Using R for Structural Equation Model: A transaction cost measurement

Pairach Piboonrugnroj and Stephen M. Disney

Logistics Systems Dynamics Group,
Cardiff University

August 16^{th}, 2011

Motivation

Supply Chain Management Matters

- Businesses are trying to reduce their *transaction costs* to improve their business performance and relationships.

$$Supplier \longrightarrow Manufacturer \longrightarrow Retailer \longrightarrow Customers$$

- However,a measurement of transaction cost is still limited.
  - Only in manufacturing context NOT in service sector.
  - Aspects of governance problem and opportunity costs are missing.

Motivation

Software Matters

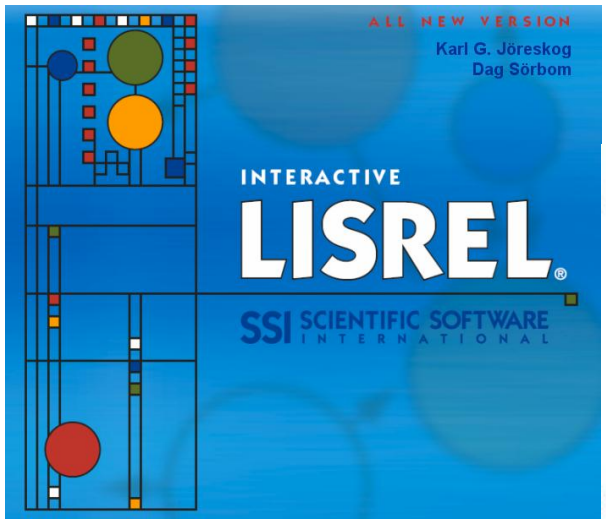
- SEM can be fitted by various software but costly
  - Lisrel8.8 = USD 396
  - IBM SPSS Amos = USD 695
  - Mplus6.11 = USD 195-350 (student price)
  - StataSE12 = USD 895
- Can I use R to run SEM?
  - Identical output to those of other commercial software?
  - Any difficulties or problems from the non-technical user aspect?

Aims

1. To develop the measurement of *transaction costs*.
2. To empirically test such a measurement.
3. To compare the use of R packages for SEM with other software *via a non-technical aspect*, an outside**R**!

Analysis Method

- Structural Equation Model (Bollen, 1989)
  = Factor Analysis + Regression (Path Analysis)
- Proprietary software i.e.,



Analysis Method

- Packages in R (R Development Core Team, 2011)

  - sem (John Fox, 2006)
  - OpenMx 1.0.7
    S. Boker, M. Neale, H. Maes, M. Wilde, M. Spiegel, T. Brick, J. Spies, R. Estabrook, S. Kenny, T. Bates, P. Mehta, and J. Fox, 2011)
  - lavaan 0.4-9 (Yves Rosseel, 2011)

