Structural Equation Modeling with Latent Variables using R

Muhammad Yaseen

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 GENERAL STRUCTURAL EQUATION MODELS

1.1 INTRODUCTION

General Structural Equation Model

The *general structural equation model* consists of a *measurement model* that specifies the relation of observed to latent variables and a *latent variable model* that shows the influence of latent variables on each other.

1.2 MODEL

The first component of the structural equations is the latent variable model:

Latent Variable Model

$$\boldsymbol{\eta}_i = \mathbf{B}\boldsymbol{\eta}_i + \boldsymbol{\Gamma}\boldsymbol{\xi}_i + \boldsymbol{\zeta}_i \tag{1}$$

In (1), $\boldsymbol{\eta}_i$ (eta), the vector of latent endogenous random variables, is $m \times 1$; $\boldsymbol{\xi}_i$ (xi) the latent exogenous random variables, is $n \times 1$; $\mathbf{B}$ is the $m \times m$ coefficient matrix showing the influence of the latent endogenous variables on each other; $\boldsymbol{\Gamma}$ (Gamma) is the $m \times n$ coefficient matrix for the effects of $\boldsymbol{\xi}_i$ on $\boldsymbol{\eta}_i$. The matrix $(\mathbf{I} - \mathbf{B})$ is non-singular. $\boldsymbol{\zeta}_i$ (zeta) is the disturbance vector that is assumed to have an expected value of zero [$\mathbb{E}(\boldsymbol{\zeta}_i) = \mathbf{0}$] and which is uncorrelated with $\boldsymbol{\xi}_i$.

The second component of the general system is the measurement model:

Measurement Model

$$\mathbf{y}_i = \boldsymbol{\Lambda}_y \boldsymbol{\eta}_i + \boldsymbol{\epsilon}_i \tag{2}$$
$$\mathbf{x}_i = \boldsymbol{\Lambda}_x \boldsymbol{\xi}_i + \boldsymbol{\delta}_i \tag{3}$$

The $\mathbf{y}_i$ $(p \times 1)$ and the $\mathbf{x}_i$ $(q \times 1)$ vectors are observed variables, $\boldsymbol{\Lambda}_y$ $(p \times m)$ (Lambda) and $\boldsymbol{\Lambda}_x$ $(q \times n)$ are the coefficient matrices that show the relation of $\mathbf{y}_i$ to $\boldsymbol{\eta}_i$ and $\mathbf{x}_i$ to $\boldsymbol{\xi}_i$ respectively, and $\boldsymbol{\epsilon}_i$ $(p \times 1)$ (epsilon) and $\boldsymbol{\delta}_i$ $(q \times 1)$ (delta) are the errors of measurement for $\mathbf{y}_i$ and $\mathbf{x}_i$, respectively. The errors of measurement are assumed to be uncorrelated with $\boldsymbol{\eta}_i$, $\boldsymbol{\xi}_i$ and $\boldsymbol{\zeta}_i$ and with each other.

Also $\mathbb{V}(\boldsymbol{\xi}_i) = \boldsymbol{\Phi}$ (Phi), $\mathbb{V}(\boldsymbol{\zeta}_i) = \boldsymbol{\Psi}$ (Psi), $\mathbb{V}(\boldsymbol{\epsilon}_i) = \boldsymbol{\Theta}_\epsilon$ (Theta), $\mathbb{V}(\boldsymbol{\delta}_i) = \boldsymbol{\Theta}_\delta$ (Theta) and $\mathbb{V}\left(\begin{bmatrix}\mathbf{y}_i \\ \mathbf{x}_i\end{bmatrix}\right) = \boldsymbol{\Sigma}$ (Sigma).

1.3 Steps in SEM Modeling

Steps in SEM Modeling

1. *Specification*
2. *Identification*
3. *Estimation*
4. *Testing and Diagnostics*
5. *Respecification*

1.3.1 *Specification*

1. What latent variables?
2. Relation between latent variables?
3. What measures?
4. Relation between measures and latent variables?

Latent Variables: Variables of Interest But Not Directly Measureable

Common in Sciences: Intelligence, Worker Productivity, Diseases, Happiness, Value of House, Carrying Capacity, "Free" Market, Disturbance Variables, etc

Latent Variables: Variables of Interest But Not Directly Measureable

1.3.2 *Implied Covariance Matrix*

$$H_0 : \boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\boldsymbol{\theta})$$

$$\begin{aligned} \boldsymbol{\Sigma} &= \text{Population Covariance Matrix} \\ \boldsymbol{\theta} &= \text{Vector of Parameters} \\ \boldsymbol{\Sigma}(\boldsymbol{\theta}) &= \text{Model Implied Covariance Matrix} \end{aligned}$$

Each Model $\Rightarrow \boldsymbol{\Sigma}(\boldsymbol{\theta})$

1.3.3 *Identification*

1.3.3.1 *Introduction*

Unique values for parameters?

If $\boldsymbol{\Sigma}(\boldsymbol{\theta}_1) = \boldsymbol{\Sigma}(\boldsymbol{\theta}_2)$ then $\boldsymbol{\theta}_1 = \boldsymbol{\theta}_2$

1.3.3.2 *Establishing Identification*

- Algebraic Means: $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\boldsymbol{\theta})$ solve for $\boldsymbol{\theta}$
- Identification Rules
- Empirical Tests

1.3.4 *Estimation*

1.3.4.1 *Full Information*

- Maximum Likelihood (ML)

- Generalized Least Squares (GLS)
- Unweighted Least Squares (ULS)
- Weighted Least Squares (WLS)

1.3.4.2 *Limited Information*

- Two-Stage Least Squares (2SLS)

1.3.5 *Testing and Diagnostics*

- $H_0 : \boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\boldsymbol{\theta})$
  χ^2 Test
  $\chi^2_m = (N-1)$ (Fit Function Minimum)
  $df = \frac{1}{2}(p+q)(p+q+1) -$ No of parameters

- Overall Model Fit
  $\chi^2_b =$ Chi-square test statistics for baseline model
  $\chi^2_m =$ Chi-square test statistics for hypothesized model
  $df_b =$ degrees of freedom for baseline model
  $df_m =$ degrees of freedom for hypothesized model
  Incremental Fit Index (IFI) $= \frac{\chi^2_b - \chi^2_m}{\chi^2_b - df_m}$
  Tucker Lewis index (TLI) $= \frac{\chi^2_b/df_b - \chi^2_m/df_m}{\chi^2_b/df_b - 1}$
  Root Mean Square Error of Approximation (RMSEA) $= \sqrt{\frac{\chi^2_m - df_m}{(N-1)df_m}}$

- Residuals $\left(\mathbf{S} - \boldsymbol{\Sigma}\left(\widehat{\boldsymbol{\theta}}\right)\right)$
- Component Fit
- Statistical Power

1.3.6 *Respecification*

- Substantive-Based Revisions
- Lagrangian Multiplier
- Wald
- Residuals $\left(\mathbf{S} - \boldsymbol{\Sigma}\left(\widehat{\boldsymbol{\theta}}\right)\right)$

Table 1: Notation for the General Structural Equation Model

Symbol	*Name*	*Dimension*	*Meaning*
N		1×1	Number of observations
m		1×1	Number of latent endogenous variables
n		1×1	Number of latent exogenous variables
p		1×1	Number of indicators of latent endogenous variables
q		1×1	Number of indicators of latent exogenous variable
$\boldsymbol{\eta}_i$	eta	$m \times 1$	Latent endogenous variables (for observation i)
$\boldsymbol{\xi}_i$	xi	$n \times 1$	Latent exogenous variables (for observation i)
$\boldsymbol{\zeta}_i$	zeta	$m \times 1$	Structural disturbances (errors in equations)
$\mathbf{B}$		$m \times m$	Structural parameters relating latent endogenous to endogenous variables
$\boldsymbol{\Gamma}$	Gamma	$n \times n$	Structural parameters relating latent endogenous to exogenous variables
$\mathbf{y}_i$		$p \times 1$	Indicators of latent endogenous variables
$\mathbf{x}_i$		$q \times 1$	Indicators of latent exogenous variables
$\boldsymbol{\epsilon}_i$	epsilon	$p \times 1$	Measurement errors in endogenous indicators
$\boldsymbol{\delta}_i$	delta	$q \times 1$	Measurement errors in exogenous indicators
$\boldsymbol{\Lambda}_y$	Lambda	$p \times m$	Factor loadings relating endogenous indicators to latent endogenous variables
$\boldsymbol{\Lambda}_x$	Lambda	$q \times n$	Factor loadings relating exogenous indicators to latent exogenous variables
$\boldsymbol{\Phi}$	Phi	$n \times n$	Covariances among latent exogenous variables
$\boldsymbol{\Psi}$	Psi	$m \times m$	Covariances among structural disturbances
$\boldsymbol{\Theta}_\epsilon$	Theta	$p \times p$	Covariances among measurement errors in endogenous indicators
$\boldsymbol{\Theta}_\delta$	Theta	$q \times q$	Covariances among measurement errors in exogenous indicators
$\boldsymbol{\Sigma}$	Sigma	$(p+q) \times (p+q)$	Covariances among observed (indicator) variables

2

LINEAR MODEL

2.1 INTRODUCTION

The model relating the normal dependent variable Y with the explanatory variables $X_1, X_2, \cdots, X_q$ is

$$Y = \gamma_1 X_1 + \gamma_2 X_2 + \cdots + \gamma_q X_q + \epsilon \tag{4}$$
$$Y = \mathbf{x}'\boldsymbol{\alpha} + \epsilon = \boldsymbol{\alpha}'\mathbf{x} + \epsilon \tag{5}$$

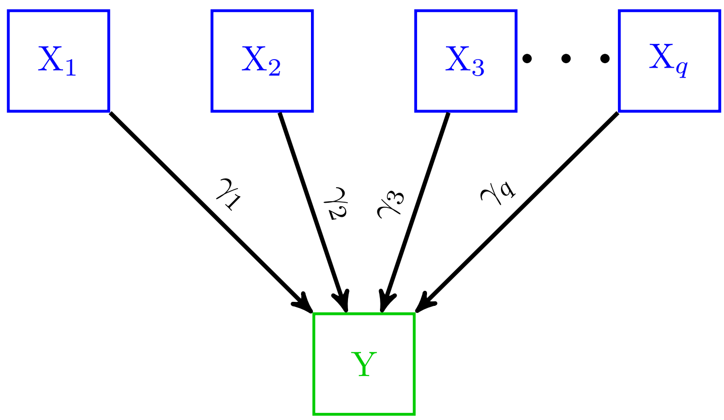

Figure 1: Path Diagram of Linear Model

Example 1. *A mortgage department of a large bank is studying its recent loans. Of particular interest is how such factors as the value of the home (in thousands of dollars), education level of the head of the household, age of the head of the household and current monthly mortgage payment (in dollars) relate to the family income. Are these variables effect predictors of the income of the household?*

2.2 LINEAR MODELING APPROACH

Linear Model (LM) can only measure direct effects

Linear Model (LM) can only measure direct effects

Example 2. *A mortgage department of a large bank is studying its recent loans. Of particular interest is how such factors as the value of the home (in thousands of dollars), education level of the head of the household, age of the head of the household and current monthly mortgage payment (in dollars) relate to the family income. Are these variables effect predictors of the income of the household?*

