



## Full length article

A z-Tree implementation of the Dynamic Experiments for Estimating Preferences [DEEP] method<sup>☆</sup>Filip Fidanoski<sup>a,\*</sup>, Timothy Johnson<sup>b</sup><sup>a</sup> Department of Economics [DSE], University of Bologna, Bologna, Italy<sup>b</sup> Business Experimental Research Laboratory [BIZLAB], UNSW Business School, University of New South Wales, Sydney, NSW, Australia

## ARTICLE INFO

## Article history:

Received 18 April 2022

Received in revised form 20 February 2023

Accepted 27 February 2023

Available online 1 March 2023

## JEL classification:

C88

D90

D81

## Keywords:

Adaptive methods

Elicitation of preferences

Risk preferences

Time preferences

z-Tree software

## ABSTRACT

Among the myriad preference elicitation methods used in experimental economics and finance, adaptive elicitation methods are a (relatively) recent innovation. Here we present a ready-made and user-friendly z-Tree application for the elicitation of risk- and time-preference parameters from the most prominent adaptive elicitation method, Dynamic Experiments for Estimating Preferences (Toubia et al., 2013). In addition to the software application, we include the code and statistical scripts for data processing when using this method that enables econometric estimation of the individual and aggregate risk- and time-preference parameters.

© 2023 Elsevier B.V. All rights reserved.

## 1. Introduction

Many decisions people make involve risk and intertemporal trade-offs. People's utility functions are thus characterized by "preferences over goods defined by a time period and a state of nature, preferences over the temporal allocation of goods, and preferences over outcomes as realizations of uncertain states of nature" (Andersen et al., 2008, p. 583).

Studying risk preferences and time preferences and their interaction can provide insights both for theory and practice. They jointly moderate people's behavior in almost all domains. Since

<sup>☆</sup> The files discussed in this article are available on GitHub at [https://github.com/FilipFidanoski/DEEP\\_z-tree](https://github.com/FilipFidanoski/DEEP_z-tree), and are licensed under the GNU GPL v3. Filip Fidanoski gratefully acknowledges funding under the UNSW Scientia Ph.D. Scholarship, Australia and the Australian Government Research Training Program Scholarship (HDR Completion Scholarship), as well as (partial) funding from the Australian Research Council [ARC] Linkage Grant LP150100608. We benefited from discussions, in alphabetic order (by first name), with Aleksa Zlojutro, Andreas Ortmann, Kiril Simeonovski, and Vinayak Dixit. Also, we want to thank Agnieszka Tymula, Aurélien Baillon, Barton Lee, Yu Yang, and other attendees at the Economics Seminar at the UNSW Business School [June 2021] and 2022 Sydney Experimental and Behavioural Research Group Seminar Series as well as the three anonymous referees. Special thanks to Nicole Sung and Elena Keller for editorial suggestions. Any errors are our own.

\* Corresponding author.

E-mail addresses: [filip.fidanoski@unibo.it](mailto:filip.fidanoski@unibo.it) (F. Fidanoski), [t.johnson@unsw.edu.au](mailto:t.johnson@unsw.edu.au) (T. Johnson).

risk and time preferences do not reveal themselves readily, it is essential to elicit risk- and time-preference parameters in a robust and comprehensive manner.

However, it is not straightforward to identify the proper elicitation mechanism [EM] for risk- and time-preference parameters. Hence, scholars have developed dozens of mechanisms/methods illustrating how to elicit risk- and time-preference parameters in experimental settings. Most of the EMs differ in design, incentive compatibility, effort levels, estimation strategies, and so on. In addition, the risk- and time-preference parameters estimated from different EMs when applying the same estimation strategies, surprisingly, and troublingly, differ significantly (Isaac and James, 2000; Loomes and Pogrebna, 2014; Crosetto and Filippin, 2016; Pedroni et al., 2017; Zhou and Hey, 2018; Holzmeister and Stefan, 2021; Friedman et al., 2022).

Maybe because of these differences, and a lack of consensus about the appropriateness of different methods, the elicitation of risk- and time-preference parameters remains a lively, interesting, and growing area of research. Previous experiments mostly ignored the family of adaptive EMs. One such method that outperforms fixed-sequence, bisection titration methods (which are based on the pre-set list of choices, and which are standard methods in psychology (Weber and Chapman, 2005; Hardisty et al., 2013)), and staircase tasks (such as the one proposed by Falk et al. (2018)) is the Dynamic Experiments for Estimating Preferences [DEEP] method (Toubia et al., 2013).

Technically, the DEEP method is designed to determine an optimal sequence of choices for a participant by leveraging heterogeneous binary choices made by them and by maximizing the Hessian matrix of the posterior distribution based on their previous choices. The method can provide information about risk- and time-preference parameters both at the individual and the aggregate level. Although the DEEP method is considered as an efficient and reliable EM (Toubia et al., 2013), Murphy and ten Brincke (2018) pointed out that it is more challenging to implement it in experiments. Perhaps the limited use of the DEEP method<sup>1</sup> in economic studies is because of the programming demands which pose a significant barrier for including the DEEP method in experimental research.

We hope that our study will contribute to popularizing the DEEP method and other adaptive EMs among experimentalists. It is of critical importance “to take some treatment that has been demonstrated using traditional methods and see if one can use these alternative [adaptive] methods to come to the same conclusion, with the same statistical reliability but more quickly and typically with smaller samples” (Harrison, 2014, p. 757). In this paper, we provide a ready-made Zurich Toolbox for Ready-made Economic Experiments [z-Tree] (Fischbacher, 2007 [1999]) application for the DEEP method. First, we briefly review the adaptive nature of the DEEP method and its structure, we then provide an incentive-compatible version of the DEEP method, before we move on to the z-Tree implementation.<sup>2</sup> We also include a Visual Basic for Application [VBA] script for data preparation and a STATA script for the baseline estimation of the risk- and time-preference parameters from the DEEP method. Finally, we summarize the findings of our study and discuss future extensions.

## 2. Adaptive EMs

There are different EMs and ways to elicit “latent” risk and time preferences in the experimental literature. The popularity and overall quality of EMs depend on several factors, such as the design, simplicity, time required to complete the method, effort level, incentive-compatibility, estimation possibilities, and external validity. There is no consensus regarding which is the best method; quite possibly efficiency and precision of an EM is contextual. Lilleholt (2019, p. 250) stated that “future studies should strive to develop new and better ways of measuring individual risk preferences. In this regard, a promising line of research is the (relatively) recent development of adaptive elicitation tasks which have been found to reduce measurement error and outperform standard elicitation procedures on a number of important parameters”. Relative to non-adaptive EMs, i.e., EMs that present participants with the same choices, adaptive EMs are individualized and present the optimal choice path. It is important to tailor individualized choice paths to the participants because “not all experimental designs are created equal” (Myung et al., 2013, p. 54).

In this section, we focus on adaptive methods for eliciting risk- and time-preference parameters that are well-known in the marketing and psychology literature but are not extensively employed in the elicitation batteries in economics and finance to this date. We illustrate the mechanism and the structure of the typical adaptive EM vis-à-vis the typical non-adaptive EM in Fig. 1a and Fig. 1b.

<sup>1</sup> The advantages and caveats of the DEEP method (e.g., the potential of conducting a field experiment with this method) are elaborated on in Section 3.

<sup>2</sup> Note that we discuss the structure of the DEEP\_RISK application, which intends to provide a mechanism for eliciting risk-preference parameter(s), but the logic and code of the DEEP\_TIME application, which intends to provide a mechanism for eliciting time-preference parameter(s), are identical. Importantly, the logic and structure of the software applications are the same. Therefore, we restrict our elaboration in the manuscript to DEEP\_RISK.

