Relevant Items MaxDiff
Origin, Motivation, and Options

Chris Chapman
+contributions **Eric Bahna**

Quant UX Association
Google

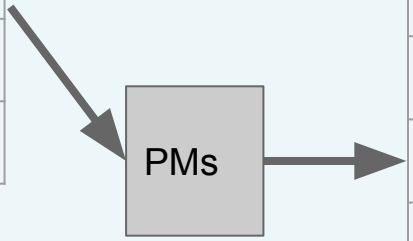
July 2025
Sawtooth Software Webinar Series

The Product Management Problem

⇒ Prioritize moderate to long lists of
features / initiatives / messages / preferences / needs

Sparse customer data → *poor* global prioritization

	FR1	FR2	FR3	FR4	FR5	FR6
CustomerA	P1	P1		P1		
CustomerB		P0				
CustomerC			P1			
CustomerD					P1	

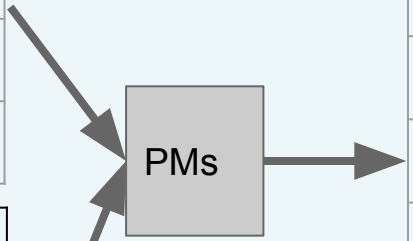


Rank	Feature	Priority
1	FR2	P0
2	FR1	P0
3	FR4	P1
4	FR5	P1
5	FR3	P2
6	FR6	P2

Dense customer data → *strong* global prioritization

	FR1	FR2	FR3	FR4	FR5	FR6
CustomerA	P1	P1		P1		
CustomerB		P0				
CustomerC			P1			
CustomerD					P1	

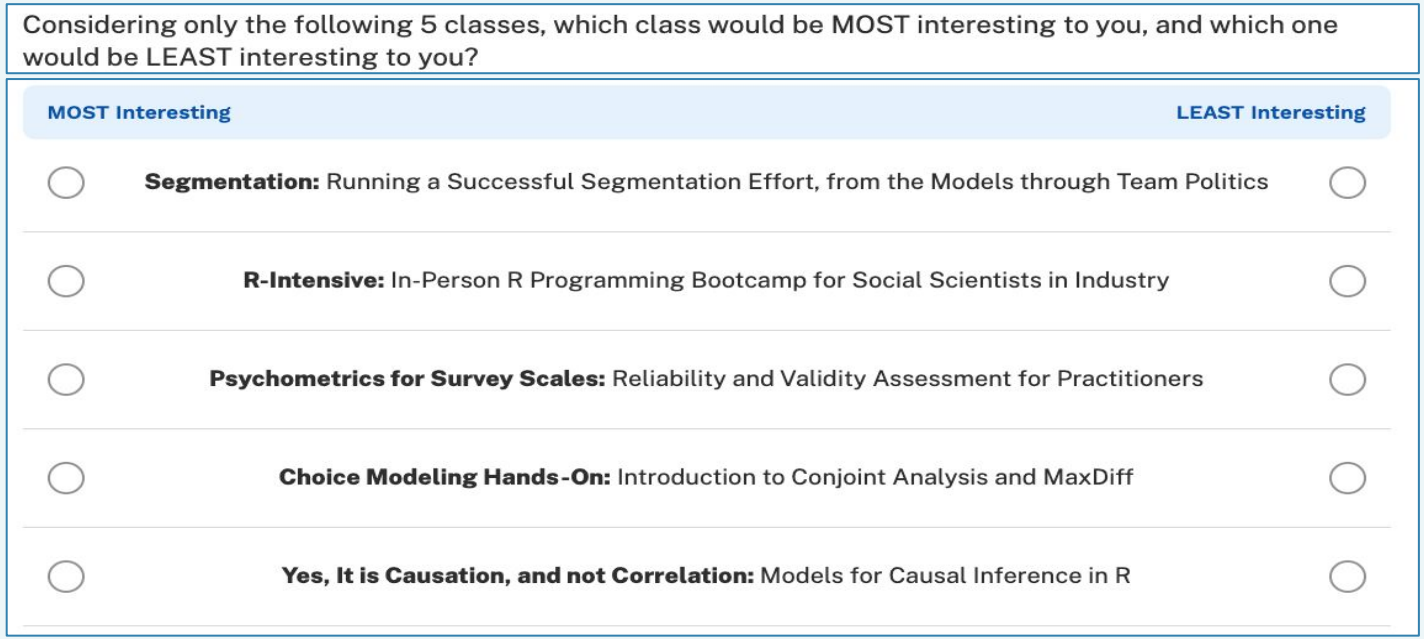
	FR1	FR2	FR3	FR4	FR5	FR6
CustomerA	16	11	17	21	24	11
CustomerB	26	2	8	25	12	27
CustomerC	5	15	6	42	23	9
CustomerD	3	11	8	28	23	27



Rank	Feature	Priority
1	FR4	P0
2	FR2	P0
3	FR5	P1
4	FR6	P1
5	FR1	P2
6	FR3	P2

Basic Approach: **MaxDiff**

- Given a list of **many items** (often 12-40; can be any number)
- … Ask for **preference** of a **few at a time** *[easy to answer]*



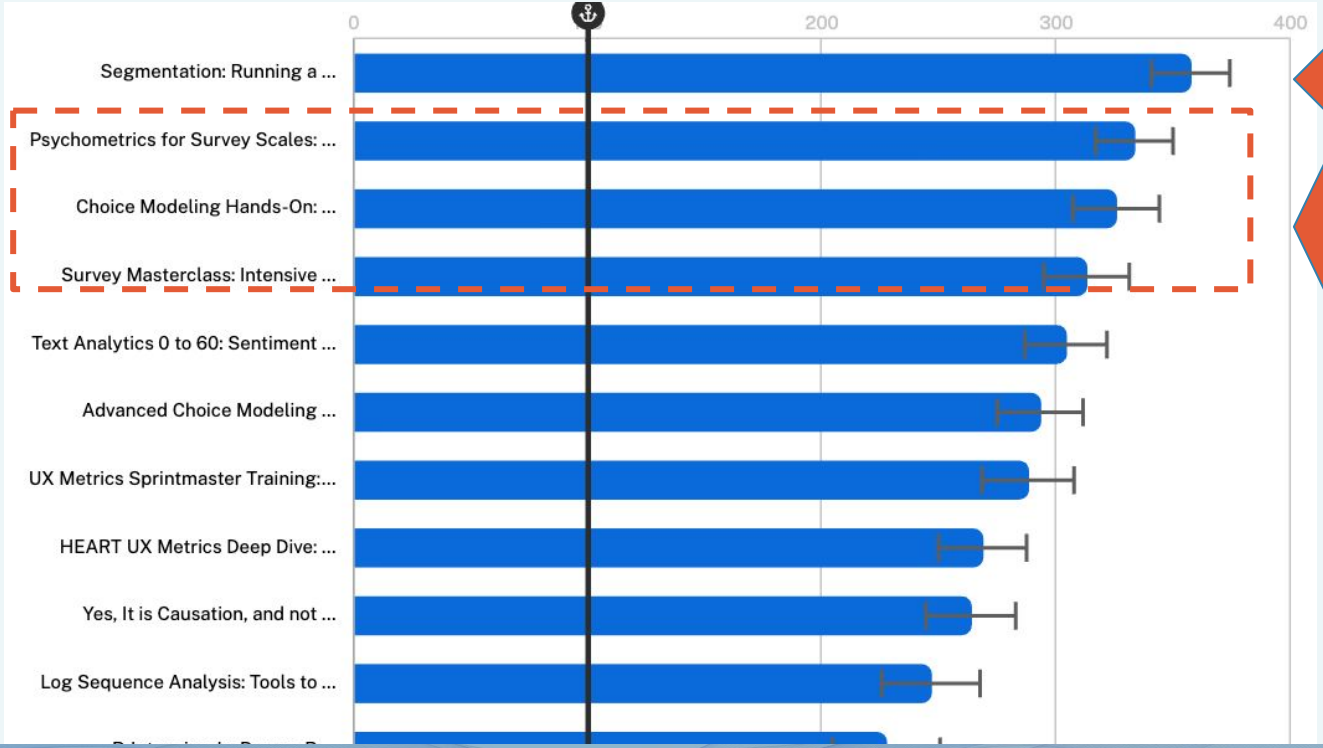
Basic Approach: **MaxDiff**

- Given a list of **many items** (often 12-40; can be any number)
- … Ask for **preference** of a **few at a time**

- … **Randomize sets** to avoid order effects *[unbiased]*
- … **Repeat** a few times *[more data]*
- … Model the preference **statistically** *[powerful]*

MaxDiff Results

- Ranked order preference with **meaningful magnitudes**



Result

- Ranked order preference with meaningful magnitudes

The "anchor" tells us that there is positive interest in *all* of the offerings

But … Problems with Standard MaxDiff

- Data Quality & Item relevance
- Respondent experience
- Non-actionable results

Data Quality

- *"I don't know -- someone else does that task."*
- Respondents must state a preference whether they know about the item or not.
- B2B Tasks & Large companies → Specialized Roles
  - Engineers
  - Salespeople
  - Finance
  - Operations
  - Security
  - Management

Respondent Experiences in Their Own Words

- “A bit **tedious**”
- “It was **LONG!**”
- “Quite **long**.”
- “Would be nice to have "no opinion" on a particular set to not introduce noise.”

Non-actionable Results

- We are **“wasting” participant’s time** if all items in a MaxDiff task are unimportant to them.

- Differentiating amongst the “worst” items is less valuable than **differentiating amongst the “best.”**

Some other MaxDiff Options

- **Adaptive MaxDiff** (Orme, 2006):
  Tournament-style progressive selection of items. More complex to program, less focused at beginning of survey. By itself, **doesn't solve "I don't do that."**

- **Express MaxDiff** (Wirth & Wolfrath, 2012):
  Selects subset of items to show each respondent. No insight at individual level on non-selected items. Addresses a **different problem** (long item list).

