

# ***Conventional Linear Regression***

## ***Introduction***

This example demonstrates a conventional regression analysis, predicting a single observed variable as a linear combination of three other observed variables. It also introduces the concept of **identifiability**.

## ***About the Data***

Warren, White, and Fuller (1974) studied 98 managers of farm cooperatives. We will use the following four measurements:

Test	Explanation
<i>performance</i>	A 24-item test of performance related to “planning, organization, controlling, coordinating, and directing”
<i>knowledge</i>	A 26-item test of knowledge of “economic phases of management directed toward profit-making...and product knowledge”
<i>value</i>	A 30-item test of “tendency to rationally evaluate means to an economic end”
<i>satisfaction</i>	An 11-item test of “gratification obtained...from performing the managerial role”

A fifth measure, *past training*, was also reported, but we will not use it.

In this example, you will use the Excel worksheet *Warren5v* in the file *UserGuide.xls*, which is located in the *Examples* folder. If you performed a typical installation, the path is *C:\Program Files\IBM\SPSS\Amos\20\Examples\<language>*.

Here are the sample variances and covariances:

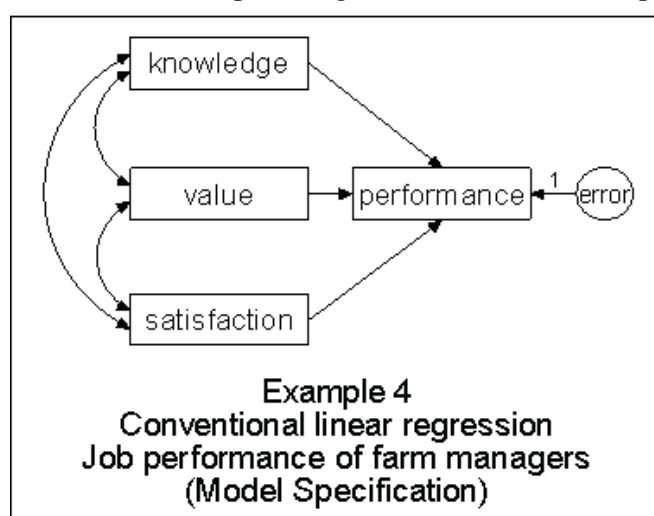
rowtype_	varname_	performance	knowledge	value	satisfaction	past_training
n		98	98	98	98	98
cov	performance	0.0209				
cov	knowledge	0.0177	0.052			
cov	value	0.0245	0.028	0.1212		
cov	satisfaction	0.0046	0.0044	-0.0063	0.0901	
cov	past_training	0.0187	0.0192	0.0353	-0.0066	0.0946
mean		0.0589	1.3796	2.8773	2.4613	2.1174

*Warren5v* also contains the sample means. Raw data are not available, but they are not needed by Amos for most analyses, as long as the sample moments (that is, means, variances, and covariances) are provided. In fact, only sample variances and covariances are required in this example. We will not need the sample means in *Warren5v* for the time being, and Amos will ignore them.

## Analysis of the Data

Suppose you want to use scores on *knowledge*, *value*, and *satisfaction* to predict *performance*. More specifically, suppose you think that *performance* scores can be approximated by a linear combination of *knowledge*, *value*, and *satisfaction*. The prediction will not be perfect, however, and the model should thus include an *error* variable.

Here is the initial path diagram for this relationship:



The single-headed arrows represent linear dependencies. For example, the arrow leading from *knowledge* to *performance* indicates that performance scores depend, in part, on knowledge. The variable *error* is enclosed in a circle because it is not directly observed. Error represents much more than random fluctuations in performance scores due to measurement error. Error also represents a composite of age, socioeconomic status, verbal ability, and anything else on which performance may depend but which was not measured in this study. This variable is essential because the path diagram is supposed to show *all* variables that affect performance scores. Without the circle, the path diagram would make the implausible claim that performance is an *exact* linear combination of knowledge, value, and satisfaction.

The double-headed arrows in the path diagram connect variables that may be correlated with each other. The absence of a double-headed arrow connecting *error* with any other variable indicates that *error* is assumed to be uncorrelated with every other predictor variable—a fundamental assumption in linear regression. *Performance* is also not connected to any other variable by a double-headed arrow, but this is for a different reason. Since performance depends on the other variables, it goes without saying that it might be correlated with them.

## Specifying the Model

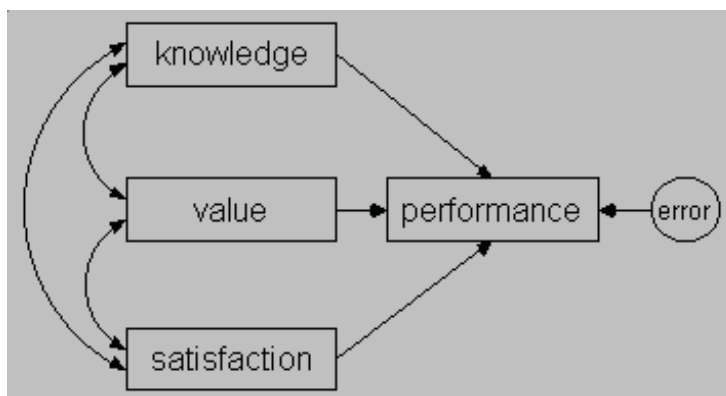
Using what you learned in the first three examples, do the following:

- ▶ Start a new path diagram.
- ▶ Specify that the dataset to be analyzed is in the Excel worksheet *Warren5v* in the file *UserGuide.xls*.
- ▶ Draw four rectangles and label them knowledge, value, satisfaction, and performance.
- ▶ Draw an ellipse for the *error* variable.
- ▶ Draw single-headed arrows that point from the **exogenous**, or predictor, variables (*knowledge*, *value*, *satisfaction*, and *error*) to the **endogenous**, or response, variable (*performance*).

**Note:** Endogenous variables have at least one single-headed path pointing toward them. Exogenous variables, in contrast, send out only single-headed paths but do not receive any.

- Draw three double-headed arrows that connect the observed exogenous variables (*knowledge*, *satisfaction*, and *value*).