Adaptive EMs maximize the value of the information set acquired from a participant’s choices in the elicitation tasks and efficiently assess the decision-making profile of the participant in the experiment. In other words, the logic behind these tasks is that they optimize the choices and information acquisition in the experiment instantly (in real time) and thus they optimize the choice path that the participants will face in the experiment. Also, as shown in Fig. 1a, they remove the possibility of order effects and strategic behavior (Johnson et al., 2014) because the choices that the participant faces in the experiment are “personalized” and the “hand of the experimenter is invisible”. Thus, these methods outperform traditional non-adaptive EMs, illustrated in Fig. 1b, in terms of their accuracy and prediction validity, and they minimize the possibility of “cherry-picking” (Binmore and Shaked, 2007). Within this paper, we define the “cherry-picking” phenomenon as an issue that arises when the experimenter designs the choice stimuli themselves<sup>3</sup> (scaling and tailoring the choice stimuli manually). In contrast, when the adaptive methodology is implemented, the choice stimuli follow the efficient and formal (econometric) procedure. The stimuli that are designed by the experimenter harm the generalizability of the experimental design because it intends to answer particular research questions, instead of providing a broader perspective that is comparable to other studies in the field of interest. In addition, it is possible that the experimenter designs inadequate choice stimuli and presents them to the participants in the experiment. This could produce misleading conclusions, a problem that is mostly absent in the context of the adaptive elicitation methodology. Ultimately, when the elicitation of the parameters of interest is conducted in an efficient way i.e., simpler, and shorter format that is free from the experimenter demand effects, the results obtained from the rigorous estimation strategy are more reliable (Gillen et al., 2019). In other words, the adaptive EMs are of tremendous importance for making the research more credible.

Moreover, there is a pronounced practical value of these EMs because the risk-preference and time-preference parameters obtained from the adaptive EMs are comparable – which means that they are based/shown on the same metric (e.g., Constant Relative Risk Aversion [CRRA] parameter in Expected Utility Theory [EUT]) – to those parameters obtained from the non-adaptive EMs under the same empirical strategy, decision theory, and parametric specification. Statistically, in terms of the comparative performances of traditional (non-adaptive) and adaptive EMs, Chapman et al. (2018) conducted a parameter recovery study [Monte Carlo simulation] and compared traditional EMs [Multiple Price List (MPL) and Lottery Menu (Eckel and Grossman, 2002)] vis-à-vis adaptive EMs [Dynamically Optimized Sequential Experimentation {DOSE}] (Wang et al., 2010) 10 and DOSE 20]. They found that the average inaccuracy for the risk-preference part is 37% [MPL] and 35% [Lottery Menu] in the traditional EMs relative to

<sup>3</sup> The experimenter chooses the scale and context of the static EM i.e., they pre-define the set of choices that are part of small/limited choice sets that involve only several stimuli. In other words, Choice 2 is the same for all participants in the experiment in regard to the offered probabilities (or delays if the EM intends to elicit time preferences) and stakes if one implements the static EM. For example, Choice 2 in the static EM could present the following two options to participants: (1) LEFT option: a chance of 8 in 10 to win \$10 and a chance of 2 in 10 to win \$2, or (2) RIGHT option: a chance of 5 in 10 to win \$15 and a chance of 5 in 10 to lose \$2. This is pre-defined and the same for everyone as well as completely independent from the previous choice. Also, the loss context is fixed and pre-defined for the choices in the static EM, which means that it is already fixed whether Choice 2 involves losses. In contrast, the adaptive EM optimizes the sequence of choices with regard to the offered probabilities (or delays if the EM intends to elicit time preferences) and (negative/positive) stakes for each participant in the experiment, by maximizing the information advantage from the previous choice.

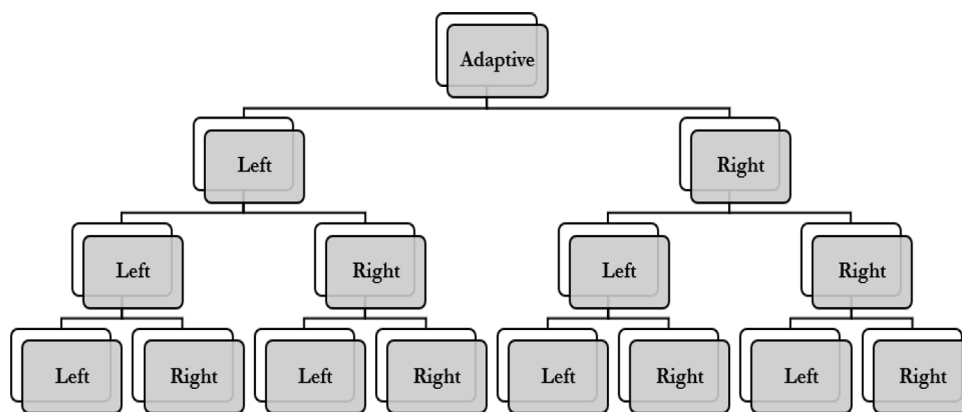


Fig. 1a. Typical adaptive EM.

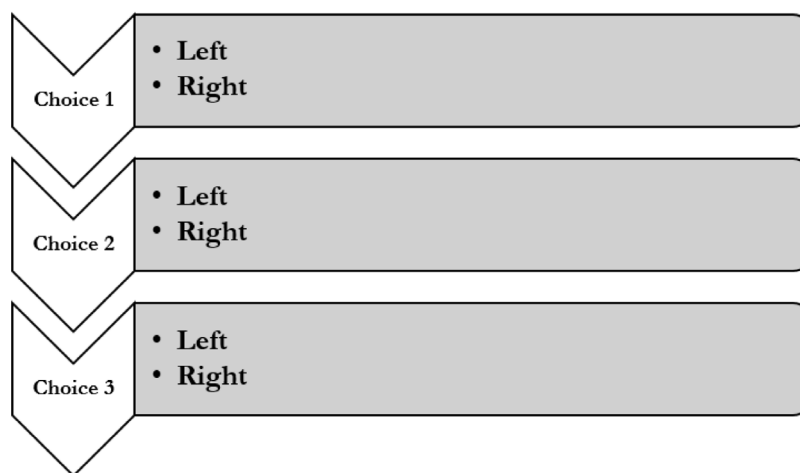


Fig. 1b. Typical non-adaptive EM.

21% [DOSE 10] and 15% [DOSE 20] in the adaptive EMs. In other words, the estimates generated by adaptive EMs can be more than twice as accurate as those estimates obtained from the traditional EMs.

Adaptive EMs are innovative, less noisy, more robust to measurement errors, provide informative responses, inflate the statistical power, and substantially improve efficiency (Cavagnaro et al., 2009). They can provide reliable information about the underlying risk- and time-preference estimates (Richter and Mata, 2018) but their importance is limited to the laboratory setting and is not adequate for natural field experiments. In a methodological paper, Daviet (2019) summarized the four representative adaptive EMs, including the one that he proposed in the study. Specifically, he summarized the main characteristics and structure elements of the following EMs: DEEP, DOSE, Adaptive Design Optimization [ADO] (Cavagnaro et al., 2010), and Sequential Optimal Inference [SOI] (Daviet, 2019). We have prepared a comprehensive table describing the estimation strategy and possibilities of these four adaptive EMs, based on the comments from Daviet (2019), and we additionally included the Bayesian Rapid Optimal Adaptive Designs [BROAD] method (Ray et al., 2013) due to the fact that there are laboratory experiments where the BROAD task is incorporated in the elicitation battery. See more details in Table 1.

Based on Table 1 [Column 7 in Table 1 follows comments of Daviet (2019)], we can conclude that adaptive EMs have specific estimation advantages over traditional EMs. Specifically, the DEEP method allows estimation in the continuous parameter space and high-dimensional models, the DOSE method optimizes the

parameter space and reduces inference obstacles, the BROAD method is computationally efficient and speedy, the ADO method is suitable for efficient discrimination among models, and the SOI method works with the general inference model. Further, the presentation format is simple and understandable for participants in the experiment; they are not lengthy and thus do not require a high effort level. Finally, they can be incentivized which is crucial because it is known that incentives play an important role in decision-making experiments (Camerer and Hogarth, 1999; Herwig and Ortmann, 2001; Holt and Laury, 2002, 2014; Harrison et al., 2005).

