Information Frictions, Overconfidence, and Learning: Experimental Evidence from a Floodplain. Documentation.

Sofia Badini

12 November 2024

CONTENTS

CHAPTER

ONE

INTRODUCTION

1.1 This project

This is the documentation for the replication package of the paper "Information frictions, overconfidence, and learning: Experimental evidence from a floodplain" (PDF, pre-analysis plan), by Sofia Badini.

Abstract. I use an online experiment to study whether offering information to floodplain residents is sufficient to change their perceived risk exposure and demand for insurance. The participants are offered information on the flood risk profile at their address and on the rules over compensation of flood damages. I find that respondents tend to misperceive their risk category according to publicly available flood maps, but express high levels of confidence in their guesses. When not prompted to engage with the information they are offered, one third of them read nothing. Respondents who are asked to read information on their risk profile tend to stop reading any further and report a lower willingness-to-pay for insurance. However, this effect does not seem to be driven by respondents learning more from the information they are provided with, at least based on how they update their beliefs. Instead, I find suggestive evidence of backlash to information among residents of high risk areas and individuals who initially underestimated their risk category.

1.2 Structure of the project

This project uses the econ-project-templates by Hans-Martin von Gaudecker, which in turn use pytask, a workflow management system for reproducible data analysis. Therefore, this project is fully and automatically reproducible conditional on having conda, a modern LaTeX distribution (e.g. MikTex), and the text editor VS Code installed. You will also need to install R. Please see the documentation of econ-project-templates for an overview of the dependencies that you need to install (and for detailed explanations on why the template looks the way it does).

The minimum information that you need to know is the following:

- The important scripts in this project are stored in the folder *src*, specifically in the sub-folders *experiment_floodplain/data_management*, *experiment_floodplain/analysis*, and *experiment_floodplain/final*.
- Upon replication of the project, a new folder named *bld* will appear in the root folder (where *src* is). *bld* contains all the output of this project.
- *bld* also contains this website, under the sub-folder *documentation*. The documentation is also produced in PDF version: it is the file `documentation.pdf` (under *bld/documentation/latex*, a copy is in the root folder).
- **The single most important file produced by this replication package is the .pdf** `paper_results_replication.pdf`, which reproduces all the figures and tables in the paper (under *bld/latex*, a copy will appear in the root folder).

1.3 Issues with data sharing

This project uses survey data collected with the platform Qualtrics. The survey respondents were contacted via postcards mailed to their addresses, as the Dutch government maintains a database of publicly available geocoded addresses for the entire country. Each postcard included a unique code, which served both as a password to log-in to the online survey, and as an identifier (survey respondents were displayed address-specific information). I additionally determine whether each address was flooded in July 2021 using a dataset of postcode-level data provided by the government (which is not available to the public).

These two datasets will be shared once this paper is published, and will be delivered as a dataset in .csv format (named `original_survey_data.csv` and `rvo_pc6.csv`). The original data collected via Qualtrics will be anonymized and will include only the subset of the original Qualtrics data that I used for the analysis.

> **Note**
>
> **The analysis will be automatically performed on the real data if the .csv files are placed in the folder** *src/experiment_floodplain/data*. By default, this replication package uses fake data, based on the original dataset and created with the Python library Synthetic Data Vault (SDV), in place of the real survey data. Moreover, by default the flood status is attributed at random. See also the section Data Management.

1.4 Replication

To reproduce, first navigate to the root of the project (where *src* is). Then, open your terminal emulator and run, line by line:

```
conda create --name condalock python # create empty environment
conda activate condalock # activate environment
pip install conda-lock # install conda-lock
```

The lines above install conda-lock in a clean environment. conda-lock is a Python package to generate reproducible conda environments – i.e., directories containing the packages needed to run code. To install the environment for this project, after the lines above you additionally need to run:

```
conda-lock install --name experiment_floodplain environment.conda-lock.yml #␣
↪create project environment with conda-lock
conda activate experiment_floodplain # activate project environment
```

Finally, to execute the project run:

```
pip install -e . # install the project
pytask # execute the project
```

1.4.1 If your machine runs Windows11

... you may want to use the file `environment_win11.conda-lock.yml` instead of `environment.conda-lock.yml`. The former file uses an older version of `pytask`, as the newest releases appear to... behave capriciously with Windows11.

CHAPTER

TWO

DATA MANAGEMENT

Documentation of the code in *experiment_floodplain/data_management*. The data management process consists of three tasks:

- **Task 1**: In the folder *task_download_data*. Downloads the data necessary to replicate the project.
- **Task 2**: In the folder *task_create_target_population*. Creates the target population, i.e., creates a dataset of the addresses that were eligible for being contacted (what is called "full sample" throughout the paper). The result is the dataset `target_population.csv` in *bld/data*.
- **Task 3**: In the folder *task_sample_survey_recipients*. Creates a dataset of the addresses that were in fact contacted. The result is the dataset `survey_recipients.csv` in *bld/data*.

> **Note**
>
> The dataset `survey_recipients.csv` needs `rvo_pc6.csv` to be reproduced. The latter will be shared upon publication of this paper. By default, this project creates a slightly different version of `survey_recipients.csv` to illustrate how the code works. I provide the (anonymized) dataset of the people who were in fact contacted under *bld/replication_data/SURVEY*.

- **Task 4**: In the folder *task_clean_survey_data*. Cleans the original Qualtrics survey data. The result is the dataset `survey_data.csv` in *bld/data*.

> **Note**
>
> Because of data protection concerns, the original dataset –which is anonymized and contains only a subset of the original Qualtrics data– will be shared upon publication of this paper. By default, this replication package uses fake data created based on the original dataset with the Python library Synthetic Data Vault (SDV). See also the Introduction of this replication package.

2.1 Script for Task 1: downloading the data

2.1.1 Script `task_download_data.py`

This script downloads publicly available data necessary for the replication of this paper (available via 4TU.ResearchData) and saves them to *bld/replication_data*. The datasets are the following:

- **BAG data**: The *BAG* folder contains data about all the (full) addresses in the Netherlands, derived from the Addresses and Buildings Key Registry (*Basisregistraties Adressen en Gebouwen*, or BAG) and acquired in .csv format via Geotoko, in July 2022.

  The .csv file provides a variety of building-related information, such as the function of each (dwelling within a) building, its perimeter and size, and its year of construction. The dataset codebook (in Dutch) is included in the folder.

- **ENW data**: The *ENW* folder contains shapefiles representing those areas of the Southern regions of The Netherlands (Limburg and North Brabant) that were flooded in July 2021. These maps have been shared by the ENW (*Expertise Netwerk Waterveiligheid*), the association of Dutch flood protection specialists, via the 4TU.ResearchData repository.

  The maps are based on best available information collected via aerial photography and during fieldwork at the flooded sites, and include realized inundation extent (in the sub-folders *floodsGeul*, *floodsMaas*, and *floodsRoer*), areas evacuated via emergency ordinances (in the sub-folder *evacuations*), and locations where incidents to water management infrastructure occurred (in the sub-folder *incidents*).

- **Risicokaart data**: The *RISICOKAART* folder contains shapefiles of the flood maps developed for the European Floods Directive (ROR2) delivered at the end of 2019, obtained in October 2021 by contacting lbo@risicokaart.nl. The layer can be viewed by the general population via an interactive app available at the Risicokaart website.

  Flood maps have been developed for four different scenarios: Large probability (10% risk in any given year), medium probability (1% risk in any given year), small probability (0.1% risk in any given year, or 1 in 1,000 years flood), and scenario of extraordinary events (0.01% risk in any given year, or or 1 in 10,000 years flood). Data on predicted flood extents under each scenario, represented by polygons classified by maximum water depth, are in the folder *floodmaps2019*.

  The folder *otherinfo* contains additional data on the location of primary water defences (which protects The Netherlands against floods from major rivers and the sea) and of regional water defences (which protects The Netherlands against floods from smaller rivers).

- **Survey data**: The *SURVEY* folder contains the dataset of survey recipients without identifying information (e.g., no addresses), named `survey_recipients.csv`, and a synthentic dataset named `synthetic_survey_data.csv` generated using Synthetic Data Vault (SDV) and based on the original Qualtrics data.

  > **Note**
  >
  > This project automatically uses the real survey respondents' data if the corresponding file is placed under *src/experiment_floodplain/data*. See also the Introduction of this documentation.

  The *SURVEY* folder contains two more datasets: `original_variables.csv`, which details which original variables will be cleaned (where by original I mean the "raw" data collected by Qualtrics), and `final_variables.csv`, which details which variables will appear in the final dataset.

2.1.2 Script `generate_synthetic_data.py`

In the folder *generate_synthetic_data*.

This script generates synthetic survey data based on the original one, using the python package Synthetic Data Vault (this package is not part of the environment and needs to be installed separately, in case you want to run this script). The resulting dataset, `synthetic_survey_data.csv`, is downloaded to *bld/replication_data/SURVEY*.

> **Note**
>
> This task is not part of the project workflow.

2.2 Scripts for Task 2: creating the target population

2.2.1 Script `task_clean_BAG_data.py`

This script cleans the raw data in *bld/replication_data/BAG*, creates a DataFrame and saves it as .csv file in *bld/data/BAG*.

The final dataset contains full addresses and some building-related characteristics for all (in-between, detached, semi-detached, corner or two-under-one-roof) houses in Limburg that, as of 2022:

- Belong to buildings "in use";
- Are used for residential purposes;
- Have a one-to-one relationship with their building (e.g. no houses that share the same building with shops or other spaces);
- Have unique addresses based on street, housenumber, and postcode.

These restrictions are supposed to facilitate reaching our sample via survey invitation letters and to avoid ambiguities on the effective flood exposure of the survey respondents (for example, I drop apartments because it is not possible to tell on which floor they are located).

2.2.2 Script `task_clean_ENW_data.py`

This script cleans the raw data in *bld/replication_data/ENW*, creates a GeoDataFrame and saves it as geopackage (.GPKG) in *bld/data/ENW*.

The final dataset contains four columns:

- `FEATURE`, indicating the event type (incident, evacuation, or flood);
- `geometry`, containing the shapely Polygons or shapely Point associated with the `FEATURE`;
- `Status`, which is always 1 except for some evacuations (as some areas were not formally evacuated);
- `COMMENT` containing additional information (e.g. incident acronym as found in ENW " Hoogwater 2021 Feiten en Duiding" report).

2.2.3 Script `task_assign_flood_exposure.py`

This script assigns each address in the clean BAG dataset in *bld/data/BAG* to (1) flooded areas in July 2021 according to ENW data, (2) 6-digits postcodes from which claims for house-related and house content-related damages were filed to the RVO after the floods in July 2021, and (3) flood probability and maximum water depth according to the Risicokaart flood maps. The final dataset is saved as a .csv file named `full_sample.csv` in *bld/data*.

The RVO data are necessary to create the indicator variable `FLOODED`, which in turn is necessary for the sampling of survey recipients. The RVO data are not available to the public, but will be available to researchers as a .csv file named `rvo_pc6.csv` once the paper is published. See also the Introduction of this documentation.

Note

This project automatically uses the RVO data if the .csv file is placed under *src/experiment_floodplain/data*. Otherwise, the values of the variable `FLOODED` are generated at random.

2.3 Scripts for Task 3: sampling the survey recipients

2.3.1 Script `task_add_survey_weights_to_full_sample.py`

This script adds the survey weights to the full sample of addresses eligible to be contacted. I oversample in areas affected by the July 2021 floods. The resulting dataset is `full_sample_with_weights.csv` in *bld/data*.

Note

The sampling weights will be the true ones only if `rvo_pc6.csv` is placed under *src/experiment_floodplain/data*. If not, the variable `FLOODED` (on which the sampling strategy depends) will be assigned at random, thus affecting the sampling weights. This does not really matter to reproduce the results in the paper, because I provide the dataset of true survey recipients (without identifying information) in *bld/replication_data/SURVEY*.

2.3.2 Script `task_create_pilot_sample.py`

This script creates the Qualtrics contact list for the pilot sample. This script produces three .csv files, all saved in *bld/data/survey_recipients*:

- `pilot_sample.csv`, which contains the sampled addresses.
- `pilot_balance.csv`, which compares a number of average covariate values between the population and (both unweighted and weighted) sample.
- `pilot_qualtrics.csv`, which has the same dimensions of `pilotSample.csv` but only contains variables needed for the qualtrics survey, including unique codes to be used as authenticators.

2.3.3 Script `task_create_main_sample.py`

This script creates the Qualtrics contact list for the main sample. This script produces three .csv files, all saved in *bld/data/survey_recipients*:

- `main_sample.csv`, which contains the sampled addresses.
- `main_balance.csv`, which compares a number of average covariate values between the population and (both unweighted and weighted) sample.
- `main_qualtrics.csv`, which has the same dimensions of `main_sample.csv` but only contains variables needed for the qualtrics survey, again including unique codes to be used as authenticators.

2.3.4 Script `task_create_survey_recipients_dataset.py`

This script adjusts the survey recipients dataset to reflect the addresses that were in fact contacted *after* data collection. In particular:

- Add indicator variable for incorrect addresses (where mail was rejected).
- Add indicator variable for whether the address was reached during data collection, and at what stage.

The resulting dataset is saved as survey_recipients.csv in *bld/data*.

Note

This dataset is identical to survey_recipients.csv in *bld/replication_data/SURVEY* only if the dataset *rvo_pc6.csv* is placed under *src/experiment_floodplain/data*.

2.4 Scripts for Task 4: cleaning the survey data

2.4.1 Sript `task_clean_survey_data.py`

This script cleans the original Qualtrics data.

2.5 Auxiliary modules

2.5.1 Script `merge_data.py`

In the folder *task_create_target_population*. This script contains functions to merge data from various sources.

`add_floodmaps_indicators`(*df*, *waterdepth_column*, *floodrisk_column*)

Derive indicators of flood risk and maximum water depth from existing columns named waterdepth_column and floodrisk_column in Pandas.DataFrame df.

`assign_address_to_admin_region`(*addressesGDF*, *admin_path*, *admin_level*)

Assign each address in addGDF to specified administrative region.

I do a spatial merge (i.e. I assign each address to the administrative region that contains the shapely.Point associated with such address, as recovered from the lat and lon columns) instead of merging on administrative level's code.

