SEM part I

Structural Equation Modeling

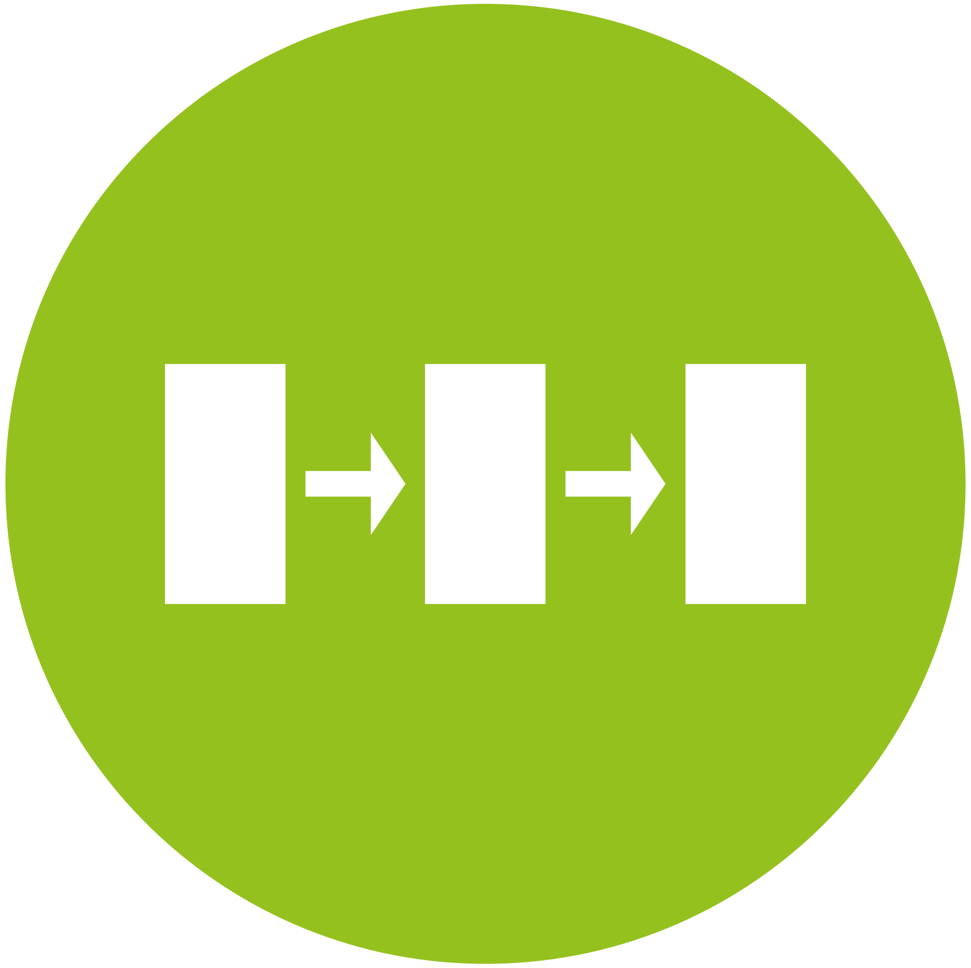
Intro

Today's goal:

Teach the idea behind Structural Equation Modeling, and already some practice as well.

Outline:

- Rationale behind SEM
- Testing marginal effect models



Why SEM?

Testing many mediations, with power!