- **Sparse MaxDiff** (Wirth & Wolfrath, 2012):
  Uses all items from a long list per respondent, with few if any repetitions across choices. Low individual-level precision. Addresses **different problem**.

Relevant Items MaxDiff

AKA the artistic endeavor formerly known as

"Constructed / Augmented MaxDiff"

Initial B2B Study

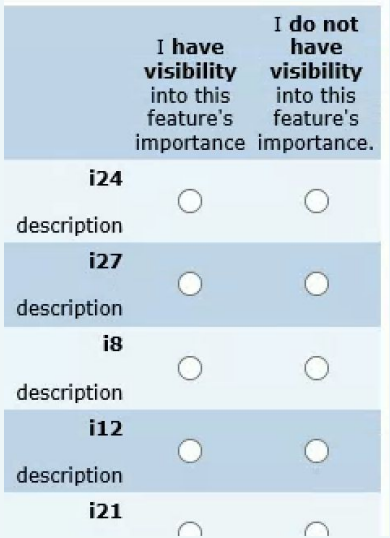
The problem:

We wanted ...

IT administrators to assess the **importance of features**
... but only that are **relevant** to their roles
... and to save time, are **at least somewhat important**

Relevant Items Screens the List Before MaxDiff

“Relevant to you?”

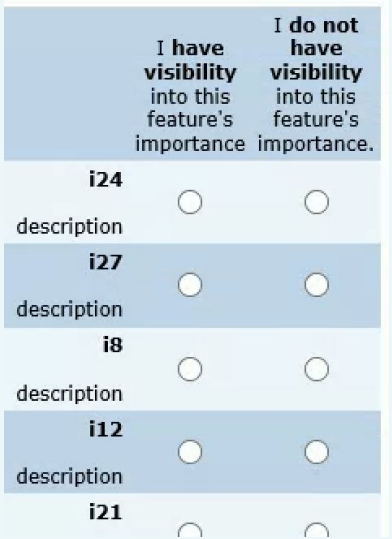


	I **have visibility** into this feature's importance	I **do not have visibility** into this feature's importance.
i24 description	○	○
i27 description	○	○
i8 description	○	○
i12 description	○	○
i21	○	○

Yes → Add item to MaxDiff list

Relevant Items Screens the List Before MaxDiff

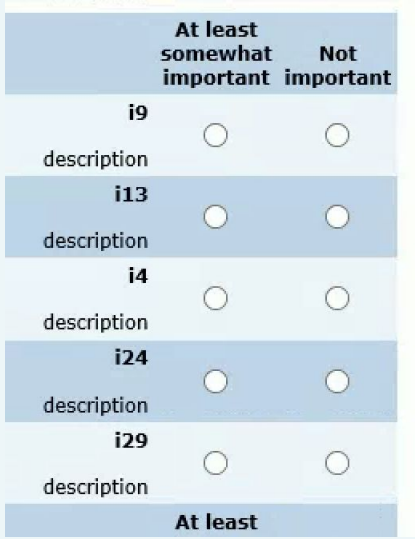
“Relevant to you?”



	I **have visibility** into this feature's importance	I **do not have visibility** into this feature's importance.
i24 description	○	○
i27 description	○	○
i8 description	○	○
i12 description	○	○
i21	○	○

AND/OR

“Important at all?”



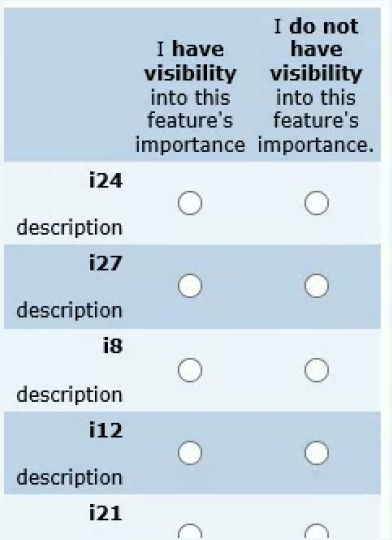
	At least somewhat important	**Not important**
i9 description	○	○
i13 description	○	○
i4 description	○	○
i24 description	○	○
i29 description	○	○
At least		

Yes → Add item to MaxDiff list

No → Remove item to save choice time

Relevant Items Screens the List Before MaxDiff

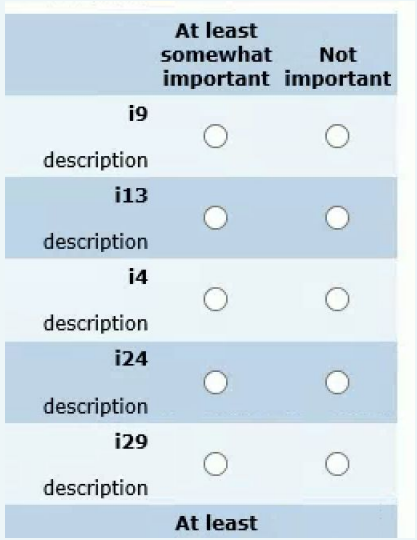
"Relevant to you?"



	I **have visibility** into this feature's importance	I **do not have visibility** into this feature's importance.
i24 description	○	○
i27 description	○	○
i8 description	○	○
i12 description	○	○
i21	○	○

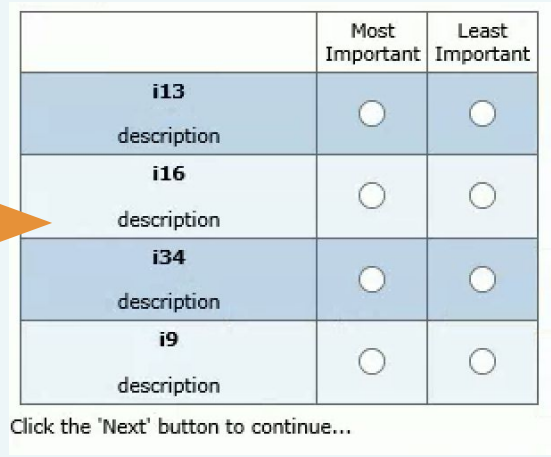
AND/OR

"Important at all?"



	At least somewhat important	**Not important**
i9 description	○	○
i13 description	○	○
i4 description	○	○
i24 description	○	○
i29 description	○	○
At least		

Then, MaxDiff



	Most Important	Least Important
i13 description	○	○
i16 description	○	○
i34 description	○	○
i9 description	○	○

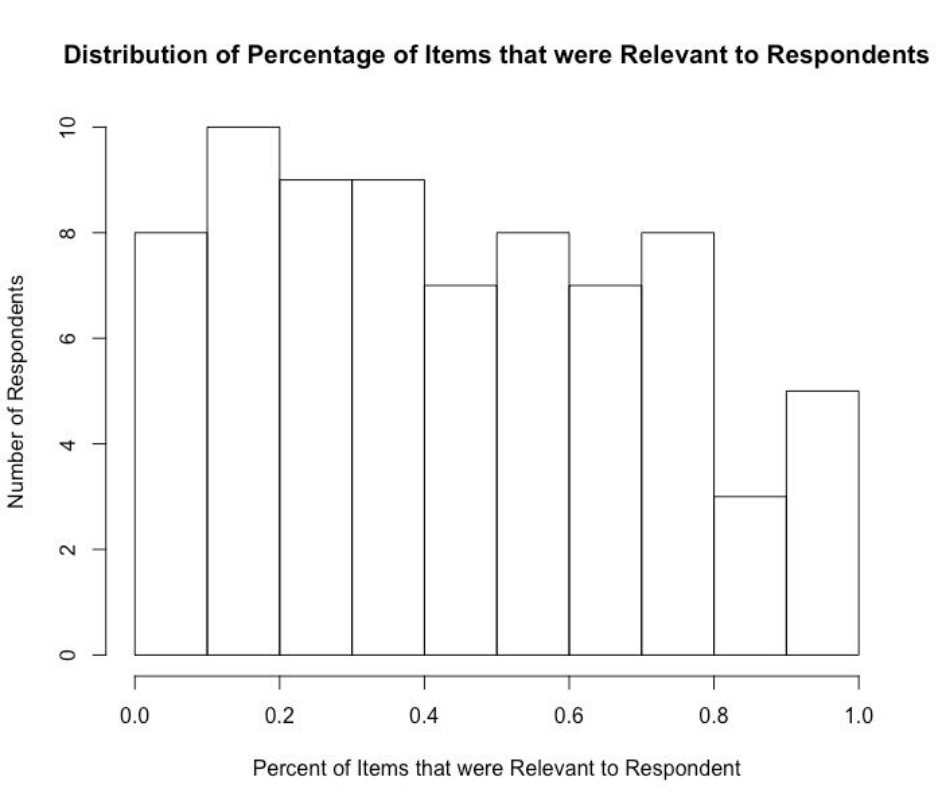
Click the 'Next' button to continue...

