Conducting User Experiments with Personalized Systems

Bart Knijnenburg

Slides and resources: https://www.usabart.nl/QRMS/

POPROX NEWS

POPROX NEWS

Outline

1. What is a user experiment?
2. Developing a research model
3. Study design and sample size
4. Measurement
5. Evaluating research models

POPROX NEWS

1. What is a user experiment?

What is a user experiment?

User Experiment—Systematically tests how different **system aspects** (manipulations) influence the users' **experience** and **behavior** (observations).

From UMAP's Content Expectations:

- "Evaluations of proposed solutions/applications must be commensurate with the claims made in the paper. Depending on the intended contribution, this may include simulation studies, offline evaluations, A/B tests, or **controlled user experiments**."

Why (not) do an experiment?

Why experiment?

- Establishing causality
- Observe users' interaction and ask questions
- Quantitatively test theories/hypotheses

Why not do it?

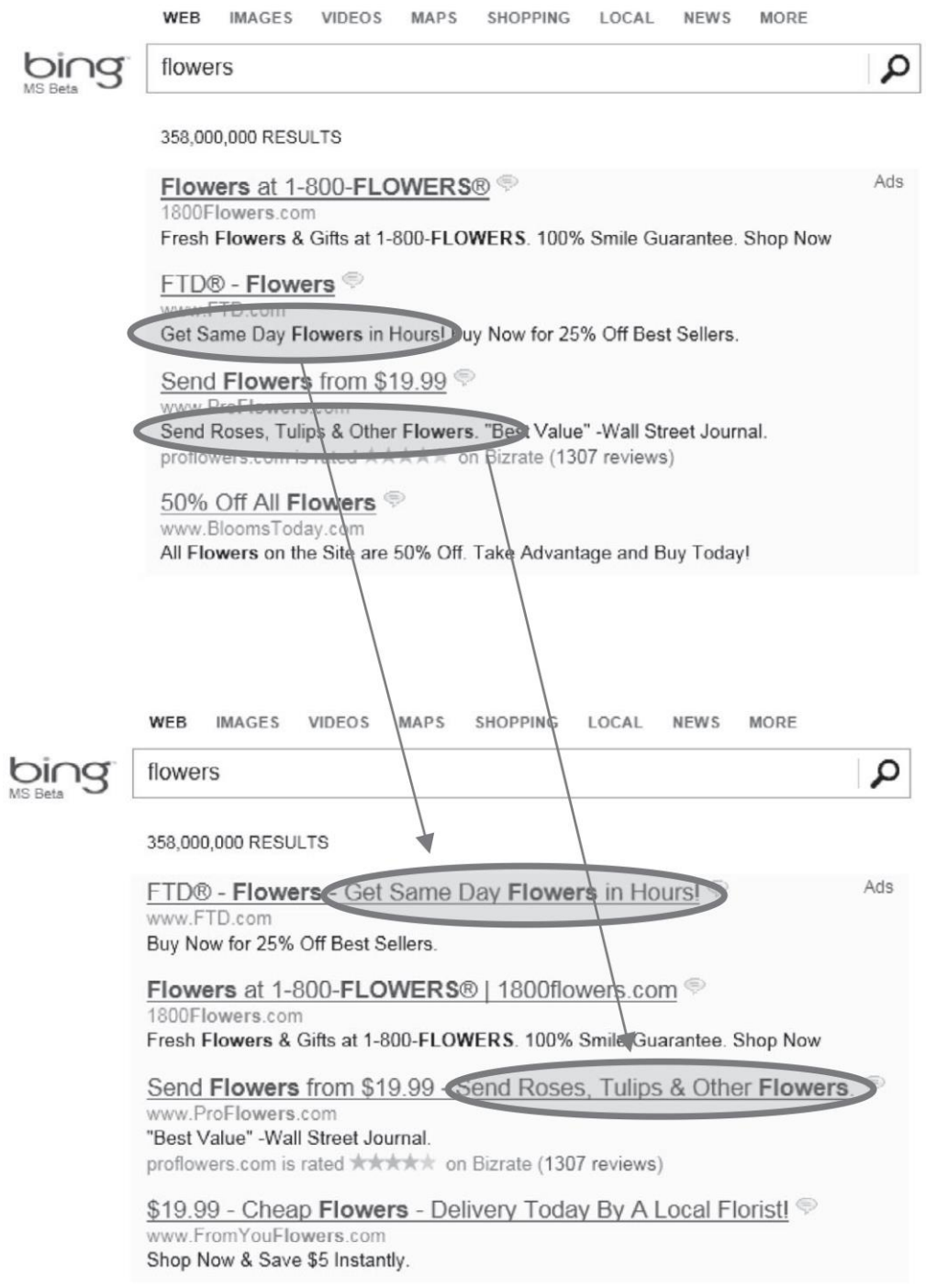
- Running experiments can be expensive and difficult
- Lack of ecological validity
- Possible solution: POPROX!

Bing search example

In 2012 a Bing employee suggested **lengthening the title line** of search ads by combining it with the text from the first line below the title (Kohavi et al. 2020)

Is this a good idea?

Bing experiments showed **12% increase** in clicks, or **$100M increase** annually



Steps for running an experiment

1. Establish hypothesis/hypotheses
2. Select observation(s), system aspect(s) and levels
3. Choose an experimental design
4. Determine sample size
5. Select participants and randomly assign them to treatments
6. Conduct the experiment
7. Analyze the data and draw conclusions

Exercise – research topic

What are your current research interests?

- What are you trying to learn about personalized systems?
- What are you trying to learn about people?
- …about their interaction?

Is a user experiment the best way of studying this topic?

Other options: offline testing, focus groups, interviews, surveys, diary studies, …

POPROX NEWS

2. Developing a research model

Establish hypothesis (or hypotheses)

Hypothesis: clear, testable statement of how a system aspect / independent variable (X) affects a specific outcome / response variable (Y)

Bad example: “Can you test if my system is **good**?”

Why is this bad?

What does good mean?

- Learnability? (e.g. number of errors?)
- Efficiency? (e.g. time to task completion?)
- Usage satisfaction? (e.g. usability scale?)
- Outcome quality? (e.g. survey?)

We need to define **measures**!

Improvement: “Can you test if my AI system scores **high** on this **usability** scale?”

However…

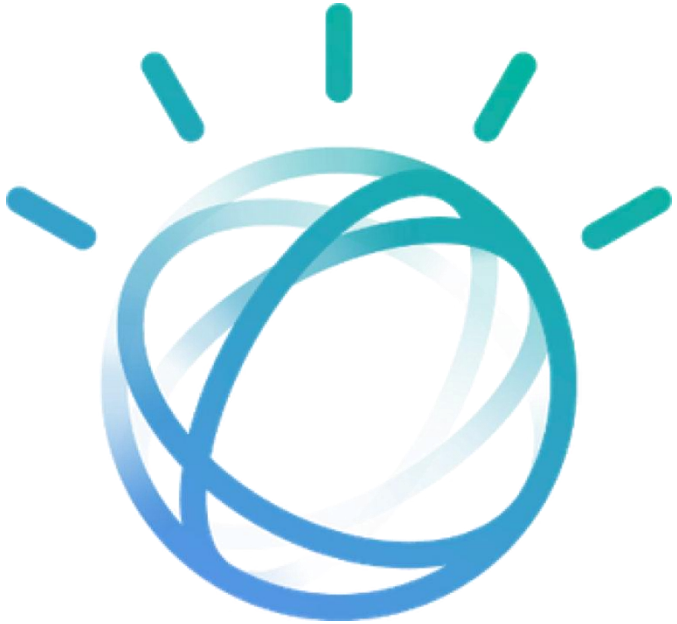
What does high mean?

- Is 3.6 out of 5 on a 5-point scale “high”?
- What are 1 and 5?
- What is the difference between 3.6 and 3.7?

We need to **compare** the UI against something

Improvement: “Can you test if my AI system scores high on this usability scale **compared to this other AI system**?”

Testing A vs. B



System A

System B

However…

Say we find that it scores higher… why does it?