Considering the increasing attention, and pronounced internal (Schley and Peters, 2014; Jachimowicz et al., 2017; Atlas et al., 2017; Nofsinger and Shank, 2019; Webb and Shu, 2017; Soutschek et al., 2020; Patterson and Shank, 2020; Shank et al., 2020; Sun et al., 2021) and external validity of the DEEP method in the existing literature (Li et al., 2016, 2020), we restrict our attention exclusively to the DEEP method in the remaining parts of the paper. We further restrict our discussion to only the incentivized version of the DEEP method, as developed by us, and its comprehensive (not reduced) and optimal format [DEEP 12<sup>4</sup>]

<sup>4</sup> Section 3 contains information about how to calculate the number of potential option combinations for the different DEEP (choice) versions. For example, if one wants to implement DEEP 5 (5 choices), “only” 32 (potential) option combinations exist that could be presented to the participants in the experiment as opposed to 1024 potential option combinations in the case of implementing the DEEP 10 (10 choices). Additionally, although the format – the number of choices – depends on the research questions, we firmly believe

**Table 1**  
Selected adaptive EMs.

EMs	Design	Simplicity	Time	Effort level	Incentives	Estimation	External validity	Internal validity
DEEP	Descriptive/ transparent	Simple	3-5 min	Small	Incentive compatible	Continuous parameter estimation	Yes	Yes
DOSE	Descriptive/ transparent	Simple	5-10 min	Small	Incentive compatible	Exact optimization	-	Yes
BROAD	Descriptive/ transparent	Simple	5-10 min	Small	Incentive compatible	Equivalent Class Edge Cutting [EC <sup>2</sup> ] utilization	-	Yes
ADO	Descriptive/ transparent	Simple	5-10 min	Small	Incentive compatible	Exact optimization Model selection	-	Yes
SOI	-	-	-	-	-	Continuous parameter estimation Exact optimization Model selection General inference model	-	-

that produces a robust estimate. If required to address a particular research question, it is possible (with minimal work involved) for experimenters to modify our program and, thus, to eliminate the incentives and to reduce the number of choices. As a case in point, if external validity is the main goal of the research study, DEEP 5 is found as the optimal version (Li et al., 2020).

### 3. DEEP method

The DEEP method intends to elicit risk- and time-preference parameters by adjusting the choice path presented to each participant adaptively/dynamically. It intends to find an optimal sequence of choices that the participant is facing in the experiment by adjusting their heterogeneous choices and maximizing the Hessian matrix of posterior distributions (Toubia et al., 2013). Some of the most important features of the DEEP method are that: (1) it adaptively optimizes the sequence of choices that participants face; (2) it provides information about the distribution of parameters across individuals (heterogeneity); (3) it is possible to compare the estimates to other EMs (as they could be expressed on the same metric, such as CRRA function); (4) it models stochastic error; (5) it is computationally efficient (i.e., the method pre-calculates a table of all possible choice paths); (6) it estimates high-dimensional models and a continuous parameter space; (7) it is possible to extend the elicitation to a wide range of economic preferences; and (8) it has pronounced internal and external validity.

The precision and efficiency of the subsequent choice that the participant is facing in the DEEP method are improved by maximizing the information obtained in previous choices. Technically, as the participant progresses through the tasks, there is a computation in the background that is defining choice  $i + 1$  in the DEEP method, conditional on the response of the participant in choice

that the DEEP 12 is superior to other formats because of the larger number of (potential) option combinations. The simulation exercise of the DOSE method that is an adaptive EM (Wang et al., 2010), and the remarks that are made for the traditional EMs further confirm this. For illustration, the Random Lottery Pair Design (Hey and Orme, 1994; Harrison and Rutström, 2008) method “had been used earlier in a pilot experiment reported in Hey and Cagno (1990). In this pilot, subjects were asked just 60 questions and we discovered that 60 was fewer than we could reasonably ask within the attention span of the subjects” (Hey and Orme, 1994, p.1295). Hence, implementing more choices could potentially provide a basis for a more robust and more reliable estimation of the parameters of interest.

$i$ . Hence, for every choice, the DEEP method adaptively identifies the mode of the posterior distribution, using Newton’s method, and determines choice  $i + 1$ , following the decision in choice  $i$ , that maximizes the expected value i.e., a norm of the Hessian matrix of the posterior at its mode. In other words, the choice path is defined adaptively by updating the information regarding the value of the mode of the posterior distribution based on the decision in choice  $i$  and thereby determining choice  $i + 1$  in the DEEP method. Therefore, choice  $i + 1$  represents the choice that maximizes the expected value of the determinant of the Hessian matrix of the posterior distribution evaluated at its mode. Strictly speaking, the subsequent choice in the DEEP method is designed to reduce the asymptotic covariance matrix of the maximum likelihood estimate [MLE] of the risk- and time-preference parameters, considering the stochastic component (Toubia et al., 2013). This method is computationally efficient because it is easier to determine the mode of the posterior than it is to integrate over the posterior. Furthermore, the DEEP method is particularly useful when the posterior is bell-shaped (Daviet, 2019).

To illustrate, let us consider the first two choices in our incentivized version of the DEEP\_RISK method (designed for elicitation of latent risk parameters) in Fig. 2. Choice 1 in the DEEP method is fixed and the decision of the participant in Choice 1 determines Choice 2. Specifically, in Choice 1, the participant is asked to choose between the LEFT option that offers a 9 in 10 chance of winning \$0.05 and a 1 in 10 chance of winning \$1.25, and the RIGHT option that offers a 3 in 10 chance of winning \$5.00 and a 7 in 10 chance of losing \$1.00. There is no indifference button as in the Random Lottery Pair Design method (Hey and Orme, 1994; Harrison and Rutström, 2008) because here we are interested only in strict preference relations. (However, if a research question is targeted at eliciting weak preference relations, it is possible to add an indifference button by adding a new standard box between boxes “Choose the LEFT option” and “Choose the RIGHT option” in the software application.)

Based on the decision in Choice 1, participants are presented with different options in Choice 2. In the DEEP method, participants that have chosen the LEFT option over the RIGHT option in Choice 1, will be asked in Choice 2 to decide between the LEFT option that offers winning \$0.05 with 7 in 10 chances and losing \$0.75 with 3 in 10 chances, and the RIGHT option that offers winning \$5.00 with 3 in 10 chances or losing \$1.00 with 7 in 10 chances. In contrast, participants that have chosen the RIGHT

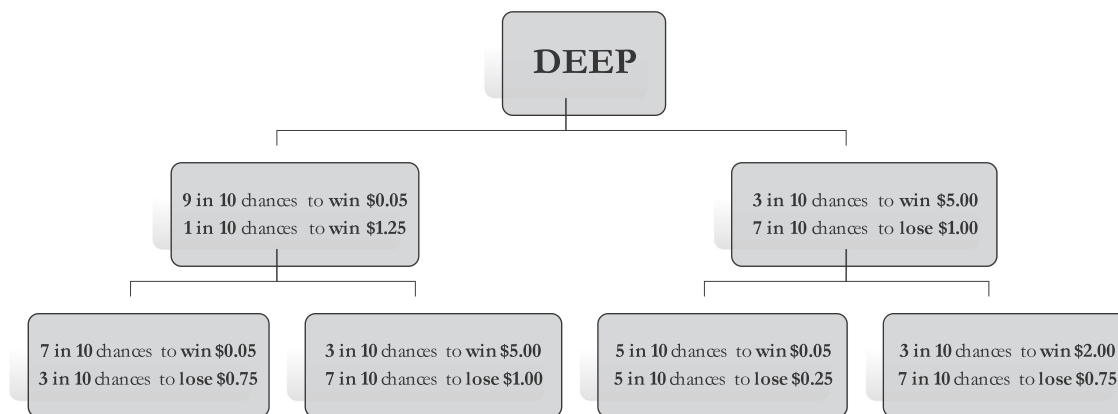


Fig. 2. Choice 1 and Choice 2 in the DEEP\_RISK method.

option over the LEFT option in Choice 1, will be asked in Choice 2 to decide between the LEFT option that offers winning \$0.05 with 5 in 10 chances and losing \$0.25 with 5 in 10 chances, and the RIGHT option that offers winning \$2.00 with 3 in 10 chances or losing \$0.75 with 7 in 10 chances. The same logic applies to subsequent choices. Note that participants are not informed about the subsequent choices in the DEEP method. Therefore, order effects, one of the most perennial issues in the elicitation literature, do not exist in the DEEP method.