Your path diagram should look like this:



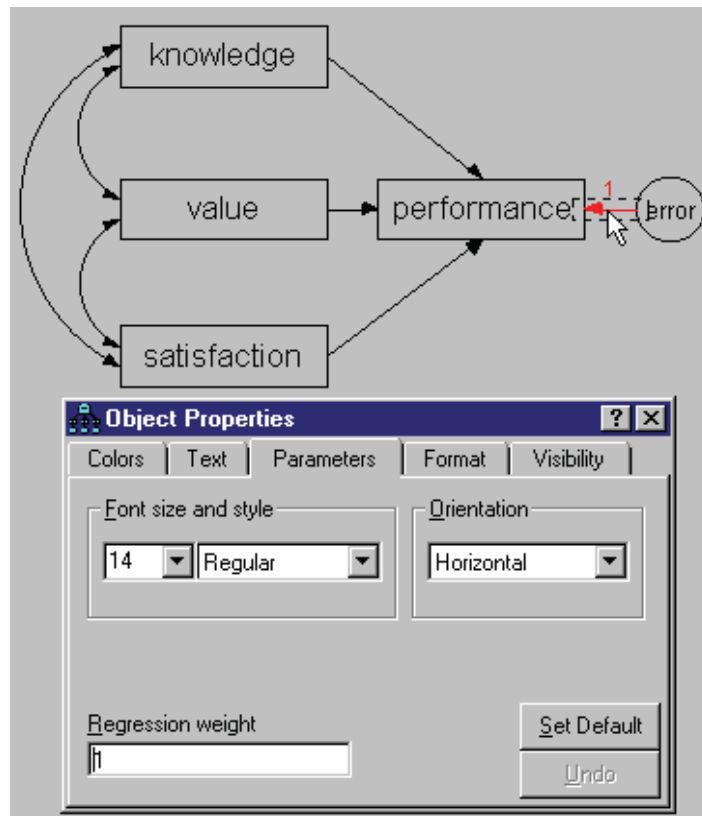
## Identification

In this example, it is impossible to estimate the regression weight for the regression of *performance* on *error*, and, at the same time, estimate the variance of *error*. It is like having someone tell you, “I bought \$5 worth of widgets,” and attempting to infer both the price of each widget and the number of widgets purchased. There is just not enough information.

You can solve this *identification* problem by fixing either the regression weight applied to *error* in predicting *performance*, or the variance of the *error* variable itself, at an arbitrary, nonzero value. Let’s fix the regression weight at 1. This will yield the same estimates as conventional linear regression.

## Fixing Regression Weights

- Right-click the arrow that points from error to performance and choose Object Properties from the pop-up menu.
- Click the Parameters tab.
- Type 1 in the Regression weight box.



Setting a regression weight equal to 1 for every *error* variable can be tedious. Fortunately, Amos Graphics provides a default solution that works well in most cases.

- ▶ Click the Add a unique variable to an existing variable button.
- ▶ Click an endogenous variable.

Amos automatically attaches an error variable to it, complete with a fixed regression weight of 1. Clicking the endogenous variable repeatedly changes the position of the error variable.

## Viewing the Text Output

Here are the maximum likelihood estimates:

<b>Regression Weights: (Group number 1 - Default model)</b>					
	Estimate	S.E.	C.R.	P	Label
performance<---knowledge	.26	.05	4.82	***	
performance<---value	.15	.04	4.14	***	
performance<---satisfaction	.05	.04	1.27	.20	
<b>Covariances: (Group number 1 - Default model)</b>					
	Estimate	S.E.	C.R.	P	Label
knowledge<-->satisfaction	.00	.01	.63	.53	
value <-->satisfaction	-.01	.01	-.59	.55	
knowledge<-->value	.03	.01	3.28	.00	
<b>Variances: (Group number 1 - Default model)</b>					
	Estimate	S.E.	C.R.	P	Label
knowledge	.05	.01	6.96	***	
value	.12	.02	6.96	***	
satisfaction	.09	.01	6.96	***	
error	.01	.00	6.96	***	

Amos does not display the path *performance*  $\leftarrow$  *error* because its value is fixed at the default value of 1. You may wonder how much the other estimates would be affected if a different constant had been chosen. It turns out that only the variance estimate for *error* is affected by such a change.

The following table shows the variance estimate that results from various choices for the *performance*  $\leftarrow$  *error* regression weight.

Fixed regression weight	Estimated variance of error
0.5	0.050
0.707	0.025
1.0	0.0125
1.414	0.00625
2.0	0.00313

Suppose you fixed the path coefficient at 2 instead of 1. Then the variance estimate would be divided by a factor of 4. You can extrapolate the rule that multiplying the path coefficient by a fixed factor goes along with dividing the error variance by the square

of the same factor. Extending this, the product of the squared regression weight and the error variance is always a constant. This is what we mean when we say the regression weight (together with the error variance) is **unidentified**. If you assign a value to one of them, the other can be estimated, but they cannot both be estimated at the same time.

The identifiability problem just discussed arises from the fact that the variance of a variable, and any regression weights associated with it, depends on the units in which the variable is measured. Since *error* is an unobserved variable, there is no natural way to specify a measurement unit for it. Assigning an arbitrary value to a regression weight associated with *error* can be thought of as a way of indirectly choosing a unit of measurement for *error*. Every unobserved variable presents this identifiability problem, which must be resolved by imposing some constraint that determines its unit of measurement.

Changing the scale unit of the unobserved *error* variable does not change the overall model fit. In all the analyses, you get:

Chi-square = 0.00  
 Degrees of freedom = 0  
 Probability level cannot be computed

There are four sample variances and six sample covariances, for a total of 10 sample moments. There are three regression paths, four model variances, and three model covariances, for a total of 10 parameters that must be estimated. Hence, the model has zero degrees of freedom. Such a model is often called **saturated** or **just-identified**.

The standardized coefficient estimates are as follows:

**Standardized Regression Weights: (Group number 1 - Default model)**

	Estimate
performance<--- knowledge	.41
performance<--- value	.35
performance<--- satisfaction	.10

**Correlations: (Group number 1 - Default model)**

	Estimate
knowledge<--> satisfaction	.06
value <--> satisfaction	-.06
knowledge<--> value	.35

The standardized regression weights and the correlations are independent of the units in which all variables are measured; therefore, they are not affected by the choice of identification constraints.

Squared multiple correlations are also independent of units of measurement. Amos displays a squared multiple correlation for each endogenous variable.