The reason is that I am forced to use data from different years, and any administrative level of a certain address (but especially the low-level ones) can change over time. For example, postcodes can be reassigned to a different buurt or wijk.

The function returns a dataframe which contains a column equal to the index of the merged administrative level dataframe.

Parameters

- **addressesGDF** (*geopandas.DataFrame*) – Dataframe of addresses. Needs to have an active geometry column to perform the spatial merge.
- **admin_path** (*str or Pathlib object*) – Path to geopandas.DataFrame of administrative region. Needs to have an active geometry column to perform the spatial merge.
- **admin_level** (*str*) – Level of administrative region. Can be "pc6", "pc5", "buurt", "wijk", or "gemeente".
- **path** (*str or Pathlib object*) – Path to save the final dataframe to.

Returns

geopandas.DataFrame

assign_admin_geometries_to_addresses(*df*, *admin_path*, *admin_level*)

Merge each row of df to the appropriate administrative region boundary in the admin_path geopandas.GeoDataFrame.

Parameters

- **df** (*pandas.DataFrame*) – Dataframe of BAG addresses, with index_{admin_level} columns (needed to merge on) and either LAT and LON columns or geometry column. It is supposed to be the dataframe saved as population.csv in *bld.data*.
- **admin_path** (*str or Pathlib object*) – Path to administrative dataset to load. It is supposed to be one of the datasets in *bld.data.CSB*.
- **admin_level** (*str*) – Administrative level of dataset, for example "pc5", "pc6", "buurt", "wijk".

Returns

geopandas.GeoDataFrame

assign_flood_risk(*key*, *RKdatadict*, *bagGDF*, *depthDict*, *max_distance=500*)

Assign each row in bagGDF to flooded or non-flooded status according to given scenario of Risicokaart flood maps.

Parameters

- **key** (*str*) – Refers to Risicokaart flood scenario.
- **RKdatadict** (*dict*) – Dictionary whose keys refer to the Risicokaart scenarios, and whose values refer to the path to the Risicokaart flood maps.
- **bagGDF** (*geopandas.Dataframe*) – BAG dataset. Needs to have geometry column.
- **depthDict** (*dict*) – Dictionary of water depth labels.

Returns

geopandas.Dataframes

drop_duplicates_with_lower_waterdepth(*gdf*, *duplicated_row*, *key*)

For geodataframe gdf, corresponding to a key scenario, drop all the rows with the index of duplicated_row besides the one with the highest value for NEAR_WATERDEPTH_{key}_INDICATOR.

get_conditional_waterdepth(*df*)

Compute minimum water depth, conditional on being flooded, for df. Requires df to have column FLOOD_STATUS, WATERDEPTH_10_INDICATOR, WATERDEPTH_100_INDICATOR, WATERDEPTH_1000_INDICATOR, WATERDEPTH_10000_INDICATOR, and WATERDEPTH_MAX.

get_flood_status(*gdf*)

Get flood risk status for each column in gdf (takes value "floodable", "nearly floodable", "never flooded").

get_flood_variables(*gdf*)

Get flood risk variables for gdf. In particular, this function generates the following variables (where "scenario" takes values 10, 100, 1000, and 10000):

- FLOOD_{scenario} for scenario in (10, 100, 1000, 10000): Whether the address is within a Risicokaart flooded geometry under given scenario.
- WATERDEPTH_{scenario} for scenario in (10, 100, 1000, 10000): Maximum water depth of the flooded geometry the address is within under given scenario.
- WATERDEPTH_{scenario}_INDICATOR for scenario in (10, 100, 1000, 10000): Maximum water depth of the flooded geometry the address is within under given scenario.

get_july_floods_indicators(*gdf*)

Get flood exposure variables for gdf. In particular, this function generates the following variables:

- **FLOODED_ENW_UNPREDICTED_{suffix}: Areas that were flooded in July 2021 according**
  to ENW data, but that never flood according to Risicokaart flood maps.
- **FLOODED_RVO_UNPREDICTED_{suffix}: Areas from which flood damage claims were filed**
  after the July 2021 floods, but that never flood according to Risicokaart flood maps. Only computed if the RVO data are present.

The suffix "_STRICT" indicates that we consider unpredictable all the floods that happened outside of the geometries in the Risicokaart maps. The suffix "_LAX" indicates that we consider unpredictable all the floods that happened farther away than 500m from the geometries in the Risicokaart maps.

get_most_likely_scenario(*df*, *extended=False*)

Get most likely scenario for which each addres in df floods. Setting extended to True considers also houses within 500m from flooded areas.

get_waterdepth_indicator(*gdf*, *waterdepths*, *var_name*)

Convert water depth variable named var_name_waterdepth in gdf to numeric, for scenario in waterdepths.

Parameters

- **gdf** (*geopandas.GeoDataFrame*) – Dataframe.
- **waterdepths** (*list*) – List of some or all values in 10, 100, 1000, 10000.
- **var_name** (*str*) – Fixed part of variable name in original dataset.

Returns
geopandas.GeoDataFrame

load_BAG_chunk(*BAGpath*, *BAGdict*, *chunk*)

Read 1 million rows of BAG data .csv file, after skipping the number of rows indicated by the chunk argument. Read the first row as columns' names.

Parameters

- **BAGpath** (*str or Pathlib object*) – Path to BAG data.
- **BAGdict** (*dict*) – Dictionary of labels for columns of BAG dataframe.
- **chunk** (*int*) – Number of rows to skip.

Returns
Pandas.DataFrame

merge_flood_datasets_to_bag(*scenarios*, *RKdatadict*, *bagGDF*)

Merge 4 datasets assigning BAG addresse to each of the 4 Risicokaart scenarios to full BAG data.

Parameters

- **scenarios** (*list of pandas.Dataframes*) – List of dataframes, one for each Risicokaart scenario.
- **RKdatadict** (*dict*) – Dictionary whose keys refer to the Risicokaart scenarios.
- **bagGDF** (*geopandas.Dataframe*) – BAG dataset.

Returns
geopandas.Dataframe

remove_redundant_flood_rows(*gdf*, *key*, *finalGDF*)

For geodataframe gdf, corresponding to a key scenario, keep the duplicated row with the highest maximum depth.

2.5.2 Script sample.py

In the folder *task_sample_survey_recipients*. Auxiliary functions to generate sample of survey respondents.

format_data_for_qualtrics(*qualtricsDF*)

Format pandas.DataFrame that needs to be uploaded as Qualtrics contact list.

get_covariates_dataset(*covariatesDF*, *valuesDF*, *weights=False*)

Compute (weighted) average values of variables in covariatesDF from valuesDF. Missing values are automatically excluded.

Parameters

- **covariatesDF** (*pandas.DataFrame*) – Dataframe of covariates. Need to have columns named "CATEGORY", "DESCRIPTION", "VARLEVEL", "VARNAME", and "WEIGHTS".

- **valuesDF** (*pandas.DataFrame*) – Dataframes of covariates' values. Need to have a column named "VARNAME" containing the variables in covariatesDF.
- **weights** (*Bool*) – Weights to compute weighted mean. If True, will be pulled from valuesDF.WEIGHTS. Default is False

Returns

pandas.DataFrame

id_generator(*seed*, *length=8*, *restriction=False*)

Generate random string of given length, given seed.

2.5.3 Script clean_data.py

In the folder *task_clean_survey_data*. This script contains functions used in `task_clean_survey_data.py` to clean the original Qualtrics data in *bld/data/SURVEY*.

add_floodmaps_indicators(*df*, *waterdepth_column*, *floodrisk_column*)

Derive indicators of flood risk and maximum water depth from existing columns named waterdepth_column and floodrisk_column in Pandas.DataFrame df.

add_outcomes_dummies(*df*)

Add columns to indicate (1) whether respondent has at least one outcome, and (2) whether a given outcome is present, to Pandas.DataFrame df. The latter needs to have columns "worry_RE", "risk_RE", "damages_RE", "comptot_RE", "compshare_RE", "wtp_info", and "wtp_insurance".

add_revise_beliefs_variables(*df*)

Add variables providing information on expected direction of beliefs updating conditional on baseline information frictions for risk and damages.

add_time_spent_on_treatment_text(*df*)

Add the time spent on the treatment text ("{text}_time_page_submit") to the appropriate total_seconds_-prefixed column for each survey respondent.

I derived the time spent on each text from the toggle box, therefore by design the columns prefixed by total_seconds_ are empty whenever the text was not in the toggle box page (e.g. total_seconds_maps is empty for survey respondents in treatment 2).

assign_variable_quartile(*df*, *var*)

Compute and assign quartile to each observation of variable var in Pandas.DataFrame df.

assign_variable_tercile(*df*, *var*)

Compute and assign tercile to each observation of variable var in Pandas.DataFrame df.

clean_beliefs_data(*df*)

Clean dataframe df containing questions of Qualtrics survey related to prior and posterion beliefs. The labels for questions related to confidence in beliefs are in the dictionary confDict.

clean_covariates_data(*df*)

Clean answers to questions about households and survey respondent.

clean_info_data(*df*, *error*)

Clean dataframe df containing questions of Qualtrics survey related to information quality. The labels for questions related to confidence in the answers are in the dictionary confDict.

Parameters

- **df** (*pandas.DataFrame*) – Dataframe of survey responses.
- **error** (*float or int*) – Percentage points threshold for which a participants' estimate of the compensated claims is considered incorrect.

Returns
pandas.DataFrame.

clean_tech_data(*df*)

Clean dataframe df containing questions of Qualtrics survey related to technical fields (e.g.: time of survey completion, answers to consent forms...)

clean_text_evaluation(*df*)

Clean evaluation of treatment texts.

clean_treatment_data(*df*)

Clean dataframe df containing questions of Qualtrics survey related to experimental treatment(s).

clean_variables_block(*qdf*, *rvar_df*, *fvar_df*, *block*, *fun*)

Clean block of variables in qdf.

Parameters

- **qdf** (*Pandas.DataFrame*) – Dataset of raw Qualtrics variables.
- **rvar_df** (*Pandas.DataFrame*) – Dataset containing the names of the raw Qualtrics variables to clean.
- **fvar_df** (*Pandas.DataFrame*) – Dataset containing the names of the final clean variables.
- **block** (*Pandas.DataFrame*) – Block of variables to clean (BLOCK in rvar_df and in fvar_df).
- **fun** (*fun*) – Function to clean the specified block of variable.

Returns
Pandas.DataFrame

clean_wtp_data(*df*)

Clean data related to willingness-to-pay elicitations.

compute_belief_update(*df*, *prior_belief*, *posterior_belief*)

Compute difference between variables posterior_belief and prior_belief in Pandas.DataFrame df.

compute_prior_beliefs_zscore(*df*)

Compute z-score of prior beliefs in Pandas.DataFrame df.

The variables I use are:

- **risk_standardized: Standardized measure of beliefs about**
  10-year flood risk.
- **damages_wins975_1000_standardized: Standardized measure**
  of beliefs about damages, winsorized at the 97.5% percentile, in thoursand of euros.
- **comptot_standardized: Standardized measure of beliefs about**
  total compensation.

- **compshare_standardized: Standardized measure of beliefs**
  about compensation from the government.

The standardized beliefs (i.e. the z-scores) increase with expected net damages, i.e. the higher risk_standardized and damages_wins975_1000_standardized, and the lowest comptot_standardized and compshare_standardized, the higher the average z-score.

`compute_time_spent_on_info`(*series*)

Compute time spent on each toggle box containing information.

The Qualtrics survey is programmed in such a way that three toggle boxes are presented simultaneously to each survey respondent, and only one toggle box can remain open at any given time. Any open toggle box will be closed if the survey respondent clicks on it again or clicks on another toggle box. Moreover, each toggle box is associated to a variable storing timestamps of each time the toggle box was clicked. As a result, it is possible to compute the time spent on each toggle box from the timestamps.

For each survey respondents (whose timestamps for all toggle boxes is stored in series), this algorithm figures out the order in which the toggle boxes have been clicked, and uses it to derive the total number of seconds spent on each toggle box.

`create_info_friction_indicators`(*df*, *error*)

Create variables that take value 0 if survey respondents answered a question correctly and 1 otherwise. This serves to aggregated information frictions.

Parameters

- **df** (*pandas.DataFrame*) – Dataframe of survey responses.
- **error** (*float or int*) – Percentage points threshold for which a participants' estimate of the compensated claims is considered incorrect.

Returns
pandas.DataFrame.

`derive_wtp_for_info`(*df*)

Derive willingness to pay for information from incentivized survey task.

`format_timestamps`(*df*)

Extract datetime objects from relevant columns in Pandas.DataFrame df, i.e. those with prefix "timestamps". The latter columnes contains the (stringed) timestamps referring to each time each toggle box was clicked.

For example, if the first row of column "timestamps_maps" contains the value: 'Fri Dec 23 2022 14:18:24 GMT+0100 (CET);', this means that the first-row survey respondent clicked the toggle box containing information on floodmaps exactly one time, on Friday, December 23th 2022, at 14:18:24 Central European Time.

`make_adjustment_for_pilot_data`(*df*)

For some observations (less than 20, from the pilot), the timestamp of the submit button in the toggle boxes page is missing (my mistake, I misunderstood how the total time spent on the page is saved by Qualtrics. I thought Qualtrics saved timestamps). As a result, I need to compute the time spent on the last toggle box differently.

`merge_same_question_different_treatment`(*df*, *colname*)

Take columns named "colname", "colname.1", "colname.2", and "colname.3" (for whatever string colname is equal to) from Pandas.DataFrame df, and merge them to one unique column named

"colname". Each row must have a well-defined value for only one of the "colname" columns, and a numpy nan elsewhere.

```
>>> colDF = pd.DataFrame([
        [1, np.nan, np.nan, np.nan],
        [np.nan, 1, np.nan, np.nan],
        [np.nan, np.nan, 1, np.nan],
        [np.nan, np.nan, np.nan, 1]
        ],
    columns=["c", "c.1", "c.2", "c.3"]
    )
>>> merge_same_question_different_treatment(colDF, "c")
    c
0  1.0
1  1.0
2  1.0
3  1.0
```

return_boxes_order(*series*)

Sort order of tuple of string and datetime object.

return_order_of_clicked_boxes(*df*)

Add columns indicating order of clicked boxes to df.

return_ordered_tuple(*series*)

Return tuple of string and first element of each value in Pandas.Series series.

standardize_variable(*df*, *var*)

Standardized variable var in Pandas.DataFrame df.

winsorize(*df*, *columns*, *quantile*)

Winsorize list of columns in Pandas.DataFrame df according to upper quantile quantile.

