Advanced SEM

in MPlus and R (lavaan)

Advanced SEM

My goal:

Teach how to do advanced SEM in MPlus and R (lavaan)

My approach:

- Basic CFA example
- Example of SEM, for a user experiment (with manipulations and behavioral outcomes)
- Advanced topics (if we get to them): Multi-level SEM, interaction effects in SEM, and cluster analysis

Slides

Feel free to share these slides with anyone

This is an "advanced" slide deck; for the "basics", visit www.usabart.nl/QRMS

If you want to use these slides in your own lectures, use the above link for attribution

Why MPlus / R?

MPlus and R have advanced SEM capabilities:

- Able to handle non-normal variables
- Able to handle repeated measures (lavaan: either or)
- Able to handle interactions (some with a trick)
- Find total effects, look at mod-indices, etc.
- MPlus has great support and course videos

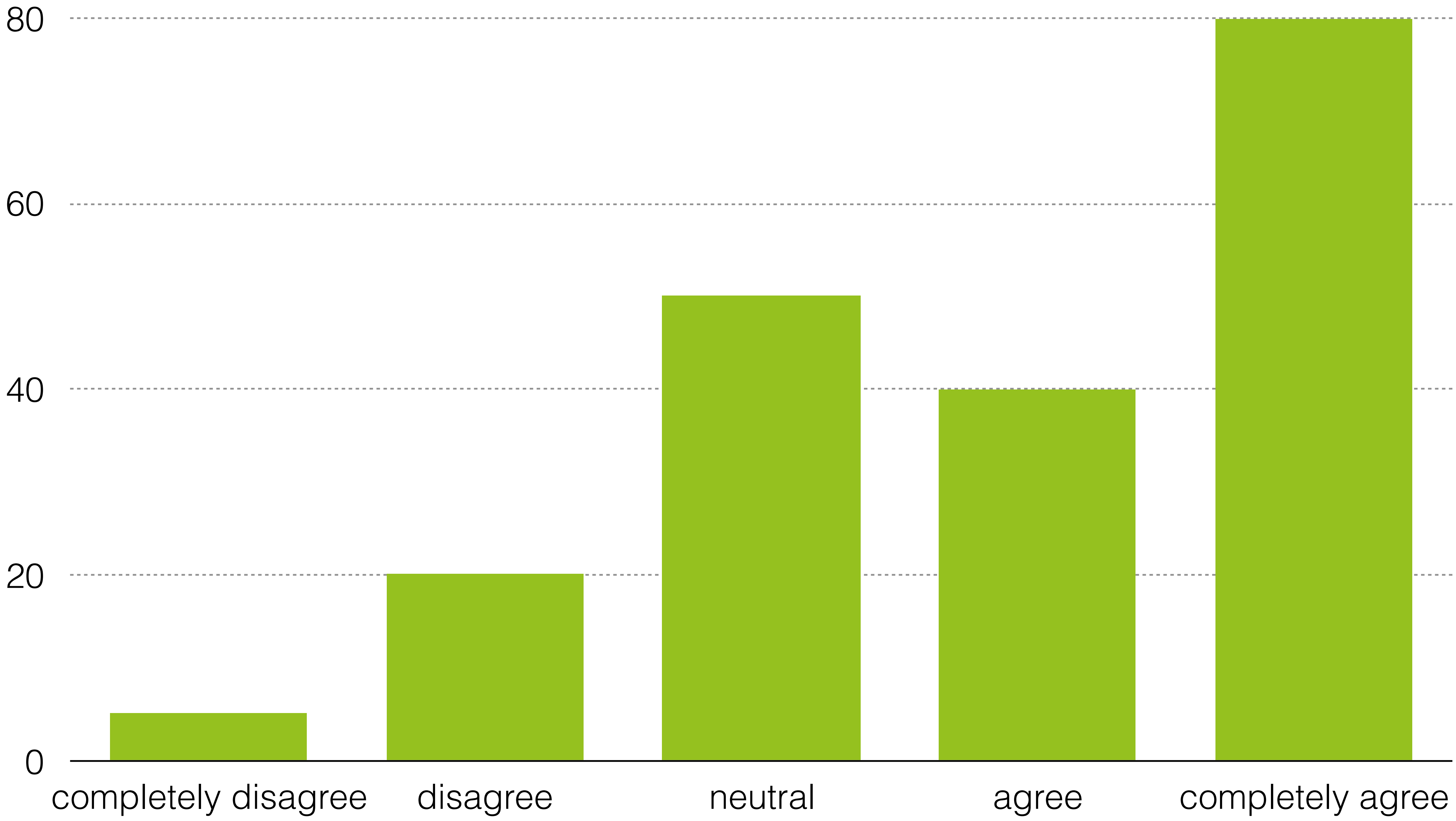
CFA

Confirmatory Factor Analysis in R and MPlus

CFA

- I assume that you have done this stuff before
    - I will show you how to do it in MPlus and R (lavaan)
- Benefit: Model 5- or 7-point scales as ordered categorical variables, rather than unbounded normally distributed variables
    - Does **not** assume that the difference between "completely disagree" and "disagree" is the same as between "neutral" and "agree"
    - Allows for "skewed" items

CFA


80
60
40
20
0
completely disagree
disagree
neutral
agree
completely agree

Example

Knijnenburg et al. (2012): "Inspectability and Control in Social Recommenders", *RecSys'12*

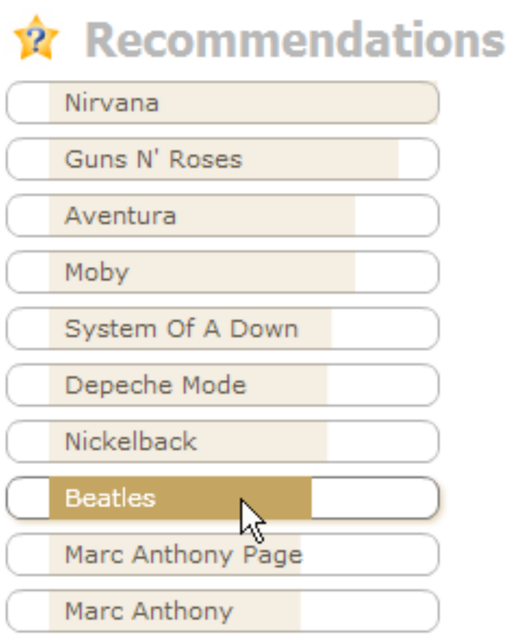
3 control conditions:

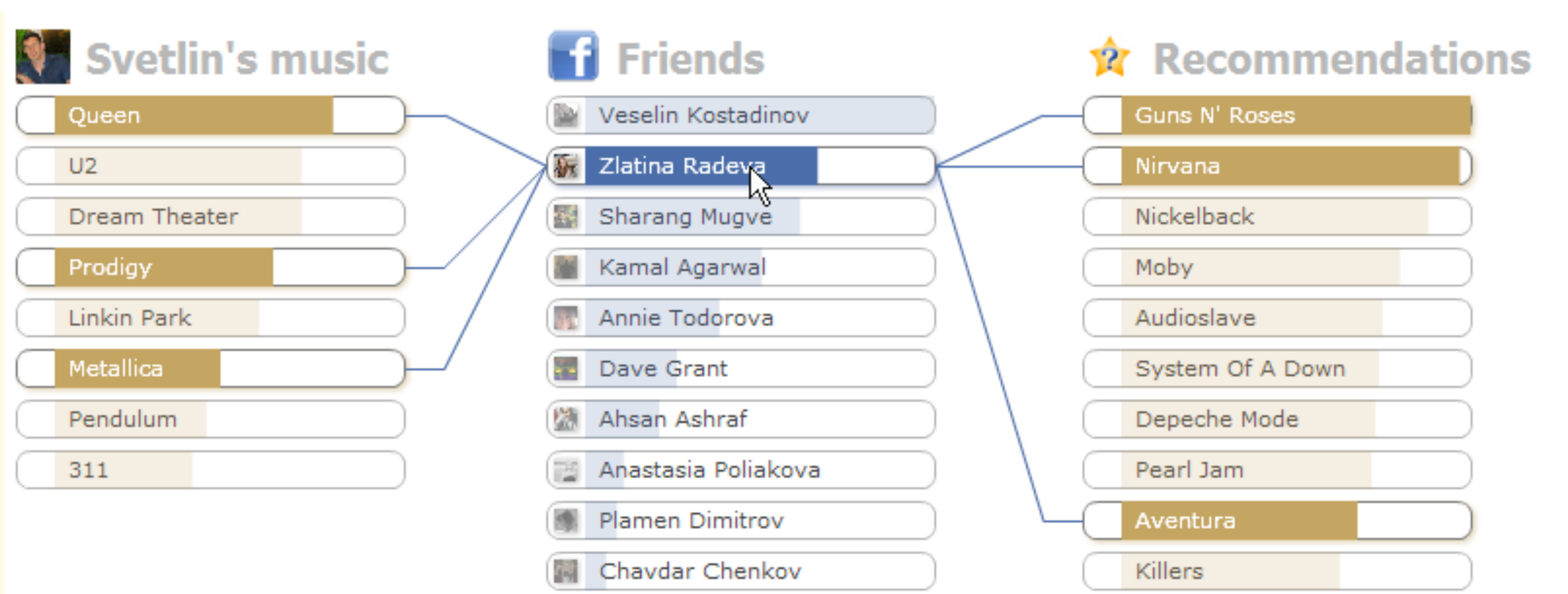
- No control (just use likes)
- Item control (weigh likes)
- Friend control (weigh friends)

Example

2 inspectability conditions:

- List of recommendations vs. recommendation graph





Example

Dataset:

- s1-s7: satisfaction with the system
- q1-q6: perceived recommendation quality
- c1-c5: perceived control
- u1-u5: understandability
- cgraph: inspectability (0: list, 1: graph)
- citem-cfriend: control (baseline: no control)

Example

Construct	Item
System satisfaction	I would recommend TasteWeights to others. TasteWeights is useless. TasteWeights makes me more aware of my choice options. I can make better music choices with TasteWeights. I can find better music using TasteWeights. Using TasteWeights is a pleasant experience. TasteWeights has no real benefit for me.
Perceived Recommendation Quality	I liked the artists/bands recommended by the TasteWeights system. The recommended artists/bands fitted my preference. The recommended artists/bands were well chosen. The recommended artists/bands were relevant. TasteWeights recommended too many bad artists/bands. I didn't like any of the recommended artists/bands.
Perceived Control	I had limited control over the way TasteWeights made recommendations. TasteWeights restricted me in my choice of music. Compared to how I normally get recommendations, TasteWeights was very limited. I would like to have more control over the recommendations. I decided which information was used for recommendations.
Understandability	The recommendation process is not transparent. I understand how TasteWeights came up with the recommendations. TasteWeights explained the reasoning behind the recommendations. I am unsure how the recommendations were generated. The recommendation process is clear to me.

Example

Prepare the data (csv, space separated, …)

In RStudio:

- Import the dataset
- Install and load package 'lavaan'
- Write model definition: model <- '[definition]'
- Run model: fit <- cfa(model, [params])
- Inspect model output: summary(fit, [params])

Example

Write model definition:

```
model <- 'satisf =~ s1+s2+s3+s4+s5+s6+s7
quality =~ q1+q2+q3+q4+q5+q6
control =~ c1+c2+c3+c4+c5
underst =~ u1+u2+u3+u4+u5'
```

Run model:

```
fit <- cfa(model, data=twq, ordered=names(twq))
```

Inspect model output:

```
summary(fit, rsquare=TRUE, fit.measures=TRUE)
```

Example

In MPlus:

- Remove heading row from data file
- Make a new file in MPlus with the dataset and model definition
- Save file as model.inp
- Run the model, this will create and open model.out
- Inspect model output file

Example

Write dataset and model definition:

```
DATA: FILE IS twq.datm;
VARIABLE:
 names are s1 s2 s3 s4 s5 s6 s7 q1 q2 q3 q4 q5 q6
  c1 c2 c3 c4 c5 u1 u2 u3 u4 u5 cgraph citem cfriend;

 usevariables are s1-u5;
 categorical are s1-u5;

MODEL:
 satisf  by s1-s7;
 quality by q1-q6;
 control by c1-c5;
 underst by u1-u5;
```

Scaling a factor

Factors are **latent** variables

based on a linear combination of their indicators

They have no "scale"

Their mean and variance are **arbitrary**

We don't care about means

We only make comparisons anyway

We have to choose a variance

There are two methods for this…

Scaling a factor

Method 1: set one factor loading to 1.00

- All other loadings are relative to this one
- This is useful for between-dataset variance comparisons
- Regression coefficients are harder to interpret

Method 2: standardize the factor variance to 1.00

- Regression coefficients are then standardized effects

Scaling a factor

In R, change:

```
fit <- cfa(model, data=twq, ordered=names(twq), std.lv=TRUE)
```

In MPlus, add:

```
OUTPUT:
 standardized;
```

Modification indices

With high residuals, two things can happen:

1. Items may significantly load on other factors
2. There may be significant cross-correlation

MPlus/R can automatically detect these

In R, run:

```
modindices(fit,power=TRUE,sort=TRUE)
```

In MPlus, add to the output section:

```
modindices(3.84);
```

Evaluation

Improve the model based on **item-fit** statistics:

- Look at r-squared for each item (should be > 0.40)
- Look at modification indices (no “large” values)

Check construct **validity** based on **factor fit** statistics:

- Convergent validity: AVE > 0.5
- Discriminant validity: $\sqrt{\text{AVE}}$ > highest factor correlation

Evaluate the model based on **model fit** statistics:

- Chi-square test, CFI, TLI, RMSEA

Improve model

Based on r-squared, iteratively remove items:

c5 (r-squared = 0.180)

u1 (r-squared = 0.324)

Based on modification indices, remove item:

u3 loads on control (modification index = 15.287)

Construct validity

Satisfaction:

AVE = 0.709, √(AVE) = 0.842, largest correlation = 0.762

Quality:

AVE = 0.737, √(AVE) = 0.859, largest correlation = 0.687

Control:

AVE = 0.643, √(AVE) = 0.802, largest correlation = 0.762

Understandability:

AVE = 0.874, √(AVE) = 0.935, largest correlation = 0.341

Model-fit

Chi-square test of model fit:

- Tests whether there any significant misfit between estimated and observed correlation matrix
- Often this is true ($p < .05$)... models are rarely perfect!
- Alternative metric: chi-squared / df < 3 (good fit) or < 2 (great fit)

Model-fit

CFI and TLI:

- Relative improvement over baseline model; ranging from 0.00 to 1.00
- CFI should be > 0.96 and TLI should be > 0.95

RMSEA:

- Root mean square error of approximation
- Overall measure of misfit
- Should be < 0.05, and its confidence intervall should not exceed 0.10.

Model-fit

Use the "robust" column in R:

- Chi-Square value: 288.517, df: 164 (value/df = 1.76, good)
- CFI: 0.990, TLI: 0.989 (both good)
- RMSEA: 0.053 (slightly high), 90% CI: [0.043, 0.063] (ok)

Summary

Specify and run your CFA

Alter the model until all remaining items fit

- Make sure you have at least 3 items per factor!

Report final loadings, factor fit, and model fit

Summary

We conducted a CFA and examined the validity and reliability scores of the constructs measured in our study.

Upon inspection of the CFA model, we removed items c5 (communality: 0.180) and u1 (communality: 0.324), as well as item u3 (high cross-loadings with several other factors). The remaining items shared at least 48% of their variance with their designated construct.

Summary

To ensure the convergent validity of constructs, we examined the average variance extracted (AVE) of each construct. The AVEs were all higher than the recommended value of 0.50, indicating adequate convergent validity.

To ensure discriminant validity, we ascertained that the square root of the AVE for each construct was higher than the correlations of the construct with other constructs.

Finally, to confirm scale reliability we calculated Cronbach's alpha for each factor. Alpha scores were higher than 0.84, indicating excellent scale reliability.