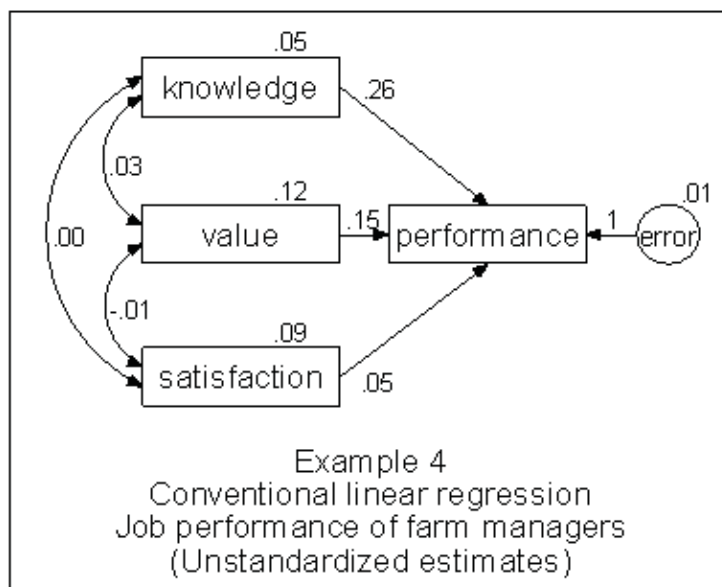
**Squared Multiple Correlations: (Group number 1 - Default model)**

	Estimate
performance	.40

**Note:** The squared multiple correlation of a variable is the proportion of its variance that is accounted for by its predictors. In the present example, *knowledge*, *value*, and *satisfaction* account for 40% of the variance of *performance*.

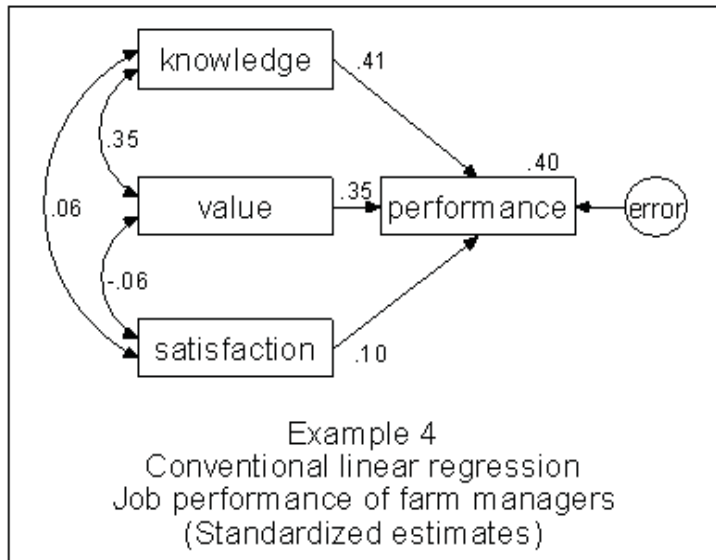
## Viewing Graphics Output

The following path diagram output shows unstandardized values:





Here is the standardized solution:



## Viewing Additional Text Output

- In the tree diagram in the upper left pane of the Amos Output window, click Variable Summary.

Variable Summary (Group number 1)	
Your model contains the following variables (Group number 1)	
Observed, endogenous variables	
performance	
Observed, exogenous variables	
knowledge	
value	
satisfaction	
Unobserved, exogenous variables	
error	
Variable counts (Group number 1)	
Number of variables in your model:	5
Number of observed variables:	4
Number of unobserved variables:	1
Number of exogenous variables:	4
Number of endogenous variables:	1

**Endogenous** variables are those that have single-headed arrows pointing to them; they depend on other variables. **Exogenous** variables are those that do not have single-headed arrows pointing to them; they do not depend on other variables.

Inspecting the preceding list will help you catch the most common (and insidious) errors in an input file: typing errors. If you try to type **performance** twice but unintentionally misspell it as **preformance** one of those times, both versions will appear on the list.

- Now click Notes for Model in the upper left pane of the Amos Output window.

The following output indicates that there are no feedback loops in the path diagram:

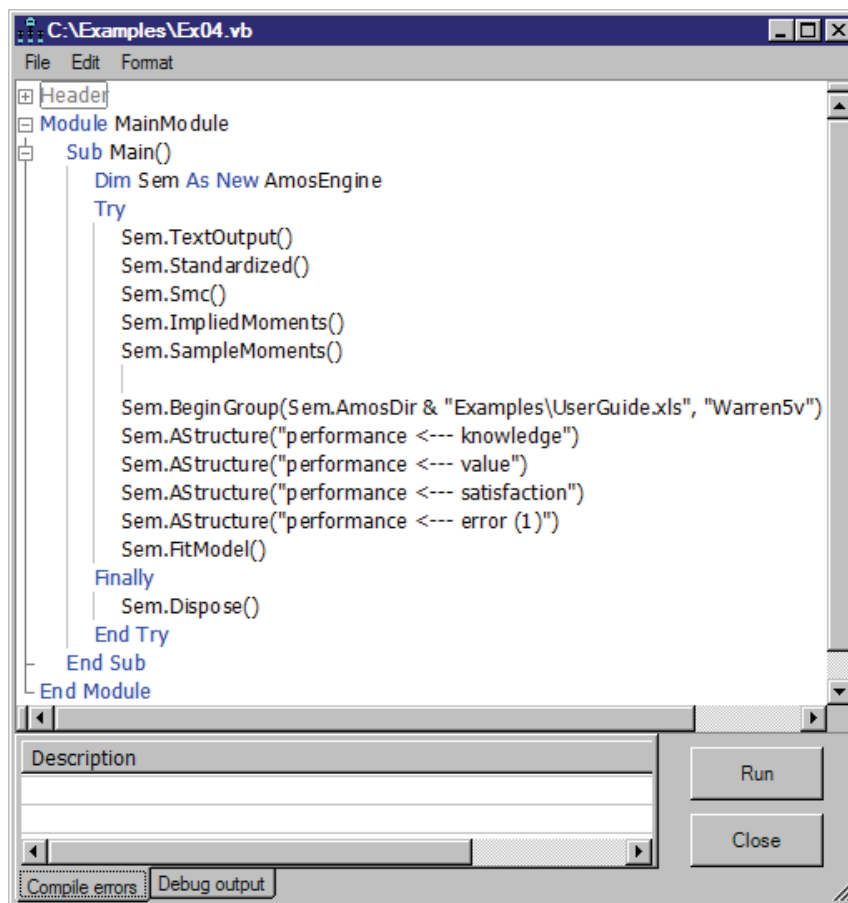
Notes for Group (Group number 1) The model is recursive.
---

Later you will see path diagrams where you can pick a variable and, by tracing along the single-headed arrows, follow a path that leads back to the same variable.