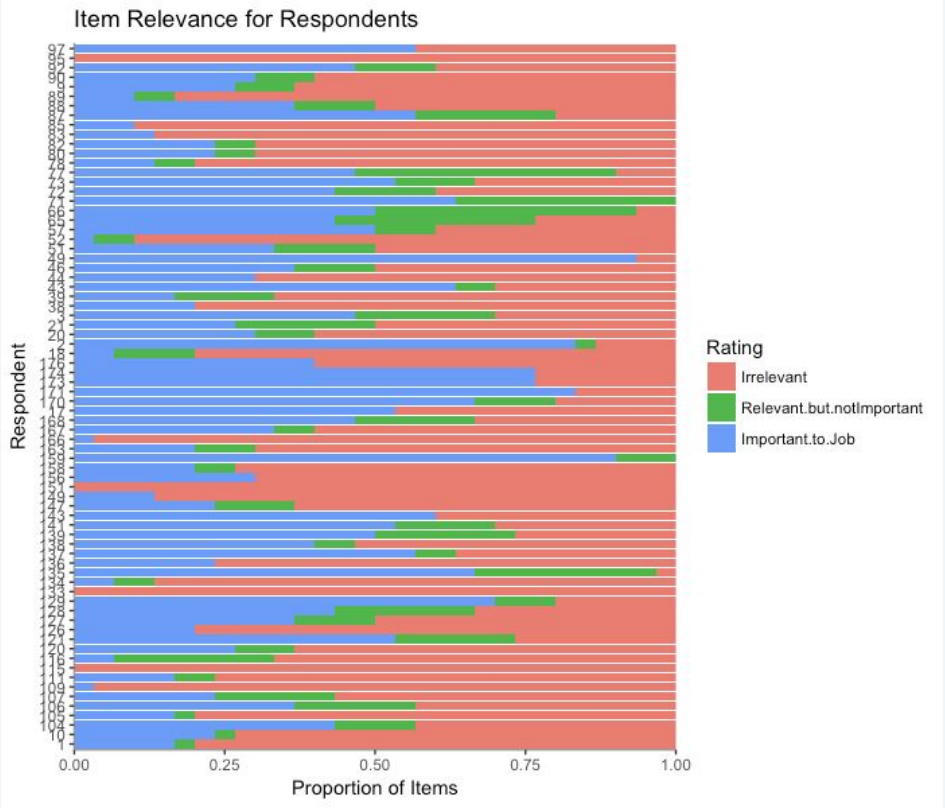
Yes → Add item to MaxDiff list

No → Remove item to save choice time

MaxDiff is tailored to the list of relevant items

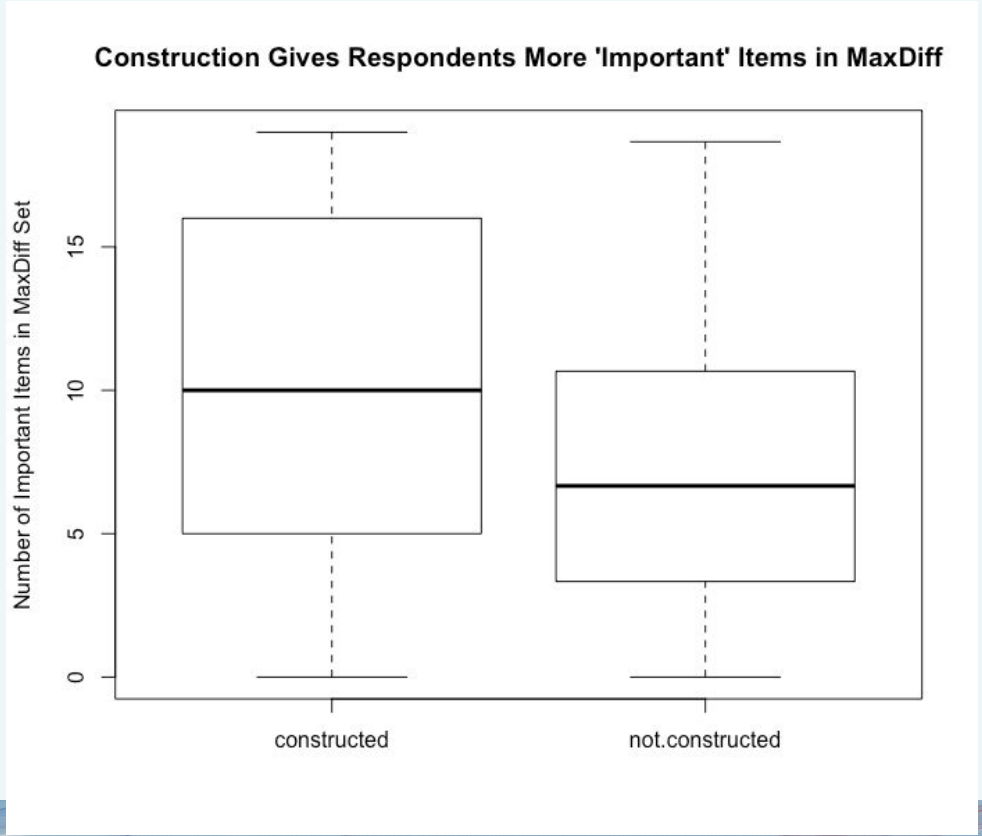
Results: **55% of Items Irrelevant** to Median Respondent





RI showed 50% More “Important” Items in MaxDiff

- Respondents see fewer unimportant items on average
- Better focus on top priorities



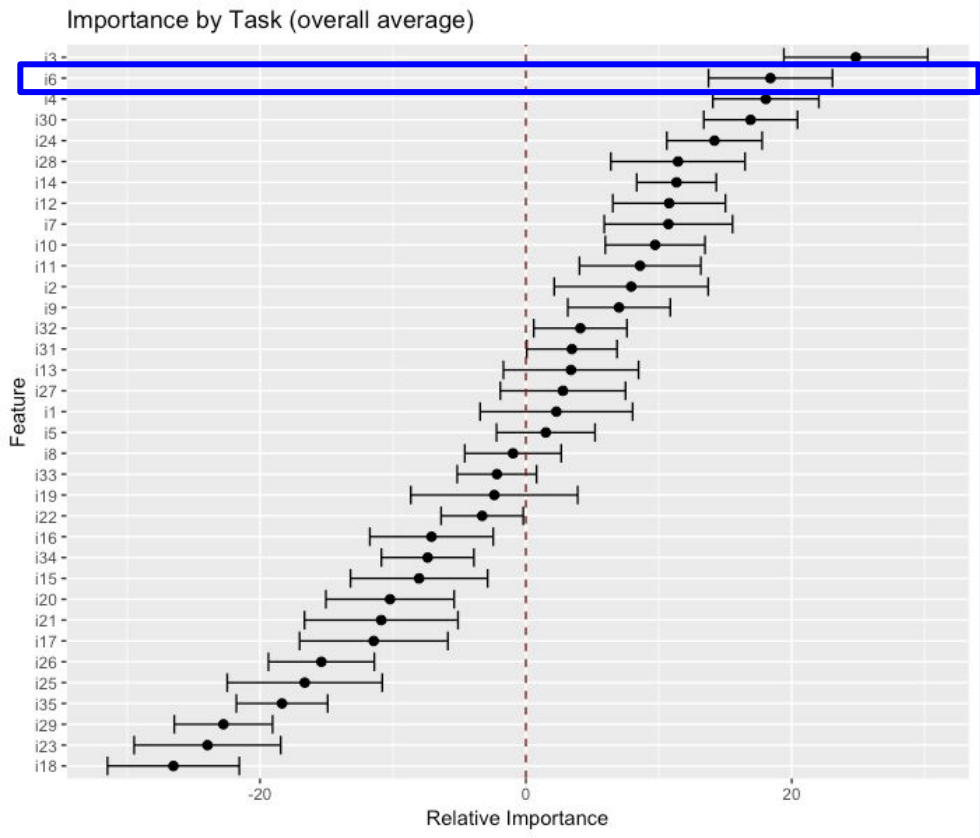
Result: Change in Business Priorities

- Better focus led to **higher estimate for item “i6”**
- #2 priority overall

  … and …

- #35 in cost - cheapest on the list

⇒ i6 went to top priority!



Results: Respondent and Executive **Feedback**

- Respondent feedback
  - “Format of this survey feels much **easier**”
  - “**Shorter** and **easier** to get through.”
  - “this time around it was a lot **quicker**.”
  - “Thanks so much for implementing the 'is this important to you' section! **Awesome** stuff!”

- Executive support
  - Funding for internal tool development (that was then; no longer needed!)
  - Advocacy across product areas
  - Support to teach 25+ Google classes on MaxDiff ⇒ 250+ participants

Implementation in Sawtooth Discover

Two Approaches

- Pre-screen for **Relevance** *(understanding, experience)*
  "Which of these movies have you seen?"
  "Which of these tasks do you perform?"

  — OR —

- Pre-screen for **Importance** *(liking, expectation, preference)*
  "Which features are at least somewhat important?"
  "Which destinations would you consider visiting?"

Two Approaches

- Pre-screen for **Relevance**
  “Which of these movies have you seen?”

- Pre-screen for **Importance**
  “Which features are at least somewhat important?”

Why not both simultaneously?