Data

- Questionnaire survey
- Tourism industry of Thailand
- 53 usable responses

LISREL

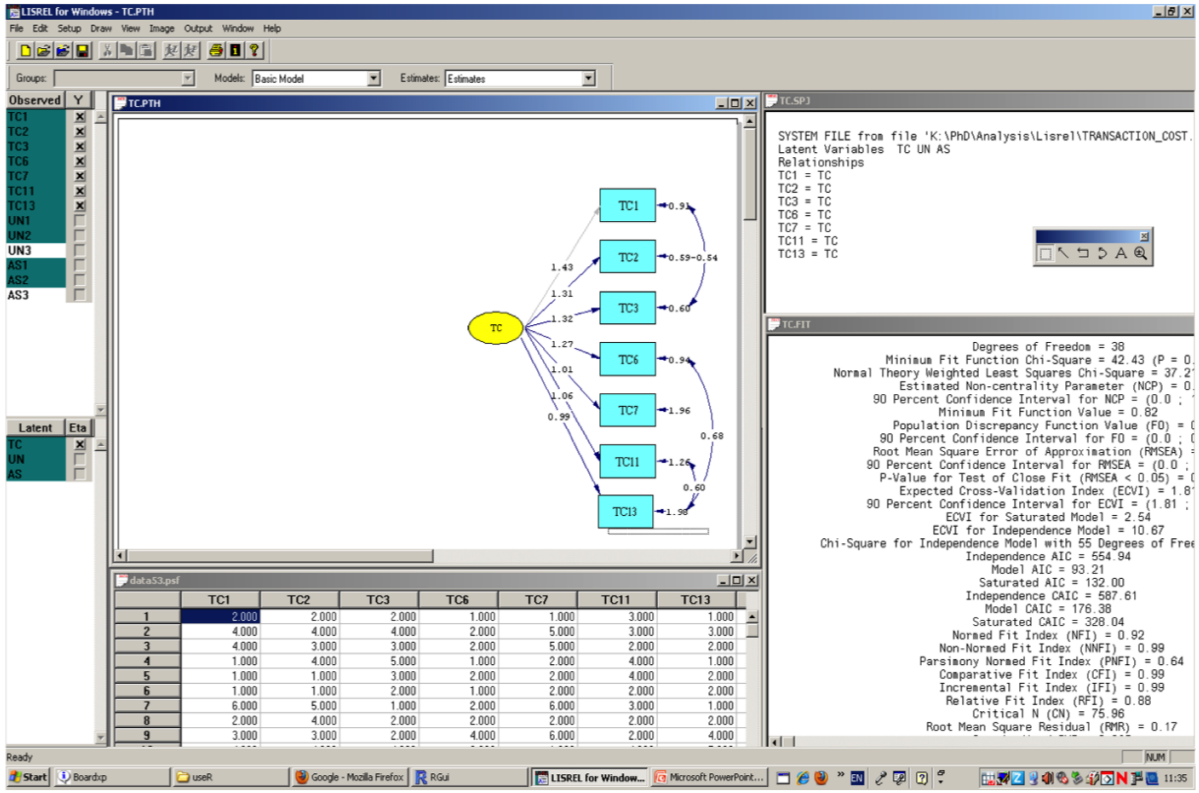
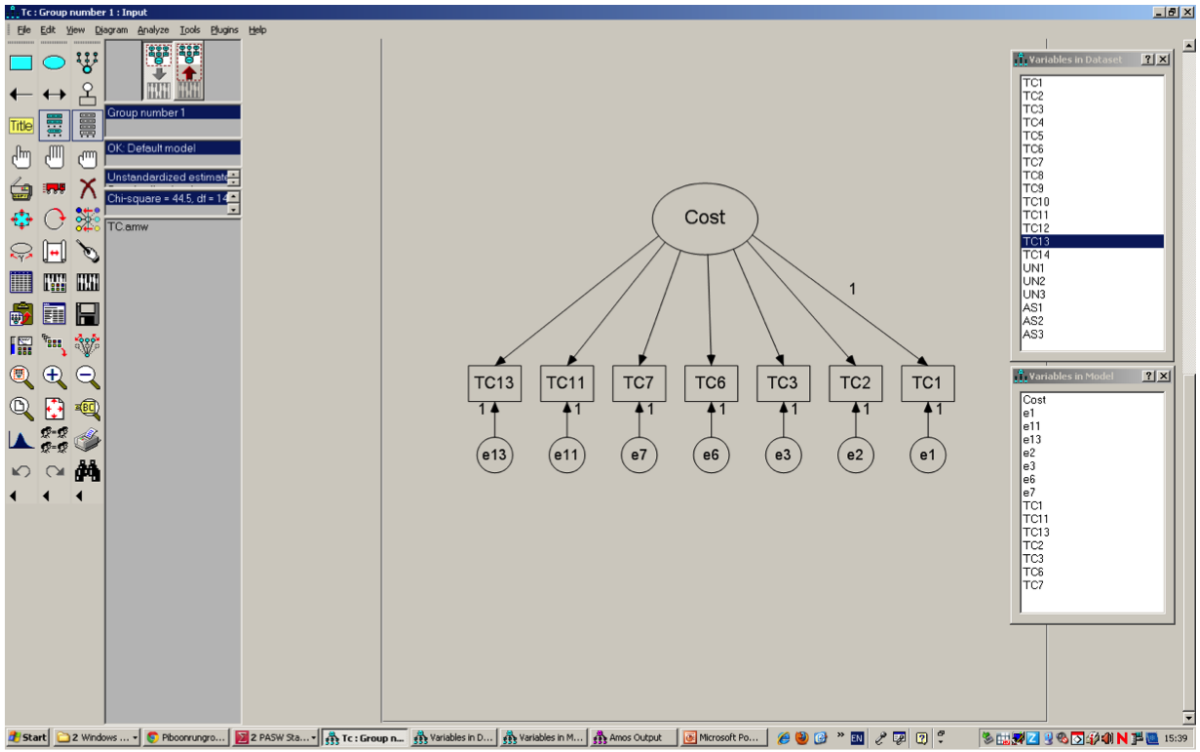