The set of possible choices that can potentially appear in the choice path is 40 for the LEFT option and 36 for the RIGHT option. In Fig. 2, we can see that there are two possible options in Choice 1 and four possible options in Choice 2. By the same token, theoretically, there are 4096 possible combinations for Choice 12 on the choice path ( $2^n$ , where  $n$  denotes the number of choices offered [e.g., DEEP 5 or DEEP 12]) or 1,048,576 choice paths in total. Consequently, the number of potential option combinations in the DEEP method is impressive. It minimizes the role of experimenter demand effects and eliminates potential order effects. Given the high number of potential combinations, issues resulting from slow computation and presentation in the experiment are a valid concern. Nevertheless, speed, computation, and responsiveness of the software implementation of the DEEP method are not an issue. The choices are presented to the subjects in the experiment almost instantly because of the efficient algorithm as well as due to the software architecture.

The design of the method is simple and intuitive, and the DEEP method does not require extraordinary cognitive abilities. Also, it does not take a long time nor require an excessive effort level in the experiment. Therefore, it is even appropriate for eliciting risk- and time-preference parameters in experiments where the risk aversion and discounting behavior are not the main research focus but moderate the variable of interest.

The main limitation of the DEEP method is that it is not adequate for retrospective analysis (Sloman et al., 2020). Also, it follows a specific (estimation) algorithm as shown in Table 1, and is not adequate for natural field experiments. However, based on emerging technologies in software tools for conducting experiments, researchers can use the DEEP method in artefactual (field) experiments (Harrison and List, 2004). Also, Frydman and Jin (2022, p. 208) pointed out that the DEEP and other adaptive methods “should condition not only on the history of the participant’s choices but also on the history of the presented choice sets”. However, relative to static EMs, these disadvantages are minor and are made up for by the significant advantages of this method.

## 4. A z-Tree implementation

### 4.1. z-Tree software and additional/supporting tools

Toubia et al. (2013) provided a non-incentivized (flat/hypothetical incentives) version of the DEEP method that is supported by the software developers from the Center for Decision Sciences [CDS] at the Columbia Business School. They posted a JavaScript code on their GitHub profile as well as the estimation tools. Recently, they posted the Python code as well. The authors recommended the use of Qualtrics or Lime Survey to implement their version of the DEEP method, which can be done by importing the code that is available both on the CDS page and accompanying GitHub folder (available at the following link).

However, there is limited access to Qualtrics due to the high costs associated with obtaining a license, and the code is relatively extensive – more than 700 source lines of code – and thus it is not straightforward to implement modifications. Moreover, considering that adaptive EMs are still rare in economic papers, mostly because of the programming demands that are associated with their use, we have provided an implementation of the incentivized version of the most prominent adaptive EM, the DEEP method, in z-Tree software. We firmly believe that this will increase the interest in adaptive EMs among economists.

In short, z-Tree is arguably the most popular software for developing and carrying out economic experiments. It was originally developed at the University of Zurich, and now is supported and administrated as a joint project of the University of Zurich and the University of Konstanz. To highlight the popularity of the software, there are more than 500 citations each year in academic and non-academic studies, meaning that more than one study is published each day, on average, that used z-Tree software.

Recently, several tools have been developed as an alternative to z-Tree, such as open-source, online, and object-oriented software [oTree] (Chen et al., 2016) [1338 citations on Google Scholar to this date], classEx (Giamattei and Lambsdorff, 2019), Tokyo toolbox for large-scale combinatorial auction experiments [t-Tree] (Kazumori and Belch, 2019), and many others. Nevertheless, z-Tree is still the dominant player in the market for experimental software programs. For instance, Duch et al. (2020) showed that z-Tree [52%] is still twice as popular as its main competitor oTree [26%] and more than two times as popular as other software packages that were used for experimental research at the Cologne Laboratory for Economic Research [CLER].<sup>5</sup>

<sup>5</sup> A major limitation is that this analysis builds on the pre-COVID experience of only one particular laboratory. Therefore, it is fair to say that this result should

The software is built in a client-server fashion. That means that a server [Tree], the experimenter's machine, controls a client [Leaf], the participant's machine. The data collection is genuine and bi-directional. Namely, the server sends pieces of information related to the experiment to the client(s), and it also collects pieces of information from the client(s). It is a typical input-output architecture. The most important requirement is that both the z-Tree and z-leaf files are placed in the same folder, and they should be compatible [same version<sup>6</sup>].

There is an uncomplicated transfer of data from the experiment to the statistical packages. In addition, there are several add-ins and supporting tools that facilitate and simplify this process. We briefly review some of them. Stefan Palan's "z-tools" add-in that can be used for data preparation and organization in Microsoft Excel is one such tool. This tool can be downloaded from the personal webpage of the author. In addition, Kan Takeuchi has written and posted an ADO-file, "ztree2stata", on his personal webpage which imports a z-tree data file, converts it into STATA format, and prepares it for the analysis. (He did the same for Python language.) Motivated by the increasing popularity of the R programming language for statistical analysis (R Development Core Team, 2018), Kirchkamp (2019) prepared the R-package "zTree" which imports raw data from z-Tree directly into the R environment. For z-Tree implementation of the DEEP method, we have adjusted/pre-written a VBA code that transforms and organizes raw data from the DEEP method. The VBA code is flexible, it is easy to use in Microsoft Excel, and speeds up the analysis. The code accompanies this paper and is available for download at the following link.

The flexibility and desirability of z-Tree is further enhanced by the novel and innovative tools that have emerged in recent years. Previously, one of the main disadvantages of z-Tree was that it was limited to brick-and-mortar laboratories, and thus unavailable in an online environment. Amid the coronavirus pandemic in 2019, a novel client-integrating architecture for conducting z-Tree experiments over the internet, called "z-Tree unleashed" was developed (Duch et al., 2020). In addition, a Python script for retrieving output files from z-Tree experiments known as "TreeRing" emerged (Jiang and Li, 2019) as well as the text-editing tools known as "zBrac" (Sara and Schröter, 2019).

#### 4.2. Implementation

We have adopted some of the main principles that are relevant for conducting economic experiments: non-deceptive practices and incentivization. Thus, we will now proceed to present a z-Tree implementation of the DEEP method where participants are getting a full information set in the instructions and are paid through a random (lottery) incentive system (Grether and Plott, 1979; Cox and Sadiraj, 2019) for their effort in the DEEP method. They are informed that there are 12 choices with two options each. Importantly, they are told that, at the end of the experiment, the program will randomly play out one choice which will determine their payout in the task. Nevertheless, they know that every choice can be payment-relevant and thus they are asked to approach every choice as if it were the one that would

be accepted cautiously, because many scholars switched their experimental work to an online setup during the COVID-19 pandemic and oTree software is more suited to such online experiments. Updated analyses are required to make better judgements about the future relevance of z-Tree.