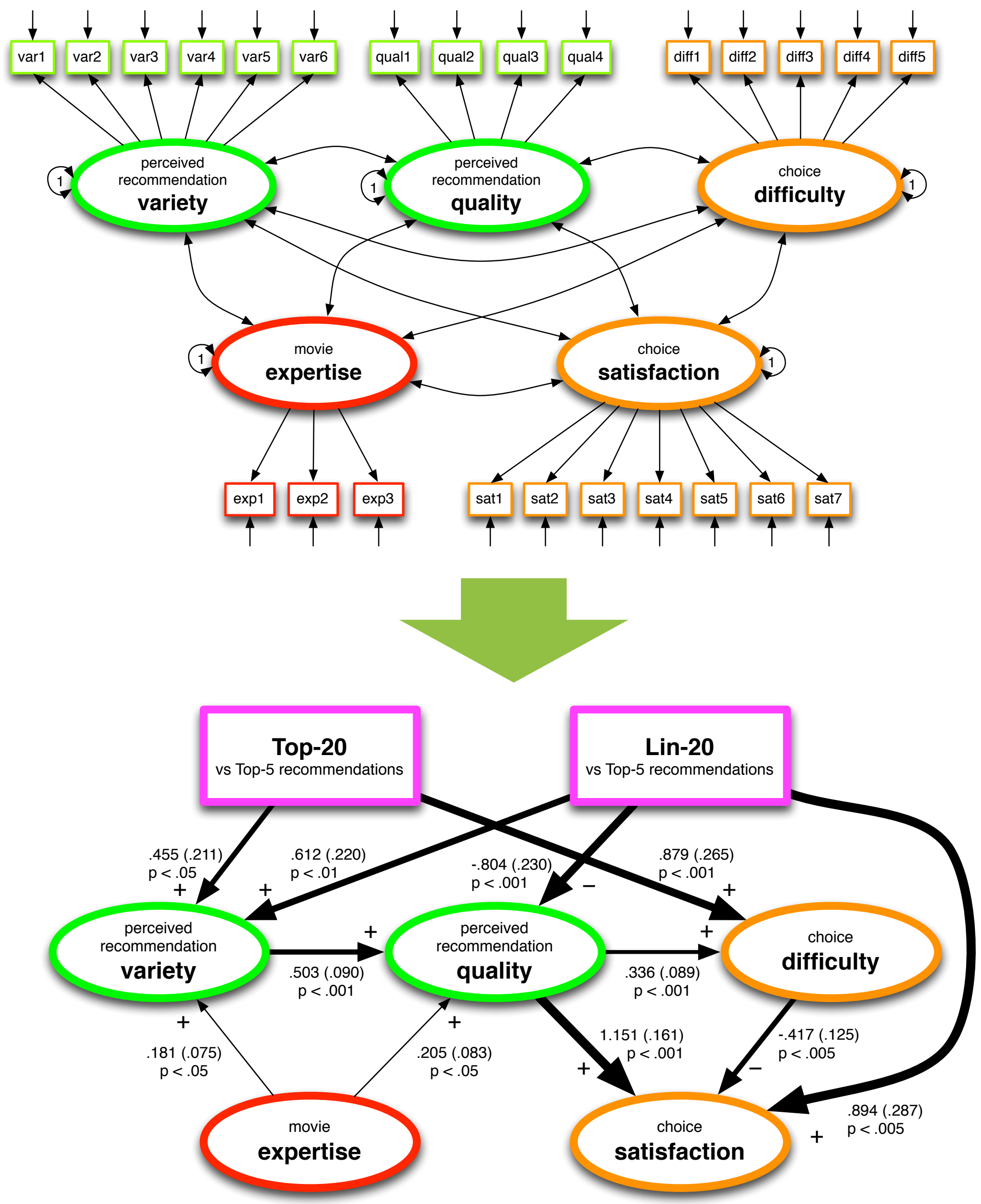
Why SEM?

Combine **factor analysis** and **path models**

- Turn items into factors
- Test causal relations

Very **simple reporting**

- Report overall fit + effect of each causal relation
- A path that explains the effects



Why SEM?