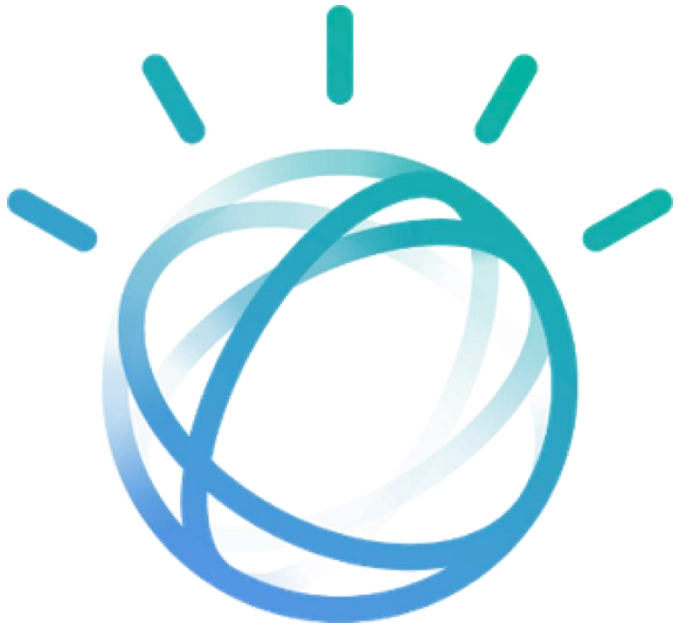
- different skills
- different user models
- different voice

Apply the concept of **ceteris paribus** to get rid of confounding variables

- Keep everything the same, except for the thing you want to test (the manipulation)
- Any difference can be attributed to the manipulation

Ceteris Paribus

New version with **one** added/changed feature



Previous version

System aspects, levels, and observations

The independent variable (X) is a **system aspect** that is manipulated across different **levels**

- In the previous example, the factor is “explanation” and levels are “with” and “without”

Simplest case: manipulate one system aspect with two levels:

- **Control group (CG):** users who receive baseline treatment (no explanations)
- **Treatment group (TG):** users who receive improved treatment (explanations)

The response variable (Y) is a measured **observation** of the consequence of the manipulation (user experience or behavior)

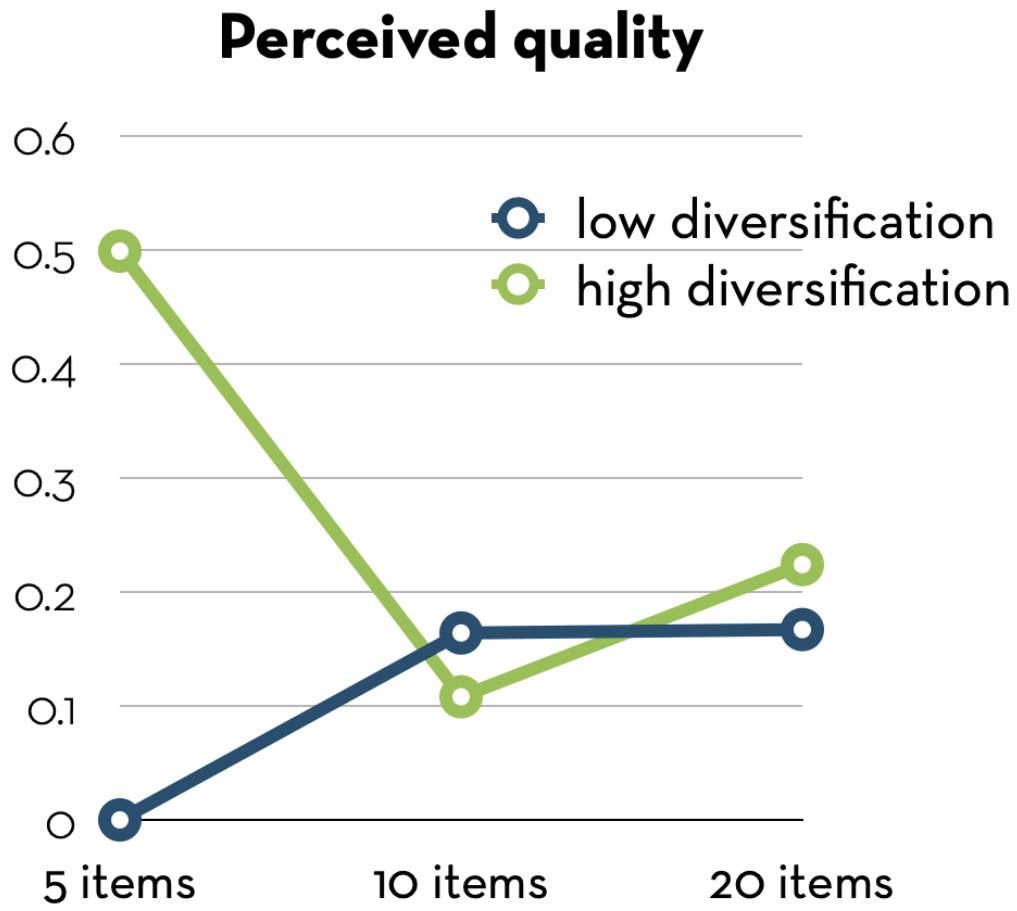
- For the example: “usability score”

On picking manipulations

- Manipulate aspects of the system that you want to test:
  - Aspects of the user model (algorithm, objective function, input data, …)
  - The adaptative behavior (personalized content, layout, …)
  - Other system aspects (interaction mechanisms, explanations, visualizations, …)
- TIP: Study one system aspect at a time, unless you expect them to interact
  - **Interaction:** the effect of one manipulation depends on the level of another manipulation

Interaction effects - example

- Two manipulations at the same time:
  - What is the combined effect of list diversity and list length on perceived recommendation quality?
  - Each combination is a **treatment**
- Test for the interaction effect!
  - However, more treatments = more participants



On picking levels

Always answer “in comparison to what?” by having a baseline control condition or other realistic alternatives

Stronger manipulations (i.e., large differences in levels) will tend to show differences in the outcome but the levels should

- be realistic and practical (explainable RS example)
- cover a range of interest

The more levels you have, the more you fragment your sample (or extend study length in a within-subject design)

On picking levels - example

What's the effect of showing "recent news"?

- Proposed system for treatment group:
  - Filter out any items > 1 month old
- What should my control group see?
  - Filter out items < 1 month old?
  - Unfiltered recommendations?
  - Filter out items > 3 months old?

You should test against a reasonable alternative

"Absence of evidence is not evidence of absence"

Related question: how long should the study run?

Applying theory

To learn something from a study, we need a theory behind the effect

- This makes the work generalizable
- This may suggest future work

Measure **mediating variables**

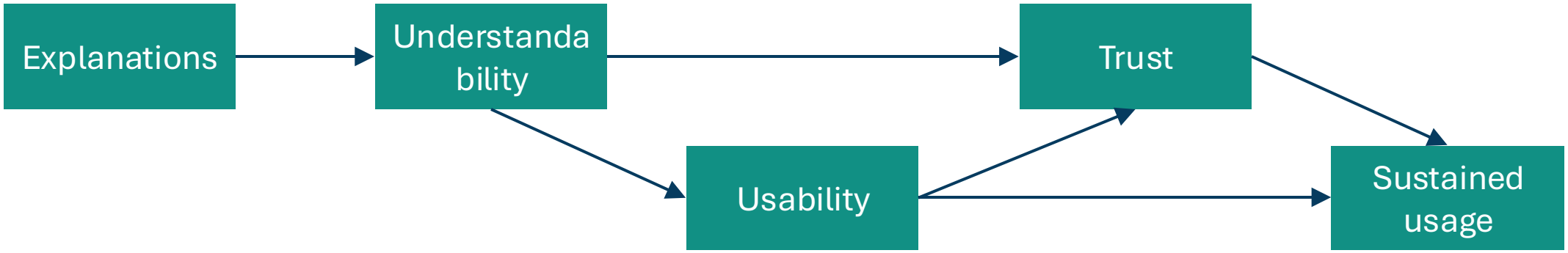
Find out how they mediate the effect on usability