Summary

Construct	**Item**	**Loading**
System satisfaction Alpha: 0.92 AVE: 0.709	I would recommend TasteWeights to others.	0.888
	TasteWeights is useless.	-0.885
	TasteWeights makes me more aware of my choice options.	0.768
	I can make better music choices with TasteWeights.	0.822
	I can find better music using TasteWeights.	0.889
	Using TasteWeights is a pleasant experience.	0.786
	TasteWeights has no real benefit for me.	-0.845
Perceived Recommendation Quality Alpha: 0.90 AVE: 0.737	I liked the artists/bands recommended by the TasteWeights system.	0.950
	The recommended artists/bands fitted my preference.	0.950
	The recommended artists/bands were well chosen.	0.942
	The recommended artists/bands were relevant.	0.804
	TasteWeights recommended too many bad artists/bands.	-0.697
	I didn't like any of the recommended artists/bands.	-0.775
Perceived Control Alpha: 0.84 AVE: 0.643	I had limited control over the way TasteWeights made recommendations.	0.700
	TasteWeights restricted me in my choice of music.	0.859
	Compared to how I normally get recommendations, TasteWeights was very limited.	0.911
	I would like to have more control over the recommendations.	0.716
	I decided which information was used for recommendations.	
Understandability Alpha: 0.92 AVE: 0.874	The recommendation process is not transparent.	
	I understand how TasteWeights came up with the recommendations.	0.893
	TasteWeights explained the reasoning behind the recommendations.	
	I am unsure how the recommendations were generated.	-0.923
	The recommendation process is clear to me.	0.987

Summary

Construct	Item	Loading	Response Frequencies				
			-2	-1	0	1	2
System satisfaction Alpha: 0.92 AVE: 0.709	I would recommend TasteWeights to others.	0.888	9	32	47	128	51
	TasteWeights is useless.	-0.885	99	106	29	27	6
	TasteWeights makes me more aware of my choice options.	0.768	11	43	56	125	32
	I can make better music choices with TasteWeights.	0.822	12	50	70	95	40
	I can find better music using TasteWeights.	0.889	14	45	62	109	37
	Using TasteWeights is a pleasant experience.	0.786	0	11	38	130	88
	TasteWeights has no real benefit for me.	-0.845	56	91	49	53	18
Perceived Recommendation Quality Alpha: 0.90 AVE: 0.737	I liked the artists/bands recommended by the TasteWeights system.	0.950	6	30	27	125	79
	The recommended artists/bands fitted my preference.	0.950	10	30	24	123	80
	The recommended artists/bands were well chosen.	0.942	10	35	26	101	95
	The recommended artists/bands were relevant.	0.804	4	18	14	120	111
	TasteWeights recommended too many bad artists/bands.	-0.697	104	88	45	20	10
	I didn't like any of the recommended artists/bands.	-0.775	174	61	16	14	2
Perceived Control Alpha: 0.84 AVE: 0.643	I had limited control over the way TasteWeights made recommendations.	0.700	13	52	48	112	42
	TasteWeights restricted me in my choice of music.	0.859	40	90	45	76	16
	Compared to how I normally get recommendations, TasteWeights was very limited.	0.911	36	86	53	68	24
	I would like to have more control over the recommendations.	0.716	8	27	38	130	64
	I decided which information was used for recommendations.		42	82	50	79	14
Understandability Alpha: 0.92 AVE: 0.874	The recommendation process is not transparent.		24	77	76	68	22
	I understand how TasteWeights came up with the recommendations.	0.893	8	41	17	127	74
	TasteWeights explained the reasoning behind the recommendations.		28	59	46	91	43
	I am unsure how the recommendations were generated.	-0.923	71	90	28	62	16
	The recommendation process is clear to me.	0.987	14	65	23	101	64

Summary

	Alpha	AVE	Satisfaction	Quality	Control	Underst.
Satisfaction	0.92	0.709	**0.842**	0.687	–0.762	0.336
Quality	0.90	0.737	0.687	**0.859**	–0.646	0.282
Control	0.84	0.643	–0.762	–0.646	**0.802**	–0.341
Underst.	0.92	0.874	0.336	0.282	–0.341	**0.935**

diagonal: $\sqrt{(AVE)}$

off-diagonal: correlations

Learn more?

Learn it yourself:

- Sections on CFA in Rex Kline, "Principles and Practice of Structural Equation Modeling", 3rd ed.
- MPlus: check the video tutorials at www.statmodel.com

SEM

Structural Equation Modeling in R and MPlus

SEM

Steps involved in constructing a SEM for an **experiment**:

(a method that is confirmatory, but leaves room for data-driven changes in the model)

Step 1: Build your CFA

Step 2: Analyze the marginal effects of the manipulations

Step 3: Test and trim a saturated model

Marginal effects

First analysis: manipulations —> factors

MIMIC model (Multiple Indicators, Multiple Causes)

The SEM equivalent of a t-test / (factorial) ANOVA

Only for experiments (not for surveys)

Steps involved:

- Build your CFA
- Create dummies for your experimental conditions
- Run regressions factor-by-factor

Create your CFA

Take the final CFA

E.g., in R:

```
model <- 'satisf =~ s1+s2+s3+s4+s5+s6+s7
quality =~ q1+q2+q3+q4+q5+q6
control =~ c1+c2+c3+c4
underst =~ u2+u4+u5’
```

Don’t run it yet! We are going to add extra lines to this model…

Create dummies

Main effects are already built for our dataset:

- Control conditions ("no control" is the baseline):

  `citem cfriend`

- Inspectability conditions ("list view" is the baseline):

  `cgraph`

What about the interaction effect?

- We need to create dummies for that too!

Create dummies

In MPlus, add:

```
DEFINE:
    cig = citem * cgraph;
    cfg = cfriend * cgraph;
```

In R, run:

```
twq$cig = twq$citem * twq$cgraph;
twq$cfg = twq$cfriend * twq$cgraph;
```

Run regressions

In MPlus (note the different notation for standardization!):

```
<...>

DEFINE:
 cig = citem * cgraph;
 cfg = cfriend * cgraph;

MODEL:
 satisf BY s1* s2-s7;
 quality BY q1* q2-q6;
 control BY c1* c2-c4;
 underst BY u2* u4-u5;
 satisf-underst@1;

 satisf ON citem cfriend cgraph cig cfg;
```

Run regressions

In R:

```
model <- 'satisf =~ s1+s2+s3+s4+s5+s6+s7
quality =~ q1+q2+q3+q4+q5+q6
control =~ c1+c2+c3+c4
underst =~ u2+u4+u5
satisf ~ citem+cfriend+cgraph+cig+cfg';

fit <-
sem(model,data=twq,ordered=names(twq[1:23]),std.lv=TRUE);

summary(fit);
```

Results

Note: effects are not significant (but that's okay for now)

	Estimate	S.E.	Est./S.E.	Two-Tailed P-Value
SATISF ON				
CITEM	0.269	0.233	1.155	0.248
CFRIEND	0.197	0.223	0.883	0.377
CGRAPH	0.375	0.221	1.696	0.090
CIG	-0.131	0.320	-0.409	0.683
CFG	-0.048	0.309	-0.157	0.875

Interpretation

Citem: effect of item control vs. no control in the list view condition

Cfriend: effect of friend control vs. no control in the list view condition

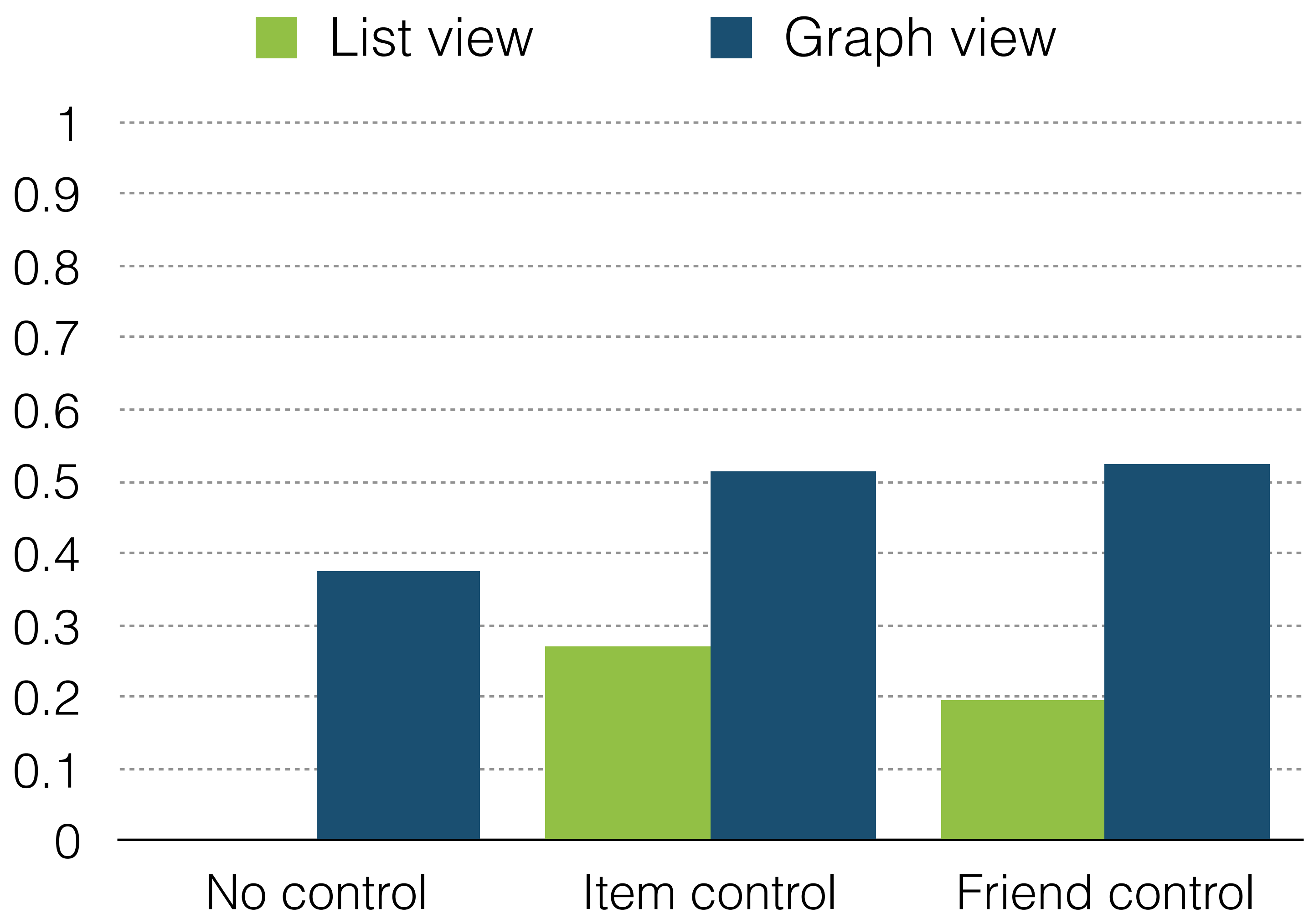
Cgraph: effect of graph view vs. list view in the “no control” condition

Cig: additional effect of item control in the graph view condition (or: additional effect of graph view in the item control condition)

Cfg: additional effect of friend control in the graph view condition (or: additional effect of graph view in the friend control condition)

Graph

Note: no control, list view is set to zero!



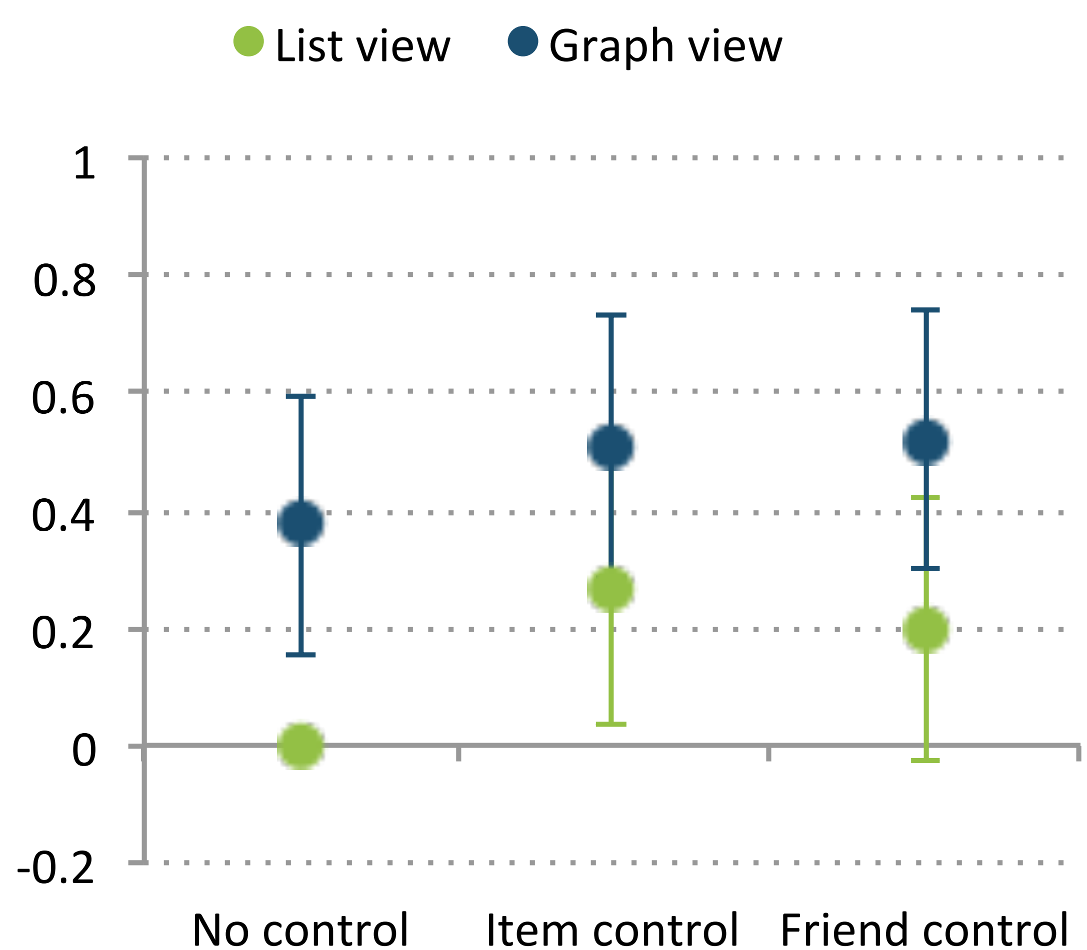
For a better graph

```
<...>

DEFINE:
 cil = citem * (1-cgraph);
 cfl = cfriend * (1-cgraph);
 cng = (1-citem) * (1-cfriend) * cgraph;
 cig = citem * cgraph;
 cfg = cfriend * cgraph;

MODEL:
 satisf BY s1* s2-s7;
 quality BY q1* q2-q6;
 control BY c1* c2-c4;
 underst BY u2* u4-u5;
 satisf-underst@1;

 satisf ON cil cfl cng cig cfg;
```

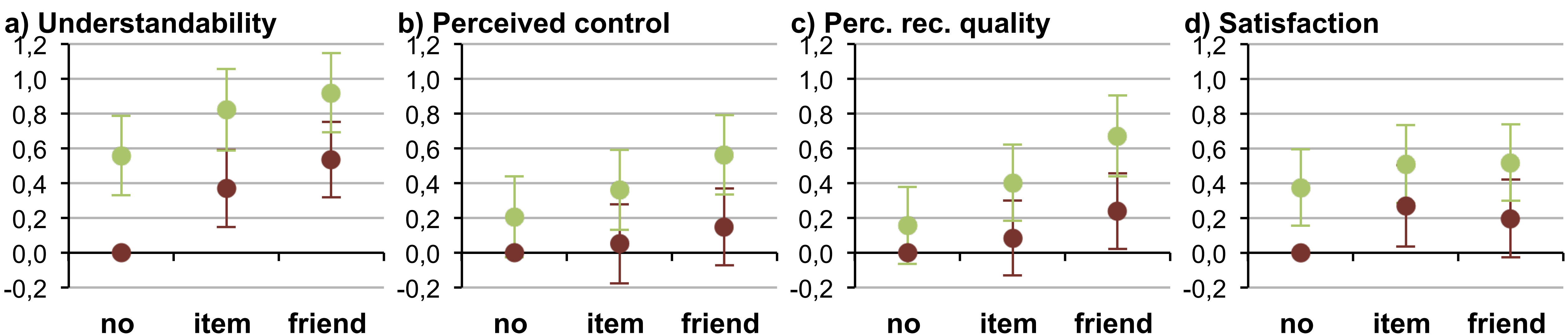
Better graph

Includes error bars (+/- 1 SE)

Easier to see that baseline is fixed to zero



Repeat



From: Knijnenburg et al. (2012): "Inspectability and Control in Social Recommenders", *RecSys'12*

Main finding

Main effects of inspectability and control conditions on understandability (no interaction effect)

	Estimate	S.E.	Est./S.E.	Two-Tailed P-Value
UNDERST ON				
CITEM	0.365	0.229	1.598	0.110
CFRIEND	0.562	0.223	2.525	0.012
CGRAPH	0.596	0.232	2.566	0.010
CIG	-0.050	0.332	-0.151	0.880
CFG	-0.169	0.326	-0.519	0.604

Modeling: theory

Creating a research model

Modeling: theory

Do this **before** you do your study!