Parameters

- **df** (*Pandas.DataFrame*) – Dataframe of interest.
- **columns** (*list*) – List of column(s) in df.
- **quantile** (*number*) – Upper quantile, must be between 0 and 1.

Returns

Pandas.DataFrame with winsorized columns.

CHAPTER

THREE

DATASET USED IN THE ANALYSIS

The next pages show the codebook of the experimental dataset, obtained after cleaning the original Qualtrics data. A more extensive version of the same codebook is provided with the survey data in *bld/experiment_floodplain_replication_data/SURVEY*.

Table 1: Codebook

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
uniqueadd_id	ID unique to each address (i.e., to each survey respondent)			Alphanumeric code
flood_max_1_in_100_years	Whether address is in the 1 in 100 years flood category			0, 1
flood_max_1_in_1000_years	Whether address is in the 1 in 1000 years flood category			0, 1
flood_max_1_in_10000_years	Whether address is in the 1 in 10000 years flood category			0, 1
waterdepth_max_less_than_05m	Whether address is in the 0.5m maximum water depth category			0, 1
waterdepth_max_between_05_and_1m	Whether address is in the 0.5m to 1m maximum water depth category			0, 1
waterdepth_max_between_1_and_15m	Whether address is in the 1m to 1.5m maximum water depth category			0, 1
waterdepth_max_between_15_and_2m	Whether address is in the 1.5m to 2m maximum water depth category			0, 1
waterdepth_max_between_2_and_5m	Whether address is in the 2m to 5m maximum water depth category			0, 1
waterdepth_max_more_than_5m	Whether address is in the more than 5m maximum water depth category			0, 1
waterdepth_max_over_2m	Whether address is in the more than 2m maximum water depth category (coarser category for randomization integrity checks)	waterdepth_max_less_than_05m, waterdepth_max_between_05_and waterdepth_max_between_1_and_ waterdepth_max_between_15_and waterdepth_max_between_2_and_ waterdepth_max_more_than_5m		0, 1
FLOODED	Whether the address is in the July 2021 floodplain			0, 1
dist_floodareas	Distance from (closest) flooded area			Non-negative number
GEMNAME_2022	Gemeente in 2022			Gemeente name
consent	Consent to participate into the survey, in words			I do NOT consent to participate in this survey, I consent to participate in this survey
consent_numeric	Consent to participate into the survey, numeric	consent		0, 1
consent_followup	Consent to be contacted for follow-up surveys, in words			I do NOT consent to be contacted via email for a follow-up survey, I do consent to be contacted via email for a follow-up survey
consent_followup_numeric	Consent to be contacted for follow-up surveys, numeric	consent_followup		0, 1
StartDate	When the respondents first clicked the survey link			Time stamp
EndDate	When the respondent submitted their survey			Time stamp

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
Progress	The progress a respondent made in the survey before finishing, in percentage of the total survey questiond			Integer between 0 and 100
duration_seconds	The number of seconds it took the respondent to complete the survey.			Non-negative number
Finished	Whether the response was submitted or closed			0, 1 OR FALSE, TRUE
RecordedDate	When a survey was recorded in Qualtrics			Time stamp
UserLanguage	Respondent's language code			EN, NL
payment	How the survey respondent would be like to receive payment, if selected		Would you prefer to receive your voucher via email (as a numeric code) or via physical mail?	Email, Physical mail
household_members	Total number of household members		Including yourself, how many people – including children – live here regularly as members of this household? Please enter a number.	1, 2, 3+
hh_member_residing	Whether there is a household member who has been residing at the address longer than the respondent		Is there another household member who has been residing at this address longer than you?	yes, no
hh_member_how_long	Time spent at the address by the household member who has been residing there longer than the respondent (if applicable)		How long has the household member been residing at this address?	Less than 1 year, Between 1 and 5 years, Between 5 and 10 years, Between 10 and 20 years, More than 20 years
homeownership	Whether the house is owned		Do you own the house at this address?	yes, no, prefer not to say
preparedness	Measures against floods implemented by the household		The measures below may reduce your flood risk or reduce your damages in case of a flood. Please select the measures among these implemented in your house or by your household, if any. You can select multiple options.	EITHER None of the measures above OR Don't know, OR one or more of the following: My household has insurance coverage against floods from extreme rain, My household has insurance coverage against riverine floods, My house is elevated above the street level, My house walls and/or floors are built with water-resistant materials, My house has anti-backflow valves installed on pipes, My house has a pump and/or other systems to drain flood water installed, In my house there are sandbags or other water barriers (e.g. water-proof basement windows), My valuable assets are on higher floors or elevated areas
preparedness_how_many_measures	Number of measures against floods implemented by the household	preparedness		Integer between 0 and 8
preparedness_high	Number of measures against floods implemented by the household is higher than the average number (average and median coincinde and are equal to 1)	prepared-ness_how_many_masure		0, 1

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
house_elevation	By how much the house is elevated		By how much is your house elevated?	Up to 50 cm above the street level, Between 50 cm and 1 m above the street level, More than 1 m above the street level
julyflood_damages	Whether the household experienced financial damages during the 2021 floods		Did you face damages during the floods that hit Limburg in July 2021?	yes, no
julyflood_comp	Whether the household received compensation after the 2021 floods		Have the damages been compensated, either by the government or by your insurance company?	Yes, Partially, No and I am not eligible for compensation, Not yet but I am eligible for compensation
household_income	Household income		What was your household's monthly net income in 2021?	Less than 1200 EUR, Between 1200 and 1700 EUR, Between 1700 and 2500 EUR, Between 2500 and 4000 EUR, More than 4000 EUR, Prefer not to say, Don't know
household_members_1	Single-member household	household_members		0, 1
household_members_2	Two-members household	household_members		0, 1
household_members_3+	Three-members or more household	household_members		0, 1
household_members_missing_value	Missing value in household members	household_members		0, 1
homeownership_missing_value	Missing value in homeownership	homeownership		0, 1
homeownership_prefer_not_to_say	Prefer not to say whether homeowner	homeownership		0, 1
homeownership_no	House is not owned	homeownership		0, 1
homeownership_yes	House is owned	homeownership		0, 1
preparedness_dont_know	Does not know whether measures against flood risk are implemented	preparedness		0, 1
preparedness_no_measures	Implemented no measures against flood risk	preparedness		0, 1
preparedness_missing_value	Missing value in preparedness	preparedness		0, 1
preparedness_has_insurance_rain	Household has insurance coverage against floods from extreme rain	preparedness		0, 1
preparedness_has_insurance_rivers	Household has insurance coverage against riverine floods	preparedness		0, 1
preparedness_house_elevated	House is elevated above the street level	preparedness		0, 1
preparedness_house_water_resistant	House walls and/or floors are built with water-resistant materials	preparedness		0, 1
preparedness_house_has_valves	House has anti-backflow valves installed on pipes	preparedness		0, 1
preparedness_house_has_pump	House has a pump and/or other systems to drain flood water installed	preparedness		0, 1
preparedness_house_has_sandbags	House has sandbags or other water barriers	preparedness		0, 1
preparedness_assets_are_high	Valuable assets are on higher floors or elevated areas	preparedness		0, 1
julyflood_damages_missing_value	Missing value julyflood_damages	julyflood_damages		0, 1
julyflood_damages_no	Did not experience financial damages during the 2021 floods	julyflood_damages		0, 1
julyflood_damages_yes	Did experience financial damages during the 2021 floods	julyflood_damages		0, 1
household_income_less_than_1200_eur	Household income is less than 1200 EUR per month	household_income		0, 1

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
household_income_between_1200_and_1700	Household income is between 1200 and 1700 EUR per month	household_income		0, 1
household_income_between_1700_and_2500	Household income is between 1700 and 2500 EUR per month	household_income		0, 1
household_income_between_2500_and_4000	Household income is between 2500 and 4000 EUR per month	household_income		0, 1
household_income_more_than_4000_eur	Household income is over 4000 EUR per month	household_income		0, 1
household_income_dont_know	Does not know household income	household_income		0, 1
household_income_missing_value	Missing value in household_income	household_income		0, 1
household_income_prefer_not_to_say	Prefer not to say household income	household_income		0, 1
gender	Gender of survey respondent		What is your gender?	Female, Male, Other, Prefer not to say
age	Age of survey respondent		What is your age?	Between 18 and 24 years old, Between 25 and 44 years old, between 45 and 64 years old, Older than 65 years old
education	Education of survey respondent		What is your highest education level?	WO, HBO, MBO-2 or MBO-3 or MBO-4, MBO-1, None of the above/other
how_long_residing	Time spent at the address by survey respondent		How long have you been residing at this address?	Less than 1 year, Between 1 and 5 years, Between 5 and 10 years, Between 10 and 20 years, More than 20 years
how_long_planning	Time the survey respondent plans to spend at the address		How long do you plan to live at this address?	Less than 1 year, Between 1 and 5 years, Between 5 and 10 years, Between 10 and 20 years, More than 20 years
municipality	Whether the first residence of the survey respondent was in the same municipality of the address		As far as you remember, was the first house you ever lived in in this municipality?	yes, no, don't know
own_background	Whether survey respondent is Dutch		Were you born in The Netherlands	yes, no, prefer not to say
agree_transparent	How much survey respondent agrees with statement on flood risk management in The Netherlands		How much do you agree with this statement? "In The Netherlands, the objectives of water management programs (for example, Room for the River) are communicated in a transparent way."	Strongly disagree, Somewhat disagree, Neither agree nor disagree, Somewhat agree, Strongly agree
source_info	Consulted sources about flood risk		Do you ever consult sources to learn about flood risk in The Netherlands? If yes, which of the options below? You can select multiple options.	EITHER I don't consult any source OR one or more of the following: Government websites, Other websites, My insurance company, Neighbours or friends, Other
source_info_how_many	Number of sources consulted about flood risk	source_info		Integer between 0 and 5

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
gender_female	Survey respondent is female	gender		0, 1
gender_male	Survey respondent is male	gender		0, 1
gender_missing_value	Missing value in gender	gender		0, 1
gender_other	Survey respondent's gender is other	gender		0, 1
gender_prefer_not_to_say	Survey respondent prefers not to say gender	gender		0, 1
age_between_18_and_24_years_old	Survey respondent is between 18 and 24 years old	age		0, 1
age_between_25_and_44_years_old	Survey respondent is between 25 and 44 years old	age		0, 1
age_between_45_and_64_years_old	Survey respondent is between 45 and 64 years old	age		0, 1
age_older_than_65_years_old	Survey respondent is 65 years old or older	age		0, 1
age_missing_value	Missing value in age	age		0, 1
age_prefer_not_to_say	Survey respondent prefers not to say age	age		0, 1
education_hbo	Education is HBO	education		0, 1
education_mbo1	Education is MBO2	education		0, 1
education_mbo2_or_mbo3_or_mbo4	Education is MBO2 or MBO3 or MBO4	education		0, 1
education_wo	Education is WO	education		0, 1
education_none_of_the_above_other	Education is other	education		0, 1
education_missing_value	Missing value in education	education		0, 1
how_long_residing_less_than_1_year	Residing at the address for less than 1 year	how_long_residing		0, 1
how_long_residing_between_1_and_5_	Residing at the address for between 1 and 5 years	how_long_residing		0, 1
how_long_residing_between_10_and_2	Residing at the address for between 10 and 20 years	how_long_residing		0, 1
how_long_residing_between_5_and_10	Residing at the address for between 5 and 10 years	how_long_residing		0, 1
how_long_residing_more_than_20_yea	Residing at the address for more than 20 years	how_long_residing		0, 1
how_long_residing_missing_value	Missing value in how long residing at the address	how_long_residing		0, 1
municipality_no	First address ever not in the same municipality	municipality		0, 1
municipality_yes	First address ever in the same municipality	municipality		0, 1
municipality_dont_know	Doesn't know if first address ever in the same municipality	municipality		0, 1
municipality_missing_value	Missing value in municipality	municipality		0, 1
own_background_missing_value	Missing value in whether born in The Netherlands	own_background		0, 1
own_background_no	Not born in The Netherlands	own_background		0, 1
own_background_yes	Born in The Netherlands	own_background		0, 1
own_background_prefer_not_to_say	Prefer not to say if born in The Netherlands	own_background		0, 1
source_info_none	No source about flood risk consulted	source_info		0, 1
source_info_missing_value	Missing value in consulted sources about flood risk	source_info		0, 1
source_info_gov	Consult government websites about flood risk	source_info		0, 1
source_info_web	Consult other websites about flood risk	source_info		0, 1
source_info_ins	Consult insurance company about flood risk	source_info		0, 1
source_info_friends	Consult friends or neighbours about flood risk	source_info		0, 1
source_info_other	Consult other sources about flood risk	source_info		0, 1
time_residing_m10	Residing at the address for more than 10 years	how_long_residing		0, 1
time_planning_m10	Planning to reside at the address for more than 10 years	how_long_planning		0, 1
age_younger_than_45	Younger than 45 years old	age		0, 1
household_income_min2500	Household income lower than 2500 EUR per month	household_income		0, 1
household_income_m2500	Household income higher than 2500 EUR per month	household_income		0, 1