You can do both although it is more challenging (*more later*)

Example: Movie Ratings

Consider movie preferences. We might want to know:

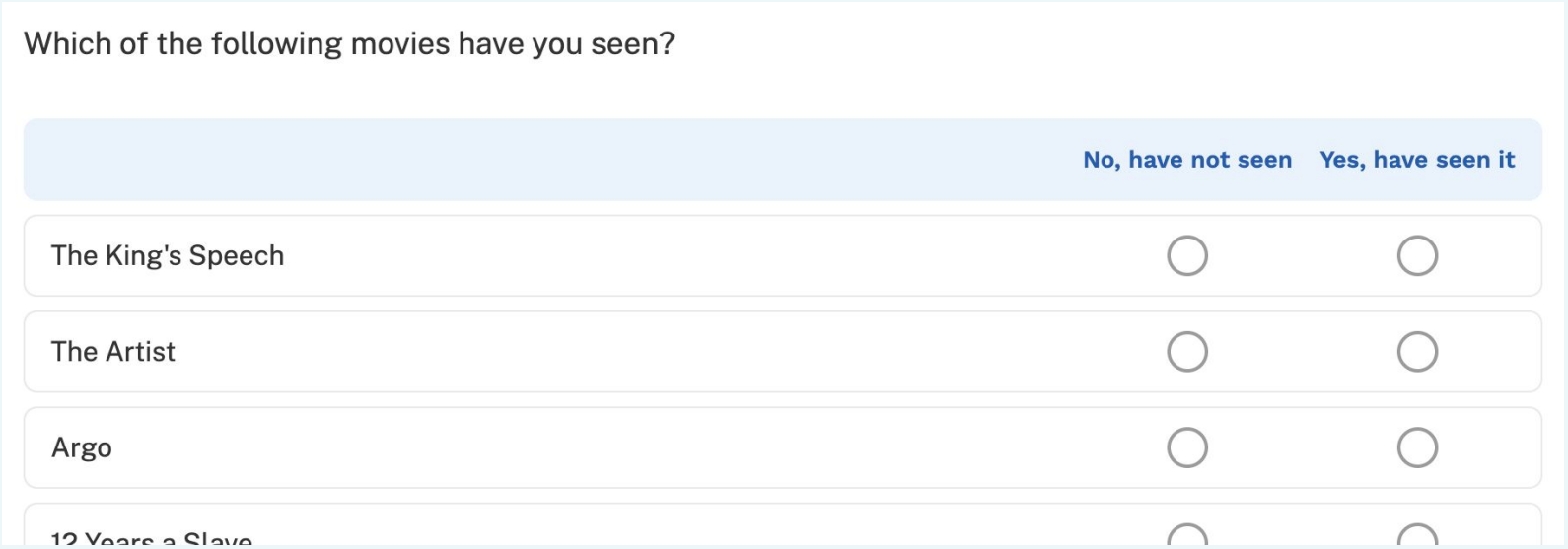
- **Of movies you've seen**, which do you like most?
  [screen for "Relevance"]

- **Among all movies**, which do you believe are best?
  [screen for "Importance" (i.e., preference)]

Items source: *Anonymous (2025). Oscar Winners & Nominees. Online spreadsheet.*

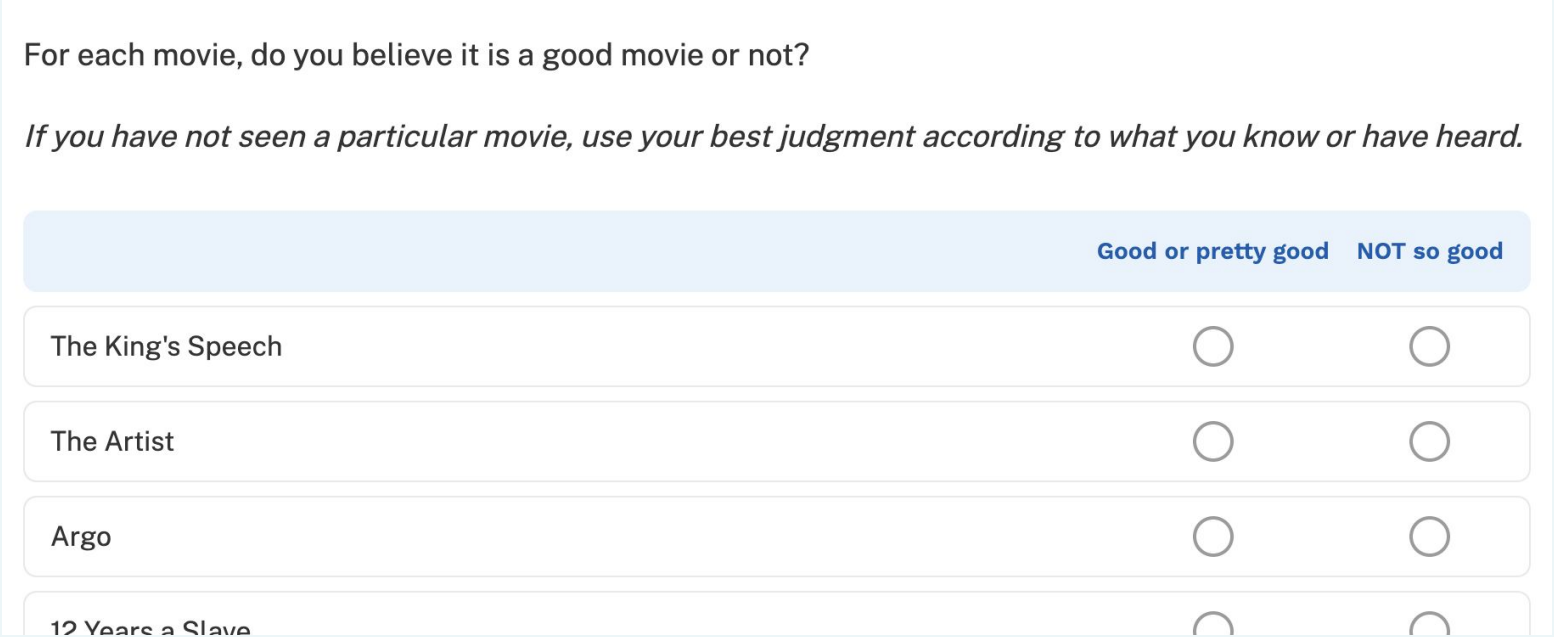
Screen for **Relevance**

- Ask about only items that a respondent is familiar with, that are relevant to them, etc.
- **Goal**: Collect good data and not confuse respondents



Screen for Importance

- Focus on items that are at least somewhat important
- **Goal**: shorten the task and focus on items at the “top”



Try it live! We'll see the results later

Consider movie preferences. We might want to know:

- **Of movies you've seen**, which did you like most?
  *[screen for "**Relevance**"]*

- **Among all the movies**, which do you think are best?
  *[screen for "**Importance**" (i.e., preference)]*

Example with both options: **http://bit.ly/ 3 I o J O q W**

Items source: *Anonymous (2025). Oscar Winners & Nominees. Online spreadsheet.*

Step 1: Create the Relevant Items List

- **Create the master list** we will select from
  - MovieList ... with all the movies in it

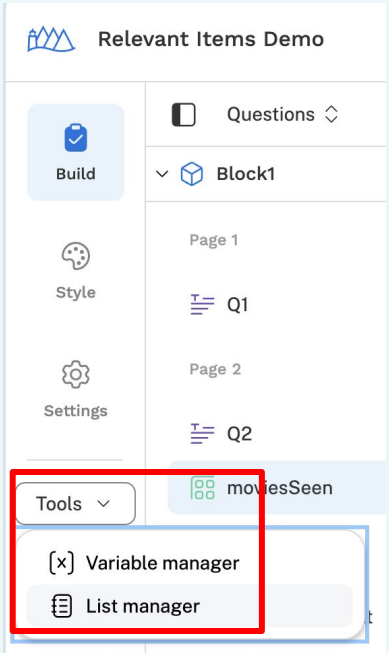
- Add a **survey item to select** from that master list

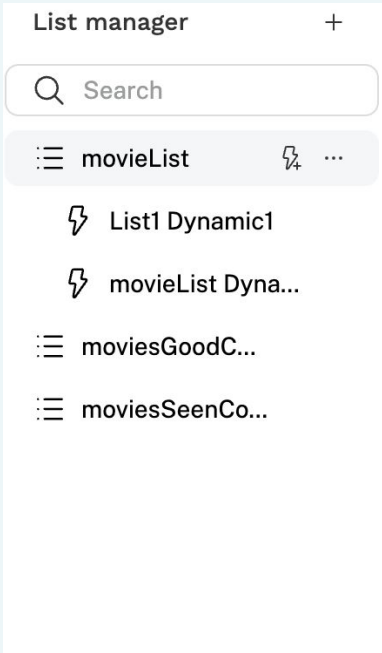
- Create a **dynamic list** to capture the selected items