Evaluate the data using path modeling or structural equation modeling

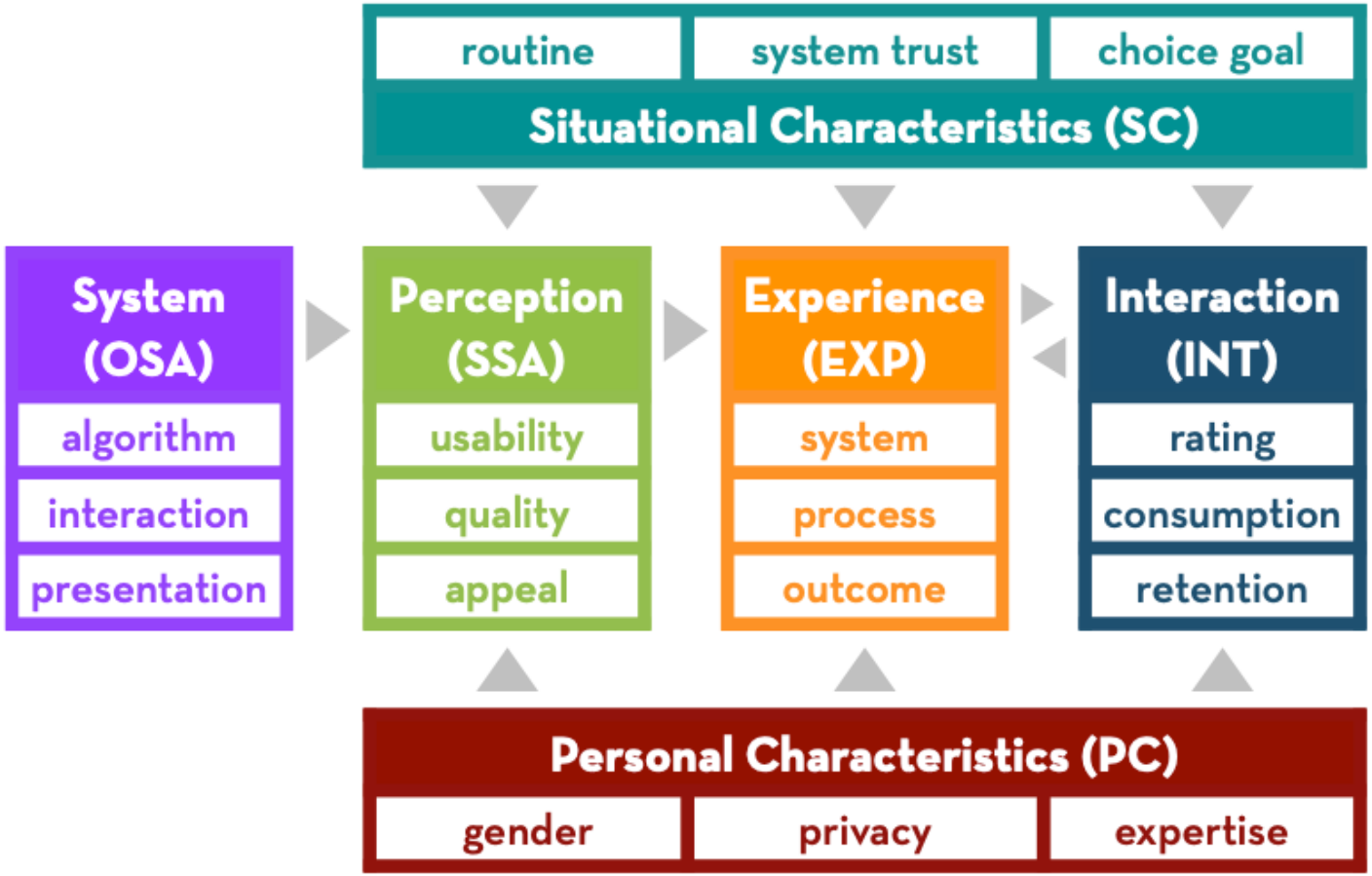
Research model (mediations)

Why would explanations increase system usability?

- Explanations increase understandability
- Understandability adds to an overall perception of usability
- Understandability and usability improve trust
- Trust leads to more sustained usage



Theoretical framework - example



Knijnenburg et al. (2012)
Explaining the User Experience of Recommender Systems

Knijnenburg, & Willemsen (2015)
Evaluating Recommender Systems with User Experiments

Exercise – research model

Develop a research model for an experiment you would like to run:

- Whare are you testing? Specify the manipulation(s)
- What is the expected effect? Determine the main outcome variable(s)
- Why does this effect happen? Think of the mediators
- Put them all together in a research model

Bonus points: ground your research model in an existing theory of human behavior / user experience!

POPROX NEWS

3. Study design and sample size

Between-subjects designs

After-only with control:

TG: (R)	X	O1
CG: (R)		O2

- More than two groups possible
- Downside: Large between-user variation requires a larger sample to detect a difference

Before-after with control:

TG: (R)	O1	X	O2
CG: (R)	O3		O4

- O1 and O3 are **pre-measures**, which estimate user effects
- Simplest example of a **repeated-measures** design
- Downside: Pre-measures may change users' sensitivity to the treatment

Within-subjects designs

(R)	X1	O11	X2	O12	…
(R)	X2	O21	X1	O22	…

- Also called **within-subject design:** each user receives each treatment
- You can estimate user and treatment effects, reducing sample size requirements (with after-only design user effect is confounded with treatment and we rely on randomization). Users are their own controls
- Downside: possible **carryover** effects, where receiving one treatment may affect the user's response to a subsequent treatment.
  - A possible solution is to give respondents the baseline condition for some time before switching treatments
- More than two treatments possible

Which design?

Should I do within-subjects or between-subjects?

- Use **between-subjects** designs for **user experience**
  - Closer to a real-world usage situation
  - No unwanted spill-over effects
- Use **within-subjects** designs for **psychological** research
  - Effects are typically smaller
  - Nice to control between-subjects variability

Example: unexpected confounds

Idea: run a news recommendation study, between subjects

- Start control group on October 28, 2024
- Start treatment group on November 11, 2024

What's the problem here?

Solution: Randomize the assignment of conditions to participants

Randomization neutralizes (but doesn't eliminate) participant variation

Determine sample size - example

- Do married men weigh more than single men?
  - Find 4 married men: Nm = 4, Meanm = 182, SDm = 15
  - Find 4 single men: Ns = 4, Means = 170, SDs = 15
- Effect size: 12 lbs
  - Is this a large effect? —> Need to standardize it!
  - Cohen's d = (Meanm – Means)/pooled SD
  - (182–170)/15 = 0.8… this is indeed a large effect
- Is it significant? No! p = .301

- Small studies (N << 100) may find medium or large effects that are not significant
  - Waste of resources! (unless they are pilot studies)

Determine sample size - example

- Do married men weigh more than single men?
  - Find 4000 married men: Nm = 4000, Mm = 177.5, SDm = 15
  - Find 4000 single men: Ns = 4000, Ms = 176.5, SDs = 15
- Effect size: 1 lb
  - Is this a large effect?
  - (177.5–176.5)/15 = 0.067… this is a very small effect
- Is it significant? Yes! p = .0014

- Large studies (N >> 100) may find very small effects that are significant
  - Also a waste of resources! (could have done with way fewer participants)

Determine sample size

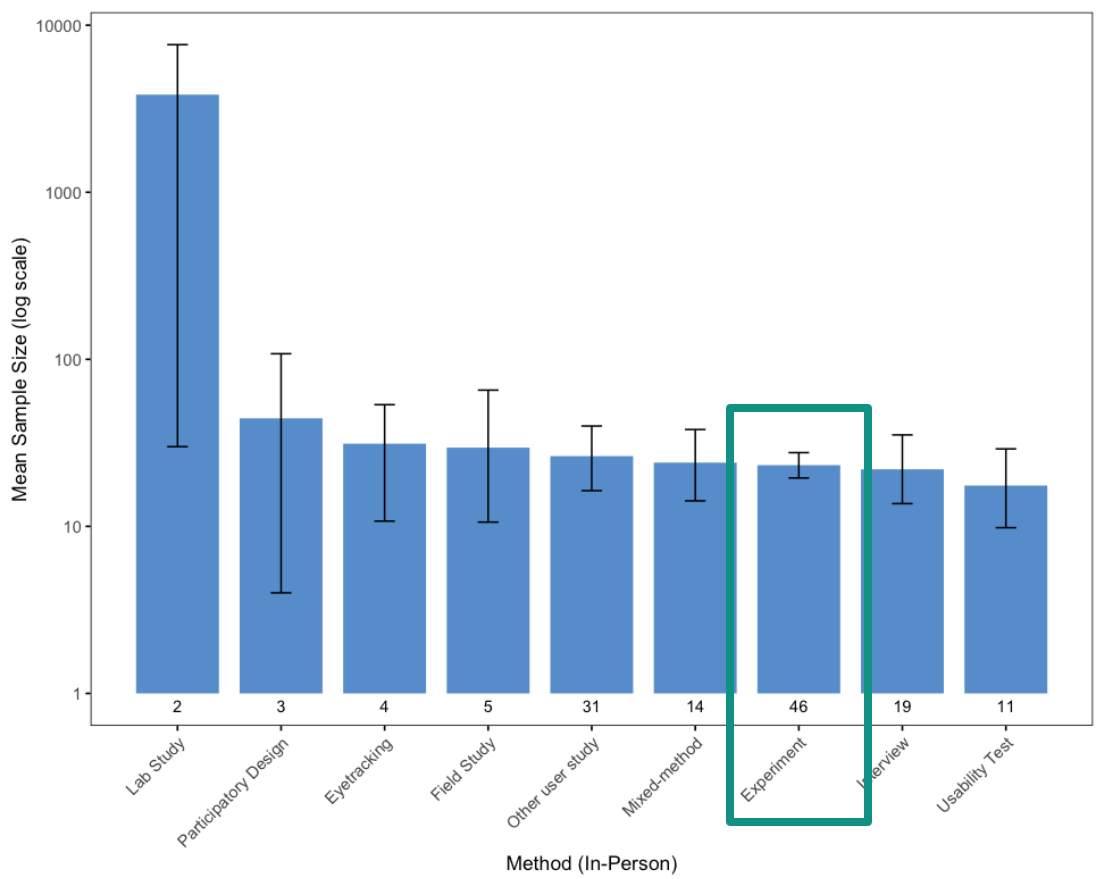
Power: the probability that a significance test will detect an effect given an effect exists

