The purpose of this tutorial is to outline the application of a group of two-stage Markov models that have been used to quantify recollective and nonrecollective retrieval processes (Brainerd, Aydin, & Reyna, 2012; Brainerd & Reyna, 2010; Brainerd, Reyna, & Howe, 2009; Gomes, Brainerd, & Stein, 2013). The tutorial provides a step-by-step guide on how to compute the relevant statistics to obtain parameter estimates and goodness-of-fit statistics. Because the models measure retrieval operations that can be broadly separated into recollective (direct access, *D*) and nonrecollective ones (reconstruction, *R*, and familiarity judgment, *J*), we also refer to them as dual-retrieval models. For specific information about the models and the theory underlying them, please see Brainerd et al. (2009).

# Section I: Dual Retrieval Models

In addition to the original dual-retrieval model with 11 free parameters (Brainerd & Reyna, 2010; Brainerd et al., 2009), we have developed four other reduced models with 6 free parameters. Although the five models measure both recollective and nonrecollective processes, the original model can only fit data from experiments with at least 4 trials ( $T_1$ , $T_2$ , $T_3$ , $T_4$ ), whereas the reduced models can fit data from experiments with only 3 fixed trials ( $T_1$ , $T_2$ , $T_3$ ). However, the procedures described here apply to all five models. Below is a short description of each model, separated by study (S) and test (T) design, and their respective starting vector (**W**) and transition matrix (**M**):

For a S<sub>1</sub>T<sub>1</sub>T<sub>2</sub>, S<sub>2</sub>T<sub>3</sub>, S<sub>3</sub>T<sub>4</sub> (noncanonical) design: The original dual-retrieval model with 11 free parameters can be applied to such data. The model has 4 direct access parameters (*D*<sub>1</sub>, *D*<sub>2</sub>, *D*<sub>3</sub>, *D*<sub>3</sub>), 2 reconstruction parameters (*R*<sub>1</sub>, *R*<sub>2</sub>), 4 familiarity judgment parameters (*J*<sub>1</sub>, *J*<sub>2</sub>, *J*<sub>3</sub>, *J*<sub>3</sub>), and 1 forgetting parameter (*f*).

Original	model:
<u>Ongina</u>	mouch

		L(2)	Pe(2)	Pc(2)	U(2)
	L(1)	1	0	0	0
M4 =	RE(1)	0	(1-f)(1-J3E)	(1-f)J3E	f
<b>m</b> 1 -	Rc(1)	0	(1-f)(1-J3c)	(1-f)J3c	f
	<u>U(</u> 1)	0	0	0	1
		<u> </u>	P <sub>E</sub> (n+2)	Pc(n+2)	U(n+2)
	L(n+1)	1	0	0	0
M2 =	PE(n+1)	Dзе	(1-Дзе)(1-Јзе)	(1-Дзе)Јзе	0
	<u>Pc(n+1)</u>	Dзс	(1-D3c)(1-J3c)	(1-Дзс)Јзс	0
	U(n+1)	D2	(1-D2)R2(1-J2)	(1-D2)R2J2	(1-D2)(1-R2)

For a S<sub>1</sub>T<sub>1</sub>, S<sub>2</sub>T<sub>2</sub>, S<sub>3</sub>T<sub>3</sub> (canonical) design: Four dual-retrieval models with 6 free parameters can be applied to such data. We refer to such models as the learning on errors model ("error model" for short), learning on successes (success model), and learning on both errors and successes model (both model). Such reduced models have 2 direct access parameters  $(D_1, D_2)$ , 2 reconstruction parameters ( $R_1$ ,  $R_2$ ), and 2 familiarity parameters ( $J_1$ ,  $J_2$ ). The difference between them lies on the State L late learning entry parameter D<sub>2</sub>, namely whether entry in State L after trial 1 depends on prior incorrect recall (error model), prior correct recall (success model), or does not depend on prior correct or incorrect recall (both model). Finally, the fourth reduced model is an alternative version of the error model in which there is 1 reconstruction parameter

(R) and 3 familiarity judgment parameters  $(J_1, J_2, J_3)$  in addition to the 2 direct access parameters.

#### Error model:

U(n+1) L(n+1) PE(n+1)Pc(n+1) 1 0 0 0 L(n) PE(n) D2  $(1-D_2)(1-J_2)$ (1-D2)J2 0  $\mathbf{M} = \mathbf{Pc}(\mathbf{n})$ 0  $(1-J_2)$ J2 0 U(n)  $(1-D_2)R_2(1-J_2)$ (1-D2)R2J2 (1-D2)(1-R2) D2

 $\mathbf{W} = [\underline{L}(1), Pe(1), Pc(1), U(1)] = [D_1, (1-D_1)R_1(1-J_1), (1-D_1)R_1J_1, (1-D_1)(1-R_1)]$ 

Alternative Error model:

 $W = [L(1), Pe(1), Pc(1), U(1)] = [D_1, (1-D_1)R(1-J_1), (1-D_1)RJ_1, (1-D_1)(1-R)]$ 

	L(2)	Pe(2)	Pc(2)	U(2)
L(1)	1	0	0	0
$\frac{\text{Re}(1)}{M_1 = \text{Rc}(1)}$	D2	(1-D2)(1-J2)	(1-D2)J2	0
U(1)	0	(1-J2)	J2	0
	D2	(1-D2)R(1-J2)	(1-D2)RJ2	(1-D2)(1-R)
	L(3)	Pe(3)	Pc(3)	U(3)
L(2)	1	0	0	0
$\frac{\text{Re}(2)}{M_2 = \frac{\text{Re}(2)}{2}}$	D2	(1-D2)(1-J3)	(1-D2)J3	0
U(2)	0	(1- <i>J</i> 3)	Jз	0
	D2	(1-D2)R(1-J3)	(1-D2)RJ3	(1-D2)(1-R)

Success model:

W = [L(1), PE(1), Pc(1)]	), U(1)] = [D1, (1-D1)R1(1-	J1), (1-D1)R1J1, (1-D1)(1-R1)]
--------------------------	-----------------------------	--------------------------------

4	<u> </u>	PE(n+1)	Pc(n+1)	U(n+1)
L(n)	1	0	0	0
$\frac{P_{E}(n)}{M = P_{O}(n)}$	0	(1- <i>J</i> 2)	J2	0
W - 55(1)	D2	(1-D2)(1-J2)	(1-D2)J2	0
<u>Ų(</u> n)	0	R2(1-J2)	R2J2	(1- <i>R</i> 2)
12				10

Both model:

	L(n+1)	PE(n+1)	Pc(n+1)	U(n+1)
L(n)	1	0	0	0
$\frac{P_{E}(n)}{M = P_{C}(n)}$	D2	(1-D2)(1-J2)	(1-D2)J2	0
<u>Ų(n)</u>	D2	(1-D2)(1-J2)	(1-D2)J2	0
	D2	(1-D2)R2(1-J2)	(1-D2)R2J2	(1-D2)(1-R2)

 $W = [L(1), Pe(1), Pc(1), U(1)] = [D_1, (1-D_1)R_1(1-J_1), (1-D_1)R_1J_1, (1-D_1)(1-R_1)]$ 

# Section II: Example

This example used a database composed of 15 subjects that learned to recall a list of 40 words using the  $S_1T_1T_2$ ,  $S_2T_3$ ,  $S_3T_4$  noncanonical design mentioned in Section 1. <u>The database is available on the website</u> (Database I). The model applied to the data was the original dual retrieval model.