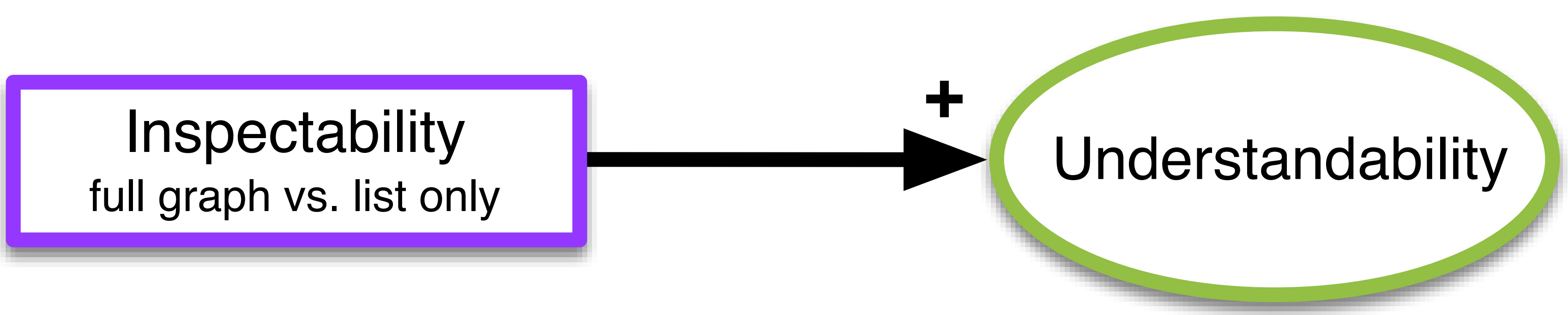
Motivate expected effects, based on:

- previous work
- theory
- common sense

If in doubt, create alternate specifications!

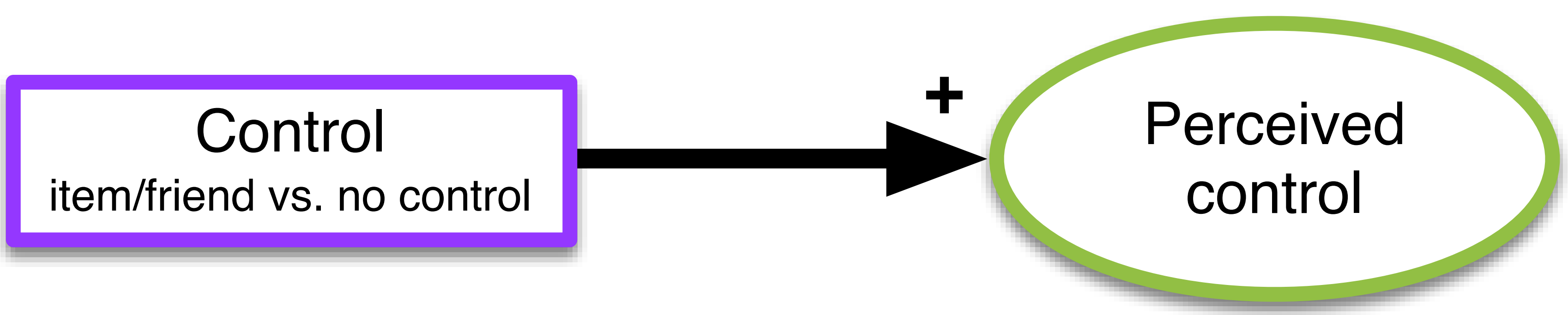
Inspectability

Herlocker argues that explanation provides transparency, "exposing the reasoning behind a recommendation".



Control

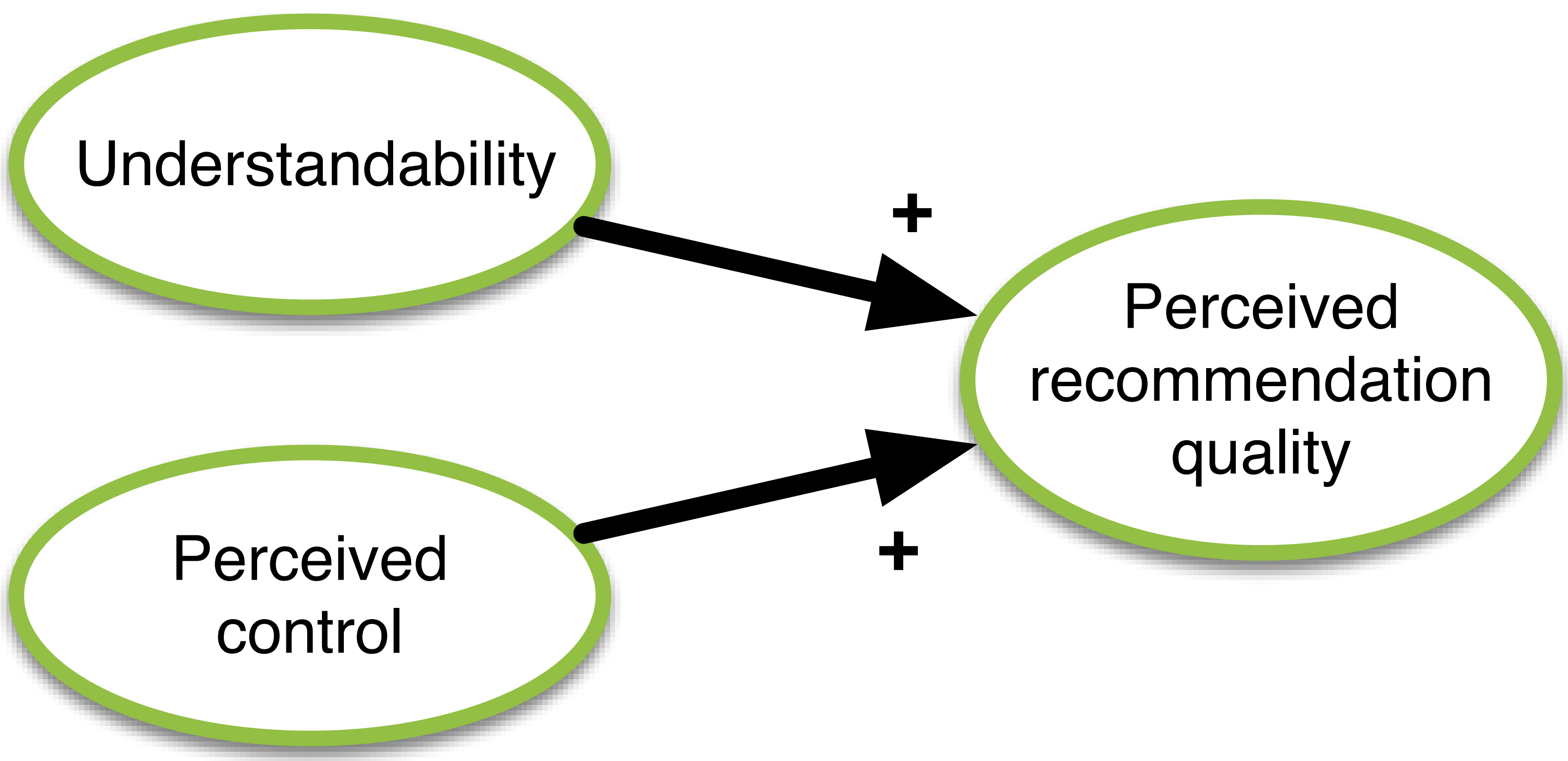
Multiple studies highlight the benefits of interactive interfaces that support control over the recommendation process.



Perceived quality

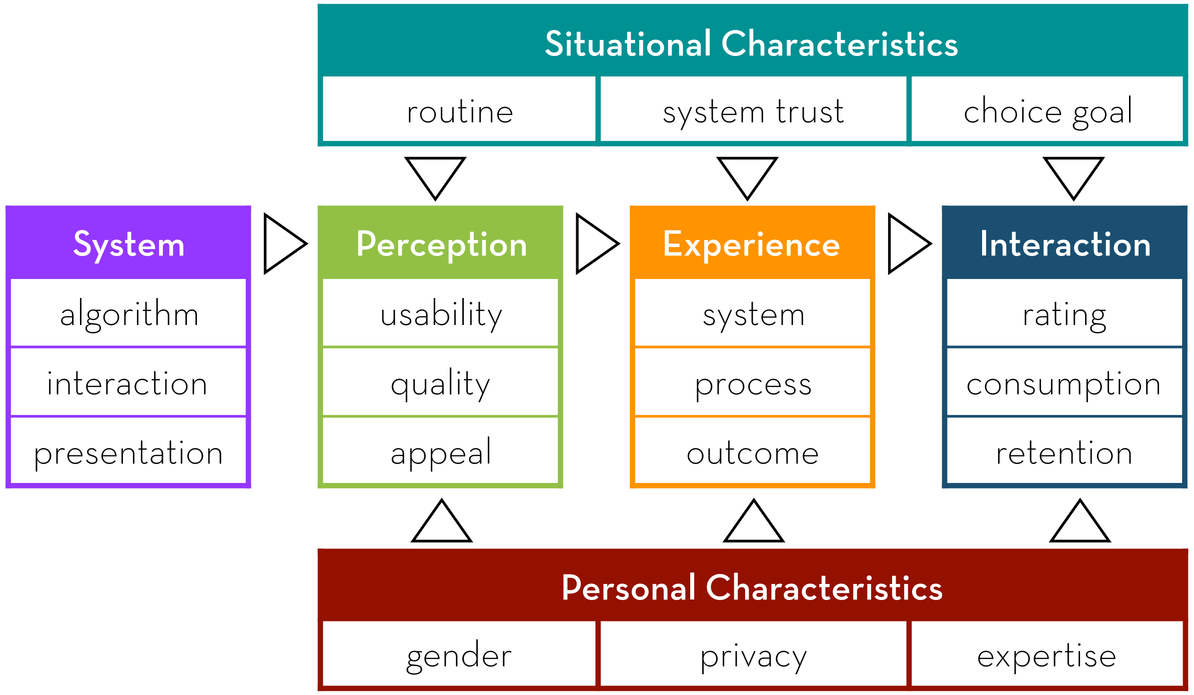
Tintarev and Masthoff show that explanations make it easier to judge the quality of recommendations.

McNee et al. found that study participants preferred user-controlled interfaces because these systems “best understood their tastes”.



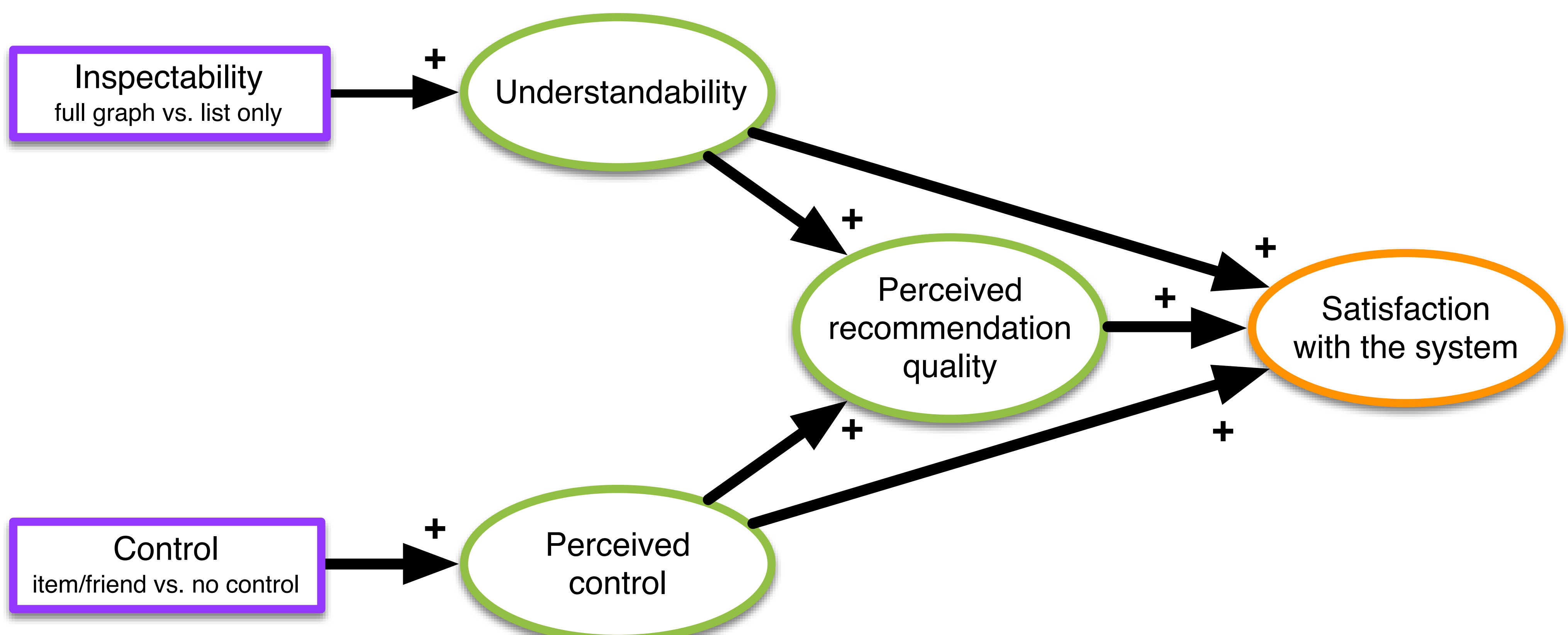
Satisfaction

Knijnenburg et al. developed a framework that describes how certain manipulations influence subjective system aspects (i.e. understandability, perceived control and recommendation quality), which in turn influence user experience (i.e. system satisfaction).



Satisfaction

Knijnenburg et al. developed a framework that describes how certain manipulations influence subjective system aspects (i.e. understandability, perceived control and recommendation quality), which in turn influence user experience (i.e. system satisfaction).



