Factor Analysis Examples

PC method vs ML method on stock data

STAT 32950-24620

Spring 2023 (3/30, wk2)

Orthogonal Factor Model

$$X = \mu + LF + \varepsilon$$

- $X = [X_1 \ \cdots \ X_p]'$

— Observable, random, $E(X) = \mu$

- $F = [F_1 \ \cdots \ F_m]'$

— Latent factors, unobservable, random

- $L = L_{p \times m}$

— Factor loading matrix, parameters, to be estimated

Orthogonal factor model assumptions

Assumptions on common factors

- $E(F) = 0_m$
- $Cov(F) = I_m$

Assumptions on errors (specific factors)

- $E(\varepsilon) = 0_p$,

Assumptions on covariance between common and specific factors

- $Cov(\varepsilon) = diag\{\psi_1, \cdots, \psi_p\} = \Psi$
- $Cov(F, \varepsilon) = E(F\varepsilon') = 0_{m \times p}$
- $Cov(\varepsilon, F) = E(\varepsilon F') = 0_{p \times m}$

Covariance structure in factor model

Key structure assumption of the model:

$$\Sigma = LL' + \Psi$$

where

$$\Sigma = Cov(X)$$

under the factor relation

$$X = \mu + LF + \epsilon$$

FA parameter estimation methods

- By Principal Components
- By Maximum Likelihood

The Maximum Likelihood method is usually preferred.

Principal factor estimation method

$$L_m = \left[\sqrt{\lambda_1}e_1, \cdots, \sqrt{\lambda_m}e_m\right], \quad m \leq p$$

When $m = p$: $\Sigma = L_p L_p'$

When $m < p$: $\Sigma = L_m L_m' + \quad \cdots$

In orthogonal factor model

$$X = \mu + LF + \varepsilon$$

PC method uses the approximation

$$\Sigma \approx L_m L_m' + \Psi$$

for factor model with $m < p$ factors.

Obtain the e-vector e_i for the L matrix

```
# Using original data (alternative: var. variance=1 later)
summary(princomp(stock), loading=T)
```

```
## Importance of components:
##                         Comp.1  Comp.2  Comp.3  Comp.4  (
## Standard deviation     0.0368 0.02635 0.01585 0.01188 0.
## Proportion of Variance 0.5293 0.27133 0.09822 0.05518 0.
## Cumulative Proportion  0.5293 0.80059 0.89881 0.95399 1.
##
## Loadings:
##            Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
## JPMorgan    0.223  0.625  0.326  0.663  0.118
## Citibank    0.307  0.570 -0.250 -0.414 -0.589
## WellsFargo  0.155  0.345        -0.497  0.780
## Shell       0.639 -0.248 -0.642  0.309  0.148
## Exxon       0.651 -0.322  0.646 -0.216
```


Obtain λ_i for the L matrix

```
rtev = princomp(stock)$sdev   # =sqrt of e-values of cov(X)
sqrt(eigen(cov(stock))$values)   # verify: same as above
```

```
## [1] 0.03698 0.02648 0.01593 0.01194 0.01090
```

```
rtev
```

```
##  Comp.1  Comp.2  Comp.3  Comp.4  Comp.5
## 0.03680 0.02635 0.01585 0.01188 0.01085
```

Two common factors (PC method)

Choose $m = 2$

Take the first 2 PC's

$$Y_i = e_i' X, \quad i = 1, 2.$$

On original data (using covariance matrix)

```
princomp(stock)$loading[,1:2]
```

```
##             Comp.1  Comp.2
## JPMorgan    0.2228  0.6252
## Citibank    0.3073  0.5704
## WellsFargo  0.1548  0.3445
## Shell       0.6390 -0.2479
## Exxon       0.6509 -0.3218
```


