Statistical Perceptions Serverless Survey

Contents

For Researchers

For Developers

These documents outline how to use for researchers and for potential contributors.

Details that are for contributors denoted with the `</>` icon, when mixed into a page with other information it will be in a dropdown.

Working with Configuration Files

A configuration file is yaml

YAML Format

There are two supported formats for the configuration file

No shared parameters

This file is setup like:

```yaml
- question_id: unique_id_for_q1
  name_of_var1_for_q1: value_for_var1_q1
  name_of_var2_for_q1: value_for_var2_q1
  figure_values:
    name_of_fig_var1_for_q1: value_for_fig_var1_for_q1
- question_id: unique_id_for_q2
  name_of_var1_for_q2: value_for_var1_q2
  name_of_var3_for_q2: value_for_var3_q2
  name_of_var3_for_q2: |
        value_for_var3_q2_line_1
        value_for_var3_q2_line_3
        value_for_var3_q2_line_3
  figure_values:
    name_of_fig_var1_for_q2: value_for_fig_var1_for_q2
```

Notes:

- Each `name_of_varX_for_qY` has to be a variable that the `make_question_page` function accepts
- any variables not specified will get the default value as stated in the documentation
- `figure_values` is a special variable that takes more variables. the names of the fig variables are defined for each question
- the variables can be in any order
- `question_id` must be stated, there is no default value for it
- only the first variable for each question gets a `-`
- `name_of_var3_for_q2` is an example of how to format a long value if you do not leave it on a single line.

Some come from the question and others are for the pate

Shared parameters

To share values across question it can be set up so that the top level is a single entry with two keys (`shared` and `unique`) where the `shared` key includes the parameter values that are to be applied to all questiona and `unique` includes a list defining individiual questions as above. Any values defined in both, the `unique` will overwrite the `shared` value.

For example:

```
shared:
  name_of_var1_shared: value_for_var1_shared
  name_of_var2_shared: value_for_var2_shared
  figure_values:
    name_of_fig_var2_shared: value_for_fig_var2_shared
unique:
- question_id: unique_id_for_q1
  name_of_var4_for_q1: value_for_var4_q1
  figure_values:
    name_of_fig_var1_for_q1: value_for_fig_var1_for_q1
- question_id: unique_id_for_q2
  name_of_var4_for_q2: value_for_var4_q2
  figure_values:
    name_of_fig_var1_for_q2: value_for_fig_var1_for_q2
```

This is equivalent to (but, for large number of questions, more compact than):

```
- question_id: unique_id_for_q1
  name_of_var4_for_q1: value_for_var4_q1
  name_of_var1_shared: value_for_var1_shared
  name_of_var2_shared: value_for_var2_shared
  figure_values:
    name_of_fig_var2_shared: value_for_fig_var2_shared
    name_of_fig_var1_for_q1: value_for_fig_var1_for_q1
- question_id: unique_id_for_q2
  name_of_var4_for_q2: value_for_var4_q2
  name_of_var1_shared: value_for_var1_shared
  name_of_var2_shared: value_for_var2_shared
  figure_values:
    name_of_fig_var2_shared: value_for_fig_var2_shared
    name_of_fig_var1_for_q2: value_for_fig_var1_for_q2
```

Configuring your study

To configure the study you will need the urls to each follow-up survey. They do not have to be fully configured first though.

Each question is a single page with a figure on it

Page level Settings

These settings control the rest of the question page, other than the figure.

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[1], line 1
----> 1 import ssbuilder
      2 # import markdown
      3 from IPython.display import Markdown

File /opt/hostedtoolcache/Python/3.8.18/x64/lib/python3.8/site-packages/ssbuilder/__init__.py:4
      2 from .single_normal_curve import NormalCurveSlider
      3 from .tradeoff_questions import TradeoffLine, TradeoffBar
----> 4 from .utils import  md_params, check_query_length
      5 from .utils import merge_dir_csvs

File /opt/hostedtoolcache/Python/3.8.18/x64/lib/python3.8/site-packages/ssbuilder/utils.py:159
    146         est_chars += prefix_length
    148     return est_chars
    151 @click.command()
    152 @click.argument('folder', type=click.Path(exists=True))
    153 @click.option('-m', '--merge-on', default='id', multiple=True)
    154 @click.option('-h', '--header', default=0)
    155 @click.option('-s', '--skip-row', multiple=True, type=int)
    156 @click.option('-o','--out-name',default=None)
    157 @click.option('-v','--verbose',is_flag=True)
    158 @click.option('-c','--complete-only',is_flag=True)
--> 159 def merge_dir_csvs(folder,merge_on='id',out_name=None, header=0, skip_row_list=[1,2],
    160                    verbose=False, skip_row =None,complete_only=False):
    161     '''
    162     merge all csvs in a folder into a single CSV file
    163
  (...)
    179         print extra information out
    180     '''
    181     if skip_row:

File /opt/hostedtoolcache/Python/3.8.18/x64/lib/python3.8/site-packages/click/decorators.py:373, in option.<
    372 def decorator(f: FC) -> FC:
--> 373     _param_memo(f, cls(param_decls, **attrs))
    374     return f

File /opt/hostedtoolcache/Python/3.8.18/x64/lib/python3.8/site-packages/click/core.py:2536, in Option.__init_
    2533     help = inspect.cleandoc(help)
    2535 default_is_missing = "default" not in attrs
-> 2536 super().__init__(param_decls, type=type, multiple=multiple, **attrs)
    2538 if prompt is True:
    2539     if self.name is None:

File /opt/hostedtoolcache/Python/3.8.18/x64/lib/python3.8/site-packages/click/core.py:2151, in Parameter.__in
    2149         check_default = next(_check_iter(check_default), None)
    2150     except TypeError:
-> 2151         raise ValueError(
    2152             "'default' must be a list when 'multiple' is true."
    2153         ) from None
    2155 # Can be None for multiple with empty default.
    2156 if nargs != 1 and check_default is not None:

ValueError: 'default' must be a list when 'multiple' is true.
```

> ⚠ **Warning**
>
> You cannot use `end` as a question ID, or `end.html` as an output file name

Question Text

Questions' text goes in the page level parameter `question_text`. It may include markdown formatting to be rendered.

Some key examples:

- `**bold**`
- `*italic*`
- `- bulleted text`
- `[link display text](url/for/link)`
- a blank space at the end of aline will make a new paragraph

Figure specific Settings

These settings vary by question type and the options are detailed on Question Types

Build Level Options

Serverless Survey has some settings that are for a whole study or about how to process the configuration file.

These are set as CLI arguments if you build offline. If you use our template repo, you will have a set of options for these controls in your actions tab.
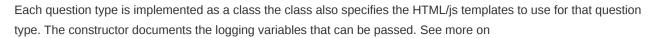
This is where the path to save the output html files are set as well as the url of the hosted site for generating the instructions. There is also the option to generate only a fragment or to put all questions on a single page (eg for IRB review or paper supplemental materials).

```bash
%%bash
ssgeneratehtml --help
```

Question Types

> 🔔 **Question Implementation**   ⌄
>
> Each question type is implemented as a class the class also specifies the HTML/js templates to use for that question
> type. The constructor documents the logging variables that can be passed. See more on

All questions must have certain parameters:

- `question_text`
- `question_id`

the `question_text` can include markdown formatting which will be rendered with the markdown package

Skip to main content

```
Cell In[1], line 1
----> 1 import ssbuilder
      2 # import markdown
      3 from IPython.display import display, Markdown

File /opt/hostedtoolcache/Python/3.8.18/x64/lib/python3.8/site-packages/ssbuilder/__init__.py:4
      2 from .single_normal_curve import NormalCurveSlider
      3 from .tradeoff_questions import TradeoffLine, TradeoffBar
----> 4 from .utils import  md_params, check_query_length
      5 from .utils import merge_dir_csvs

File /opt/hostedtoolcache/Python/3.8.18/x64/lib/python3.8/site-packages/ssbuilder/utils.py:159
    146         est_chars += prefix_length
    148     return est_chars
    151 @click.command()
    152 @click.argument('folder', type=click.Path(exists=True))
    153 @click.option('-m', '--merge-on', default='id', multiple=True)
    154 @click.option('-h', '--header', default=0)
    155 @click.option('-s', '--skip-row', multiple=True, type=int)
    156 @click.option('-o','--out-name',default=None)
    157 @click.option('-v','--verbose',is_flag=True)
    158 @click.option('-c','--complete-only',is_flag=True)
--> 159 def merge_dir_csvs(folder,merge_on='id',out_name=None, header=0, skip_row_list=[1,2],
    160                    verbose=False, skip_row =None,complete_only=False):
    161     '''
    162     merge all csvs in a folder into a single CSV file
    163
   (...)
    179         print extra information out
    180     '''
    181     if skip_row:

File /opt/hostedtoolcache/Python/3.8.18/x64/lib/python3.8/site-packages/click/decorators.py:373, in option.<
    372 def decorator(f: FC) -> FC:
--> 373     _param_memo(f, cls(param_decls, **attrs))
    374     return f

File /opt/hostedtoolcache/Python/3.8.18/x64/lib/python3.8/site-packages/click/core.py:2536, in Option.__init_
    2533     help = inspect.cleandoc(help)
    2535 default_is_missing = "default" not in attrs
-> 2536 super().__init__(param_decls, type=type, multiple=multiple, **attrs)
    2538 if prompt is True:
    2539     if self.name is None:

File /opt/hostedtoolcache/Python/3.8.18/x64/lib/python3.8/site-packages/click/core.py:2151, in Parameter.__i
    2149         check_default = next(_check_iter(check_default), None)
    2150     except TypeError:
-> 2151         raise ValueError(
    2152             "'default' must be a list when 'multiple' is true."
    2153         ) from None
    2155 # Can be None for multiple with empty default.
    2156 if nargs != 1 and check_default is not None:

ValueError: 'default' must be a list when 'multiple' is true.
```

Normal Curve Questions

This question has two normal curves, one moves and one does not.

Example

Settings

Tradeoff Questions

this question type trades off between two two extremes over a number of models in the middle

Both formats accept the same dataset files, where each row represents one bar heigh for one slider location. The values should be compatible with deisplay in the figures, for example converting .74 to 75 to display percentages. There may be extra columns that are not used.

An example (that is used in the example plots):

Line Graph Tradeoff

This uses lines to show all metrics on one set of axes.

Example

Settings

Use `figure_type: TradeoffLine` with the following parameters for use in `figure_values`:

Bar Graph Tradeoff

This uses a set of bar graphs.

Example

Settings

Use `figure_type: TradeoffBar` with the following parameters for use in `figure_values`

Coordinating with Qualtrics

> ⚠ **Important**
>
> This is the base documentation that may need updates

Basic use

1. the first qualtrics survey sends embedded data via forwarding only
2. middle ones receive data and send data
3. the last one receives data only

Sending the ID from Qualtrics

1. Add an embedded data block with the identifier to forward (eg panel ID or Response ID)
2. redirect end of survey to a url
3. embed the response id in the forwarding url:

Template

```
https://statistical-perceptions.github.io/IdentiCurve/<question_out_html_file>.html?id=<qualtrics piped text
```

Example

```
https://statistical-perceptions.github.io/sample-nobackend/?id=${e://Field/ResponseID}
```

in this case, I only had one question page so there is no question_id set and I used the ResponseID feild.

note:

- if needed, we can pass more than a single unique identifier on, but that requires code changes

Recieving Data into Qualtrics

1. set up embedded data as the first block on the workflows tab. set the variables as per the instructions output
2. (if applicable) use piped text to refer to those values in the question text
3. (optional) add a branch after the embedded data to have people skip the survey "if id is Equal to demo"

- getting data from url

Branching to different ss questions based on qualtrics answers

1. on the survey flow tab add a branch
2. set the condition to be based on a question
3. add the url to forward to
4. (if applicable) add the variable value to the forward url

Template

```
https://statistical-perceptions.github.io/IdentiCurve/<question_out_html_file>.html?id=<qualtrics piped text
```

Example

```
https://statistical-perceptions.github.io/sample-nobackend/?id=${e://Field/ResponseID}&group=w
```

in this case, I only had one question page so there is no question_id set and I used the ResponseID feild.

Branching to different ss questions based on embedded data

1. make sure the embedded data received includes the pass through variable

Skip to main content

3. set the condition to be based on embedded data and choose the values
4. add the url to forward to
5. (if applicable) add the variable value to the forward url

Template

```
https://org-or-user.github.io/repo/question_out_html_file.html?id=<qualtrics piped text>&var=value
```

Example

```
https://statistical-perceptions.github.io/IdentiCurve/nc2t1w.html?id=${e://Field/ResponseID}&group=${e://Fiel
```

Semi-automatic forwarding to different qustions

In order to have a survey where there are different versions of the questions for different participants depending on a response of a question, set up a question in qualtrics where the response will be the "logic variable." Then make it so that the logic variable is passed along with the ID and its values appear in the url so that the variable in piped text can make it work.

1. set up configurations so that the logic variable values are in the question ids (or html file names)
2. make sure embedded data received includes the variable used for logic
3. set up the url like (for `group` as the logic variable)

```
https://ghorg.github.io/repo/page_url${e://Field/logicvariable}.html?id=${e://Field/ResponseID}&group=${e://I
```

For example if the following is in qualtrics as the forward url

```
https://statistical-perceptions.github.io/IdentiCurve/nc2t1${e://Field/group}.html?id=${e://Field/ResponseID]
```

when `group=a` the url will become

```
https://statistical-perceptions.github.io/IdentiCurve/nc2t1a.html?id=RH2904&group=a
```

Qualtrics Help

- Passing with Query Strings: about the url
- Piped text: about getting and formatting variables for the url or for using responses to modify qualtrics questions
- embedded data: for storing data to send and receive it

Question Types

Skip to main content

Each question type is implemented as a class the class also specifies the HTML/js templates to use for that question type. The constructor documents the logging variables that can be passed.

Normal Curve Questions

This question has two normal curves, one moves and one does not.

Trade Off Questions

this quesiton typ trades off between two two extremes over a number of models in the middle

Configuration Details

Page level

Building from config