Modeling: practice

Testing your research model

Modeling: practice

Steps:

- Build and trim the core model
- Get model fit statistics
- Optional: expand the model
- Reporting

Model building

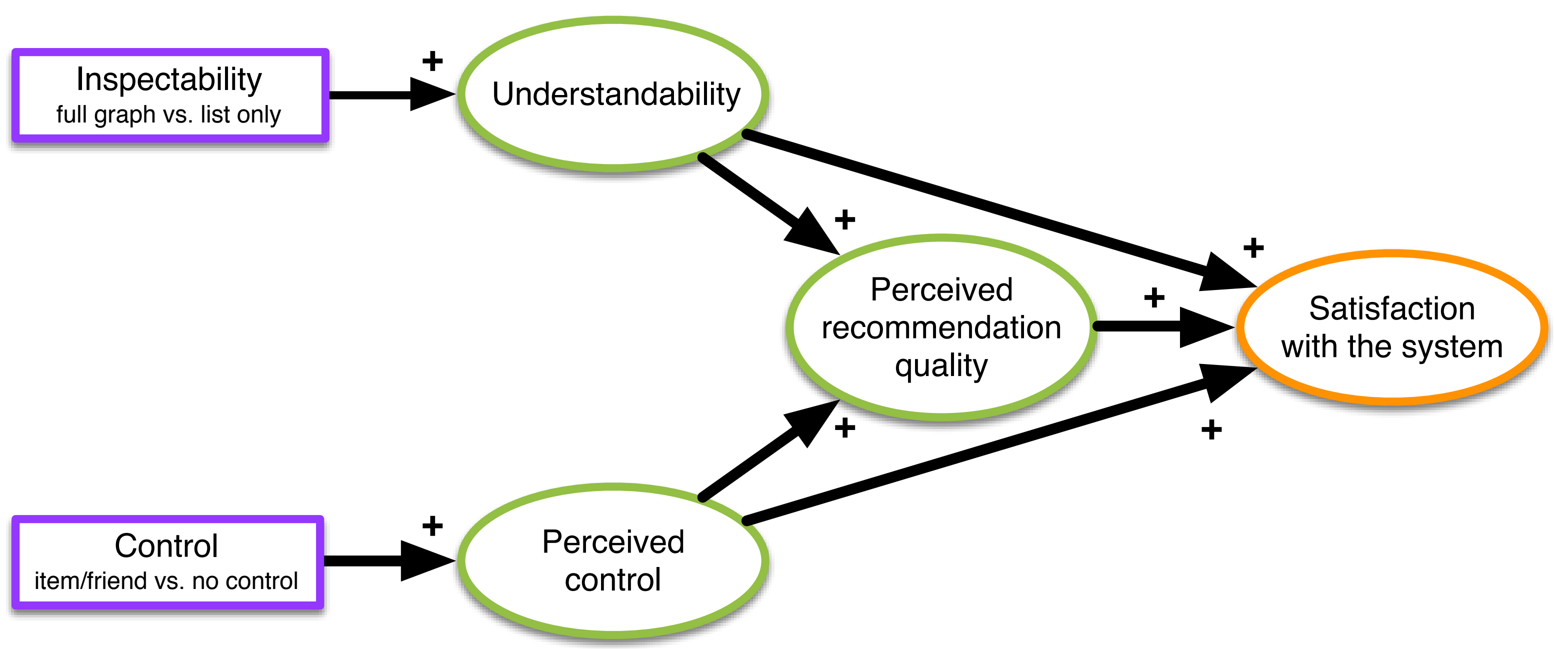
Steps:

- Determine the causal order and create a saturated model
- Trim the model
- Inspect modification indices
- Try alternative specifications, pick the best alternative (optional)

Causal order

Find the causal order of your model

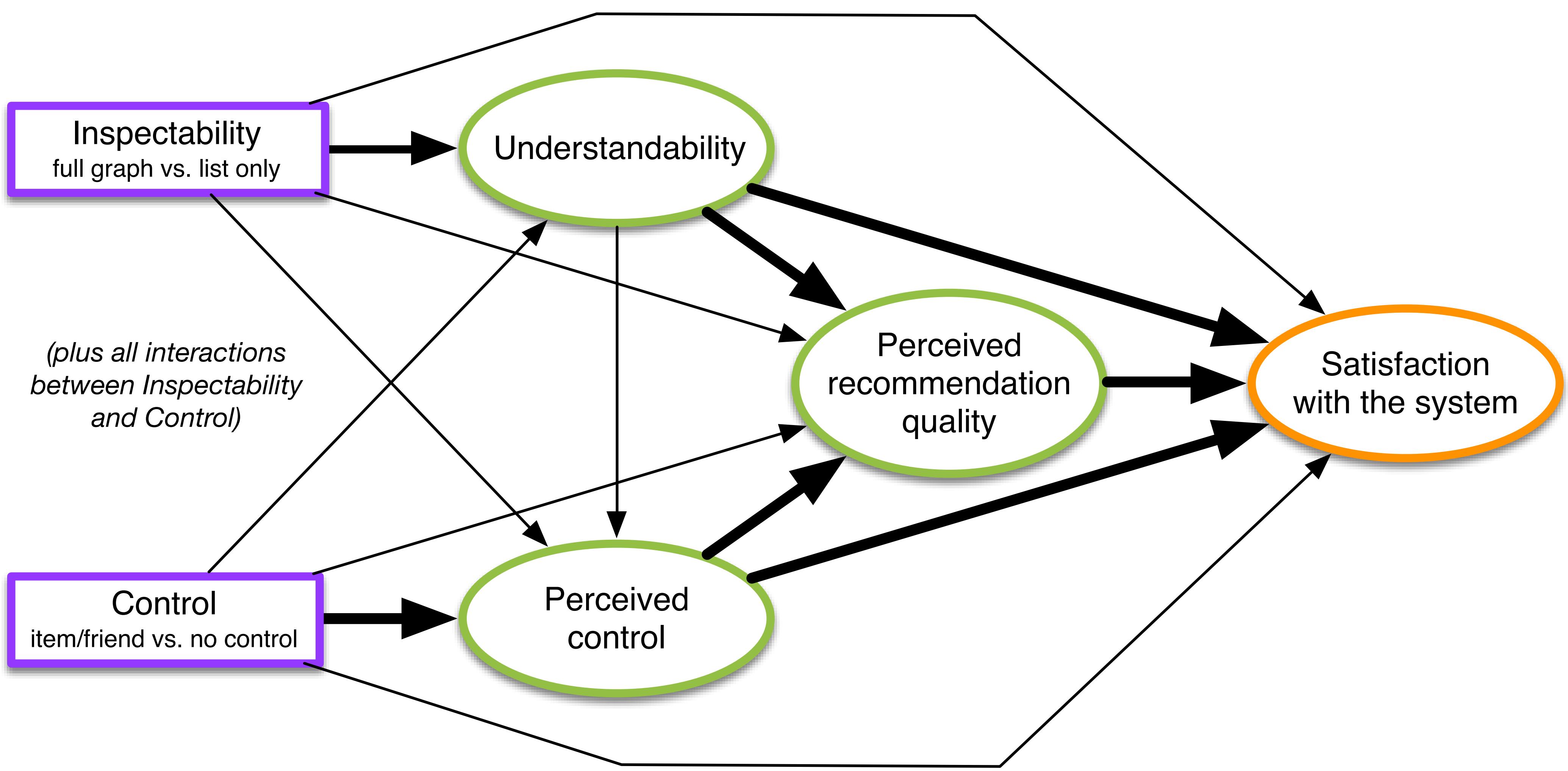
(fill the gaps where necessary)



conditions -> understandability -> perceived control -> perceived recommendation quality -> satisfaction

Saturated model

Fill in all forward-going arrows



Run model

In MPlus:

```
MODEL:
 satisf BY s1* s2-s7;
 quality BY q1* q2-q6;
 control BY c1* c2-c4;
 underst BY u2* u4-u5;
 satisf-underst@1;

 satisf ON quality control underst citem cfriend cgraph cig cfg;
 quality ON control underst citem cfriend cgraph cig cfg;
 control ON underst citem cfriend cgraph cig cfg;
 underst ON citem cfriend cgraph cig cfg;
```

Run model

In R:

```
model <- 'satisf =~ s1+s2+s3+s4+s5+s6+s7
  quality =~ q1+q2+q3+q4+q5+q6
  control =~ c1+c2+c3+c4
  underst =~ u2+u4+u5
  satisf ~ quality+control+underst+citem+cfriend+cgraph+cig+cfg
  quality ~ control+underst+citem+cfriend+cgraph+cig+cfg
  control ~ underst+citem+cfriend+cgraph+cig+cfg
  underst ~ citem+cfriend+cgraph+cig+cfg';

fit <- sem(model,data=twq,ordered=names(twq[1:23]),std.lv=TRUE);

summary(fit);
```

Trim model

Rules:

- Start with the least significant and least interesting effects (those that were added for saturation)
- Work iteratively
- Manipulations with >2 conditions: remove all dummies at once (if only one is significant, keep the others as well)
- Interaction+main effects: never remove main effect before the interaction effect (if only the interaction is significant, keep the main effect regardless)

Results

	Estimate	S.E.	Est./S.E.	Two-Tailed P-Value
SATISF ON				
QUALITY	0.438	0.076	5.744	0.000
CONTROL	-0.832	0.108	-7.711	0.000
UNDERST	0.105	0.078	1.354	0.176
QUALITY ON				
CONTROL	-0.757	0.085	-8.877	0.000
UNDERST	0.057	0.076	0.754	0.451
CONTROL ON				
UNDERST	-0.322	0.069	-4.685	0.000
SATISF ON				
CITEM	0.313	0.263	1.190	0.234
CFRIEND	0.004	0.256	0.014	0.988
CGRAPH	0.297	0.228	1.302	0.193
CIG	-0.389	0.356	-1.092	0.275
CFG	-0.391	0.356	-1.097	0.273

Results

	Estimate	S.E.	Est./S.E.	Two-Tailed P-Value
QUALITY ON				
CITEM	0.041	0.203	0.203	0.839
CFRIEND	0.157	0.250	0.628	0.530
CGRAPH	0.000	0.235	-0.001	0.999
CIG	**0.105**	**0.316**	**0.333**	**0.739**
CFG	**0.182**	**0.373**	**0.488**	**0.625**
CONTROL ON				
CITEM	0.057	0.243	0.234	0.815
CFRIEND	0.024	0.221	0.109	0.913
CGRAPH	-0.024	0.240	-0.100	0.921
CIG	**-0.132**	**0.343**	**-0.384**	**0.701**
CFG	**-0.273**	**0.330**	**-0.828**	**0.408**
UNDERST ON				
CITEM	0.365	0.229	1.596	0.110
CFRIEND	0.562	0.223	2.522	0.012
CGRAPH	0.596	0.232	2.568	0.010
CIG	**-0.050**	**0.332**	**-0.149**	**0.881**
CFG	**-0.169**	**0.326**	**-0.518**	**0.604**

Trimming steps

Remove interactions -> (1) understandability, (2) quality, (3) control, and (4) satisfaction

with the latter, also remove the dummies from usevariables

Remove cgraph -> (1) satisfaction, and (2) quality

Trimming steps

Remove citem and cfriend -> control

But wait… did we not hypothesize that effect?

- Yes, but we still have citem+cfriend -> underst -> control!

In other words: the effect of item and friend control on perceived control is mediated by understandability!

- Argument: "Controlling items/friends gives me a better understanding of how the system works, so in turn I feel more in control"

Trimming steps

Remove citem and cfriend -> satisfaction

Remove understandability -> recommendation quality

- We hypothesized this effect, but it is still mediated by control.
- Argument: "Understanding the recommendations gives me a feeling of control, which in turn makes me like the recommendations better."

Remove understandability -> satisfaction

- Same thing

Trimming steps

Remove citem and cfriend -> recommendation quality

Remove cgraph -> control

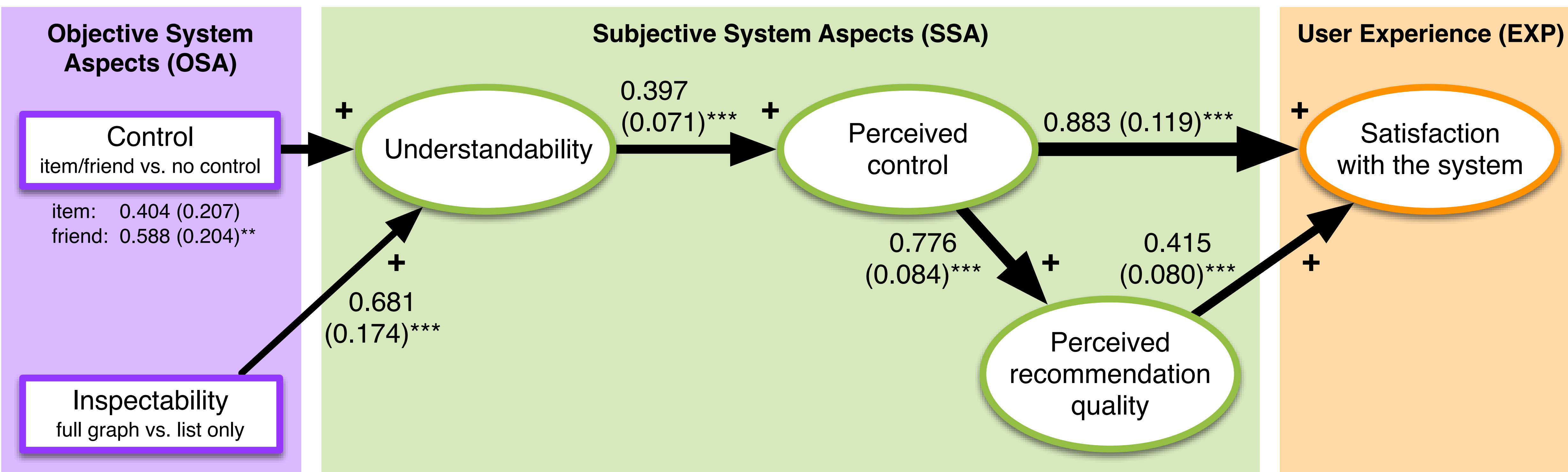
Again: still mediated by understandability

Stop! All remaining effects are significant!

Trimmed model

	Estimate	S.E.	Est./S.E.	Two-Tailed P-Value
SATISF ON				
QUALITY	0.415	0.080	5.211	0.000
CONTROL	-0.883	0.119	-7.398	0.000
QUALITY ON				
CONTROL	-0.776	0.084	-9.198	0.000
CONTROL ON				
UNDERST	-0.397	0.071	-5.619	0.000
UNDERST ON				
CITEM	0.404	0.207	1.950	0.051
CFRIEND	0.588	0.204	2.878	0.004
CGRAPH	0.681	0.174	3.924	0.000

Trimmed model



Modindices

```
ON/BY Statements

SATISF   ON UNDERST  /
UNDERST  BY SATISF          4.037      0.098      0.063      0.063
CONTROL  ON SATISF   /
SATISF   BY CONTROL         6.912      0.313      0.489      0.489
UNDERST  ON CONTROL  /
CONTROL  BY UNDERST        13.256      0.288      0.288      0.288

ON Statements

SATISF   ON CGRAPH          4.119      0.238      0.140      0.070
QUALITY  ON CFRIEND         6.691      0.301      0.230      0.108
QUALITY  ON CGRAPH          6.613      0.245      0.187      0.094
CONTROL  ON CGRAPH          9.164     -0.213     -0.196     -0.098
```

Some of these we removed earlier

For some of these we already have the alternate direction

Assess model fit

Item and factor fit should not have changed much

(please double-check!)

Great model fit!