Factor loadings for m=2 (PC method)

$$L_2 = \left[\sqrt{\lambda_1} e_1, \sqrt{\lambda_2} e_2\right]$$

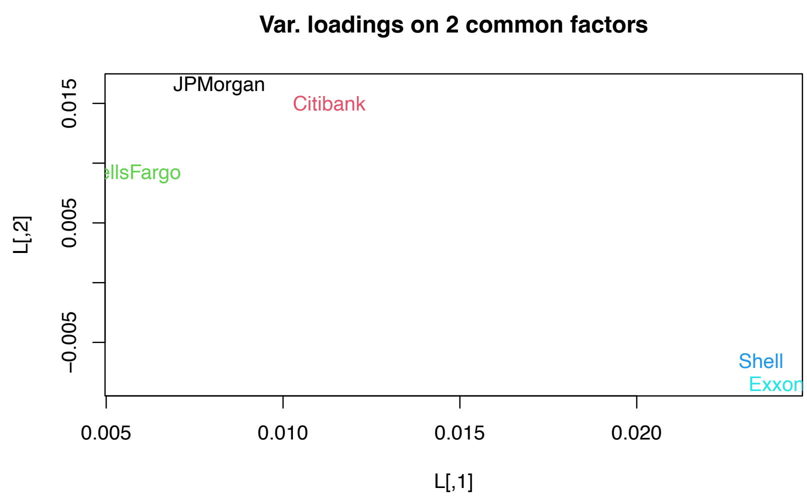
Scale the PC variables by $\sqrt{\lambda_i}$ to get common factor loadings:

```
L = cbind(rtev[1]*princomp(stock)$loading[,1],
          rtev[2]*princomp(stock)$loading[,2]); L
```

```
##                  [,1]      [,2]
## JPMorgan   0.008200  0.016475
## Citibank   0.011309  0.015030
## WellsFargo 0.005697  0.009078
## Shell      0.023515 -0.006534
## Exxon      0.023955 -0.008481
```

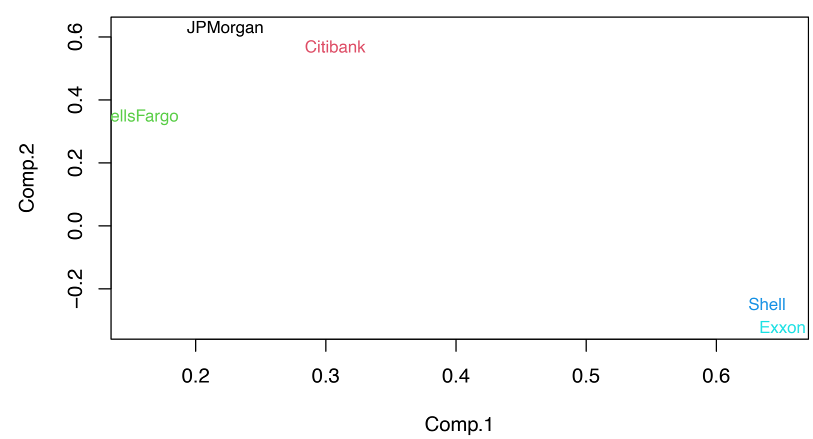

Plot of variables on two factors (PC method)

```
plot(L,type="n",main="Var. loadings on 2 common factors")
text(L,labels=(row.names(L)),col=1:5)
```



Comparison: Plot of variables on PC1 and PC2

```
plot(princomp(stock)$loading[,1:2],type="n");
text(princomp(stock)$loading[,1:2],
     labels=(colnames(stock)),cex=.8,col=1:5)
```



Covariance structure estimation (PC method)

Model assumption

$$\Sigma = LL' + \Psi$$

PC estimation method:

Let

$$\widehat{L} = \widehat{L}_m$$

Then

$$\widehat{\Sigma} \approx \widehat{L}\widehat{L}' + \widehat{\Psi}$$

Estimate of LL'

