The Trust Game:
Salience, Beliefs, and Social History

by

Andre Hofmeyr, Harold Kincaid and Brian Albert Monroe⌘

September 2023

Abstract

The trust game is the standard experimental measure of trust and reciprocity in the social sciences. However, trust game experiments typically do not satisfy the salience precept, which is required to instantiate a microeconomic system in the lab. In three experiments, we find that when subjects are given all relevant information about the mapping between their actions and the earnings of both players, there is a dramatic increase in amounts returned. Beliefs about amounts returned, though, are pessimistic relative to the actual return behaviour we observe. After providing information on these returns in a graphical, easily intelligible manner, there is a marked increase in amounts sent, suggesting that the provision of social history information can be welfare enhancing for both players, and attributed to belief updating. Our results challenge the stylised facts of the trust game, and demonstrate that institutions matter.

Keywords: Experimental methodology, trust, salience, beliefs, social history.

JEL Codes: C72, C92, D83.

⌘ School of Economics, University of Cape Town, Cape Town, South Africa (Hofmeyr, Kincaid); Research Unit in Behavioural Economics and Neuroeconomics, University of Cape Town, Cape Town, South Africa (Hofmeyr, Kincaid, Monroe).

We are grateful to the University of Cape Town for funding this research; in all other respects the funder had no involvement in the research project.

We thank Tarryn Beattie for diligent research assistance, and Nat Wilcox for sharing his set of lottery pairs for our risk preference task. We also thank participants at the CEAR Africa workshop in 2019; the INEM conference in 2021; the FUR conference in 2022; the CEAR Africa workshop in 2023; the M-BEES symposium in 2023; the ESA World Meetings in 2023; and seminar participants at Durham University.

We are grateful to Glenn Harrison and Don Ross for comments.

Corresponding author: Andre Hofmeyr (andre.hofmeyr@uct.ac.za)

1. Introduction

The investment game has become the canonical behavioural measure of trust and reciprocity in economics and many cognate disciplines, and is now commonly referred to as the *trust* game. It is a (deceptively) simple strategic interaction between two agents. In its original form, both players receive the same endowment of, say, \$10, and player 1 is asked how much of this amount, if any, they want to send to player 2, knowing that the experimenter will triple the amount sent. After player 2 has received the tripled amount, they are asked how much, if any, they want to return to player 1. The amount sent by player 1 ostensibly measures trust, with larger amounts sent indicating higher levels of trust, while the amount returned by player 2 captures reciprocity.

The trust game's widespread adoption since Berg, Dickhaut and McCabe (BDM) [1995] arguably stems from its simplicity, and the fact that money is (typically) on the line, implying that experimental subjects have an incentive to reveal their preferences for trust and reciprocity. It also facilitates comparisons of trust across groups when typical survey questions (e.g., Generally speaking, would you say that most people can be trusted or that you cannot be too careful in dealing with people?)[1] may be interpreted differently by different people.

Drawing inferences from behaviour in the trust game may not be as simple as originally intended, though. For example, Cox [2004] shows that amounts sent and returned may be affected by other-regarding motives, particularly given that the amount sent by player 1 is multiplied by a factor greater than one. Other researchers have stressed the role of risk attitudes in amounts sent, given that this is an inherently risky "investment," because there is no guarantee that any of it will be returned (see Chetty et al. [2021] for a recent, systematic review and experimental results). Amounts sent will also be determined by a person's subjective beliefs about amounts conditionally returned.

[1] This question is used in the *World Values Survey*, see Haerpfer et al. [2022]

While we incorporate these considerations in our econometric analyses, our focus is more fundamental. A pillar or *precept* of experimental economics that allows one to define a microeconomic system in the laboratory is salience (Smith [1982]). Salience ultimately requires that there is a clear mapping (for the subject) between choices and expected rewards in an experiment. Our contention is that the standard implementation of the trust game, where both the first mover and second mover[2] are given the same endowment, undermines salience by obscuring the translation from actions to the monetary rewards that subjects actually earn in the experiment. This lack of salience is not innocuous, because the calculations required to determine the *terminal* payoff consequences of choices are not trivial.

