Example

2

Testing Hypotheses

Introduction

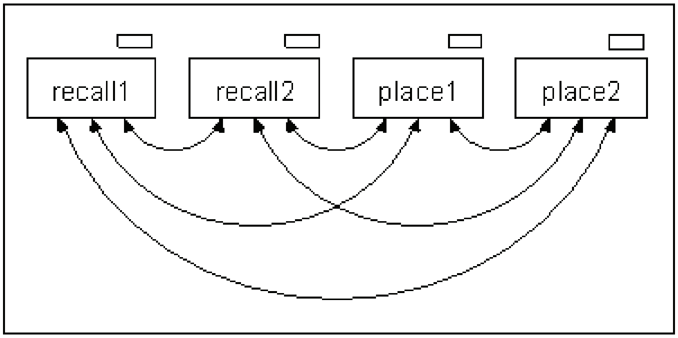
This example demonstrates how you can use Amos to test simple hypotheses about variances and covariances. It also introduces the chi-square test for goodness of fit and elaborates on the concept of degrees of freedom.

About the Data

We will use Attig's (1983) spatial memory data, which were described in Example 1. We will also begin with the same path diagram as in Example 1. To demonstrate the ability of Amos to use different data formats, this example uses a data file in SPSS Statistics format instead of an Excel file.

Parameters Constraints

The following is the path diagram from Example 1. We can think of the variable objects as having small boxes nearby (representing the variances) that are filled in once Amos has estimated the parameters.

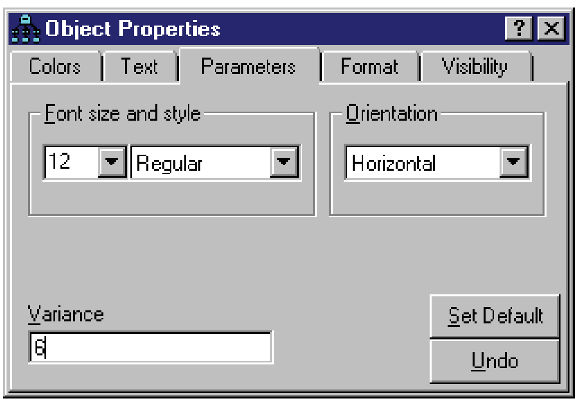


You can fill these boxes yourself instead of letting Amos fill them.

Constraining Variances

Suppose you want to set the variance of *recall1* to 6 and the variance of *recall2* to 8.

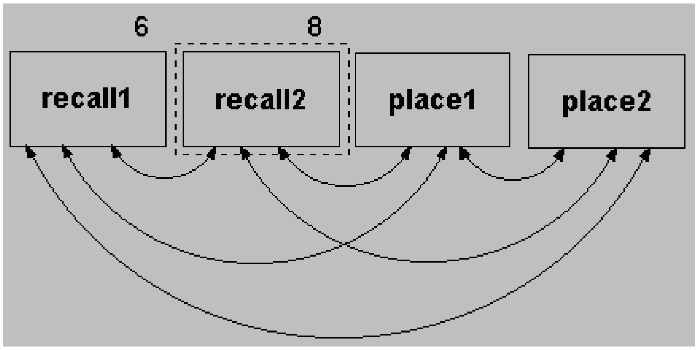
- In the drawing area, right-click *recall1* and choose Object Properties from the pop-up menu.
- Click the Parameters tab.
- In the Variance text box, type 6.



- With the Object Properties dialog box still open, click *recall2* and set its variance to 8.

- Close the dialog box.

  The path diagram displays the parameter values you just specified.



This is not a very realistic example because the numbers 6 and 8 were just picked out of the air. Meaningful parameter constraints must have some underlying rationale, perhaps being based on theory or on previous analyses of similar data.

Specifying Equal Parameters

Sometimes you will be interested in testing whether two parameters are equal in the population. You might, for example, think that the variances of *recall1* and *recall2* might be equal without having a particular value for the variances in mind. To investigate this possibility, do the following:

- In the drawing area, right-click *recall1* and choose Object Properties from the pop-up menu.