**Note:** Path diagrams that have feedback loops are called **nonrecursive**. Those that do not are called **recursive**.

## Modeling in VB.NET

The model in this example consists of a single regression equation. Each single-headed arrow in the path diagram represents a regression weight. Here is a program for estimating those regression weights:



The four lines that come after `Sem.BeginGroup` correspond to the single-headed arrows in the Amos Graphics path diagram. The (1) in the last `AStructure` line fixes the error regression weight at a constant 1.

## Assumptions about Correlations among Exogenous Variables

When executing a program, Amos makes assumptions about the correlations among exogenous variables that are not made in Amos Graphics. These assumptions simplify

the specification of many models, especially models that have parameters. The differences between specifying a model in Amos Graphics and specifying one programmatically are as follows:

- Amos Graphics is entirely WYSIWYG (What You See Is What You Get). If you draw a two-headed arrow (without constraints) between two exogenous variables, Amos Graphics will estimate their covariance. If two exogenous variables are not connected by a double-headed arrow, Amos Graphics will assume that the variables are uncorrelated.

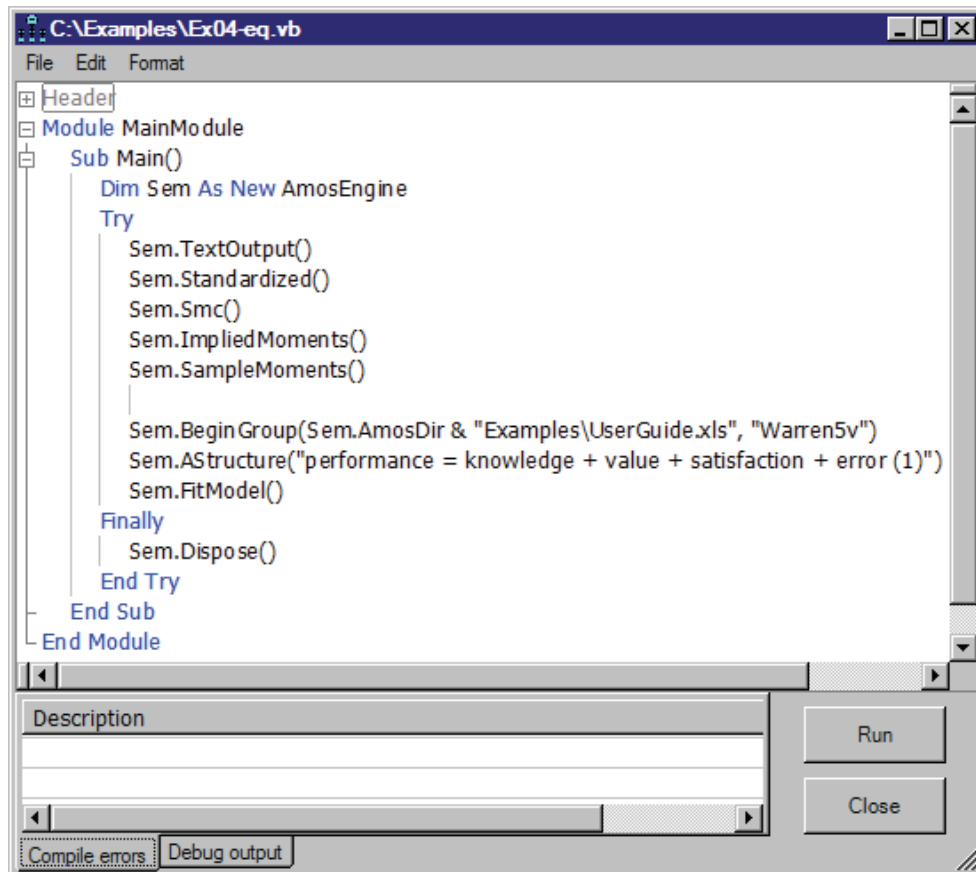
The default assumptions in an Amos program are:

- **Unique** variables (unobserved, exogenous variables that affect only one other variable) are assumed to be uncorrelated with each other and with all other exogenous variables.
- **Exogenous** variables other than unique variables are assumed to be correlated among themselves.

In Amos programs, these defaults reflect standard assumptions of conventional linear regression analysis. Thus, in this example, the program assumes that the predictors, *knowledge*, *value*, and *satisfaction*, are correlated and that *error* is uncorrelated with the predictors.

### ***Equation Format for the AStructure Method***

The AStructure method permits model specification in equation format. For instance, the single Sem.AStructure statement in the following program describes the same model as the program on p. 77 but in a single line. This program is saved under the name *Ex04-eq.vb* in the *Examples* directory.



Note that in the AStructure line above, each predictor variable (on the right side of the equation) is associated with a regression weight to be estimated. We could make these regression weights explicit through the use of empty parentheses as follows:

```
Sem.AStructure("performance = ()knowledge + ()value + ()satisfaction + error(1)")
```

The empty parentheses are optional. By default, Amos will automatically estimate a regression weight for each predictor.



# ***Unobserved Variables***

## ***Introduction***

This example demonstrates a regression analysis with unobserved variables.