<sup>6</sup> The software application that we provide is programmed in z-Tree version 4.1.11 [21-Aug-2019]. It is compatible with the newer version of z-Tree (>5.x). For those that are interested to implement it in the older versions of z-Tree (< 4.1.0), we have exported a plain text form of the treatments which is available in the GitHub folder.

determine their earnings. We have prepared computerized instructions – that are available in the accompanying GitHub folder of this paper and that can be presented to participants<sup>7</sup> – for those who want to incorporate the maximum number of DEEP choices in their experiments. Furthermore, in order to be sure that participants understood the instructions and that their probability understanding is at an acceptable level, we have taken four comprehension questions from the original study and adjusted them to the z-Tree environment.

With regard to the core of the DEEP method, in the *background stage* of the z-Tree program, we create a *globals* program and store the prototype coded *indexers*. Each indexer represents a keycode that is later converted to an index for accessing coded choices in the DEEP method. Then, in the second *globals* program we store the details of the potential choices that are presented to the participants in the experiment on the left side ["LEFT" option in the task]. First, we create an array of 40 potential choices on the left side of the screen. Second, we write the probabilities and amounts that the participant is potentially facing in the experiment when using the DEEP\_RISK application. The potential value of amounts in the LEFT option of the DEEP\_RISK method is: -\$20, -\$15, -\$10, -\$5, \$1, \$5, \$10, \$30, \$40, \$100 and \$300. Some important differences between the DEEP\_RISK and DEEP\_TIME applications are that there are no losses in the DEEP\_TIME application, the probability is always 1 (100%), and instead of probabilities there are delays.

Similarly, we have done the same for the RIGHT option. However, the array that we create for the RIGHT option in the DEEP method consists of 36 potential choices. More precisely, the potential value of amounts in the RIGHT option of the DEEP method is: -\$20, -\$15, -\$10, -\$5, \$1, \$5, \$10, \$30, \$40, \$100, \$300, and \$400. Note that the amounts are displayed in a shuffled order conditional on the previous decisions processed through the adaptive Bayesian design as is explained in Section 3 of this paper.

Finally, in the background stage, we create another *globals* program that moderates the amounts of the options and prepares the payment details. In our version, we have created a *DR\_PAYOUT\_SCALE* that adjusts the possible amounts by a factor 0.05 [5%]. Consequently, we created modified arrays that incorporate the payout scale that we have chosen. These modified choices will be displayed to the participants and will be used towards their final payments. However, if a researcher wants to keep the possible amounts of the DEEP method unadjusted, they can simply set the *DR\_PAYOUT\_SCALE* to 1.

In the next step, we discuss the main stage of the DEEP method. First, we create 12 standard boxes and simply formulate the header of the 12 choices that will be displayed to the participants. In our case, the font of the choices is *Arial 30*, and the words are centered and bold. In the standard boxes, we have written a condition that enables Choice 2 to be displayed only after Choice 1 is made. Similarly, Choice 3 is displayed after Choice 2 is made and so on until Choice 12 in the DEEP 12 version.

The most important part of the program is to define the choice path in the main stage of the DEEP method. As explained, this program calculates the current path which is a unique identifier for the specific course of action that the participant has chosen

<sup>7</sup> For scholars that want to incorporate the software, we provide a brief guideline in the GitHub folder. The flow of actions is as follows: (1) store the .ztt file and the z-leaf file in the same folder; (2) check the compatibility of the software versions; (3) open the .ztt file and the z-leaf files; (4) select *Background* → *Number of subjects* (set the number of participants that you want); and (5) select *Run* → *Start Treatment*. In the Github folder, we provide a z-Tree software of the DEEP 12 version, but researchers can easily adapt this to their favored version by following these steps: (1) open the .ztt file; (2) click on the "DEEP\_MAIN\_RISK" stage or "DEEP\_MAIN\_TIME" stage; (3) open the "subjects.do" program; (4) change "DR\_MAX\_CHOICES = 12;" to "DR\_MAX\_CHOICES = 5;"; if you want to set it to DEEP 5; and (5) click "OK" in the program, and the treatment can be started on F5.

through the DEEP process. Each choice a participant makes represents going on a different path. It is used in calculating the next choices that will be presented to the participant and this program is contained in the *ButtonBox*. In other words, this program is the heart of the DEEP method as it directs the subsequent choices based on the decisions of the participants in the experiment.

Perhaps the most complex part of the algorithm is the choice path. We will provide some basic details about how the path is computed and how it is converted from a character value into a numeric value. The part shown in Fig. 3a computes the path based on the current number of choices made.

Depending on the number of choices made, the part of the code shown in Fig. 3b extracts the current character from the string fields defined in *globals*. Further, we convert the character values into numeric values by writing the following lines: `code1 = code(char1)` and `code2 = code(char2)`. We then have to offset the codes (Fig. 3c). Finally, we now have indexes we can plug into an array that we can use to show the probability and amount values for payout purposes, as is shown in Fig. 3d.

Additionally, each of the boxes/choices, that we create in the main stages, is displayed upon the satisfaction of the conditions/indexers that are defined. You can see an illustration of this in Fig. 4.

For illustration purposes, please see an example of (potential) Choice 5 from the DEEP method in Fig. 5, that participants could face on the computer screen. In our version, we have incentivized the DEEP method, which is in contrast to the original version of the task and other studies that have employed the DEEP method (hypothetical incentives version). We have programmed two random devices in the program. The first random device is attached to the choice and the second random device is attached to the option. In other words, the first random device is selecting the payment-relevant choice and the second device is playing out the chosen option. The payment depends partly on the decision and partly on the random device. At the end of the experiment, the participant is informed about the amount that they have earned in the DEEP method. (Also, it is possible to inform them, if important for the experiment, about the choice that the random device has picked, with a minimal work involved.)

On a related note, we have created three 7-point rate scales for the DEEP method. On these scales, participants are asked to state their understanding of the DEEP method, the effort level that they exercised, and the complexity of the DEEP method.

Progressive instructions of how to incorporate the DEEP method in experiments are included in the accompanying repository. We have also included some other important details in the DEEP program itself. Thus, it should be self-explanatory for researchers who want to incorporate the DEEP method in their experimental research. *Nota bene*, we have validated the software and tested the context consistency and the temporal stability of the DEEP method in our follow-up projects. However, it is beyond the scope of this paper, hence we do not elaborate on the results in this paper.

## 5. Data output and estimation

### 5.1. Output

When the experiment that includes the DEEP method is completed, the z-Tree software automatically creates a .csv [comma-separated values] file which contains information about the choices that were displayed by the program and the decisions that were made by the participants. For example, the decision that the participant made in the first choice is stored in `dr_choices[1]`, the decision that the participant made in the second choice is stored in `dr_choices[2]` and so on. Suppose that the participant has

chosen the RIGHT option in Choice 1. After the experiment, in the .csv file `dr_choices1` is 2. By the same token, if the participant has chosen the LEFT option in Choice 2, `dr_choices2` will be 1.

Further, the probabilities of the LEFT option in the DEEP method are recorded in the .csv file as `dr_shown_chance_A1` and `dr_shown_chance_A2`, and the probabilities of the RIGHT option are `dr_shown_chance_B1` and `dr_shown_chance_B2`. As an example, the probabilities of the LEFT option of Choice 1 are written as `dr_shown_chance_A1[1]` and `dr_shown_chance_A2[1]`, and the probabilities of the RIGHT option are `dr_shown_chance_B1[1]` and `dr_shown_chance_B2[1]`. By the same logic, the probabilities of the LEFT option of Choice 2 are written as `dr_shown_chance_A1[2]` and `dr_shown_chance_A2[2]`, and the probabilities of the RIGHT option are `dr_shown_chance_B1[2]` and `dr_shown_chance_B2[2]`. The same is valid for all twelve choices in the DEEP method.