- Click the Parameters tab.

- In the Variance text box, type v_recall.

- Click *recall2* and label its variance as v_recall.

- Use the same method to label the *place1* and *place2* variances as v_place.

  It doesn't matter what label you use. The important thing is to enter the same label for each variance you want to force to be equal. The effect of using the same label is to require both of the variances to have the same value without specifying ahead of time what that value is.

Benefits of Specifying Equal Parameters

Before adding any further constraints on the model parameters, let's examine why we might want to specify that two parameters, like the variances of *recall1* and *recall2* or *place1* and *place2*, are equal. Here are two benefits:

- If you specify that two parameters are equal in the population and if you are correct in this specification, then you will get more accurate estimates, not only of the parameters that are equal but usually of the others as well. This is the only benefit if you happen to know that the parameters are equal.
- If the equality of two parameters is a mere hypothesis, requiring their estimates to be equal will result in a test of that hypothesis.

Constraining Covariances

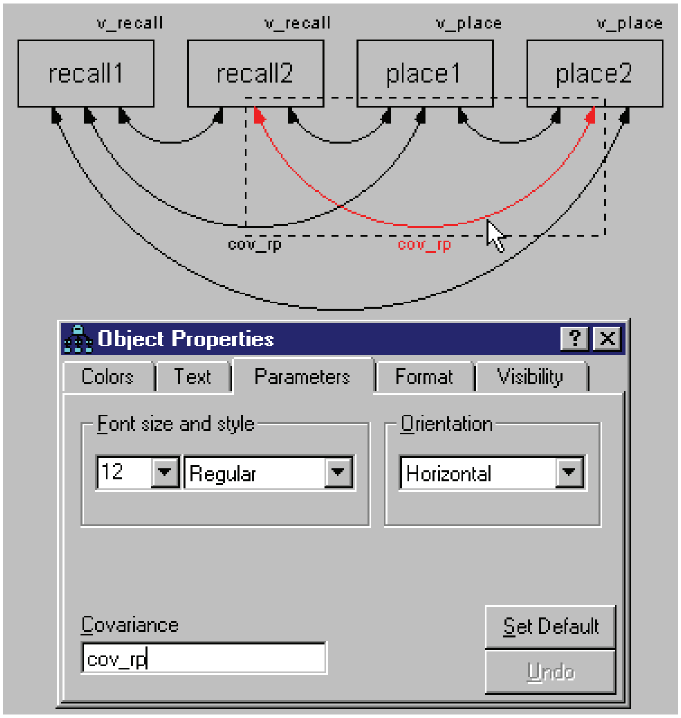
Your model may also include restrictions on parameters other than variances. For example, you may hypothesize that the covariance between *recall1* and *place1* is equal to the covariance between *recall2* and *place2*. To impose this constraint:

▶ In the drawing area, right-click the double-headed arrow that connects *recall1* and *place1*, and choose Object Properties from the pop-up menu.

▶ Click the Parameters tab.

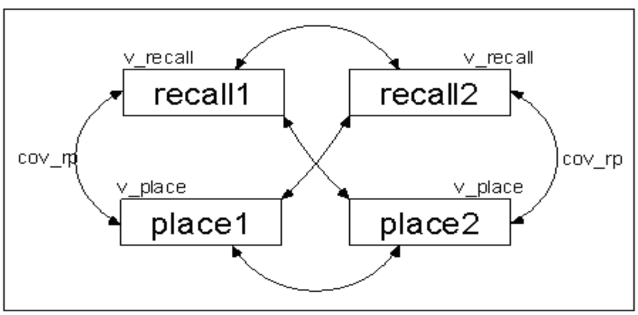
▶ In the Covariance text box, type a non-numeric string such as cov_rp.

▶ Use the same method to set the covariance between *recall2* and *place2* to cov_rp.



