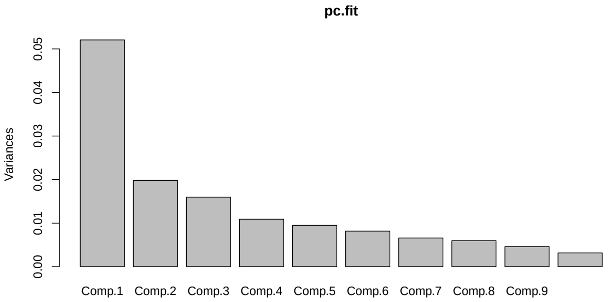
```
> chart.TimeSeries(cbind(-pc.fit$scores[, 1, drop=FALSE], berndt.df[, "MARKET",drop=F]),
+                  legend.loc="bottomleft")
```



PCA Eigenvalue Scree Plot

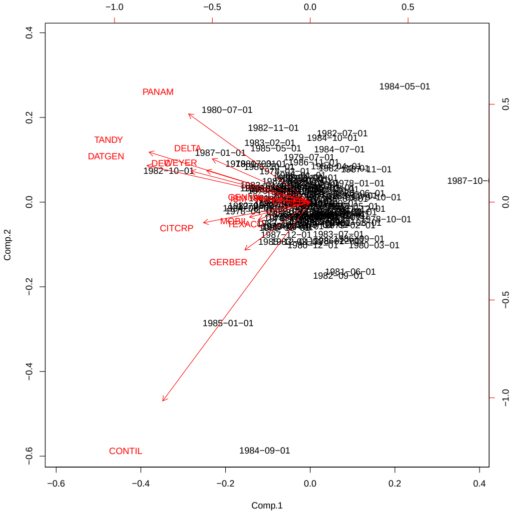
❑ 각 component가 가지는 eigenvalue 비교

```
> plot(pc.fit)
```



PCA BiPlot

❑ 가장 비중이 큰 2개의 component에 대해 각 시계열/샘플이 가지는 비중을 표시



Statistical Factor Model

- 전체 주식 종목 수 k보다 적은 $m(m < k)$개 팩터로 Covariance Matrix를 복제

$$r - \mu = \beta f + e$$

$$\Sigma_r = \beta\beta^T + D$$

$$\begin{aligned}
\text{Var}(r_i) &= \sum_{j=1}^{m} \beta_{ij}^2 + \sigma_i^2 \\
\text{Cov}(r_i, r_j) &= \sum_{k=1}^{m} \beta_{ik}\beta_{jk} \\
\text{Cov}(r_i, f_j) &= \beta_{ij}
\end{aligned}$$

- communality
  - i번째 주식의 수익률 변동성 중 팩터에 의해 설명되는 부분

$$\sum_{j=1}^{m} \beta_{ij}^2$$

Statistical Factor Model 추정

- PCA 추정법
  - PCA 분석으로 찾은 가장 큰 m 개의 eigenvector를 beta로 이용하는 방법

$$\beta = \left[\sqrt{\lambda_1} e_1 | \cdots | \sqrt{\lambda_m} e_m \right]$$

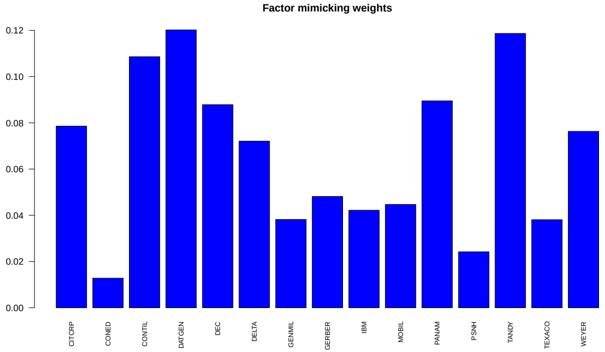
  - PCA 분석으로 추정한 factor realization을 이용하여 역으로 beta 추정
- MLE 추정법
  - beta 제한조건하에서 covariance matrix 를 복제하는 beta 추정

$$\beta^T D^{-1} \beta = \Delta$$

Statistical Factor Model 예제

❑ 가장 큰 single eigenvector를 이용하는 PCA 방법

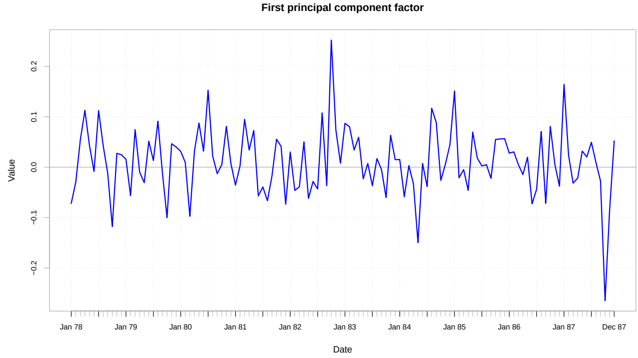
```
> p1 = pc.fit$loadings[, 1]
> p1 = p1/sum(p1)
> p1

    CITCRP      CONED     CONTIL     DATGEN        DEC      DELTA     GENMIL     GERBER        IBM
0.07855793 0.01279276 0.10858153 0.12017658 0.08783822 0.07206445 0.03818441 0.04815376 0.04218390
     MOBIL      PANAM       PSNH      TANDY     TEXACO      WEYER
0.04469800 0.08949227 0.02421715 0.11866104 0.03809685 0.07630117

> barplot(p1, horiz=F, main="Factor mimicking weights", col="blue", cex.names = 0.75, las=2)
```