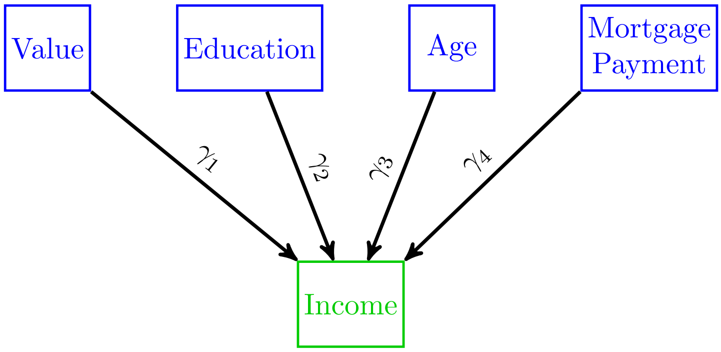

Figure 2: Path Diagram of Regression

```
load("Income.RData")
# Income
str(Income)
```

```
'data.frame':   25 obs. of  5 variables:
 $ Income         : num  40.3 39.6 40.8 40.3 40 38.1 40.4 40.7 40.8 37.1
 $ Value          : int  180 121 161 161 179 99 114 202 184 90 ...
 $ Education      : int  14 15 14 14 14 14 15 14 13 14 ...
 $ Age            : int  53 49 44 39 53 46 42 49 37 43 ...
 $ MortgagePayment: int  230 370 397 181 387 304 285 551 370 135 ...
```

```
Income.lm.fm1 <-
  lm(
      formula    = Income ~ Value + Education + Age +
                            MortgagePayment
    , data       = Income
  # , subset
  # , weights
  # , na.action
    , method      = "qr"
    , model       = TRUE
    , x           = FALSE
    , y           = FALSE
    , qr          = TRUE
    , singular.ok = TRUE
    , contrasts   = NULL
  # , offset
  # , ...
  )

summary(Income.lm.fm1)
```

```
Call:
```

```
lm(formula = Income ~ Value + Education + Age + MortgagePayment,
    data = Income, method = "qr", model = TRUE, x = FALSE, y = FALSE,
    qr = TRUE, singular.ok = TRUE, contrasts = NULL)

Residuals:
     Min       1Q   Median       3Q      Max
-1.05615 -0.37792 -0.05208  0.47685  1.20372

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)     28.438187   3.442601   8.261 7.08e-08 ***
Value            0.032302   0.005712   5.656 1.55e-05 ***
Education        0.607859   0.277795   2.188   0.0407 *
Age             -0.037207   0.035780  -1.040   0.3108
MortgagePayment -0.001345   0.001449  -0.928   0.3644
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.685 on 20 degrees of freedom
Multiple R-squared:  0.6463,    Adjusted R-squared:  0.5756
F-statistic: 9.137 on 4 and 20 DF,  p-value: 0.0002286
```

```
summary(Income.lm.fm1)$coef
```

```
                    Estimate  Std. Error    t value     Pr(>|t|)
(Intercept)     28.438187019 3.442600555  8.2606700 7.077863e-08
Value            0.032301977 0.005711513  5.6555905 1.553422e-05
Education        0.607858710 0.277794920  2.1881563 4.069801e-02
Age             -0.037207429 0.035780364 -1.0398840 3.108026e-01
MortgagePayment -0.001345319 0.001449331 -0.9282344 3.643539e-01
```

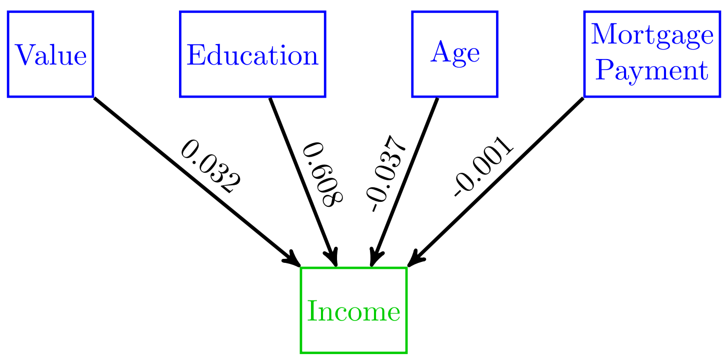

Figure 3: Path Diagram of Regression with estimates of coefficient obtained from *lm*

Strucural Equation Model (SEM) can measure direct as well as indirect effects

Strucural Equation Model (SEM) can measure direct as well as indirect effects

Example 3. *A mortgage department of a large bank is studying its recent loans. Of particular interest is how such factors as the value of the home (in thousands of dollars), education level of the head of the household, age of the head of the household and current monthly mortgage payment (in dollars) relate to the family income. Are these variables effect predictors of the income of the household?*

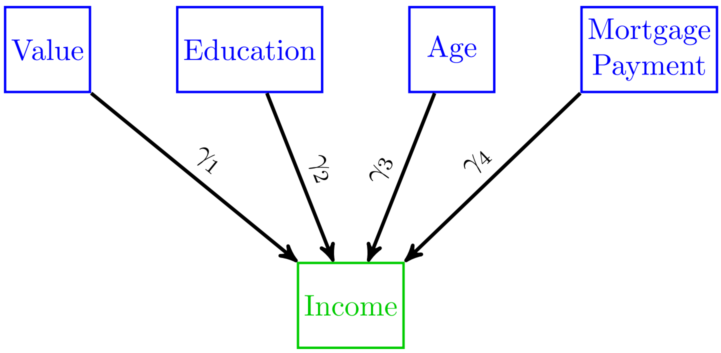

Figure 4: Path Diagram of Regression

```
library(lavaan)

Income.sem.m1 <- '
  # Regressions
    Income ~ gamma1 * Value + gamma2 * Education +
             gamma3 * Age   + gamma4 * MortgagePayment
'


Income.sem.fm1 <-
  lavaan::sem(
      model            = Income.sem.m1
    , data             = Income
    , ordered          = NULL
    , sampling.weights = NULL
    , sample.cov       = NULL
    , sample.mean      = NULL
    , sample.nobs      = NULL
    , group            = NULL
    , cluster          = NULL
  # , constraints      = ""
    , WLS.V            = NULL
    , NACOV            = NULL
```

```r
  # ,   ...
  )

# anova(Income.sem.fm1)
# coef(Income.sem.fm1)
parameterEstimates(Income.sem.fm1, standardized = TRUE)
```

```
               lhs op             rhs  label       est    se
1           Income  ~           Value gamma1     0.032 0.005
2           Income  ~       Education gamma2     0.608 0.248
3           Income  ~             Age gamma3    -0.037 0.032
4           Income  ~ MortgagePayment gamma4    -0.001 0.001
5           Income ~~          Income            0.375 0.106
6            Value ~~           Value          773.526 0.000
7            Value ~~       Education           -2.619 0.000
8            Value ~~             Age           30.408 0.000
9            Value ~~ MortgagePayment         1087.181 0.000
10       Education ~~       Education            0.458 0.000
11       Education ~~             Age            2.216 0.000
12       Education ~~ MortgagePayment          -14.822 0.000
13             Age ~~             Age           27.840 0.000
14             Age ~~ MortgagePayment          -18.584 0.000
15 MortgagePayment ~~ MortgagePayment        10739.898 0.000
        z pvalue  ci.lower  ci.upper    std.lv std.all   std.nox
1   6.323  0.000     0.022     0.042     0.032   0.872     0.031
2   2.446  0.014     0.121     1.095     0.608   0.399     0.590
3  -1.163  0.245    -0.100     0.026    -0.037  -0.191    -0.036
4  -1.038  0.299    -0.004     0.001    -0.001  -0.135    -0.001
5   3.536  0.000     0.167     0.583     0.375   0.354     0.354
6      NA     NA   773.526   773.526   773.526   1.000   773.526
7      NA     NA    -2.619    -2.619    -2.619  -0.139    -2.619
8      NA     NA    30.408    30.408    30.408   0.207    30.408
9      NA     NA  1087.181  1087.181  1087.181   0.377  1087.181
10     NA     NA     0.458     0.458     0.458   1.000     0.458
11     NA     NA     2.216     2.216     2.216   0.621     2.216
12     NA     NA   -14.822   -14.822   -14.822  -0.211   -14.822
13     NA     NA    27.840    27.840    27.840   1.000    27.840
14     NA     NA   -18.584   -18.584   -18.584  -0.034   -18.584
15     NA     NA 10739.898 10739.898 10739.898   1.000 10739.898
```

```r
# fitmeasures(Income.sem.fm1)
fitted(Income.sem.fm1)$cov
```

```
                Income    Value     Eductn    Age
Income              1.061
Value              20.800   773.526
```

```
Education              0.131     -2.619      0.458
Age                    1.318     30.408      2.216     27.840
MortgagePayment       12.351   1087.181    -14.822    -18.584
                 MrtggP
Income
Value
Education
Age
MortgagePayment 10739.898
```

```r
# residuals(Income.sem.fm1, type = "cor")
# modificationIndices(Income.sem.fm1)
var(Income)
```

```
                   Income       Value   Education        Age
Income           1.105433   21.667000   0.1365000   1.373333
Value           21.667000  805.756667  -2.7283333  31.675000
Education        0.136500   -2.728333   0.4766667   2.308333
Age              1.373333   31.675000   2.3083333  29.000000
MortgagePayment 12.865667 1132.480000 -15.4400000 -19.358333
                MortgagePayment
Income                 12.86567
Value                1132.48000
Education             -15.44000
Age                   -19.35833
MortgagePayment     11187.39333
```

```r
cor(Income)
```

```
                   Income      Value  Education         Age
Income          1.0000000  0.7259899  0.1880438  0.24255525
Value           0.7259899  1.0000000 -0.1392157  0.20721237
Education       0.1880438 -0.1392157  1.0000000  0.62085779
Age             0.2425553  0.2072124  0.6208578  1.00000000
MortgagePayment 0.1156915  0.3771934 -0.2114343 -0.03398635
                MortgagePayment
Income               0.11569153
Value                0.37719344
Education           -0.21143430
Age                 -0.03398635
MortgagePayment      1.00000000
```

```r
# vcov(Income.sem.fm1)


summary(
```

```
    object        = Income.sem.fm1
  , header        = TRUE
  , fit.measures = FALSE
  , estimates     = TRUE
  , ci            = FALSE
  , fmi           = FALSE
  , standardized = TRUE
  , rsquare       = TRUE
  , std.nox       = FALSE
  , modindices    = FALSE
  , nd            = 3L
)
```

```
lavaan 0.6-3 ended normally after 20 iterations

  Optimization method                           NLMINB
  Number of free parameters                          5

  Number of observations                            25

  Estimator                                         ML
  Model Fit Test Statistic                       0.000
  Degrees of freedom                                 0

Parameter Estimates:

  Information                                 Expected
  Information saturated (h1) model          Structured
  Standard Errors                             Standard

Regressions:
                   Estimate   Std.Err  z-value  P(>|z|)
  Income ~
    Value   (gmm1)    0.032    0.005    6.323    0.000
    Educatn (gmm2)    0.608    0.248    2.446    0.014
    Age     (gmm3)   -0.037    0.032   -1.163    0.245
    MrtggPy (gmm4)   -0.001    0.001   -1.038    0.299
   Std.lv   Std.all

     0.032    0.872
     0.608    0.399
    -0.037   -0.191
    -0.001   -0.135

Variances:
                   Estimate   Std.Err  z-value  P(>|z|)
```

```
   .Income             0.375    0.106    3.536    0.000
   Std.lv   Std.all
     0.375    0.354

R-Square:
                   Estimate
    Income            0.646
```

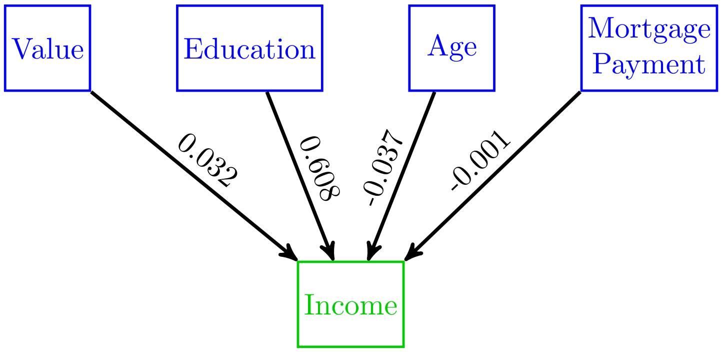

Figure 5: Path Diagram of Regression with estimates of coefficient obtained from *sem*

Example 4. *A mortgage department of a large bank is studying its recent loans. Of particular interest is how such factors as the value of the home (in thousands of dollars), education level of the head of the household, age of the head of the household and current monthly mortgage payment (in dollars) relate to the family income. Are these variables effect predictors of the income of the household?*