Moving and Formatting Objects

While a horizontal layout is fine for small examples, it is not practical for analyses that are more complex. The following is a different layout of the path diagram on which we've been working:



You can use the following tools to rearrange your path diagram until it looks like the one above:

- To move objects, choose Edit → Move from the menus, and then drag the object to its new location. You can also use the Move button to drag the endpoints of arrows.
- To copy formatting from one object to another, choose Edit → Drag Properties from the menus, select the properties you wish to apply, and then drag from one object to another.

For more information about the Drag Properties feature, refer to online Help.

Data Input

This example uses a data file in SPSS Statistics format. If you have SPSS Statistics installed, you can view the data as you load it. Even if you don't have SPSS Statistics installed, Amos will still read the data.

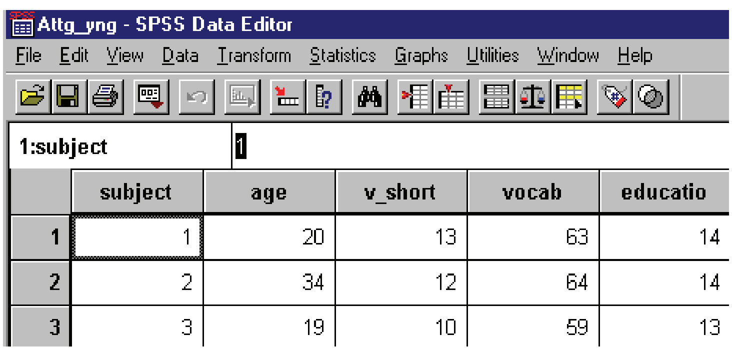
▶ From the menus, choose File → Data Files.

▶ In the Data Files dialog box, click File Name.

▶ Browse to the *Examples* folder. If you performed a typical installation, the path is *C:\Program Files\SPSS Inc\Amos 17.0\Examples*.

▶ In the Files of type list, select *SPSS Statistics* (**.sav*), click *Attg_yng*, and then click Open.

▶ If you have SPSS Statistics installed, click the View Data button in the Data Files dialog box. An SPSS Statistics window opens and displays the data.



▶ Review the data and close the data view.

▶ In the Data Files dialog box, click OK.

Performing the Analysis

▶ From the menus, choose Analyze → Calculate Estimates.

▶ In the Save As dialog box, enter a name for the file and click Save.

Amos calculates the model estimates.

Viewing Text Output

▶ From the menus, choose View → Text Output.

▶ To view the parameter estimates, click Estimates in the tree diagram in the upper left pane of the Amos Output window.

Scalar Estimates (Group number 1 - Default model)

Maximum Likelihood Estimates

Covariances: (Group number 1 - Default model)

			Estimate	S.E.	C.R.	P	Label
recall2	<-->	recall1	2.87	1.21	2.38	.02	
recall2	<-->	place2	2.71	1.82	1.49	.14	cov_rp
place2	<-->	place1	17.15	5.15	3.33	***	
recall1	<-->	place1	2.71	1.82	1.49	.14	cov_rp
recall1	<-->	place2	4.61	2.17	2.13	.03	
recall2	<-->	place1	2.22	2.22	1.00	.32	

Variances: (Group number 1 - Default model)

			Estimate	S.E.	C.R.	P	Label
recall1			7.05	1.22	5.80	***	v_recall
recall2			7.05	1.22	5.80	***	v_recall
place2			27.53	5.18	5.32	***	v_place
place1			27.53	5.18	5.32	***	v_place

You can see that the parameters that were specified to be equal do have equal estimates. The standard errors here are generally smaller than the standard errors obtained in Example 1. Also, because of the constraints on the parameters, there are now positive degrees of freedom.

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

Computation of degrees of freedom (Default model)

Number of distinct sample moments:	10
Number of distinct parameters to be estimated:	7
Degrees of freedom (10 - 7):	3

While there are still 10 sample variances and covariances, the number of parameters to be estimated is only seven. Here is how the number seven is arrived at: The variances of *recall1* and *recall2*, labeled *v_recall*, are constrained to be equal, and thus count as a single parameter. The variances of *place1* and *place2* (labeled *v_place*) count as another single parameter. A third parameter corresponds to the equal covariances *recall1* <> *place1* and *recall2* <> *place2* (labeled *cov_rp*). These three parameters, plus the four unlabeled, unrestricted covariances, add up to seven parameters that have to be estimated.