- Chi-Square value: 306.685, df: 223 (value/df = 1.38)
- CFI: 0.994, TLI: 0.993
- RMSEA: 0.037 (great), 90% CI: [0.026, 0.047]

Regression R^2

Satisfaction: 0.654

Perceived Recommendation Quality: 0.416

Perceived Control: 0.156

Understandability: 0.151

These are all quite okay

Omnibus test

In MPlus, change/add:

Under MODEL:

```
underst ON citem cfriend cgraph (p1-p3);
```

At the end:

```
MODEL TEST:
    p1=0;
    p2=0;
```

In R, change/add:

In model definition:

```
underst ~ cgraph+p1*citem+p2*cfriend
```

Then run:

```
lavTestWald(fit,'p1==0;p2==0');
```

Omnibus test

```
Wald Test of Parameter Constraints

          Value                                      8.516
          Degrees of Freedom                             2
          P-Value                                   0.0142
```

Omnibus effect of control is significant

Total effects

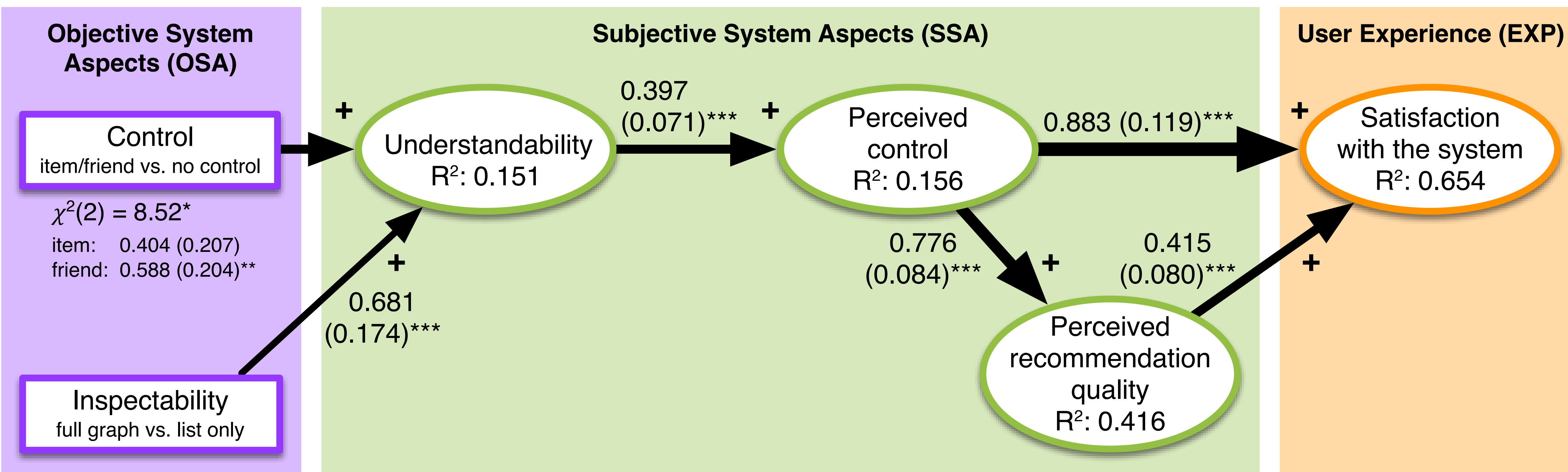
In MPlus:

```
MODEL INDIRECT:
 satisf IND citem;
 satisf IND cfriend;
 satisf IND cgraph;
 quality IND citem;
 quality IND cfriend;
 quality IND cgraph;
 control IND citem;
 control IND cfriend;
 control IND cgraph;
```

In R:

No automatic function for this; check out
http://lavaan.ugent.be/tutorial/mediation.html

Final core model



Reporting

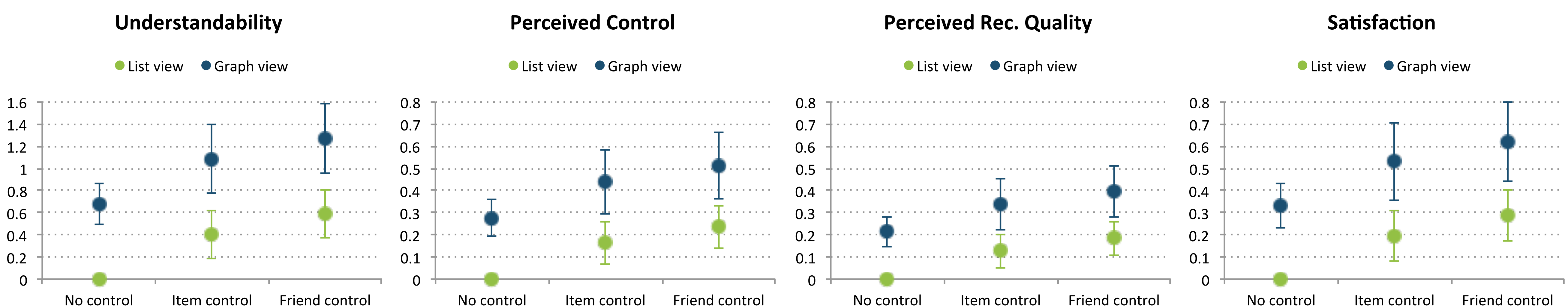
We subjected the 4 factors and the experimental conditions to structural equation modeling, which simultaneously fits the factor measurement model and the structural relations between factors and other variables. The model has a good* model fit: chi-square(223) = 306.685, $p = .0002$; RMSEA = 0.037, 90% CI: [0.026, 0.047], CFI = 0.994, TLI = 0.993.

* A model should not have a non-significant chi-square ($p > .05$), but this statistic is often regarded as too sensitive. Hu and Bentler propose cut-off values for other fit indices to be: CFI > .96, TLI > .95, and RMSEA < .05, with the upper bound of its 90% CI below 0.10.

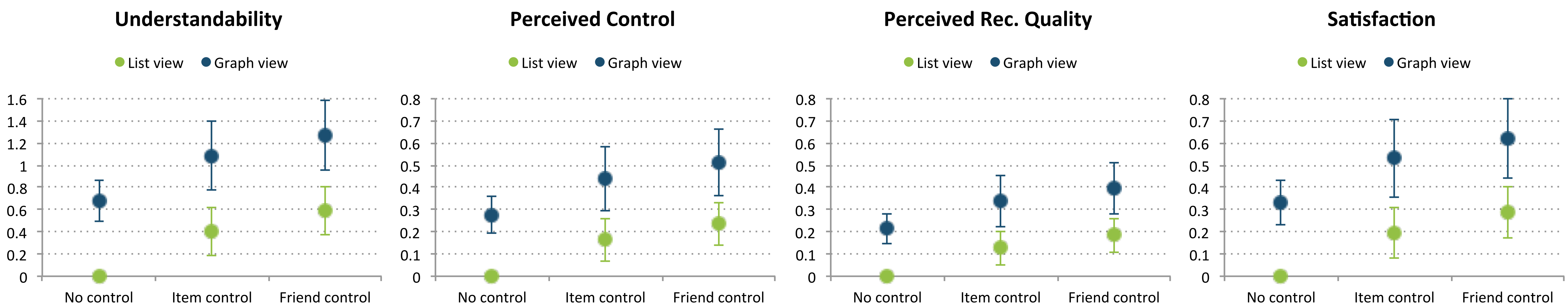
Reporting

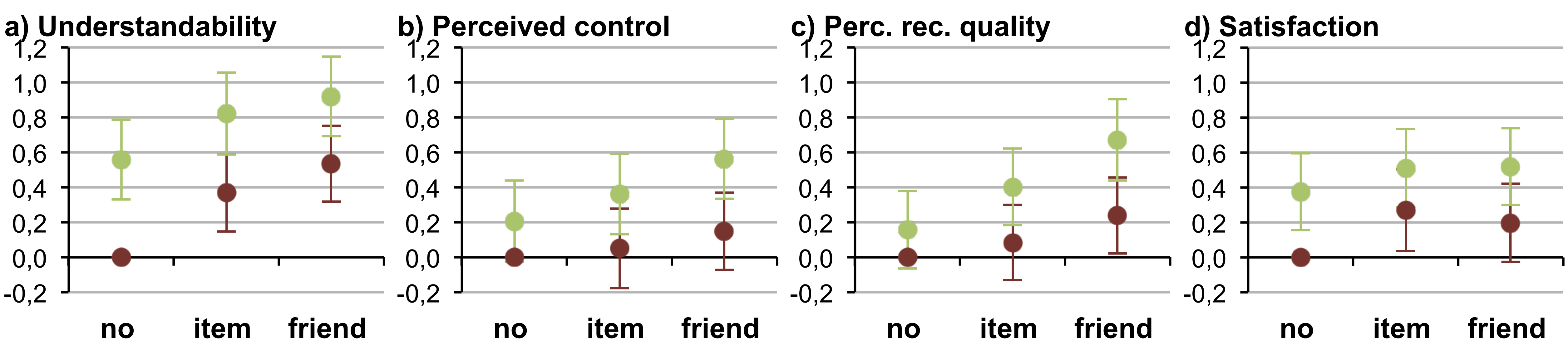
The model shows that the inspectability and control manipulations each have an independent positive effect on the understandability of the system: the full graph condition is more understandable than the list only condition, and the item control and friend control conditions are more understandable than the no control condition. Understandability is in turn related to users' perception of control, which is in turn related to the perceived quality of the recommendations. The perceived control and the perceived recommendation quality finally determine participants' satisfaction with the system.

Total effect graphs



Why different?





Why different?

Error bars are smaller because total effects are **mediated**

(mediation increases the accuracy of estimation)

Values may be different because total effects are **modeled**

(there may be some model misspecification)

Which one should I use?

Marginal effect graphs are more "honest"

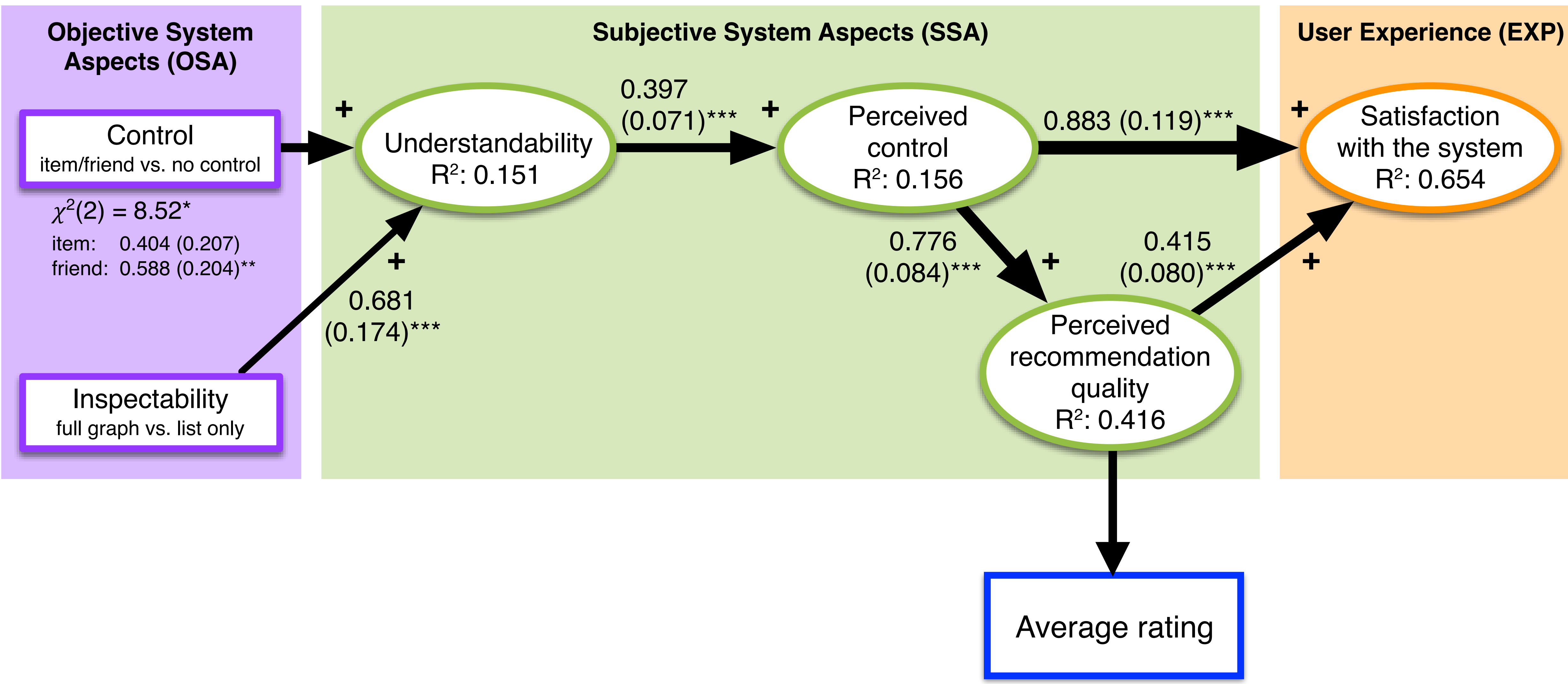
Expand the model

Expanding the model by adding additional variables

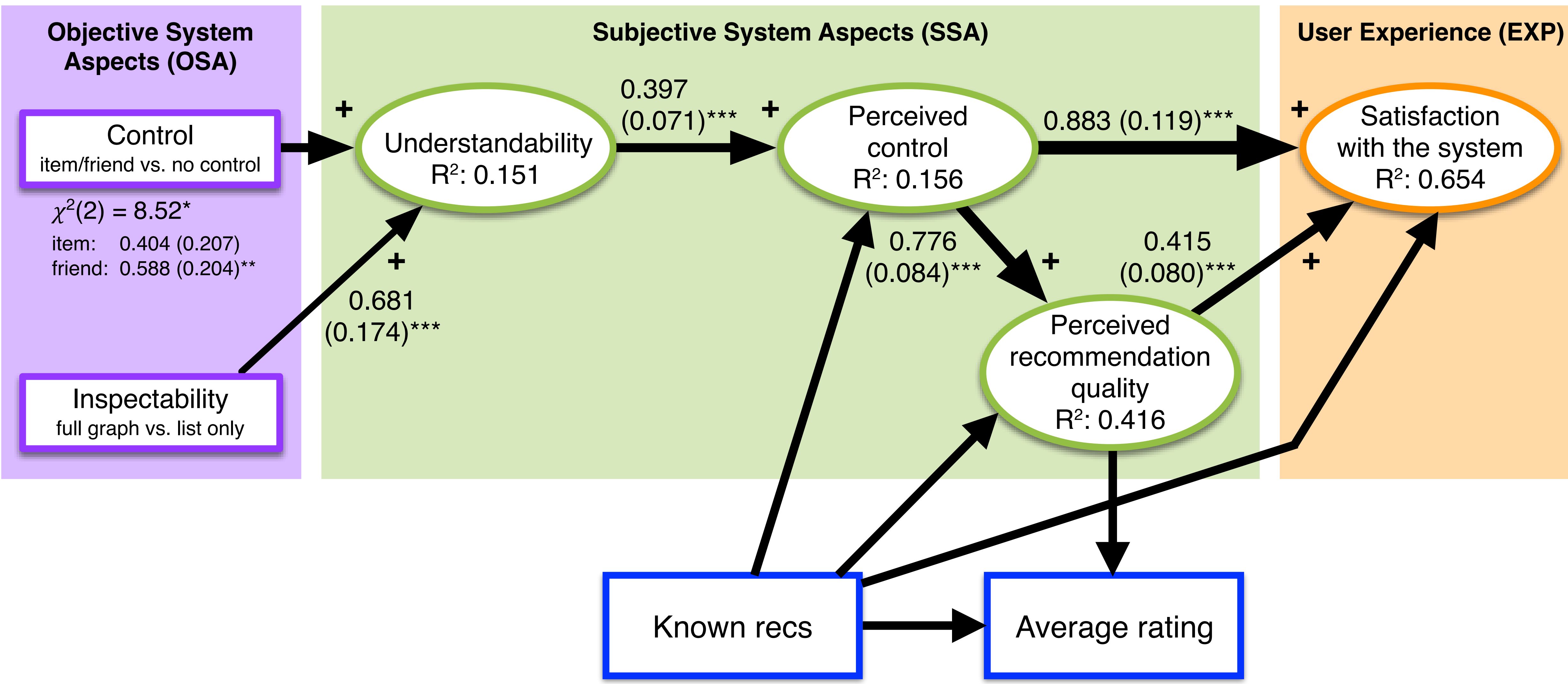
- This is typically where behavior comes in