Figure: Using Lisrel to fit the model

AMOS: Model Specification



AMOS: Model Output 1

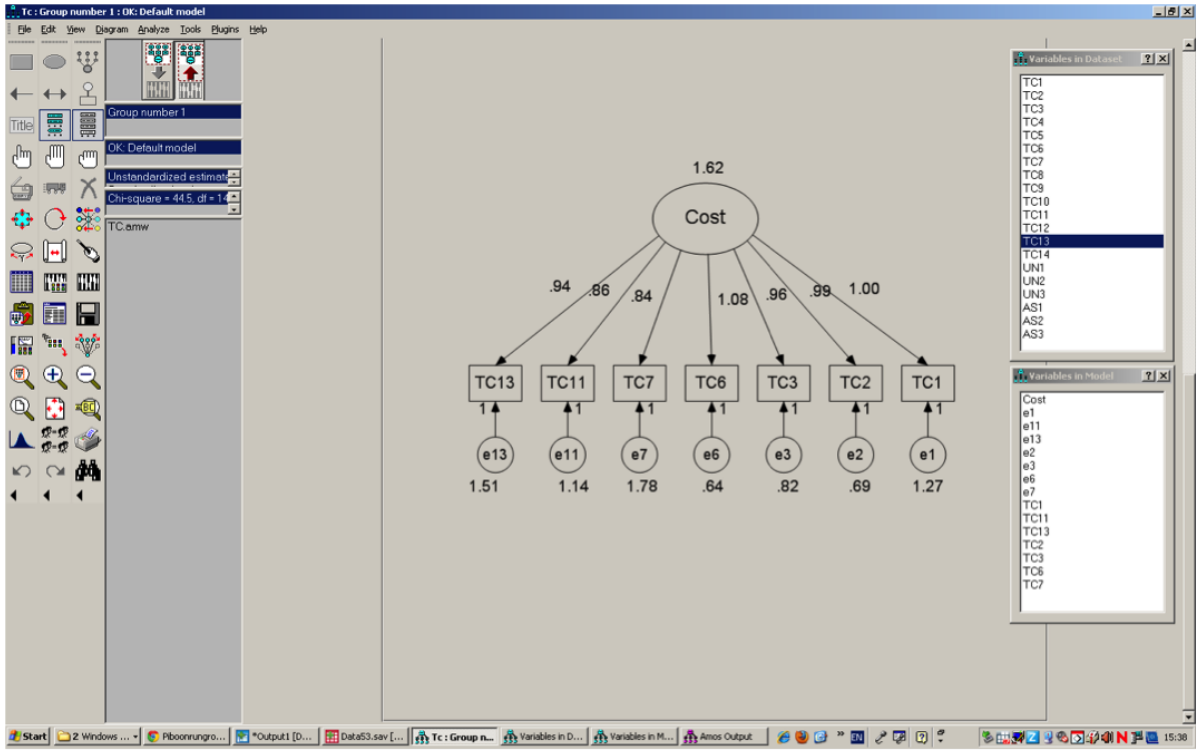
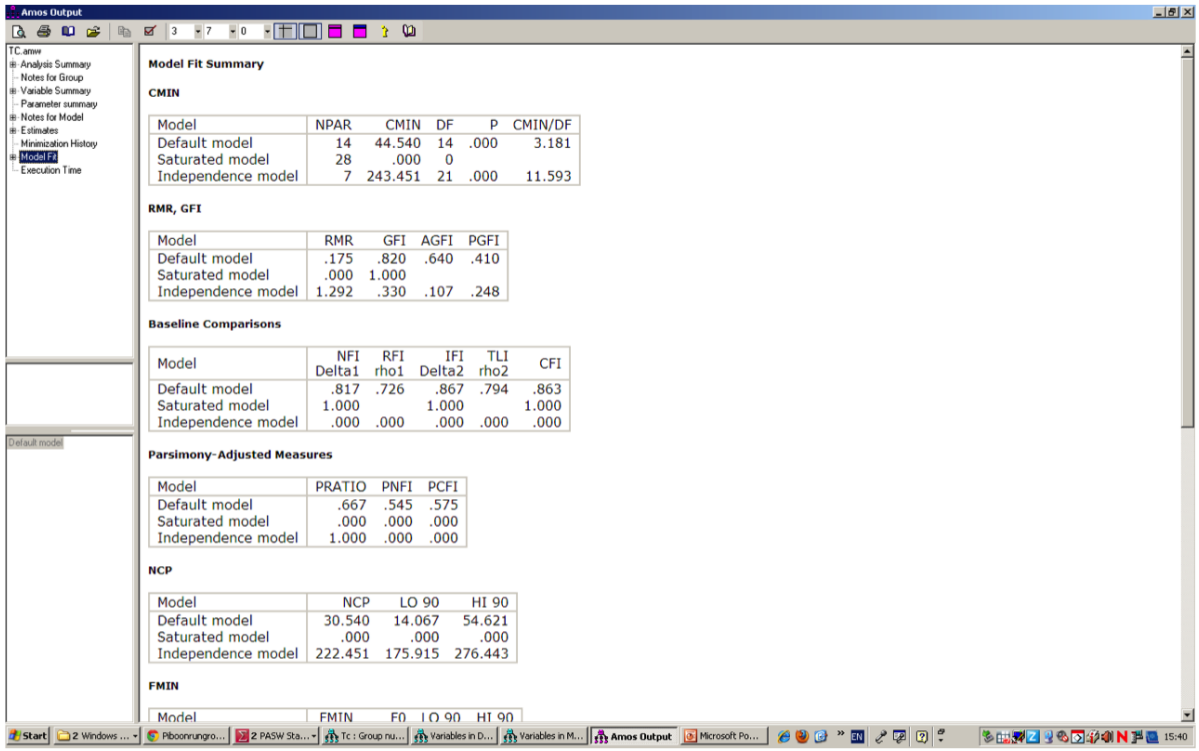