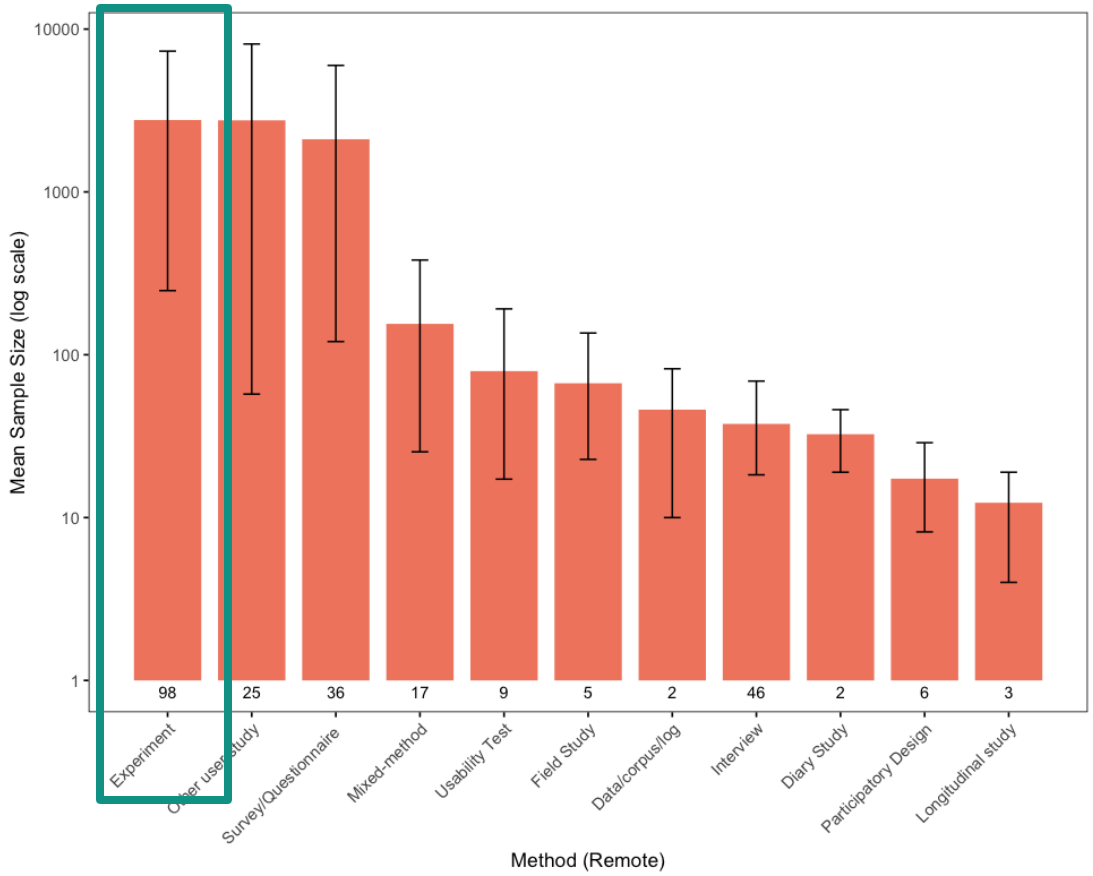
- See G*Power software
- You must specify
  - Minimum difference you want to detect (δ)
  - Whether H_1 is one- or two-sided
  - α, β and the population variance

See Walpole, et al. *Probability and statistics for engineers and scientists*

Anticipated effect size	Sample size per condition
Small	385
Medium	54
Large	25

Sample sizes – AI research at CHI'17-'24





Select participants

Inferences from your experiment apply to the **sampled population**—the group from which you have selected your users

- If you test your algorithm on freshman CS majors from The University of X, the results apply to this group in this context
- You can offer theoretical arguments for why inferences should apply more widely

You may need to repeat the experiment on other populations and in other contexts to determine whether results generalize

If the results do not hold in some other context then you have discovered boundary conditions for your theory

Internal and external validity

- **Internal validity**—The ability of the experiment to unambiguously show a cause-and-effect relationship
- **External validity**—The extent to which the results of the experiment can be generalized to other populations, contexts, and time periods

There is often a **trade-off** between them

- Highly controlled experiments may have strong IV but lack EV due to artificial ("lab") conditions.
- Studies in natural settings may have high EV but weaker IV because of the difficulty in controlling all confounds

See Calder, Phillips and Tybout (1982) for arguments on why IV should be emphasized in theoretical research

Some threats to internal validity

- **Hawthorne (placebo) effect:** users respond differently because they know they are being treated
  - Users should be **blind**, not knowing their treatment assignment
  - Control users should receive a **placebo** treatment
- **Experimental mortality:** Differential loss of respondents from different groups
  - Report difference in attrition
- **Regression effect:** If you test many variations, the one that works best in a study may not work as well in the real world ("you won't do as well in rollout as in test")

Running complex studies (POPROX promo)

The lack of research infrastructure often limits…

…the **complexity** of research studies

- Small samples from a limited population
- Short, ”single shot” studies; carryover and interactive test effects

…the **validity** of research studies

- Invited/compensated participation (no inherent motivation)
- Only immediate outcomes measurable

POPROX allows for longer-term studies with motivated users; with a built-in control group and default pre-/post-measures!

Some additional reading

- Maxwell, S. E., Delaney, H. D., & Kelley, K. (2017). Designing experiments and analyzing data: A model comparison perspective. Routledge. Psychology perspective, comprehensive
- Pearl, J., & Mackenzie, D. (2018). The book of why: the new science of cause and effect. CS, DOE history, causal models, mediation
- Kohavi, R., Tang, D., & Xu, Y. (2020). Trustworthy online controlled experiments: A practical guide to A/B testing. Cambridge University Press. CS, practical and relevant to POPROX
- Knijnenburg, B. P., & Willemsen, M. C. (2015). Evaluating recommender systems with user experiments. Recommender Systems Handbook, tailored to recommender systems
- Owen, A (2020). A First Course in Experimental Design, Stanford. Statistics, modern treatment of 100-year-old topic
- Online resource: www.usabart.nl/methods

Exercise – study design

Design a study to test the research model you developed:

- Using a within- or between-subjects design? Pre- and/or post-measures?
- What is the sampling population? Sample size?
- Precautions to increase internal validity?
- Ways to improve realism?

Bonus points: conduct a small pilot study to determine the expected effect size!

POPROX NEWS

4. Measurement

Measuring outcomes

Ask yourself: What is the goal of your system?

1. User model accuracy?
2. Conversion / user retention?
3. User experience?
4. Users' goal fulfillment?
5. Positive societal outcomes?

- Offline tests can only do #1 (kind of)
- Online A/B tests add #2
- Most experiments add #3 (and a bit of #4)
- With POPROX we aim to support all 5 goals!

Traditional (offline) metrics

Justification: used extensively in the ML/AI community, but confusion exists about the best way to calculate them

- Construct: Model accuracy
  - Binary outcome: Precision, recall, F1 metric, AUC
  - Numeric outcome: Mean absolute error (MAE), Root mean squared error (RMSE)
  - Ranking: Normalized discounted cumulative gain (NDCG), Mean reciprocal rank (MRR), other measures (ERR, RBP)
- Construct: Success and Conversion
  - Click-through rate (CTR)
  - Hit rate (fraction of recommended items with a click)

Equity, fairness, and diversity

Justification: impact to society; help exposure users to a wider range of content

- Construct: Distribution Equity
  - Gini index
  - Expected exposure
  - User space coverage vs item coverage (related to fairness due to exposure measure)
- Construct: Diversity
  - Intra-list diversity (average pairwise distance of recommended items)
  - Inter-recommendation diversity (similar to item coverage, it is an overall system measure)
    - Aggregate intra-list diversity (Gini coefficient, Gini-Simpson's index, entropy)
- Construct: Fairness across items
  - Equity of amortized attention