## ***About the Data***

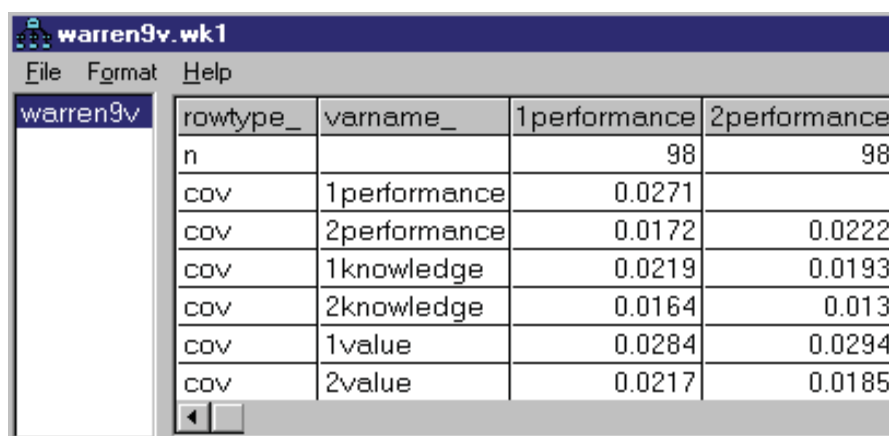
The variables in the previous example were surely unreliable to some degree. The fact that the reliability of *performance* is unknown presents a minor problem when it comes to interpreting the fact that the predictors account for only 39.9% of the variance of *performance*. If the test were extremely unreliable, that fact in itself would explain why the performance score could not be predicted accurately. Unreliability of the predictors, on the other hand, presents a more serious problem because it can lead to biased estimates of regression weights.

The present example, based on Rock, et al. (1977), will assess the reliabilities of the four tests included in the previous analysis. It will also obtain estimates of regression weights for perfectly reliable, hypothetical versions of the four tests. Rock, et al. re-examined the data of Warren, White, and Fuller (1974) that were discussed in the previous example. This time, each test was randomly split into two halves, and each half was scored separately.

Here is a list of the input variables:

Variable name	Description
<i>1performance</i>	12-item subtest of Role Performance
<i>2performance</i>	12-item subtest of Role Performance
<i>1knowledge</i>	13-item subtest of Knowledge
<i>2knowledge</i>	13-item subtest of Knowledge
<i>1value</i>	15-item subtest of Value Orientation
<i>2value</i>	15-item subtest of Value Orientation
<i>1satisfaction</i>	5-item subtest of Role Satisfaction
<i>2satisfaction</i>	6-item subtest of Role Satisfaction
<i>past_training</i>	degree of formal education

For this example, we will use a Lotus data file, *Warren9v.wk1*, to obtain the sample variances and covariances of these subtests. The sample means that appear in the file will not be used in this example. Statistics on formal education (*past\_training*) are present in the file, but they also will not enter into the present analysis. The following is a portion of the dataset:

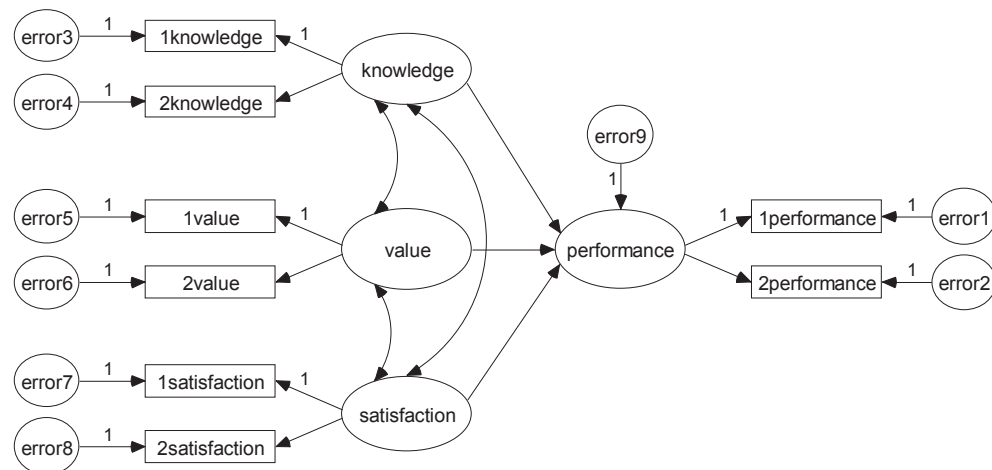


rowtype_	varname_	1performance	2performance
n		98	98
cov	1performance	0.0271	
cov	2performance	0.0172	0.0222
cov	1knowledge	0.0219	0.0193
cov	2knowledge	0.0164	0.013
cov	1value	0.0284	0.0294
cov	2value	0.0217	0.0185



## Model A

The following path diagram presents a model for the eight subtests:

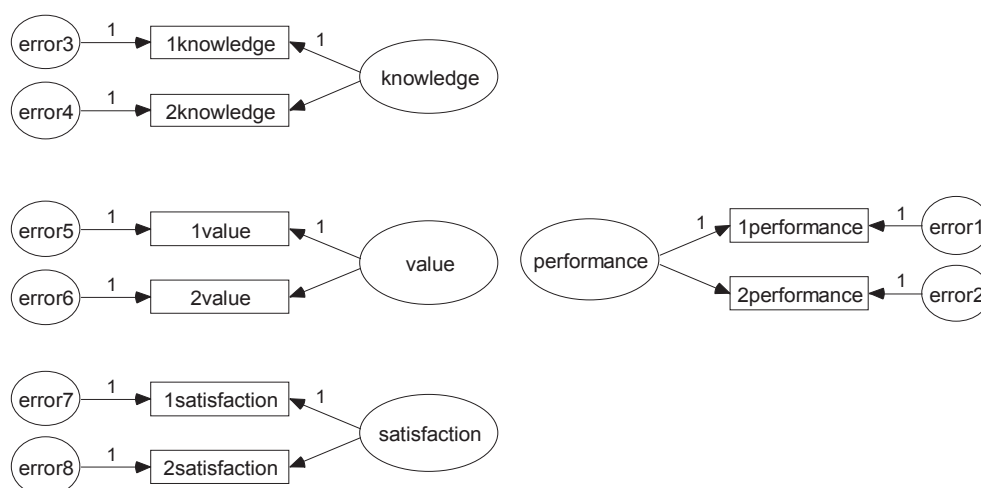


Example 5: Model A  
Regression with unobserved variables  
Job performance of farm managers  
Warren, White and Fuller (1974)  
Standardized estimates

Four ellipses in the figure are labeled *knowledge*, *value*, *satisfaction*, and *performance*. They represent unobserved variables that are indirectly measured by the eight split-half tests.