Figure: Using Amos to fit the model

AMOS: Model Output 2



Amos Output

TC.amw
- Analysis Summary
- Notes for Group
- Variable Summary
- Parameter summary
- Notes for Model
- Estimates
- Minimization History
- Model Fit
- Execution Time

Default model

Model Fit Summary

CMIN

Model	NPAR	CMIN	DF	P	CMIN/DF
Default model	14	44.540	14	.000	3.181
Saturated model	28	.000	0		
Independence model	7	243.451	21	.000	11.593

RMR, GFI

Model	RMR	GFI	AGFI	PGFI
Default model	.175	.820	.640	.410
Saturated model	.000	1.000		
Independence model	1.292	.330	.107	.248

Baseline Comparisons

Model	NFI Delta1	RFI rho1	IFI Delta2	TLI rho2	CFI
Default model	.817	.726	.867	.794	.863
Saturated model	1.000		1.000		1.000
Independence model	.000	.000	.000	.000	.000

Parsimony-Adjusted Measures

Model	PRATIO	PNFI	PCFI
Default model	.667	.545	.575
Saturated model	.000	.000	.000
Independence model	1.000	.000	.000

NCP

Model	NCP	LO 90	HI 90
Default model	30.540	14.067	54.621
Saturated model	.000	.000	.000
Independence model	222.451	175.915	276.443

FMIN

Model	FMIN	F0	LO 90	HI 90

Figure: Using Amos to fit the model

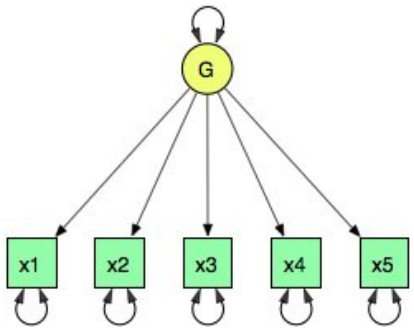
'sem' Package: Code

```
##-------------------------------------------------##
##  Introduction to Structural Equation Modelling  ##
##               Pairach Piboonrungroj             ##
##         R useR conference August 2011           ##
##-------------------------------------------------##
install.packages("sem")
library(sem)
# 1. Load data
hoteldata <- read.csv("/Users/pairachpiboonrungroj/Documents/PhD/Analysis/R/
 useR/cleandata.csv")
#Input covariance matrix
rm(data.tc.1)
data.tc.1 <- cor(hoteldata)

#hotel.cov <- cov(data.sem)
hotel.cor <- cor(hoteldata)
 #  path parameter  start-value
model.TC.1 <- specify.model()
    TC   -> TC1,    gamma1,  NA # measurement item
    TC   -> TC2,    gamma2,  NA
    TC   -> TC3,    gamma3,  NA
    TC   -> TC6,    gamma6,  NA
    TC   -> TC7,    gamma7,  NA
    TC   -> TC11,   gamma11, NA
    TC   -> TC13,   gamma13, NA
    TC1  <-> TC1,   e1,      NA # measurement error
    TC2  <-> TC2,   e2,      NA
    TC3  <-> TC3,   e3,      NA
    TC6  <-> TC6,   e6,      NA
    TC7  <-> TC7,   e7,      NA
    TC11 <-> TC11,  e11,     NA
    TC13 <-> TC13,  e13,     NA
    TC   <-> TC,    NA,      1

model.TC.1

sem.TC.1 <- sem(model.TC.1, data.tc.1, 53)
# print result (fit indices, parameters, hypothesis tests)
summary(sem.TC.1)
# standardised coefficients (loadings)
std.coef(sem.TC.1)
```