## 1. Structure of the database

The database must follow a specific format in order to compute the frequencies of correct (C) and incorrect (E) recall across trials using the data count program, namely

- The first row contains the names of variables, with the data range beginning in the second row.
- The first column contains the subject ID number.
- In the first row, each column after the first one will contain both the name of the to-be-recalled items and the trial number. Specifically, the variable names must have the following format: [name of the item][lowercase letter t][trial number]. For instance, column #2 can be called SPIDERt1, meaning the item SPIDER on trial 1. The program will look for this structure in order to compute items' history of recall across trials, so make sure that each item has the same name across trials (e.g., SPIDERt1, SPIDERt2, SPIDERt3, SPIDERt4). NOTE 1: The word list cannot contain different set of words on different trials. The program will not compute the frequency of C-E patterns across trials if this is the case. NOTE 2: Only words used in the recall test can have the above format. Thus, it is possible to have other variables (e.g., age), provided that these variables are not written in the format above.

- Use binary coding to assign correct recall (1) and incorrect recall (0).
- Ensure that there are no empty cells within the data range.
- The worksheet in which the database is stored must be called DATA.

_			-			
	118	-	(	f <sub>x</sub>		_
1	А	В	С	D	E	
1	ID	EXAMEt1	COBRAt1	BARATt1	ZANGAt1	СН
2	1	0	0	0	0	
3	2	0	1	0	0	
4	3	0	0	1	1	
5	4	0	0	0	0	
6	5	0	0	0	1	
7	6	0	0	0	0	
8	7	0	0	0	1	
9	8	1	1	0	1	
10	9	0	0	0	0	
11	10	0	1	0	0	
12	11	1	0	0	0	
13	12	0	0	0	0	
14	13	1	1	0	0	
15	14	1	0	0	0	
16	15	1	0	0	1	
17						
18						
19						

Here is a screen shot of the sample database:

## 2. Computing frequencies of C-E patterns across trials

The data count program (see Programs and Model Files) is an Excel VBA macro that reads the database and computes C-E frequencies. Because an item is either recalled (1 = C) or not (0 = E) on each test, there are a total of  $2^n$  patterns of C-E responses across trials, in which *n* is the total number of trials. In our example, the database is composed of 4 separate recall tests and, therefore, there are  $2^4 = 16$ patters of C-E responses (CCCC, CCCE, ..., EEEE). For instance, if a given word is recalled successfully across all 4 trials, that would be counted as CCCC = 1; if a given word is recalled successfully on the first 3 trials but unsuccessfully on the last trial, that would be counted as CCCE = 1. This is done for all items and subjects in the DATA worksheet of the data count Excel file, and the final frequencies are the ones aggregated across both.

- 1. Open the data count program and the database.
- 2. Transfer the responses from the database to the DATA worksheet in the data count program.
- Enable the use of macros in Excel (for security reasons, the default setting is not to allow their use). Go to Tools > Options > Security > Macro Security and then reduce the security level to Low.

- 4. Make sure that the structure is the same as noted earlier in Section 2.1.
- 5. Press Ctrl+U (or the button) to run the macro.
- 6. The results will be pasted on the RESULTS worksheet of the data count program.

For our database, the frequencies were the following:

CCCC: 167 CCCE: 7 CCEC: 9 CCEE: 2 CECC: 12 CECE: 4 CEEC: 5 CEEE: 3 ECCC: 24 ECCE: 2 ECEC: 1 ECEE: 1 EECC: 123 EECE: 12 EEEC: 103 EEEE: 125

In our database, there were 167 cases in which an item was recalled on all tests (CCCC), and 125 in which an item was never recalled (EEEE). Of course, the sum of all frequencies should be equal to the product between (sample size) and (number of items), which is 15 x 40 = 600 in this case, and the proportion of items recalled on the *i*th test is simply the sum of C-E patters in which there is a C on the *i*th test divided by the sum of all frequencies. For instance, the proportion of items recalled on the  $3^{rd}$  test was = (CCCC + CCCE + CECC + CECE + ECCC + ECCE + EECC + EECE) / [(sample size) x (number of items)] = .59.

#### 3. Obtaining parameter estimates and goodness-of-fit statistics using GPT

Once the frequencies of C-E patterns across trials have been computed, you are ready to enter them into the GPT.

- a) Open the GPT
- b) Go to File > Open > Models and load the original dual-retrieval model .pt2 file (see Programs and Model Files)

Window Help			
New	+		
Open	•	Models	
Save		Out put	
Save as	T		
Exit			
c:\users\ryan\desktop\lab\dual process tutorial\tutorial hypothesis testing.pt2			
C:\Users\Ryan\Desktop\Lab\Dual Process tutorial\Joint Model.pt2	- 1		
merged.pt2	- 1		

c) Once you have loaded the model, you can start entering the C-E frequencies: Right click on the model window > Model > Input data

General Processing Tree Models			
File Edit Window Option Help			
🔮 c:\users\ryan\desktop\lab\dual process tutorial\11 parameters - 4 fix	ed trials.pt2		
Save Model Version Clear Model History BrainerdReynaAgi	ingRecallModel		•
BrainerdReynaAgngRecallModel           □         D1           □         1-D1           □         R1           □         1-R1           □         1-D2           □         R2			
br n2 br 1-R2 i- D2	GPT Models Display Mode	•	
id	Model	+	Hypothesis test (set parameters equal)
H- H2 1-B2	Options	•	Input data
	Checkmark tree nodes Find and replace	•	Estimation, simulation, or power analysis Simplify tree

d) Enter the frequencies in their respective cell in the first column, as follows:

							Save Model Version	Clear Model His
	64			Data			BrainerdReynaAgingF	RecallModel
	Root	Name	Example	Empty	Empty	Empty I	⊡ 1-D1	
	BrainerdRey	10000	167	0	0	0 1	i ⊕ B1 ⊡ 1-B1	
	BrainerdRey	r CCCE	7	0	0	0 1	D2	
	BrainerdRey	r CCEC	9	0	0	0 1	⊡ 1·D2 ⊕ 82	
	BrainerdRey	r CCEE	2	0	0	0 1		
	BrainerdRey	r CECC	12	0	0	0 1	ll □ 1	-D2
	BrainerdRey	r CECE	4	0	0	0 1	È.	R2
	BrainerdRey	r CEEC	5	0	0	0 1		1-112
	BrainerdRey	r CEEE	3	0	0	0 1		
	BrainerdRey	r ECCC	24	0	0	0 1		
	BrainerdRey	FECCE	2	0	0	0 1		
	BrainerdRey	ECEC	1	0	0	0 1		
	BrainerdRey	ECEE	1	0	0	0 1		
	BrainerdRey	EECC	123	0	0	0 1		
	BrainerdRey	EECE	12	0	0	0 1		
	BrainerdRey	rEEEC	103	0	0	0 1		
	BrainerdRey	EEEE	125	0	0	0 1		
	•					۲		
				Apply				
*			(	Cancel				

- e) Add a name to the column by clicking on it with the right button, then rename.
- f) Click apply and save the file

You are now able to compute parameter estimates and goodness-of-fit statistics.