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
edu_hbo_or_higher	Education HBO or WO	education		0, 1
worry	Worry that house will be flooded, in words (before experiment)		How worried are you about your house being flooded?	Not worried at all, Not so worried, Somewhat worried, Very worried, Extremely worried
worry_numeric	Worry that house will be flooded, numeric (before experiment)	worry		Integer between 1 and 5
risk	Subjective probability that house will be flooded at least one in the next 10 years (before experiment)		What do you think is the probability that your house will be flooded at least once in the next 10 years? Please type a number between 0 (no flood will happen) and 100 (at least one flood will certainly happen). You can use a dot to indicate decimals.	Number between 0 and 100
damages	Subjective damages expected in case of a flood, in EUR (before experiment)		Imagine your house were flooded. How much damage would you expect (in EUR)?	Integer greater than 0
damages_wins975	Subjective damages expected in case of a flood, in EUR, winsorized at the 97.5 percentile (before experiment)	damages		Integer greater than 0
damages_wins99	Subjective damages expected in case of a flood, in EUR, winsorized at the 99 percentile (before experiment)	damages		Integer greater than 0
damages_1000	Subjective damages expected in case of a flood, in kEUR (before experiment)	damages		Integer greater than 0
damages_wins975_1000	Subjective damages expected in case of a flood, in kEUR, winsorized at the 97.5 percentile (before experiment)	damages		Integer greater than 0
damages_wins99_1000	Subjective damages expected in case of a flood, in kEUR, winsorized at the 99 percentile (before experiment)	damages		Integer greater than 0
comptot	Subjective share of compensated damages in case of a flood (before experiment)		What do you think is the percentage of damages that would be compensated, either by your insurer or by the government? 0 means “I expect I would have to pay for all the damages myself”. 100 means “I expect I would be fully compensated”.	Integer between 0 and 100
compshare	Subjective share of damages compensated by the government in case of a flood, as share of total compensation (before experiment)		Of the damages you think would be compensated, which percentage do you think would be paid by the government? 0 means “I expect that all compensation would be paid by my insurance company”. 100 means “I expect that all compensation would be paid by the government”.	Integer between 0 and 100
risk_conf	Confidence in subjective probability that house will be flood at least one in the next 10 years (before experiment)		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
damages_conf	Confidence in subjective damages expected in case of a flood (before experiment)		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10
comptot_conf	Confidence in subjective share of compensated damages in case of a flood (before experiment)		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10
compshare_conf	Confidence in subjective share of damages compensated by the government in case of a flood (before experiment)		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10
average_beliefs_confidence	Average confidence in subjective beliefs	risk, damages, comptot, compshare		Integer between 1 and 10
worry_RE	Worry that house will be flooded, in words (after experiment)		How worried are you about your house being flooded?	Not worried at all, Not so worried, Somewhat worried, Very worried, Extremely worried
worry_RE_numeric	Worry that house will be flooded, numeric (after experiment)	worry_RE		Integer between 1 and 5
risk_RE	Subjective probability that house will be flooded at least one in the next 10 years (after experiment)		What do you think is the probability that your house will be flooded at least once in the next 10 years? Please type a number between 0 (no flood will happen) and 100 (at least one flood will certainly happen). You can use a dot to indicate decimals.	Number between 0 and 100
damages_RE	Subjective damages expected in case of a flood, in EUR (after experiment)		Imagine your house were flooded. How much damage would you expect (in EUR)?	Integer greater than 0
damages_RE_wins975	Subjective damages expected in case of a flood, in EUR, winsorized at the 97.5 percentile (after experiment)	damages_RE		Integer greater than 0
damages_RE_wins99	Subjective damages expected in case of a flood, in EUR, winsorized at the 99 percentile (after experiment)	damages_RE		Integer greater than 0
damages_RE_1000	Subjective damages expected in case of a flood, in kEUR (after experiment)	damages_RE		Integer greater than 0
damages_RE_wins975_1000	Subjective damages expected in case of a flood, in kEUR, winsorized at the 97.5 percentile (after experiment)	damages_RE		Integer greater than 0
damages_RE_wins99_1000	Subjective damages expected in case of a flood, in kEUR, winsorized at the 99 percentile (after experiment)	damages_RE		Integer greater than 0
comptot_RE	Subjective share of compensated damages in case of a flood (after experiment)		What do you think is the percentage of damages that would be compensated, either by your insurer or by the government? 0 means “I expect I would have to pay for all the damages myself”. 100 means “I expect I would be fully compensated”.	Integer between 0 and 100

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
compshare_RE	Subjective share of damages compensated by the government in case of a flood, as share of total compensation (after experiment)		Of the damages you think would be compensated, which percentage do you think would be paid by the government? 0 means "I expect that all compensation would be paid by my insurance company". 100 means "I expect that all compensation would be paid by the government".	Integer between 0 and 100
risk_conf_RE	Confidence in subjective probability that house will be flood at least one in the next 10 years (after experiment)		How confident are you in your answer, from 1 ("Very unsure") to 10 ("Very sure")?	Integer between 1 and 10
damages_conf_RE	Confidence in subjective damages expected in case of a flood (after experiment)		How confident are you in your answer, from 1 ("Very unsure") to 10 ("Very sure")?	Integer between 1 and 10
comptot_conf_RE	Confidence in subjective share of compensated damages in case of a flood (after experiment)		How confident are you in your answer, from 1 ("Very unsure") to 10 ("Very sure")?	Integer between 1 and 10
compshare_conf_RE	Confidence in subjective share of damages compensated by the government in case of a flood (after experiment)		How confident are you in your answer, from 1 ("Very unsure") to 10 ("Very sure")?	Integer between 1 and 10
risk_update	Update in subjective probability that house will be flooded at least one in the next 10 years	risk, risk_RE		Integer between -100 and 100
damages_update	Update in subjective damages expected in case of a flood, in EUR	damages, damages_RE		Any number
damages_wins99_update	Update in subjective damages expected in case of a flood, in EUR, winsorized at the 99 percentile	damages_wins99, damages_RE_wins99		Any number
damages_wins975_update	Update in subjective damages expected in case of a flood, in EUR, winsorized at the 97.5 percentile	damages_wins975, damages_RE_wins975		Any number
damages_1000_update	Update in subjective damages expected in case of a flood, in kEUR	damages_1000, damages_RE_1000		Any number
damages_wins99_1000_update	Update in subjective damages expected in case of a flood, in kEUR, winsorized at the 99 percentile	damages_wins99_1000, damages_RE_wins99_1000		Any number
damages_wins975_1000_update	Update in subjective damages expected in case of a flood, in kEUR, winsorized at the 97.5 percentile	damages_wins975_1000, damages_RE_wins975_1000		Any number
comptot_update	Update in subjective share of compensated damages in case of a flood	comptot, comptot_RE		Integer between -100 and 100
compshare_update	Update in subjective share of damages compensated by the government in case of a flood, as share of total compensation	compshare, compshare_RE		Integer between -100 and 100
worry_numeric_update	Update in worry	worry_numeric, worry_RE_numeric		Integer between 0 and 4

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
risk_update_any	Whether there is an update in belief about risk	risk_update		0, 1
damages_update_any	Whether there is an update in belief about damages	damages_update		0, 1
comptot_update_any	Whether there is an update in belief about total compensation	comptot_update		0, 1
compshare_update_any	Whether there is an update in belief about government compensation	compshare_update		0, 1
worry_numeric_update_any	Whether there is an update in worry measure	worry_numeric_update		0, 1
risk_revise	Direction in which the belief about risk was revised	risk_update		upward, downward, none
damages_revise	Direction in which the belief about damages was revised	damages_update		upward, downward, none
comptot_revise	Direction in which the belief about total compensation was revised	comptot_update		upward, downward, none
compshare_revise	Direction in which the belief about government compensation was revised	compshare_update		upward, downward, none
worry_numeric_revise	Direction in which the worry measure was revised	worry_numeric_update		upward, downward, none
risk_quartile	Quartile of subjective probability that house will be flooded at least one in the next 10 years (before experiment)	risk		q1, q2, q3, q4
damages_quartile	Quartile of subjective damages expected in case of a flood, in EUR (before experiment)	damages		q1, q2, q3, q4
damages_wins99_quartile	Quartile of subjective damages expected in case of a flood, in EUR, winsorized at the 99 percentile (before experiment)	damages_wins99		q1, q2, q3, q4
damages_wins975_quartile	Quartile of subjective damages expected in case of a flood, in EUR, winsorized at the 97.5 percentile (before experiment)	damages_wins975		q1, q2, q3, q4
damages_1000_quartile	Quartile of subjective damages expected in case of a flood, in kEUR (before experiment)	damages_1000		q1, q2, q3, q4
damages_wins99_1000_quartile	Quartile of subjective damages expected in case of a flood, in kEUR, winsorized at the 99 percentile (before experiment)	damages_wins99		q1, q2, q3, q4
damages_wins975_1000_quartile	Quartile of subjective damages expected in case of a flood, in kEUR, winsorized at the 97.5 percentile (before experiment)	damages_wins975		q1, q2, q3, q4
comptot_quartile	Quartile of subjective share of compensated damages in case of a flood (before experiment)	comptot		q1, q2, q3, q4
compshare_quartile	Quartile of subjective share of damages compensated by the government in case of a flood, as share of total compensation (before experiment)	compshar		q1, q2, q3, q4
risk_tercile	Tercile of subjective probability that house will be flooded at least one in the next 10 years (before experiment)	risk		t1, t2, t3
damages_tercile	Tercile of subjective damages expected in case of a flood, in EUR (before experiment)	damages		t1, t2, t3
damages_wins99_tercile	Tercile of subjective damages expected in case of a flood, in EUR, winsorized at the 99 percentile (before experiment)	damages_wins99		t1, t2, t3
damages_wins975_tercile	Tercile of subjective damages expected in case of a flood, in EUR, winsorized at the 97.5 percentile (before experiment)	damages_wins975		t1, t2, t3

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
damages_1000_tercile	Tercile of subjective damages expected in case of a flood, in kEUR (before experiment)	damages_1000		t1, t2, t3
damages_wins99_1000_tercile	Tercile of subjective damages expected in case of a flood, in kEUR, winsorized at the 99 percentile (before experiment)	damages_wins99		t1, t2, t3
damages_wins975_1000_tercile	Tercile of subjective damages expected in case of a flood, in kEUR, winsorized at the 97.5 percentile (before experiment)	damages_wins975		t1, t2, t3
comptot_tercile	Tercile of subjective share of compensated damages in case of a flood (before experiment)	comptot		t1, t2, t3
compshare_tercile	Tercile of subjective share of damages compensated by the government in case of a flood, as share of total compensation (before experiment)	compshar		t1, t2, t3
risk_standardized	Standardized subjective probability that house will be flooded at least one in the next 10 years (before experiment)	risk		Any number
damages_standardized	Standardized subjective damages expected in case of a flood, in EUR (before experiment)	damages		Any number
damages_wins99_standardized	Standardized subjective damages expected in case of a flood, in EUR, winsorized at the 99 percentile (before experiment)	damages_wins99		Any number
damages_wins975_standardized	Standardized subjective damages expected in case of a flood, in EUR, winsorized at the 97.5 percentile (before experiment)	damages_wins975		Any number
damages_1000_standardized	Standardized subjective damages expected in case of a flood, in kEUR (before experiment)	damages_1000		Any number
damages_wins99_1000_standardized	Standardized subjective damages expected in case of a flood, in kEUR, winsorized at the 99 percentile (before experiment)	damages_wins99_1000		Any number
damages_wins975_1000_standardized	Standardized subjective damages expected in case of a flood, in kEUR, winsorized at the 97.5 percentile (before experiment)	damages_wins975_1000		Any number
comptot_standardized	Standardized subjective share of compensated damages in case of a flood (before experiment)	comptot		Any number
compshare_standardized	Standardized subjective share of damages compensated by the government in case of a flood, as share of total compensation (before experiment)	compshare		Any number
prior_beliefs_zscore	Z-score of prior beliefs, where standardized beliefs (i.e. the z-scores) increase with expected net damages, i.e. the higher the beliefs risk and damages and the lowest the beliefs about compensation, the higher the average z-score	risk_standardized, damages_1000_standardized, comptot_standardized, compshare_standardized		Any number
prior_beliefs_zscore_quartile	Quartile of z-score of prior beliefs	prior_beliefs_zscore		q1, q2, q3, q4
risk_should_revise	Direction in which the belief about risk should have been revised, according to baseline information frictions	friction_floodmaps		upward, downward, none
risk_revise_expected	Whether the belief about risk was revised correctly, according to baseline information frictions	risk_update, risk_should_revise		0, 1

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
damages_should_revise	Direction in which the belief about damages should have been revised, according to baseline information frictions	friction_waterdepth		upward, downward, none
damages_revise_expected	Whether the belief about damages was revised correctly, according to baseline information frictions	damages_update, damages_should_revise		0, 1
compshare_should_revise	Whether the belief about government compensation should have been revised downward, according to baseline information frictions	compshare_update, friction_WTS_indicator, friction_WTScomp_indicator		
compshare_revise_expected	Whether the belief about government compensation was revised correctly, according to baseline information frictions	compshare_update, compshare_should_revise		
correct_floodmaps	Flood risk category according to Risicokaart flood maps, in words			Medium probability (1 flood every 100 years), Small probability (1 flood every 1000 years), Extremely small probability (1 flood every 10,000 years)
correct_floodmaps_numeric	Flood risk category according to Risicokaart flood maps, numeric	correct_floodmaps		Integer between 1 and 3
correct_waterdepth	Maximum water depth category according to Risicokaart flood maps, in words			less than 0.5m, between 0.5 and 1m, between 1 and 1.5m, between 1.5 and 2m, between 2 and 5m, more than 5m
correct_waterdepth_numeric	Maximum water depth category according to Risicokaart flood maps, numeric	correct_waterdepth		Integer between 1 and 6
restoration	Subjective coverage of Grensmaas restoration project		The "Grensmaas" project is the largest river project in progress in the Netherlands. How many kilometers of the Maas river does this project cover?	About 10 km, About 20 km, About 30 km, About 40 km, About 50 km
stated_floodmaps	Subective flood category according to maps, in words		According to official flood maps, what is the maximum flood risk that applies to your address?	Large probability (1 flood every 10 years), Medium probability (1 flood every 100 years), Small probability (1 flood every 1000 years), Extremely small probability (1 flood every 10,000 years), No flood risk
stated_floodmaps_numeric	Subective flood category according to maps, numeric	stated_floodmaps		Integer between 0 and 4
stated_waterdepth	Subjective water depth category according to flood maps, in words		According to official flood maps, what is the maximum water depth your house would be exposed to in case of a flood?	0 cm, Up to 50 cm, Between 50 cm and 1 m, Between 1 and 1.5 m, Between 1.5 and 2 m, Between 2 and 5 m, More than 5 m
stated_waterdepth_numeric	Subjective water depth category according to flood maps, numeric	stated_waterdepth		Integer between 0 and 6
WTS	Subjective understanding of whether WTS is automatically triggered, in words		WTS After the floods in July 2021, the government invoked the Wet Tegemoetkoming Schade bij rampen en zware ongevallen (WTS) act. Is the WTS automatically triggered after any flood event?	yes, no