We conduct a set of experiments explicitly designed to promote salience, and our results are striking. First, player 2 returns far larger amounts than typically observed in trust game experiments. Second, subjective beliefs about amounts conditionally returned track the common result that player 1 tends to break even from the decision to trust, in the sense that the amount returned is equal to the amount sent, even though this pattern is not observed in our data. Finally, when we provide "social history" information to subjects about amounts returned in our first experiment, we observe a significant increase in amounts sent, which can be explained by belief updating.

Section 2 explains why the trust game undermines salience, discusses studies that elicit subjective beliefs about player 2 behaviour, and reviews experiments using social history treatments. Section 3 describes our experimental design, which involved three different experiments with three different samples all drawn from the University of Cape Town (UCT) student population. Section 4 explains our econometric approach. Section 5 presents the results and Section 6 concludes.

[2] We use the terms "player 1 and player 2," and "first mover and second mover" interchangeably.

2. Salience, Beliefs, and Social History in the Trust Game

A. *Salience*

Do standard implementations of the trust game undermine **salience**? We realised they do when designing a set of experiments with the aim of structurally estimating a model of amounts sent in the trust game that takes into account other-regarding motives, subjective beliefs about amounts returned by player 2, and risk preferences.[3] For these experiments, we wanted to minimise the cognitive burden associated with mathematical calculations by directly performing these calculations for our subjects as they clicked through different options on their computer screens. We were motivated, in part, by Smith [1982, p. 934] who argues, "A third procedure can be directly inferred from the Siegel (1961) results, *namely to design the procedures, displays and computing aids of an experiment so as to make the experimental task as simple and transparent for the subject as is possible* [our emphasis] without, of course, compromising the essential features of the institution under study."

In a standard[4] trust game, experiment instructions make it clear that both players are endowed with the same amount of money, say E = \$10. Player 1 can send an amount $c \in$ {\$0, \$1, …, \$10} that will be multiplied by a factor greater than one, typically three, and received by player 2. The amount sent by player 1 is then communicated to player 2, who must decide how much to return of the experimenter-augmented amount: $d \in 3 \times c$. Player 1's payoff is therefore $E - c + d$, and player 2's payoff is $E + 3c - d$. Assuming no other-regarding preferences, the subgame perfect Nash equilibrium (SPNE) of the game is ($\text{Amount}_{\text{Sent}}$, $\mathbf{Amount}_{\mathbf{Returned}}$) = (c, d) = (\$0, **\$0**), because player 1 anticipates that player 2 will return zero, regardless of any amount they receive, so player 1 sends zero as the first mover.[5]

[3] We do not focus on this model here, but it is worthwhile understanding our original motivation because it helps to explain the experimental design choices we made.

[4] By "standard," we refer to the original design of BDM where both the first mover and second mover receive the same endowment. Other designs have varied the endowment of the second mover. We discuss an experiment in Section 3 (Sapienza, Toldra-Simats and Zingales [2013]) where the second mover received no endowment.

[5] Player 2's strategy that forms part of the SPNE is the ordered list $d = (0, \ldots, 0)$, which we simplify as d = (**\$0**).

When describing and analysing the game, BDM [p. 124] ignore the initial endowment E of player 2, stating that, "While subjects in room B pocket their show-up fees, subjects in room A must decide how much of their $10 to send to an anonymous counterpart in room B." However, both player 1 and player 2 in the BDM experiments received an actual participation fee of $10, which was not linked to the choices they made in the investment game. Thus, player 2's endowment was not a show-up fee, but rather a feature of the strategic interaction between the players that should be included in the analysis of the game.

While BDM's experiment instructions explain the *game form*, agents in game theory have preferences over *outcomes*, not actions, so analysis of the *game* requires that we incorporate player 2's endowment when drawing inferences about behaviour. Including player 2's endowment does not change the SPNE, but it does make the mapping from actions to rewards in the experiment more difficult to calculate, thereby undermining salience.[6] For example, if player 1 sends $6 then they are left with $4, while player 2 has (3 × $6 =) $18 + $10 (the endowment) = $28 in total (ignoring the players' participation fees of $10, which are not a feature of the game). Will subjects behave differently if these actual payoff contingencies are shown to them?