- a) Return to the model window
- b) Right click > Model > Estimation, simulation or power analysis

💭 General Processing Tree Models - [f:\a research a\dual-recall model zero-file	es\gpt files\11 parameters model - 4 fixed tri	als\11 parameters - 4 💶 🗙
Sile Edit Window Option Model Help		_ <u>8</u> ×
File Edit Add and Edit States Options Model Windows Help		
<u>NGQQASI</u>		
Save Model Version Clear Model History BrainerdReynaAgingRecallMode		•
BranerdReynsAgingRecallModel → D1 → 101 ⊕ R1 → 1-R1 → 1-R2 ⊕ R2 ⊕ 1-D2 ⊕ R2 ⊕ R2	GPT Models Display Mode Model Options Checkmark tree nodes Find and replace Simplify tree	parameters equal) 1, or power analysis
Show Property Panel 🔲 Show Model Memo 📄 Show Graph		
		1

- c) In the select models column, select BrainerdReynaAgingRecallModel, which is the default (unconstrained) model
- d) In the select data for estimation, select the first row or whatever row that contains the frequencies you have entered previously
- e) Then click Run
- f) Once the program stops running, the results will be presented in the **Output Tab** and will also be pasted on the **Output Window** (see Hu & Phillips, 1999, for more information). The maximum likelihood estimates of each parameter, their respective standard deviation (SD), and the goodness-of-fit statistic ( $\chi^2$ ) pasted on the output window should look like this:

#### Parameter estimates

D1	=	0.16064,	SD =	0.04945,	(initial value:	0.42594)
D2	=	0.11067,	SD =	0.10306,	(initial value:	0.34826)
D3C	=	0.27659,	SD =	0.15190,	(initial value:	0.84037)
D3E	=	0.00000,	SD =	0.00000,	(initial value:	0.24153)
F	=	0.04525,	SD =	0.05494,	(initial value:	0.09628)
J1	=	0.74471,	SD =	0.05506,	(initial value:	0.04693)

.

J2 = 0.64327, SD = 0.09721, (initial value: 0.06274) J3C = 0.82413, SD = 0.05476, (initial value: 0.80138) J3E = 0.75965, SD = 0.09422, (initial value: 0.20643) R1 = 0.30027, SD = 0.04601, (initial value: 0.01425) R2 = 0.43920, SD = 0.10311, (initial value: 0.06042)

Test will be a Chi-square with 4 degrees of freedom Chi-square [4] = 3.75306.

# Section III: Hypothesis Testing

Comparisons between parameter estimates can be performed via likelihood ratio tests (LRT) (e.g., Brainerd, Reyna, & Howe, 2009). LRT can be broadly separated into two types: (a) within- and (b) between-condition tests. The former type tests hypotheses regarding parameter differences within the same experimental condition, which can be either differences between parameters (e.g., HO:  $R_1 = R_2$  vs. H1:  $R_1 \neq R_2$ ) or differences between parameters and a constant (e.g., HO: f = 0 vs. H1:  $f \neq 0$ ). The latter type tests hypotheses that have to do with localizing treatment effects, that is, whether parameter estimates differ across any set of k > 1 conditions (e.g., HO:  $J_1$ , *condition*  $_2$  vs. H1:  $J_1$ , *condition*  $_1 \neq J_1$ , *condition*  $_2$ ). Between-condition tests can be further divided into experiment-wise tests, condition-wise tests, and parameter-wise tests. Because Type I error increases as a function of the number of tests, it is often advised to perform condition-wise and parameter-wise tests, or adjust the family-wise alpha level using a correction such as Bonferroni ( $\alpha_{adjusted} = \alpha / n$ , in which n is the total number of comparisons) or Sidak ( $\alpha_{adjusted} = 1 - (1 - \alpha)^{1/n}$ ).

In GPT, the test statistic in all cases is computed by subtracting the fit statistic of the unconstrained model (i.e., a model in which all parameters of the model are free) from the constrained model (a model in which one imposes the restrictions of interest, e.g., f = 0). The resulting difference is a statistic asymptotically distributed as  $\chi^2$ , with df equal to the difference in free parameters between the two models. The critical  $\chi^2$  value of the test can therefore be calculated and compared against the observed  $\chi^2$ .

Next, we describe how to perform LRT using the GPT program. We will be using the database provided on the lab website (Database II). Note that the database contains three worksheets, each containing data from a different experimental condition.

## A. Within-Conditions Test: Parameter = Constant

In this section, we will perform a within-condition hypothesis test on the data from Condition 1. The null hypothesis being tested is that  $D_1 = 0$ .

1. Compute the goodness of fit statistic for the unconstrained model by following the instructions in Section 2. It was equal to 7.83.



- 2. Next, we will compute the goodness of fit statistic for the constrained model.
- 3. Click on the show property panel in the model window, and the following screen should appear:

e Edit Window Option Model Help	
Edit   Add and Edit States   Options   Model   Windows Help	
Vueentingverlidesktopfilebildual process tutorial/11 parameters - 4 files de Property Property StateFrage   State Information Parameter BranerdReyn Name   Nature    Nature	d hinkupt2 = 1 = 1 = 101 ⊕ 101 ⊕ 101 ⊕ 101 ⊕ 101 ⊕ 101 ⊕ 101 ⊕ 101 ⊕ 102 ⊕ 182 = 182
Desception	
Show Property Panel Show Model Messo	

4. Select the parameter of interest. In this case, click on  $D_1$  in the model window.

# Created by Carlos Gomes (cf365@cornell.edu) and Ryan Yeh (ry58@cornell.edu) 11

	Last update: July 2 <sup>nd</sup> , 2013
General Processing Tree Models	
File Edit Window Option Model Help	
File Edit Add and Edit States Options Model Windows Help	
?	
Edit Pronertu	E- BrainerdBeynaAgingBegallModel
Desert	D1 □
Property	in 1-D1 in B1 − 1
Аррју	
	E F
StatePage	□ 02 □ 1·D2
State Information	B2 B2
Paranete Di	⊟
Nature Ending	⊡… 1-D3C
Nature johung	J3C
Cat Parameter as Constant	
Chate Estimation at Bandom	D3E
I♥ Statts Estimation at handoin	J3E
	1-J3E
Description	□-H2
Default	⊡ 1·D2
	⊡ H2
	1J2
	1-R2
🔽 Show Property Panel 🦳 Show Model Memo 🦳 Show Graph	

5. Click "Set Parameter as Constant", and specify the value. In our case, D1 = 0.001:

c:\users\ryan\deskt	top\lab\dual process tutorial\11 parameters - 4 fixed trials.pt2	
StatePage Consta	Property Apply ant Parameters	
	Setting Parameter as Constant           C D1         0.00100           Image: Total and the set of the	□       1-02         □       -         □       -         □       -         □       -         □       -         □       -         □       -         □       -         □       -         □       -         □       -         □       1-03C         □       -

- 6. Click Apply
- 7. Deselect the "show property panel" and return to the main window.