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
WTS_numeric	Subjective understanding of whether WTS is automatically triggered, numeric	WTS		0, 1
WTScomp	Subjective understanding of whether WTS ensures damage compensation, in words		When the WTS is invoked, does it mean that all flood-related damages will be compensated by the government?	yes, no
WTScomp_numeric	Subjective understanding of whether WTS ensures damage compensation, numeric	WTScomp		0, 1
claims	Subjective claims compensated by the government after the 2021 floods		Which percentage of damage claims filed because of the July 2021 floods do you think has been paid out by the government by now, from 0% (none of the claims) to 100% (all of the claims)?	Integer between 0 and 100
ins_rain	Subjective insurance availability for floods from extreme rain, in words		Are flood damages due to excessive rain insurable in The Netherlands?	Never, or almost never, By some insurers, Always, or almost always
ins_rain_numeric	Subjective insurance availability for floods from extreme rain, numeric	ins_rain		Integer between 0 and 2
ins_primary	Subjective insurance availability for floods from failure of primary defenses, in words		Are flood damages due to the breach of primary defences insurable in The Netherlands? Primary defences protect The Netherlands against flooding from the North Sea and major rivers (like the Rhine or the Maas).	Never, or almost never, By some insurers, Always, or almost always
ins_primary_numeric	Subjective insurance availability for floods from failure of primary defenses, numeric	ins_primary		Integer between 0 and 2
ins_secondary	Subjective insurance availability for floods from failure of secondary defenses, in words		Are flood damages due to the breach of secondary defences insurable in The Netherlands? Secondary defences protect The Netherlands against flooding from smaller rivers.	Never, or almost never, By some insurers, Always, or almost always
ins_secondary_numeric	Subjective insurance availability for floods from failure of secondary defenses, numeric	ins_secondary		Integer between 0 and 2
restoration_conf	Confidence in subjective coverage of Grensmaas restoration project		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10
floodmaps_conf	Confidence in subective flood category according to maps		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10
waterdepth_conf	Confidence in subjective water depth category according to flood maps		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
WTS_conf	Confidence in subjective understanding of whether WTS is automatically triggered		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10
WTScomp_conf	Confidence in subjective understanding of whether WTS ensures damage compensation		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10
claims_conf	Confidence in subjective claims compensated by the government after the 2021 floods		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10
ins_rain_conf	Confidence in subjective insurance availability for floods from extreme rain		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10
ins_primary_conf	Confidence in subjective insurance availability for floods from failure of primary defenses		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10
ins_secondary_conf	Confidence in subjective insurance availability for floods from failure of secondary defenses		How confident are you in your answer, from 1 (“Very unsure”) to 10 (“Very sure”)?	Integer between 1 and 10
average_info_confidence	Average confidence across answers to flood-related information-based questions	floodmaps_conf, waterdepth_conf, WTS_conf, WTScomp_conf, claims_conf, ins_rain_conf, ins_primary_conf, ins_secondary_conf		Integer between 1 and 10
friction_floodmaps	Number of categories by which flood risk category is overestimated or underestimated	stated_floodmaps, correct_floodmaps		Integer between -3 and 3
friction_floodmaps_indicator	Whether flood risk category is overestimated/underestimated	stated_floodmaps, correct_floodmaps		0, 1
friction_waterdepth	Number of categories by which water depth category is overestimated or underestimated	stated_waterdepth, correct_waterdepth		Integer between -6 and 6
friction_waterdepth_indicator	Whether water depth category is overestimated/underestimated	stated_waterdepth, correct_waterdepth		0, 1
friction_WTS_indicator	Whether rules on triggering of WTS are misunderstood	WTS		0, 1
friction_WTScomp_indicator	Whether rules on WTS compensation are misunderstood	WTScomp		0, 1
friction_claims_difference	Difference between stated claims paid out by the government after the 2021 floods and effective claims	claims		Integer between -53 and 47
friction_claims	Whether claims paid out by the government after the 2021 floods are overestimated, underestimated, or reported correctly (correctly means within 10 pp. from the true value, which is 53)	claims		overestimate, underestimate, about right

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
friction_claims_indicator	Whether claims paid out by the government after the 2021 floods are overestimated/underestimated or reported correctly (correctly means within 10 pp. from the true value, which is 53)	claims		0, 1
friction_ins_rain	Number of categories by which insurance availability in case of floods from excessive rain is overestimated/underestimated	ins_rain		Integer between 0 and 2
friction_ins_rain_indicator	Whether insurance availability in case of floods from excessive rain is overestimated/underestimated	ins_rain		0, 1
friction_ins_primary	Number of categories by which insurance availability in case of floods from failure of primary defences is overestimated/underestimated	ins_primary		Integer between 0 and 2
friction_ins_primary_indicator	Whether insurance availability in case of floods from failure of primary defences is overestimated/underestimated	ins_primary		0, 1
friction_ins_secondary	Number of categories by which insurance availability in case of floods from failure of secondary defences is overestimated/underestimated	ins_secondary		Integer between 0 and 2
friction_ins_secondary_indicator	Whether insurance availability in case of floods from failure of secondary defences is overestimated/underestimated	ins_secondary		0, 1
friction_topic_wts	Whether at least one answer to the information-based questions on compensation is incorrect	WTS, WTScomp, claims		0, 1
friction_topic_maps	Whether at least one answer to the information-based questions on flood maps categories is incorrect	floodmaps, waterdepth		0, 1
friction_topic_insurance	Whether at least one answer to the information-based questions on insurance is incorrect	ins_rain, ins_primary, ins_secondary		0, 1
total_frictions	Total number of wrong answers to information-based questions	friction_floodmaps_indicator, friction_waterdepth_indicator, friction_WTS_indicator, friction_WTScomp_indicator, friction_claims_indicator, friction_ins_rain_indicator, friction_ins_primary_indicator, friction_ins_secondary_indicator		Integer between 0 and 8
confidently_wrong_floodmaps	Whether the answer to the flood risk category question is incorrect and its associated confidence is 8 or higher	friction_floodmpa_indicator, floodmaps_conf		0, 1
confidently_wrong_waterdepth	Whether the answer to the maximum water depth category question is incorrect and its associated confidence is 8 or higher	friction_waterdepth_indicator, waterdepth_conf		0, 1

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
confidently_wrong_WTS	Whether the answer to the triggering of the WTS is incorrect and its associated confidence is 8 or higher	friction_WTS_indicator, WTS_conf		0, 1
confidently_wrong_WTScomp	Whether the answer to the WTS compensation question is incorrect and its associated confidence is 8 or higher	friction_WTScomp_indicator, WTScomp_conf		0, 1
confidently_wrong_claims	Whether the answer to the claims question is incorrect and its associated confidence is 8 or higher	friction_claims_indicator, claims_conf		0, 1
confidently_wrong_ins_primary	Whether the answer to the question on flood insurance from primary defences is incorrect and its associated confidence is 8 or higher	friction_ins_primary_indicator, ins_primary_conf		0, 1
confidently_wrong_ins_secondary	Whether the answer to the question on flood insurance from secondary defences is incorrect and its associated confidence is 8 or higher	friction_ins_secondary_indicator ins_secondary_conf		0, 1
confidently_wrong_ins_rain	Whether the answer to the question on flood insurance from extreme rain is incorrect and its associated confidence is 8 or higher	friction_ins_rain_indicator, ins_rain_conf		0, 1
confidently_wrong_answers	Number of questions to which the participant gave confidently wrong answers	confidently_wrong_floodmaps, confidently_wrong_waterdepth, confidently_wrong_WTS, confidently_wrong_WTScomp, confidently_wrong_claims, confidently_wrong_ins_primary, confidently_wrong_ins_secondary confidently_wrong_ins_rain		Integer between 0 and 7 (only flood-related questions are considered)
confidently_wrong_indicator	Whether the participant has been confidently wrong at least in one answer	confidently_wrong_answers		0, 1
info_value_maps	Value of information on flood maps		If the Dutch government were to provide information to you, how valuable would you find these pieces of information, from 1 (“Not valuable at all”) to 5 (“Very valuable”)?	Number between 1 and 5
info_value_compensation	Value of information on government compensation		If the Dutch government were to provide information to you, how valuable would you find these pieces of information, from 1 (“Not valuable at all”) to 5 (“Very valuable”)?	Number between 1 and 5

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
info_value_insurance	Value of information on insurance		If the Dutch government were to provide information to you, how valuable would you find these pieces of information, from 1 (“Not valuable at all”) to 5 (“Very valuable”)?	Number between 1 and 5
info_value_nature	Value of information on water management and nature conservation		If the Dutch government were to provide information to you, how valuable would you find these pieces of information, from 1 (“Not valuable at all”) to 5 (“Very valuable”)?	Number between 1 and 5
wtp_scenario	Scenario drawn in incentivized willingness-to-pay for information lottery (among the 11 scenarios)			Number between 1 and 6 in 0.5 increment
wtp_answer	Survey respondent’s answer for the scenario drawn in incentivized willingness-to-pay for information lottery (among the 11 answers to 11 scenarios)			I want to receive the money, I want to receive the information
wtp_insurance	Stated willingness-to-pay for hypothetcal insurance contract		Imagine there was an insurance policy that completely covered flood damages to your house and home contents. How much would your household be willing to pay for a monthly premium (in EUR)? You can type 0 as an answer.	Non-negative integer
wtp_insurance_wins975	Stated willingness-to-pay for hypothetcal insurance contract, winsorized at the 97.5th percentile	wtp_insurance		Non-negative integer
wtp_insurance_wins99	Stated willingness-to-pay for hypothetcal insurance contract, winsorized at the 99th percentile	wtp_insurance		Non-negative integer
wtp_info	Willigness-to-pay for information	get_info_1EUR, get_info_1.5EUR, get_info_2EUR, get_info_2.5EUR, get_info_3EUR, get_info_3.5EUR, get_info_4EUR, get_info_4.5EUR, get_info_5EUR, get_info_5.5EUR, get_info_6EUR		0 or -99 (representing malformed willingness-to-pay) or number between 1 and 6 in 0.5 increment

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
get_info_1EUR	Would like to receive information for 1 EUR		If you want, we can share with you information on which Dutch insurance companies have extended coverage for flood damages. We want to understand how valuable this information is for Dutch households. Remember that you may receive money for filling out this survey. You can choose between receiving this piece of information or receiving more money in case you are selected for payment. For each amount of money below, please choose whether you would prefer the piece of information or more money. Of all the choices, only one will be selected at random and your decision for that amount will be implemented.	0, 1
get_info_1.5EUR	Would like to receive information for 1.5 EUR		If you want, we can share with you information on which Dutch insurance companies have extended coverage for flood damages. We want to understand how valuable this information is for Dutch households. Remember that you may receive money for filling out this survey. You can choose between receiving this piece of information or receiving more money in case you are selected for payment. For each amount of money below, please choose whether you would prefer the piece of information or more money. Of all the choices, only one will be selected at random and your decision for that amount will be implemented.	0, 1

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
get_info_2EUR	Would like to receive information for 2 EUR		If you want, we can share with you information on which Dutch insurance companies have extended coverage for flood damages. We want to understand how valuable this information is for Dutch households. Remember that you may receive money for filling out this survey. You can choose between receiving this piece of information or receiving more money in case you are selected for payment. For each amount of money below, please choose whether you would prefer the piece of information or more money. Of all the choices, only one will be selected at random and your decision for that amount will be implemented.	0, 1
get_info_2.5EUR	Would like to receive information for 2.5 EUR		If you want, we can share with you information on which Dutch insurance companies have extended coverage for flood damages. We want to understand how valuable this information is for Dutch households. Remember that you may receive money for filling out this survey. You can choose between receiving this piece of information or receiving more money in case you are selected for payment. For each amount of money below, please choose whether you would prefer the piece of information or more money. Of all the choices, only one will be selected at random and your decision for that amount will be implemented.	0, 1

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
get_info_3EUR	Would like to receive information for 3 EUR		If you want, we can share with you information on which Dutch insurance companies have extended coverage for flood damages. We want to understand how valuable this information is for Dutch households. Remember that you may receive money for filling out this survey. You can choose between receiving this piece of information or receiving more money in case you are selected for payment. For each amount of money below, please choose whether you would prefer the piece of information or more money. Of all the choices, only one will be selected at random and your decision for that amount will be implemented.	0, 1
get_info_3.5EUR	Would like to receive information for 3.5 EUR		If you want, we can share with you information on which Dutch insurance companies have extended coverage for flood damages. We want to understand how valuable this information is for Dutch households. Remember that you may receive money for filling out this survey. You can choose between receiving this piece of information or receiving more money in case you are selected for payment. For each amount of money below, please choose whether you would prefer the piece of information or more money. Of all the choices, only one will be selected at random and your decision for that amount will be implemented.	0, 1

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
get_info_4EUR	Would like to receive information for 4 EUR		If you want, we can share with you information on which Dutch insurance companies have extended coverage for flood damages. We want to understand how valuable this information is for Dutch households. Remember that you may receive money for filling out this survey. You can choose between receiving this piece of information or receiving more money in case you are selected for payment. For each amount of money below, please choose whether you would prefer the piece of information or more money. Of all the choices, only one will be selected at random and your decision for that amount will be implemented.	0, 1
get_info_4.5EUR	Would like to receive information for 4.5 EUR		If you want, we can share with you information on which Dutch insurance companies have extended coverage for flood damages. We want to understand how valuable this information is for Dutch households. Remember that you may receive money for filling out this survey. You can choose between receiving this piece of information or receiving more money in case you are selected for payment. For each amount of money below, please choose whether you would prefer the piece of information or more money. Of all the choices, only one will be selected at random and your decision for that amount will be implemented.	0, 1

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
get_info_5EUR	Would like to receive information for 5 EUR		If you want, we can share with you information on which Dutch insurance companies have extended coverage for flood damages. We want to understand how valuable this information is for Dutch households. Remember that you may receive money for filling out this survey. You can choose between receiving this piece of information or receiving more money in case you are selected for payment. For each amount of money below, please choose whether you would prefer the piece of information or more money. Of all the choices, only one will be selected at random and your decision for that amount will be implemented.	0, 1
get_info_5.5EUR	Would like to receive information for 5.5 EUR		If you want, we can share with you information on which Dutch insurance companies have extended coverage for flood damages. We want to understand how valuable this information is for Dutch households. Remember that you may receive money for filling out this survey. You can choose between receiving this piece of information or receiving more money in case you are selected for payment. For each amount of money below, please choose whether you would prefer the piece of information or more money. Of all the choices, only one will be selected at random and your decision for that amount will be implemented.	0, 1