Redo model tests and additional stats

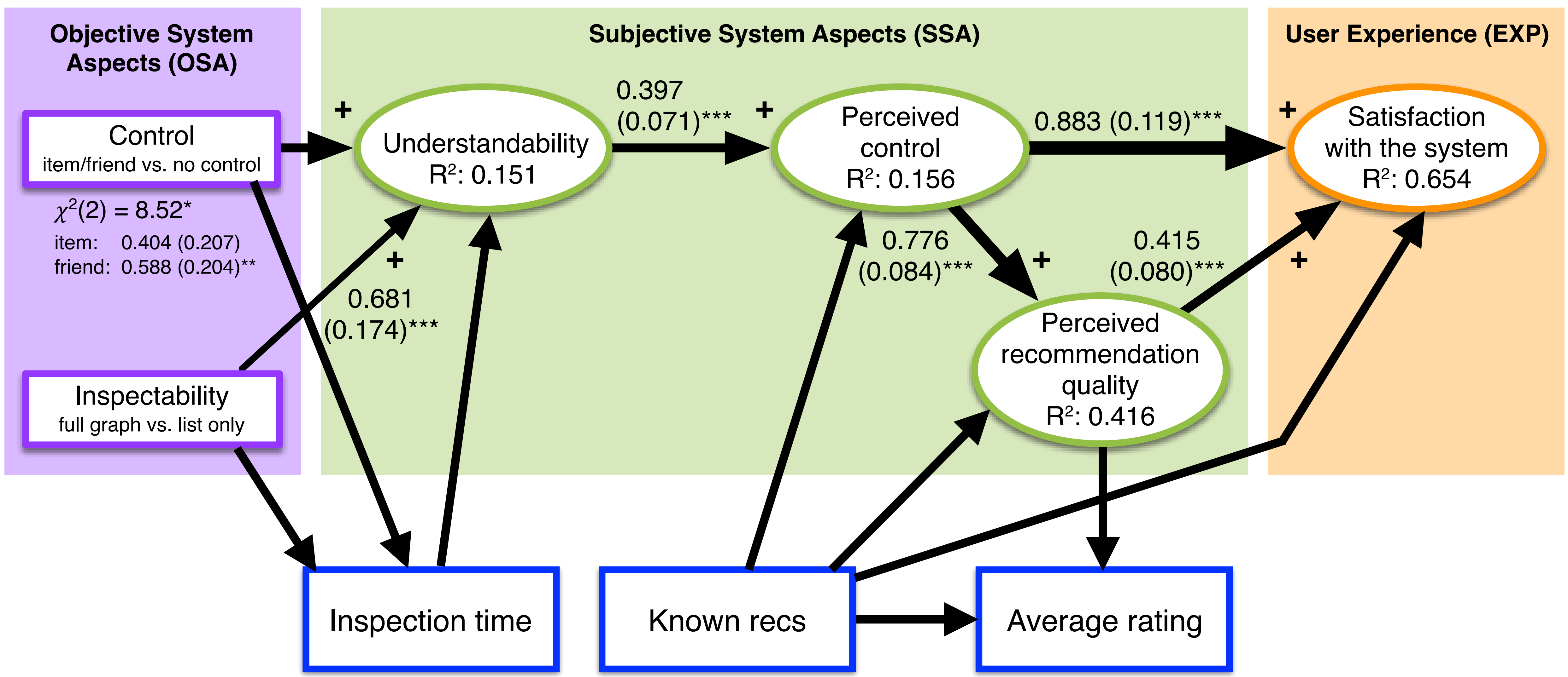
Expand the model



Expand the model



Expand the model



Final model

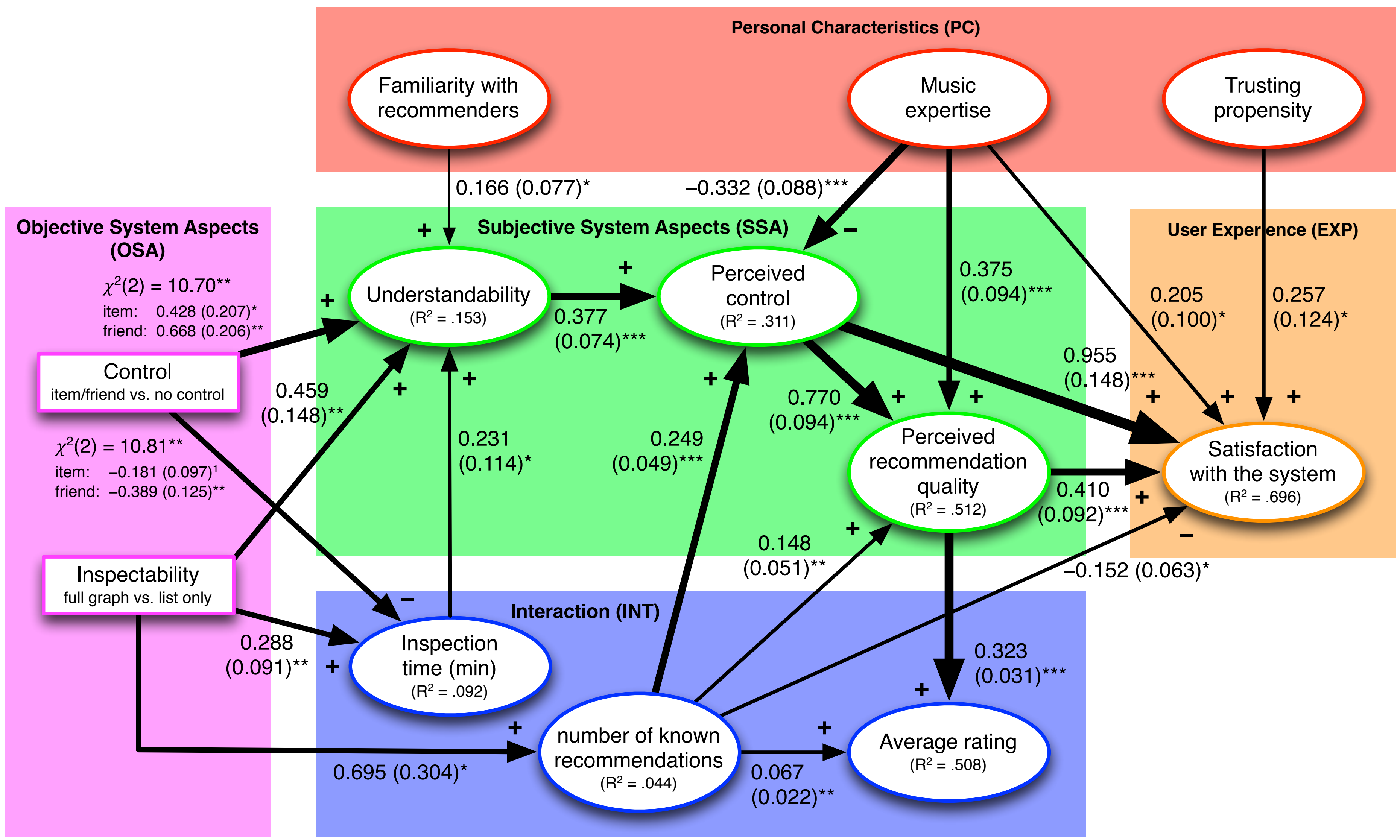


Figure 3. The structural equation model for the data of the experiment. Significance levels: *** $p < .001$, ** $p < .01$, 'ns' $p > .05$. R^2 is the proportion of variance explained by the model. Numbers on the arrows (and their thickness) represent the β coefficients (and standard error) of the effect. Factors are scaled to have an SD of 1.

From: Knijnenburg et al. (2012): "Inspectability and Control in Social Recommenders", *RecSys'12*

Learn more?

Learn it yourself:

Rex Kline, "Principles and Practice of Structural Equation Modeling", 3rd ed.

MPlus: check the video tutorials at www.statmodel.com

Part 4: Advanced

the really cool stuff…

Advanced Topics

In this part I discuss the following advanced topics:

- Multi-level SEM
- Interaction effects in SEM
- Cluster analysis

Multi-level SEM

in MPlus

Multi-level SEM

Repeated measurements

e.g. participants make 30 decisions

(Partially) within-subjects design

e.g. participants are randomly assigned to 1 of 3 games, and test it once with sound on and once with sound off

Grouped data

e.g. participants perform tasks in groups of 5

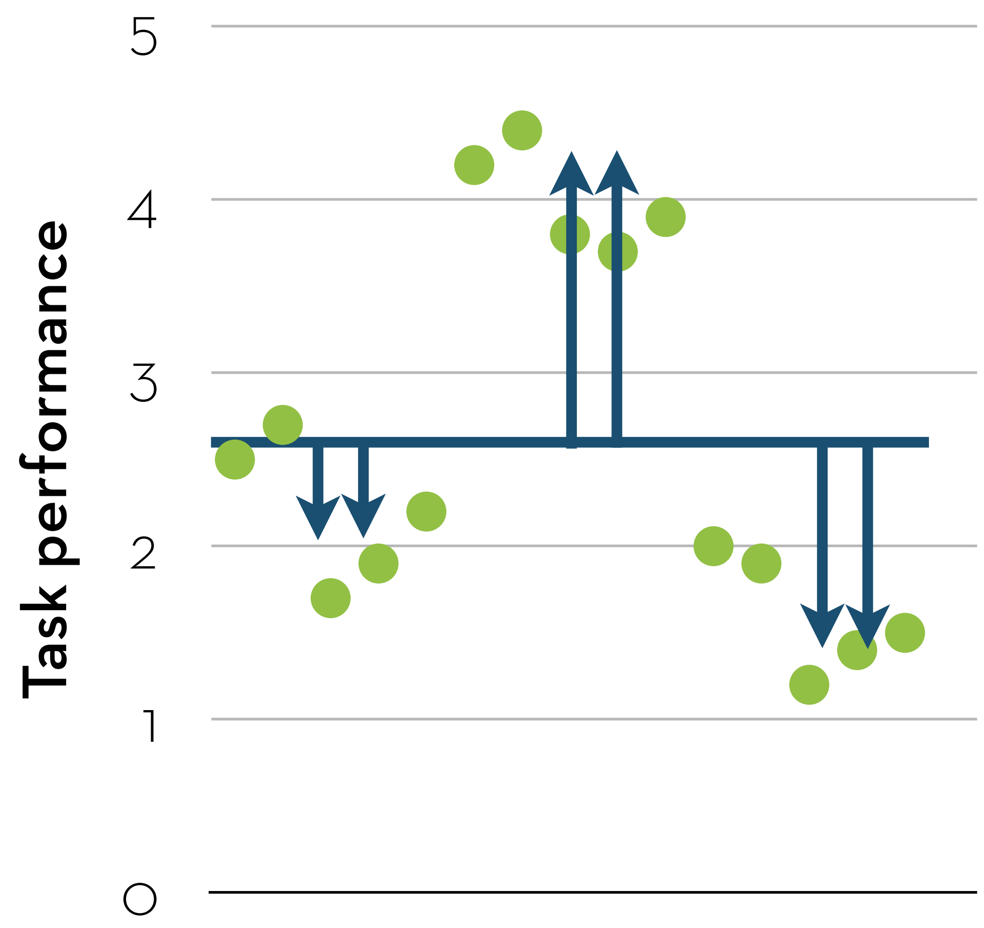
A combination of the above

Correlated errors

Consequence: errors are correlated

There will be a user-bias (and maybe an task-bias)

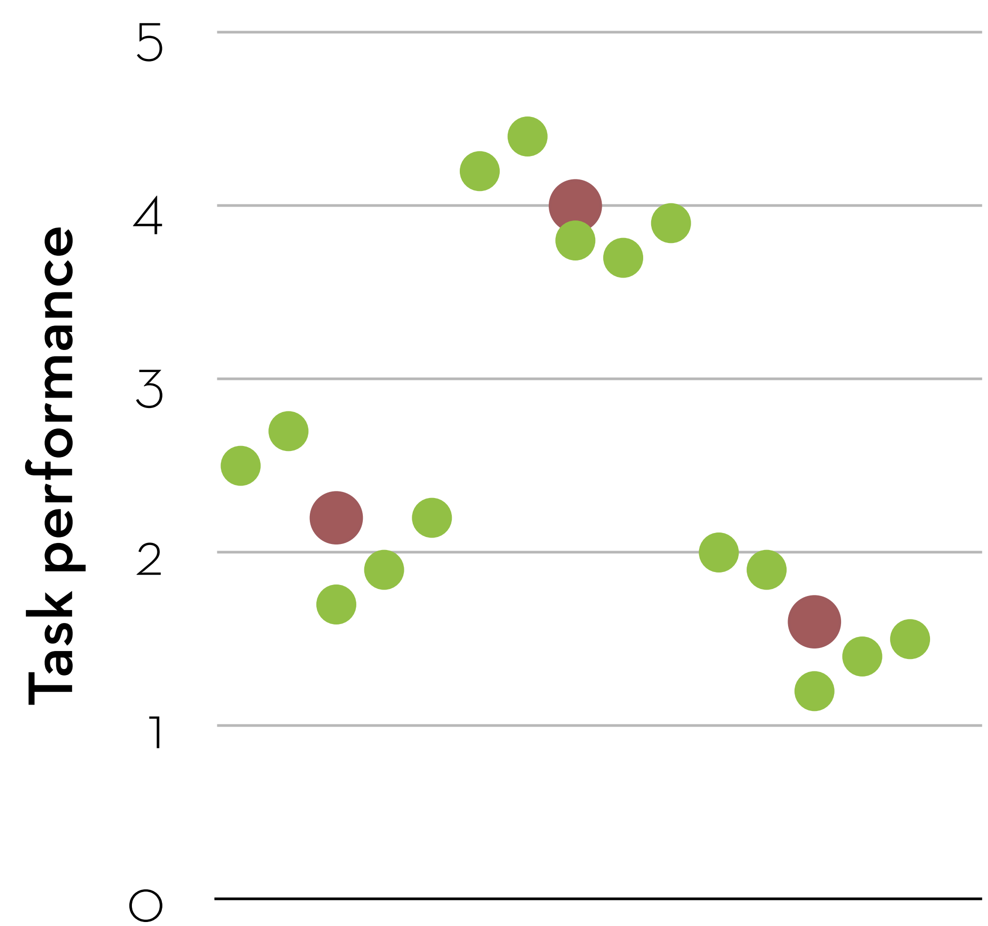
Golden rule: data-points should be **independent**



OK solution...

Take the average of the repeated measurements

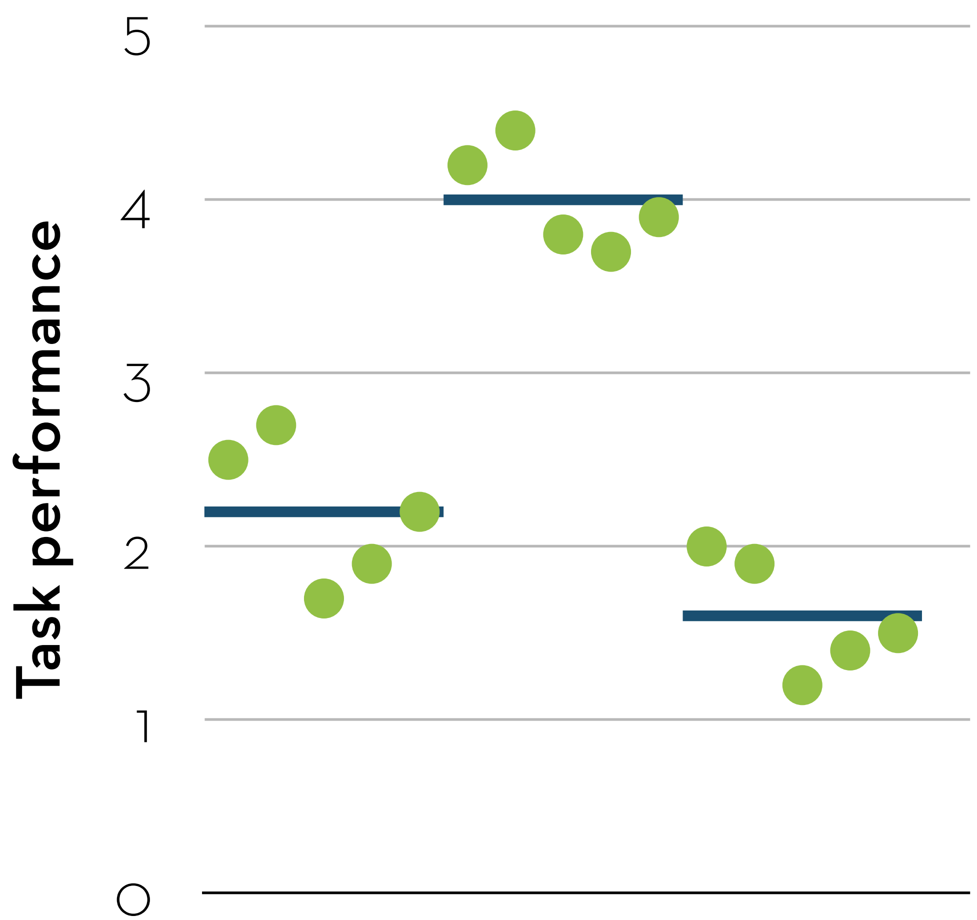
- Reduces the number of observations
- It becomes impossible to make inferences about individual tasks/users/etc.