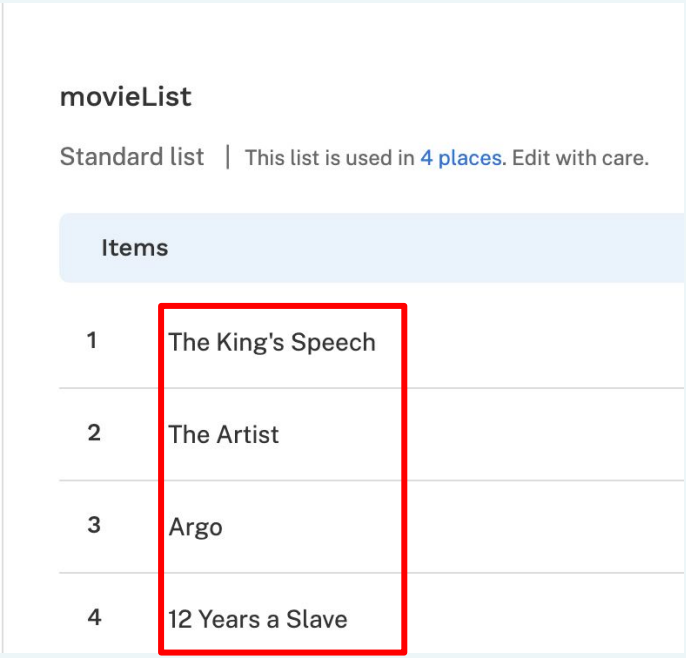
Step by Step in Sawtooth Discover

Step 1a: Create the Relevant Items List

- **Create the master list we will select from**

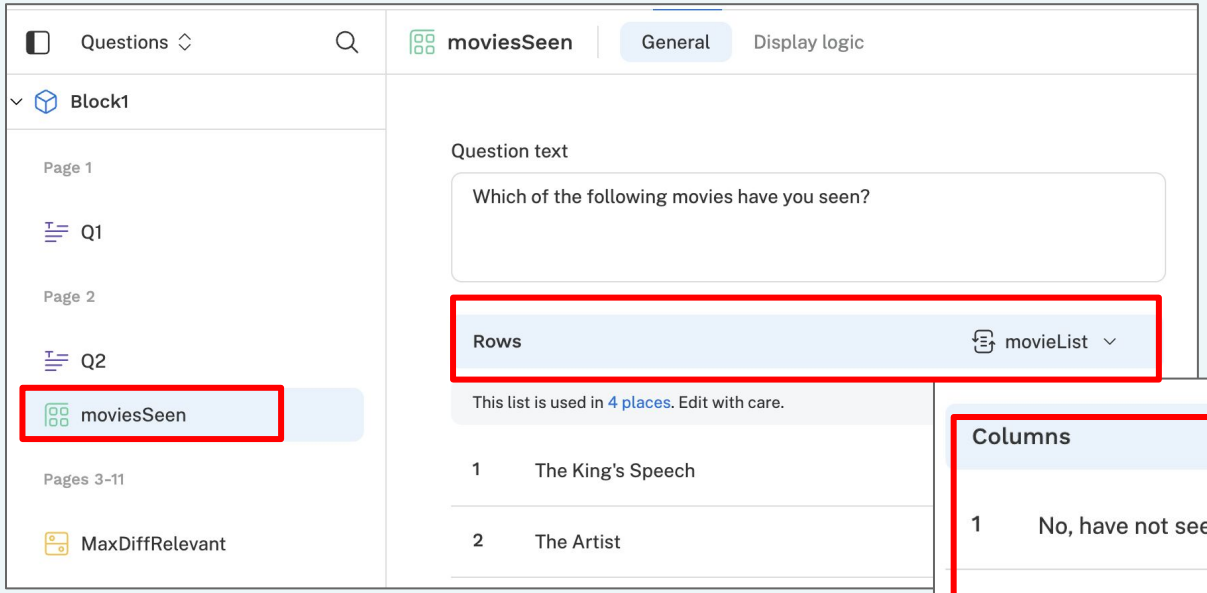


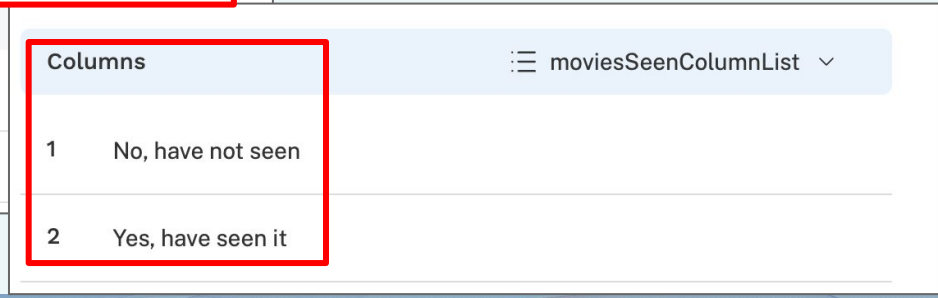




Step 1b: Create the Relevant Items List

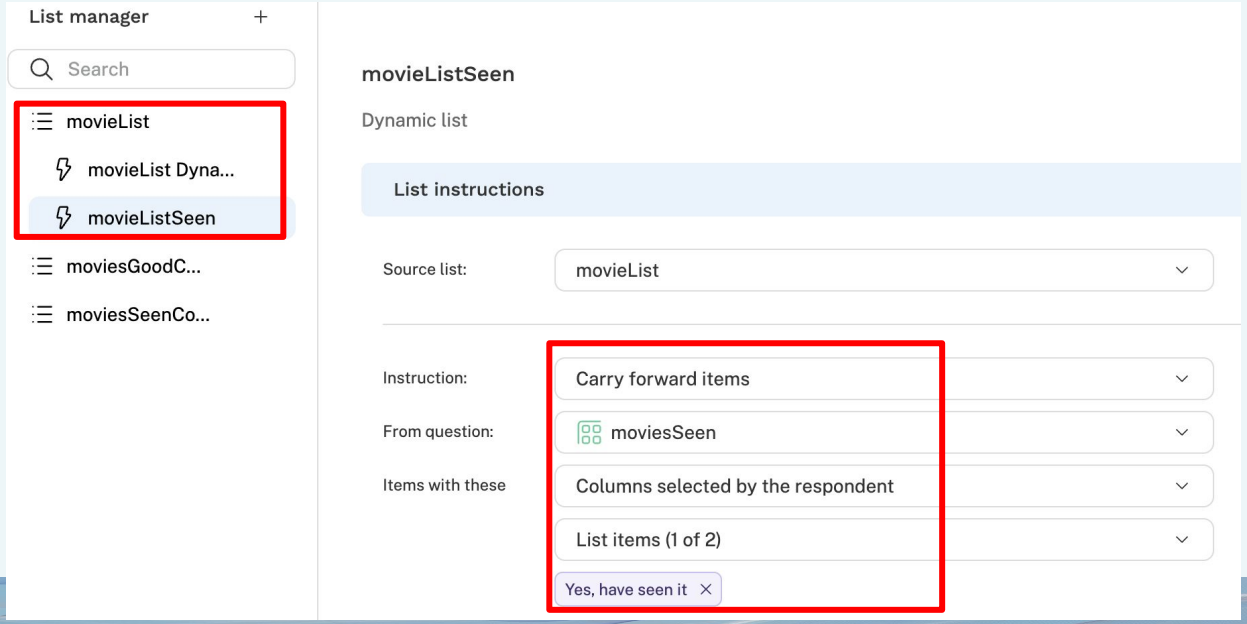
- Create the master list we will select from
- **Add a survey item to select from that master list**





Step 1c: Create the Relevant Items List

- Create the master list we will select from
- Add a survey item to select from that master list
- **Create a dynamic list to capture the selected items**

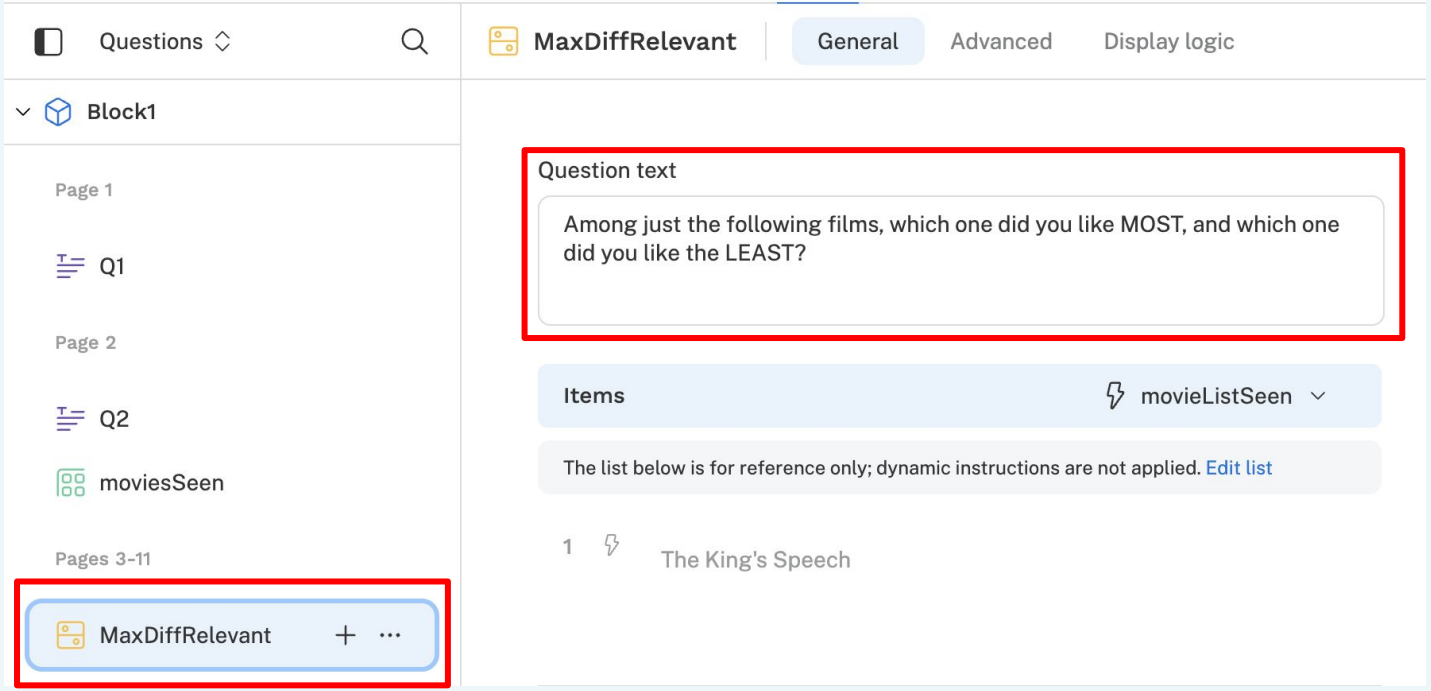


Step 2: Add a Relevant Items MaxDiff exercise

- Add the MaxDiff exercise
- Set it to use the dynamic list
- Set it to be a Relevant Items exercise