The degrees of freedom (10 – 7 = 3) may also be thought of as the number of constraints placed on the original 10 variances and covariances.

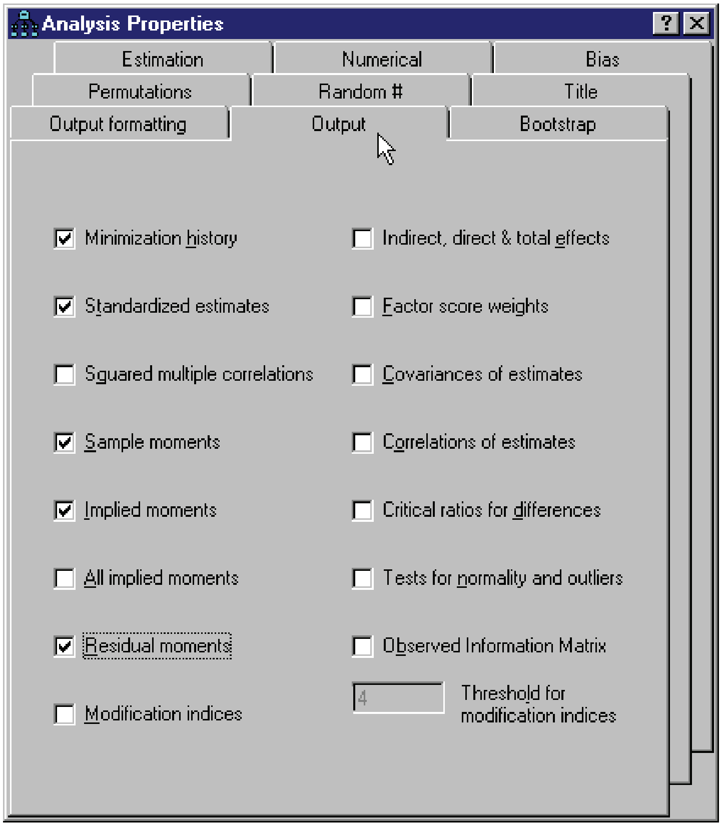
Optional Output

The output we just discussed is all generated by default. You can also request additional output:

▶ From the menus, choose View → Analysis Properties.

▶ Click the Output tab.

▶ Ensure that the following check boxes are selected: Minimization history, Standardized estimates, Sample moments, Implied moments, and Residual moments.



▶ From the menus, choose Analyze → Calculate Estimates.

Amos recalculates the model estimates.

Covariance Matrix Estimates

▶ To see the sample variances and covariances collected into a matrix, choose View → Text Output from the menus.

▶ Click Sample Moments in the tree diagram in the upper left corner of the Amos Output window.

The following is the sample covariance matrix:

Sample Covariances (Group number 1)

	place1	place2	recall1	recall2
place1	33.58			
place2	17.90	22.16		
recall1	4.34	3.57	5.79	
recall2	2.01	.43	2.56	7.94

► In the tree diagram, expand Estimates and then click Matrices.

The following is the matrix of implied covariances:

Implied Covariances (Group number 1 - Default model)

	place1	place2	recall1	recall2
place1	27.53			
place2	17.15	27.53		
recall1	2.71	4.61	7.05	
recall2	2.22	2.71	2.87	7.05

Note the differences between the sample and implied covariance matrices. Because the model imposes three constraints on the covariance structure, the implied variances and covariances are different from the sample values. For example, the sample variance of *place1* is 33.58, but the implied variance is 27.53. To obtain a matrix of residual covariances (sample covariances minus implied covariances), put a check mark next to Residual moments on the Output tab and repeat the analysis.