Online behavior metrics

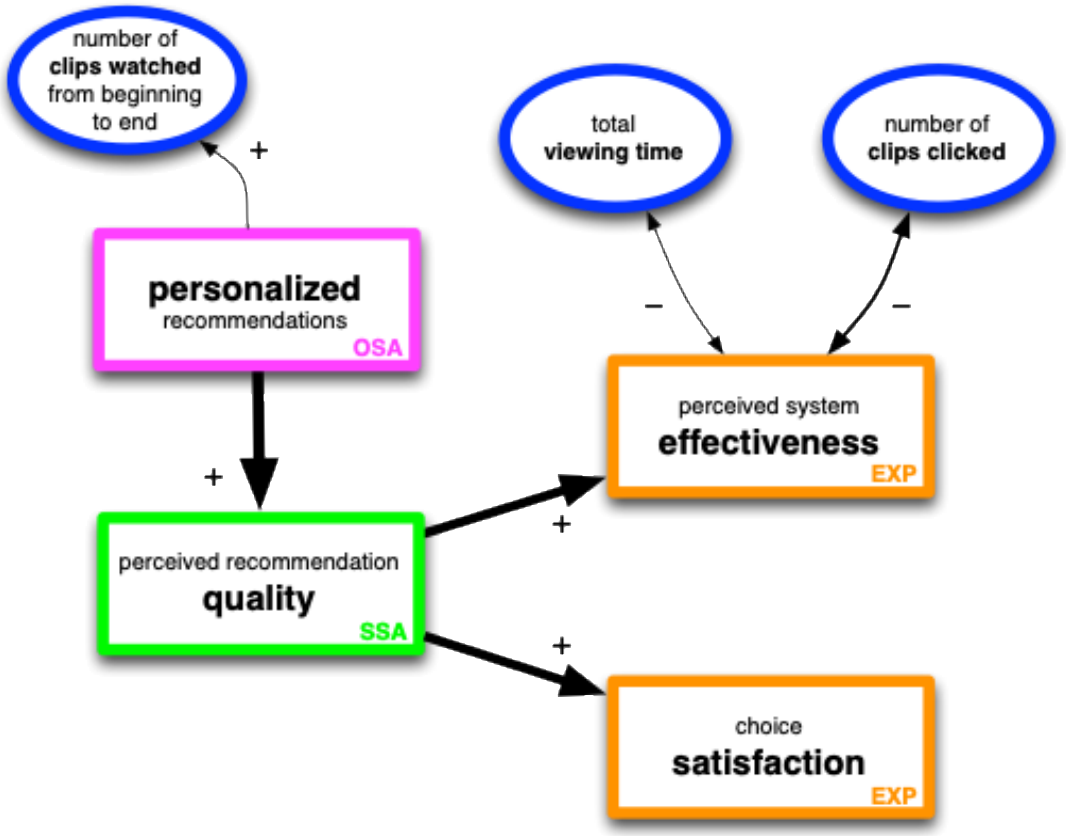
Justification: used extensively in (industry) field trials; “ground truth” user behaviors

- Construct: Engagement
  - Implicit item-based metrics: clicks, engagement time, social sharing
  - “System” metrics: dwell time, use frequency / consistency, bounce rate
- Construct: Interest
  - Explicit item-based feedback metrics, e.g. rating, thumbs up/down, like button
  - Referral / forwarding
- Construct: Retention
  - Time since last use

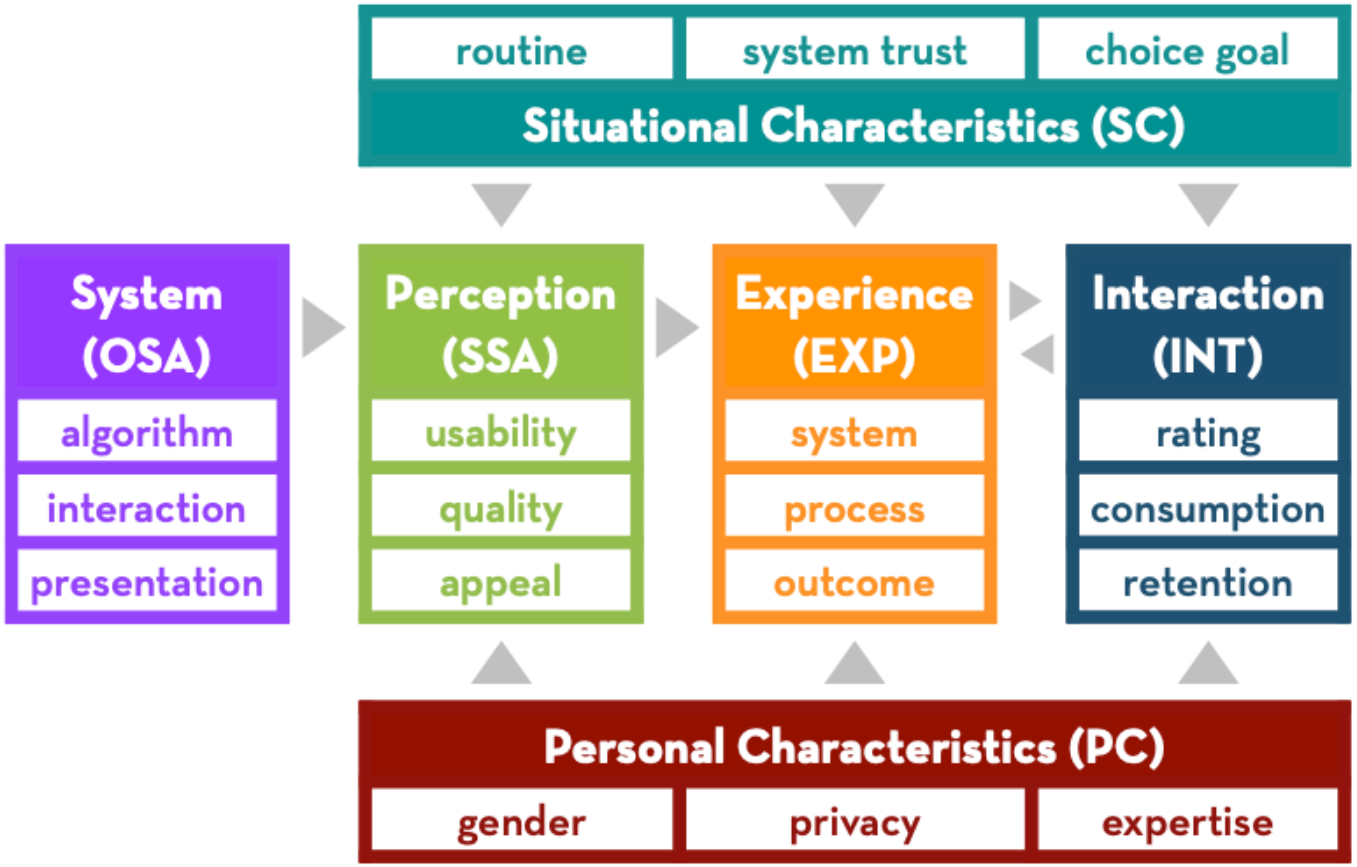
Why go subjective?

“Testing a recommender against a random videoclip system, the number of clicked clips and total viewing time went down!”

Asking for subjective evaluations explains why this happened



User-centric metrics



Justification: widespread use in AI/ML user experiments

- Measuring user perceptions (SSA) and experiences (EXP) to explain the effect of system manipulations (OSA) on user behavior (INT)
- Helps to produce robust, generalizable study results

User-centric metrics

- Subjective System Aspects (SSA)
  - Constructs: Perceived recommendation quality, perceived recommendation diversity, understandability, perceived control, perceived use effort / ease of use
- User Experience (EXP)
  - Constructs: System satisfaction, perceived system effectiveness/usefulness, choice difficulty (usage satisfaction), choice satisfaction
- Metrics:
  - Each construct is a multi-item scale (usually 4-7 items) of statements rated on a 7-point agreement scale.
  - Multi-item scales allow for an evaluation of their validity and robustness

Developing your own scales? See DeVellis (2011). Scale Development: Theory and Applications

Self-actualization metrics

Justification: use ML/AI technology to support users in developing, exploring, their long-term goals and ambitions, using constructs such as:

- Interest coverage (do the adaptations cover all my interests?)
- Goal clarification potential (does the system help me explore and understand my goals?)
- Goal development potential (does the system help me move beyond / develop my current goals?)
- Perceived self-actualization (does the system help me meaningfully improve my life?)

See Knijnenburg et al. (2016). Recommender Systems for Self-Actualization

Uses and gratifications metrics

Justification: used extensively in communication research; ideal for measuring the real-world impact of our system. Highly predictive of future use.