As an example, suppose player 1, the first mover, sends $8 to player 2. How much would player 2 need to return to equalise the players' earnings (inclusive of the second mover endowment)? These calculations are not trivial. The answer is that player 2 would need to return $16, because when added to the $2 left of player 1's endowment, they earn $18 in total. Player 2 received (3 × $8 =) $24, which when added to their $10 endowment gives them $34. Only by returning $16 do both players end up earning $18 from the experiment.[7] Again, by ignoring player 2's endowment, it would be easy for a subject to think that if they return the same amount sent to them by player 1, or even slightly more, then they would be repaying "trust." In reality, if

[6] Ortmann, Fitzgerald and Boeing [2000, p. 82] make a related point when re-examining the BDM results with a modified experimental design, "Confusion and/or lack of understanding of the nature of the experimental situation may have arisen from the relatively long instructions and a fairly complicated experimental design."

[7] While this mental arithmetic may be easy for some people, we expect heterogeneity in subjects' willingness and ability to do these calculations.

player 2 sends back the same amount that was sent to them, they walk away with significantly more money from the experiment.

In Section 3, we discuss the institution we designed with our experimental software to promote salience in the trust game.

B. *Beliefs*

A number of studies have investigated **beliefs** about amounts returned in the trust game. Some researchers elicit beliefs using incentivised experimental methods, while others use unincentivised survey questions asking subjects how much they expect the second mover to return.

Savage [1971] defines beliefs as subjective probabilities over possible events and reviews methods for eliciting full belief *distributions* as opposed to measures of central tendency of the distribution, such as the mean or mode.

With regard to unincentivised survey questions about expected returns, subject responses are difficult to interpret. For example, consider the common within-subject design where player 1 is asked to report "what they expect" player 2 to return to them either before or after making their amount sent choice (Ashraf, Bohnet and Piankov [2006], Buchan, Croson and Solnick [2008], Chaudhuri and Gangadharan [2007], Eckel and Wilson [2004], and Ortmann, Fitzgerald and Boeing [2000]). Does player 1 report the mean of their belief distribution over the possible amounts (or proportions) that player 2 can return, or do they report the amount they believe most likely to be returned, which is the mode of their belief distribution? These two measures of central tendency, the mean and the mode, of the subjective belief distribution are almost certainly not the same given the discrete event space in the trust game; see Section 5 where we find that the mean and mode do not coincide in our data. Some subjects may reasonably assume the question asks for the mode, while other subjects, just as reasonably, assume the question asks for the mean. Manski [2004, p. 1338] notes that subjective interpretations of qualitative *verbal* questions on expectations may vary widely across subjects. We believe unincentivised elicitation of *numeric* expectations are likewise too ambiguous for consistent interpretation.

We have identified three studies where researchers elicit beliefs about amounts (or proportions) returned using incentivised experimental methods. Costa-Gomes, Huck and Weizsäcker (CHW) [2010] use a within-subject design where subjects first decided what *share* of their endowment to transfer as player 1, and the Quadratic Scoring Rule (QSR) was then used to elicit the *mean* of their belief distribution over *proportions* returned.[8] This method is only incentive compatible for risk neutral subjects who obey subjective expected utility theory[9], and it just elicits one measure of central tendency, not the full belief distribution.

Sapienza, Toldra-Simats and Zingales (SPZ) [2013] use a within-subject design where subjects first chose an amount to send as player 1, their conditional beliefs were elicited about amounts returned for every positive amount sent, and they then chose amounts to return as player 2 using the strategy method. SPZ elicit beliefs with what we refer to as the *interval* elicitation method, because subjects are rewarded if their belief about an amount returned is within an interval of the second mover's actual decision. As SPZ [2013, Appendix B, p. 3] explain to subjects, "You earn $10 for every amount sent in which your estimation matches the responder's decision (with a 10% margin of error)." The interval elicitation method approximates the mode of each (conditional) belief distribution; it does not elicit full belief distributions.

Finally, Vyrastekova and Garikipati (VG) [2005] use a within-subject design where subjects completed the trust game assuming the roles of player 1 *and* player 2 (the strategy method was employed), and the QSR was then used to elicit beliefs about amounts sent by player 1, and conditional belief *distributions* for amounts returned by player 2.

[8] By eliciting the mean of the *proportion* returned, CHW do not focus on conditional beliefs about *amounts* returned for different amounts sent. They cannot, therefore, investigate whether beliefs about amounts returned differ as a function of amounts sent.