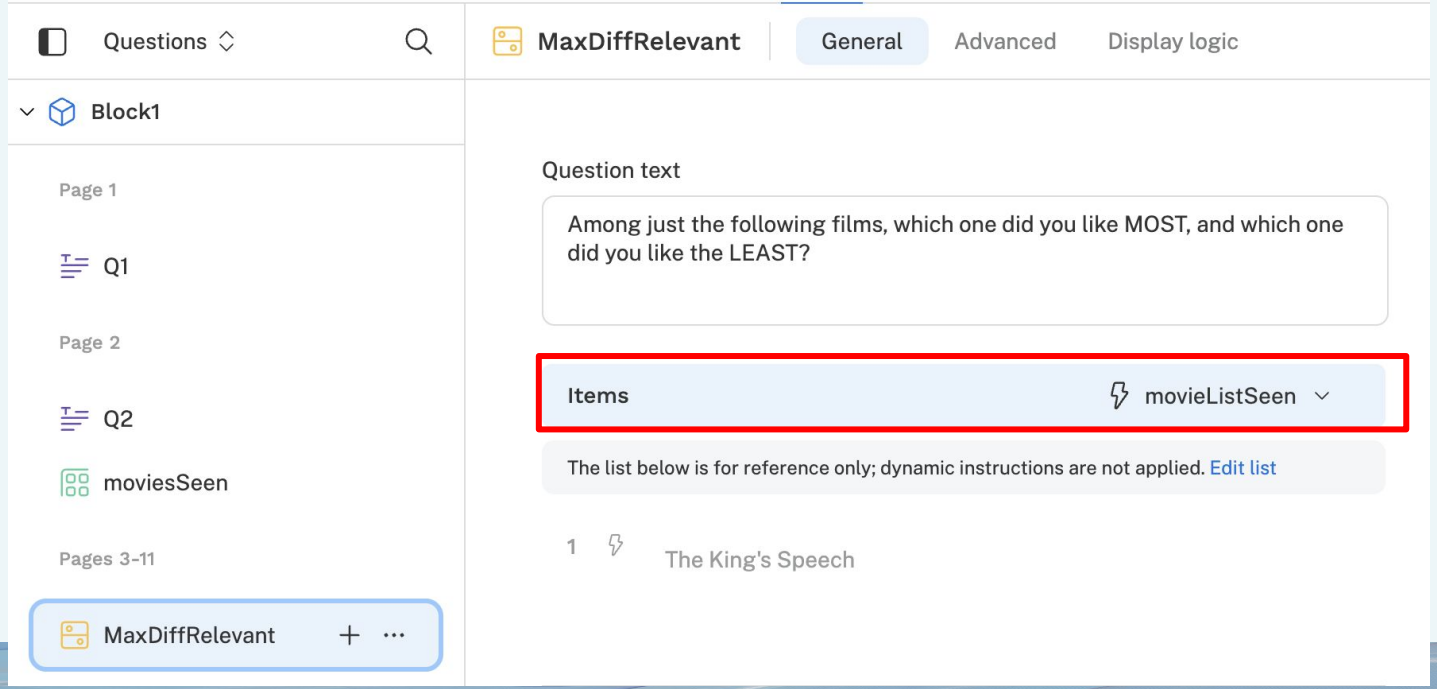
Step 2a: Add a Relevant Items MaxDiff exercise

- **Add the MaxDiff exercise**



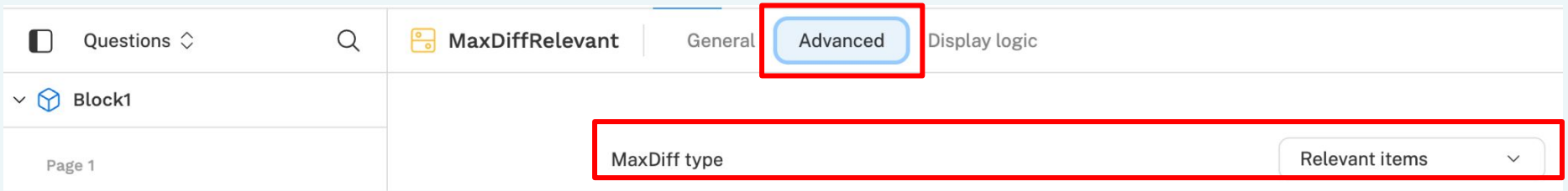
Step 2b: Add a Relevant Items MaxDiff exercise

- Add the MaxDiff exercise
- **Set it to use the dynamic list**



Step 2c: Add a Relevant Items MaxDiff exercise

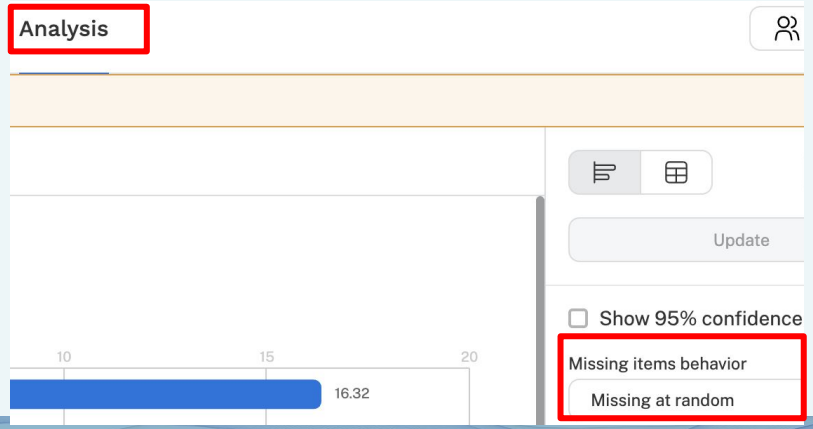
- Add the MaxDiff exercise
- Set it to use the dynamic list
- **Set it to be a Relevant Items exercise**



Estimation Settings Depend on the Path

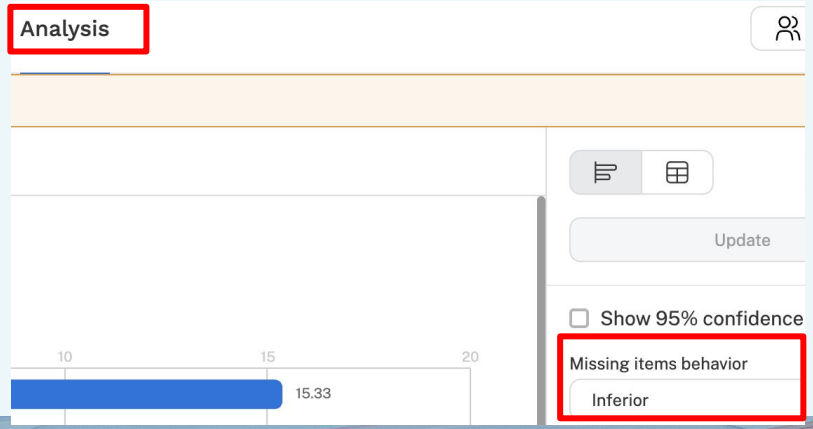
- **Relevance**
  Set HB estimation to use
  "Missing at random"

  Unfamiliar (unselected) items are **not** penalized



- **Importance**
  Set HB estimation to use
  "Missing | Inferior"

  Items below the cut in screening **are** penalized



Results?

Live inspection in Discover

Discussion

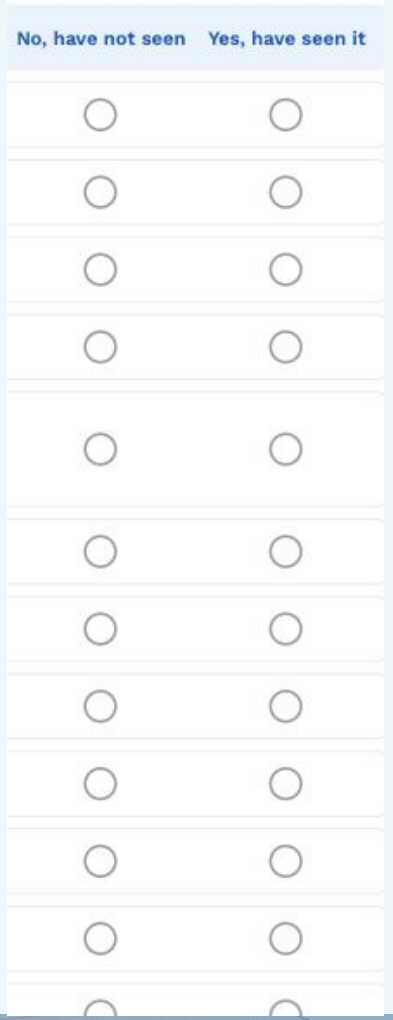
The Tradeoff

Benefits

- **Focused** & more enjoyable MaxDiff
- **Shorter** surveys with fewer tasks
- **Higher quality** data

But …

- ***Screening task*** itself may become long (*next slide*)
- ***Survey platform*** support
  (requires Sawtooth, or custom programming & R code)



What if the screening list is very long?

Problem: too many items to pre-screen them all

Possibilities:

- Break screening into **chunks** so they only rate a few at a time
- **Pre-test** the item list with Bandit MaxDiff or similar and trim it
- Randomly screen **subsets** of items
- **Group items** and programmatically include according to a grouping factors

  For example:

  Check: Role = Security ⇒ Add 8 security items *[scripted]*

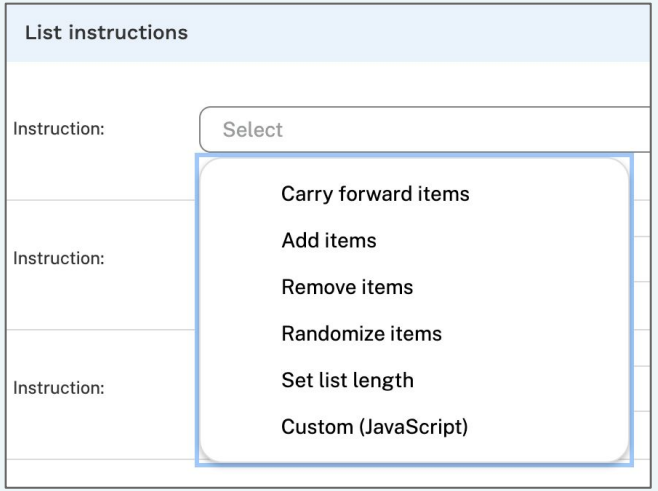
Questions & Options

Q: What if a respondent selects zero or a few items?
A: Discover skips MaxDiff if there are not enough items

Q: Can I force certain items to appear every time?
A: Yes, add instructions for that in the dynamic list tool.