'sem' Package: Output

```
> # print result (fit indices, parameters, hypothesis tests)
> summary(sem.TC.1)

 Model Chisquare =  44.54   Df =  14 Pr(>Chisq) = 4.8429e-05
 Chisquare (null model) =  243.45   Df =  21
 Goodness-of-fit index =  0.81983
 Adjusted goodness-of-fit index =  0.63967
 RMSEA index =  0.20482   90% CI: (NA, NA)
 Bentler-Bonnett NFI =  0.81705
 Tucker-Lewis NNFI =  0.79407
 Bentler CFI =  0.86271
 SRMR =  0.067909
 BIC =  -11.044

 Normalized Residuals
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-0.8220 -0.2840  0.0000  0.0125  0.3350  0.8620

 Parameter Estimates
        Estimate Std Error z value Pr(>|z|)
gamma1  0.74793  0.122606  6.1003  1.0587e-09 TC1 <--- TC
gamma2  0.83540  0.116348  7.1802  6.9611e-13 TC2 <--- TC
gamma3  0.80301  0.118532  6.7746  1.2472e-11 TC3 <--- TC
gamma6  0.86297  0.113765  7.5856  3.3085e-14 TC6 <--- TC
gamma7  0.62515  0.129333  4.8337  1.3402e-06 TC7 <--- TC
gamma11 0.71731  0.125024  5.7374  9.6159e-09 TC11 <--- TC
gamma13 0.69967  0.126639  5.5249  3.2965e-08 TC13 <--- TC
e1      0.44060  0.100493  4.3844  1.1633e-05 TC1 <--> TC1
e2      0.30211  0.079559  3.7973  1.4630e-04 TC2 <--> TC2
e3      0.35517  0.086406  4.1105  3.9479e-05 TC3 <--> TC3
e6      0.25528  0.071294  3.5806  3.4281e-04 TC6 <--> TC6
e7      0.60918  0.127416  4.7811  1.7438e-06 TC7 <--> TC7
e11     0.48547  0.108776  4.4630  8.0804e-06 TC11 <--> TC11
e13     0.51047  0.113940  4.4801  7.4592e-06 TC13 <--> TC13

 Iterations =  14
> # standardised coefficients (loadings)
> std.coef(sem.TC.1)
               Std. Estimate
gamma1   gamma1     0.7479309    TC1 <--- TC
gamma2   gamma2     0.8354004    TC2 <--- TC
```

'OpenMx' package




```
require(OpenMx)
data(demoOneFactor)
manifests <- names(demoOneFactor)
latents <- c("G")
factorModel <- mxModel("One Factor",
      type="RAM",
      manifestVars = manifests,
      latentVars = latents,
      mxPath(from=latents, to=manifests),
      mxPath(from=manifests, arrows=2),
      mxPath(from=latents, arrows=2,
            free=FALSE, values=1.0),
      mxData(cov(demoOneFactor), type="cov",
            numObs=500))
summary(mxRun(factorModel))
```

Figure: OpenMx using Path Specification

'OpenMx' package

$$\mathbf{R} = \mathbf{ALA}' + \mathbf{U}$$

```
require(OpenMx)
data(demoOneFactor)
factorModel <- mxModel("One Factor",
      mxMatrix("Full", 5, 1, values=0.2,
           free=TRUE, name="A"),
      mxMatrix("Symm", 1, 1, values=1,
           free=FALSE, name="L"),
      mxMatrix("Diag", 5, 5, values=1,
           free=TRUE, name="U"),
      mxAlgebra(A %*% L %*% t(A) + U, name="R"),
      mxMLObjective("R", dimnames = names(demoOneFactor)),
      mxData(cov(demoOneFactor), type="cov", numObs=500))
summary(mxRun(factorModel))
```

Figure: OpenMx using Matrix Specification

'OpenMx' package: Code

```
1
2  require(OpenMx)
3  hoteldata <- read.csv("/Users/pairachpiboonrungroj/Documents/PhD/Analysis/R/
     useR/cleandataCFA.csv")
4  manifests <- names(hoteldata)
5  latents <- c("TC")
6  CFA.TC <- mxModel("One Factor",
7        type="RAM",
8        manifestVars = manifests,
9        latentVars = latents,
10       mxPath(from=latents, to=manifests),
11       mxPath(from=manifests, arrows=2),
12       mxPath(from=latents, arrows=2,
13             free=FALSE, values=1.0),
14       mxData(cov(hoteldata), type="cov",
15             numObs=53))
16 FactorFit.TC <- mxRun(CFA.TC)
17 summary(FactorFit.TC)
18
```