8. Click Save the Model Version, and name the model. In this case, we named the constrained model, D1 = 0.

Estimation Simulation	Power analysis				
Select Model and Data	Output				
Select models			Select data for esti	imation	
Buinesd Beyna Aging Re	01				
temprode constrained b D1 = 0	Zero frequency Zero frequency				
	While	Minimize application	After	Print all resul	
				Bun	
		Estimation stop when GOF	1E-10	Cancel	

9. Select the constrained model and compute the goodness of fit.

10. The goodness of fit statistic was equal to 13.85:

EEE,	 149.00000,	 149.12898	1 1 1 1	1 1 1 1	
Carameter es D1 = - D2 - D3C - D3E - F - J1 - J3C -	timates 0.00100, 0.00000, 0.00000, 0.00399, 0.87715, 0.90575, 0.84910,	Fixed as constan SD = 0.00000, SD = 0.02000, SD = 0.02252, SD = 0.02262, SD = 0.03626, SD = 0.02252, SD = 0.01818,	t (initial value: (initial value: (initial value: (initial value: (initial value: (initial value:	0.09633) 0.76151) 0.21281) 0.24408) 0.08967) 0.17769) 0.36607)	
J3E = R1 = R2 = Hissubic(S) the paramet Stopping Crit umber of ite	0.82589, 0.24220, 0.35979, a Chi-squar 13.84876 er(s) are r eria: Diffe rations:280	SD = 0.11347, SD = 0.02094, SD = 0.02295, e with 5 degrees estricted as cons rence of log-like	(initial value: (initial value: (initial value: of freedom tant lihood = 1E-10	0.69937) 0.51069) 0.02666)	

11. One can now perform the hypothesis test by comparing the two goodness of fit statistics:

Constrained = 13.85 Unconstrained = 7.83

12. Calculate the difference between the two statistics to obtain the test statistic:

**Test statistic =** 13.85 - 7.83 = 6.02

13. Compare the test statistic to the test's critical value. For instance, at a significance criterion of .05, the critical value of the test statistic is  $\chi^2(1) = 3.84$ . Because the test statistic is higher than the critical value, we reject the null hypothesis of the test.

## **B.** Within-Condition Test: Parameter<sub>i</sub> = Parameter<sub>j</sub>, for $i \neq j$

In this section, we will perform a within-condition hypothesis test on the data from Condition 1. The null hypothesis being tested is that  $D_1 = D_2$ .

- 1. Compute the goodness of fit statistic for the unconstrained model by following the instructions in Section 2. It was equal 7.83.
- 2. Next, we will compute the goodness of fit statistic for the constrained model.
- 3. Right Click ---> Model ---> Hypothesis Test
- 4. Deselect "Same Level"
- 5. Select D1 in the left hand column and then select D2 in the right hand column.

## Created by Carlos Gomes (<u>cf365@cornell.edu</u>) and Ryan Yeh (<u>ry58@cornell.edu</u>) **14**

				Last update:	July 2 <sup>nd</sup> , 201
General Processing Tree File Window Help File Edit Windows Hel ?	e Models - [c:\users\ryan\c	lesktop\l <mark>ab\dua</mark> l process	tutorial\11 parameters - 4 fix	(ed trials.pt2]	_ O X
Set Up Hypothesis	Г 	Same level	Parameters	BrainerdReynaAgingRecall D1 ⊡ 1-D1 ⊡ 1-D1	Model
2 (R1 1-R1) 3 (J1 1-R1) 4 (F, 1-F) 5 (D2 1-D2) 7 (R2 1-R2) 8 (J2 1-J2) 9 (D3C, 1-D3C) 14 (D3E, 1-D3E) 14 (D3E, 1-D3E) 19 (D2, 1-D2) 21 (R2 1-R2) 22 (J2, 1-J2) 22 (J32, 1-J3E) 19 (D3E, 1-D3E) 32 (J3E, 1-D3E) 32 (J3E, 1-J3E) 40 (D3E, 1-D3E) 37 (J3E, 1-J3E) 42 (J3E, 1-J3E) 43 (J3E, 1-J3E) 44 (J3E, 1-J3E) 45 (J3E, 1-J3E) 45 (J3E, 1-J3E) 45 (J3E, 1-J3E) 47 (J	теру D1 1-D1	> □2  _ 1.02  _	2 [P1, 1, P1]         3 U1, 1, 1, 1]         4 [F, 1+F]         5 (02, 102)         7 (R2, 1-R2)         8 (J2, 1, 1, 2)         9 (D3C, 1-D3C)         11 (J3C, 1-J3E)         14 (D3E, 1-D3E)         16 (J3E, 1-J3E)         19 (D3C, 1-D3C)         19 (D3C, 1-D3E)         19 (D3C, 1-D3C)         21 (R2, 1-R2)         22 (J2, 1-J3E)         19 (D3C, 1-D3C)         27 (D3C, 1-D3C)         29 (J3C, 1-D3C)         30 (D3C, 1-D3C)         30 (D3C, 1-D3C)         32 (J3C, 1-J3C)         37 (J3E, 1-J3E)         40 (D3E, 1-D3E)         42 (J3E, 1-J3E)         42 (J3E, 1-J3E)         42 (J3E, 1-J3E)	i - 1-R1	12
Show Property Panel	Show Model Memo 🔲	Show Graph 🔽 Rej	olace All Instances 🧮 Hide Hy	pothesis Panel	

- 6. Click Apply and Done
- 7. Click "Save Model Version", and name the constrained model appropriately. Here, we named it D1 = D2.
- 8. Compute the goodness of fit statistic for the constrained model.

Untitled - Gpt	
♥ ★	
Parameter estimates         D1 = 0.08526, SD = 0.03160, (initial value: D3C = 0.00000, SD = 0.00000, (initial value: D3E = 0.17912, SD = 0.54951, (initial value: F = 0.01114, SD = 0.03282, (initial value: J1 = 0.80017, SD = 0.07349, (initial value: J2 = 0.89860, SD = 0.03715, (initial value: J3C = 0.75871, SD = 0.05640, (initial value: R1 = 0.16924, SD = 0.17083, (initial value: R2 = 0.29759, SD = 0.03981, (initial value: Test will be a Chi-square with 5 degrees of freedom	0.79253) 0.27417) 0.65016) 0.34093) 0.68624) 0.78468) 0.60185) 0.72953) 0.48712) 0.39493)
Stopping Criteria: Difference of log-likelihood = 1E-10 number of iterations:2873	•

9. One can now perform the hypothesis test by comparing the two goodness of fit statistics.

Constrained= 7.83

Unconstrained: 10.77

10. Calculate the difference between the two chi-square statistics to obtain the test statistic.

Test statistic: 10.77 – 7.83 = 2.94

11. Compare the test statistic to the test's critical value. For instance, at a significance criterion of .05, the critical value of the test statistic is  $\chi^2(1) = 3.84$ . Because the test statistic is lower than the critical value, we do not reject the null hypothesis of the test.

## C. Between-Condition Tests: Experiment-wise test

Between-condition tests answer questions regarding treatment effects (e.g., was direct access higher for short lists relative to long lists?). Running between-condition tests consist of three steps: (a) running an experiment-wise test in which the null hypothesis holds that all parameters are *the same across all experimental conditions*; (b) running condition-wise tests if the null hypothesis of the experiment-wise test was rejected, in which the null hypotheses hold that all parameters are *the same between two conditions*; and (c) running parameter-wise tests if the null hypothesis of the relevant condition-wise test was rejected, in which the null hypotheses hold that *a parameter of the model is equal between two conditions* (e.g., HO:  $J_{1, Condition 1} = J_{1, Condition 2}$ ).