Get $\widehat{L}\,\widehat{L}'$, the estimate of LL'

```
LLT= L%*%t(L)
round(10000*LLT,3)
```

```
##           JPMorgan Citibank WellsFargo Shell Exxon
## JPMorgan     3.387    3.404      1.963 0.852 0.567
## Citibank     3.404    3.538      2.009 1.677 1.434
## WellsFargo   1.963    2.009      1.149 0.747 0.595
## Shell        0.852    1.677      0.747 5.957 6.187
## Exxon        0.567    1.434      0.595 6.187 6.458
```


Estimate of specific factor matrix Φ (PC method)

Get the estimate of $\Psi = Cov(\varepsilon)$ $(\times 10^4)$

```
## get the specific factors
Psi = diag(cov(stock)) - diag(LLT)
round(10000*Psi,3)
```

```
##   JPMorgan   Citibank WellsFargo      Shell      Exxon
##      0.946      0.849      1.091      1.268      1.199
```

Compared with sample variance $s_{ii} = s_i^2$ $(\times 10^4)$

```
## get sample variance
round(10000*diag(cov(stock)),3)
```

```
##   JPMorgan   Citibank WellsFargo      Shell      Exxon
##      4.333      4.387      2.240      7.225      7.657
```


Estimate of communality h_i^2 (PC method)

Get the estimate of communality $h_i^2 = \ell_{i1}^2 + \cdots + \ell_{im}^2$ $(\times 10^4)$,

the portion of $Var(X_i)$ explained by common factors,

which are diagonal elements of estimated LL':

```
## get communality h_i^2
round(10000*diag(LLT),3)
```

```
##   JPMorgan   Citibank WellsFargo      Shell      Exxon
##      3.387      3.538      1.149      5.957      6.458
```

%vaiance explained by each common factor (PC)

For $j = 1, \cdots, m$.

$$\frac{\hat{\ell}_{1j}^2 + \cdots + \hat{\ell}_{pj}^2}{s_{11} + \cdots + s_{pp}} = \frac{(\sqrt{\hat{\lambda}_j} e_j)(\sqrt{\hat{\lambda}_j} e_j)'}{tr(\mathbf{S})} = \frac{\hat{\lambda}_j \|e_j\|^2}{tr(\mathbf{S})} = \frac{\hat{\lambda}_j}{\sum_{i=1}^p \hat{\lambda}_i}$$

```
eigen(cov(stock))$value[1:2]/sum(eigen(cov(stock))$value)
```

```
## [1] 0.5293 0.2713
```


The approximation $\Sigma \approx L_2 L_2' + \Psi$ $(\times 10^4)$

```
round((LLT + diag(Psi))*10000,2)
```

```
##            JPMorgan Citibank WellsFargo Shell Exxon
## JPMorgan       4.33     3.40       1.96  0.85  0.57
## Citibank       3.40     4.39       2.01  1.68  1.43
## WellsFargo     1.96     2.01       2.24  0.75  0.59
## Shell          0.85     1.68       0.75  7.22  6.19
## Exxon          0.57     1.43       0.59  6.19  7.66
```

Sample Σ $(\times 10^4)$

```
round((cov(stock))*10000,2)
```

```
##            JPMorgan Citibank WellsFargo Shell Exxon
## JPMorgan       4.33     2.76       1.59  0.64  0.89
## Citibank       2.76     4.39       1.80  1.81  1.23
## WellsFargo     1.59     1.80       2.24  0.73  0.61
## Shell          0.64     1.81       0.73  7.22  5.08
## Exxon          0.89     1.23       0.61  5.08  7.66
```


Maximum loading with m=5 factors (PC method)

```
L5=
princomp(stock)$loading[,1:5]%*%
  +diag((princomp(stock)$sdev))
round(L5,4)
```

```
##              [,1]    [,2]    [,3]    [,4]    [,5]
## JPMorgan   0.0082  0.0165  0.0052  0.0079  0.0013
## Citibank   0.0113  0.0150 -0.0040 -0.0049 -0.0064
## WellsFargo 0.0057  0.0091 -0.0006 -0.0059  0.0085
## Shell      0.0235 -0.0065 -0.0102  0.0037  0.0016
## Exxon      0.0240 -0.0085  0.0102 -0.0026 -0.0010
```