Complete path diagram with variances and covariances/correlations

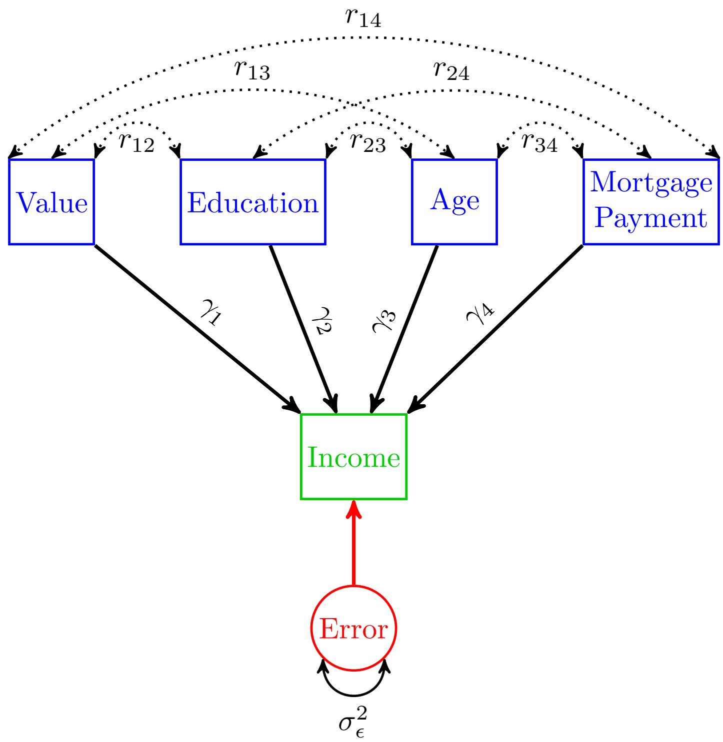

Figure 6: Path Diagram of Regression

```
library(lavaan)
Income.sem.m2 <- '
  # Regressions
    Income ~ gamma1 * Value + gamma2 * Education +
             gamma3 * Age   + gamma4 * MortgagePayment

  # Variances and Covariances
    Value     ~~ r12 * Education
    Value     ~~ r13 * Age
    Value     ~~ r14 * MortgagePayment
    Education ~~ r23 * Age
    Education ~~ r24 * MortgagePayment
    Age       ~~ r34 * MortgagePayment
    Income    ~~ sigma2I * Income
'

Income.sem.fm2 <-
  lavaan::sem(
      model            = Income.sem.m2
    , data             = Income
    , ordered          = NULL
    , sampling.weights = NULL
    , sample.cov       = NULL
    , sample.mean      = NULL
```

```r
     , sample.nobs      = NULL
     , group            = NULL
     , cluster          = NULL
   # , constraints      = ""
     , WLS.V            = NULL
     , NACOV            = NULL
   # ,   ...
   )

# anova(Income.sem.fm2)
# coef(Income.sem.fm2)
parameterEstimates(Income.sem.fm2, standardized = TRUE)
```

```
               lhs op             rhs   label       est       se
1           Income  ~           Value  gamma1     0.032    0.005
2           Income  ~       Education  gamma2     0.608    0.248
3           Income  ~             Age  gamma3    -0.037    0.032
4           Income  ~ MortgagePayment  gamma4    -0.001    0.001
5            Value ~~       Education     r12    -2.619    3.799
6            Value ~~             Age     r13    30.408   29.973
7            Value ~~ MortgagePayment     r14  1087.181  616.102
8        Education ~~             Age     r23     2.216    0.840
9        Education ~~ MortgagePayment     r24   -14.822   14.331
10             Age ~~ MortgagePayment     r34   -18.584  109.425
11          Income ~~          Income sigma2I     0.375    0.106
12           Value ~~           Value           773.526  218.786
13       Education ~~       Education             0.458    0.129
14             Age ~~             Age            27.840    7.874
15 MortgagePayment ~~ MortgagePayment         10739.898 3037.702
        z pvalue ci.lower   ci.upper    std.lv std.all std.nox
1   6.323  0.000    0.022      0.042     0.032   0.872   0.872
2   2.446  0.014    0.121      1.095     0.608   0.399   0.399
3  -1.163  0.245   -0.100      0.026    -0.037  -0.191  -0.191
4  -1.038  0.299   -0.004      0.001    -0.001  -0.135  -0.135
5  -0.689  0.491  -10.065      4.827    -2.619  -0.139  -0.139
6   1.015  0.310  -28.338     89.154    30.408   0.207   0.207
7   1.765  0.078 -120.358   2294.719  1087.181   0.377   0.377
8   2.637  0.008    0.569      3.863     2.216   0.621   0.621
9  -1.034  0.301  -42.910     13.265   -14.822  -0.211  -0.211
10 -0.170  0.865 -233.052    195.884   -18.584  -0.034  -0.034
11  3.536  0.000    0.167      0.583     0.375   0.354   0.354
12  3.536  0.000  344.713   1202.340   773.526   1.000   1.000
13  3.536  0.000    0.204      0.711     0.458   1.000   1.000
14  3.536  0.000   12.407     43.273    27.840   1.000   1.000
15  3.536  0.000 4786.112  16693.684 10739.898   1.000   1.000
```

```r
# fitmeasures(Income.sem.fm2)
fitted(Income.sem.fm2)$cov
```

```
                Income    Value     Eductn    Age
Income             1.061
Value             20.800   773.526
Education          0.131    -2.619     0.458
Age                1.318    30.408     2.216    27.840
MortgagePayment   12.351  1087.181   -14.822   -18.584
                MrtggP
Income
Value
Education
Age
MortgagePayment 10739.898
```

```r
# residuals(Income.sem.fm2, type = "cor")
# modificationIndices(Income.sem.fm2)
var(Income)
```

```
                   Income       Value   Education         Age
Income           1.105433   21.667000   0.1365000    1.373333
Value           21.667000  805.756667  -2.7283333   31.675000
Education        0.136500   -2.728333   0.4766667    2.308333
Age              1.373333   31.675000   2.3083333   29.000000
MortgagePayment 12.865667 1132.480000 -15.4400000 -19.358333
                MortgagePayment
Income                 12.86567
Value                1132.48000
Education             -15.44000
Age                   -19.35833
MortgagePayment     11187.39333
```

```r
cor(Income)
```

```
                   Income      Value   Education         Age
Income          1.0000000  0.7259899  0.1880438  0.24255525
Value           0.7259899  1.0000000 -0.1392157  0.20721237
Education       0.1880438 -0.1392157  1.0000000  0.62085779
Age             0.2425553  0.2072124  0.6208578  1.00000000
MortgagePayment 0.1156915  0.3771934 -0.2114343 -0.03398635
                MortgagePayment
Income               0.11569153
Value                0.37719344
Education           -0.21143430
Age                 -0.03398635
MortgagePayment      1.00000000
```

```r
# vcov(Income.sem.fm2)



summary(
    object       = Income.sem.fm2
  , header       = TRUE
  , fit.measures = TRUE
  , estimates    = TRUE
  , ci           = FALSE
  , fmi          = FALSE
  , standardized = TRUE
  , rsquare      = TRUE
  , std.nox      = FALSE
  , modindices   = FALSE
  , nd           = 3L
)
```

```
lavaan 0.6-3 ended normally after 114 iterations

  Optimization method                           NLMINB
  Number of free parameters                         15

  Number of observations                            25

  Estimator                                         ML
  Model Fit Test Statistic                       0.000
  Degrees of freedom                                 0
  Minimum Function Value               0.0000000000000

Model test baseline model:

  Minimum Function Test Statistic               47.107
  Degrees of freedom                                10
  P-value                                        0.000

User model versus baseline model:

  Comparative Fit Index (CFI)                    1.000
  Tucker-Lewis Index (TLI)                       1.000

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)               -385.524
  Loglikelihood unrestricted model (H1)       -385.524
```

```
  Number of free parameters                         15
  Akaike (AIC)                                 801.048
  Bayesian (BIC)                               819.331
  Sample-size adjusted Bayesian (BIC)          772.815

Root Mean Square Error of Approximation:

  RMSEA                                          0.000
  90 Percent Confidence Interval          0.000  0.000
  P-value RMSEA <= 0.05                             NA

Standardized Root Mean Square Residual:

  SRMR                                           0.000

Parameter Estimates:

  Information                                 Expected
  Information saturated (h1) model          Structured
  Standard Errors                             Standard

Regressions:
                   Estimate   Std.Err  z-value  P(>|z|)
  Income ~
    Value   (gmm1)     0.032    0.005    6.323    0.000
    Educatn (gmm2)     0.608    0.248    2.446    0.014
    Age     (gmm3)    -0.037    0.032   -1.163    0.245
    MrtggPy (gmm4)    -0.001    0.001   -1.038    0.299
   Std.lv   Std.all

     0.032    0.872
     0.608    0.399
    -0.037   -0.191
    -0.001   -0.135

Covariances:
                   Estimate   Std.Err  z-value  P(>|z|)
  Value ~~
    Educatin (r12)    -2.619    3.799   -0.689    0.491
    Age      (r13)    30.408   29.973    1.015    0.310
    MrtggPym (r14)  1087.181  616.102    1.765    0.078
  Education ~~
    Age      (r23)     2.216    0.840    2.637    0.008
    MrtggPym (r24)   -14.822   14.331   -1.034    0.301
  Age ~~
    MrtggPym (r34)   -18.584  109.425   -0.170    0.865
```

```
   Std.lv   Std.all

    -2.619   -0.139
    30.408    0.207
  1087.181    0.377

     2.216    0.621
   -14.822   -0.211

   -18.584   -0.034

Variances:
                   Estimate   Std.Err  z-value  P(>|z|)
   .Income  (sg2I)     0.375    0.106    3.536    0.000
    Value            773.526  218.786    3.536    0.000
    Educatn            0.458    0.129    3.536    0.000
    Age               27.840    7.874    3.536    0.000
    MrtggPy        10739.898 3037.702    3.536    0.000
   Std.lv   Std.all
     0.375    0.354
   773.526    1.000
     0.458    1.000
    27.840    1.000
 10739.898    1.000

R-Square:
                   Estimate
    Income             0.646
```

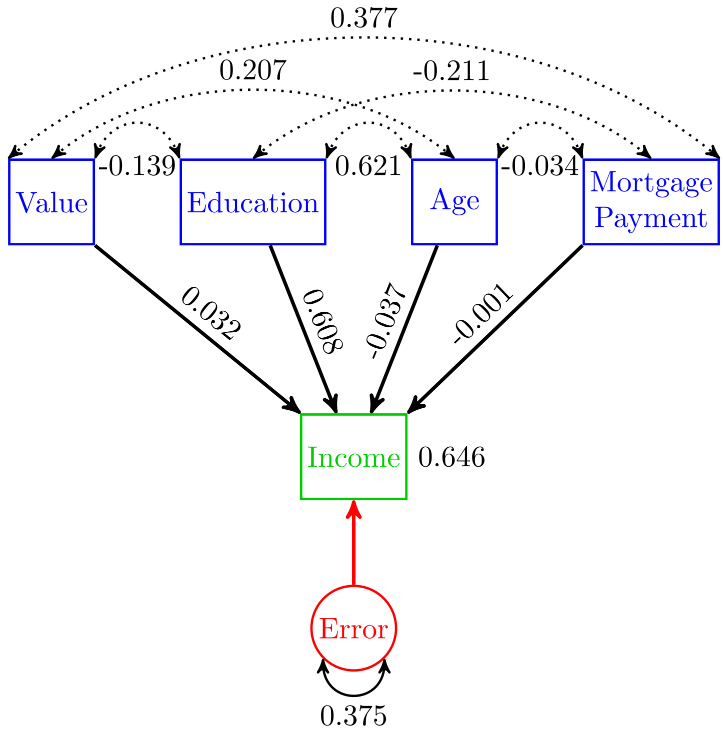

Figure 7: Path Diagram of Regression with estimates of coefficient

3 MULTIVARIATE LINEAR MODEL

3.1 INTRODUCTION

The model relating the normal dependent variables $Y_1, Y_2, \cdots, Y_p$ with the explanatory variables $X_1, X_2, \cdots, X_q$ is

$$\begin{aligned}
Y_1 &= \gamma_{11}X_1 + \gamma_{12}X_2 + \cdots + \gamma_{1q}X_q + \epsilon_1 \\
Y_2 &= \gamma_{21}X_1 + \gamma_{22}X_2 + \cdots + \gamma_{2q}X_q + \epsilon_2 \\
\vdots \quad & \quad \vdots \qquad\quad \vdots \qquad\quad \vdots \qquad\quad \vdots \qquad\quad \vdots \\
Y_p &= \gamma_{p1}X_1 + \gamma_{p2}X_2 + \cdots + \gamma_{pq}X_q + \epsilon_p
\end{aligned}$$

In matrix form the multivariate linear model may be written as

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_p \end{bmatrix} = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \dots & \gamma_{1q} \\ \gamma_{21} & \gamma_{22} & \dots & \gamma_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{p1} & \gamma_{p2} & \dots & \gamma_{pq} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_q \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_p \end{bmatrix} \tag{6}$$

$$\mathbf{y} = \boldsymbol{\Gamma}\mathbf{x} + \boldsymbol{\epsilon} \tag{7}$$

Therefore, the multivariate linear model for the i-th observation is

$$\mathbf{y}_i = \boldsymbol{\Gamma}\boldsymbol{x}_i + \boldsymbol{\epsilon}_i \tag{8}$$

The $\mathbf{y}_i$ $(p \times 1)$ and the $\mathbf{x}_i$ $(q \times 1)$ vectors are observed variables, $\boldsymbol{\Gamma}$ (Gamma) is the $p \times q$ coefficient matrix for the effects of $\mathbf{x}_i$ on $\mathbf{y}_i$. $\boldsymbol{\epsilon}_i$ (epsilon) is the disturbance vector that is assumed to have an expected value of zero $[\mathbb{E}(\boldsymbol{\epsilon}_i) = \mathbf{o}]$.