This section will begin with the instructions for an experiment-wise test. In our example, there are three experimental conditions, the experiment-wise test and condition-wise test are the same.

The unconstrained model in between-conditions tests is a joint conditions model. Follow the instructions below to create the joint model.

1. Enter the error-success frequencies for both conditions.

notv	Root	Name	Condition 1	Condition 2	Condition 3	Empty	Empty	Empty	Empty	Empty
	BrainerdReyr	0000	66	60	41	0	0	0	0	0
	BrainerdReyr	CCCE	8	9	10	0	0	0	0	0
	BrainerdReyr	CCEC	13	15	9	0	0	0	0	0
	BrainerdReyr	CCEE	1	8	2	0	0	0	0	0
	BrainerdReyr	CECC	5	5	3	0	0	0	0	0
	BrainerdReyr	CECE	1	2	5	0	0	0	0	0
	BrainerdReyr	CEEC	1	3	1	0	0	0	0	0
	BrainerdReyr	CEEE	1	5	1	0	0	0	0	0
	BrainerdReyr	ECCC	8	3	8	0	0	0	0	0
	BrainerdReyr	ECCE	1	1	4	0	0	0	0	0
	BrainerdReyr	ECEC	1	1	4	0	0	0	0	0
	BrainerdReyr	ECEE	1	2	3	0	0	0	0	0
	BrainerdReyr	EECC	87	71	68	0	0	0	0	0
	BrainerdReyr	EECE	26	18	20	0	0	0	0	0
	BrainerdReyr	EEEC	81	80	83	0	0	0	0	0
	BrainerdReyr	EEEE	149	167	188	0	0	0	0	0
					Арр	oly				

- 2. Now create a joint conditions model as follows:
  - a. Click Model ---> Join Models

General Processing Tree Models		
File Edit Window Ontion M	lodel Help	A REAL PROPERTY OF A REAL PROPER
File Edit Add and Edit States	Update	
🕒 🕐 p1=p2 🎢	Join Models	
c:\users\rvan\desktop\lab\dua	Duplicate Current Model	a.pt2
Save Model Version	Estimate Parameters Simulate models	uRecalModel
D1	Power Analysis	
i⊟ 1-D1 i∰ R1	Analyze RT data	
⊡ 1-R1   D2	Contingency table	
i - 1-D2 i - R2 i - 1-R2 i - 1-R2 i - 1-D2 i - R2 i - R2 i - 1-R2 i - R2 i - 1-R2		

- b. Under "suffix to the parameters", enter "C1", to denote data from Condition 1.
- c. Under "Data set to use", select "Condition 1"
- d. Click "Ok"

## Created by Carlos Gomes (cf365@cornell.edu) and Ryan Yeh (ry58@cornell.edu) 17

🛏 😾 K-F 06		Last update: July 2 <sup>nd</sup> , 2013
h:\research and lab\brainerd lab v	vebsite\tutorial v2\gpt\11 parameters - 4 fixed trials v2.pt2	
Save Model Version Clear M	fodel History model	<u> </u>
Ė 🤦	Create New Models	- <u>v</u>
	11 parameters - 4 fixed trials v2.pt2	
⊡ 1-B1 □ D2 ⊡ 1-D2	ir included Suffix to the parameters	-
Ė, R2	C Reverse selection Data set to use Condition 1	
□ J2 □ 1-D3 □ 1-D3 □ 1-J2 □ 1-J2 □ 1-J2 □ 1-D3 □ 1-D3 □ 1-R2 □ 1-R2 □ 1-D2 □ 1-D2 □ 1-D2 □ 1-D2 □ 1-D2 □ 1-D3 □ 1-D2 □ 1-D2	v BrainerdReynaAgingRecallModel	
	OK Cancel Show hin	*
Show Property Panel Show		
🕅 Unt 🗗 💷 🖾		
		100

e. You will now see two models in the main screen. One will have the name, "merge.pt2". This is the model you just created.

👱 merged.pt2						
Save Model Version Clear Model History model	•					
	• •					
Show Property Panel Show Model Memo Show Graph						
🔮 h:\research and lab\brainerd lab website\tutorial v2\gpt\11 parameters - 4 fixed trials v2.pt2						
Save Model Version Clear Model History model	•					
BrainerdReynaAgingRecallModel           ⊢         D1           ⊡         1-D1           ⊡         R1           ⊡         1-R1           □         D2           ⊡         1-D2           ⊡         R2	E					
<u> </u>						
Show Property Panel Show Model Memo Show Graph						
🗱 Unt 🗊 🔲 🔀						
	1.					

- f. Now, we have to add the dataset from condition 2. Click on the model you just created (merged.pt2)
- g. Model ---> Join Model
- h. Enter "C2" for Suffix for Parameters, and select "Condition 2" from "data set to use"

r Model History	181	
Create New Models		
merged.pt2 11 parameter	ers - 4 fixed trials v2.pt2	
✓ included	Suffix to the parameters	
C Reverse selection	Data set to use Con	dition 2
✓ BrainerdReyna4	AgingRecallModel	
OK Cancel	]	🗖 Show hints

- i. Click "Ok"
- 3. The new window will show all 3 models: the original model, and the two models you just created. The model that has both "Condition 1" and "Condition 2" is the joint model that we want. Select the joint model.

🔮 merged.pt2	
Save Model Version Clear Model History model	•
Image: model     Image: BrainerdReynaAgingRecallModel_C1       Image: BrainerdReynaAgingRecallModel_C2	
Show Property Panel 🔲 Show Model Memo 👘 Show Graph	
👱 merged.pt2	
Save Model Version Clear Model History model	-
⊡         ✓ BrainerdReynaAgingRecallModel_C1           □         □1_C1           □         1.01_C1           □         ℝ-	<b>^</b>
🔽 Show Property Panel 🔲 Show Model Memo 🛛 T Show Graph	
🔮 h:\research and lab\brainerd lab website\tutorial v2\gpt\11 parameters - 4 fixed trials v2.pt2 📃 🕻	
Save Model Version Clear Model History model	-
□-√ BrainerdReynaAgingRecallModel □- D1 □- 1-D1 <u>□</u> - R1	* •
Show Property Panel 🔲 Show Model Memo 👘 Show Graph	
🕅 Unt 🗊 🗆 🖾	
	//

4. To include the data from Condition 3, first close the window of all models but (a) the original model and (b) the last joint model you created, then repeat the procedure described above

## Created by Carlos Gomes (cf365@cornell.edu) and Ryan Yeh (ry58@cornell.edu) 19

Last update: July 2<sup>nd</sup>, 2013

💇 merged.pt2	
Save Model Version Clear Model History model	•
Image: Constraint of the synaphy o	
🔮 merged.pt2	
Save Model Version Clear Model History model	•
⊞- model     ⊞-✓ BrainerdReynaAgingRecallModel_C1       ⊞-     BrainerdReynaAgingRecallModel_C2	
F Show Property Panel F Show Model Memo F Show Graph	
h:\research and lab\brainerd lab website\tutorial v2\gpt\11 parameters - 4 fixed trials v2.pt2	
Save Model Version Clear Model History model	•
BrainerdReynaAgingRecallModel     D1     D1     D1     Utr [For dels, please see Data page.     Show Property Panel  Show Model Memo  Show Graph	Â

