Example Notebook to Analyse and Visualize Data from a Micro Scenario-based Study

Philipp Brauner

Abstract

This R notebook serves as an example, demonstrating the analysis and visualization of micro-scenario-based studies. Micro-scenarios provide an approach for evaluating the social acceptance of technologies and the determining factors, along with a visual-spatial mapping of the results. They enable the simultaneous assessment of multiple technologies, ranking them based on different criteria, and analyzing how individual factors and technology-based attributions correlate with the overall assessment of technologies. Utilizing synthetic survey data (generated in a separate notebook), this notebook illustrates how to recode the data, aggregate scenario scores as user factors, calculate topic scores, and visualize them using the R programming language, along with `ggplot` and `tidyverse`.

Introduction

The micro-scenario approach simplifies measuring people's opinions on different topics. It connects these opinions to individual user factors (research perspective 1), ranks the topics, and creates a visual map to pinpoint conflicting issues (research perspective 2), all within a single survey.

For instance, consider analysing risk-utility trade-offs among various technologies: Do individuals attribute varying risks and utilities to distinct technologies? Are people predisposed to different risk or utility perceptions? Is the comparability of risk-utility trade-offs consistent across different technologies, and can these trade-offs be quantified? Figure 1 illustrates the overall approach.

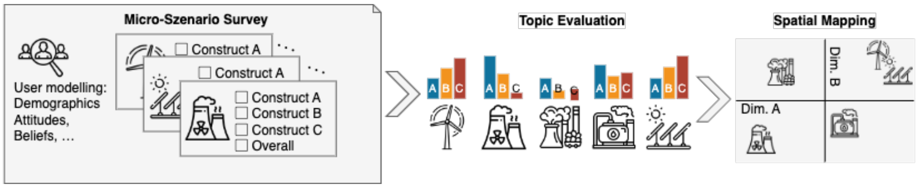


Figure 1: The micro-scenario approach involves consolidating evaluations of diverse topics in a single survey. These evaluations are treated as topic assessments and spatially mapped to analyze the interrelationships among them.

The main article provides comprehensive insights into this approach and outlines the methodology for designing and analyzing studies. You can locate and cite the main article here:

> Mapping Acceptance: Assessing Emerging Technologies and Concepts through Micro Scenarios, Philipp Brauner, http://arxiv.org/abs/2402.01551 (2024)

This notebook demonstrates the calculation of data for the two research perspectives of the micro-scenario approach (Perspective 1: user factor and Perspective 2: topic factor) using R. Note that all transformations and calculations can also be performed using other software.

In this example, we utilize synthetic data generated to resemble real survey data. This choice simplifies the follow-through of our approach, eliminating the need for cleaning the data from irrelevant variables or erroneous participant inputs. Additionally, the synthetic data adheres to pre-specified properties. The creation of the synthetic data is detailed in the companion notebook within the same folder.

The rest of this notebook is organized as follows: Firstly, we load the necessary packages, followed by loading the synthetic data as our input (replace this with your actual data). Secondly, we transform the data into the long format (refer to, for instance, https://tidyr.tidyverse.org/reference/pivot_longer.html), proceed to analyse the data as a user factor (research perspective 1), and subsequently as a topic factor which includes visualizing the outcomes (research perspective 2).