Numerical rounding errors

We should have $L_5 L_5' = \Sigma$ in theory, but there are rounding errors...

```
round((L5%*%t(L5))*10000,2)
```

```
##            JPMorgan Citibank WellsFargo Shell Exxon
## JPMorgan       4.29     2.73       1.57  0.63  0.88
## Citibank       2.73     4.34       1.78  1.80  1.22
## WellsFargo     1.57     1.78       2.22  0.73  0.60
## Shell          0.63     1.80       0.73  7.15  5.03
## Exxon          0.88     1.22       0.60  5.03  7.58
```

```
round((cov(stock))*10000,2)
```

```
##            JPMorgan Citibank WellsFargo Shell Exxon
## JPMorgan       4.33     2.76       1.59  0.64  0.89
## Citibank       2.76     4.39       1.80  1.81  1.23
## WellsFargo     1.59     1.80       2.24  0.73  0.61
## Shell          0.64     1.81       0.73  7.22  5.08
## Exxon          0.89     1.23       0.61  5.08  7.66
```

Factor model estimation by Maximum Likelihood method

Create normalized data with variance = 1

```
normdata=(as.matrix(stock))%*%diag(1/sqrt(diag(cov(stock)))
dim(normdata)
```

```
## [1] 103   5
```

```
cov(normdata)
```

```
##        [,1]   [,2]   [,3]   [,4]   [,5]
## [1,] 1.0000 0.6323 0.5105 0.1146 0.1545
## [2,] 0.6323 1.0000 0.5741 0.3223 0.2127
## [3,] 0.5105 0.5741 1.0000 0.1825 0.1462
## [4,] 0.1146 0.3223 0.1825 1.0000 0.6834
## [5,] 0.1545 0.2127 0.1462 0.6834 1.0000
```



```
ML2 =factanal(normdata,2, rotation="none")
ML2$call
```

```
## factanal(x = normdata, factors = 2, rotation = "none")
```

```
ML2$method
```

```
## [1] "mle"
```

```
ML2$uniquenesses
```

```
## [1] 0.4165 0.2747 0.5420 0.0050 0.5298
```

```
ML2$loadings
```

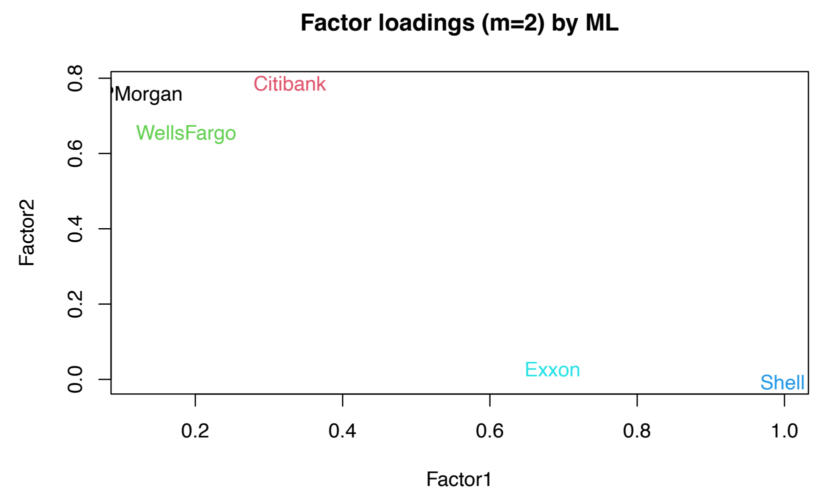

Proportion of vaiance explained by 2 common factors (ML)