## Measurement Model

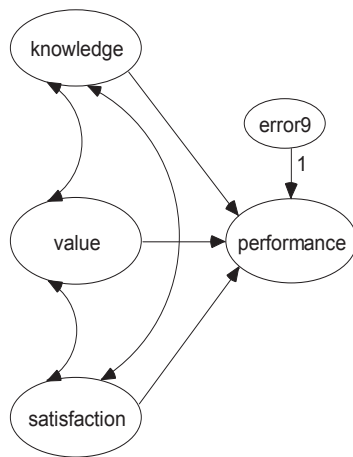
The portion of the model that specifies how the observed variables depend on the unobserved, or latent, variables is sometimes called the **measurement model**. The current model has four distinct measurement submodels.



Consider, for instance, the *knowledge* submodel: The scores of the two split-half subtests, *1knowledge* and *2knowledge*, are hypothesized to depend on the single underlying, but not directly observed variable, *knowledge*. According to the model, scores on the two subtests may still disagree, owing to the influence of *error3* and *error4*, which represent errors of measurement in the two subtests. *1knowledge* and *2knowledge* are called **indicators** of the latent variable *knowledge*. The measurement model for *knowledge* forms a pattern that is repeated three more times in the path diagram shown above.

## Structural Model

The portion of the model that specifies how the latent variables are related to each other is sometimes called the **structural model**.



The structural part of the current model is the same as the one in Example 4. It is only in the measurement model that this example differs from the one in Example 4.

## Identification

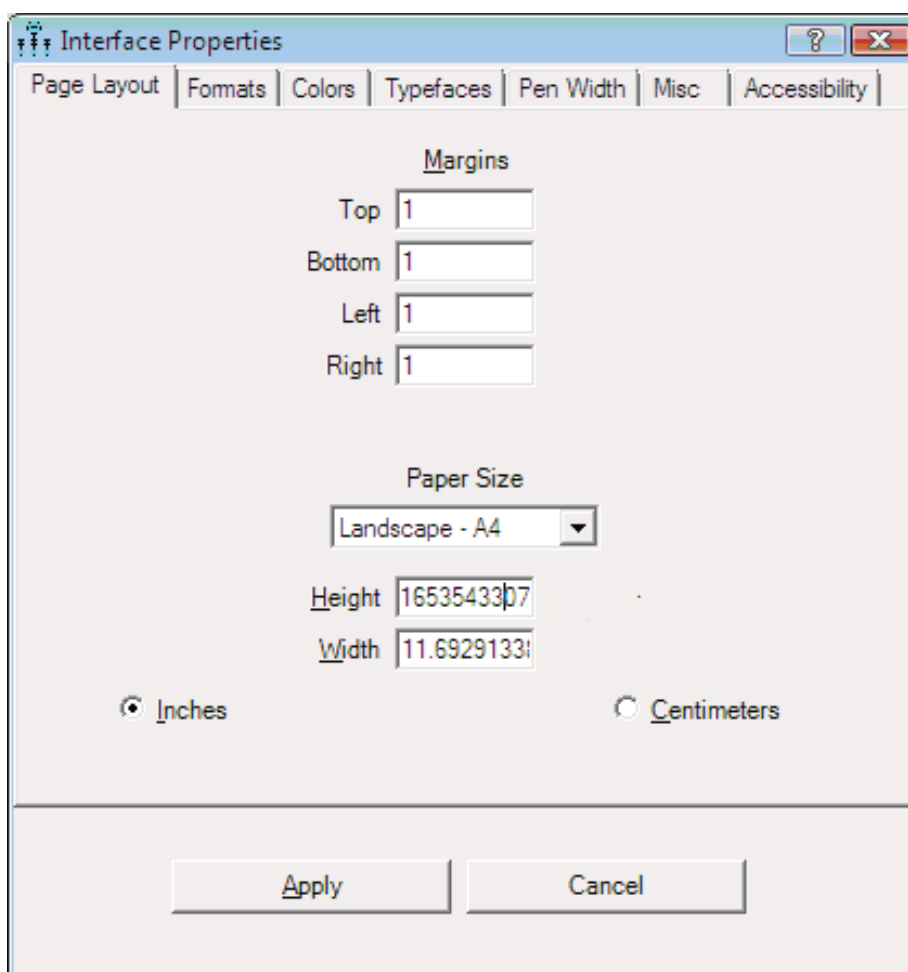
With 13 unobserved variables in this model, it is certainly not identified. It will be necessary to fix the unit of measurement of each unobserved variable by suitable constraints on the parameters. This can be done by repeating 13 times the trick that was used for the single unobserved variable in Example 4: Find a single-headed arrow leading away from each unobserved variable in the path diagram, and fix the corresponding regression weight to an arbitrary value such as 1. If there is more than one single-headed arrow leading away from an unobserved variable, any one of them will do. The path diagram for “Model A” on p. 83 shows one satisfactory choice of identifiability constraints.

## Specifying the Model

Because the path diagram is wider than it is tall, you may want to change the shape of the drawing area so that it fits the path diagram better. By default, the drawing area in Amos is taller than it is wide so that it is suitable for printing in portrait mode.

## Changing the Orientation of the Drawing Area

- ▶ From the menus, choose View → Interface Properties.
- ▶ In the Interface Properties dialog box, click the Page Layout tab.
- ▶ Set Paper Size to one of the “Landscape” paper sizes, such as Landscape - A4.

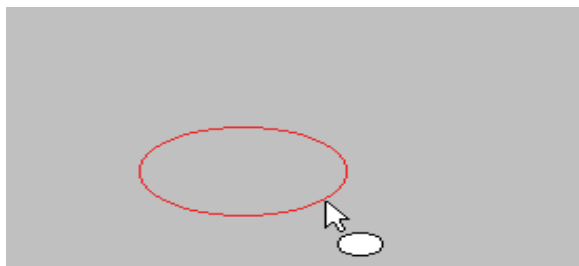


- ▶ Click Apply.