Preparation

Load required libraries

In our analysis, we mainly use the `tidyverse` and `ggplot` packages.

```r
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.4
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.4.4     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.0
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

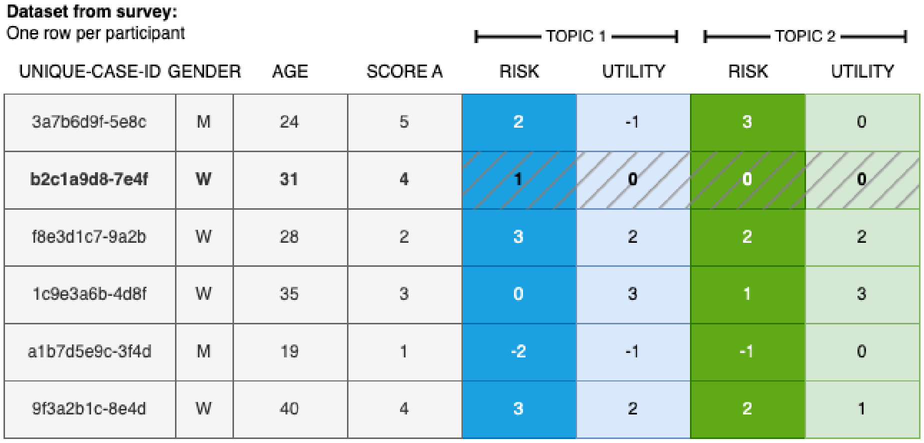
```r
library(scales)  # format_percent
```

```
Attaching package: 'scales'

The following object is masked from 'package:purrr':

    discard

The following object is masked from 'package:readr':

    col_factor
```

```r
library(ggplot2) # graphics
library(ggrepel) # label placement in the scatter plot
library(knitr)   # Tables
```

Load Data

In this demonstration, we will load the synthetic data that emulates the properties found in real survey data. The other notebook demonstrates the creation of the synthetic data. Figure 2 illustrates the structure of a standard dataset from survey tools, where each row represents the responses from an individual participant.



Dataset from survey:
One row per participant

				TOPIC 1		TOPIC 2	
UNIQUE-CASE-ID	GENDER	AGE	SCORE A	RISK	UTILITY	RISK	UTILITY
3a7b6d9f-5e8c	M	24	5	2	-1	3	0
b2c1a9d8-7e4f	**W**	**31**	**4**	**1**	**0**	**0**	**0**
f8e3d1c7-9a2b	W	28	2	3	2	2	2
1c9e3a6b-4d8f	W	35	3	0	3	1	3
a1b7d5e9c-3f4d	M	19	1	-2	-1	-1	0
9f3a2b1c-8e4d	W	40	4	3	2	2	1

Figure 2: Illustration of typical survey data utilizing the micro-scenario approach, featuring user demographics, additional user factors, and topic evaluations.

The data structure closely resembles the data export from the Qualtrics survey tool. The process of generating synthetic data is documented in the companion notebook within the same folder.

```
data <- readRDS(”syntheticdata.rds”)
```

The structure of the data looks as follows:

```
'data.frame':   100 obs. of  26 variables:
 $ id          : chr  ”fakeparticipantid-1” ”fakeparticipantid-2” ”fakeparticipantid-3” ”fakeparticipantid-4” ...
 $ uservariable: num  12.01 10.62 6.69 9.05 11.82 ...
 $ a1_matrix_1 : num  2 1 1 1 3 2 1 4 3 3 ...
 $ a2_matrix_1 : num  3 2 2 2 2 3 3 3 4 1 ...
 $ a3_matrix_1 : num  2 2 2 1 4 3 2 4 2 3 ...
 $ a4_matrix_1 : num  3 3 2 3 5 4 3 4 2 3 ...
 $ a5_matrix_1 : num  3 3 2 3 4 4 4 5 3 3 ...
 $ a6_matrix_1 : num  5 4 3 4 4 5 3 4 2 3 ...
 $ a7_matrix_1 : num  4 4 3 3 4 5 3 5 4 3 ...
 $ a8_matrix_1 : num  4 4 3 4 5 5 4 6 5 4 ...
 $ a9_matrix_1 : num  5 5 3 3 6 5 4 5 5 4 ...
 $ a10_matrix_1: num  6 5 5 4 4 6 5 6 5 4 ...
 $ a11_matrix_1: num  6 5 5 6 6 7 6 7 6 5 ...
 $ a12_matrix_1: num  5 5 5 4 6 7 5 7 5 6 ...
 $ a1_matrix_2 : num  1 3 3 3 2 1 1 2 3 2 ...
 $ a2_matrix_2 : num  1 2 3 3 2 2 2 1 3 3 ...
 $ a3_matrix_2 : num  1 2 2 1 2 2 2 1 2 2 ...
 $ a4_matrix_2 : num  2 2 4 4 2 2 2 1 2 3 ...
 $ a5_matrix_2 : num  4 4 5 4 4 3 5 4 6 4 ...
 $ a6_matrix_2 : num  3 3 3 3 2 1 3 2 4 3 ...
 $ a7_matrix_2 : num  3 3 4 3 4 3 2 3 4 3 ...
 $ a8_matrix_2 : num  2 4 4 4 3 3 3 2 2 2 ...
 $ a9_matrix_2 : num  3 4 5 3 3 3 3 1 3 4 ...
 $ a10_matrix_2: num  3 3 4 5 3 3 4 3 4 4 ...
 $ a11_matrix_2: num  3 3 4 4 3 4 4 2 3 4 ...
 $ a12_matrix_2: num  4 4 5 6 4 3 5 4 6 4 ...
```

The loaded dataset has various variables. Initially, there's a unique user identifier (`id`), followed by a user variable (e.g., attitude towards a topic). Subsequently, there are an arbitrary number of topic assessments (N in our example) with variables for each evaluation dimension. In this instance, we use *perceived risk* and *perceived utility* as examples for the topic evaluations. However, one can employ different or additional evaluation dimensions (as detailed in the article).

The variables for the topic evaluations adhere to a standardized naming scheme, i.e., `a01_matrix_02`, where `01` denotes the ID of the queried topic, `02` represents the queried evaluation dimension, and `matrix` stands for the name of the variable block in the survey tool. This naming scheme is employed by Quartics.

Analysis of the data

Once the (synthetic) survey data is loaded into the variable `data`, we can commence the actual analysis.

Setup

Firstly, read the list of queried topics and their labels from a `.csv` file (adjustable based on your needs). Secondly, define the queried evaluation dimensions. In this instance, we have a vector of two dimensions, but one can define more based on your research questions and survey structure.

```
TOPICS <- read.csv2(”matrixlabels.csv”)
DIMENSIONS = c(”risk”, ”utility”)
```

Long Format

Next, the topic evaluations from the survey data is transformed into the long format using `pivot_longer` (one row with a single value for each participant, topic, and evaluation dimension; one row per observation). Hereto, we use that the variables for the topic evaluations in the original data table have a systematic naming convention (see above).

The resulting data set contains a participant identifier, identifier for the topic and the evaluation dimension, and lastly a column for the value. We use this format as the foundation for the later transformation steps.

```
evaluationsLong <- data %>%
  # selects columns id and ”aNUMBER_matrix_NUMBER” (scheme from loop&merge)
  dplyr::select(id, matches(”a\\d+\\_matrix\\_\\d+”)) %>%
  tidyr::pivot_longer(
    cols = matches(”a\\d+\\_matrix\\_\\d+”), # topics and their evaluations
    names_to = c(”question”, ”dimension”),
    names_pattern = ”(.*)_matrix_(.*)”,   # Separate topic ID and evaluation ID
    values_to = ”value”,
    values_drop_na = FALSE) %>%
    dplyr::mutate( dimension = as.numeric(dimension) ) %>%
    dplyr::mutate( dimension = DIMENSIONS[dimension]) %>%  # change to readable dimension names
    dplyr::mutate( value = -(((value - 1)/3) - 1))  # rescale value from [ 1...7 ] to [ -100%...100% ]

# Recode some of the evaluation dimensions if necessary
evaluationsLong <- evaluationsLong %>%
  dplyr::mutate( value = if_else(dimension!=”risk”, value, -value))
```

Perspective 1: As user factor

The initial perspective provides a straightforward view of the data. The different presented scenarios serve as a basis for the repeated measurement of the same latent construct and the resulting score can be interpreted as a user factor (or individual differences).

For each evaluation dimension (e.g., *risk* and *utility*), we compute average scores across all queried topics. Using these cores one can, for instance, investigate if the overall attributions differ among participants or if they correlate with other queried user factors. For example, exploring if the average risk attributed to all topics relates to a general disposition to risk measured using other psychometric scales.

Subsequently, we rejoin these user factors with the original data using, for instance, `dplyr::left_join()`. Afterwards, the calculated average evaluations can be regarded as individual differences and correlated with other user factors obtained from the survey.