Two advantages:

1. Path models allow for **simple mediation analysis**

   All paths are tested at once

2. Factor models allow for **more precise tests**

   Knowledge about scale reliability is taken into account

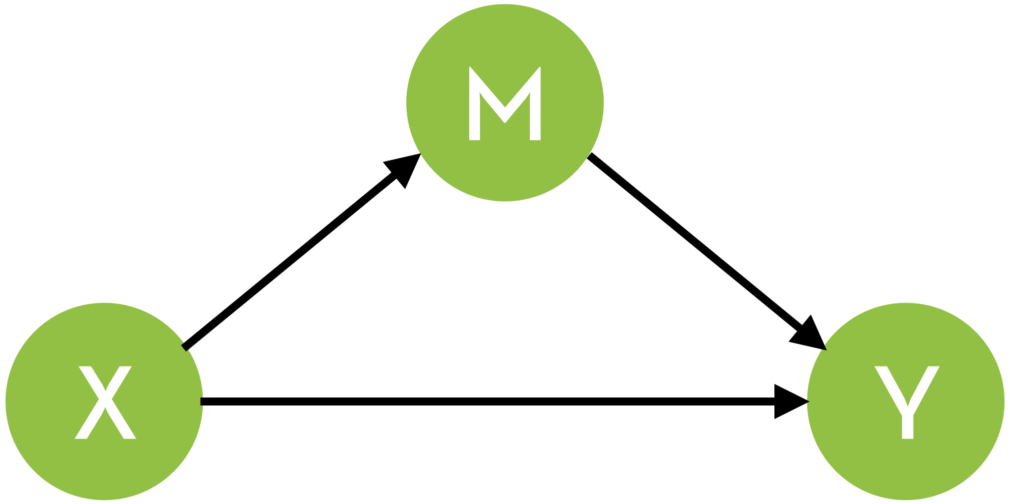
Mediation Analysis

X -> M -> Y

- Does the system (X) influence usability (Y) via understandability (M)?

Types of mediation

- Partial mediation
- Full mediation
- Negative mediation



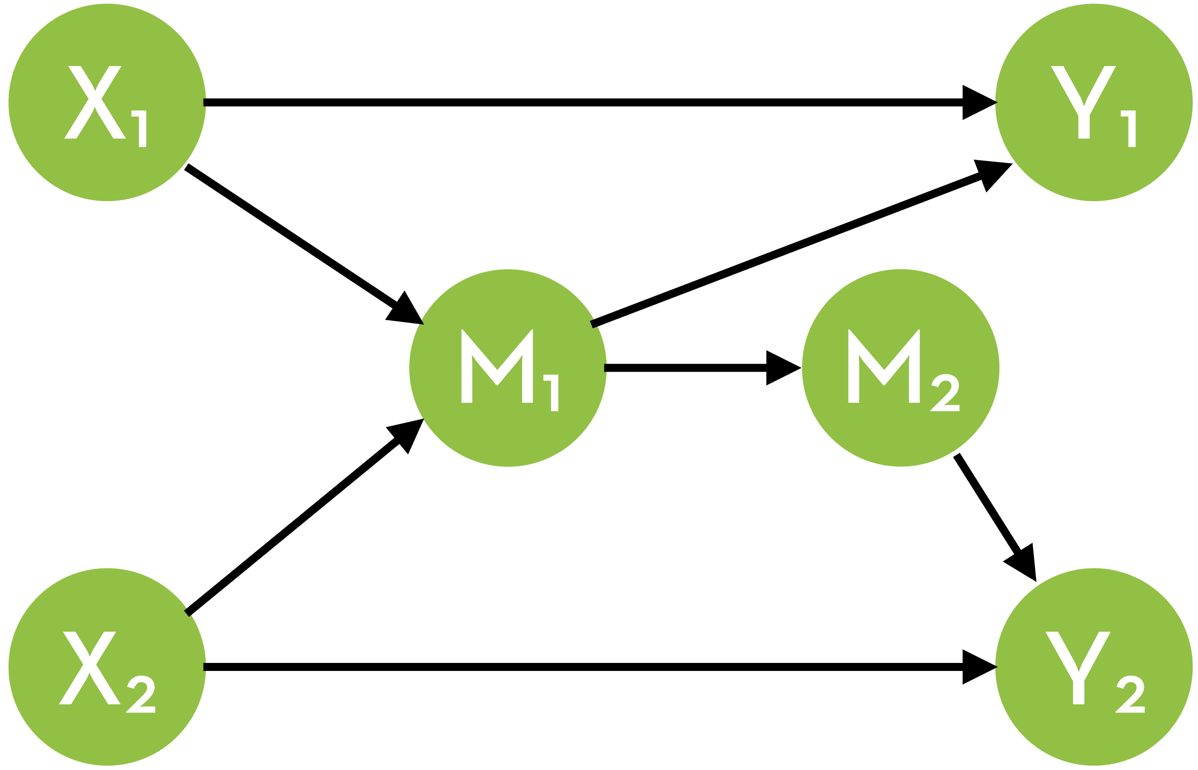
Problems...