## Creating the Path Diagram

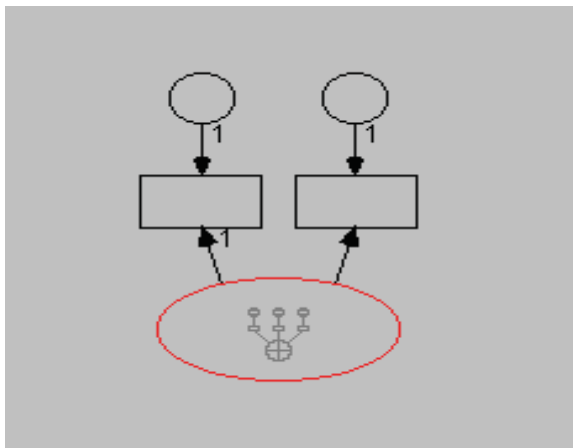
Now you are ready to draw the model as shown in the path diagram on page 83. There are a number of ways to do this. One is to start by drawing the measurement model first. Here, we draw the measurement model for one of the latent variables, *knowledge*, and then use it as a pattern for the other three.

- Draw an ellipse for the unobserved variable *knowledge*.



- From the menus, choose Diagram → Draw Indicator Variable.
- Click twice inside the ellipse.

Each click creates one indicator variable for *knowledge*:



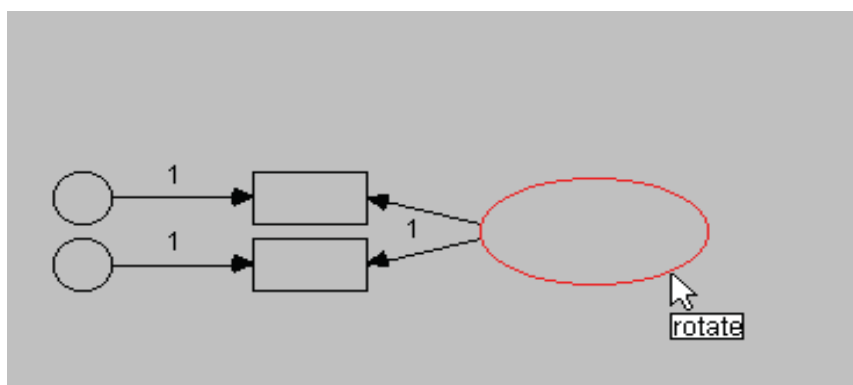
As you can see, with the Draw indicator variable button enabled, you can click multiple times on an unobserved variable to create multiple indicators, complete with unique or error variables. Amos Graphics maintains suitable spacing among the indicators and inserts identification constraints automatically.

## Rotating Indicators

The indicators appear by default above the *knowledge* ellipse, but you can change their location.

- ▶ From the menus, choose Edit → Rotate.
- ▶ Click the *knowledge* ellipse.

Each time you click the *knowledge* ellipse, its indicators rotate 90° clockwise. If you click the ellipse three times, its indicators will look like this:

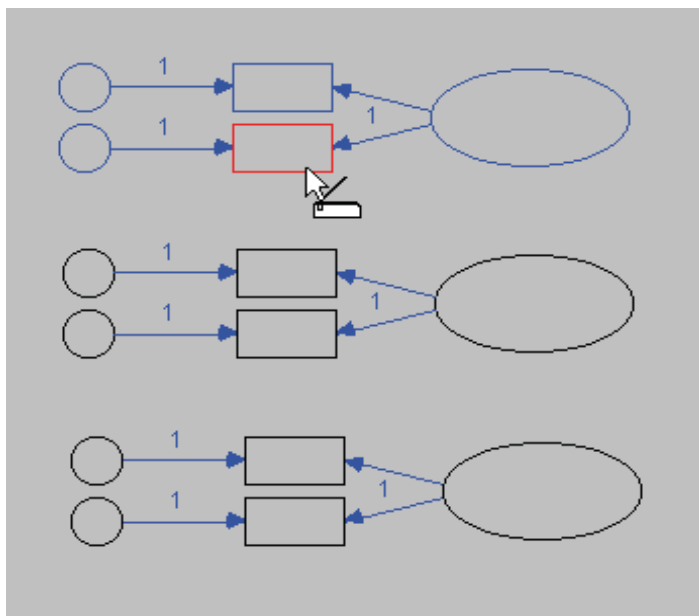


## Duplicating Measurement Models

The next step is to create measurement models for *value* and *satisfaction*.

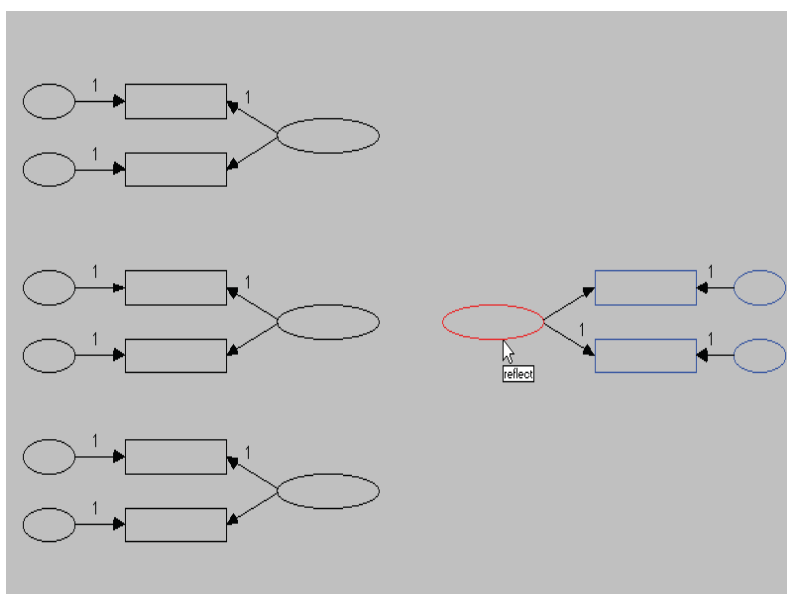
- ▶ From the menus, choose Edit → Select All.  
The measurement model turns blue.
- ▶ From the menus, choose Edit → Duplicate.
- ▶ Click any part of the measurement model, and drag a copy to beneath the original.
- ▶ Repeat to create a third measurement model above the original.

Your path diagram should now look like this:



- Create a fourth copy for *performance*, and position it to the right of the original.
- From the menus, choose Edit → Reflect.

This repositions the two indicators of *performance* as follows:



## Entering Variable Names

- ▶ Right-click each object and select Object Properties from the pop-up menu
- ▶ In the Object Properties dialog box, click the Text tab, and enter a name into the Variable Name text box.