- Originally developed to explain media choice; measure how the recommendations contribute to a user's life goals:
  - Utilitarian tips and advice ("suggestions that you can use")
  - Curation and learning ("makes me smarter")
  - Social facilitation--promotes positive social contacts
  - Feel good, inspiration--being uplifted and motivated to be a better person
  - Surprise and serendipity--encountering something surprising or out of the ordinary
  - Expose me to different perspectives

Covariates

Justification: measure things about the user to make sure that innovations have an equitable impact (or: show how certain manipulations affect only a subset of users)

- Demographics (mostly stable)
  - Age, gender, location, education, SES, political leaning, domain interests, etc.
- Personal characteristics (mostly stable)
  - Constructs: domain knowledge, choice maximization, need for cognition, privacy concerns, familiarity with technology
- Situational characteristics (may vary over time)
  - Constructs: trust (multi-faceted), mood, current goals (see uses and gratifications)

A note on demographics

Reasons to collect:

- Provide context about the conducted study (also: reproducibility)
- Increase balance and coverage of the collected sample
- Test for outcome fairness across certain demographics
- Uncover moderation effects

Reasons to avoid:

- Some demographics are considered private information
- A combination of demographics may re-identify the participant
- Participants may feel discomfort disclosing their demographics

Worse when also collecting user modeling data!

Demographics – AI research at CHI'17-'24

Top demographics reported: gender (66%), age (58%), education (13%), race/ethnicity/nationality (9%)

Reporting methods for age: central tendency (59%), range (51%), specific values (13%), categories (3%)

Number of top demographics per paper:

Number of demographics reported	# Studies (% of Total)
0 of 4	120 (26.7%)
1 of 4	76 (16.9%)
2 of 4	187 (41.6%)
3 of 4	58 (12.9%)
All 4	9 (2.0%)
Total	**450**

Another POPROX promo slide

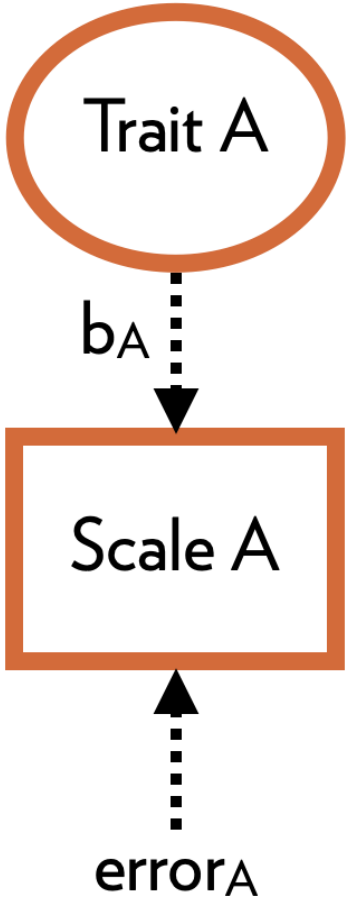
- POPROX measures all mentioned constructs with at least one item each, by default!
  - Added bonus: We have a history of baseline values for each metric for each user
- You can select scales to **expand** to multiple items
  - Can be helpful for robustness
- You can also **add your own** scales
  - Tailored to your study
- Above all: because these are **real users**, the goal fulfillment metrics have external validity

Multi-item measurement

- For subjective traits, single-item measurements tend to lack **content validity**
  - Each participant may interpret the item differently
  - This reduces precision and conceptual clarity
- Accurate measurement requires a **shared conceptual understanding** between all participants and researcher
- Solution: use a multi-item **scale**

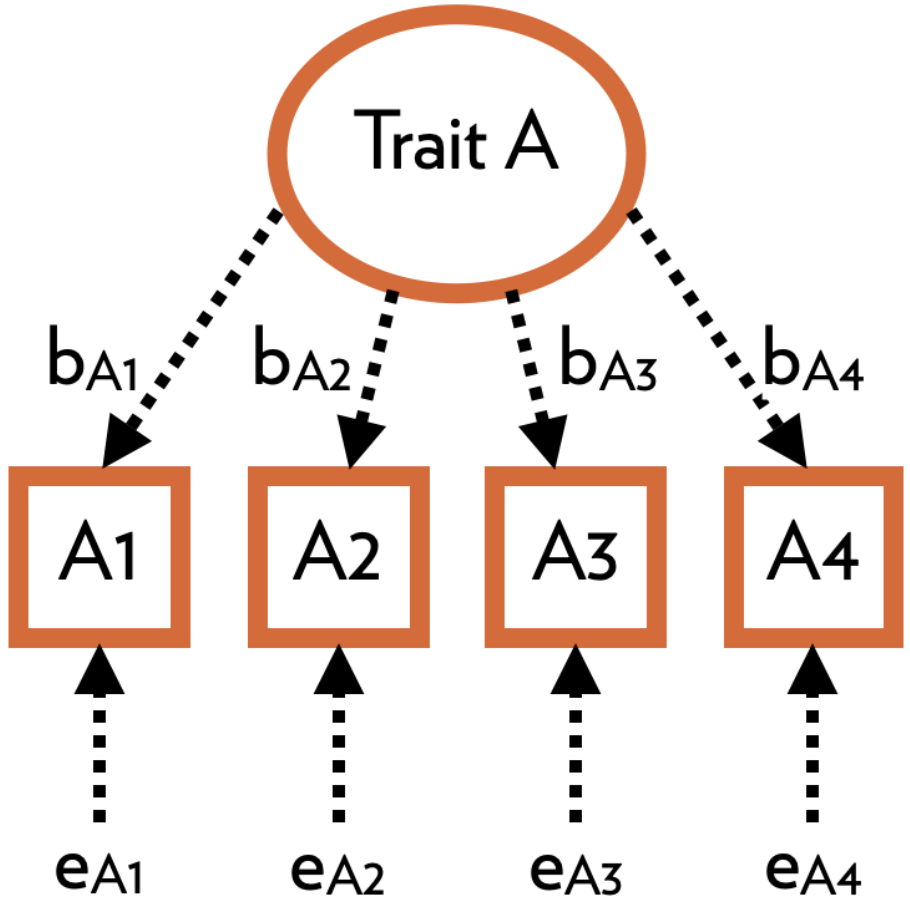
Multi-item measurement

- Even then, a scale is an **imperfect** way of measuring a subjective trait
  - Our real goal is to measure the trait, not the scale
- We can think of the traits as latent variables and the scales as observed variables
  - The trait causes my answers on the scale
- Like a regression with an unobserved X
  - Scale A = a + bATrait A + errorA
  - The R^2 of this regression determines how well we are measuring Trait A



Multi-item measurement

- How do we get this R2?
  - Trick: if you have multiple items, you can derive b's from the correlation between the items
- See example:
  - The b's are "loadings"
  - The e's are "uniqueness"
  - R^2 = 1-e is called "communality"
- Each item uses the others as a yardstick
  - Once a scale is validated, a single item may suffice*



Exercise – measurement

Operationalize the measurements in your research model:

- How would you evaluate the accuracy of the user model?
- What user behaviors do you want to track/observe?
- What subjective constructs do you want to measure?
- What are important covariates to include?

Bonus points: find existing scales for the constructs you want to measure!

POPROX NEWS

5. Evaluating research models

Running example: TasteWeights data

twq.dat (download at https://www.usabart.nl/QRMS/), variables:

- cgraph: inspectability (0: list, 1: graph)
- citem-cfriend: control (baseline: no control)
- cig (citem * cgraph) and cfg (cfriend * cgraph)
- s1-s7: satisfaction with the system
- q1-q6: perceived recommendation quality
- c1-c5: perceived control
- u1-u5: understandability
- e1-e4: user music expertise
- t1-t6: propensity to trust
- f1-f6: familiarity with recommenders
- average rating of, and number of known items in, the top 10
- time taken to inspect the recommendations

Step 1: CFA

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 cfa (load package lavaan):

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

Inspect model output:

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

Output (selected)

	Estimate	Std.err	Z-value	P(>|z|)
Latent variables:				
satisf =~				
s1	0.888	0.018	49.590	0.000
s2	-0.885	0.018	-48.737	0.000
s3	0.771	0.029	26.954	0.000
s4	0.821	0.025	32.363	0.000
s5	0.889	0.018	50.566	0.000
s6	0.788	0.031	25.358	0.000
s7	-0.845	0.022	-38.245	0.000
quality =~				
q1	0.950	0.013	72.421	0.000
q2	0.949	0.013	72.948	0.000
q3	0.942	0.012	77.547	0.000
q4	0.805	0.033	24.257	0.000
q5	-0.699	0.042	-16.684	0.000
q6	-0.774	0.040	-19.373	0.000

Output (selected)

control =~				
c1	0.712	0.038	18.684	0.000
c2	0.855	0.024	35.624	0.000
c3	0.905	0.022	41.698	0.000
c4	0.723	0.037	19.314	0.000
c5	-0.424	0.056	-7.571	0.000
underst =~				
u1	-0.557	0.047	-11.785	0.000
u2	0.899	0.016	57.857	0.000
u3	0.737	0.030	24.753	0.000
u4	-0.918	0.016	-58.229	0.000
u5	0.984	0.010	97.787	0.000

These are the loadings (the regression bs on the arrows going from the factor to the item)
They should be > 0.70 (because R^2 = loading^2 should be > 0.5)
Negative loadings are for negative items (please check!!)