Mediation Analysis is a lot of work

- Many tests to conduct
- Many findings to report

Gets even more complicated with more "interesting" models

No "overall" test of the model



Solution: SEM

Tests all mediations at once

Gives you overall model fit statistics

Allows you to find out easily if a certain mediation is full or partial

There is an option to calculate total effects

- A bit difficult in R, but not impossible

More precise tests

Let's say we have a factor F measuring trait Y, with AVE = 0.64

- On average, 64% of the item variance is communality, 36% is uniqueness

If we **sum the items** of the factor as S, this results in 36% error

- This is random noise that does not measure Y

Result: no regression with S as dependent can have an R-squared > 0.64!

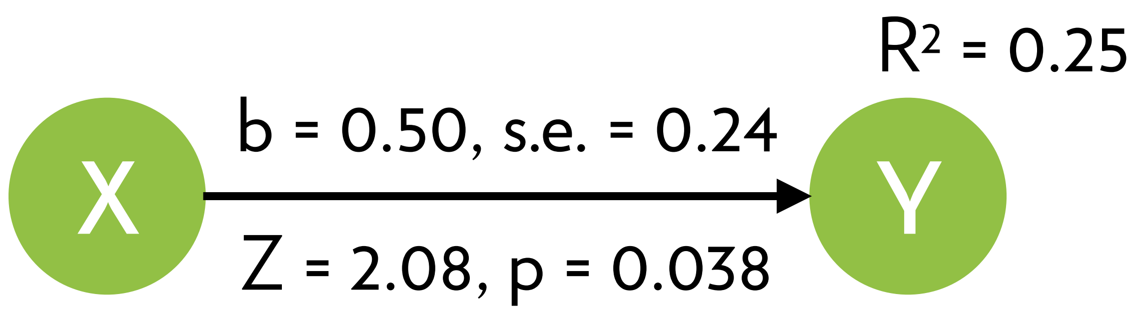
Sum score...

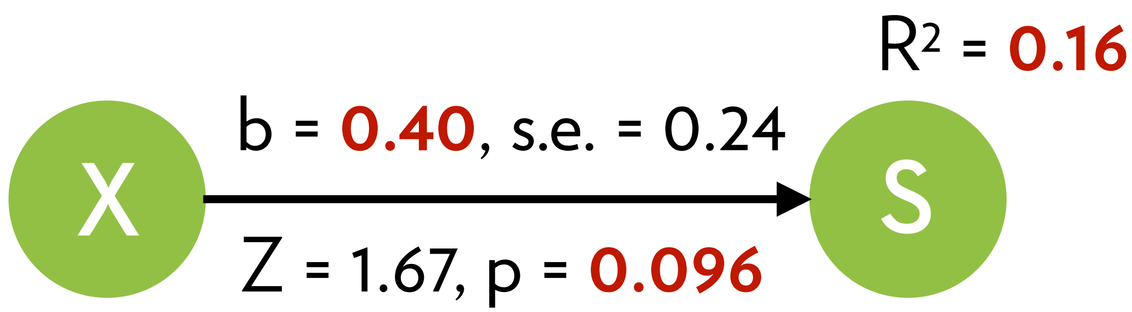
Any regression coefficient will be **attenuated** by the AVE of S!

Take for instance this X, which potentially explains 25% of the variance of Y...

...it only explains 16% of the variance of S!

...and the effect is non-significant!