$$\frac{\hat{\ell}_{1j}^2 + \cdots + \hat{\ell}_{pj}^2}{s_{11} + \cdots + s_{pp}} = \frac{\hat{\ell}_{1j}^2 + \cdots + \hat{\ell}_{pj}^2}{p}, \qquad j = 1, \cdots, m.$$

```
##
## Loadings:
##      Factor1 Factor2
## [1,]  0.121   0.754
## [2,]  0.328   0.786
## [3,]  0.188   0.650
## [4,]  0.997
## [5,]  0.685
##
##                Factor1 Factor2
## SS loadings      1.622   1.610
## Proportion Var   0.324   0.322
## Cumulative Var   0.324   0.646
```


Plots of variables on two factors

```
Lm = factanal(normdata,2,rotation="none")$loading[,1:2]
plot(Lm,type="n",main="Factor loadings (m=2) by ML")
text(Lm,labels=(row.names(L)),col=1:5)
```



Remarks on ML method for FA

- ML method for FA always estimate correlation matrix

```
factanal(stock,2,rotation="none") #same as using normdata
```

- Large sample test for the number of common factors*

```
ML2$STATISTIC  # Bartlett chisq approximation for LR test
```

```
## objective
##     1.974
```

```
ML2$dof   # [(p-m)^2 - p - m]/2
```

```
## [1] 1
```

```
ML2$PVAL  # p-value
```

```
## objective
##      0.16
```


Factor estimations by ML vs PC

Check residuals $\Sigma - LL' - \Psi$

ML (m=2)

```
Lm = factanal(normdata,2, rotation="none")$loading[,1:2]
Psim = factanal(normdata,2, rotation="none")$uniq

# Residual matrix for m = 2, using ML method
round(cor(stock) - Lm%*%t(Lm) - diag(Psim),3)
```

```
##            JPMorgan Citibank WellsFargo Shell  Exxon
## JPMorgan      0.000    0.000     -0.003     0  0.052
## Citibank      0.000    0.000      0.002     0 -0.033
## WellsFargo   -0.003    0.002      0.000     0  0.001
## Shell         0.000    0.000      0.000     0  0.000
## Exxon         0.052   -0.033      0.001     0  0.000
```


Check model performance

Residual matrix $= \Sigma - LL' - \Psi$

PC (m=2) on correlation matrix

```
### Residua matrix for m = 2, using PC method
normrtev = princomp(stock,cor=T)$sdev
Ln = cbind(normrtev[1]*princomp(stock,cor=T)$loading[,1],
           normrtev[2]*princomp(stock,cor=T)$loading[,2])
round(cor(stock) - Ln%*%t(Ln) - diag(rep(1,5)
                              - diag(Ln%*%t(Ln))),3)
```

```
##            JPMorgan Citibank WellsFargo  Shell  Exxon
## JPMorgan      0.000   -0.099     -0.185 -0.025  0.056
## Citibank     -0.099    0.000     -0.134  0.014 -0.054
## WellsFargo   -0.185   -0.134      0.000  0.003  0.006
## Shell        -0.025    0.014      0.003  0.000 -0.156
## Exxon         0.056   -0.054      0.006 -0.156  0.000
```


Which method is better for this data set?

- Compare proportions of variation explained
- Compare residual matrix element size

```
# ML methods sum-abs and sum-sq of residual(ij), m=2
c(sum(abs(cor(stock) - Lm%*%t(Lm) - diag(Psim))),
  sum((cor(stock) - Lm%*%t(Lm) - diag(Psim))^2))
```

```
## [1] 0.180946 0.007612
```

```
#PC method sum-abs, sum-sq of residuals, m=2, corr mat
c(sum(abs(cor(stock) - Ln%*%t(Ln) - diag(rep(1,5)
- diag(Ln%*%t(Ln))))), sum((cor(stock) -
Ln%*%t(Ln) - diag(rep(1,5) - diag(Ln%*%t(Ln))))^2))
```

```
## [1] 1.4630 0.1861
```