```
evaluationByParticipant <- evaluationsLong %>%
  tidyr::pivot_wider(names_from = dimension, values_from = value) %>%
  dplyr::group_by(id) %>%
  dplyr::summarize(
    across(
      all_of( DIMENSIONS ),  # Select only evaluation dimensions
      list( mean = ~mean(., na.rm = TRUE),
#            median = ~median(., na.rm = TRUE),
            sd = ~sd(., na.rm = TRUE)),
      .names = ”{.col}_{.fn}”  # Scheme to define column names
    ), .groups=”drop”
  ) %>%
  dplyr::left_join(data, by=”id”)
```

Example research questions:

How is the average perceived *risk* of the participants?

```
summary(evaluationByParticipant$risk_mean)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
-0.5278 -0.2500 -0.1111 -0.1303  0.0000  0.3333 
```

How is the average perceived *utility* of the participants?

```
summary(evaluationByParticipant$utility_mean)
```

```
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
-0.05556  0.15972  0.30556  0.27583  0.36111  0.66667 
```

Does *uservariable* from the survey correlate to *risk*?

```
cor.test(evaluationByParticipant$risk_mean,
    evaluationByParticipant$uservariable)
```

```

    Pearson's product-moment correlation

data:  evaluationByParticipant$risk_mean and evaluationByParticipant$uservariable
t = 3.4023, df = 98, p-value = 0.0009689
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.1373725 0.4901478
sample estimates:
      cor 
0.3250207 
```

Does uservariable correlate with the average *perceived utility*?

```
cor.test(evaluationByParticipant$utility_mean,
    evaluationByParticipant$uservariable)
```

```

    Pearson's product-moment correlation

data:  evaluationByParticipant$utility_mean and evaluationByParticipant$uservariable
t = 2.7281, df = 98, p-value = 0.007551
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.07306894 0.43917467
sample estimates:
      cor 
```

```
0.2656741
```

Perspective 2: Topic factors

Next, we delve into the analysis of topic evaluations: Instead of looking at how individual participants perceive the topics as a whole, only all test subject assessments are assigned to the respective topics, for example in order to be able to rank the technologies in terms of the evaluation dimensions. We start with reporting the average evaluations (e.g., *risk* and *utility*) across all queried topics.

Calculate Average evaluations

Using the long format, we group by evaluation dimension and aggregate across all topics and participants. Note: For a complete sample, the results are equivalent to perspective 1 (see above). Table 1 and Figure 3 show the outcome of this calculation.

```
# MEAN and SD of all evaluation dimensions across all queried topics
evaluationByDimension <- evaluationsLong %>%
  dplyr::group_by( dimension ) %>%
  dplyr::summarise( mean = mean(value, na.rm = TRUE),
                    sd = sd(value, na.rm = TRUE),
                    .groups=”drop”)
```

Table 1: Averages for each evaluation dimension across all queried topics and across all participants.

dimension	mean	sd
risk	-0.13	0.45
utility	0.28	0.39