Statistical Factor Model 예제

❑ factor realization

```
> f1 = returns.mat %*% p1
> chart.TimeSeries(f1, main="First principal component factor", colorset="blue")
```



Statistical Factor Model 예제

❑ beta 추정

```
> n.obs = nrow(returns.mat)
> X.mat = cbind(rep(1,n.obs), f1)
> colnames(X.mat) = c("intercept", "Factor 1")
> XX.mat = crossprod(X.mat)
> G.hat = solve(XX.mat)%*%crossprod(X.mat,returns.mat)
> beta.hat = G.hat[2,]
> E.hat = returns.mat - X.mat%*%G.hat
> diagD.hat = diag(crossprod(E.hat)/(n.obs-2))
> sumSquares = apply(returns.mat, 2, function(x) {sum( (x - mean(x))^2 )})
> R.square = 1 - (n.obs-2)*diagD.hat/sumSquares
> cbind(beta.hat, diagD.hat, R.square)

        beta.hat   diagD.hat   R.square
CITCRP 0.9467156 0.002674264 0.59554424
CONED  0.1541678 0.002444255 0.04097136
CONTIL 1.3085353 0.015379927 0.32846719
DATGEN 1.4482693 0.007189044 0.56175824
DEC    1.0585540 0.004989735 0.49663640
DELTA  0.8684615 0.005967208 0.35704275
GENMIL 0.4601671 0.003335760 0.21807650
GERBER 0.5803095 0.006283634 0.19058470
IBM    0.5083656 0.002377929 0.32317516
MOBIL  0.5386635 0.005229151 0.19600439
PANAM  1.0784873 0.012409694 0.29167999
PSNH   0.2918452 0.011711303 0.03096336
TANDY  1.4300053 0.007426707 0.54745528
TEXACO 0.4591119 0.005480424 0.14455217
WEYER  0.9195189 0.003582860 0.50903652
```

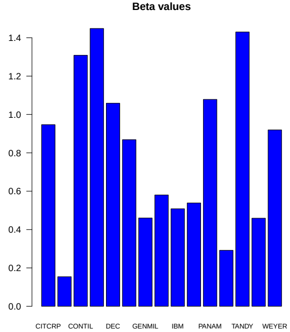
Statistical Factor Model 예제

❑ beta 추정

```
> n.obs = nrow(returns.mat)
> X.mat = cbind(rep(1,n.obs), f1)
> colnames(X.mat) = c("intercept", "Factor 1")
> XX.mat = crossprod(X.mat)
> G.hat = solve(XX.mat)%*%crossprod(X.mat,returns.mat)
> beta.hat = G.hat[2,]
> E.hat = returns.mat - X.mat%*%G.hat
> diagD.hat = diag(crossprod(E.hat)/(n.obs-2))
> sumSquares = apply(returns.mat, 2, function(x) {sum( (x - mean(x))^2 )})
> R.square = 1 - (n.obs-2)*diagD.hat/sumSquares
> cbind(beta.hat, diagD.hat, R.square)

        beta.hat   diagD.hat   R.square
CITCRP 0.9467156 0.002674264 0.59554424
CONED  0.1541678 0.002444255 0.04097136
CONTIL 1.3085353 0.015379927 0.32846719
DATGEN 1.4482693 0.007189044 0.56175824
DEC    1.0585540 0.004989735 0.49663640
DELTA  0.8684615 0.005967208 0.35704275
GENMIL 0.4601671 0.003335760 0.21807650
GERBER 0.5803095 0.006283634 0.19058470
IBM    0.5083656 0.002377929 0.32317516
MOBIL  0.5386635 0.005229151 0.19600439
PANAM  1.0784873 0.012409694 0.29167999
PSNH   0.2918452 0.011711303 0.03096336
TANDY  1.4300053 0.007426707 0.54745528
TEXACO 0.4591119 0.005480424 0.14455217
WEYER  0.9195189 0.003582860 0.50903652
```

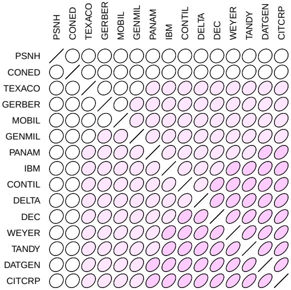
Statistical Factor Model 예제

```
> par(mfrow=c(1,2))
> barplot(beta.hat, main="Beta values", col="blue", cex.names = 0.75, las=1)
> barplot(R.square, main="R-square values", col="blue", cex.names = 0.75, las=1)
> par(mfrow=c(1,1))
```



Statistical Factor Model 예제

```
> cov.pc1 = as.numeric(var(f1))*beta.hat%*%t(beta.hat) + diag(diagD.hat)
> cor.pc1 = cov2cor(cov.pc1)
> rownames(cor.pc1) = colnames(cor.pc1)
> ord <- order(cor.pc1[1,])
> ordered.cor.pc1 <- cor.pc1[ord, ord]
> plotcorr(ordered.cor.pc1, col=cm.colors(11)[5*ordered.cor.pc1 + 6])
```



Statistical Factor Model 예제

```
> w.gmin.pc1 = solve(cov.pc1)%*%rep(1,nrow(cov.pc1))
> w.gmin.pc1 = w.gmin.pc1/sum(w.gmin.pc1)
> colnames(w.gmin.pc1) = "principal.components"
> par(mfrow=c(2,1))
> barplot(t(w.gmin.pc1), horiz=F, main="Principal Component Weights", col="blue", cex.names = 0.75, las=2)
> barplot(t(w.gmin.sample), horiz=F, main="Sample Weights", col="blue", cex.names = 0.75, las=2)
> par(mfrow=c(1,1))
```