[9] In Section 4, we discuss our approach to the elicitation and estimation of full belief distributions for every positive amount sent. Our approach relaxes the assumptions of risk neutrality and subjective expected utility theory.

These incentivised studies all use within-subject designs: they elicit behaviour in the trust game *and* beliefs about behaviour in the trust game. As Blanco et al. [2010] explain, this design can incentivise subjects to report *beliefs* to hedge against *behaviour* in a game. In the trust game, player 1 may expect to receive a large return from player 2, but to hedge against the risk that this is not the case, they could report pessimistic beliefs about amounts returned. To mitigate the possibility of hedging, CHW and VG elicit beliefs *after* subject choices in both roles in the trust game, while SPZ do not tell subjects that their beliefs will be elicited after their amount sent choices and before their amount return choices. We are agnostic about whether subjects exploit hedging opportunities in these designs, but we used a between-subject experiment to elicit beliefs about amounts returned to remove the possibility of hedging.

SPZ argue that eliciting beliefs using proper scoring rules, such as the QSR, is complex and cognitively demanding for subjects. Their choice of the interval elicitation method was based on its simplicity. By contrast, CHW and VG use the QSR and explain how it works in their instructions. We agree that trying to explain the mechanics of the QSR to subjects is challenging, particularly because there is likely to be heterogeneity in understanding. But we do not think one needs to explain the calculations underlying the QSR if one uses an experimental design that shows subjects the earning implications of their choices; see Section 3 for more details.[10]

CHW, SPZ and VG all focus, in part, on the relationship between amounts sent and beliefs about amounts returned in the trust game. They find that amounts sent in the trust game are associated with beliefs about player 2 returns: the more a subject expects to receive in return, the larger the amount they tend to send. We enrich this literature by investigating whether beliefs about amounts returned track actual return behaviour, and apply Bayesian updating to explain how the provision of social history information affects amounts sent.

[10] As an analogy, a subject does not need to understand an internal combustion engine for the relationship between the gas pedal and the car's acceleration to be salient.

C. *Social History*

Does the provision of **social history** information influence behaviour in trust games? After collecting data from 32 *pairs* of student subjects (32 first movers, and 32 second movers) at the University of Minnesota, BDM ran a social history treatment where they presented a new set of 28 student subject pairs with the amounts sent and returned by the original 32 pairs. BDM used this treatment to investigate whether information provision bolsters trust and reciprocity, or whether it increases the prevalence of the SPNE strategy profile. BDM [p. 141] used a table to convey this information to subjects. The table listed every possible amount that could be sent ($0 - $10), the number of people choosing to send this amount, the average amount returned for each amount sent, and the average profit for each amount sent.

The table clearly shows that the only amounts sent that resulted in a positive average profit for the first mover were $5 (half of the endowment) and $10 (the full endowment). For all other amounts sent, the average profit was either zero or negative. Thus, the table suggests that amounts returned are conditional on (specific) amounts sent. However, the small sample size used to compile the table, coupled with zero, one, or two data points for six of the possible amounts sent, provides relatively limited information for subjects in the social history treatment to update their beliefs about amounts returned.

BDM found no statistically significant difference in amounts *sent* across their baseline and social history treatments: the average amount sent in the baseline treatment was $5.16, compared to the average amount sent in the social history treatment of $5.36. By contrast, BDM found that the average amount *returned* in the social history treatment ($6.46) was significantly higher than the average amount returned in their baseline treatment ($4.66). Thus, BDM's social history treatment had very little impact on average amounts sent, but a relatively large change in amounts returned, relative to their baseline treatment.

Ortmann, Fitzgerald and Boeing (OFB) [2000] sought to test the robustness of the BDM results by changing the way in which subjects were shown social history information, and by attempting to prompt strategic reasoning through a questionnaire.[11] They found that their baseline results were not significantly different to BDM. In addition, amounts sent in the OFB treatments did not differ significantly from their baseline results. OFB [p. 88] conclude that, "Our re-examination of the well-known BDM results suggest that they are quite robust. Even a presentation mode whose focus is on relative rather than absolute returns together with strategic reasoning prompts do not, counter to our expectation, manage to derail them." However, OFB's baseline results were generated by an even smaller sample than BDM: 16 pairs of student subjects in OFB compared to 32 pairs in BDM. This (very) small sample, and a lack of information on returns for all possible amounts sent[12], limit the inferences subjects could draw about amounts returned in the OFB treatments from the baseline results.