```
overallDimension <- ggplot(evaluationByDimension,
  aes(x = dimension, y = mean)) +
  geom_bar(stat = ”identity”) +
  scale_y_continuous(labels = percent_format(),
                     limits=c( -1, +1 )) +
  labs(x = ”Evaluation Dimension”,
       y = ”Values”,
       title = ”Average Evaluation across all Dimensions and Participants”)
overallDimension
```

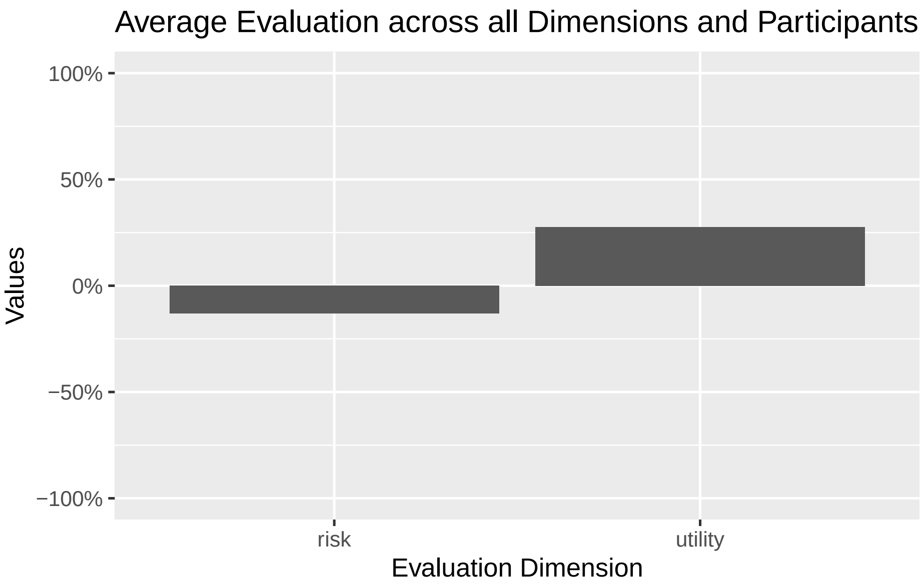
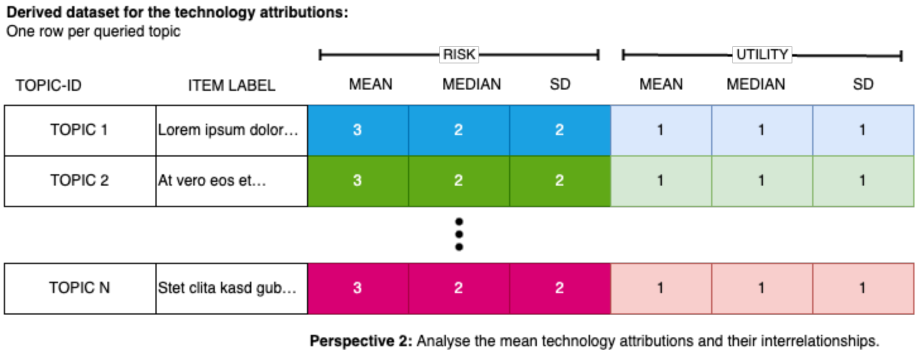


Figure 3: Mean evaluation across all topics and aggregated across all participants.

Prepare Individual Topics

Now, we compute the average evaluations for each topic across all participants. The resulting data frame contains N rows for the number of topics queried and rows for the arithmetic mean and standard deviation for each evaluated dimension (e.g., *risk* and *utility*). Finally, we associate labels with each topic using `dplyr::left_join()`. Figure 4 illustrates the structure of the resulting data.



Derived dataset for the technology attributions:
One row per queried topic

		RISK			UTILITY		
TOPIC-ID	ITEM LABEL	MEAN	MEDIAN	SD	MEAN	MEDIAN	SD
TOPIC 1	Lorem ipsum dolor…	3	2	2	1	1	1
TOPIC 2	At vero eos et…	3	2	2	1	1	1
			⋮				
TOPIC N	Stet clita kasd gub…	3	2	2	1	1	1

Perspective 2: Analyse the mean technology attributions and their interrelationships.

Figure 4: The resulting data format displays the evaluation of topics. Each row contains the mean evaluation (along with its dispersion) for a specific topic. This structured data can be subjected to further analysis.

The output can be tabulated, sorted, or filtered based on highest/lowest evaluations, and visualized. Table 2 displays the unsorted and unfiltered results.