In order to be able to estimate the parameters of interest, we need the amounts in the DEEP method that were displayed to the participant. The amounts of the LEFT option in the DEEP method are written in the .csv file as `dr_shown_amount_A1` and `dr_shown_amount_A2`, and the amounts of the RIGHT option are `dr_shown_amount_B1` and `dr_shown_amount_B2`. Hence, the amounts of the LEFT option of Choice 1 are written as `dr_shown_amount_A1[1]` and `dr_shown_amount_A2[1]`, and the amounts of the RIGHT option are `dr_shown_amount_B1[1]` and `dr_shown_amount_B2[1]`. The same is valid for all twelve choices in the DEEP 12 version of this elicitation method.

There are several possibilities for how to organize the data output from the DEEP method, as we elaborated in Section 4. Apart from the z-Tree program, we have written a VBA code that prepares the data for statistical software within a short time. The VBA code is included in the GitHub repository that accompanies this paper and it can be modified based on one's preferences. This code speeds up the analysis of the data output of the DEEP method, which is discussed in the next subsection.

### 5.2. Estimation

We have prepared several scripts that can process the data output from the DEEP method, after "cleaning" raw data from the experiment in the adjusted VBA code, in STATA environment. For example, the download folder consists of one STATA script that can be used for MLE of the DEEP method and for detecting the parameters of interest under the CRRA assumption [note that Toubia et al. (2013) provided a code for the Hierarchical Model Estimation, which is a recommended estimation strategy]. We have also provided the codes for evaluating the data under Expected Utility Theory [EUT], Separable Prospect Theory [SPT], Rank-Dependent Theory [RDT] and Mixture Model Estimation, and one can find the code for estimation (in RStudio) of the Prospect Theory parameters from the webpage of the CDS. Note that we implement these under different utility/functional specifications (Constant Relative Risk Aversion [CRRA], Constant Absolute Risk Aversion [CARA], Power, and Expo-Power) and models (Quiqqin's model, Tversky-Kahneman, and two- and three-parameters Prelec's model).

In our folder, STATA scripts are written under the cookbook procedure (Harrison and Rutström, 2008; Harrison et al., 2018): we are declaring variables in the program, we define the parametric assumption that we plan to use for the estimation and adjust the choices that involve losses. Then, we simply evaluate both options in the presented choices and we calculate the likelihood. In the STATA scripts, we have written lines for estimating the parameters of interest at both individual and aggregate levels. The decision whether researchers estimate the parameters of interest at the individual or at the aggregate level depends on their research question.

Importantly, Hey (1995) stressed that the stochastic component should be an integral part of decision-making theories. Thus,

```

// compute path -- adapted javascript algorithm
path = 0;

for (i, n)
{
  path = path + (dr_choices[i] - 1); // original algorithm is 0 and 1. we are 1 and 2, so adjust
  path = path * 2;
}

path = path + 1; // arrays in ztree start from 1, so need to add 1
dr_path[n] = path;

a

if (n == 2) { char1 = mid(db_coded_indexers_2, path, 1); char2 = mid(db_coded_indexers_2, path + 1, 1); }
elseif (n == 3) { char1 = mid(db_coded_indexers_3, path, 1); char2 = mid(db_coded_indexers_3, path + 1, 1); }
elseif (n == 4) { char1 = mid(db_coded_indexers_4, path, 1); char2 = mid(db_coded_indexers_4, path + 1, 1); }
elseif (n == 5) { char1 = mid(db_coded_indexers_5, path, 1); char2 = mid(db_coded_indexers_5, path + 1, 1); }
elseif (n == 6) { char1 = mid(db_coded_indexers_6, path, 1); char2 = mid(db_coded_indexers_6, path + 1, 1); }
elseif (n == 7) { char1 = mid(db_coded_indexers_7, path, 1); char2 = mid(db_coded_indexers_7, path + 1, 1); }
elseif (n == 8) { char1 = mid(db_coded_indexers_8, path, 1); char2 = mid(db_coded_indexers_8, path + 1, 1); }
elseif (n == 9) { char1 = mid(db_coded_indexers_9, path, 1); char2 = mid(db_coded_indexers_9, path + 1, 1); }
elseif (n == 10) { char1 = mid(db_coded_indexers_10, path, 1); char2 = mid(db_coded_indexers_10, path + 1, 1); }
elseif (n == 11) { char1 = mid(db_coded_indexers_11, path, 1); char2 = mid(db_coded_indexers_11, path + 1, 1); }
elseif (n == 12) { char1 = mid(db_coded_indexers_12, path, 1); char2 = mid(db_coded_indexers_12, path + 1, 1); }

b

// transform characters
if (code1 >= 97) { code1 = code1 - 97 + 26; }
elseif (code1 >= 65) { code1 = code1 - 65; }
elseif (code1 >= 48) { code1 = code1 - 48 + 52; }
elseif (code1 >= 42) { code1 = code1 - 42 + 62; }

if (code2 >= 97) { code2 = code2 - 97 + 26; }
elseif (code2 >= 65) { code2 = code2 - 65; }
elseif (code2 >= 48) { code2 = code2 - 48 + 52; }
elseif (code2 >= 42) { code2 = code2 - 42 + 62; }

// add one because ztree index start at 1
code1 = code1 + 1;
code2 = code2 + 1;

c

// collect payout data
dr_shown_chance_A1[n] = dr_outcomes_A_chance_1[code1];
dr_shown_amount_A1[n] = dr_outcomes_A_amount_1[code1];
dr_shown_chance_A2[n] = dr_outcomes_A_chance_2[code1];
dr_shown_amount_A2[n] = dr_outcomes_A_amount_2[code1];
dr_shown_chance_B1[n] = dr_outcomes_B_chance_1[code2];
dr_shown_amount_B1[n] = dr_outcomes_B_amount_1[code2];
dr_shown_chance_B2[n] = dr_outcomes_B_chance_2[code2];
dr_shown_amount_B2[n] = dr_outcomes_B_amount_2[code2];

d

```

Fig. 3. Computation of the choice path.

we also prepare an extension by introducing some stochastic errors which we then account for in the DEEP method. Specifically, we have prepared two versions of stochastic specifications: one known as Fechner error and the second one called Luce error. These stochastic errors indicate the sensitivities of the estimated parameters.

## 6. Discussion

We present a ready-made application for risk- and time-preference elicitation. To the best of our knowledge, this is the first attempt at programming a comprehensive adaptive elicitation method in a z-Tree environment. We believe that the



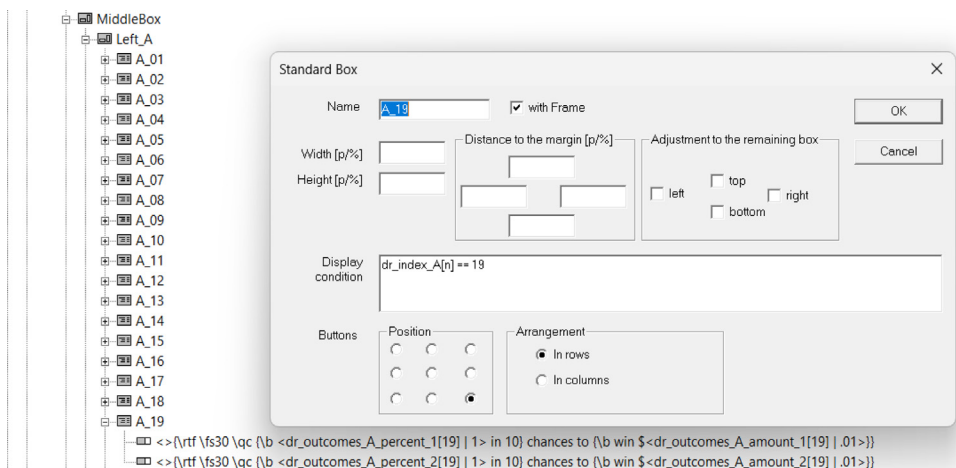


Fig. 4. Standard box of the 19th potential choice in the LEFT option.