Good solution

Two approaches:

- define a random intercept for each user (GLMM)
- impose an error covariance structure (GEE)



GEE-like SEM

Under VARIABLE:

Specify id variable (cluster = userid)

Under ANALYSIS:

Specify complex model (type = complex)

GEE-like SEM

Advantages:

Simple specification, works just like regular SEM

Disadvantages:

Only two levels; no random slopes or double intercepts

GLMM-like SEM

Under VARIABLE:

- Specify within-subjects variables (within = a b c)
- Specify between-subjects variables (between = x y z)
- Specify id variable (cluster = userid)

Under ANALYSIS:

- Specify two-level model (type = twolevel)

Under MODEL:

- Specify %within% and %between% effects

GLMM-like SEM

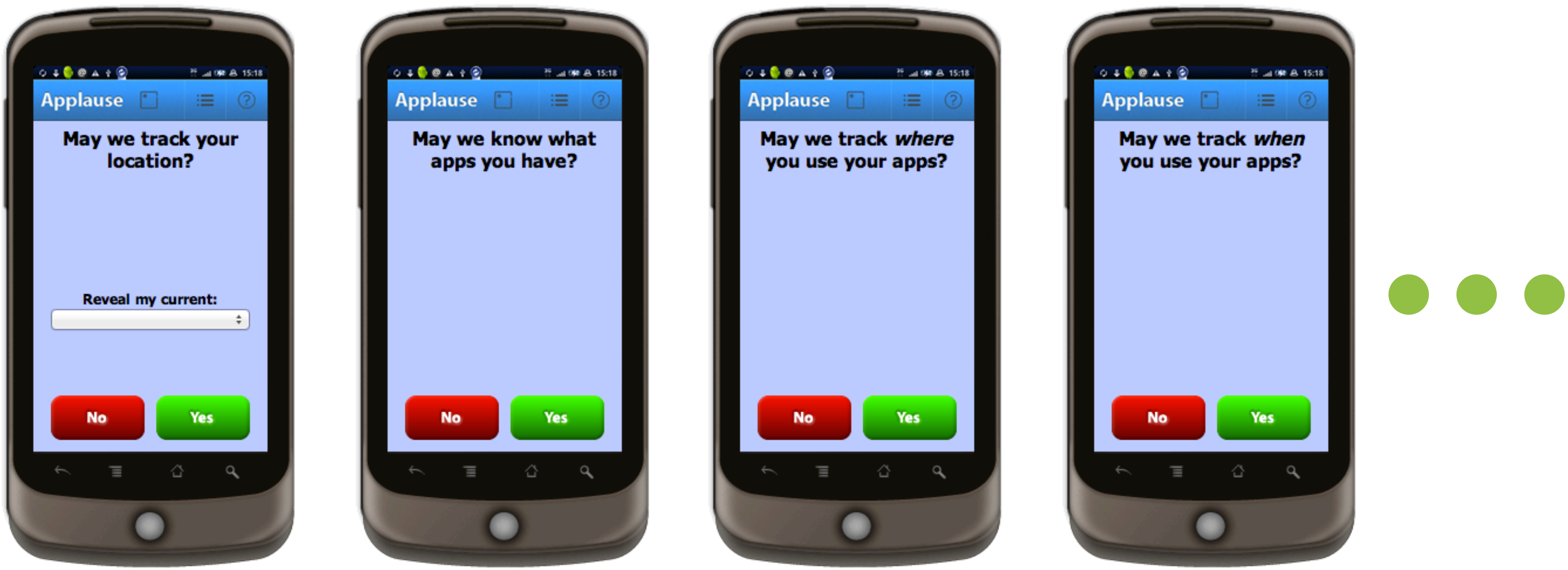
Advantages:

- Can do more than two levels ("threelevel"), and even combine with GEE ("twolevel complex")
- Does intercepts; also random slopes ("twolevel random")
- The random slope can be a dependent variable in another regression (cross-level interactions)

Disadvantages:

- Cannot use categorical indicators
- Can take a long time to estimate (especially "random")

Example



Applause
May we track your location?
Reveal my current:
No
Yes
Applause
May we know what apps you have?
No
Yes
Applause
May we track *where* you use your apps?
No
Yes
Applause
May we track *when* you use your apps?
No
Yes

Example

Context data first

Demographical data first

Example

5 justification types

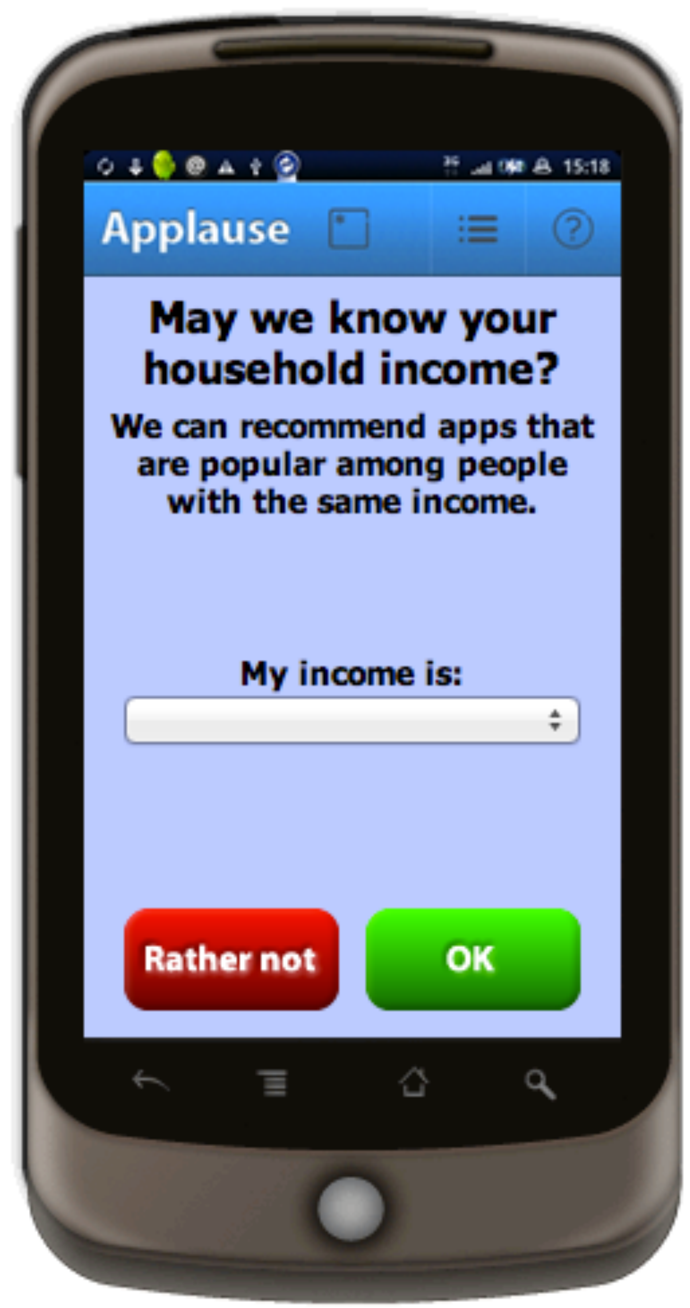
None

Useful for you

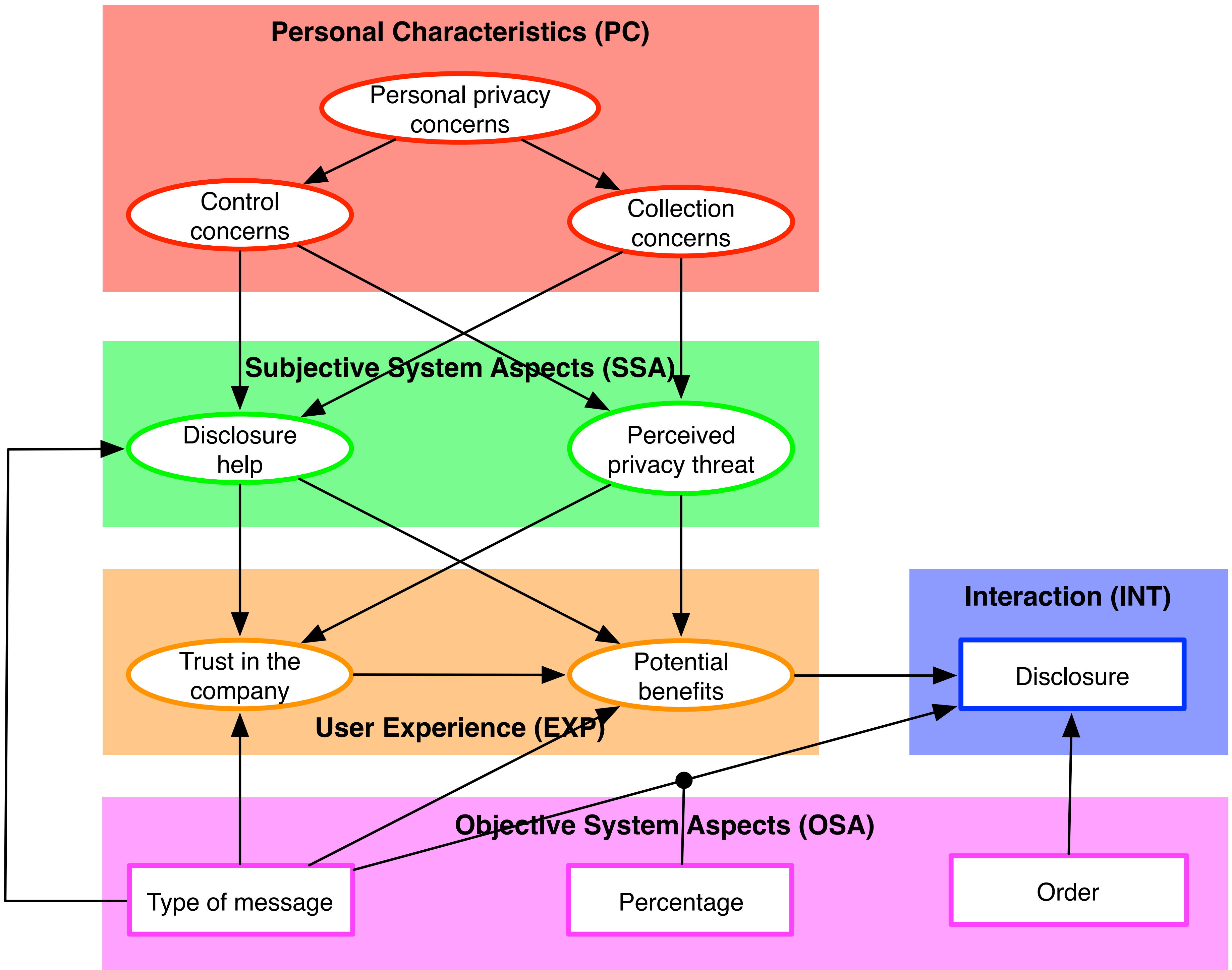
Number of others

Useful for others

Explanation



Example



Personal Characteristics (PC)
Personal privacy concerns
Control concerns
Collection concerns
Subjective System Aspects (SSA)
Disclosure help
Perceived privacy threat
Trust in the company
Potential benefits
User Experience (EXP)
Interaction (INT)
Disclosure
Objective System Aspects (OSA)
Type of message
Percentage
Order

Learn more?

Learn it yourself:

- MPlus course videos (topics 7 and 8)

Interaction effects

in SEM

Interaction effects

What is the combined effect of x1 and x2 on y?

Possibilities:

- **Additive effect**
- Super-additive effect
- Sub-additive effect
- Cross-over

	x1 = low	x1 = high
x2 = low	0	5
x2 = high	5	10

Interaction effects

What is the combined effect of x1 and x2 on y?

Possibilities:

- Additive effect
- **Super-additive effect**
- Sub-additive effect
- Cross-over

	x1 = low	**x1 = high**
x2 = low	0	5
x2 = high	5	**15**

Interaction effects

What is the combined effect of x1 and x2 on y?

Possibilities:

- Additive effect
- Super-additive effect
- **Sub-additive effect**
- Cross-over

	x1 = low	**x1 = high**
x2 = low	0	5
x2 = high	5	**5**

Interaction effects

What is the combined effect of x1 and x2 on y?

Possibilities:

- Additive effect
- Super-additive effect
- Sub-additive effect
- **Cross-over**

	x1 = low	**x1 = high**
x2 = low	0	5
x2 = high	5	**0**

Model specification

This is easy in regressions

Just multiply the dependent variables!

y ~ x1*x2

More difficult in SEM

Depends on type of variables:

manipulation * manipulation

manipulation * factor

factor * factor

Model specification

manipulation * manipulation is easy:

- Just create the dummies!
- See SEM slides for an example

manipulation * factor:

- Multiple groups model or predicted random slopes model

factor * factor:

- Predicted random slopes model

Two approaches

"Predicted random slopes model"

Pro: Works for all types of variables

Con: Cannot use categorical indicators

Con: Can take a long time to estimate

"Multiple groups model"

Pro: Easier to estimate

Pro: Can sometimes use categorical indicators*

Con: Does not work for factor * factor interactions

Random slopes

Under ANALYSIS:

Specify random slopes (type = random)

Specify integration (algorithm = integration)

Under MODEL:

Specify the moderated effect as random: s | y on x;

Regress the slope on the moderator: s on m;

Add main effect of moderator: y on m;

Factor * factor

Example: is the effect of perceived control on perceived recommendation quality dependent on understandability?

In regression terms:

quality ~ control*underst

In SEM:

s | quality ON control;

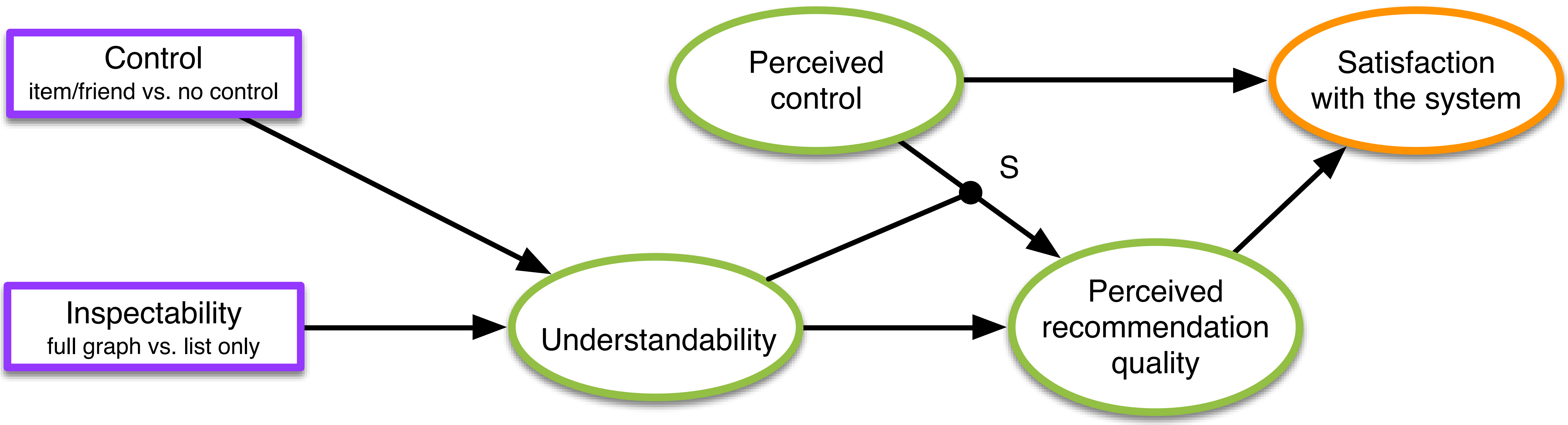
s ON underst;

quality ON underst;

Factor * factor

```
ANALYSIS:
 type = random;
 algorithm = integration;

MODEL:
 satisf BY s1* s2-s7;
 quality BY q1* q2-q6;
 control BY c1* c2-c4;
 underst BY u2* u4-u5;
 satisf-underst@1;

 satisf ON quality control;
 s | quality ON control;
 s ON underst;
 quality ON underst;
 underst ON citem cfriend cgraph;
```

Factor * factor



Factor * condition

Example: is the effect of perceived control on perceived recommendation quality dependent on the control condition?