Figure: OpenMx using Path Specification

'OpenMx' package: Output1

```
>
> require(OpenMx)
Loading required package: OpenMx
> hoteldata <- read.csv("/Users/pairachpiboonrungroj/Documents/PhD/Analysis/
R/useR/cleandataCFA.csv")
> manifests <- names(hoteldata)
> latents <- c("TC")
> CFA.TC <- mxModel("One Factor",
+       type="RAM",
+       manifestVars = manifests,
+       latentVars = latents,
+       mxPath(from=latents, to=manifests),
+       mxPath(from=manifests, arrows=2),
+       mxPath(from=latents, arrows=2,
+             free=FALSE, values=1.0),
+       mxData(cov(hoteldata), type="cov",
+             numObs=53))
> FactorFit.TC <- mxRun(CFA.TC)
Running One Factor
> summary(FactorFit.TC)
data:
$`One Factor.data`
$`One Factor.data`$cov
          TC1      TC2      TC3      TC6      TC7     TC11     TC13
TC1  2.945573 1.937228 1.345428 1.851597 1.448113 1.219158 1.284107
TC2  1.937228 2.311321 1.695573 1.754354 1.358491 1.186865 1.192671
TC3  1.345428 1.695573 2.354862 1.626996 1.276488 1.658563 1.379173
TC6  1.851597 1.754354 1.626996 2.566763 1.349419 1.416546 1.973512
TC7  1.448113 1.358491 1.276488 1.349419 2.980406 1.404572 1.485849
TC11 1.219158 1.186865 1.658563 1.416546 1.404572 2.383164 1.701379
TC13 1.284107 1.192671 1.379173 1.973512 1.485849 1.701379 3.005806
```

'OpenMx' package: Output2

```
free parameters:
   name matrix  row  col  Estimate Std.Error
1  <NA>      A  TC1   TC 1.2836489 0.2103818
2  <NA>      A  TC2   TC 1.2700609 0.1768347
3  <NA>      A  TC3   TC 1.2322648 0.1818500
4  <NA>      A  TC6   TC 1.3825805 0.1822115
5  <NA>      A  TC7   TC 1.0792584 0.2232472
6  <NA>      A TC11   TC 1.1073405 0.1929696
7  <NA>      A TC13   TC 1.2130293 0.2195215
8  <NA>      S  TC1  TC1 1.2978175 0.2958840
9  <NA>      S  TC2  TC2 0.6982638 0.1837838
10 <NA>      S  TC3  TC3 0.8363844 0.2033757
11 <NA>      S  TC6  TC6 0.6552330 0.1828873
12 <NA>      S  TC7  TC7 1.8156069 0.3796282
13 <NA>      S TC11 TC11 1.1569600 0.2591265
14 <NA>      S TC13 TC13 1.5343629 0.3423551

observed statistics:  28
estimated parameters:  14
degrees of freedom:  14
-2 log likelihood:  517.56
saturated -2 log likelihood:  473.0199
number of observations:  53
chi-square:  44.54013
p:  4.842873e-05
AIC (Mx):  16.54013
BIC (Mx):  -5.521976
adjusted BIC:
RMSEA:  0.2028773
timestamp: 2011-08-16 11:19:23
frontend time: 1.696389 secs
backend time: 0.01884508 secs
independent submodels time: 0.0001170635 secs
wall clock time: 1.715351 secs
cpu time: 1.715351 secs
openmx version number: 1.0.7-1706
```