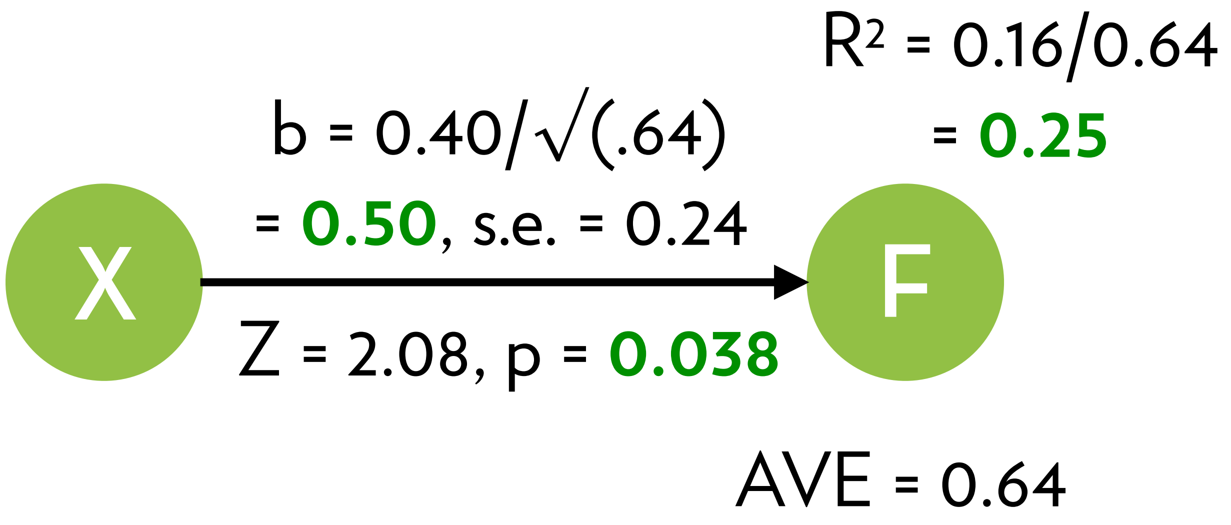
Solution: SEM

In SEM, we keep the factors!

If we use F instead of S, we **know** that the AVE is 0.64

> …so we can **compensate** for the incurred measurement error

This leads to much more precise tests



Estimates

In a SEM you can get the following estimates (all at once):

- Item loadings and communality
- Factor fit statistics (AVE)
- R^2 for every dependent variable
- Regression coefficients for all regressions (B, s.e., p-values)
- ANOVA-like tests for manipulations with > 2 conditions (but you need to manually create dummies)
- Total (mediated and non-mediated) effects*

Fit statistics

Same fit statistics as in CFA. As a reminder:

- Item-fit: Loadings, communality, modification indices
- Factor-fit: Average Variance Extracted
- Model-fit: Chi-square test, CFI, TLI, RMSEA

Also: modification indices for model improvement purposes

- Not just for items/factors, but also for regression coefficients!

What else?

Any variable can be binary (logistic regression), ordered categorical, or count (Poisson regression)

You can do multilevel linear models (LME)

- There is an easy method and a hard method (the latter can estimate random intercepts and slopes)

You can even combine the two (GLME), but unfortunately not in lavaan…

- You can use MPlus if you encounter this situation

What else?

You can do interaction effects, to some extent:

manipulation * manipulation is easy

Just create the correct dummies

manipulation * factor is harder

Can be done with a "multiple group model" or with a "predicted random slopes model"

factor * factor is even harder

Can only be done with a "predicted random slopes model"

What else?

A note on interactions:

A "Predicted random slopes model"...

...cannot use categorical indicators

... and can take a long time to estimate

A "Multiple groups model"...

...can only sometimes use categorical indicators

...does not work for factor * factor interactions

Tip: try to avoid interactions involving a factor!