The following is the matrix of residual covariances:

Residual Covariances (Group number 1 - Default model)

	place1	place2	recall1	recall2
place1	6.05			
place2	.76	-5.37		
recall1	1.63	-1.03	-1.27	
recall2	-.21	-2.28	-.32	.89

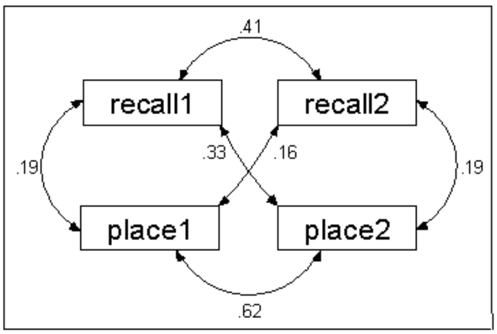
Displaying Covariance and Variance Estimates on the Path Diagram

As in Example 1, you can display the covariance and variance estimates on the path diagram.

▶ Click the Show the output path diagram button.

▶ In the Parameter Formats pane to the left of the drawing area, click Unstandardized estimates. Alternatively, you can request correlation estimates in the path diagram by clicking Standardized estimates.

The following is the path diagram showing correlations:



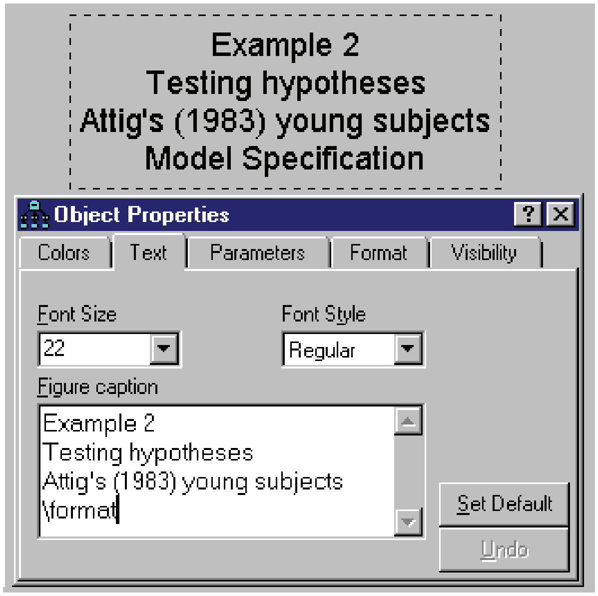
Labeling Output

It may be difficult to remember whether the displayed values are covariances or correlations. To avoid this problem, you can use Amos to label the output.

▶ Open the file *Ex02.amw*.

▶ Right-click the caption at the bottom of the path diagram, and choose Object Properties from the pop-up menu.

▶ Click the Text tab.



Notice the word \format in the bottom line of the figure caption. Words that begin with a backward slash, like \format, are called **text macros**. Amos replaces text macros with information about the currently displayed model. The text macro \format will be replaced by the heading *Model Specification*, *Unstandardized estimates*, or *Standardized estimates*, depending on which version of the path diagram is displayed.

Hypothesis Testing

The implied covariances are the best estimates of the population variances and covariances under the null hypothesis. (The null hypothesis is that the parameters required to have equal estimates are truly equal in the population.) As we know from Example 1, the sample covariances are the best estimates obtained without making any assumptions about the population values. A comparison of these two matrices is relevant to the question of whether the null hypothesis is correct. If the null hypothesis is correct, both the implied and sample covariances are maximum likelihood estimates of the corresponding population values (although the implied covariances are better estimates). Consequently, you would expect the two matrices to resemble each other. On the other hand, if the null hypothesis is wrong, only the sample covariances are maximum likelihood estimates, and there is no reason to expect them to resemble the implied covariances.