```r
evaluationByTopic <-  evaluationsLong %>%
  tidyr::pivot_wider(
    names_from = dimension,
    values_from = value) %>%
  dplyr::group_by( question ) %>%
  dplyr::summarize(
    across(
      all_of( DIMENSIONS ),  # Select the variables from var_names
      list(mean = ~mean(., na.rm = TRUE),
           sd = ~sd(., na.rm = TRUE)),
      .names = ”{.col}_{.fn}”    # Define column names for the results
    ), .groups=”drop”
  ) %>%
  dplyr::left_join(TOPICS, by=”question”) # attach question labels
```

Table 2: Average evaluation of the queried topics.

label	risk_mean	risk_sd	utility_mean	utility_sd
Topic 1	-0.65	0.25	0.67	0.25
Topic 10	0.20	0.24	0.05	0.24
Topic 11	0.47	0.28	0.03	0.26
Topic 12	0.38	0.25	-0.12	0.27
Topic 2	-0.58	0.26	0.57	0.28
Topic 3	-0.51	0.25	0.68	0.20
Topic 4	-0.45	0.27	0.50	0.24
Topic 5 (deliberate outlier)	-0.36	0.25	-0.26	0.27
Topic 6	-0.15	0.25	0.40	0.25
Topic 7	-0.15	0.27	0.29	0.26
Topic 8	0.06	0.25	0.29	0.24
Topic 9	0.17	0.26	0.21	0.28

Topic Correlations

Next, we analyse the correlation between the evaluation dimensions across the topics. In the example in Table 3, we investigate if the attribute *risk* is related to the attributed *utility* for the different topics under consideration. In this example, we have only two target variables for the topic evaluations. With more variables, more complex analyses become possible: Such as determining if and to what degree a linear model with *risk* and *utility* explains the overall *valence* towards the queried topics.

Note: Our analysis focuses on the correlations between the topics as attributed by the participants, rather than individual differences among participants.

```
evaluationByTopic %>%
  dplyr::select(ends_with("_mean")) %>%
  correlation::correlation() %>%
 kable()
```

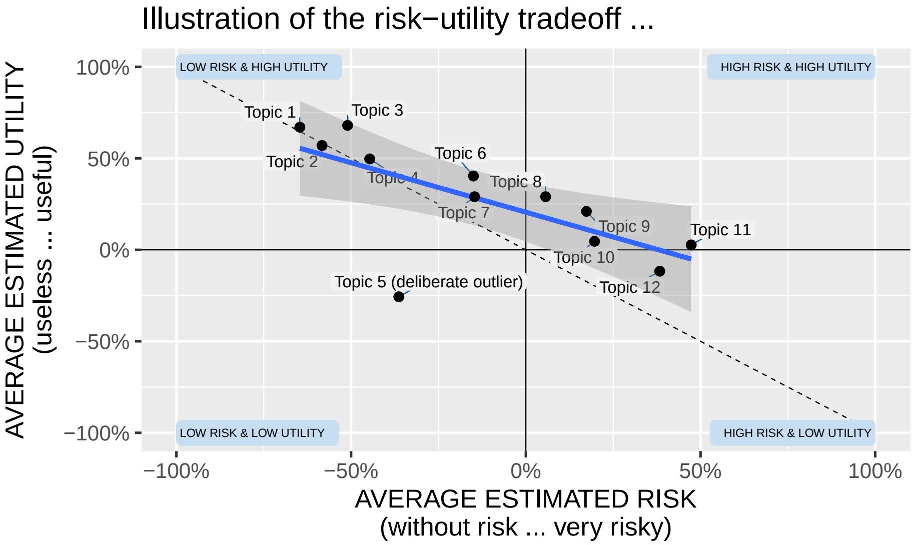
Table 3: Correlations between the evaluation dimensions across all topics

Parameter1	Parameter2	r	CI	CI_low	CI_high	t	df_error	p	Method	n_Obs
risk_mean	utility_mean	-0.6817385	0.95	-0.9025260	-0.1771451	-2.946772	10	0.0146152	Pearson correlation	12

Visualize the Topics

Finally, the results are presented through a scatter plot. The plot in Figure 5 allows for the visual identification of the dispersion of topics on a spatial map defined by the evaluation dimension. It helps assess if there is a (linear) relationship between the queried evaluation dimensions of the topics, the slope and intercept of that relationship, and if some topics exhibit significantly different evaluations compared to others (outliers).