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
get_info_6EUR	Would like to receive information for 6 EUR		If you want, we can share with you information on which Dutch insurance companies have extended coverage for flood damages. We want to understand how valuable this information is for Dutch households. Remember that you may receive money for filling out this survey. You can choose between receiving this piece of information or receiving more money in case you are selected for payment. For each amount of money below, please choose whether you would prefer the piece of information or more money. Of all the choices, only one will be selected at random and your decision for that amount will be implemented.	0, 1
cbond_link_click	Whether survey respondent clicked on the link to the Consumntenbond blog post			0, 1
cbond_info_time_first_click	Time elapsed before first click on Consumentenbond blog post survey screen (in seconds)			Non-negative number
cbond_info_time_last_click	Time elapsed before last click on Consumentenbond blog post survey screen (in seconds)			Non-negative number
cbond_info_time_page_submit	Time spent on Consumentenbond blog post survey screen (in seconds)			Non-negative number
cbond_info_time_click_count	Times clicked on Consumentenbond blog post survey screen			Non-negative integer
worry_RE_present	Whether the survey respondent gave the outcome “worry”	worry_RE		0, 1
risk_RE_present	Whether the survey respondent gave the outcome “risk”	risk_RE		0, 1
damages_RE_present	Whether the survey respondent gave the outcome “damages”	damages_RE		0, 1
comptot_RE_present	Whether the survey respondent gave the outcome “comptot”	comptot_RE		0, 1
compshare_RE_present	Whether the survey respondent gave the outcome “compshare”	compshare_RE		0, 1
wtp_info_present	Whether the survey respondent gave the outcome “wtp_info”	wtp_info		0, 1
wtp_insurance_present	Whether the survey respondent gave the outcome “wtp_insurance”	wtp_insurance		0, 1
any_outcome	At least one outcome is present	worry_RE, risk_RE, damages_RE, comptot_RE, compshare_RE, wtp_info, wtp_insurance		0, 1
treatment	Experimental arm the survey respondent is randomly assigned to			1, 2, 3, 4

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
text_time_first_click	Time elapsed before first click on experimental text survey screen (in seconds)			Non-negative number
text_time_last_click	Time elapsed before last click on experimental text survey screen (in seconds)			Non-negative number
text_time_page_submit	Time spent on experimental text survey screen (in seconds)			Non-negative number
text_time_click_count	Times clicked on experimental text survey screen			Non-negative integer
info_time_first_click	Time elapsed before first click on survey screen with correct answers to information-based questions (in seconds)			Non-negative number
info_time_last_click	Time elapsed before last click on survey screen with correct answers to information-based questions (in seconds)			Non-negative number
info_time_page_submit	Time spent on survey screen with correct answers to information-based questions (in seconds)			Non-negative number
info_time_click_count	Times clicked on survey screen with correct answers to information-based questions			Non-negative integer
text_attention	Answer to attention check question, also referred to as text comprehension question		For treatment 1: Which animals and plants are found in the Rivierpark Maasvallei?, for treatment 2: Under the "Small probability" scenario (1 flood every 1000 years) and under the "Medium probability" scenario (1 flood every 100 years)… , for treatment 3: Under the WTS act, does the government fully compensate all flood damages?, for treatment 4: After the floods in Limburg in July 2021, insurance companies covered…	For treatment 1: Rare zinc and lime flora OR Lime flora, skylarks, and goldfinches OR Rare zinc, lime flora, and beavers OR Skylarks, goldfinches, and beavers, for treatment 2: The address floods OR The address floods only with "Small probability" OR The address floods only with "Medium probability" OR The address does not flood, for treatment 3: Yes, in all cases OR Yes, as long as the damages were not insurable, not recoverable, and not preventable OR Not necessarily, even if the damages were not insurable, not recoverable, and not preventable OR Never, even if the damages were not insurable, not recoverable, and not preventable, for treatment 4: Over 200 million euros in damages, 95% of the total damages from the floods OR Over 200 million euros in damages, 95% of the damage claims they received from households OR Over 350 million euros in damages, between 35% and 60% of the total damages from the floods OR Over 600 million euros in damages, 95% of the damage claims they received from households

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
text_attention_passed	Whether the respondent passed the attention check	text_attention, flood_max_1_in_100_years, flood_max_1_in_1000_years		0, 1
text_easy_to_understand	The extent to which the survey respondent finds the experimental text easy to understand (in words)		Did you find that this text was easy to understand?	Not at all, Not so much, Somewhat, Very, Extremely
text_easy_to_understand_numeric	The extent to which the survey respondent finds the experimental text easy to understand (numeric)	text_easy_to_understand		Integer between 1 and 5
text_nice_read	The extent to which the survey respondent finds the experimental text a nice read (in words)		Did you find that this text was a nice read?	Not at all, Not so much, Somewhat, Very, Extremely
text_nice_read_numeric	The extent to which the survey respondent finds the experimental text a nice read (numeric)	text_nice_read		Integer between 1 and 5
text_emotions	The extent to which the survey respondent finds that the experimental text conveyed emotions (in words)		Did you find that this text conveyed emotions?	Not at all, Not so much, Somewhat, Very, Extremely
text_emotions_numeric	The extent to which the survey respondent finds that the experimental text conveyed emotions (numeric)	text_emotions		Integer between 1 and 5
clicks_maps	Times the survey respondent has clicked on the maps toggle box			Non-negative integer
clicks_maps_indicator	Whether the survey respondent has clicked on the maps toggle box	clicks_maps		0, 1
clicks_wts	Times the survey respondent has clicked on the WTS toggle box			Non-negative integer
clicks_wts_indicator	Whether the survey respondent has clicked on the WTS toggle box	clicks_wts		0, 1
clicks_insurance	Times the survey respondent has clicked on the insurance toggle box			Non-negative integer
clicks_insurance_indicator	Whether the survey respondent has clicked on the insurance toggle box	clicks_insurance		0, 1
clicks_decoy	Times the survey respondent has clicked on the decoy (nature conservation) toggle box			Non-negative integer
clicks_decoy_indicator	Whether the survey respondent has clicked on the decoy (nature conservation) toggle box	clicks_decoy		0, 1
clicked_how_many_boxes	Number of boxes clicked by the survey respodent	clicks_maps, clicks_wts, clicks_insurance, clicks_decoy		Integer between 0 and 3
clicked_boxes_0	Whether the survey respondent clicked 0 of the toggle boxes	clicked_how_many_boxes		0, 1
clicked_boxes_1	Whether the survey respondent clicked 1 of the toggle boxes	clicked_how_many_boxes		0, 1
clicked_boxes_2	Whether the survey respondent clicked 2 of the toggle boxes	clicked_how_many_boxes		0, 1
clicked_boxes_3	Whether the survey respondent clicked 3 of the toggle boxes	clicked_how_many_boxes		0, 1
first_box_clicked	Name of the first toggleable box clicked by the survey respondent	timestamps_maps, timestamps_wts, timestamps_insurance, timestamps_decoy		0, 1

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
second_box_clicked	Name of the second toggleable box clicked by the survey respondent	timestamps_maps, timestamps_wts, timestamps_insurance, timestamps_decoy		0, 1
third_box_clicked	Name of the third toggleable box clicked by the survey respondent	timestamps_maps, timestamps_wts, timestamps_insurance, timestamps_decoy		0, 1
read_all_flood_texts	Survey respondent has read all flood-related texts	clicks_maps_indicator, clicks_wts_indicator, clicks_insurance_indicator		0, 1
timestamps_maps	Time stamp(s) of when the maps toggle box was clicked			Time stamp
timestamps_wts	Time stamp(s) of when the WTS toggle box was clicked			Time stamp
timestamps_insurance	Time stamp(s) of when the insurance toggle box was clicked			Time stamp
timestamps_decoy	Time stamp(s) of when the decoy (nature conservation) toggle box was clicked			Time stamp
timestamps_submit	Time stamp(s) of when the submit button was clicked			Time stamp
total_seconds_maps	Total seconds spent on maps toggle box	timestamps_maps, timestamps_wts, timestamps_insurance, timestamps_decoy, timestamps_submit		Non-negative number
conditional_total_seconds_maps	Total seconds spent on maps toggle box, conditional on clicking (0 seconds become missing values)	total_seconds_maps		Non-negative number
conditional_total_seconds_maps_wins975	Total seconds spent on maps toggle box, conditional on clicking (0 seconds become missing values), winsorized at th 97.5th percentile	total_seconds_maps		Non-negative number
conditional_total_seconds_maps_wins99	Total seconds spent on maps toggle box, conditional on clicking (0 seconds become missing values), winsorized at th 99th percentile	total_seconds_maps		Non-negative number
total_seconds_wts	Total seconds spent on WTS toggle box	timestamps_maps, timestamps_wts, timestamps_insurance, timestamps_decoy, timestamps_submit		Non-negative number
conditional_total_seconds_wts	Total seconds spent on WTS toggle box, conditional on clicking (0 seconds become missing values)	total_seconds_wts		Non-negative number
conditional_total_seconds_wts_wins975	Total seconds spent on WTS toggle box, conditional on clicking (0 seconds become missing values), winsorized at th 97.5th percentile	total_seconds_wts		Non-negative number
conditional_total_seconds_wts_wins99	Total seconds spent on WTS toggle box, conditional on clicking (0 seconds become missing values), winsorized at th 99th percentile	total_seconds_wts		Non-negative number

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
total_seconds_insurance	Total seconds spent on insurance toggle box	timestamps_maps, timestamps_wts, timestamps_insurance, timestamps_decoy, timestamps_submit		Non-negative number
conditional_total_seconds_insurance	Total seconds spent on insurance toggle box, conditional on clicking (0 seconds become missing values)	total_seconds_insurance		Non-negative number
conditional_total_seconds_insurance_wins97	Total seconds spent on insurance toggle box, conditional on clicking (0 seconds become missing values), winsorized at th 97.5th percentile	total_seconds_insurance		Non-negative number
conditional_total_seconds_insurance_wins99	Total seconds spent on insurance toggle box, conditional on clicking (0 seconds become missing values), winsorized at th 99th percentile	total_seconds_insurance		Non-negative number
total_seconds_decoy	Total seconds spent on decoy (nature conservation) toggle box	timestamps_maps, timestamps_wts, timestamps_insurance, timestamps_decoy, timestamps_submit		Non-negative number
conditional_total_seconds_decoy	Total seconds spent on decoy (nature conservation) toggle box, conditional on clicking (0 seconds become missing values)	total_seconds_decoy		Non-negative number
conditional_total_seconds_decoy_wins975	Total seconds spent on decoy (nature conservation) toggle box, conditional on clicking (0 seconds become missing values), winsorized at th 97.5th percentile	total_seconds_decoy		Non-negative number
conditional_total_seconds_decoy_wins99	Total seconds spent on decoy (nature conservation) toggle box, conditional on clicking (0 seconds become missing values), winsorized at th 99th percentile	total_seconds_decoy		Non-negative number
info_time_toggle	Total time spent on page with toggle boxes	info_time_page_submit, info_time_last_click		Non-negative number
approximate_time	Approximate time spent on the last box clicked in the toggle box survey screen. Only necessary for a hanfdul of observations in the very first pilot, where the time stamp in the submit button is missing.	info_time_toggle, total_seconds_maps, total_seconds_wts, total_seconds_insurance, total_seconds_decoy		Non-negative number
total_seconds_treatment_text	Total seconds spent on treatment text	treatment, total_seconds_decoy, total_seconds_maps, total_seconds_wts, total_seconds_insurance		Non-negative number
total_seconds_treatment_text_wins975	Total seconds spent on treatment text, winsorized at the 97.5th percentile	total_seconds_treatment_text		Non-negative number
total_seconds_treatment_text_wins99	Total seconds spent on treatment text, winsorized at the 99th percentile	total_seconds_treatment_text		Non-negative number

continues on next page

Table 1 – continued from previous page

VARIABLE	DESCRIPTION	DERIVED FROM	SURVEY QUESTION	VALUES
total_seconds_other_texts	Total seconds spent on non-treatment texts	treatment, total_seconds_decoy, total_seconds_maps, total_seconds_wts, total_seconds_insurance		Non-negative number
total_seconds_other_texts_wins975	Total seconds spent on non-treatment texts, winsorized at the 97.5th percentile	total_seconds_other_texts		Non-negative number
total_seconds_other_texts_wins99	Total seconds spent on non-treatment texts, winsorized at the 99th percentile	total_seconds_other_texts		Non-negative number
total_seconds_all_texts	Total seconds spent on all texts	total_seconds_decoy, total_seconds_maps, total_seconds_wts, total_seconds_insurance		Non-negative number
total_seconds_all_texts_wins975	Total seconds spent on all texts, winsorized at the 97.5th percentile	total_seconds_all_texts		Non-negative number
total_seconds_all_texts_wins99	Total seconds spent on all texts, winsorized at the 99th percentile	total_seconds_all_texts		Non-negative number
WEIGHTS	Sample weight			Positive number

CHAPTER

FOUR

ANALYSIS

Documentation of the code in *experiment_floodplain/analysis*. The analysis consists of three tasks:

- **Task 1**: In the folder *task_descriptive_analysis*. These scripts compute summary statistics over survey respondents' beliefs (`task_summary_beliefs.py`), information quality (`task_summary_information_frictions.py`) and reading behavior during the experimental part of the survey (`task_summary_reading_behavior.py`).
- **Task 2**: In the folder *task_check_experiment_randomization*. Checks for the integrity of the experimental randomization (`task_check_randomization.py`) and compare characteristics of survey respondents, survey recipients, and full sample (`task_check_survey_respondents.py`).
- **Task 3**: In the folder *task_estimate_treatment_effects*.