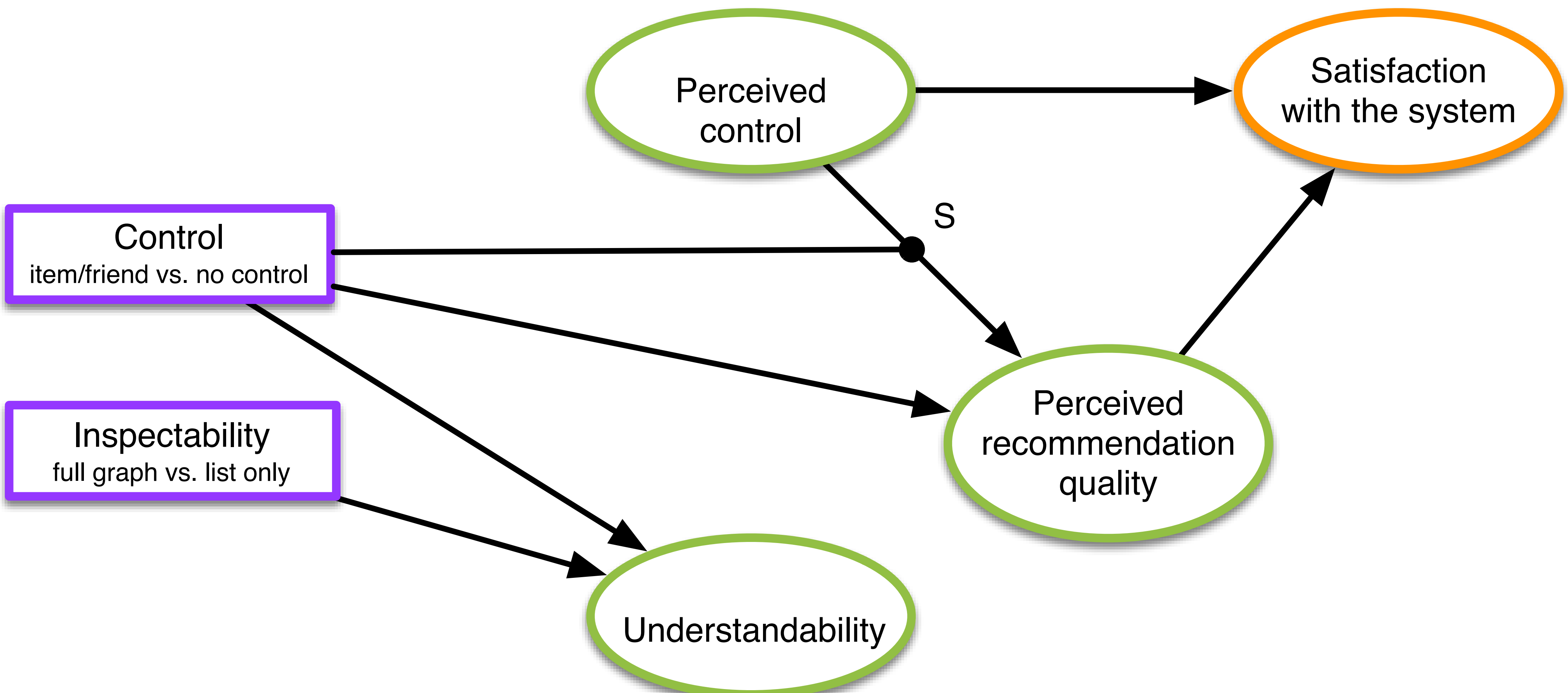
In SEM:

```
s | quality ON control;
s ON citem cfriend;
quality ON citem cfriend;
```

Factor * condition

```
ANALYSIS:
 type = random;
 algorithm = integration;

MODEL:
 satisf BY s1* s2-s7;
 quality BY q1* q2-q6;
 control BY c1* c2-c4;
 underst BY u2* u4-u5;
 satisf-underst@1;

 satisf ON quality control;
 s | quality ON control;
 s ON citem cfriend;
 quality ON citem cfriend;
 underst ON citem cfriend cgraph;
```

Factor * condition



Multiple groups

Under VARIABLE:

- Specify the moderating manipulation as a “grouping” variable: grouping = cctrl(0=none 1=item 2=friend)

Add a MODEL section for all groups except the baseline

- Model item:
- Model friend:

Add corresponding labels to each MODEL to restrict the moderation

Factor * condition

```
MODEL:
 satisf BY s1* s2-s7;
 quality BY q1* q2-q6;
 control BY c1* c2-c4;
 underst BY u2* u4-u5;
 satisf-underst@1;

 satisf ON quality control (1-2);
 quality ON control (p1);
 control ON underst (4);
 underst ON cgraph (5);

 [satisf] (6);
 [quality] (7);
 [control] (8);
 [underst];
```

```
MODEL item:
 satisf ON quality control (1-2);
 quality ON control (p2);
 control ON underst (4);
 underst ON cgraph (5);

 [satisf] (6);
 [quality] (7);
 [control] (8);
 [underst];

MODEL friend:
 satisf ON quality control (1-2);
 quality ON control (p3);
 control ON underst (4);
 underst ON cgraph (5);

 [satisf] (6);
 [quality] (7);
 [control] (8);
 [underst];
```

Learn more?

Learn it yourself:

- Difficult... MPlus course videos do not cover this explicitly

Cluster Analysis

using Latent Categorical Analysis and
Mixture Factor Analysis

Cluster Analysis

Putting people into distinct groups…

…based on how they answer certain questions

…based on behavioral patterns

…etc

Two versions:

Based on "raw data": Latent Categorical Analysis

Based on factors: Mixture Factor Analysis

Dataset

ID	Items
1	Wall
2	Status updates
3	Shared links
4	Notes
5	Photos
6	Hometown
7	Location (city)
8	Location (state/province)
9	Residence (street address)
10	~~Employer~~
11	Phone number
12	Email address
13	Religious views
14	Interests (favorite movies, etc.)
15	Facebook groups
16	~~Friend list~~

LCA

Under VARIABLE:

Specify the number of classes: classes = c(2)

Under ANALYSIS:

Specify mixture model: type = mixture

Optionally, specify iterations etc

MFA

Under VARIABLE:

Specify the number of classes: classes = c(2)

Under ANALYSIS:

Specify mixture model: type = mixture

Optionally, specify iterations etc (often needed!)

Under MODEL:

Add %overall% and then the factor model

Prepare to wait :-)

How many classes?

Balance the following criteria

- Minimum of BIC
- Maximum entropy
- Loglikelihood levels off
- p-value of successor > .05 (use Lo-Mendell-Rubin adjusted LRT test, available in output: tech4)
- **Solution makes sense**

Results

Table 9
A comparison of the fit of MFA models with different numbers of classes.

	BIC	Entropy	LL	# of par.	*p*-Value
1 class	16,837		−8277.147	48	
2 classes	16,578	0.973	−8133.179	53	0.0069
3 classes	**16,442**	**0.998**	−8050.552	58	0.0002
4 classes	16,468	0.998	−8048.736	63	**0.407**
5 classes	16,482	0.878	−8041.459	68	0.999
6 classes	**16,351**	0.897	−7960.902	73	0.812
7 classes	16,359	0.852	−7950.412	78	0.893

The bold values are mentioned in the text as indicators of the optimal number of dimensions.

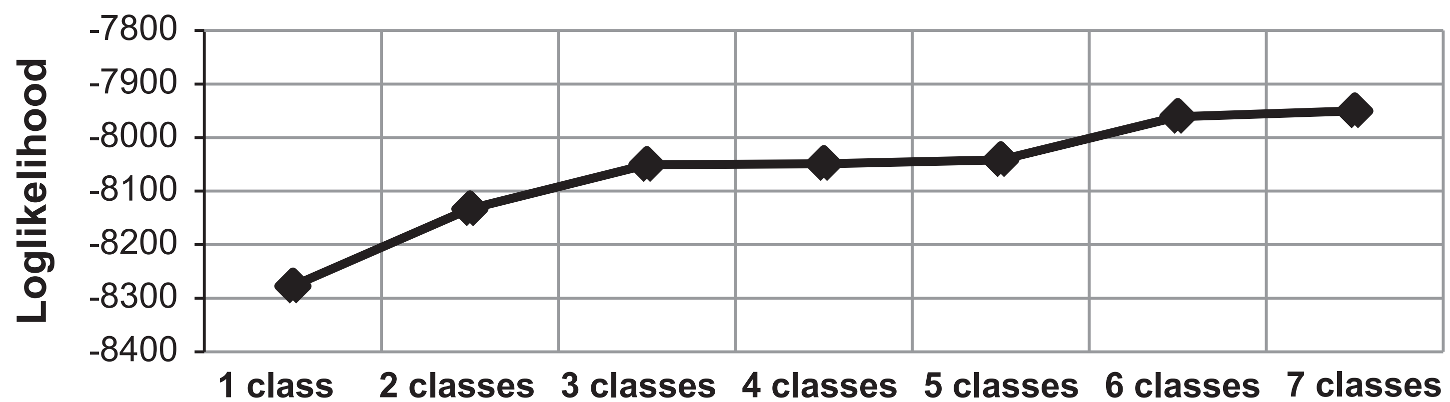
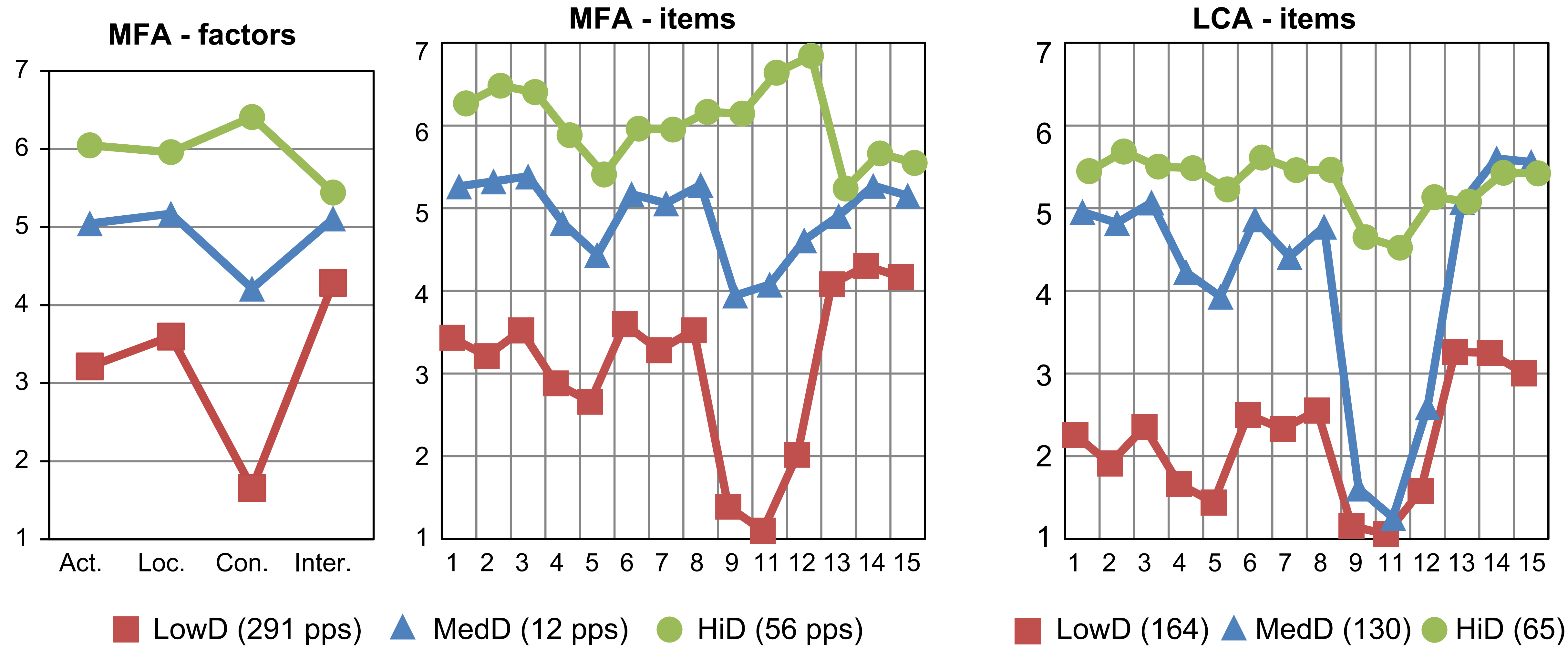


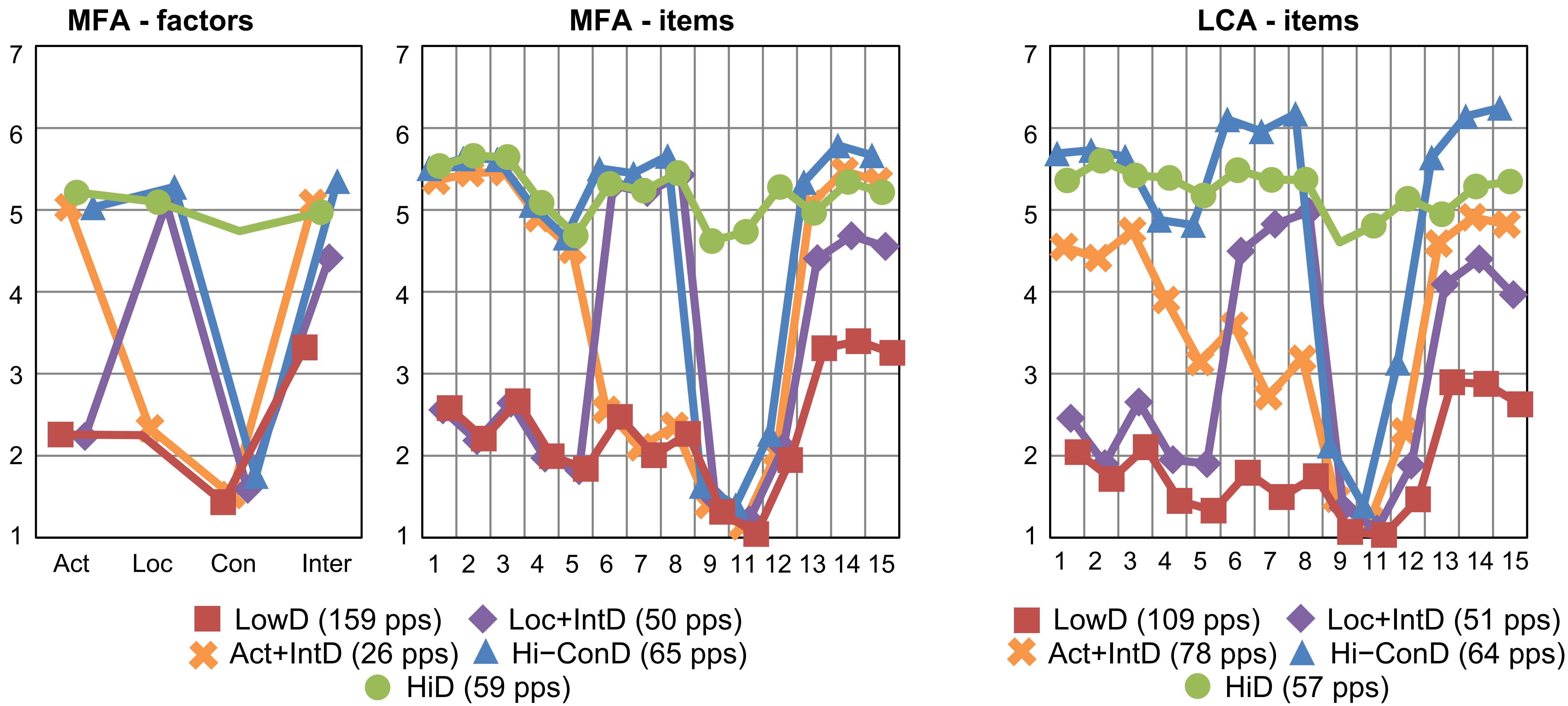
Fig. 8. Change in loglikelihood between subsequent MFA models.

Results



From: Knijnenburg et al. (2012): "Dimensionality of information disclosure behavior ", *IJHCS 71(12)*

Results



MFA - factors
MFA - items
LCA - items
7
6
5
4
3
2
1
Act
Loc
Con
Inter
1 2 3 4 5 6 7 8 9 11 12 13 14 15
LowD (159 pps)
Loc+IntD (50 pps)
Act+IntD (26 pps)
Hi−ConD (65 pps)
HiD (59 pps)
LowD (109 pps)
Loc+IntD (51 pps)
Act+IntD (78 pps)
Hi−ConD (64 pps)
HiD (57 pps)

Learn more?

Learn it yourself:

MPlus course videos (topic 5)

"It is the mark of a truly intelligent person to be moved by statistics."

T H A N K S !

George Bernard Shaw