- 5. Select the model window of the Joint Model Unconstrained
- 6. Delete all but the first data column of the joint model and rename the first column
  - a. Go to Input Data
  - b. Delete all columns but the first one
  - c. Select the first column and rename it to reflect that the first column has the data from all conditions (e.g., Conditions 1 2 3)
  - d. Click Apply
- 7. Compute the goodness of fit for the unconstrained joint model you just created.
  - a. Go to the Estimation, Simulation, and Power analysis window



General Processing Tree Models - [merged.pt2]	# 1/ Dawn	-		×
File Edit Window Option Model Help				
File Edit Add and Edit States Options Model Windows Help				
di <mark>de 9 8 8 1</mark>				
Save Model Version Clear Model History model				•
<ul> <li></li></ul>	GPT Models			
	Display Mode		II II · · · · · · · · · · · · · · · · ·	٦.
	Model		Hypothesis test (set parameters equal)	
	Checkmark tree nodes		Estimation cimulation or neuror analysis	
	Find and replace		Simplify tree	-1
Show Property Panel 🦳 Show Model Memo 🦳 Show Graph				
				//

b. Select the model called Joint Model Unconstrained and the joint dataset (always the first one)

Window Option Mo	del Help	s s			
Estimation, simulation	on, and Power A	nalysis			
Estimation Simulation	Power analysis				
Select Model and Data	Output				
Select models		9	elect data for estim	ation	
model	Conditions 1 2 3				
tempmodel	Zero frequency Zero frequency				E
	While	<ul> <li>Minimize application</li> <li>Minimize model window</li> </ul>	After	Print all resul	
		Estimation stop when GOF Maximum number of	1E-10 5000	Run Cancel	
			1		

c. Click on Run



8. Note the goodness of fit of the unconstrained joint model.

Chi-square[12]= 25.59775.

- 9. We are now ready to create the constrained joint model for the experiment-wise test.
  - a. Go to the Hypothesis Test window



10. Set all the parameters equal to each other among all three conditions using the Apply button. D1C1 = D1C2 , ..., R2C2 = R2C3 (set all parameters equal among the three conditions)

## Created by Carlos Gomes (<u>cf365@cornell.edu</u>) and Ryan Yeh (<u>ry58@cornell.edu</u>) 22

	Last update: July 2 <sup>nd</sup> , 2013
💔 General Processing Tree Models - [merged.pt2]	
Sile Window Help	_ <i>B</i> ×
File         Window         Help           Set Upgotheria	
Show Property Panel   Show Model Memo   Show Graph   Heplace All Instances   Hide Hypothesis Panel	

- 11. Click Done when you finish
- 12. The model you just created will be called "tempmodel"
- 13. Obtain the parameter estimates for the constrained model.
  - a. Go to the Estimation, Simulation, and Power analysis window

General Processing Tree Models - [merged.pt2]					
壁 File Window Help					_ 8 ×
Save Model Version Clear Model History	tempmodel				
Save Model Version         Clear Model History           ⊕         101_C1           ⊕         11_C1           ⊕         181_C1           ⊨         0.2_C1           ⊕         142_C1           ⊕         10_C1           ⊕         12_C1           ⊕         12_C1           ⊕         12_C1					<u>*</u>
		GPT Models Display Mode Model Options Checkmark tre Find and repla	ee nodes	Hypothesis test (set parameters eq Input data Estimation, simulation, or power a Simplify tree	ual) =
Show Property Panel Show Model Memo	Show Graph	Hide Hypothesis Panel			

b. Select the model called tempmodel and the joint dataset (always the first one)

## Created by Carlos Gomes (cf365@cornell.edu) and Ryan Yeh (ry58@cornell.edu) 23



c. Click on Run

<u>Parameter est</u> D1_C1 = D2_C1 = D3C_C1 = D3E_C1 = J1_C1 = J2_C1 = J3C_C1 = J3C_C1 = R1_C1 = R1_C1 = R1_C1 = R2_C1 =	0.07047, SD = 0.04623, SD = 0.00000, SD = 0.00000, SD = 0.76159, SD = 0.7653, SD = 0.73653, SD = 0.65714, SD = 0.18820, SD = 0.3810 SD =	0.01660, (initial value: 0.04206, (initial value: 0.00000, (initial value: 0.00000, (initial value: 0.00000, (initial value: 0.03715, (initial value: 0.04004, (initial value: 0.03500, (initial value: 0.04424, (initial value: 0.01799, (initial value: 0.04552 (initial value:	0.10635) 0.27040) 0.66987) 0.75464) 0.16029) 0.73815) 0.01273) 0.87204) 0.55193) 0.89750) 0.09057)
Test vill be a Chi-square[34] Stopping Crite number of iter	Chi-square with = 62.28860. ria: Difference c ations:3406	34 degrees of freedom of log-likelihood = 1E-10	

14. One can now perform the hypothesis test by comparing the two goodness of fit statistics.

Constrained: Chi-square[34]= 62.28860 Unconstrained: Chi square[12]= 25.59775

15. Calculate the difference between the two chi-square statistics to obtain the test statistic.

**Test statistic:** 62.29 – 25.60 = 36.69

16. Compare the test statistic to the critical value. For instance, at a significance criterion of .05, the critical value of the test statistic is  $\chi^2(22) = 33.92$ . Because the test statistic is higher than the

critical value, we reject the null hypothesis of no difference among the three experimental conditions.

## D. Between-Condition Tests: Condition-wise test

There are three conditions in our example. In this section, we show how to test the null hypothesis of no difference between the parameters of any two conditions in an experiment (e.g., Parameters  $_{\text{condition 1}}$  = Parameters  $_{\text{condition 2}}$ ). The following example illustrates a condition-wise test between Conditions 1 and 3.

1. Create a joint model of Conditions 1 and 3 as described in the previous section.

2 General Processing Tree Models	
File Edit Window Option Model Help	
File Edit Add and Edit States Options Model Windows Help	
👱 merged.pt2	
Save Model Version Clear Model History model	•
Image: model     Image: Strainerd ReynaAgingRecalModel_C3       Image: Strainerd ReynaAgingRecalModel_C3	
Show Property Panel Show Model Memo Show Graph	
Se merged.pt2	
Save Model Version Clear Model History model	-
B Brainer#ReynAgingRecalModel_C1            -D1_C1           B-10_C1           B-11_C1           B-12_C1           B-12_C1           B-10_2_C1           B-10_2_C1           B-10_2_C1           B-10_2_C1           B-10_2_C1           B-10_2_C1           B-10_2_C1           B-10_2_C1           B-10_2_C1           B-10_2_C1	
Update state descriptions	
LI-R2_C1      Update state descriptions      Show Property Panel Show Model Memo Show Graph	į,
Image: Show Property Panel     Show Model Memo     Show Graph	