Q: Can I add random items to ensure coverage?
A: Good idea! Use the dynamic list tool to do that.

Q: Can I screen both Relevance *AND* Importance?
A: Reconsider and simplify to use one or the other!
A: or, Use Lighthouse Studio and see the R appendix
A: or, *[experimental]* Anchored MaxDiff + Relevant Items



MOST IMPORTANT POINT

Pre-test, pre-test, pre-test! Live.

It is difficult to get the wording right.
It is easy to make mistakes with lists.

Oh, BTW

Follow us at the Quant UX Association!

- **Virtual & affordable conference** in November
- **Classes year round** in various locations & online

Visit quantuxa.org and join the mailing list

Review & Comparison

When Respondents …	Try …
… understand every concept	Standard MaxDiff
… don't understand one or two concepts	any MaxDiff approach + "Information Acceleration"
… shouldn't rate concepts that don't apply to them	Relevant Items MaxDiff *["relevance" approach]*
… need a shorter survey; item list is too long	Sparse MaxDiff *[limited individual estimates]* Express MaxDiff *[limited individual estimates]* Relevant Items MaxDiff *["importance" approach]*
… are identifying the top items from a very long list	Bandit MaxDiff
… get tired of reporting about items at "the bottom"	Relevant Items MaxDiff *["importance" approach]*

Conclusions for Relevant Items MaxDiff

- **Higher quality data**: Respondents see items that are relevant to them

- **More data:** 2.0 - 3.5x as many implicit choice tasks in our tests

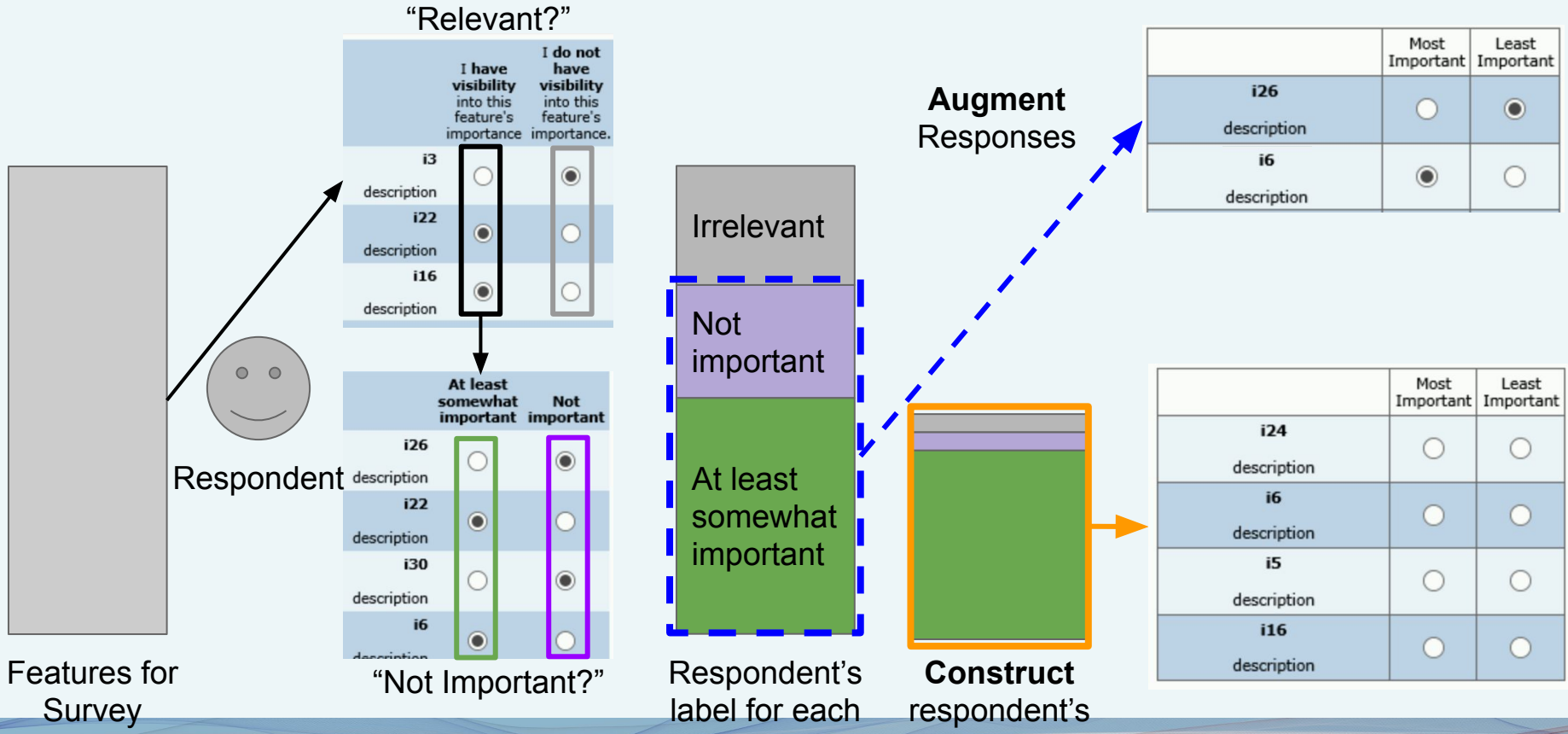
- **Happier respondents**
  - MaxDiff items are more relevant
  - Shorter surveys because respondents consider fewer items

- **References**: Relevant Items reference for Sawtooth Discover
  Original Technical Whitepaper (Chapman & Bahna; *pp. 1-12*)