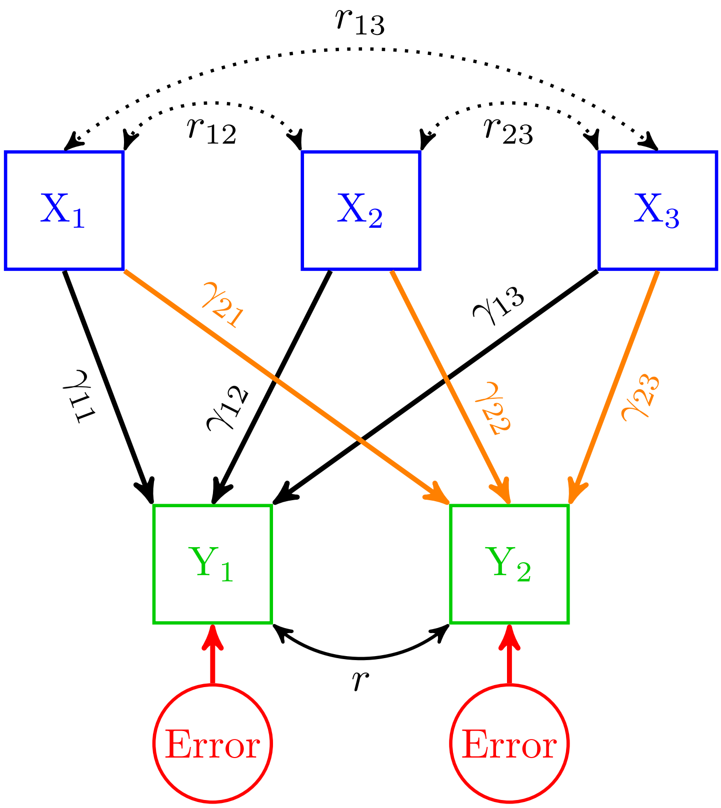

Figure 8: Path Diagram for Multivariate Linear Model

4 STRUCTURAL EQUATION MODELING WITH OBSERVED VARIABLES

4.1 INTRODUCTION

The general model of structural equations with observed variables is

$$
\begin{array}{lllllllllll}
Y_1 & = \beta_{12}Y_2 & + \beta_{13}Y_3 & + \cdots + & \beta_{1p}Y_p & & + \gamma_{11}X_1 & + \gamma_{12}X_2 & + \cdots + & \gamma_{1q}X_q & + \epsilon_1 \\
Y_2 & = \beta_{21}Y_1 & + \beta_{23}Y_3 & + \cdots + & \beta_{2p}Y_p & & + \gamma_{21}X_1 & + \gamma_{22}X_2 & + \cdots + & \gamma_{2q}X_q & + \epsilon_2 \\
\vdots & \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots \\
Y_p & = \beta_{p1}Y_1 & + \beta_{p2}Y_2 & + \cdots + & \beta_{p(p-1)}Y_{p-1} & & + \gamma_{p1}X_1 & + \gamma_{p2}X_2 & + \cdots + & \gamma_{pq}X_q & + \epsilon_p
\end{array}
$$

In matrix form the general model of structural equations with observed variables may be written as

$$
\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_p \end{bmatrix} = \begin{bmatrix} 0 & \beta_{12} & \ldots & \beta_{1p} \\ \beta_{21} & 0 & \ldots & \beta_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{p1} & \beta_{p2} & \ldots & 0 \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_p \end{bmatrix} + \begin{bmatrix} \gamma_{11} & \gamma_{12} & \ldots & \gamma_{1q} \\ \gamma_{21} & \gamma_{22} & \ldots & \gamma_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{p1} & \gamma_{p2} & \ldots & \gamma_{pq} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_q \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_p \end{bmatrix} \tag{9}
$$

$$
\mathbf{y} = \mathbf{B}\mathbf{y} + \boldsymbol{\Gamma}\mathbf{x} + \boldsymbol{\epsilon} \tag{10}
$$

Therefore, the general model of structural equations with observed variables for the i-th observation is

$$
\mathbf{y}_i = \mathbf{B}\mathbf{y}_i + \boldsymbol{\Gamma}\boldsymbol{x}_i + \boldsymbol{\epsilon}_i \tag{11}
$$

The $\mathbf{y}_i$ $(p \times 1)$ and the $\mathbf{x}_i$ $(q \times 1)$ vectors are observed variables, $\mathbf{B}$ is the $p \times p$ coefficient matrix showing the influence of the endogenous variables on each other; $\boldsymbol{\Gamma}$ (Gamma) is the $p \times q$ coefficient matrix for the effects of $\mathbf{x}_i$ on $\mathbf{y}_i$. The matrix $(\mathbf{I} - \mathbf{B})$ is non-singular. $\boldsymbol{\epsilon}_i$ (zeta) is the disturbance vector that is assumed to have an expected value of zero $[\mathbb{E}(\boldsymbol{\epsilon}_i) = \mathbf{o}]$.

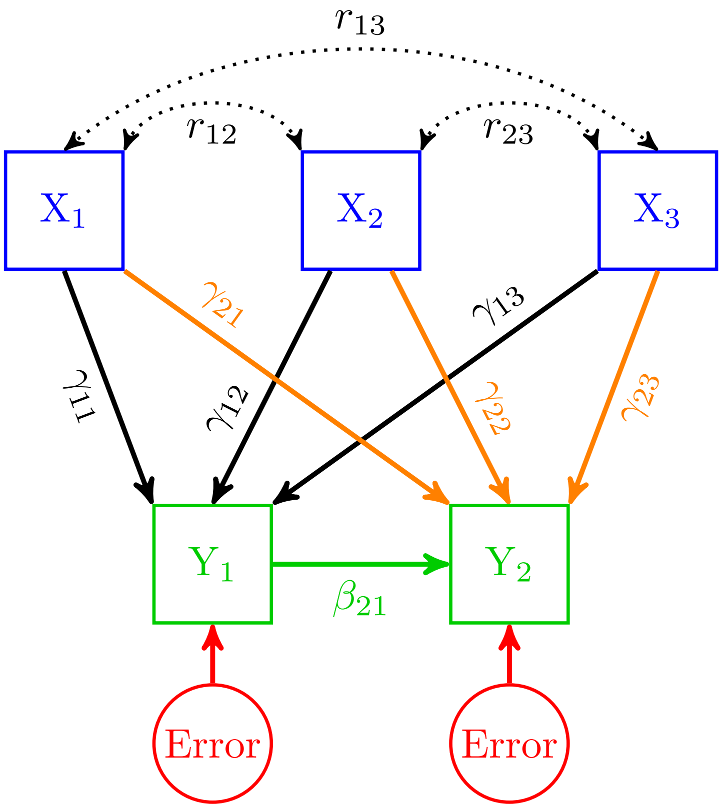

Figure 9: Path Diagram for Structural Equations Model with Observed Variables

Complete path diagram with direct as well as indirect effects and corresponding variances and covariances/correlations

Example 5. *A mortgage department of a large bank is studying its recent loans. Of particular interest is how such factors as the value of the home (in thousands of dollars), education level of the head of the household, age of the head of the household and current monthly mortgage payment (in dollars) relate to the family income. Are these variables effect predictors of the income of the household?*