lavaan Package

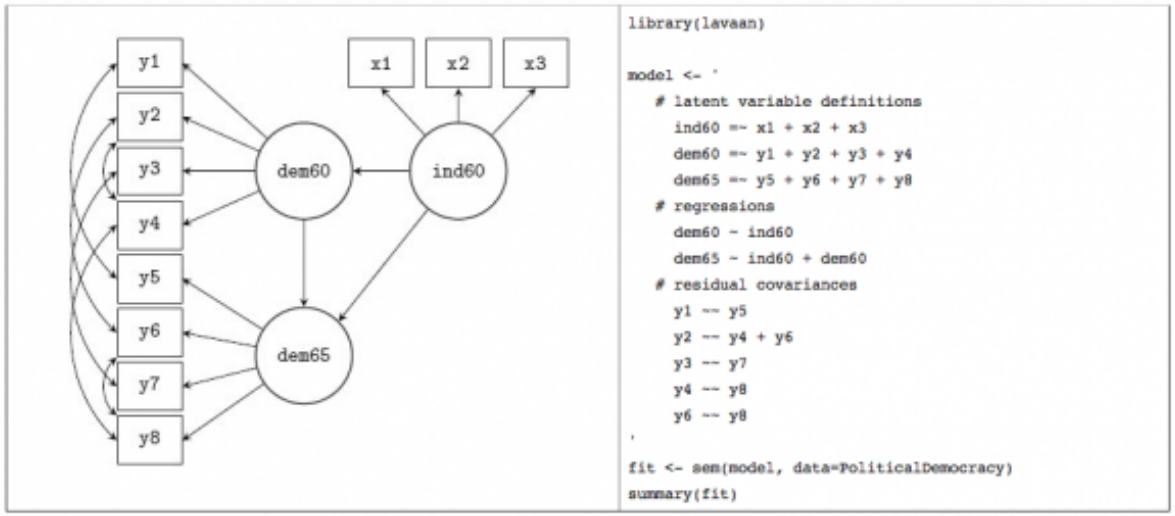


Figure: SEM description in lavaan website

lavaan package: model syntax

```
#An analysis for useR conference 2011

# 1. Load data
hoteldata <- read.csv("/Users/pairachpiboonrungroj/Documents/PhD/Analysis/R/
 useR/cleandata.csv")
# 2. Install Package
install.packages("lavaan")

# 3. Load Package
library(lavaan)
# 4. Structural Model

TC.Model <- '
# latent variable definitions
cost =~ TC1 + TC2 + TC3 +TC6 + TC7   +TC11  + TC13
'

fitTC <- sem(TC.Model, data = hoteldata)
summary(fitTC, standardized = TRUE, fit.measures=TRUE)


```

Figure: lavaan model syntax

lavaan package: output1

```
Lavaan (0.4-9) converged normally after 24 iterations

  Number of observations                            53

  Estimator                                         ML
  Minimum Function Chi-square                   45.397
  Degrees of freedom                                14
  P-value                                        0.000

Chi-square test baseline model:

  Minimum Function Chi-square                  248.133
  Degrees of freedom                                21
  P-value                                        0.000

Full model versus baseline model:

  Comparative Fit Index (CFI)                    0.862
  Tucker-Lewis Index (TLI)                       0.793

Loglikelihood and Information Criteria:

  Loglikelihood user model (H0)                -601.149
  Loglikelihood unrestricted model (H1)        -578.451

  Number of free parameters                         14
  Akaike (AIC)                                1230.299
  Bayesian (BIC)                              1257.883
  Sample-size adjusted Bayesian (BIC)         1213.909

Root Mean Square Error of Approximation:

  RMSEA                                          0.206
  90 Percent Confidence Interval          0.141  0.274
  P-value RMSEA <= 0.05                          0.000

Standardized Root Mean Square Residual:

  SRMR                                           0.068
```

Figure: lavaan model syntax

lavaan package: output2

```
Parameter estimates:

  Information                                 Expected
  Standard Errors                             Standard

                   Estimate  Std.err  Z-value  P(>|z|)   Std.lv  Std.all
Latent variables:
  cost =~
    TC1               1.000                               1.271    0.748
    TC2               0.989    0.160    6.194    0.000    1.258    0.835
    TC3               0.960    0.162    5.931    0.000    1.221    0.803
    TC6               1.077    0.168    6.414    0.000    1.369    0.863
    TC7               0.841    0.186    4.514    0.000    1.069    0.625
    TC11              0.863    0.165    5.239    0.000    1.097    0.717
    TC13              0.945    0.185    5.098    0.000    1.202    0.700

Variances:
    TC1               1.273    0.280    4.543    0.000    1.273    0.441
    TC2               0.685    0.170    4.037    0.000    0.685    0.302
    TC3               0.821    0.192    4.278    0.000    0.821    0.355
    TC6               0.643    0.172    3.745    0.000    0.643    0.255
    TC7               1.781    0.368    4.845    0.000    1.781    0.609
    TC11              1.135    0.244    4.644    0.000    1.135    0.485
    TC13              1.505    0.321    4.693    0.000    1.505    0.510
    cost              1.617    0.521    3.104    0.002    1.000    1.000
```

Figure: lavaan model syntax

Result Comparison

One latent variable: TC
Seven manifest variables: TC1 - TC7

Table: SEM outputs from two proprietary software and three R packages

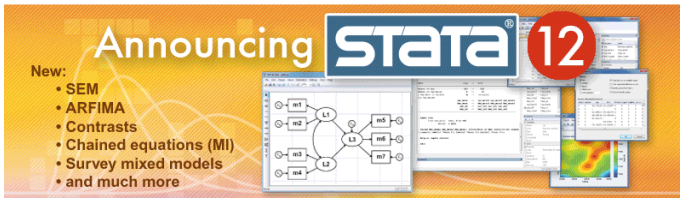
Fit Indices	**LISREL**	**AMOS**	`sem`	`OpenMx`	`lavaan`
χ^2	44.54	44.540	44.54	44.54013	45.397
(df)	(14)	(14)	(14)	(14)	(14)
CFI	0.86	0.863	0.86271	NA	0.862
GFI	0.82	0.82	0.81983	NA	NA
NFI	0.82	0.817	0.81705	NA	NA
NNFI	0.79	NA	0.79407	NA	NA
TLI	NA	0.794	NA	NA	0.793
RMSEA	0.20	0.205	0.20482	0.2028773	0.206
SRMR	0.07	NA	0.067909	NA	0.068