Output (selected)

Covariances:

satisf ~~				
quality	0.686	0.033	20.503	0.000
control	-0.760	0.028	-26.913	0.000
underst	0.353	0.048	7.320	0.000
quality ~~				
control	-0.648	0.040	-16.041	0.000
underst	0.278	0.058	4.752	0.000
control ~~				
underst	-0.382	0.051	-7.486	0.000

These are the factor correlations (the numbers on the arrows going from one factor to another)
They should not be too high (more about this later)
Note: the control factor turns out to be "lack of control" (that happens sometimes)

Output (selected)

R-Square:

s1	0.788
s2	0.782
s3	0.594
s4	0.674
s5	0.790
s6	0.621
s7	0.714
q1	0.903
q2	0.901
q3	0.888
q4	0.648
q5	0.489
q6	0.599
c1	0.506
c2	0.731
c3	0.820
c4	0.522
c5	**0.179**
u1	**0.310**
u2	0.808
u3	0.544
u4	0.843
u5	0.968

R-square, also called "variance extracted" or "communality"
Should be > 0.50 (or at the very least > 0.40)

Step 1.1: Improve the model

- Remove items with low communality
  - check for r-square < 0.40 (or maybe 0.50)
  - In this model, we first remove c5 (r-squared = 0.180), then u1 (r-squared = 0.324)
- Remove items with high cross-loadings or residual correlations
  - check the modification indices: mods <- modindices(fit, power=TRUE)
  - Only show the ones that are significant and large:
    mods <- mods[grep("*", mods$decision),]
  - In this model, **u3** has a high cross-loading with satisfaction and quality, correlations with c1 and c6
- Iterate, but try to keep at least three items
  - if necessary, specify a model with cross-loadings or residual correlations… but try to avoid this!

Step 1.2: Inspect the model

- Inspect the item-fit (this should be good by now)
- Determine factor-fit:
  - Convergent validity: Average Variance Extracted (AVE = average R-squared per factor) > 0.50
  - Discriminant validity: $\sqrt{(AVE)}$ > largest correlation with other factors
- Determine model-fit:
  - Chi-square test: whether there is any misfit (this is often true!)
  - Alternative fit metrics: chi-squared / df < 3, CFI > 0.96, TLI > 0.95, RMSEA < 0.05 and a CI not exceeding 0.10.

Factor-fit of the example

- Satisfaction:
  - AVE = 0.709, $\sqrt{(AVE)}$ = 0.842, largest correlation = 0.762
- Quality:
  - AVE = 0.737, $\sqrt{(AVE)}$ = 0.859, largest correlation = 0.687
- Control:
  - AVE = 0.643, $\sqrt{(AVE)}$ = 0.802, largest correlation = 0.762
- Understandability:
  - AVE = 0.894, $\sqrt{(AVE)}$ = 0.946, largest correlation = 0.326

Model-fit of the example

Model Test User Model:

	Standard	Scaled
Test Statistic	161.635	284.199
Degrees of freedom	164	164
P-value (Chi-square)	0.538	0.000
Scaling correction factor		0.760
Shift parameter		71.415
simple second-order correction		

Model Test Baseline Model:

Test statistic	45418.183	14072.760
Degrees of freedom	190	190
P-value	0.000	0.000
Scaling correction factor		3.258

Model shows significant misfit, But Chi-square / df is good: 284 / 164 = 1.73

This tests if the model is better than the worst possible model (unsurprisingly, it is…)

Model-fit of the example

User model versus baseline model:

Comparative Fit Index (CFI)	1.000	0.991
Tucker-Lewis Index (TLI)	1.000	0.990

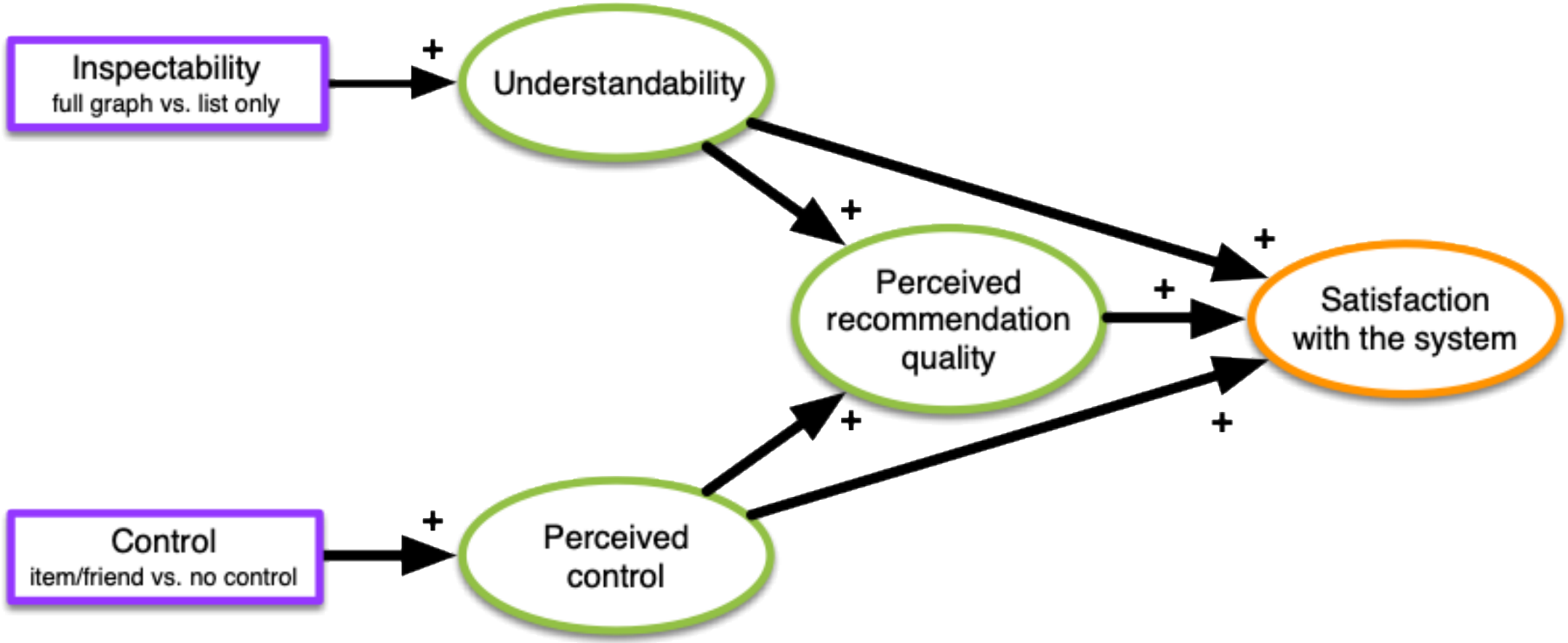
Root Mean Square Error of Approximation:

RMSEA	0.000	0.053		
90 Percent Confidence Interval	0.000	0.027	0.043	0.063
P-value RMSEA <= 0.05	1.000	0.311		