- 2. Select the model window of the joint model
- 3. Delete all but the first data column of the joint model and rename the first column a. Go

to Input Data

- b. Delete all columns but the first one
- c. Select the first column and rename it to reflect that the first column has the data from all conditions (e.g., Conditions 1 3)

- d. Click Apply
- 4. Compute the goodness of fit for the unconstrained joint model you just created.
  - a. Go to the Estimation, Simulation, and Power analysis window

General Processing Tree Models			
File Edit Window Option Model Help			
File Edit Add and Edit States Options Model Windows Help			
D C pip2 2			
Merged.pt2			
Save Model Version Clear Model History model			-
B- model     B- ✓ BranetdReynsAgnpRecalModel_C1     B- BranetdReynsAgnpRecalModel_C3			
	GPT Models		
	Display Mode		
	Model + Hypoth	esis test (set parameters equal)	
	Options   Input de	ata	
	Checkmark tree nodes	on, simulation, or power analysis	
	Find and replace	tree	
Show Property Panel Show Model Memo Show Graph			
🕅 Unt 🗊 🗉 🔀 🕎 h:\r 🗊 💷 🄀			

b. Select the model called Joint Model Unconstrained and the joint dataset (always the first one)

General Processing Tree Models           File         Edit         Window         Option         Model         Help           File         Edit         Add and Edit States         Options         Model         Window           D         D         p=p:1         BE         BE <th>ws   Help  </th> <th></th> <th></th>	ws   Help		
merged.pt2	Estimation simulation and	Power Analysis	
Save Model Version     Clear Model History     model     tempmode	Estimation Simulation Power Select Model and Data Output	analysis	^
	Select models	• Select data for estimation	
E 1A1_C3 D2_C3 E 102_C3 E 72_C3 E 72_C3 E 102_C3 E 72_C3 E 102_C3 E	tempmodel Zero Zero Zero Zero Zero Zero Zero Zero	equency equenc	
1 Snow Property Panel   Show Model Memo		Estimation stop when GOF 1E-10 Run Maximum number of 5000 Cancel	
🕱 Unt 🕫 🗆 🕱 💁 h:\r 🖨 🖬	· · · · · · · · · · · · · · · · · · ·		

c. Click on Run

General Processing Tree Models	
9 File Window Help	- 6 ×
File Edit Windows Help	
?	
👷 merged.pt2	
Untitled - Gpt	
★ 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
	×
Parameter estimates	
$ \begin{array}{rcrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrr$	
waber of iterations:5000	º

5. Note the goodness of fit of the unconstrained joint model.

Chi-square[8]= 16.19670.

6. We are now ready to create the constrained joint model for the condition-wise test.

👰 General Processing Tree Models		
File Edit Window Option Model Help		
File Edit Add and Edit States Options Model Windows Help		
2		
I merged.pt2		
Save Model Version Clear Model History		-
■         ■	GPT Models         Display Mode         Model         Model         Imput data         Checkmark tree nodes         Find and replace	eters equal) power analysis
Show Property Panel C Show Model Memo C Show Graph		
Wunt (De a ) 💆 k.v (De a a a		

a. Go to the Hypothesis Test window

7. Set all the parameters equal to each other between the two conditions using the Apply button (i.e., D1C1 = D1C3 , ..., R2C1 = R2C3)

## Created by Carlos Gomes (<u>cf365@cornell.edu</u>) and Ryan Yeh (<u>ry58@cornell.edu</u>) 27

General Processing Tree Models     File Edit Window Option Model Help     File Edit Add and Edit States Options   Model   Windows Help     ?	
Set Up Hypothesis	model     temponod     tem
Direct constant     Apply     Done     00[01 C1, 101 C1]       210 C1, 101 C1]     Image: C1, 102 C1]     Image: C1, 102 C1]       110 C2, C1, 102 C1]     Image: C1, 102 C1]       110 C2, C1, 102 C1]     Image: C1, 102 C1]       110 C2, C1, 102 C1]     Image: C1, 102 C1]       110 C2, C1, 102 C1]     Image: C1, 102 C1]       110 C2, C1, 102 C1]     Image: C1, 102 C1]       121 C2, C1, 102 C1]     Image: C1, 102 C1]       121 C2, C1, 102 C1]     Image: C1, 102 C1]       121 C2, C1, 102 C1]     Image: C1, 102 C1]       121 C2, C1, 102 C1]     Image: C1, 102 C1]       121 C2, C1, 102 C1]     Image: C1, 102 C1]       121 C2, C1, 102 C1]     Image: C1, 102 C1]       121 C2, C1, 103 C1]     Image: C1, 103 C1]       121 C2, C1, 103 C1]     Image: C1, 103 C1]       121 C2, C1, 103 C1]     Image: C1, 103 C1]       121 C2, C1, 103 C1]     Image: C1, 103 C1]       121 C2, C1, 103 C1]     Image: C1, 103 C1]       121 C2, C1, 103 C1]     Image: C1, 103 C1]       121 C2, C1, 103 C1]     Image: C1, 103 C1]       121 C2, C1, 103 C1]     Image: C1, 103 C1]       121 C2, C1, 103 C1]     Image: C1, 103 C1]       121 C2, C1, 103 C1]     Image: C1, 103 C1]       121 C2, C1, 103 C1]     Image: C1, 103 C1]       121 C2, C1, 103 C1]     Image: C1, 103	Image: Hol_CL       Image: Hol_C
I Srow Friquety Farter I Srow House Method I Show Bridgh M Heptade All Instances   Hide Hypo	ress rame
🖬 Untur @ O X ) 🔮 httra @ O X )	

- 8. Click Done when you finish
- 9. The model you created will be called "tempmodel"
- 10. Obtain the parameter estimates for the constrained model.
  - a. Go to the Estimation, Simulation, and Power analysis window

General Processing Tree Models		
File Edit Window Option Model Help		
File Edit Add and Edit States Options Model Windows Help		
2		
9 merged.pt2		
Save Model Version Clear Model History tempmode		
Image: Second	GPT Models Display Mode Model Options Checkmark tree nodes Find and replace	Hypothesis test (set parameters equal)  Input data  Estimation, simulation, or power analysis  Simplify tree
Show Property Panel   Show Model Metho   Show Graph	j nue nypuriesis nanei	
🛱 Unt @0 12) 🖤 k/r @0 12		

b. Select the model called Joint Model Constrained Omnibus and the joint dataset (always the first one)



Click on Run

General Processing Tree Models	
👰 File Window Help	- 5 ×
File Edit Windows Help	
?	
merged.pt2	
Untitled - Gpt	
Output Table	
CEEC_C3.         3.00000,         2.37556           CEEE_C3.         1.00000,         0.71819           ECCC_C3.         8.00000,         2.2294           ECCE_C3.         4.00000,         2.8310           ECCE_C3.         4.00000,         2.93006           ECEC_C3.         4.00000,         7.93010           ECEC_C3.         6.00000,         7.93010           ECEC_C3.         6.00000,         7.93010           ECEC_C3.         8.00000,         7.93000           EEEE_C3.         2.00000,         2.00000           EEEE_C3.         1.88.00000,         168.50000	
Parameter estimates	
$\begin{array}{llllllllllllllllllllllllllllllllllll$	
Stopping Criteria: Difference of log-likelihood = 1E-10 number of iterations:3693	
	li.