Conclusion 1

- Considering opportunity cost and governance problem, this study proposed and tested a new measurement for transaction cost but yet well fitted one (very small sample size).
- R packages can be used to fit SEM
  - Identical or almost to those of commercial software.
  - lavaan is probably the most useR-friendly package in R.
  - OpenMx offers alternative approach (Matrix specification) and powerful.
- Challenges of R packages for SEM

Conclusion 2

- Challenges of R packages for SEM
  - More user-friendly?
    - by just *Drawing* like AMOS? Proposed in OpenMx
    - by just *Clicking* as a plugin in Rcmdr(John Fox)
  - Publishing SEM research using R package(s)?
  - SEM is available in Stata12 (either drawing or coding)



- Comparing with more advance SEM model e.g., multiple group, multilevel or growth curve model

Acknowledgements

"The author is grateful to
the **Royal Thai Government**
through the *Commission on Higher Education*
for financial support of this study."

Thank you very much

Any suggestion?

Result

- Cronbachs alpha was greater than 0.7
- Chi-square $= 40.244$, (d.f. $= 37$, $p = 0.329$)
  CFI $= 0.989$, TLI $= 0.983$ and RMSEA $= 0.041$.
- Coefficients: uncertainty 0.458 ($p = 0.031$) and asset specificity 0.622 ($p < 0.001$)

Construct measures with reliability and factor loadings

Measurement Items	Factor Loadings
Asset specificity ($\alpha = .718$)	
In building the relationship with my firm, this supplier . . .	
. . . has an operating process that has been tailored.	1.000
. . . has made specific investments in resources.	0.862
Uncertainty ($\alpha = .702$)	
My firm can accurately predict the performance of this supplier in our next transaction.	1.000
My firm knows that this supplier will adapt quickly, should we have change our specifications at short notice.	0.693
Transaction cost ($\alpha = .880$)	
It is very complicated and difficult to write a contract.	1.000
It took a significant effort to gather the critical information.	0.916
It is very difficult to monitor the performance of this supplier.	0.926
It takes a lot of effort to solve problems in our relationship.	0.890
This supplier tends to take advantage from my hotel with guile.	0.705
It is very difficult to assess the performance of this supplier.	0.742
We should better select other suppliers.	0.688

'sem' package

- sem

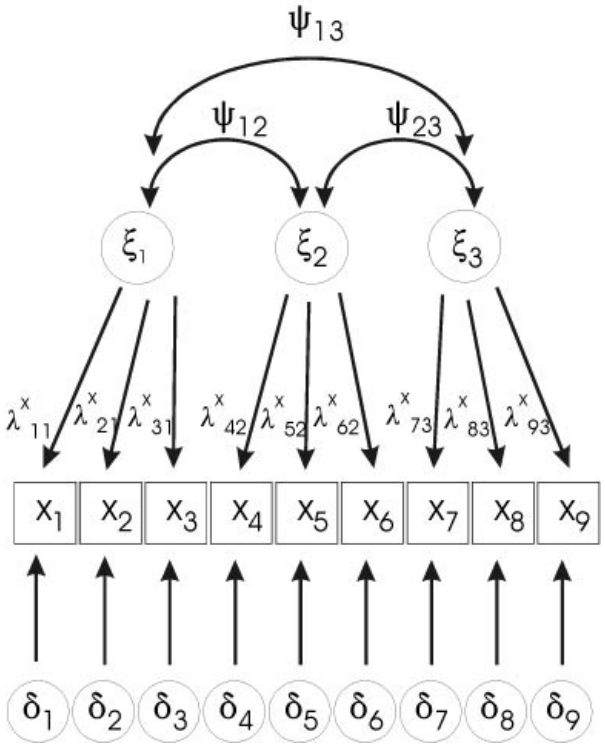


Figure: http://socserv.socsci.mcmaster.ca/jfox/Courses/Brazil-2008/index.html

Supply Chain Collaborations

Definition

"At least two firms in the same supply chain work together to achieve their mutual goals"
(Mentzer et al., 2001; Simatupang and Sridharan, 2005).

Costs and Benefits of Supply Chain Collaborations

Costs and Benefits

- **Costs**: Finding the best suppliers (e.g., price, quality), Monitoring (QC) (Barratt, 2004; Holweg et al., 2005)
- **Benefits**: Better level of responsiveness and service level (Speckman, 1998; Holweg et al., 2005)