```
scatterPlot <- evaluationByTopic %>%
  ggplot( aes( x = risk_mean,
               y = utility_mean,
               label = shortlabel)) +
  coord_cartesian(clip = "on") +
  geom_vline(xintercept = 0, size = 0.25, color="black", linetype=1) +
  geom_hline(yintercept = 0, size = 0.25, color="black", linetype=1) +
  # diagonal line indicating where both dimensions are congruent
  annotate("segment",
           x = -1, y = +1,
           xend = +1, yend = -1,
           colour = "black",
           linewidth = 0.25,
           linetype = 2) +
  # Annotate the quadrants
  geom_label(aes(x = -1, y = -1, label = "LOW RISK & LOW UTILITY"),
             vjust = "middle", hjust = "inward",
             size = 1.75,
             label.size = NA, color="black", fill = "#c7ddf2") +
  geom_label(aes(x = -1, y = +1, label = "LOW RISK & HIGH UTILITY"),
             vjust = "middle", hjust = "inward",
             size = 1.75,
             label.size = NA, color=”black”, fill = ”#c7ddf2”) +
  geom_label(aes(x = +1, y = -1, label = ”HIGH RISK & LOW UTILITY”),
             vjust = ”middle”, hjust = ”inward”,
             size = 1.75,
             label.size = NA, color=”black”, fill = ”#c7ddf2”) +
  geom_label(aes(x = +1, y = +1, label = ”HIGH RISK & HIGH UTILITY”),
             vjust = ”middle”, hjust = ”inward”,
             size = 1.75,
             label.size = NA, color=”black”, fill = ”#c7ddf2”) +
  # add the labels...
  geom_label_repel(
    max.time = 3,
    color = ”black”,
    fill = ”gray95”,
    force_pull   = 0,
    max.overlaps = Inf,
    ylim = c(-Inf, Inf),
    xlim = c(-Inf, Inf),
    segment.color =”#00549f”,
    segment.size = 0.25,
    min.segment.length = 0,
    size = 2.5,
    label.size = NA,
    label.padding = 0.105,
    box.padding = 0.125
  ) +
  geom_smooth(method = ”lm”, se = TRUE) +
  geom_point() +      # geom for the data points
  labs( title = ”Illustration of the risk-utility tradeoff ...”,
        caption = ”Based on synthetic data for illustrative purposes. See linked companion notebook under
        x = ”AVERAGE ESTIMATED RISK\n(without risk – very risky)”,
        y = ”AVERAGE ESTIMATED UTILITY\n(useless – useful)”) +
  scale_x_continuous(labels = percent_format(), limits=c( -1, +1 )) +
  scale_y_continuous(labels = percent_format(), limits=c( -1, +1 ))
scatterPlot

ggsave(”simulatedriskutility.pdf”,
       plot = scatterPlot,
       width = 8, height = 6,
       units = ”in”)
```



synthetic data for illustrative purposes. See linked companion notebook under https://osf.io/96ep5/

Figure 5: Scatter plot of the evaluations of the micro scenarios.

Closing remarks

This notebook showcases the analysis and visualization of surveys using the micro-scenario approach. It includes executable code for examining both research perspectives (individual differences and topic evaluation), which can be adjusted to suit your own survey and data. Ensure accurate coding and polarization of input variables.

It is crucial to recognize the limitations of this approach (e.g., when point estimations are acceptable, potential bias from topic sampling), and refer to the main articles for further guidance and strategies for mitigation.