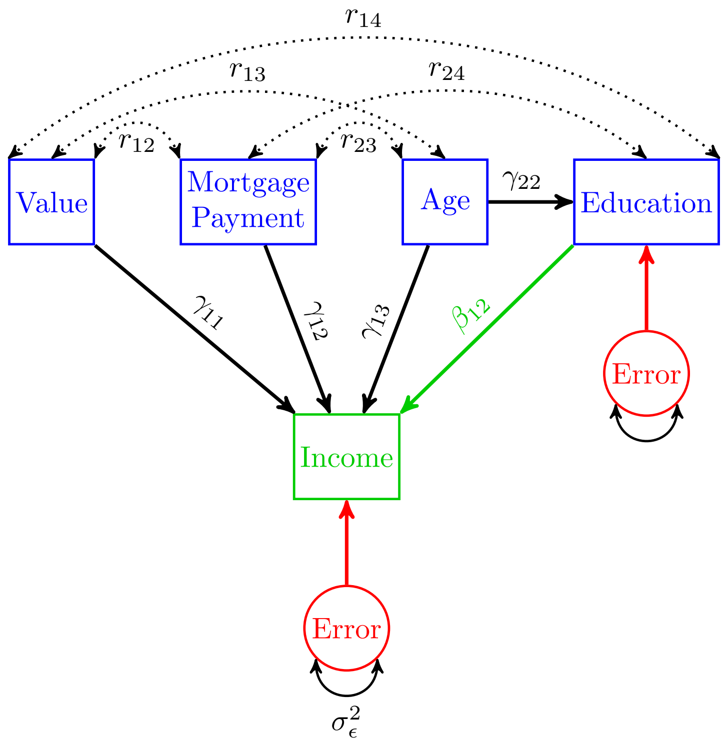

Figure 10: Path Diagram with Indirecet Effects

```r
load("Income.RData")
# Income
str(Income)
```

```
'data.frame':   25 obs. of  5 variables:
 $ Income         : num  40.3 39.6 40.8 40.3 40 38.1 40.4 40.7 40.8 37.1 ...
 $ Value          : int  180 121 161 161 179 99 114 202 184 90 ...
 $ Education      : int  14 15 14 14 14 14 15 14 13 14 ...
 $ Age            : int  53 49 44 39 53 46 42 49 37 43 ...
 $ MortgagePayment: int  230 370 397 181 387 304 285 551 370 135 ...
```

```r
library(lavaan)

Income.sem.m3 <- '
  # Regressions
    Income    ~ beta12 * Education + gamma11 * Value +
                gamma12 * MortgagePayment + gamma13 * Age
    Education ~ gamma22 * Age

  # Variances and Covariances
    Value           ~~ r12 * MortgagePayment
    Value           ~~ r13 * Age
    Value           ~~ r14 * Education
    MortgagePayment ~~ r23 * Age
```

```
    MortgagePayment ~~ r24 *Education
    Education       ~~ sigma2E * Education
    Income          ~~ sigma2I * Income

# Indirect Effects
  IndEf := gamma22*beta12

# Total Effects (Direct + Indirect Effets)
  TotEf := gamma13 + (gamma22*beta12)
'


Income.sem.fm3 <-
  lavaan::sem(
      model              = Income.sem.m3
    , data               = Income
    , ordered            = NULL
    , sampling.weights = NULL
    , sample.cov         = NULL
    , sample.mean        = NULL
    , sample.nobs        = NULL
    , group              = NULL
    , cluster            = NULL
  # , constraints        = ""
    , WLS.V              = NULL
    , NACOV              = NULL
  # ,   ...
  )

# anova(Income.sem.fm3)
# coef(Income.sem.fm3)
parameterEstimates(Income.sem.fm3, standardized = TRUE)
```

```
               lhs op             rhs   label      est
1           Income  ~       Education  beta12    0.608
2           Income  ~           Value gamma11    0.032
3           Income  ~ MortgagePayment gamma12   -0.001
4           Income  ~             Age gamma13   -0.037
5        Education  ~             Age gamma22    0.080
6            Value ~~ MortgagePayment     r12 1087.181
7            Value ~~             Age     r13   30.408
8        Education ~~           Value     r14   -5.040
9  MortgagePayment ~~             Age     r23  -18.584
10       Education ~~ MortgagePayment     r24  -13.343
11       Education ~~       Education sigma2E    0.281
12          Income ~~          Income sigma2I    0.375
13           Value ~~           Value          773.526
```

```
14 MortgagePayment ~~         MortgagePayment        10739.898
15             Age ~~                     Age           27.840
16           IndEf :=         gamma22*beta12   IndEf     0.048
17           TotEf := gamma13+(gamma22*beta12)   TotEf     0.011
         se      z pvalue ci.lower  ci.upper    std.lv std.all
1     0.248  2.446  0.014    0.121     1.095     0.608   0.399
2     0.005  6.323  0.000    0.022     0.042     0.032   0.872
3     0.001 -1.038  0.299   -0.004     0.001    -0.001  -0.135
4     0.032 -1.163  0.245   -0.100     0.026    -0.037  -0.191
5     0.020  3.960  0.000    0.040     0.119     0.080   0.621
6   616.102  1.765  0.078 -120.358  2294.719  1087.181   0.377
7    29.973  1.015  0.310  -28.338    89.154    30.408   0.207
8     3.057 -1.649  0.099  -11.031     0.951    -5.040  -0.342
9   109.425 -0.170  0.865 -233.052   195.884   -18.584  -0.034
10   11.304 -1.180  0.238  -35.499     8.813   -13.343  -0.243
11    0.080  3.536  0.000    0.125     0.437     0.281   0.615
12    0.106  3.536  0.000    0.167     0.583     0.375   0.354
13  218.786  3.536  0.000  344.713  1202.340   773.526   1.000
14 3037.702  3.536  0.000 4786.111 16693.684 10739.898   1.000
15    7.874  3.536  0.000   12.407    43.273    27.840   1.000
16    0.023  2.081  0.037    0.003     0.094     0.048   0.248
17    0.027  0.415  0.678   -0.042     0.064     0.011   0.057
   std.nox
1    0.399
2    0.872
3   -0.135
4   -0.191
5    0.621
6    0.377
7    0.207
8   -0.342
9   -0.034
10  -0.243
11   0.615
12   0.354
13   1.000
14   1.000
15   1.000
16   0.248
17   0.057
```

```r
# fitmeasures(Income.sem.fm3)
fitted(Income.sem.fm3)$cov
```

```
                Income    Eductn    Value     MrtggP
Income              1.061
Education           0.131     0.458
```

```
Value                20.800    -2.619   773.526
MortgagePayment      12.351   -14.822  1087.181 10739.898
Age                   1.318     2.216    30.408   -18.584
                Age
Income
Education
Value
MortgagePayment
Age              27.840
```

```r
# residuals(Income.sem.fm3, type = "cor")
# modificationIndices(Income.sem.fm3)
var(Income)
```

```
                   Income       Value   Education        Age
Income           1.105433   21.667000   0.1365000   1.373333
Value           21.667000  805.756667  -2.7283333  31.675000
Education        0.136500   -2.728333   0.4766667   2.308333
Age              1.373333   31.675000   2.3083333  29.000000
MortgagePayment 12.865667 1132.480000 -15.4400000 -19.358333
                MortgagePayment
Income                 12.86567
Value                1132.48000
Education             -15.44000
Age                   -19.35833
MortgagePayment     11187.39333
```

```r
# vcov(Income.sem.fm3)



summary(
    object       = Income.sem.fm3
  , header       = TRUE
  , fit.measures = FALSE
  , estimates    = TRUE
  , ci           = FALSE
  , fmi          = FALSE
  , standardized = TRUE
  , rsquare      = TRUE
  , std.nox      = FALSE
  , modindices   = FALSE
  , nd           = 3L
)
```

```
lavaan 0.6-3 ended normally after 112 iterations
```

```
  Optimization method                           NLMINB
  Number of free parameters                         15

  Number of observations                            25

  Estimator                                         ML
  Model Fit Test Statistic                       0.000
  Degrees of freedom                                 0
  Minimum Function Value               0.0000000000000

Parameter Estimates:

  Information                                 Expected
  Information saturated (h1) model          Structured
  Standard Errors                             Standard

Regressions:
                   Estimate   Std.Err  z-value  P(>|z|)
  Income ~
    Educatn (bt12)     0.608    0.248    2.446    0.014
    Value   (gm11)     0.032    0.005    6.323    0.000
    MrtggPy (gm12)    -0.001    0.001   -1.038    0.299
    Age     (gm13)    -0.037    0.032   -1.163    0.245
  Education ~
    Age     (gm22)     0.080    0.020    3.960    0.000
   Std.lv   Std.all

     0.608    0.399
     0.032    0.872
    -0.001   -0.135
    -0.037   -0.191

     0.080    0.621

Covariances:
                     Estimate   Std.Err  z-value  P(>|z|)
  Value ~~
    MrtggPym (r12)    1087.181  616.102    1.765    0.078
    Age      (r13)      30.408   29.973    1.015    0.310
 .Education ~~
    Value    (r14)      -5.040    3.057   -1.649    0.099
  MortgagePayment ~~
    Age      (r23)     -18.584  109.425   -0.170    0.865
 .Education ~~
    MrtggPym (r24)     -13.343   11.304   -1.180    0.238
   Std.lv   Std.all
```

```
  1087.181    0.377
    30.408    0.207

    -5.040   -0.342

   -18.584   -0.034

   -13.343   -0.243

Variances:
                   Estimate   Std.Err  z-value  P(>|z|)
   .Educatn (sg2E)    0.281     0.080    3.536    0.000
   .Income  (sg2I)    0.375     0.106    3.536    0.000
    Value           773.526   218.786    3.536    0.000
    MrtggPy       10739.898  3037.702    3.536    0.000
    Age              27.840     7.874    3.536    0.000
   Std.lv   Std.all
     0.281    0.615
     0.375    0.354
   773.526    1.000
 10739.898    1.000
    27.840    1.000

R-Square:
                   Estimate
    Education         0.385
    Income            0.646

Defined Parameters:
                   Estimate   Std.Err  z-value  P(>|z|)
    IndEf             0.048     0.023    2.081    0.037
    TotEf             0.011     0.027    0.415    0.678
   Std.lv   Std.all
     0.048    0.248
     0.011    0.057
```

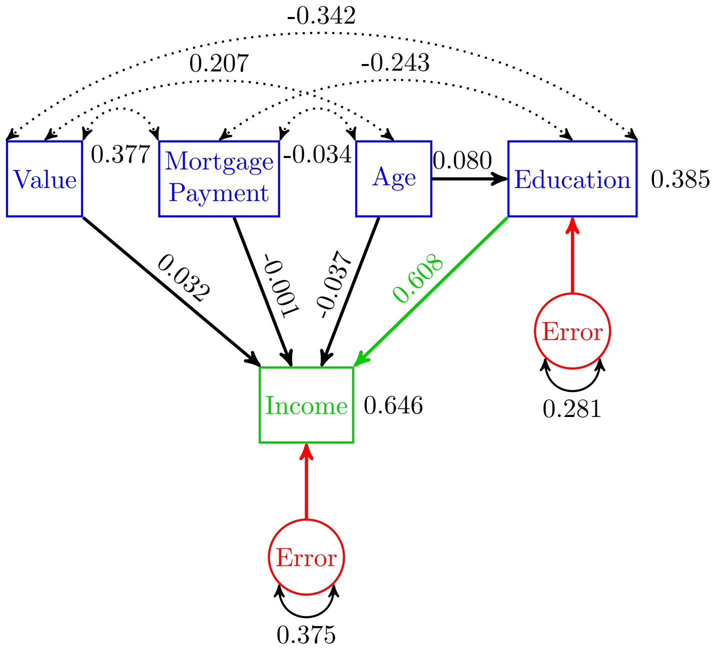

Figure 11: Path Diagram with estimates of coefficient including Indirecet Effects

Example 6. *Job Performance of Farm Managers*

Knowledge	:	*(26 Items)*	
Value	:	*(30 Items)*	*Tendency to rationally evaluate means to an economic end*
Satisfaction	:	*(11 Items)*	*Gratification from the managerial role*
Performance	:	*(24 Items)*	

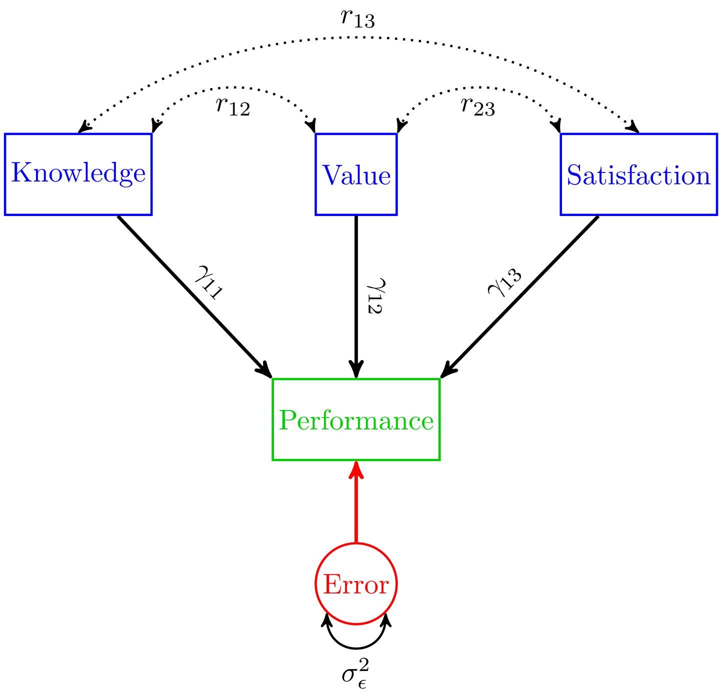

Figure 12: Path Diagram for job performance of farm managers

```r
library(lavaan)

Warren5V <-
  matrix(
    data =
      c(
        0.0209, 0.0177,  0.0245,     0.0046,
        0.0177, 0.0520,  0.0280,     0.0044,
        0.0245, 0.0280,  0.1212,    -0.0063,
        0.0046, 0.0044, -0.0063,     0.0901
      )
    , nrow  = 4
    , ncol  = 4
    , byrow = TRUE
    , dimnames = list(c("Performance",  "Knowledge",
                        "Value",    "Satisfaction")
                      , c("Performance",    "Knowledge",
                          "Value",  "Satisfaction"))
    )

Warren5V
```

```
             Performance Knowledge   Value Satisfaction
Performance       0.0209    0.0177  0.0245       0.0046
Knowledge         0.0177    0.0520  0.0280       0.0044
```

```
Value              0.0245    0.0280  0.1212      -0.0063
Satisfaction       0.0046    0.0044 -0.0063       0.0901
```

```
Warren5V.sem.m1 <- '
  # Regressions
    Performance ~ gamma11 * Knowledge + gamma12 * Value +
                  gamma13 * Satisfaction

  # Variances and Covariances
    Knowledge   ~~ r12 * Value
    Knowledge   ~~ r13 * Satisfaction
    Value       ~~ r23 * Satisfaction
    Performance ~~ sigma2P * Performance
'

Warren5V.sem.fm1 <-
  lavaan::sem(
      model            = Warren5V.sem.m1
  # , data
    , ordered          = NULL
    , sampling.weights = NULL
    , sample.cov       = Warren5V
    , sample.mean      = NULL
    , sample.nobs      = 98
    , group            = NULL
    , cluster          = NULL
  # , constraints      = ""
    , WLS.V            = NULL
    , NACOV            = NULL
  # ,   ...
  )

# methods(class = class(Warren5V.sem.fm1))
# anova(Warren5V.sem.fm1)
# coef(Warren5V.sem.fm1)
parameterEstimates(Warren5V.sem.fm1, standardized = TRUE)
```

```
            lhs op          rhs   label    est    se      z
1   Performance  ~    Knowledge gamma11  0.258 0.053  4.847
2   Performance  ~        Value gamma12  0.145 0.035  4.158
3   Performance  ~ Satisfaction gamma13  0.049 0.038  1.281
4     Knowledge ~~        Value     r12  0.028 0.008  3.293
5     Knowledge ~~ Satisfaction     r13  0.004 0.007  0.635
6         Value ~~ Satisfaction     r23 -0.006 0.010 -0.596
7   Performance ~~  Performance sigma2P  0.012 0.002  7.000
8     Knowledge ~~    Knowledge          0.051 0.007  7.000
9         Value ~~        Value          0.120 0.017  7.000
```

```
10 Satisfaction ~~ Satisfaction          0.089 0.013  7.000
   pvalue ci.lower ci.upper std.lv std.all std.nox
1   0.000    0.154    0.363  0.258   0.407   0.407
2   0.000    0.077    0.213  0.145   0.349   0.349
3   0.200   -0.026    0.123  0.049   0.101   0.101
4   0.001    0.011    0.044  0.028   0.353   0.353
5   0.525   -0.009    0.018  0.004   0.064   0.064
6   0.551   -0.027    0.014 -0.006  -0.060  -0.060
7   0.000    0.009    0.016  0.012   0.601   0.601
8   0.000    0.037    0.066  0.051   1.000   1.000
9   0.000    0.086    0.154  0.120   1.000   1.000
10  0.000    0.064    0.114  0.089   1.000   1.000
```

```r
# fitmeasures(Warren5V.sem.fm1)
fitted(Warren5V.sem.fm1)$cov
```

```
             Prfrmn Knwldg Value  Stsfct
Performance   0.021
Knowledge     0.018  0.051
Value         0.024  0.028  0.120
Satisfaction  0.005  0.004 -0.006  0.089
```

```r
# residuals(Warren5V.sem.fm1, type = "cor")
# modificationIndices(Warren5V.sem.fm1)
Warren5V
```

```
             Performance Knowledge   Value Satisfaction
Performance       0.0209    0.0177  0.0245       0.0046
Knowledge         0.0177    0.0520  0.0280       0.0044
Value             0.0245    0.0280  0.1212      -0.0063
Satisfaction      0.0046    0.0044 -0.0063       0.0901
```

```r
# vcov(Warren5V.sem.fm1)


summary(
    object       = Warren5V.sem.fm1
  , header       = TRUE
  , fit.measures = FALSE
  , estimates    = TRUE
  , ci           = FALSE
  , fmi          = FALSE
  , standardized = TRUE
  , rsquare      = TRUE
  , std.nox      = FALSE
  , modindices   = FALSE
```

```
  , nd            = 3L
)
```

```
lavaan 0.6-3 ended normally after 46 iterations

  Optimization method                           NLMINB
  Number of free parameters                         10

  Number of observations                            98

  Estimator                                         ML
  Model Fit Test Statistic                       0.000
  Degrees of freedom                                 0

Parameter Estimates:

  Information                                 Expected
  Information saturated (h1) model          Structured
  Standard Errors                             Standard

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv
  Performance ~
    Knowldg (gm11)    0.258    0.053    4.847    0.000    0.258
    Value   (gm12)    0.145    0.035    4.158    0.000    0.145
    Stsfctn (gm13)    0.049    0.038    1.281    0.200    0.049
  Std.all

    0.407
    0.349
    0.101

Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv
  Knowledge ~~
    Value    (r12)    0.028    0.008    3.293    0.001    0.028
    Satsfctn (r13)    0.004    0.007    0.635    0.525    0.004
  Value ~~
    Satsfctn (r23)   -0.006    0.010   -0.596    0.551   -0.006
  Std.all

    0.353
    0.064

   -0.060
```

```
Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv
   .Prfrmnc (sg2P)    0.012    0.002    7.000    0.000    0.012
    Knowldg           0.051    0.007    7.000    0.000    0.051
    Value             0.120    0.017    7.000    0.000    0.120
    Stsfctn           0.089    0.013    7.000    0.000    0.089
  Std.all
    0.601
    1.000
    1.000
    1.000

R-Square:
                   Estimate
    Performance       0.399
```