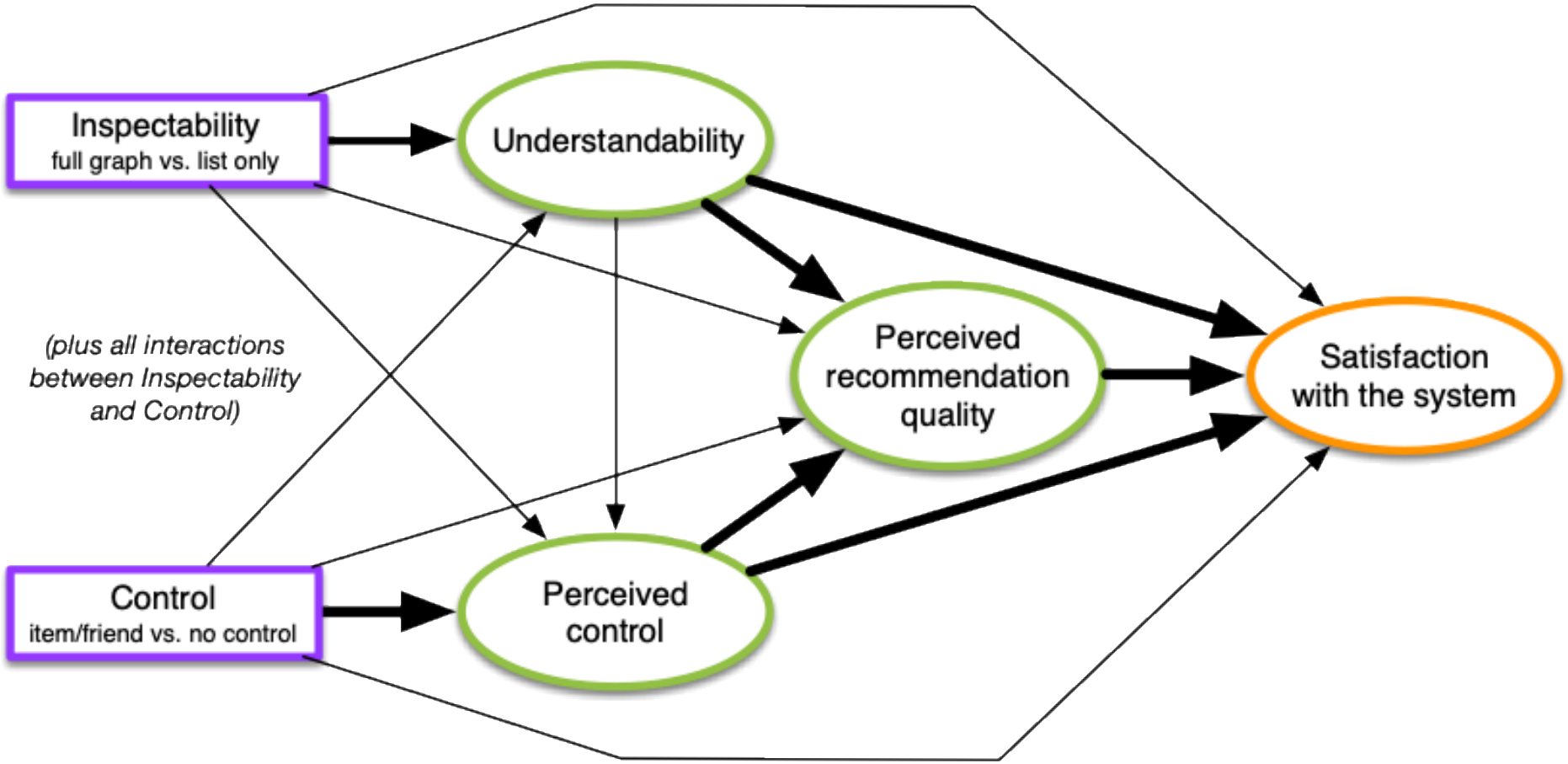
CFI (> 0.96) and TLI (> 0.95) are excellent

RMSEA > 0.05 is a bit high, but the 90% CI is ok (does not exceed 0.10)

Step 2: SEM – theoretical model



Step 2: SEM – saturated model



Step 2: SEM – model specification

```
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[9:31]), std.lv=TRUE)
summary(fit, rsquare=TRUE, fit.measures=TRUE)
```

Output (selected)

Regressions:

	Estimate	Std.Err	z-value	P(>|z|)
satisf ~				
quality	0.439	0.076	5.755	0.000
control	-0.839	0.107	-7.813	0.000
underst	0.089	0.072	1.236	0.216
citem	0.318	0.265	1.200	0.230
cfriend	0.014	0.257	0.056	0.955
cgraph	0.309	0.229	1.349	0.177
cig	-0.386	0.356	-1.082	0.279
cfg	-0.394	0.357	-1.103	0.270
quality ~				
control	-0.765	0.086	-8.913	0.000
underst	0.043	0.073	0.589	0.556
citem	0.046	0.203	0.228	0.819
cfriend	0.166	0.251	0.661	0.508
cgraph	0.010	0.236	0.041	0.968
cig	**0.106**	**0.317**	**0.334**	**0.739**
cfg	**0.179**	**0.374**	**0.478**	**0.633**

Output (selected)

control ~				
understand	-0.306	0.065	-4.705	0.000
citem	0.051	0.240	0.212	0.832
cfriend	0.006	0.221	0.026	0.979
cgraph	-0.046	0.239	-0.193	0.847
cig	**-0.148**	**0.341**	**-0.435**	**0.664**
cfg	**-0.272**	**0.331**	**-0.822**	**0.411**
understand ~				
citem	0.363	0.218	1.666	0.096
cfriend	0.529	0.215	2.465	0.014
cgraph	0.551	0.225	2.453	0.014
cig	**-0.107**	**0.323**	**-0.331**	**0.740**
cfg	**-0.177**	**0.317**	**-0.558**	**0.577**

Step 2.1 Trim the 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 one is significant, keep the others as well)
  - Interaction+main effects: never remove main effect before the interaction effect (if the interaction is significant, keep the main effect regardless)

Final trimmed model specification

```
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
                    quality ~ control
                    control ~ underst
                    underst ~ citem+cfriend+cgraph'
```

Final trimmed model - output

Regressions:

	Estimate	Std.Err	z-value	P(>|z|)
satisf ~				
quality	0.418	0.080	5.229	0.000
control	-0.888	0.120	-7.401	0.000
quality ~				
control	-0.780	0.084	-9.234	0.000
control ~				
underst	-0.368	0.066	-5.531	0.000
underst ~				
citem	0.379	0.198	1.915	0.056
cfriend	0.554	0.194	2.864	0.004
cgraph	0.622	0.164	3.789	0.000

Model fit

- Item and factor fit should not have changed much
  - (please double-check the r-squares!)
- Great model fit!
  - Chi-Square value: 305.333, df: 223 (value/df = 1.37)
  - CFI: 0.994, TLI: 0.995
  - RMSEA: 0.037 (great), 90% CI: [0.026, 0.047]

Regression R^2

- Satisfaction: 0.653
- Perceived Recommendation Quality: 0.413
- Perceived Control: 0.135
- Understandability: 0.131

These are all quite okay

Omnibus test (ANOVA)

Change the model definition:

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

Then run:

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

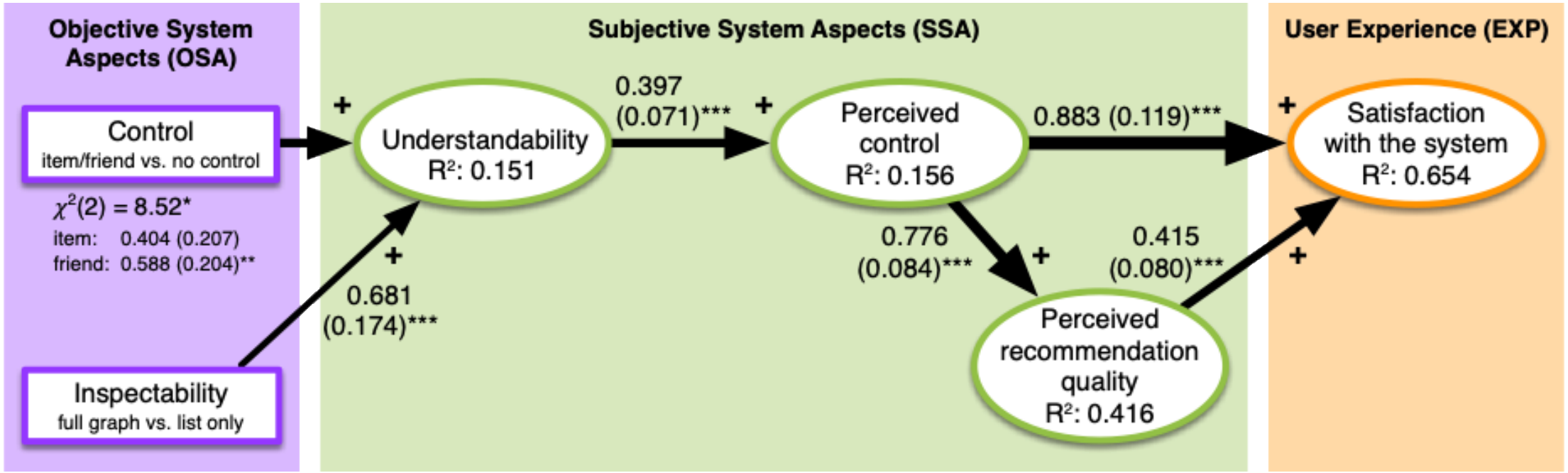
Result: Omnibus effect of control is significant (this is a chi-square test)

```
$stat
[1] 8.405182

$df
[1] 2

$p.value
[1] 0.01495677
```

Final trimmed model - graph



Some additional reading

- Kline, R. B. (2023). Principles and Practice of Structural Equation Modeling (Fifth Edition). Guilford Press.
- Online resources:
  - www.usabart.nl/eval
  - www.usabart.nl/eval2