Testing marginal effects

Using MIMIC models

Marginal effects

First analysis: manipulations —> factors

MIMIC model (Multiple Indicators, Multiple Causes)

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

Steps involved:

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

Create dummies

Main effects:

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

`citem cfriend`

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

`cgraph`

Interaction effects:

citem*cgraph and cfriend*cgraph

`cig cfg`

Create your CFA

Take the final CFA

```
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…

Add regression

Add a regression to your final CFA model:

```
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 ~ p1*citem+p2*cfriend+cgraph+p3*cig+p4*cfg’;

fit <-
sem(model,data=twq,ordered=names(twq[9:31]),std.lv=TRUE);

summary(fit);
```

Results

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

```
                         Estimate  Std.err  Z-value  P(>|z|)
...(factors)...               ...      ...      ...      ...
Regressions:
  satisf ~
    citem     (p1)      0.269    0.234    1.153    0.249
    cfriend   (p2)      0.197    0.223    0.882    0.378
    cgraph              0.375    0.221    1.694    0.090
    cig       (p3)     -0.131    0.320   -0.408    0.683
    cfg       (p4)     -0.048    0.309   -0.156    0.876
```

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)

Results

ANOVA of the interaction effect:

```
lavTestWald(fit,'p3==0;p4==0');
```

Result:

```
$stat
[1] 0.1695068

$df
[1] 2

$p.value
[1] 0.9187388
```

Result: $\chi^2(2) = 0.170$, $p = .919$

Results

ANOVA of the main effect of control:

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

Result:

```
$stat
[1] 1.453119

$df
[1] 2

$p.value
[1] 0.4835699
```

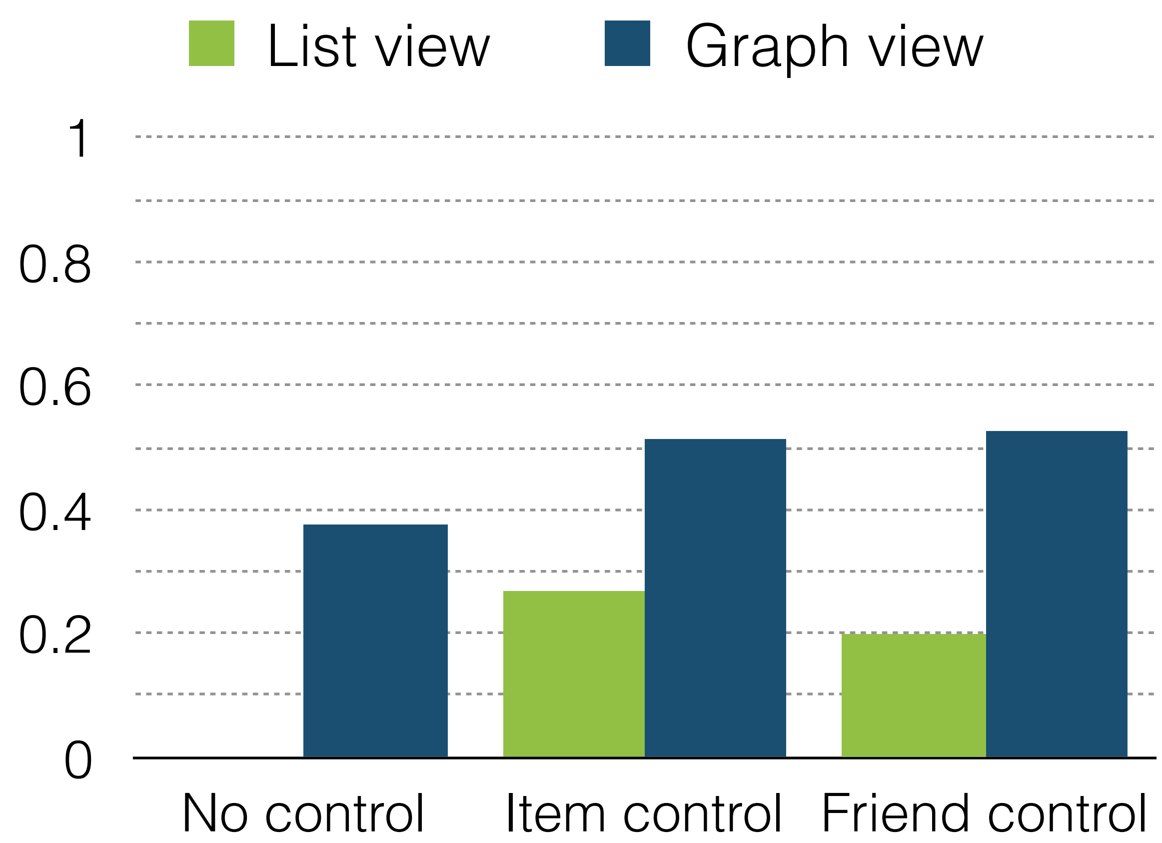
Result: $\chi^2(2) = 1.453$, $p = .484$

Graph

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

Other values: calculate the model

e.g. Friend control, graph view: cfriend + cgraph + cfg



Repeat

Repeat this process for

- quality
- control
- underst

Main finding

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

Similar to homework 1!