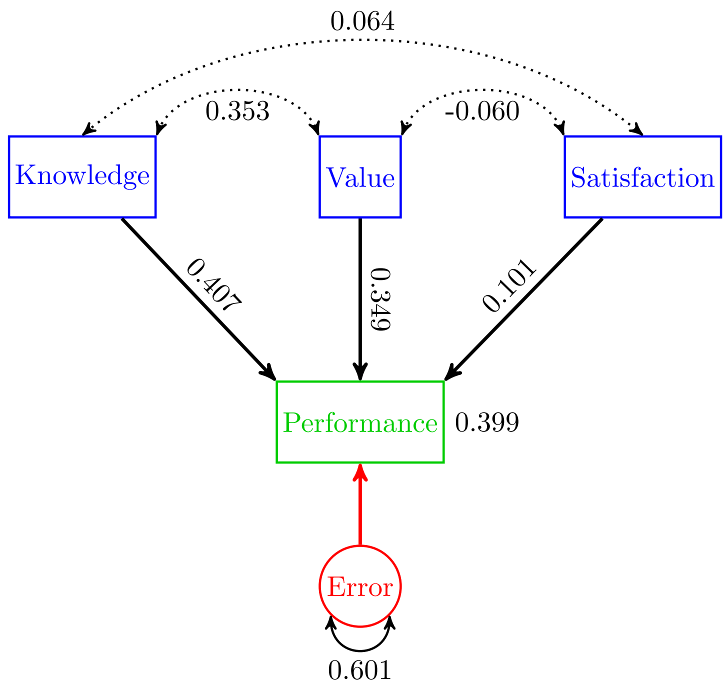

Figure 13: Path Diagram with estimates of coefficient for job performance of farm managers

5 FACTOR ANALYSIS

5.1 INTRODUCTION

The general model for confirmatory factor analysis is

$$\mathbf{y}_i = \boldsymbol{\Lambda}_y \boldsymbol{\eta}_i + \boldsymbol{\epsilon}_i \quad (12)$$

$$\mathbf{x}_i = \boldsymbol{\Lambda}_x \boldsymbol{\xi}_i + \boldsymbol{\delta}_i \quad (13)$$

The $\mathbf{y}_i$ $(p \times 1)$ and the $\mathbf{x}_i$ $(q \times 1)$ vectors are observed variables, $\boldsymbol{\Lambda}_y$ $(p \times m)$ (Lambda) and $\boldsymbol{\Lambda}_x$ $(q \times n)$ are the coefficient matrices that show the relation of $\mathbf{y}_i$ to $\boldsymbol{\eta}_i$ and $\mathbf{x}_i$ to $\boldsymbol{\xi}_i$ respectively, and $\boldsymbol{\epsilon}_i$ $(p \times 1)$ (epsilon) and $\boldsymbol{\delta}_i$ $(q \times 1)$ (delta) are the errors of measurement for $\mathbf{y}_i$ and $\mathbf{x}_i$, respectively. The errors of measurement are assumed to be uncorrelated with $\boldsymbol{\eta}_i$ and $\boldsymbol{\xi}_i$ and with each other.