The chi-square statistic is an overall measure of how much the implied covariances differ from the sample covariances.

```
Chi-square = 6.276
Degrees of freedom = 3
Probability level = 0.099
```

In general, the more the implied covariances differ from the sample covariances, the bigger the chi-square statistic will be. If the implied covariances had been identical to the sample covariances, as they were in Example 1, the chi-square statistic would have been 0. You can use the chi-square statistic to test the null hypothesis that the parameters required to have equal estimates are really equal in the population. However, it is not simply a matter of checking to see if the chi-square statistic is 0. Since the implied covariances and the sample covariances are merely estimates, you can't expect them to be identical (even if they are both estimates of the same population covariances). Actually, you would expect them to differ enough to produce a chi-square in the neighborhood of the degrees of freedom, even if the null hypothesis is true. In other words, a chi-square value of 3 would not be out of the ordinary here, even with a true null hypothesis. You can say more than that: If the null hypothesis is true, the chi-square value (6.276) is a single observation on a random variable that has an approximate chi-square distribution with three degrees of freedom. The probability is about 0.099 that such an observation would be as large as 6.276. Consequently, the evidence against the null hypothesis is not significant at the 0.05 level.

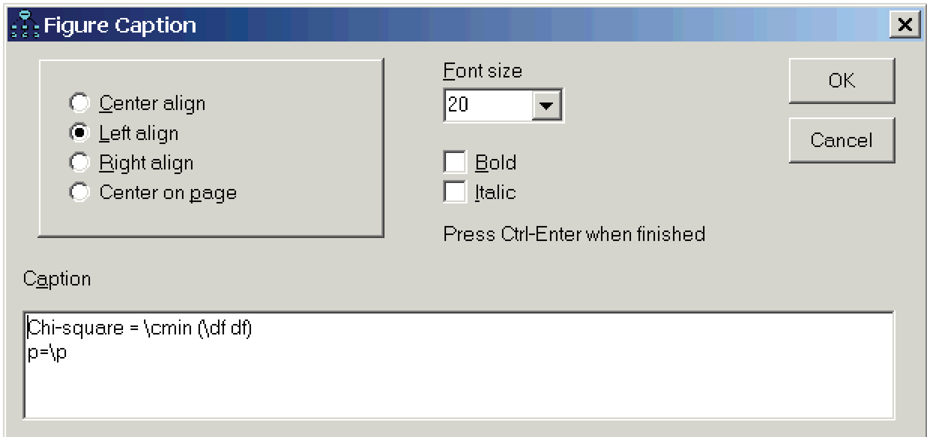
Displaying Chi-Square Statistics on the Path Diagram

You can get the chi-square statistic and its degrees of freedom to appear in a figure caption on the path diagram using the text macros \cmin and \df. Amos replaces these text macros with the numeric values of the chi-square statistic and its degrees of freedom. You can use the text macro \p to display the corresponding right-tail probability under the chi-square distribution.

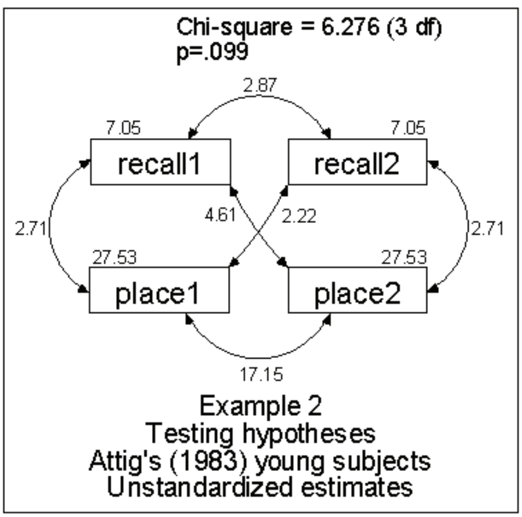
- From the menus, choose Diagram → Figure Caption.
- Click the location on the path diagram where you want the figure caption to appear.

The Figure Caption dialog box appears.