11. One can now perform the hypothesis test by comparing the two goodness of fit statistics.

Constrained: Chi-square[19]= 35.90648 Unconstrained: Chi-square[8]= 16.19670

12. Calculate the difference between the two chi-square statistics to obtain the test statistic.

**Test statistic:** 35.91 – 16.20 = 19.71

13. Compare the test statistic to the critical value. For instance, at a significance criterion of .05, the critical value of the test statistic is  $\chi^2(11) = 19.68$ . Because the test statistic is higher than the critical value, we reject the null hypothesis of no difference between Conditions 1 and 3.

## E. Between-Condition Tests: Parameter-wise test

In this section, we will perform a parameter-wise test using the data from Conditions 1 and 3, namely the null hypothesis that  $R_{2}$ , condition  $1 = R_{2}$ , condition 3.

- 1. Construct an unconstrained joint model between the two conditions.
- 2. Record the goodness of fit of the unconstrained model.

Chi-square[8]= 16.19670

3. We are now ready to construct the constrained model.

Right Click ---> Model ---> Hypothesis Test

4. Set equal the parameter "R2\_C1" to the parameter "R2\_C3"

General Processing Tree Models	
File Edit Window Option Model Help	
File Edit Add and Edit States Options Model Windows Help	
2	
Image: product and	Image: State of the synaphy state of the
will be replaced by P2 C1 1-P2 C1	
	< <u> </u>
🗆 Show Property Panel 🗖 Show Model Memo 🦳 Show Graph 🔽 Replace All Instances 🗔 Hide Hy	pothesis Panel
🕅 Unt 📴 😐 🔀 h:\r 📴 😐 🔀	

5. Click Apply, and then Click Done

- 6. As before, the model you created will be called "tempmodel"
- 7. Compute the goodness of fit for the constrained Joint Model just created. Remember to click on the appropriate model when running the parameter estimates.

General Processing Tree	Models								o x
File Edit Window Opt	tion Model Help								
File Edit Add and Edit St	tates Options Model Window	vs Help							
?									
👰 merged.pt2	,							-	
Save Model Version	Clear Model History	Estimation, simulati	on, and Power A	Analysis					•
B- model	Estimation Simulation Power analysis								*
- 🔮 tempmodi		Select Model and Data	Output						
		Select models			Select data for e	stimation			
		model	Conditions 1 3				A		
		tempmodel	Zero frequency Zero frequency						
	10100		Zero frequency Zero frequency						
	- D2_C3		Zero frequency						
			Zero frequency						
	⊟- H2_UI		Zero frequency						
			Zero frequency Zero frequency						
			Zero frequency Zero frequency				E		
	here		Zero frequency Zero frequency						
	⊡ 1J2		Zero frequency						
	<b>—</b>		Zero frequency						
			Zero frequency Zero frequency						
			Zero frequency Zero frequency						
	D2_		Zero frequency Zero frequency						
			Zero frequency						_
			Zero frequency				-		
<	0		\u/bile	Minimize application	After	Print all record		detals, please see Data page.	//. <del>*</del>
Show Property Panel	Show Model Memo		WIND	Minimize model windo	W	)+ Thirk dil toodi			
				Estimation stop when G	DF 1E-10	Run			
				Maximum number	of 5000	Cancel			
	~								
🖬 Unt 📾 😐 🔀	🖳 🖳 h:\r 🝙 💿	~							

8. Record the goodness of fit statistic for the joint constrained model.

General Processing Tree Models	- @ ×
lile Window Help	- 6
e Edit   Windows Hep	
?	
erged.pt2	
Untitled - Gpt	
Dutput   Table	
	 - 1
D1_C3 = 0.05915; D3 = 0.01991; (initial value: 0.96085)         D2_C1 = 0.05915; D5 = 0.07970; (initial value: 0.59754)         D2_C3 = 0.05924; D5 = 0.05935; (initial value: 0.59754)         D3C_C3 = 0.0000; D5 = 0.00000; (initial value: 0.99080)         D3E_C1 = 0.14882; D5 = 0.00000; (initial value: 0.91755)         D3E_C1 = 0.01431; D5 = 0.00000; (initial value: 0.9115)         F_C1 = 0.01431; D5 = 0.00000; (initial value: 0.7135)         J_C3 = 0.00000; D5 = 0.00000; (initial value: 0.7135)         J_C3 = 0.02000; D5 = 0.00000; (initial value: 0.7135)         J_C3 = 0.02000; D5 = 0.00000; (initial value: 0.7135)         J_C3 = 0.02000; D5 = 0.00000; (initial value: 0.9145)         J_C3 = 0.02314; D5 = 0.00000; (initial value: 0.81922)         J_C3 = 0.02314; D5 = 0.00000; (initial value: 0.81922)         J_ZC3 = 0.02314; D5 = 0.00000; (initial value: 0.81922)         J_ZC3 = 0.02326; D5 = 0.01972; (initial value: 0.9634)         J_ZC4 = 0.7455; D5 = 0.049780; (initial value: 0.99634)         J_Z3C1 = 0.82128; D5 = 0.09780; (initial value: 0.99655)         R_LC1 = 0.17455; D5 = 0.03686; (initial value: 0.97650)         R_LC1 = 0.17455; D5 = 0.03780; (initial value: 0.37650)         R_LC1 = 0.17455; D5 = 0.03255; (initial value: 0.37650)         R_LC1 = 0.17455; D5 = 0.03255; (initial value: 0.37650)         R_LC1 = 0.17455; D5 = 0.03255; (initial value: 0.37650)         R_LC1 = 0.17455; D5 = 0.03255; (initial value:	P

12. One can now perform the hypothesis test by comparing the two goodness of fit statistics.

Constrained = 17.19960

Unconstrained = <u>16, 19670</u>

13. Calculate the difference between the two chi-square statistics to obtain the test statistic.

Test statistic = 17.2 - 16.2 = 1

14. Compare the test statistic to the critical value. For instance, at a significance criterion of .05, the critical value of the test statistic is  $\chi^2(1) = 3.84$ . Because the test statistic is lower than the critical value, we do not reject the null hypothesis of the test.

# References

- Brainerd, C. J., Aydin, C., & Reyna, V. F. (2012). Development of dual-retrieval processes in recall: Learning, forgetting, and reminiscence. *Journal of Memory and Language, 66*, 763788. doi:10.1016/j.jml.2011.12.002
- Brainerd, C. J., & Reyna, V. F. (2010). Recollective and nonrecollective recall. *Journal of Memory and Language, 63,* 425–445. doi:10.1016/j.jml.2010.05.002
- Brainerd, C. J., Reyna, V. F., & Howe, M. L. (2009). Trichotomous processes in early memory development, aging, and cognitive impairment: A unified theory. *Psychological Review*, 116, 783–832. doi:10.1037/a0016963
- Gomes, C. F. A., Brainerd, C. J., & Stein, L. M. (2013). Effects of emotional valence and arousal on recollective and nonrecollective recall. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 39*, 663-677. doi:10.1037/a0028578
- Hu, X., & Phillips, G. A. (1999). GPT.EXE: A powerful tool for the visualization and analysis of general processing tree models. *Behavior Research Methods, Instruments, & Computers, 31*, 220-234. doi:10.3758/BF03207714