Aptitude of High School students for Science and Arts

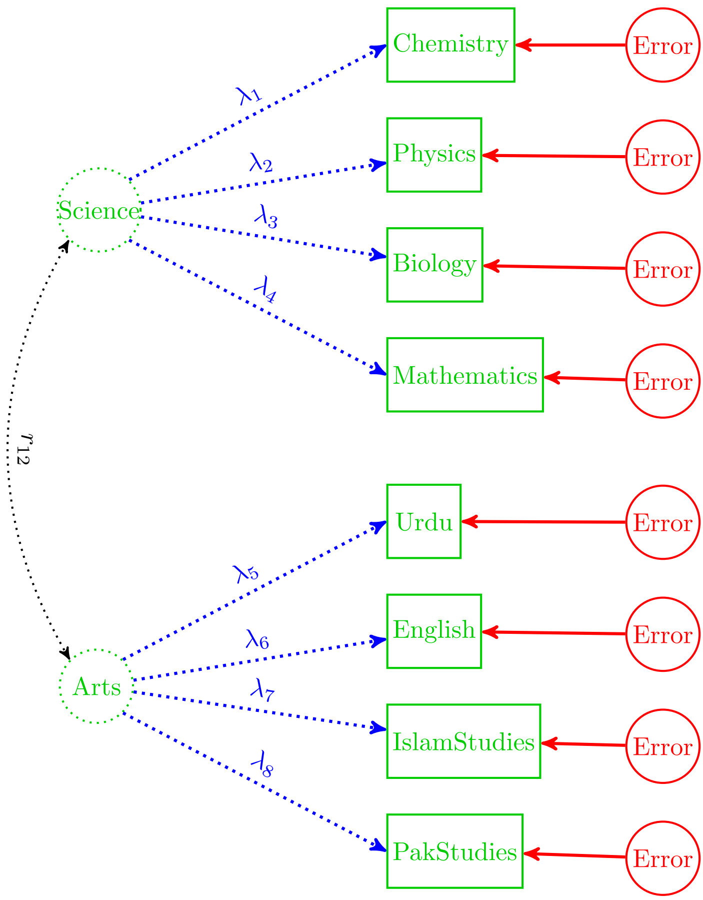

Figure 14: Path Diagram for Aptitude

Example 7. *Holzinger and Swineford (1939)*

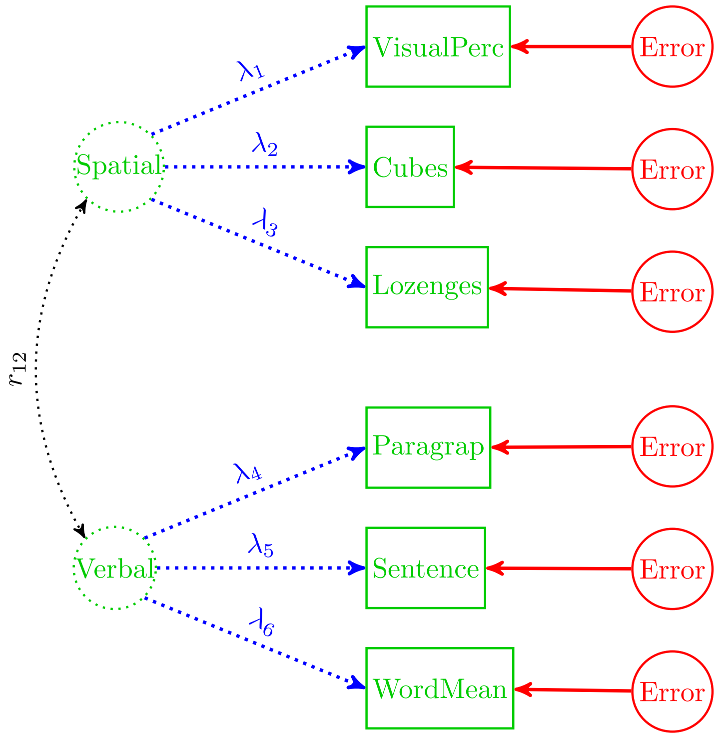

Figure 15: Path Diagram for Holzinger and Swineford (1939)

```
library(lavaan)

load("HSF.RData")
# HSF
str(HSF)
```

```
'data.frame':   73 obs. of  6 variables:
 $ VisualPerc: num  33 30 36 28 30 20 17 33 30 36 ...
 $ Cubes     : num  22 25 33 25 25 25 21 31 22 28 ...
 $ Lozenges  : num  17 20 36 9 11 6 6 30 20 22 ...
 $ Paragrap  : num  8 10 17 10 11 9 5 11 8 13 ...
 $ Sentence  : num  17 23 25 18 21 21 10 23 17 24 ...
 $ WordMean  : num  10 18 41 11 8 16 10 18 20 36 ...
```

```
HSF.sem.m1 <- '
  # Latent Variables
    Spatial =~ lambda1 * VisualPerc + lambda2 * Cubes +
               lambda3 * Lozenges
    Verbal  =~ lambda4 * Paragrap + lambda5 * Sentence +
               lambda6 * WordMean

  # Variances and Covariances
    Spatial  ~~ r12 * Verbal
'
```

```r
HSF.sem.fm1 <-
  lavaan::sem(
      model             = HSF.sem.m1
    , data              = HSF
    , ordered           = NULL
    , sampling.weights = NULL
    , sample.cov        = NULL
    , sample.mean       = NULL
    , sample.nobs       = NULL
    , group             = NULL
    , cluster           = NULL
  # , constraints       = ""
    , WLS.V             = NULL
    , NACOV             = NULL
  # ,   ...
  )

# methods(class = class(HSF.sem.fm1))
# anova(HSF.sem.fm1)
# coef(HSF.sem.fm1)
parameterEstimates(HSF.sem.fm1, standardized = TRUE)
```

```
          lhs op        rhs   label    est    se     z pvalue
1     Spatial =~ VisualPerc lambda1  1.000 0.000    NA     NA
2     Spatial =~      Cubes lambda2  0.610 0.142 4.279  0.000
3     Spatial =~   Lozenges lambda3  1.198 0.270 4.436  0.000
4      Verbal =~   Paragrap lambda4  1.000 0.000    NA     NA
5      Verbal =~   Sentence lambda5  1.334 0.159 8.379  0.000
6      Verbal =~   WordMean lambda6  2.234 0.262 8.541  0.000
7     Spatial ~~     Verbal     r12  7.315 2.553 2.865  0.004
8  VisualPerc ~~ VisualPerc         23.873 5.945 4.016  0.000
9       Cubes ~~      Cubes         11.602 2.566 4.521  0.000
10   Lozenges ~~   Lozenges         28.275 7.837 3.608  0.000
11   Paragrap ~~   Paragrap          2.834 0.863 3.286  0.001
12   Sentence ~~   Sentence          7.967 1.856 4.292  0.000
13   WordMean ~~   WordMean         19.925 4.917 4.052  0.000
14    Spatial ~~    Spatial         23.302 8.068 2.888  0.004
15     Verbal ~~     Verbal          9.682 2.144 4.516  0.000
   ci.lower ci.upper std.lv std.all std.nox
1     1.000    1.000  4.827   0.703   0.703
2     0.330    0.889  2.943   0.654   0.654
3     0.669    1.728  5.784   0.736   0.736
4     1.000    1.000  3.112   0.880   0.880
5     1.022    1.646  4.151   0.827   0.827
6     1.722    2.747  6.952   0.841   0.841
7     2.311   12.319  0.487   0.487   0.487
```

```
8    12.221   35.525 23.873   0.506   0.506
9     6.572   16.631 11.602   0.572   0.572
10   12.914   43.636 28.275   0.458   0.458
11    1.143    4.524  2.834   0.226   0.226
12    4.329   11.605  7.967   0.316   0.316
13   10.288   29.563 19.925   0.292   0.292
14    7.490   39.114  1.000   1.000   1.000
15    5.481   13.884  1.000   1.000   1.000
```

```
# fitmeasures(HSF.sem.fm1)
fitted(HSF.sem.fm1)$cov
```

```
           VslPrc Cubes  Lozngs Pargrp Sentnc WordMn
VisualPerc 47.175
Cubes      14.209 20.265
Lozenges   27.919 17.024 61.726
Paragrap    7.315  4.461  8.765 12.516
Sentence    9.759  5.950 11.692 12.916 25.197
WordMean   16.344  9.966 19.583 21.633 28.859 68.260
```

```
# residuals(HSF.sem.fm1, type = "cor")
# modificationIndices(HSF.sem.fm1)
var(HSF)
```

```
           VisualPerc     Cubes  Lozenges  Paragrap  Sentence
VisualPerc  47.829909 15.137938 26.899734  8.450723 12.820396
Cubes       15.137938 20.546804 17.658105  3.402207  4.092085
Lozenges    26.899734 17.658105 62.583714  9.181507 13.411339
Paragrap     8.450723  3.402207  9.181507 12.689878 13.042237
Sentence    12.820396  4.092085 13.411339 13.042237 25.546804
WordMean    13.217846  6.934741 24.280061 22.019597 29.245814
            WordMean
VisualPerc 13.217846
Cubes       6.934741
Lozenges   24.280061
Paragrap   22.019597
Sentence   29.245814
WordMean   69.208143
```

```
# vcov(HSF.sem.fm1)



summary(
    object       = HSF.sem.fm1
  , header       = TRUE
  , fit.measures = FALSE
```

```
  , estimates    = TRUE
  , ci           = FALSE
  , fmi          = FALSE
  , standardized = TRUE
  , rsquare      = TRUE
  , std.nox      = FALSE
  , modindices   = FALSE
  , nd           = 3L
)
```

```
lavaan 0.6-3 ended normally after 64 iterations

  Optimization method                           NLMINB
  Number of free parameters                         13

  Number of observations                            73

  Estimator                                         ML
  Model Fit Test Statistic                       7.962
  Degrees of freedom                                 8
  P-value (Chi-square)                           0.437

Parameter Estimates:

  Information                                 Expected
  Information saturated (h1) model          Structured
  Standard Errors                             Standard

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv
  Spatial =~
    VislPrc (lmb1)    1.000                               4.827
    Cubes   (lmb2)    0.610    0.142    4.279    0.000    2.943
    Lozengs (lmb3)    1.198    0.270    4.436    0.000    5.784
  Verbal =~
    Paragrp (lmb4)    1.000                               3.112
    Sentenc (lmb5)    1.334    0.159    8.379    0.000    4.151
    WordMen (lmb6)    2.234    0.262    8.541    0.000    6.952
  Std.all

    0.703
    0.654
    0.736

    0.880
    0.827
```

```
    0.841

Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv
  Spatial ~~
    Verbal   (r12)    7.315    2.553    2.865    0.004    0.487
  Std.all

    0.487

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv
   .VisualPerc       23.873    5.945    4.016    0.000   23.873
   .Cubes            11.602    2.566    4.521    0.000   11.602
   .Lozenges         28.275    7.837    3.608    0.000   28.275
   .Paragrap          2.834    0.863    3.286    0.001    2.834
   .Sentence          7.967    1.856    4.292    0.000    7.967
   .WordMean         19.925    4.917    4.052    0.000   19.925
    Spatial          23.302    8.068    2.888    0.004    1.000
    Verbal            9.682    2.144    4.516    0.000    1.000
  Std.all
    0.506
    0.572
    0.458
    0.226
    0.316
    0.292
    1.000
    1.000

R-Square:
                   Estimate
    VisualPerc        0.494
    Cubes             0.428
    Lozenges          0.542
    Paragrap          0.774
    Sentence          0.684
    WordMean          0.708
```

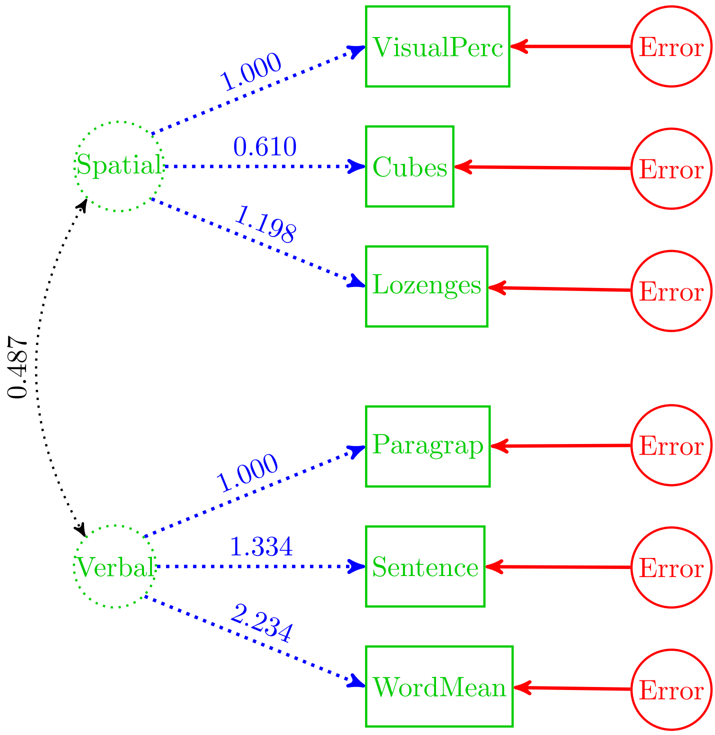

Figure 16: Path Diagram with estimates of coefficients

6 STRUCURAL EQUATION MODELING WITH LATENT VARIABLES

6.1 INTRODUCTION

General Structural Equation Model

The *general structural equation model* consists of a *measurement model* that specifies the relation of observed to latent variables and a *latent variable model* that shows the influence of latent variables on each other.

The first component of the structural equations is the latent variable model:

Latent Variable Model

$$\boldsymbol{\eta}_i = \mathbf{B}\boldsymbol{\eta}_i + \boldsymbol{\Gamma}\boldsymbol{\xi}_i + \boldsymbol{\zeta}_i \tag{14}$$

In (14), $\boldsymbol{\eta}_i$ (eta), the vector of latent endogenous random variables, is $m \times 1$; $\boldsymbol{\xi}_i$ (xi) the latent exogenous random variables, is $n \times 1$; $\mathbf{B}$ is the $m \times m$ coefficient matrix showing the influence of the latent endogenous variables on each other; $\boldsymbol{\Gamma}$ (Gamma) is the $m \times n$ coefficient matrix for the effects of $\boldsymbol{\xi}_i$ on $\boldsymbol{\eta}_i$. The matrix $(\mathbf{I} - \mathbf{B})$ is non-singular. $\boldsymbol{\zeta}_i$ (zeta) is the disturbance vector that is assumed to have an expected value of zero $[\mathbb{E}(\boldsymbol{\zeta}_i) = \mathbf{0}]$ and which is uncorrelated with $\boldsymbol{\xi}_i$.

The second component of the general system is the measurement model:

Measurement Model

$$\mathbf{y}_i = \boldsymbol{\Lambda}_y \boldsymbol{\eta}_i + \boldsymbol{\epsilon}_i \tag{15}$$
$$\mathbf{x}_i = \boldsymbol{\Lambda}_x \boldsymbol{\xi}_i + \boldsymbol{\delta}_i \tag{16}$$

The $\mathbf{y}_i$ $(p \times 1)$ and the $\mathbf{x}_i$ $(q \times 1)$ vectors are observed variables, $\boldsymbol{\Lambda}_y$ $(p \times m)$ (Lambda) and $\boldsymbol{\Lambda}_x$ $(q \times n)$ are the coefficient matrices that show the relation of $\mathbf{y}_i$ to $\boldsymbol{\eta}_i$ and $\mathbf{x}_i$ to $\boldsymbol{\xi}_i$ respectively, and $\boldsymbol{\epsilon}_i$ $(p \times 1)$ (epsilon) and $\boldsymbol{\delta}_i$ $(q \times 1)$ (delta) are the errors of measurement for $\mathbf{y}_i$ and $\mathbf{x}_i$, respectively. The errors of measurement are assumed to be uncorrelated with $\boldsymbol{\eta}_i$, $\boldsymbol{\xi}_i$ and $\boldsymbol{\zeta}_i$ and with each other.

Also $\mathbb{V}(\boldsymbol{\xi}_i) = \boldsymbol{\Phi}$ (Phi), $\mathbb{V}(\boldsymbol{\zeta}_i) = \boldsymbol{\Psi}$ (Psi), $\mathbb{V}(\boldsymbol{\epsilon}_i) = \boldsymbol{\Theta}_\epsilon$ (Theta), $\mathbb{V}(\boldsymbol{\delta}_i) = \boldsymbol{\Theta}_\delta$ (Theta) and $\mathbb{V}\left(\begin{bmatrix}\mathbf{y}_i \\ \mathbf{x}_i\end{bmatrix}\right) = \boldsymbol{\Sigma}$ (Sigma).