4.1 Scripts for Task 1: descriptive analysis

4.1.1 Script `task_summary_beliefs.py`

This script compute summary statistics for survey respondents' baseline beliefs. The results are saved as .csv files in *bld/analysis/beliefs*. These are:

Table 1: Summary statistics over reported beliefs

File name	Summary statistics of . . .
summary_beliefs_all.csv	All prior beliefs (10-year flood probability, damages, total and government compensation)
summary_beliefs_quartiles.csv	All prior beliefs (mean, median, and standard deviation) by quartile.
summary_beliefs_risk.csv	Prior beliefs about 10-year flood probability, by reported and true flood risk categories.
summary_beliefs_damages.csv	Prior beliefs about damages, by reported and true maximum water depth categories.
summary_beliefs_t1.csv	Prior beliefs, posterior beliefs, and belief updates under treatment 1 of the survey experiment ("neutral text").
summary_beliefs_updates.csv	All belief updates.
summary_beliefs_updates_direction.csv	Belief updates, by expected direction of the updates (based on baseline information frictions).
summary_risk_updates_direction.csv	Belief update over 10-year flood probability, by expected direction of the updates (based on baseline information frictions), tabulated by objective flood risk category.

4.1.2 Script `task_summary_information_frictions.py`

This script compute summary statistics for survey respondents' answers to information-based questions in the survey. The results are saved as .csv files in *bld/analysis/information*. These are:

Table 2: Summary statistics over information quality

File name	Summary statistics, questions on...
`frictions_maps.csv`	Flood maps
`frictions_insurance.csv`	Insurance availability.
`frictions_claims.csv`	Claims paid out by the government in July 2021.
`frictions_wts.csv`	WTS act on government compensation.
`confidently_incorrect_beliefs.csv`	Confidence in answers to information-based questions, people who are confidently incorrect vs. people who are not.

4.1.3 Script `task_summary_reading_behavior.py`

This script computes summary statistics on the reading behavior of survey respondents during the experimental part of the survey. The results are saved as .csv files in *bld/analysis/information*.

Table 3: Summary statistics over reading behavior

File name	Summary statistics for ...
`reading_behavior.csv`	Time spent reading.
`read_attention.csv`	Reading behavior conditional on people's attention to the experimental text, proxied by text comprehension question.
`read_nothing.csv`	Characteristics of people who read nothing besides the experimental text.
`clicks_vs_frictions.csv`	Results of Logit models for relationship between survey respondents' information frictions by topic and whether they click on the corresponding text.

4.2 Scripts for Task 2: randomization checks

4.2.1 Script `task_check_randomization.py`

This script checks whether the randomization worked as expected, for a range of pre-specified covariates stored as `covariates_randomization.csv` in *analysis/csv*. The prespecified covariates are listed in the pre-analysis plan (Section IV, Analysis, sub-section C, Treatment assignment, paragraph Integrity of randomization).

The results are saved as `randomization.csv` in *bld/analysis/randomization*.

4.2.2 Script `task_check_survey_respondents.py`

This script compares survey recipients to survey respondents and the full sample of eligible households, for a range of covariates stored in `covariates_respondents.csv` in *analysis/csv*. The results are saved as `samples_covariates.csv` in *bld/analysis/covariates*.

4.3 Scripts for Task 3: analysis of experimental results

4.3.1 Script `task_estimate_ATE.py`

This script estimates the Average Treatment Effects of the intervention on the (pre-specified) measures of beliefs updating and worry about flood risk, as well as the (pre-specified) measures of willigness-to-pay for insurance (hypothetical) and for information about Dutch insurance companies that offer protection against flood risk (incentivized).

I use (1) an unadjusted linear regression estimated via OLS (`outcomes_unadjusted.csv`), (2) a linear regression estimated via OLS that includes a small set of pre-specified covariates (`outcomes_precovs.csv`), and (3) two partially linear models that include a broader set of covariates where the nuisance functions are estimated via Lasso (`outcomes_rlasso_post.csv` and `outcomes_rlasso_double.csv`). I adjust the p-values for multiple hypotheses testing using the two-stage Benjamini, Krieger, and Yekutieli procedure for controlling the false discovery rate (FDR) as implemented in the Python package statsmodels (option "fdr_tbsky"). The four dataframes of results are saved as .csv files in *bld/analysis/outcomes*.

I additionally test whether survey respondents have a different probability to answer any given outcome-related question across treatment arms, via a logistic regression where the dependent variable indicates whether such outcome is present in the data. The results for it are saved as `whether_outcome_present.csv`, also in *bld/analysis/outcomes*.

4.3.2 Script `task_run_rlasso.py`

This script runs the R script `run_lasso.R` in *experiment_floodplain/analysis/R* and saves the result as .csv files to *bld/analysis*.

The the R script `run_lasso.R` uses the package hdm (High-Dimensional Metrics, [1]) to select controls for the ATE estimationg via Lasso end Post-Lasso methods for for high-dimensional approximately sparse models. The set of all controls is in *analysis/csv/formulas* (all the variables with `VARTYPE` equal to `PRE_COV`). Each .csv file contains the results of the estimation for a different (pre-specified) dependent variable. The .csv files are used by the script `task_estimate_ATE.py` in a later step.

4.3.3 Script `task_heterogeneity_analysis.py`

This script estimates heterogeneous treatment effects for a few pre-specified variables. The dataframes of results are saved as .csv files in *bld/analysis/heterogeneity*. In each file, each row represents an independent variable, while each column represents a dependent variable. These files are:

Table 4: Heterogeneous treatment effects

File name	Results for interaction with...
interaction_belief_confidence.csv	Confidence in prior beliefs (standardized), for each belief.
interaction_prior_beliefs.csv	Prior beliefs, for each belief.
interaction_confidently_wrong.csv	Indicator for respondents who give at least one confidently wrong answer. plus the two willingness-to-pay measures.
interaction_total_frictions.csv	Variable counting the number of incorrect answer in the survey section on information-based questions.
interaction_frictions_by_topic.csv	Indicator for information friction in a certain topic (flood maps, government compensation, insurance).
interaction_belief_update.csv	Belief update, for each belief.
interaction_belief_update_any.csv	Indicator for whether the respondent update their beliefs, for each belief.
interaction_flood_experience.csv	Indicator for self-reported experience of flood damages in 2021.

4.4 Auxiliary modules

Contain functions used in the scripts listed above.

4.4.1 Script descriptive_stats.py

In the folder *PYTHON*. Module to compute descriptive statistics (first step of the analysis).

assess_risk_update_direction(*df*)

Create column indicating whether belief updating over 10-year flood probability is expected or unexpected based on baseline information frictions with respect to Risicokaart flood maps.

compute_mean_and_ttest(*df*, *var*, *split_by*)

Split data under column var of df by the binary variable split_by, compute the mean of the two groups and test for their differrence (t-test, the variance of the two samples is not assumed to be equal).

compute_share_by_columns(*df*)

Replace values of each df column with its share with respect to the column total. Add row with total.

compute_share_of_updates(*df*, *belief*)

Compute share of belief update, for expected updates given baseline information frictions.

Parameters

- **df** (*Pandas.DataFrame*) – Dataframe.
- **belief** (*str*) – Belief.

Returns

Pandas.DataFrame.

compute_stats_by_treatment(*df*, *columns*)

Get mean, median, and standard deviation of columns in Pandas.DataFrame df.

compute_updating_stats(*df*, *belief*)

Compute share of belief updates, share of updates by direction, and average update size by direction conditional on treatment.

get_conditional_summary_stats(*df*, *belief*, *col*, *vals*, *colnames*)

Compute summary statistics of column belief in df, conditional on col taking each value in vals list.

Summary statistics are: minimum value, 25th quantile, mean, median, 75th quantile, maximum value, standard deviation (SD) and Mean Absolute Deviation (MAD).

Parameters

- **df** (*Pandas.DataFrame*) – DataFrame containing column belief.
- **belief** (*str*) – Name of column of which summary statistics should be computed.
- **col** (*str*) – Name of column conditional on which summary statistics for belief should be computed.
- **vals** (*list*) – Values taken by col.
- **colnames** (*list of str*) – Name of columns in DataFrame of results (one name for each value in vals).

Returns

Pandas.DataFrame.

get_covariates_dataset(*covariatesDF*, *valuesDF*, *weights=False*)

Compute (weighted) average values of variables in covariatesDF from valuesDF. Missing values are automatically excluded.

Parameters

- **covariatesDF** (*pandas.DataFrame*) – Dataframe of covariates. Need to have columns named "CATEGORY", "DESCRIPTION", "VARNAME", and "WEIGHTS".
- **valuesDF** (*pandas.DataFrame*) – Dataframes of covariates' values. Need to have a column named "VARNAME" containing the variables in covariatesDF.
- **weights** (*Bool*) – Weights to compute weighted mean. If True, will be pulled from valuesDF.WEIGHTS. Default is False

Returns

pandas.DataFrame

get_maps_frictions(*df*, *mapinfo*, *mapdict*)

Compare correct vs. stated information based on flood maps.

get_spearman_rho(*df*, *varlist*)

Compute Spearman's rho for df columns in varlist.

Parameters

- **df** (*Pandas.DataFrame*) – Must contains columns in varlist.

- **varlist** (*2-dimensional list*) – List of paired names of columns the Spearman's rho should be computed for, e.g.: [["A", "B"], ["C", "D"]] will return association coefficient between A and B, and C and D.

Returns
List of floats.

get_summary_stats(*df*, *beliefs*, *colnames=False*)

Get summary statistics for beliefs columns in df.

Summary statistics are: minimum value, 25th quantile, mean, median, 75th quantile, maximum value, standard deviation (SD) and Mean Absolute Deviation (MAD).

make_conditional_belief_updates_table(*df*, *groupby_col*, *groupby_col_name*)

Tabulate direction of belief updates over 10-year flood probability, classified based on baseline information frictions with respect to Risicokaart flood maps (return period of a flood at the address of residence), conditional on column groupby_col.

make_table_adminlevel_residents(*gdf*, *adminLevels*)

Compute number of residents in gdf dataset for each administrative level in adminLevels.

make_table_balance(*df_names*, *dfs*, *col_dict*)

Compute average values in dfs for columns specified in col_dict dictionary, as well as t-statistic and p-values for the mean difference of the two distributions.

Parameters

- **df_names** (*list of str*) – Name of two dataframes in dfs.
- **dfs** (*list of pandas.Dataframes*) – List of two dataframes.
- **col_dict** (*dict*) – Dictionary where they keys are the index labels of the pandas.DataFrame to be produces by this function, and values are the columns in dfs missing values should be computed for.

Returns
pandas.DataFrame

make_table_conditional_flood(*floodareasDF*, *return_periods*, *waterdepth_labels*, *extended=False*)

Compute the share of addresses that flood for the most likely return period in return_periods, at different maximum inundation depth. Then, for the same subsample of addresses, compute again the shares of addresses that flood at different inundation depths under all less likely scenarios.

Parameters

- **floodareasDF** (*pandas.DataFrame*) – Dataframe from which to compute shares. It is supposed to be the dataframe saved as addressesflooddata.csv in *bld.data*, restricted so that all observations have value 1 in the column FLOODABLE.
- **return_periods** (*list or tuple*) – Can be all or some values in (10, 100, 1000, 10000).
- **waterdepth_labels** (*list or tuple*) – Can be all or some values in ("less than 0.5m", "between 0.5 and 1m", "between 1 and 1.5m", "between 1.5 and 2m", "between 2 and 5m", "more than 5m").

- **extended** (*bool*) – If True, considers also houses within 500m from flooded areas.

Returns
pandas.DataFrame

make_table_conditional_flood_all_scenarios(*floodareasDF*, *waterdepth_labels*, *extended=False*)

Compute the share of addresses that flood for the most likely return period in return_periods, at different maximum inundation depth. Then, for the same subsample of addresses, compute again the shares of addresses that flood at different inundation depths under all less likely scenarios. Afterwards, repeat the previous two steps for all the return periods in return_periods, after removing the previous subsample of addresses.

Probabilities are conditional because, in the third step, an address is removed from less likely flood scenarios conditionally on flooding under a more likely one. For example, an address that floods once in 10 years is not included among those that flood once every 100 years. We then follow the (discrete) distribution function of maximum water depth under different scenarios for four separate "cohorts" of addresses (those that flood at most once every 10 years, those that flood at most once every 100 years, those that flood at most once every 1,000 years, those that flood at most once every 10,000 years).

Parameters

- **floodareasDF** (*pandas.DataFrame*) – Dataframe from which to compute shares. It is supposed to be the dataframe saved as addressesflooddata.csv in *bld.data*, restricted so that all observations have value 1 in the column FLOODABLE.
- **waterdepth_labels** (*list or tuple*) – Can be all or some values in ("less than 0.5m", "between 0.5 and 1m", "between 1 and 1.5m", "between 1.5 and 2m", "between 2 and 5m", "more than 5m").
- **extended** (*bool*) – If True, considers also houses within 500m from flooded areas.

Returns
pandas.DataFrame

make_table_exposure(*dfs*, *df_names*)

Compute number and share of exposed and not exposed areas for each dataframe in df.

Parameters

- **dfs** (*list of pandas.DataFrames or geopandas.GeoDataFrames*) – Dataframes.
- **df_names** (*list of str*) – One for each dataframe in dfs.

Returns
pandas.DataFrame

make_table_housingtype(*df_names*, *dfs*)

Compute number and shares of BAG houses by type, for each dataframe in dfs.

Parameters

- **dfs** (*list of pandas.Dataframes*) – Dataframes.

- **df_names** (*list of str*) – Name of dataframes in dfs.

Returns
pandas.DataFrame

make_table_missings(*df_names*, *dfs*, *col_dict*)

Compute count and share of missing values in dfs for columns specified in col_dict dictionary, as well as t-statistic and p-values for the mean difference of the two missing values distributions.

Parameters

- **df_names** (*list of str*) – Name of two dataframes in dfs.
- **dfs** (*list of pandas.Dataframes*) – List of two dataframes.
- **col_dict** (*dict*) – Dictionary where they keys are the index labels of the pandas.DataFrame to be produces by this function, and values are the columns in dfs missing values should be computed for.

Returns
pandas.DataFrame

make_table_target_population(*gdf*, *targetDict*)

Compute average values in gdf for variables in targetDict.

Parameters

- **gdf** (*pandas.DataFrame or geopandas.GeoDataFrame*) – Dataframe of interest.
- **targetDict** (*dict*) – Dictionary where they keys are the index labels of the pandas.DataFrame to be produces by this function, and values are the columns in dfs average values should be computed for.