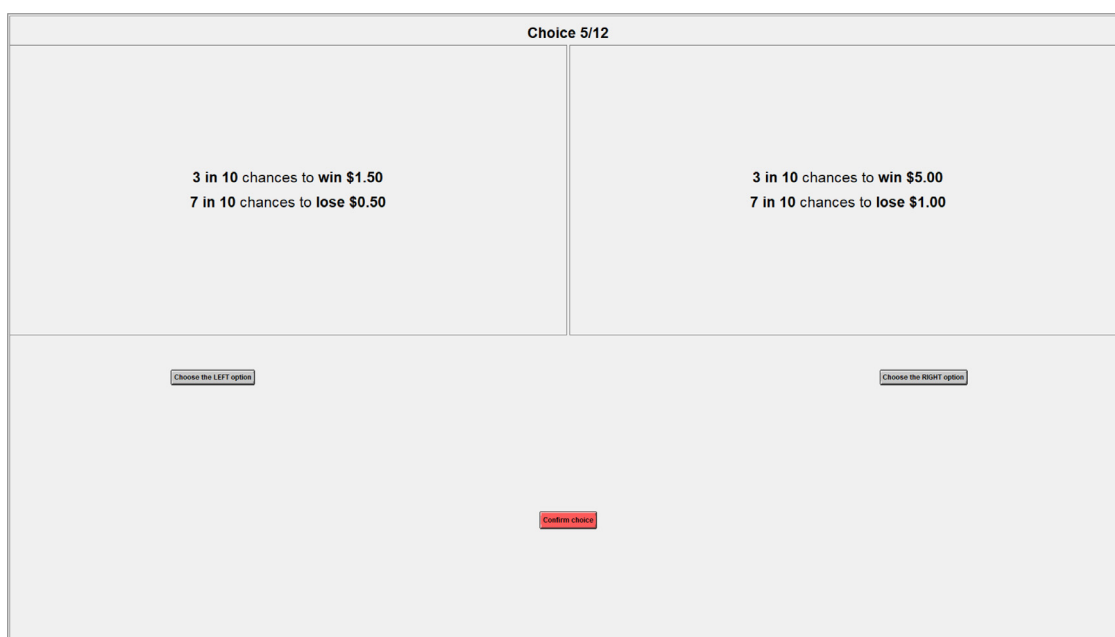


Fig. 5. Example of Choice 5 from the DEEP\_RISK method.

availability of a z-Tree application for such EMs will increase the popularity of the adaptive elicitation methodology for future economic preference elicitation.<sup>8</sup>

In addition, we have also provided some useful features, such as a VBA code for preparing data output from z-Tree and several STATA scripts for MLE of the parameters of interest when the DEEP method is applied. These files facilitate data processing and data estimation of the DEEP method.

Several extensions can be made in future work. First, one could write a STATA script to test other (different) decision theories and alternative parametric assumptions. Also, it will be interesting to elicit other economic preferences by applying the DEEP method. In terms of the context consistency, researchers could test the graphical over the descriptive presentation and compare

<sup>8</sup> Recently, Bertani et al. (2021) proposed a novel adaptive method for elicitation and estimation of the parameters of the probability weighting function. From the screenshots, it seems that they programmed the method in a z-Tree environment, but they did not include the exact reference. Nevertheless, this promising method is still not widely tested and not prominent in the literature. Moreover, z-Tree code of this method is not available.

the DEEP method to other adaptive EMs. Lastly, it is of particular relevance to test the robustness of the DEEP method by testing domain consistency and temporal stability (in a multi-period setting). Some of these issues were addressed in our follow-up projects, but there are still open research questions in this regard.

**CRedit authorship contribution statement**

**Filip Fidanoski:** Conceptualization, Methodology, Software, Validation, Writing – original draft, Writing – review & editing.  
**Timothy Johnson:** Methodology, Software, Supervision, Validation, Writing – review & editing.

**References**

Andersen, S., Harrison, G.W., Lau, M.I., Rutström, E.E., 2008. Eliciting risk and time preferences. *Econometrica* 76 (3), 583–618.  
 Atlas, S.A., Johnson, E.J., Payne, J.W., 2017. Time preferences and mortgage choice. *J. Mar. Res.* 54 (3), 415–429.  
 Bertani, N., Boukhatem, A., Diecidue, E., Perny, P., Viappiani, P., 2021. Fast and Simple Adaptive Elicitations: Experimental Test for Probability Weighting. SSRN Working Paper [3569625].