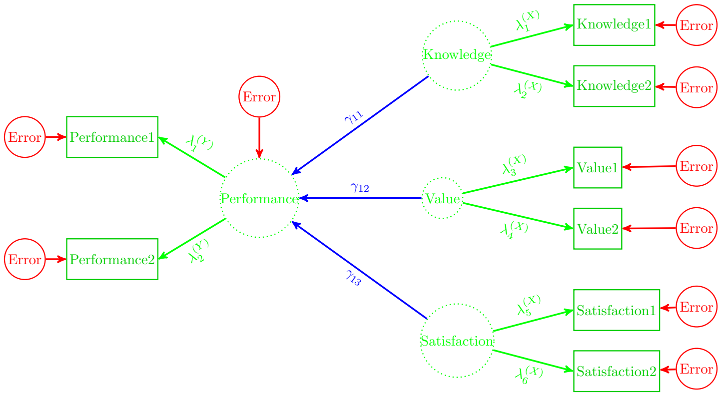

Figure 17: Path Diagram for job performance of farm managers

```r
library(lavaan)

Warren8V <-
  matrix(
    data =
c(
0.0271, 0.0172,  0.0219, 0.0164,  0.0284,  0.0217,  0.0083,  0.0074,
0.0172, 0.0222,  0.0193, 0.0130,  0.0294,  0.0185,  0.0011,  0.0015,
0.0219, 0.0193,  0.0876, 0.0317,  0.0383,  0.0356, -0.0001,  0.0035,
0.0164, 0.0130,  0.0317, 0.0568,  0.0151,  0.0230,  0.0055,  0.0089,
0.0284, 0.0294,  0.0383, 0.0151,  0.1826,  0.0774, -0.0087, -0.0007,
0.0217, 0.0185,  0.0356, 0.0230,  0.0774,  0.1473, -0.0069, -0.0088,
0.0083, 0.0011, -0.0001, 0.0055, -0.0087, -0.0069,  0.1137,  0.0722,
0.0074, 0.0015,  0.0035, 0.0089, -0.0007, -0.0088,  0.0722,  0.1024
)
    , nrow   = 8
    , ncol   = 8
    , byrow = TRUE
    , dimnames = list(c("Performance1", "Performance2",
                        "Knowledge1",    "Knowledge2",
                        "Value1",    "Value2",
                        "Satisfaction1", "Satisfaction2")
                      , c("Performance1",    "Performance2",
                          "Knowledge1", "Knowledge2",
                          "Value1", "Value2",
                          "Satisfaction1", "Satisfaction2")
    )

  )

Warren8V
```

```
              Performance1 Performance2 Knowledge1 Knowledge2
Performance1        0.0271       0.0172     0.0219     0.0164
```

```
Performance2        0.0172       0.0222      0.0193      0.0130
Knowledge1          0.0219       0.0193      0.0876      0.0317
Knowledge2          0.0164       0.0130      0.0317      0.0568
Value1              0.0284       0.0294      0.0383      0.0151
Value2              0.0217       0.0185      0.0356      0.0230
Satisfaction1       0.0083       0.0011     -0.0001      0.0055
Satisfaction2       0.0074       0.0015      0.0035      0.0089
               Value1  Value2 Satisfaction1 Satisfaction2
Performance1   0.0284  0.0217        0.0083        0.0074
Performance2   0.0294  0.0185        0.0011        0.0015
Knowledge1     0.0383  0.0356       -0.0001        0.0035
Knowledge2     0.0151  0.0230        0.0055        0.0089
Value1         0.1826  0.0774       -0.0087       -0.0007
Value2         0.0774  0.1473       -0.0069       -0.0088
Satisfaction1 -0.0087 -0.0069        0.1137        0.0722
Satisfaction2 -0.0007 -0.0088        0.0722        0.1024
```

```
Warren8V.sem.m1 <- '
  # Latent Variables
    Performance  =~ lambda1Y * Performance1 + lambda2Y * Performance2
    Knowledge    =~ lambda1X * Knowledge1 + lambda2X * Knowledge2
    Value        =~ lambda3X * Value1 + lambda4X * Value2
    Satisfaction =~ lambda5X * Satisfaction1 + lambda6X * Satisfaction2

  # Regressions
    Performance ~ gamma11 * Knowledge + gamma12 *Value +
                  gamma13 *Satisfaction

  # Variances and Covariances
    Knowledge   ~~ r12 * Value
    Knowledge   ~~ r13 * Satisfaction
    Value       ~~ r23 * Satisfaction
    Performance ~~ sigma2P * Performance
'

Warren8V.sem.fm1 <-
  lavaan::sem(
      model            = Warren8V.sem.m1
  # , data
    , ordered          = NULL
    , sampling.weights = NULL
    , sample.cov       = Warren8V
    , sample.mean      = NULL
    , sample.nobs      = 98
    , group            = NULL
    , cluster          = NULL
```

```
  # , constraints       = ""
    , WLS.V             = NULL
    , NACOV             = NULL
  # ,    ...
  )

# methods(class = class(Warren8V.sem.fm1))
# anova(Warren8V.sem.fm1)
# coef(Warren8V.sem.fm1)
parameterEstimates(Warren8V.sem.fm1, standardized = TRUE)
```

```
             lhs op            rhs    label    est    se      z
1    Performance =~   Performance1 lambda1Y  1.000 0.000     NA
2    Performance =~   Performance2 lambda2Y  0.867 0.116  7.489
3      Knowledge =~     Knowledge1 lambda1X  1.000 0.000     NA
4      Knowledge =~     Knowledge2 lambda2X  0.683 0.160  4.274
5          Value =~         Value1 lambda3X  1.000 0.000     NA
6          Value =~         Value2 lambda4X  0.763 0.184  4.149
7   Satisfaction =~  Satisfaction1 lambda5X  1.000 0.000     NA
8   Satisfaction =~  Satisfaction2 lambda6X  0.792 0.436  1.816
9    Performance  ~      Knowledge  gamma11  0.337 0.124  2.711
10   Performance  ~          Value  gamma12  0.176 0.079  2.237
11   Performance  ~   Satisfaction  gamma13  0.061 0.054  1.132
12     Knowledge ~~          Value      r12  0.037 0.012  3.052
13     Knowledge ~~   Satisfaction      r13  0.004 0.009  0.464
14         Value ~~   Satisfaction      r23 -0.008 0.013 -0.614
15   Performance ~~    Performance  sigma2P  0.007 0.003  2.590
16  Performance1 ~~   Performance1           0.007 0.002  3.126
17  Performance2 ~~   Performance2           0.007 0.002  3.891
18    Knowledge1 ~~     Knowledge1           0.041 0.011  3.629
19    Knowledge2 ~~     Knowledge2           0.035 0.007  5.193
20        Value1 ~~         Value1           0.080 0.025  3.266
21        Value2 ~~         Value2           0.087 0.018  4.916
22 Satisfaction1 ~~  Satisfaction1           0.022 0.049  0.453
23 Satisfaction2 ~~  Satisfaction2           0.045 0.031  1.428
24     Knowledge ~~      Knowledge           0.046 0.015  3.154
25         Value ~~          Value           0.100 0.032  3.164
26  Satisfaction ~~   Satisfaction           0.090 0.051  1.754
   pvalue ci.lower ci.upper std.lv std.all std.nox
1      NA    1.000    1.000  0.140   0.856   0.856
2   0.000    0.640    1.093  0.121   0.819   0.819
3      NA    1.000    1.000  0.214   0.728   0.728
4   0.000    0.370    0.997  0.146   0.618   0.618
5      NA    1.000    1.000  0.317   0.745   0.745
6   0.000    0.402    1.123  0.242   0.633   0.633
7      NA    1.000    1.000  0.300   0.896   0.896
```

```
8   0.069   -0.063    1.646  0.238   0.747   0.747
9   0.007    0.093    0.581  0.516   0.516   0.516
10  0.025    0.022    0.330  0.398   0.398   0.398
11  0.257   -0.044    0.166  0.130   0.130   0.130
12  0.002    0.013    0.060  0.542   0.542   0.542
13  0.643   -0.013    0.022  0.064   0.064   0.064
14  0.539   -0.033    0.018 -0.084  -0.084  -0.084
15  0.010    0.002    0.012  0.337   0.337   0.337
16  0.002    0.003    0.012  0.007   0.268   0.268
17  0.000    0.004    0.011  0.007   0.329   0.329
18  0.000    0.019    0.063  0.041   0.471   0.471
19  0.000    0.022    0.048  0.035   0.619   0.619
20  0.001    0.032    0.128  0.080   0.444   0.444
21  0.000    0.053    0.122  0.087   0.599   0.599
22  0.650   -0.074    0.119  0.022   0.198   0.198
23  0.153   -0.017    0.106  0.045   0.442   0.442
24  0.002    0.017    0.074  1.000   1.000   1.000
25  0.002    0.038    0.163  1.000   1.000   1.000
26  0.080   -0.011    0.191  1.000   1.000   1.000
```

```r
# fitmeasures(Warren8V.sem.fm1)
fitted(Warren8V.sem.fm1)$cov
```

```
              Prfrm1 Prfrm2 Knwld1 Knwld2 Value1 Value2 Stsfc1
Performance1   0.027
Performance2   0.017  0.022
Knowledge1     0.022  0.019  0.087
Knowledge2     0.015  0.013  0.031  0.056
Value1         0.030  0.026  0.037  0.025  0.181
Value2         0.023  0.020  0.028  0.019  0.077  0.146
Satisfaction1  0.005  0.005  0.004  0.003 -0.008 -0.006  0.113
Satisfaction2  0.004  0.004  0.003  0.002 -0.006 -0.005  0.071
              Stsfc2
Performance1
Performance2
Knowledge1
Knowledge2
Value1
Value2
Satisfaction1
Satisfaction2  0.101
```

```r
# residuals(Warren8V.sem.fm1, type = "cor")
# modificationIndices(Warren8V.sem.fm1)
Warren8V
```

```
              Performance1 Performance2 Knowledge1 Knowledge2
```

```
Performance1        0.0271       0.0172      0.0219      0.0164
Performance2        0.0172       0.0222      0.0193      0.0130
Knowledge1          0.0219       0.0193      0.0876      0.0317
Knowledge2          0.0164       0.0130      0.0317      0.0568
Value1              0.0284       0.0294      0.0383      0.0151
Value2              0.0217       0.0185      0.0356      0.0230
Satisfaction1       0.0083       0.0011     -0.0001      0.0055
Satisfaction2       0.0074       0.0015      0.0035      0.0089
                Value1  Value2 Satisfaction1 Satisfaction2
Performance1    0.0284  0.0217        0.0083        0.0074
Performance2    0.0294  0.0185        0.0011        0.0015
Knowledge1      0.0383  0.0356       -0.0001        0.0035
Knowledge2      0.0151  0.0230        0.0055        0.0089
Value1          0.1826  0.0774       -0.0087       -0.0007
Value2          0.0774  0.1473       -0.0069       -0.0088
Satisfaction1 -0.0087 -0.0069        0.1137        0.0722
Satisfaction2 -0.0007 -0.0088        0.0722        0.1024
```

```
# vcov(Warren8V.sem.fm1)



summary(
    object       = Warren8V.sem.fm1
  , header       = TRUE
  , fit.measures = FALSE
  , estimates    = TRUE
  , ci           = FALSE
  , fmi          = FALSE
  , standardized = TRUE
  , rsquare      = TRUE
  , std.nox      = FALSE
  , modindices   = FALSE
  , nd           = 3L
)
```

```
lavaan 0.6-3 ended normally after 75 iterations

  Optimization method                           NLMINB
  Number of free parameters                         22

  Number of observations                            98

  Estimator                                         ML
  Model Fit Test Statistic                      10.441
  Degrees of freedom                                14
  P-value (Chi-square)                           0.729
```

```
Parameter Estimates:

  Information                                 Expected
  Information saturated (h1) model          Structured
  Standard Errors                             Standard

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv
  Performance =~
    Prfrmn1 (lm1Y)    1.000                               0.140
    Prfrmn2 (lm2Y)    0.867    0.116    7.489    0.000    0.121
  Knowledge =~
    Knwldg1 (lm1X)    1.000                               0.214
    Knwldg2 (lm2X)    0.683    0.160    4.274    0.000    0.146
  Value =~
    Value1  (lm3X)    1.000                               0.317
    Value2  (lm4X)    0.763    0.184    4.149    0.000    0.242
  Satisfaction =~
    Stsfct1 (lm5X)    1.000                               0.300
    Stsfct2 (lm6X)    0.792    0.436    1.816    0.069    0.238
  Std.all

    0.856
    0.819

    0.728
    0.618

    0.745
    0.633

    0.896
    0.747

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv
  Performance ~
    Knowldg (gm11)    0.337    0.124    2.711    0.007    0.516
    Value   (gm12)    0.176    0.079    2.237    0.025    0.398
    Stsfctn (gm13)    0.061    0.054    1.132    0.257    0.130
  Std.all

    0.516
    0.398
    0.130
```

```
Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv
  Knowledge ~~
    Value    (r12)    0.037    0.012    3.052    0.002    0.542
    Satsfctn (r13)    0.004    0.009    0.464    0.643    0.064
  Value ~~
    Satsfctn (r23)   -0.008    0.013   -0.614    0.539   -0.084
  Std.all

    0.542
    0.064

   -0.084

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv
   .Prfrmnc (sg2P)    0.007    0.003    2.590    0.010    0.337
   .Prfrmn1           0.007    0.002    3.126    0.002    0.007
   .Prfrmn2           0.007    0.002    3.891    0.000    0.007
   .Knwldg1           0.041    0.011    3.629    0.000    0.041
   .Knwldg2           0.035    0.007    5.193    0.000    0.035
   .Value1            0.080    0.025    3.266    0.001    0.080
   .Value2            0.087    0.018    4.916    0.000    0.087
   .Stsfct1           0.022    0.049    0.453    0.650    0.022
   .Stsfct2           0.045    0.031    1.428    0.153    0.045
    Knowldg           0.046    0.015    3.154    0.002    1.000
    Value             0.100    0.032    3.164    0.002    1.000
    Stsfctn           0.090    0.051    1.754    0.080    1.000
  Std.all
    0.337
    0.268
    0.329
    0.471
    0.619
    0.444
    0.599
    0.198
    0.442
    1.000
    1.000
    1.000

R-Square:
                   Estimate
    Performance       0.663
```

Performance1	0.732
Performance2	0.671
Knowledge1	0.529
Knowledge2	0.381
Value1	0.556
Value2	0.401
Satisfaction1	0.802
Satisfaction2	0.558

BIBLIOGRAPHY

[1] Bollen, K.A. *Structural Equations with Latent Variables*. John Wiley & Sons, New Jersey, USA, 1989.

[2] Sacha Epskamp and with contributions from Simon Stuber. *semPlot: Path Diagrams and Visual Analysis of Various SEM Packages' Output*, 2017. URL `https://CRAN.R-project.org/package=semPlot`. R package version 1.1.

[3] John Fox, Zhenghua Nie, and Jarrett Byrnes. *sem: Structural Equation Models*, 2017. URL `https://CRAN.R-project.org/package=sem`. R package version 3.1-9.

[4] Michael C. Neale, Michael D. Hunter, Joshua N. Pritikin, Mahsa Zahery, Timothy R. Brick, Robert M. Kirkpatrick, Ryne Estabrook, Timothy C. Bates, Hermine H. Maes, and Steven M. Boker. OpenMx 2.0: Extended structural equation and statistical modeling. *Psychometrika*, 81(2):535–549, 2016. doi: 10.1007/s11336-014-9435-8.

[5] R Core Team. *R: A Language & Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2017. URL `http://www.R-project.org/`.

[6] Yves Rosseel. lavaan: An R package for structural equation modeling. *Journal of Statistical Software*, 48(2):1–36, 2012. URL `http://www.jstatsoft.org/v48/i02/`.