Thank you! **chris@quantuxa.org** | **quantuxa.org**

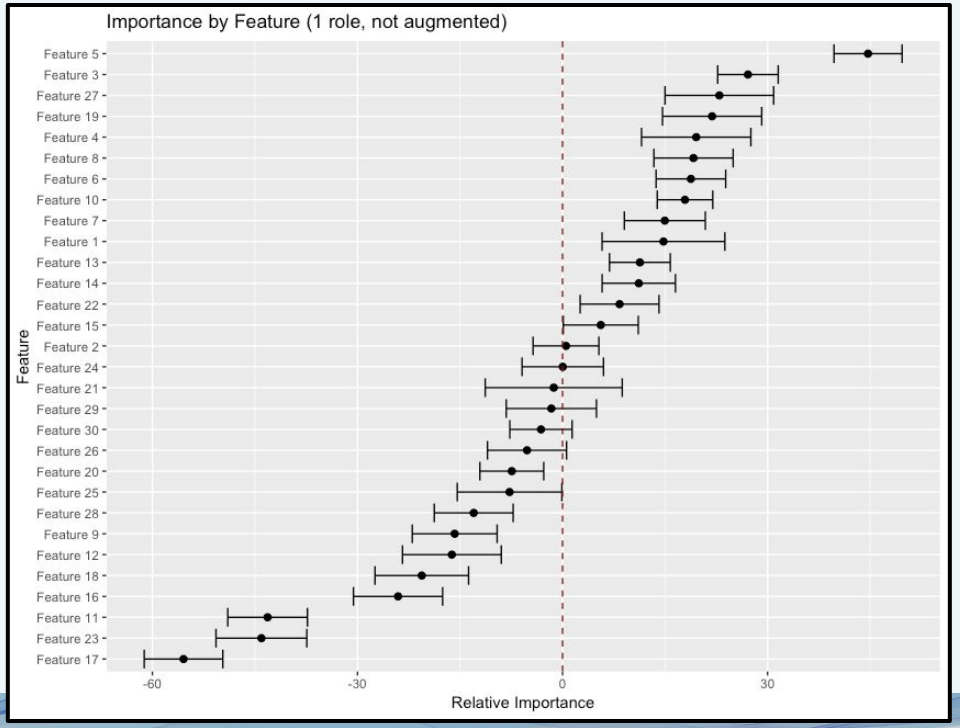
Tabled & miscellaneous

Constructed + Augmented MaxDiff

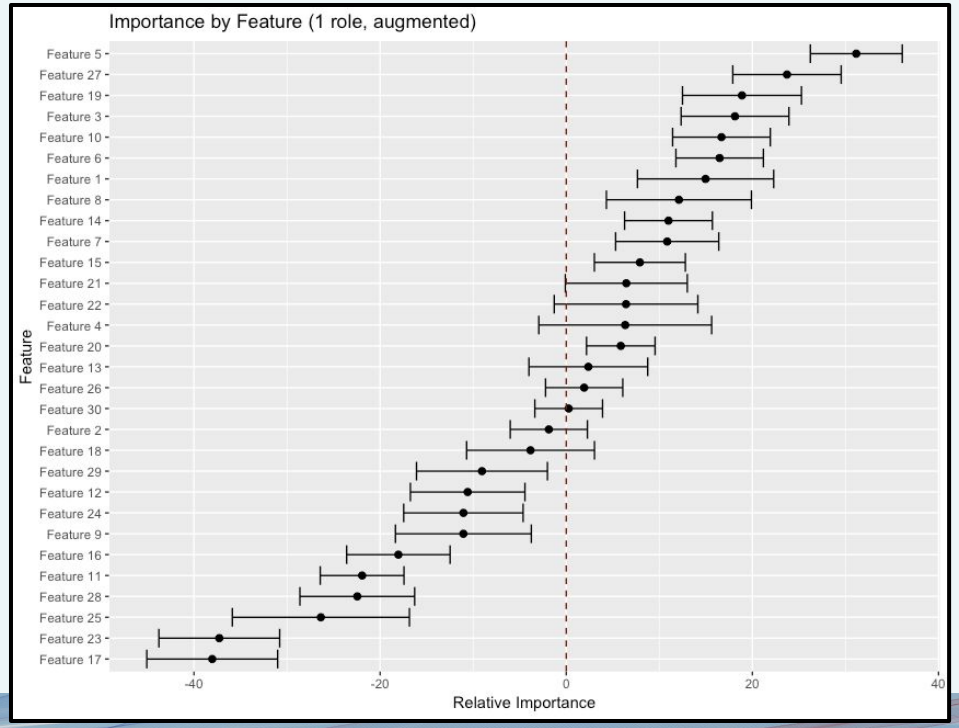


Results: With & Without Augmentation

Before Augmentation

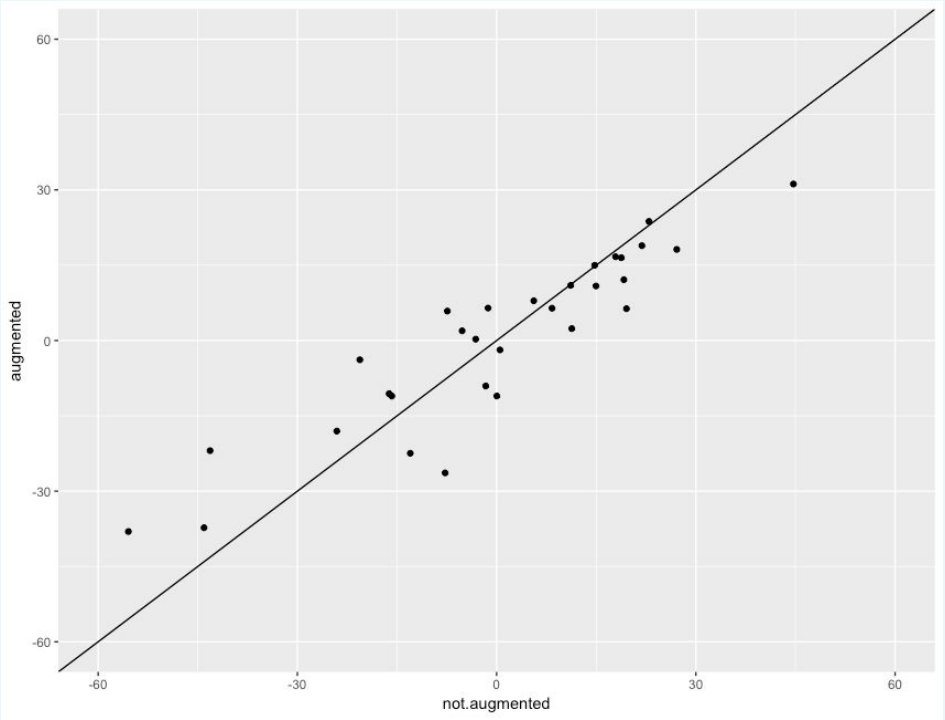


After Augmentation



Estimates with/without data augmentation

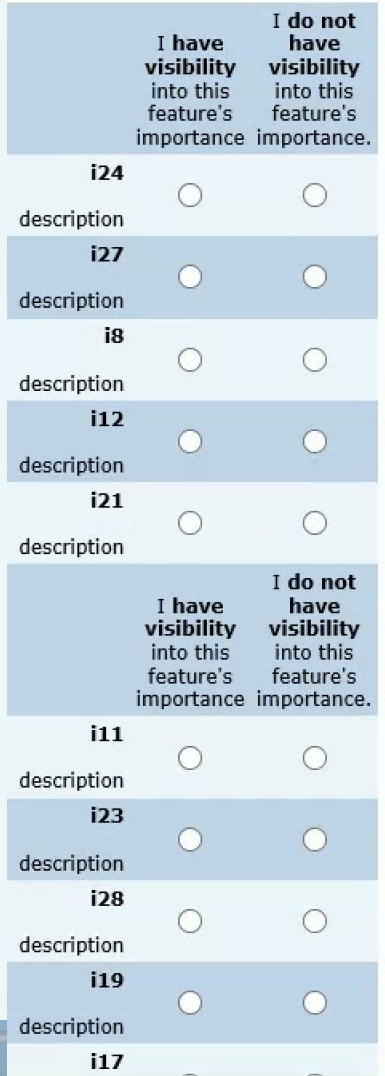
- Modest adjustments to utilities
- Pearson's r = 0.90 between augmented and non-augmented utilities in one study
- Interesting that utilities became more compressed



Design Risks

- **Initial rating for entire list of items, used to construct MaxDiff list**
  Risk: Difficult to answer long list of "what's relevant"
  Solution: Break into chunks; ask a subset at a time; aggregate
  Could chunk within a page (as shown), or several pages.

- Construction of the MaxDiff list
  Risk: Items might be never selected ⇒ degenerate model
  Solution: Add 1-3 random items to the constructed list
  We used: 12 "relevant and important to me" +
  1 "not relevant to me" + 2 "not important"
  ⇒ MaxDiff design with 15 items on constructed list



Open Topics

- If respondents select the items to rate, what does "population" mean?
  Carefully consider what "best" and "worst" mean to you.
  Want: share of preference among **overall population**? ⇒ don't construct
  ... or: share of preference among **relevant subset**? ⇒ construct

- Appropriate number of items -- if any -- to include randomly to ensure coverage
  We decided on 1 "not relevant" and 2 "not important", but that is a guess.
  Idea: Select tasks that omit those items, re-estimate, look at model stability.

- The best way to express the "*Relevant to you*?" and "*Important to you*?" ratings
  This needs careful pre-testing for appropriate wording of the task.

Appendix: R Code

Not required for *relevant* **OR** *important*
... but may be used for simultaneous
implementation of *relevant* **AND** *important*

Alternatively might try Relevant Items + Anchored MaxDiff

Referenced functions available at goo.gl/oK78kw

Features of the R Code

Data sources: Sawtooth Software (CHO file) ⇒ Common format
Qualtrics (CSV file) ⇒ Common format

Given the common data format:

Estimation: Aggregate logit (using `mlogit`)
Hierarchical Bayes (using `ChoiceModelR`)

Augmentation: Optionally augment data for "not important" implicit choices

Plotting: Plot routines for aggregate logit + upper- & lower-level HB

Example R Code: Complete Example

```
> md.define.saw <- list(                                  # define the study, e.g.:
    md.item.k        = 33,                                # K items on list
    md.item.tasks    = 10,                                # num tasks (*more omitted)
...* )
> test.read <- read.md.cho(md.define.saw)                 # convert CHO file
> md.define.saw$md.block <- test.read$md.block            # save the data

> test.aug <- md.augment(md.define.saw)                   # augment the choices
> md.define.saw$md.block <- test.aug$md.block             # update data

> test.hb <- md.hb(md.define.saw, mcmc.iters=50000)       # HB estimation
> md.define.saw$md.hb.betas.zc <- test.hb$md.hb.betas.zc  # get ZC diffs


> plot.md.range(md.define.saw, item.disguise=TRUE)        # plot upper-level ests
> plot.md.indiv(md.define.saw, item.disguise=TRUE) +      # plot lower-level ests
    theme_minimal()                                       # plots = ggplot2 objects
```

Example R Code, Part 0: Define the Study

```
> md.define.saw <- list(
    md.item.k        = 33,
    md.item.tasks    = 10,
... )
```

```
# define the study, e.g.:
# K items on list
# num of tasks
```

Example R Code, Part 1: Data

```
> md.define.saw <- list(                                    # define the study, e.g.:
    md.item.k        = 33,                                 # K items on list
    md.item.tasks    = 10,                                 # num of tasks
... )
```

```
> test.read <- read.md.cho(md.define.saw)                  # convert CHO file
Reading CHO file: MaxDiffExport/MaxDiffExport.cho
Done. Read 407 total respondents.
```

```
> md.define.saw$md.block <- test.read$md.block            # save the data
```

Example R Code, Part 2: Augmentation

```
> md.define.saw$md.block <- test.read$md.block    # save the data
> test.aug <- md.augment(md.define.saw)                  # augment the choices
Reading full data set to get augmentation variables.
Importants: 493 494 495 496 497 498 499 …
Unimportants: 592 593 594 595 596 597 …
Augmenting choices per 'adaptive' method.
Rows before adding: 40700

Augmenting adaptive data for respondent:
6  augmenting: 29 16 25 20 23 9 22 12 5 27 6 11 10 4 26 1 15 2 14 24 31 7 30
13 18 19 3 8 28 21 32 %*% 33 17 ...

Rows after augmenting data: 148660                  # <== 3X data, 1x cost!

> md.define.saw$md.block <- test.aug$md.block         # update data with new choices
```

Example R Code, Part 3: HB

```
> md.define.saw$md.block <- test.aug$md.block          # update data with new choices

> test.hb <- md.hb(md.define.saw, mcmc.iters=50000)    # HB

MCMC Iteration Beginning…
Iteration  Acceptance   RLH     Pct. Cert.   Avg. Var.   RMS     Time to End
      100  0.339        0.483   0.162        0.26        0.31    83:47
      200  0.308        0.537   0.284        0.96        0.84    81:50 ...

> md.define.saw$md.hb.betas.zc <- test.hb$md.hb.betas.zc  # zero-centered diffs
```

Example R Code: Plots

```
# upper-level
> plot.md.range(md.define.saw,
                item.disguise=TRUE)
```

```
# lower-level
# note we can add ggplot2 functions
> plot.md.indiv(md.define.saw,
                item.disguise=TRUE) +
  theme_minimal()
```