```
                           Estimate  Std.err  Z-value  P(>|z|)
...(factors)...                 ...      ...      ...      ...
Regressions:
  underst ~
    citem      (p1)       0.367    0.220    1.666    0.096
    cfriend    (p2)       0.534    0.216    2.466    0.014
    cgraph                0.556    0.227    2.450    0.014
    cig        (p3)      -0.105    0.326   -0.323    0.746
    cfg        (p4)      -0.178    0.320   -0.555    0.579
```

Main finding

ANOVA of the main effect of control:

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

Result:

```
$stat
[1] 6.376967

$df
[1] 2

$p.value
[1] 0.04123435
```

Result: $\chi^2(2) = 6.377, p = .041$

Better graph

To create a better graph, we can use a dummy for each condition (except the baseline):

no control, list view: (baseline condition)

item control, list view: cil

friend control, list view: cfl

no control, graph view: cng

item control, graph view: cig

friend control, graph view: cfg

Better graph

Add a regression to your final CFA model:

```
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 ~ cil+cfl+cng+cig+cfg’;

fit <-
sem(model,data=twq,ordered=names(twq[9:31]),std.lv=TRUE);

summary(fit);
```

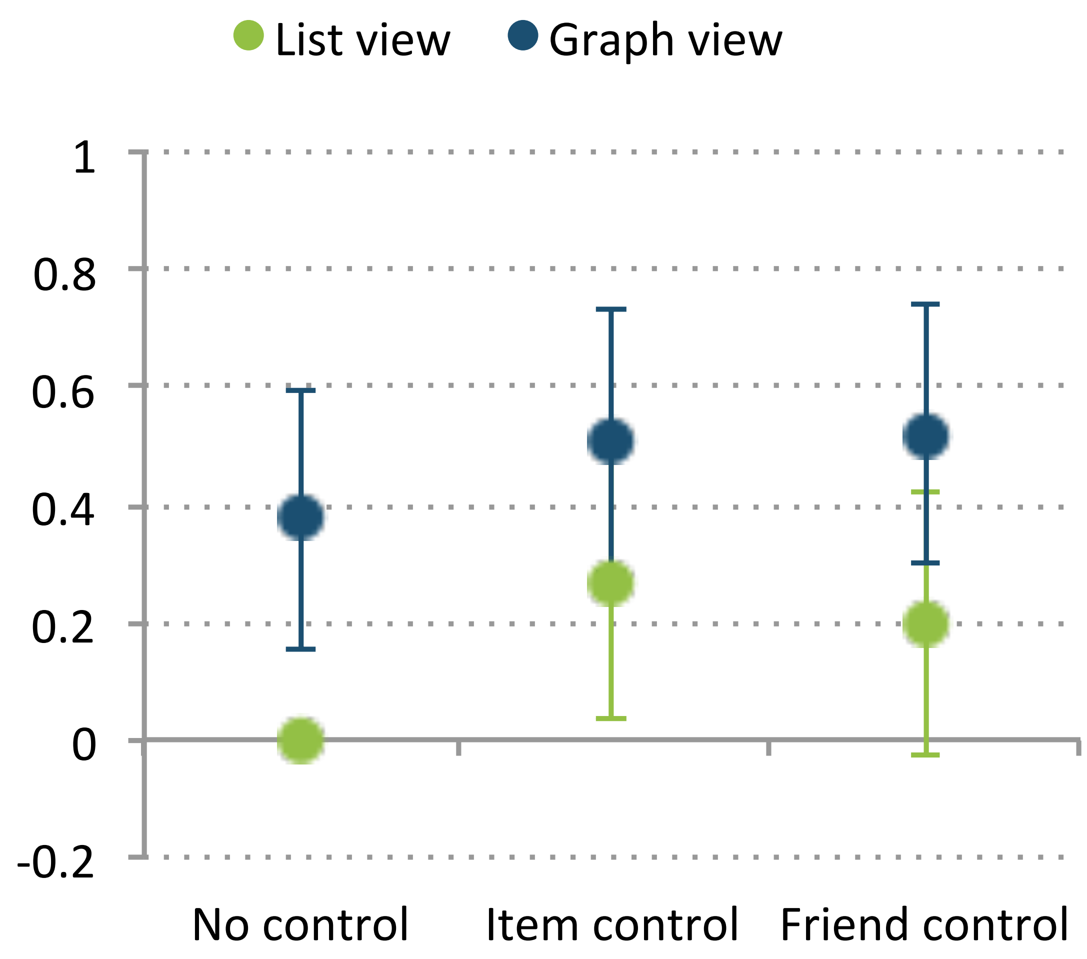
Better graph

```
                        Estimate  Std.err  Z-value  P(>|z|)
...(factors)...            ...       ...      ...      ...
Regressions:
  satisf ~
    cil                   0.270    0.234    1.153    0.249
    cfl                   0.197    0.223    0.882    0.378
    cng                   0.375    0.221    1.694    0.090
    cig                   0.510    0.226    2.259    0.024
    cfg                   0.518    0.221    2.349    0.019
```

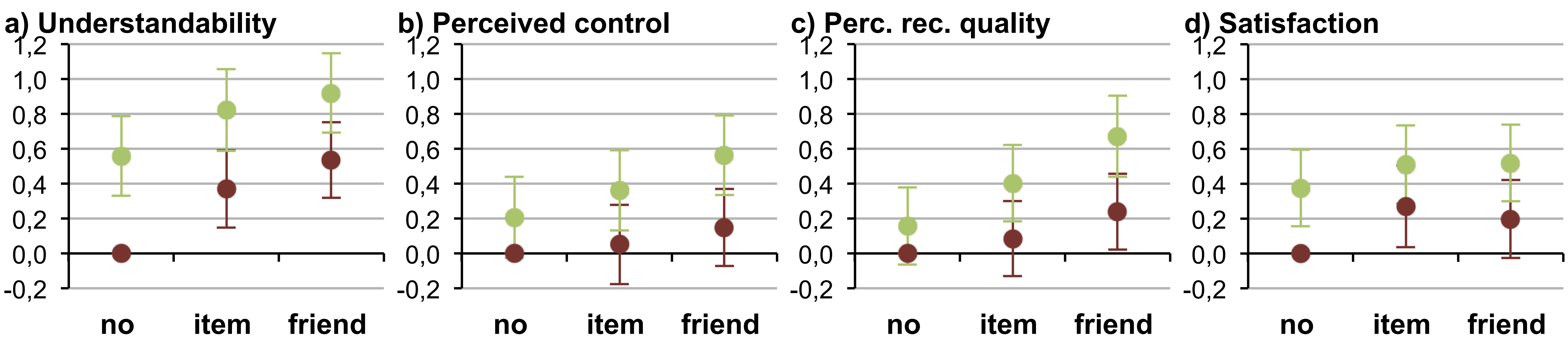
Better graph

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*

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

T H A N K S !

George Bernard Shaw