▶ In the Figure Caption dialog box, enter a caption that includes the \cmin, \df, and \p text macros, as follows:



When Amos displays the path diagram containing this caption, it appears as follows:



Modeling in VB.NET

The following program fits the constrained model of Example 2:

```
Module MainModule
    Sub Main()
        Dim Sem As New AmosEngine
        Try
            Sem.TextOutput()
            Sem.Standardized()
            Sem.ImpliedMoments()
            Sem.SampleMoments()
            Sem.ResidualMoments()

            Sem.BeginGroup(Sem.AmosDir & "Examples\Attg_yng.sav")
            Sem.AStructure("recall1 (v_recall)")
            Sem.AStructure("recall2 (v_recall)")
            Sem.AStructure("place1 (v_place)")
            Sem.AStructure("place2 (v_place)")
            Sem.AStructure("recall1 <> place1 (cov_rp)")
            Sem.AStructure("recall2 <> place2 (cov_rp)")
            Sem.FitModel()
        Finally
            Sem.Dispose()
        End Try
    End Sub
End Module
```

This table gives a line-by-line explanation of the program:

Program Statement	**Explanation**
Dim Sem As New AmosEngine	Declares Sem as an object of type AmosEngine. The methods and properties of the Sem object are used to specify and fit the model.
Sem.TextOutput	Creates an output file containing the results of the analysis. At the end of the analysis, the contents of the output file are displayed in a separate window.
Sem.Standardized() Sem.ImpliedMoments() Sem.SampleMoments() Sem.ResidualMoments()	Displays standardized estimates, implied covariances, sample covariances, and residual covariances.
Sem.BeginGroup …	Begins the model specification for a single group (that is, a single population). This line also specifies that the SPSS Statistics file *Attg_yng.sav* contains the input data. Sem.AmosDir() is the location of the Amos program directory.
Sem.AStructure("recall1 (v_recall)") Sem.AStructure("recall2 (v_recall)") Sem.AStructure("place1 (v_place)") Sem.AStructure("place2 (v_place)") Sem.AStructure("recall1 <> place1 (cov_rp)") Sem.AStructure("recall2 <> place2 (cov_rp)")	Specifies the model. The first four AStructure statements constrain the variances of the observed variables through the use of parameter names in parentheses. *Recall1* and *recall2* are required to have the same variance because both variances are labeled *v_recall*. The variances of *place1* and *place2* are similarly constrained to be equal. Each of the last two AStructure lines represents a covariance. The two covariances are both named *cov_rp*. Consequently, those covariances are constrained to be equal.
Sem.FitModel()	Fits the model.
Sem.Dispose()	Releases resources used by the Sem object. It is particularly important for your program to use an AmosEngine object's Dispose method before creating another AmosEngine object. A process is allowed to have only one instance of an AmosEngine object at a time.
Try/Finally/End Try	This Try block guarantees that the Dispose method will be called even if an error occurs during program execution.

▶ To perform the analysis, from the menus, choose File → Run.

Timing Is Everything

The AStructure lines must appear *after* BeginGroup; otherwise, Amos will not recognize that the variables named in the AStructure lines are observed variables in the *attg_yng.sav* dataset.

In general, the order of statements matters in an Amos program. In organizing an Amos program, AmosEngine methods can be divided into three general groups[1].

Group 1 — Declarative Methods

This group contains methods that tell Amos what results to compute and display. TextOutput is a Group 1 method, as are Standardized, ImpliedMoments, SampleMoments, and ResidualMoments. Many other Group 1 methods that are not used in this example are documented in the *Amos 17.0 Programming Reference Guide*.

Group 2 — Data and Model Specification Methods

This group consists of data description and model specification commands. BeginGroup and AStructure are Group 2 methods. Others are documented in the *Amos 17.0 Programming Reference Guide*.

Group 3 — Methods for Retrieving Results

These are commands to…well, retrieve results. So far, we have not used any Group 3 methods. Examples using Group 3 methods are given in the *Amos 17.0 Programming Reference Guide*.

Tip: When you write an Amos program, it is important to pay close attention to the order in which you call the Amos engine methods. The rule is that groups must appear in order: Group 1, then Group 2, and finally Group 3.

For more detailed information about timing rules and a complete listing of methods and their group membership, see the *Amos 17.0 Programming Reference Guide.*

1 There is also a fourth *special* group, consisting of only the Initialize Method. If the optional Initialize Method is used, it must come *before* the Group 1 methods.