Alternatively, you can choose View → Variables in Dataset from the menus and then drag variable names onto objects in the path diagram.

## Completing the Structural Model

There are only a few things left to do to complete the structural model.

- ▶ Draw the three covariance paths connecting *knowledge*, *value*, and *satisfaction*.
- ▶ Draw a single-headed arrow from each of the latent predictors, *knowledge*, *value*, and *satisfaction*, to the latent dependent variable, *performance*.
- ▶ Add the unobserved variable *error9* as a predictor of *performance* (from the menus, choose Diagram → Draw Unique Variable).

Your path diagram should now look like the one on p. 83. The Amos Graphics input file that contains this path diagram is *Ex05-a.amw*.

## Results for Model A

As an exercise, you might want to confirm the following degrees of freedom calculation:

Computation of degrees of freedom (Default model)	
Number of distinct sample moments:	36
Number of distinct parameters to be estimated:	22
Degrees of freedom (36 - 22):	14



The hypothesis that Model A is correct is accepted.

Chi-square = 10.335  
 Degrees of freedom = 14  
 Probability level = 0.737

The parameter estimates are affected by the identification constraints.

<b>Regression Weights: (Group number 1 - Default model)</b>					
	Estimate	S.E.	C.R.	P	Label
performance <--- knowledge	.337	.125	2.697	.007	
performance <--- satisfaction	.061	.054	1.127	.260	
performance <--- value	.176	.079	2.225	.026	
2satisfaction <--- satisfaction	.792	.438	1.806	.071	
1satisfaction <--- satisfaction	1.000				
2value <--- value	.763	.185	4.128	***	
1value <--- value	1.000				
2knowledge <--- knowledge	.683	.161	4.252	***	
1knowledge <--- knowledge	1.000				
1performance <--- performance	1.000				
2performance <--- performance	.867	.116	7.450	***	
<b>Covariances: (Group number 1 - Default model)</b>					
	Estimate	S.E.	C.R.	P	Label
value <--> knowledge	.037	.012	3.036	.002	
satisfaction <--> value	-.008	.013	-.610	.542	
satisfaction <--> knowledge	.004	.009	.462	.644	
<b>Variances: (Group number 1 - Default model)</b>					
	Estimate	S.E.	C.R.	P	Label
satisfaction	.090	.052	1.745	.081	
value	.100	.032	3.147	.002	
knowledge	.046	.015	3.138	.002	
error9	.007	.003	2.577	.010	
error3	.041	.011	3.611	***	
error4	.035	.007	5.167	***	
error5	.080	.025	3.249	.001	
error6	.087	.018	4.891	***	
error7	.022	.049	.451	.652	
error8	.045	.032	1.420	.156	
error1	.007	.002	3.110	.002	
error2	.007	.002	3.871	***	

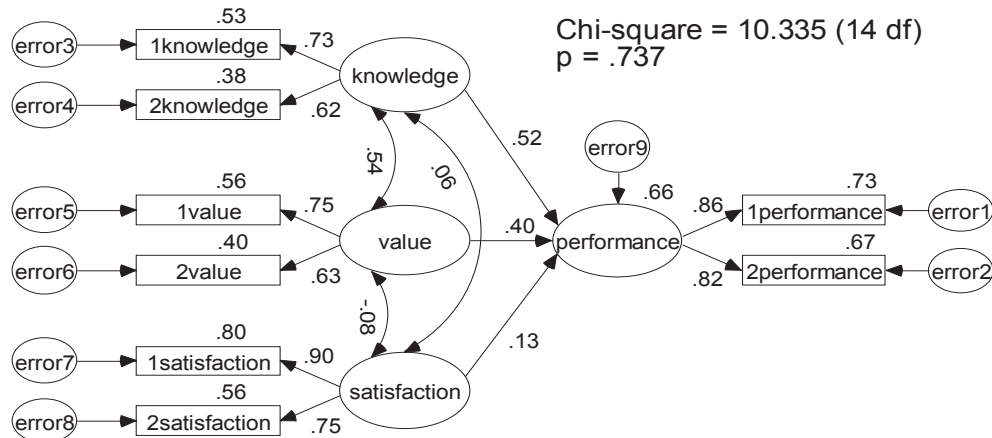
Standardized estimates, on the other hand, are not affected by the identification constraints. To calculate standardized estimates:

- From the menus, choose View → Analysis Properties.
- In the Analysis Properties dialog box, click the Output tab.
- Enable the Standardized estimates check box.

<b>Standardized Regression Weights: (Group number 1 - Default model)</b>			
			Estimate
performance <---	knowledge		.516
performance <---	satisfaction		.130
performance <---	value		.398
2satisfaction <---	satisfaction		.747
1satisfaction <---	satisfaction		.896
2value <---	value		.633
1value <---	value		.745
2knowledge <---	knowledge		.618
1knowledge <---	knowledge		.728
1performance <---	performance		.856
2performance <---	performance		.819
<b>Correlations: (Group number 1 - Default model)</b>			
			Estimate
value <-->	knowledge		.542
satisfaction <-->	value		-.084
satisfaction <-->	knowledge		.064

## Viewing the Graphics Output

The path diagram with standardized parameter estimates displayed is as follows:



Example 5: Model A  
Regression with unobserved variables  
Job performance of farm managers  
Warren, White and Fuller (1974)  
Standardized estimates

The value above *performance* indicates that *pure knowledge*, *value*, and *satisfaction* account for 66% of the variance of *performance*. The values displayed above the observed variables are reliability estimates for the eight individual subtests. A formula for the reliability of the original tests (before they were split in half) can be found in Rock et al. (1977) or any book on mental test theory.

## Model B

Assuming that Model A is correct (and there is no evidence to the contrary), consider the additional hypothesis that *1knowledge* and *2knowledge* are parallel tests. Under the parallel tests hypothesis, the regression of *1knowledge* on *knowledge* should be the same as the regression of *2knowledge* on *knowledge*. Furthermore, the *error* variables associated with *1knowledge* and *2knowledge* should have identical variances. Similar consequences flow from the assumption that *1value* and *2value* are parallel tests, as well as *1performance* and *2performance*. But it is not altogether reasonable to assume that *1satisfaction* and *2satisfaction* are parallel. One of the subtests is slightly longer than the other because the original test had an odd number of items and could not be