Returns
pandas.DataFrame

make_table_unconditional_flood(*floodareasDF*, *flood_dict*, *return_periods*, *waterdepth_labels*, *extended=False*)

Compute the share of addresses that flood, under all scenarios, at different maximum inundation depths.

Probabilities are unconditional because an address is not removed from less likely flood scenario conditionally on flooding under a more likely one. For example, an address that floods once in 10 years is also included among those that flood once every 100 years. As a result, the same address can be included in the share of flooded houses under multiple scenarios.

Parameters

- **floodareasDF** (*pandas.DataFrame*) – Dataframe from which to compute shares. It is supposed to be the dataframe saved as addressesflooddata.csv in *bld.data*.
- **flood_dict** (*dict*) – Empty dictionary to populate. Keys need to represent the flood scenarios, e.g. “1 in 10 years”, “1 in 100 years”, “1 in 1000 years”, “1 in 10000 years”.
- **return_periods** (*list or tuple*) – Can be all or some values in (10, 100, 1000, 10000).

- **waterdepth_labels** (*list or tuple*) – Can be all or some values in ("less than 0.5m", "between 0.5 and 1m", "between 1 and 1.5m", "between 1.5 and 2m", "between 2 and 5m", "more than 5m").
- **extended** (*bool*) – If True, considers also houses within 500m from flooded areas.

Returns
pandas.DataFrame

melt_friction_vs_clicks(*df*, *topics*)

Create long dataframe linking, for each participants, information frictions by topic (in list topics) to whether the participant clicked on the associated topic box.

4.4.2 Script regressions.py

In the folder *PYTHON*. Functions shared across the two modules performing checks for the integrity of the experimental randomization, and estimation of the treatment effects.

adjust_pvalues(*df*, *treatments*, *outcomes*, *pval_col='P>|z|'*)

Adjust the p-values in df using the two-stage Benjamini, Krieger, and Yekutieli procedure for controlling the false discovery rate (FDR). Add the adjusted p-values as a new column named "pvalue_fdr_tbsky" to df.

Parameters

- **df** (*Pandas.DataFrame*) – Dataframe of interest. Need to have the column "P>|z|".
- **treatments** (*list of strings*) – Index names to subset. Need to be three.
- **outcomes** (*list of strings*) – Column names to subset. Need to be seven.
- **pval_col** (*str*) – Name of p-value column. Default is P>|z|.

Returns
Pandas.DataFrame

create_dictionary_of_covariates(*formulas_df*, *outcomes*, *add_baseline_beliefs=True*, *add_het=False*, *het_covs=[]*, *covs_to_remove=[]*)

Create dictionary relating each treatment outcome to its set of covariates.

Parameters

- **formulas_df** (*Pandas.DataFrame*) – Dataframe of covariates to use.
- **outcomes** (*list*) – List of outcomes.
- **add_baseline_beliefs** (*boolean*) – Add outcome-specific baseline beliefs and confidence in beliefs as covariates. Default is True.
- **add_het** (*boolean*) – Whether to add variables for heterogeneity analysis to the list of covariates. If True, het_covs need to be a dictionary or a list of variables. Default is False.
- **het_covs** (*list or dictionary*) – List of variables for heterogeneity analysis or, if the heterogeneity analysis is outcome-specific, dictionary where keys are outcomes and values are list of variables. Default is empty list.

- **covs_to_remove** (*list or dictionary*) – List of variables to be removed from the list of covariates, or dictionary if the variables to remove are outcome-specific. Default is empty list.

Returns
dictionary.

diff_in_mean(*df1*, *df2*, *covs*)

Compute difference in means along covs between df1 and df2.

load_rlasso_datasets(*path_dict*)

Load datasets containing outcomes of Rlasso regressions.

logit_treatment_on_response(*df*, *outcome*, *dep_vars*)

Logistic regression where the dependent variable is outcome and the independent variable(s) dep_vars need to contain variable "C(treatment)".

run_wls_regressions(*data*, *outcomes*, *depvar*)

Run werighted least square regressions of depvar on outcomes, where both are columns of the Pandas.DataFrame data.

ttest(*df1*, *df2*, *covs*)

Compute t-test for difference in means along covs between df1 and df2.

weighted_least_squares_regression(*df*, *outcome*, *dep_vars*)

Run weighted least squares regression on Pandas.DataFrame df, according to formula. df needs to contain columns "WEIGHTS" and "treatment".

Parameters

- **df** (*Pandas.DataFrame*) – Dataframe of interest.
- **outcome** (*str*) – Outcome of formula.
- **dep_vars** (*str*) – Dependent variables of linear model.

Returns
Pandas.DataFrame

wls_on_beliefs_percentile(*df*, *belief*, *outcome*, *covariates*, *percentile*, *only_keep_interactions=True*)

Run weighted least squares with interaction effect over treatment and percentile of prior beliefs.

Parameters

- **df** (*Pandas.DataFrame*) – Dataframe of interest.
- **belief** (*str*) – Belief of interest.
- **outcome** (*str*) – Outcome of interest.
- **covariates** (*list of str*) – Covariates for wls regression.
- **percentile** (*str*) – "quartile" or "tercile".
- **only_keep_interactions** (*bool*) – Whether to only store interactions coefficients in the dataframe of results. Default is True.

Returns
Pandas.DataFrame

wls_with_interaction(*df*, *outcome*, *covariates*, *interaction*, *keep_only_interaction=True*, *var_to_add=None*)

Run weighted least squares with interacted treatment.

Parameters

- **df** (*Pandas.DataFrame*) – Dataframe from which variables are extracted.
- **outcome** (*str*) – Name of outcome variable.
- **covariates** (*list of strings*) – Names of covariates.
- **interaction** (*str*) – Name of interaction variable.
- **keep_only_interaction** (*boolean*) – Keep only estimated coefficient of treatment and interacted treatment. Default is True.

Returns

Pandas.DataFrame

4.4.3 Scripts `auxiliary.R` and `run_rlasso.R`

In the folder *R*.

The script `run_rlasso.R` uses the R package hdm to select controls for the ATE estimation, via Lasso end Post-Lasso methods for high-dimensional approximately sparse models. The script `auxiliary.R` contains auxiliary functions to carry out this task.

All the functions in these scripts are duly documented in the .R files. Unfortunately, I cannot export the docstrings with Sphinx (the software that builds this documentation) because the R language is still not supported.

CHAPTER

FIVE

FINAL

Documentation of the code in *experiment_floodplain/final*. These scripts create the figures and the tables in the paper.

5.1 Scripts for visualization of results

5.1.1 Script `task_plot_beliefs.py`

This script produces all the figures related to prior and posterior beliefs distributions, confidence in beliefs, and belief updating. The figures are saved in .PNG format in *bld/figures/beliefs*.

5.1.2 Script `task_plot_information.py`

This script produces all the figure related to survey respondents' information quality. The figures are saved in .PNG format in *bld/figures/information*.

5.1.3 Script `task_plot_wtp.py`

This script produces the figures related to survey respondents' elicited willingness-to-pay for insurance. The figures are saved in .PNG format in *bld/figures/willingness_to_pay*.

5.2 Scripts to create the .tex tables

In the folder *task_fill_table_templates*. All use the .txt templates in the folder *table_templates* and the data in *bld/analysis*. The .tex files are saved to *bld/tables*.

5.3 Auxiliary modules

In folder *PYTHON*.

5.3.1 Script `visualize_beliefs.py`

Visualization module for `task_plot_beliefs.py`.

`add_variables_for_plotting`(*df*, *edges*, *labels_damages*)

Add variables for plotting figures.

`get_labels_dictionary`(*all_ticks*, *ticks_to_show*)

Create dictionary of labels for plot.

histogram_scatterplot_belief_distributions(*df*, *x*, *y*, *xlabel*)

Plot histogram and scatterplot of x vs. y from Pandas.DataFrame df.

make_jointplot(*df*, *x*, *y*, *title*, *axis_spaced_by_5=False*)

Make jointplot of varables x and y in Pandas.DataFrame df, with title.

plot_all_histograms_belief_vs_confidence(*dict_keys*, *df*, *xs*, *ys*, *list_of_bins*, *xlabels*, *list_of_xticks*, *labels_dicts*, *add_missings*, *type*)

Plot multiple histograms of beliefs vs. average confidence in beliefs, and add them to a dictionary.

Parameters

- **dict_keys** (*list*) – Keys of dictionary of results, one for each figure.
- **df** (*Pandas.DataFrame*) – Dataframe with columns to plot.
- **xs** (*list of str*) – Names of columns to be plotted on the x-axis (values of reported beliefs).
- **ys** (*list of str*) – Names of columns to be plotted on the y-axis (values of reported confidence in beliefs).
- **list_of_bins** (*list of lists*) – List of lists of number of bins for histograms.
- **xlabels** (*list of str*) – List of names for x-axis labels.
- **list_of_xticks** (*list of lists*) – List of lists of x-axis coordinates for ticks.
- **labels_dicts** (*list of lists*) – Whether to rename x-axis ticks.
- **add_missings** (*list of bool*) – List of whether to include columns with missing values in the final histograms.
- **type** (*list of str*) – List of type of plot for average confidence, either "barplot" or "pointplot".

Returns

dictionary.

plot_histogram_belief_vs_confidence(*df*, *x*, *y*, *xlabel*, *xticks*, *bins*, *labels_dict=False*, *add_missing_values=False*, *rotation=0*, *type='barplot'*)

"Plot histogram of beliefs vs. average confidence in beliefs.

Parameters

- **df** (*Pandas.DataFrame*) – Dataframe with columns to plot.
- **x** (*str*) – Name of column to be plotted on the x-axis (values of reported beliefs).
- **y** (*str*) – Name of columns to be plotted on the y-axis (values of reported confidence in beliefs).
- **xlabel** (*str*) – Name for x-axis label.
- **xticks** (*list of int*) – List of x-axis coordinates for ticks.
- **bins** (*list of int*) – Number of histogram bins.

- **labels_dict** (*dict*) – Optional, dictionary of x-axis ticks and x-axis ticks' labels.
- **add_missing_values** (*bool*) – Whether to include columns with missing values in the final histograms, default is False.
- **rotation** (*int*) – Rotation of x-axis ticks, default is 0.
- **type** (*str*) – Type of plot for average confidence, either "barplot" (default) or "pointplot".
- **Returns** – matplotlib.Figure.

plot_lower_triangular_heatmap(*df_corr*, *suptitle*, *n_obs*, *cmap*)

Plot lower triangular heatmap depicting correlation between answers to multiple choice questions (on measures against flood or sources consulted about flood risk).

Parameters

- **df** (*Pandas.DataFrame*) – Dataframe of two-ways correlations.
- **suptitle** (*Str*) – Plot main title.
- **n_obs** (*int*) – Number of observations. Will be written in the plot sub-title.
- **cmap** (*palette*) – Seaborn palette.

Returns

Matplotlib.Figure

plot_updates(*df*, *query_strings*, *titles*, *xlabel*, *ylabel*, *figsize*)

Plot belief updates by direction implied by baseline information quality.

Parameters

- **df** – Dataset containing variables of interest.
- **query_strings** – Strings to select variables of interest.
- **titles** – Titles of sub-figures.
- **xlabel** – x-axis label.
- **ylabel** – y-axis label.
- **figsize** – Figure size.

Returns

matplotlib.Figure.

5.3.2 Script visualize_information.py

Visualization module for `task_plot_information.py`.

get_df_for_plotting(*df*, *noise*)

Melt dataframe df so that each answer to an information-based question is classified as correct or incorrect (value 1 or 0), by topic (flood maps, insurance, government compensation), and by confidence in the answer (number from 1 to 10).

Parameters

- **df** (*Pandas.DataFrame*) – Dataframe of interest

- **noise** (*float*) – Add noise to confidence variable. Useful to get a nicer swarmplot.

Returns
Pandas.DataFrame

plot_information_vs_confidence(*df*, *noise=0*)

Create a Seaborn swarmplot showing confidence level for incorrect vs. correct answers.

plot_total_information_vs_confidence(*df*)

Plot two histograms next two each other.

The first histogram has number of information frictions on the x-axis and share of survey respondets on the y-axis.

The second histogram has average confidence in answers to information-based questions on the x-axis and share of survey respondents on the y-axis.

The Pandas.DataFrame df needs to contain the columns "total_frictions" and "average_info_confidence".

scatter_dataframe(*df*, *cols*, *increment*)

Scatter valued of cols in df, by increment.

Args:
df (Pandas.DataFrame): Dataset. cols (list of strings): Column(s) of df whose values should be scattered. increment (float): By how much should the values of cols be scattered.

Returns
Pandas.DataFrame.

5.3.3 Script visualize_wtp.py

Visualization module for `task_plot_wtp.py`.

plot_insurance_two_arms(*data*)

Plot who stays and leaves the flood insurance market, by flood risk category and treament arms ("Neutral text" vs. "Risk profile"). Include average WTP by flood risk category for those who stay.

5.3.4 Script format_tables.py

Functions to format tables.

get(*data*, *index1*, *index2=None*)

Get values from specified index position of Pandas.DataFrame data, as list, excluding NaNs.

split_dataset(*df*, *cols*, *p_val=False*)

Extract dataset of coefficients and of standard deviations from Pandas.DataFrame df and column nams cols. Add stars to coefficients according to pvalues.

update(*key*, *n_keys*, *values*, *kwargs={}*)

Create dictionary from key, n_keys and values and update whatever kwargs dictionary is in the global space.

CHAPTER

SIX

REPLICATION OF RESULTS

This project will output a .pdf document named `paper_results_replication.pdf` (under *bld/latex*, a copy is in the root folder), which contains all tables and figures in the paper **as long as the files** `original_survey_data.csv` **and** `rvo_pc6.csv` **are placed under** *src/experiment_floodplain/data*. Otherwise, the project will use synthetic data (thus merely showing that the code works).

CHAPTER

SEVEN

REFERENCES

BIBLIOGRAPHY

[1] Victor Chernozhukov, Chris Hansen, and Martin Spindler. hdm: high-dimensional metrics. *R Journal*, 8(2):185–199, 2016. URL: https://journal.r-project.org/archive/2016/RJ-2016-040/index.html.

PYTHON MODULE INDEX

a

d

f

INDEX

A

C

D

E

F

G

H

I

L

M

P

R

S

T

U

W