- Binmore, K., Shaked, A., 2007. Experimental economics: science or what, mimeo.
- Camerer, C.F., Hogarth, R.M., 1999. The effects of financial incentives in experiments: A review and capital-labor-production framework. *J. Risk Uncertain.* 19 (1), 7–42.
- Cavagnaro, D.R., Myung, J.I., Pitt, M.A., Kujala, J.V., 2010. Adaptive design optimization: A mutual information-based approach to model discrimination in cognitive science. *Neural Comput.* 22 (4), 887–905.
- Cavagnaro, D., Myung, J.I., Pitt, M.A., Tang, Y., 2009. Better data with fewer participants and trials: improving experiment efficiency with adaptive design optimization. In: Proceedings of the Annual Meeting of the Cognitive Science Society, Vol. 31, No. 31. pp. 93–98.
- Chapman, J., Snowberg, E., Wang, S., Camerer, C., 2018. Loss Attitudes in the US Population: Evidence from Dynamically Optimized Sequential Experimentation (DOSE). [w25072], National Bureau of Economic Research.
- Chen, D.L., Schonger, M., Wickens, C., 2016. oTree—An open-source platform for laboratory, online, and field experiments. *J. Behav. Exp. Finance* 9 (March), 88–97.
- Cox, J.C., Sadiraj, V., 2019. Incentives. In: Schram, A., Ule, A. (Eds.), *Handbook of Research Methods and Applications in Experimental Economics*. Edward Elgar Publishing, Cheltenham and Northampton, pp. 9–27.
- Crosetto, P., Filippin, A., 2016. A theoretical and experimental appraisal of four risk elicitation methods. *Exp. Econ.* 19 (3), 613–641.
- Daviet, R., 2019. Sequential Optimal Inference for Experiments with Bayesian Particle Filters. SSRN Working Paper [3320251].
- Duch, M.L., Grossmann, M.R., Lauer, T., 2020. z-Tree unleashed: A novel client-integrating architecture for conducting z-Tree experiments over the internet. *J. Behav. Exp. Finance* 28, 100400.
- Eckel, C.C., Grossman, P.J., 2002. Sex differences and statistical stereotyping in attitudes toward financial risk. *Evol. Hum. Behav.* 23 (4), 281–295.
- Falk, A., Becker, A., Dohmen, T., Enke, B., Huffman, D., Sunde, U., 2018. Global evidence on economic preferences. *Q. J. Econ.* 133 (4), 1645–1692.
- Fischbacher, U., 2007. z-Tree: Zurich toolbox for ready-made economic experiments. *Exp. Econ.* 10 (2), 171–178, [Note: working paper in 1999].
- Friedman, D., Habib, S., James, D., Williams, B., 2022. Varieties of risk preference elicitation. *Games Econom. Behav.* 133 (May), 58–76.
- Frydman, C., Jin, L.J., 2022. Efficient coding and risky choice. *Q. J. Econ.* 137 (1), 161–213.
- Giamattei, M., Lamsdorff, J.G., 2019. classEx—an online tool for lab-in-the-field experiments with smartphones. *J. Behav. Exp. Finance* 22 (June), 223–231.
- Gillen, B., Snowberg, E., Yariv, L., 2019. Experimenting with measurement error: Techniques with applications to the Caltech cohort study. *J. Polit. Econ.* 127 (4), 1826–1863.
- Grether, D.M., Plott, C.R., 1979. Economic theory of choice and the preference reversal phenomenon. *Amer. Econ. Rev.* 69 (4), 623–638.
- Hardisty, D.J., Thompson, K.F., Krantz, D.H., Weber, E.U., 2013. How to measure time preferences: An experimental comparison of three methods. *Judgm. Decis. Mak.* 8 (3), 236–249.
- Harrison, G.W., 2014. Cautionary notes on the use of field experiments to address policy issues. *Oxf. Rev. Econ. Policy* 30 (4), 753–763.
- Harrison, G.W., Hofmeyr, A., Ross, D., Swarthout, J.T., 2018. Risk preferences, time preferences, and smoking behavior. *South. Econ. J.* 85 (2), 313–348.
- Harrison, G.W., Johnson, E., McInnes, M.M., Rutström, E.E., 2005. Risk aversion and incentive effects: Comment. *Amer. Econ. Rev.* 95 (3), 897–901.
- Harrison, G.W., List, J.A., 2004. Field experiments. *J. Econ. Lit.* 42 (4), 1009–1055.
- Harrison, G.W., Rutström, E.E., 2008. Risk aversion in the laboratory. In: Cox, J.C., Harrison, G.W. (Eds.), *Risk Aversion in Experiments*. In: *Research in Experimental Economics*, vol. 12, Emerald Group Publishing Limited, Bingley, pp. 41–196.
- Hertwig, R., Ortmann, A., 2001. Experimental practices in economics: A methodological challenge for psychologists. *Behav. Brain Sci.* 24 (3), 383–403.
- Hey, J.D., 1995. Experimental investigations of errors in decision making under risk. *Eur. Econ. Rev.* 39 (3–4), 633–640.
- Hey, J.D., Cagno, D.D., 1990. Circles and triangles an experimental estimation of indifference lines in the Marschak–Machina triangle. *J. Behav. Decis. Mak.* 3 (4), 279–305.
- Hey, J.D., Orme, C., 1994. Investigating generalizations of expected utility theory using experimental data. *Econometrica* 62 (6), 1291–1326.
- Holt, C.A., Laury, S.K., 2002. Risk aversion and incentive effects. *Amer. Econ. Rev.* 92 (5), 1644–1655.
- Holt, C.A., Laury, S.K., 2014. Assessment and estimation of risk preferences. In: Machina, M.J., Viscusi, W.K. (Eds.), *Handbook of the Economics of Risk and Uncertainty*, Vol. 1. Elsevier North-Holland, Oxford, pp. 135–201.
- Holzmeister, F., Stefan, M., 2021. The risk elicitation puzzle revisited: Across-methods (in)consistency. *Exp. Econ.* 24 (2), 593–616.
- Isaac, R.M., James, D., 2000. Just who are you calling risk averse. *J. Risk Uncertain.* 20 (2), 177–187.
- Jachimowicz, J.M., Chafik, S., Munrat, S., Prabhu, J.C., Weber, E.U., 2017. Community trust reduces myopic decisions of low-income individuals. *Proc. Natl. Acad. Sci.* 114 (21), 5401–5406.
- Jiang, M., Li, J., 2019. TreeRing: A GameSafe parser for z-Tree. *J. Behav. Exp. Finance* 22 (June), 90–92.
- Johnson, C., Baillon, A., Bleichrodt, H., Li, Z., Van Dolder, D., Wakker, P.P., 2014. PRINCE: An Improved Method for Measuring Incentivized Preferences. SSRN Working Paper [2504745].
- Kazumori, E., Belch, Y., 2019. t-Tree: The Tokyo toolbox for large-scale combinatorial auction experiments. *J. Behav. Exp. Finance* 24, 100235.
- Kirchkamp, O., 2019. Importing z-Tree data into R. *J. Behav. Exp. Finance* 22 (June), 1–2.
- Li, Y., Krefeld-Schwalb, A., Wall, D., Toubia, O., Johnson, E.J., Bartels, D.M., 2020. The More You Ask, the Less You Get: When Additional Questions Hurt External Validity. SSRN Working Paper [3713044].
- Li, Y., Wall, D., Johnson, E., Toubia, O., 2016. Cognitively optimized measurement of preferences. *ACR N. Am. Adv.* 44, 532–535.
- Lilleholt, L., 2019. Cognitive ability and risk aversion: A systematic review and meta-analysis. *Judgm. Decis. Mak.* 14 (3), 234–279.
- Loomes, G., Pogrebn, G., 2014. Measuring individual risk attitudes when preferences are imprecise. *Econom. J.* 124 (576), 569–593.
- Murphy, R.O., ten Brincke, R.H., 2018. Hierarchical maximum likelihood parameter estimation for cumulative prospect theory: Improving the reliability of individual risk parameter estimates. *Manage. Sci.* 64 (1), 308–326.
- Myung, J.I., Cavagnaro, D.R., Pitt, M.A., 2013. A tutorial on adaptive design optimization. *J. Math. Psych.* 57 (3–4), 53–67.
- Nofsinger, J.R., Shank, C.A., 2019. DEEP sleep: The impact of sleep on financial risk taking. *Rev. Financ. Econ.* 37 (1), 92–105.
- Patterson, F., Shank, C.A., 2020. Health Habits and Behavioral Biases. SSRN Working Paper [3538882].
- Pedroni, A., Frey, R., Bruhin, A., Dutilh, G., Hertwig, R., Rieskamp, J., 2017. The risk elicitation puzzle. *J. Nat. Hum. Behav.* 1 (11), 803–809.
- R Development Core Team, 2018. R (Version 3.5.3). R: A Language and Environment. Foundation for Statistical Computing, Vienna, Austria.
- Ray, D., Golovin, D., Krause, A., Camerer, C., 2013. Bayesian Rapid Optimal Adaptive Design (BROAD): Method and application distinguishing models of risky choice, mimeo.
- Richter, D., Mata, R., 2018. Age differences in intertemporal choice: U-shaped associations in a probability sample of German households. *Psychol. Aging* 33 (5), 782–788.
- Saral, A.S., Schröter, A.M., 2019. zBrac—A multilanguage tool for z-Tree. *J. Behav. Exp. Finance* 23 (September), 59–63.
- Schley, D.R., Peters, E., 2014. Assessing economic value symbolic-number mappings predict risky and riskless valuations. *Psychol. Sci.* 25 (3), 753–761.
- Shank, C.A., Dupoyet, B., Durand, R., Patterson, F., 2020. The Relationship Between Psychopathy and Financial Risk and Time Preferences. In: *Studies in Economics and Finance*, vol. 38, no. 1, pp. 32–49.
- Slooman, S., Broomell, S., Kusuma, T., 2020. Diagnosing pervasive issues with parameter estimation, mimeo.
- Soutschek, A., Gvozdanic, G., Kozak, R., Duvvuri, S., de Martinis, N., Harel, B., Gray, D.L., Fehr, E., Jetter, A., Tobler, P.N., 2020. Dopaminergic D1 receptor stimulation affects effort and risk preferences. *Biol. Psychiat.* 87 (7), 678–685.
- Sun, J.J., Bellezza, S., Paharia, N., 2021. Buy less, buy luxury: Understanding and overcoming product durability neglect for sustainable consumption. *J. Marketing* 85 (3), 28–43.
- Toubia, O., Johnson, E., Evgeniou, T., Delqu  , P., 2013. Dynamic experiments for estimating preferences: An adaptive method of eliciting time and risk parameters. *Manage. Sci.* 59 (3), 613–640.
- Wang, S.W., Filiba, M., Camerer, C.F., 2010. Dynamically optimized sequential experimentation (DOSE) for estimating economic preference parameters, mimeo.
- Webb, E.C., Shu, S.B., 2017. Is broad bracketing always better? How broad decision framing leads to more optimal preferences over repeated gambles. *Judgm. Decis. Mak.* 12 (4), 382–395.
- Weber, B.J., Chapman, G.B., 2005. The combined effects of risk and time on choice: Does uncertainty eliminate the immediacy effect? Does delay eliminate the certainty effect. *Organ. Behav. Hum. Decis. Process.* 96 (2), 104–118.
- Zhou, W., Hey, J., 2018. Context matters. *Exp. Econ.* 21 (4), 723–756.