Houser, Schunk and Winter (HSW) [2010] investigated the (potential) interaction between risk preferences[13] and amounts sent in the trust game by using an experimental design where first movers were either paired with a human counterpart or a computer. In the treatments where first movers were paired with a computer, the computer determined the amount returned to first movers using the distribution of returns in BDM. In the regular (human) trust game, subjects either participated in a baseline treatment where no information on past returns was provided, or a social history treatment where they were furnished with the distribution of returns from BDM.

[11] Specifically, OFB ran a baseline treatment (Treatment 1) that replicated the BDM design. They also ran a social history treatment (Treatment 2), which displayed the results from the OFB baseline treatment using the table that BDM employed. In Treatment 3, OFB presented both the table of social history results along with a figure that showed every *tripled* amount sent and the corresponding amount returned to emphasise that amounts returned were not (except in one case) equitable. In Treatment 4, OFB provided no social history information but sought to prompt strategic reasoning by asking first movers to complete a set of survey questions prior to making their amount sent choice. These questions were designed to promote salience, and to encourage first movers to think about what they would do if they were the second mover. Finally, in Treatment 5 (and Treatment 5R, which was a replication of Treatment 5), OFB combined Treatment 3 and Treatment 4 by providing social history information in a table and a figure, and including the set of survey questions for first movers.

[12] The only positive amounts sent in the OFB baseline treatment were $2, $3, $5, $9, and $10.

[13] HSW used the multiple price list of Holt and Laury [2002] to elicit risk preferences. For a detailed discussion of the HSW design, see Chetty et al. [2021].

In an earlier working paper, HSW [2006] include the figure that subjects were given in the social history treatment. The figure shows the fraction of second movers who returned specific percentages of the tripled amount they received. An example included on the figure explains that, "About 7% of all players 2 have sent back 50% (i.e. half) of the received money." There are two issues with this figure. First, HSW used the incorrect denominator when calculating the fraction of second movers that sent back particular percentages. In BDM, two of the 32 subjects sent zero, implying that the second mover could not return anything. HSW used 32 as the denominator to produce the fraction, as opposed to 30. Second, the figure is *not conditional* on the amount sent. In other words, there is no way for a subject to determine whether 0% was only returned for low amounts sent (e.g., $1 or $2) or whether this was a universal phenomenon for all amounts sent: $1, …, $10. These issues aside, HSW [2010, p. 76] found no statistically significant differences in the distributions of amounts sent across the baseline and social history treatments.

Finally, Chetty et al. [2021] replicated the design of HSW, but with a corrected, and (slightly) more informative, version of the HSW social history figure, and a richer set of 40 lottery pairs to investigate whether attitudes to risk interact with amounts sent in the trust game. Similar to the preceding results on the effects of providing social history information, Chetty et al. [2021] found no statistically significant differences in amounts sent between the baseline and social history treatments across a number of statistical models.

In sum, the provision of social history information appears to have little effect on amounts sent in the trust game. We sought to investigate the apparent null effect of social history on amounts sent by implementing a social history institution, using a graphical, easily intelligible software interface, that provides player 1 with amount sent information from a large number of subjects.

In the next section, we discuss our experimental design, explain how it bolsters salience, allows us to elicit subjective beliefs about amounts returned in the trust game, and presents social history information to subjects.

3. Experimental Design

We conducted three experiments at UCT that were designed to provide insights into the determinants of amounts sent in the trust game. In Experiment 1, 188 subjects took part in a discretised version of the trust game assuming the roles of player 1 (the first mover) *and* player 2 (the second mover), a generalised dictator game, and a risk preference task. In Experiment 2, we elicited subjects' (*n* = 106) beliefs about amounts returned in the trust game conducted in Experiment 1, and their risk preferences. Finally, in Experiment 3, we provided social history information about amounts returned in Experiment 1, and 94 subjects completed a discretised version of the trust game assuming the roles of player 1 (the first mover) *and* player 2 (the second mover), a generalised dictator game, and a risk preference task. Our final sample consists, therefore, of data provided by 388 *distinct* subjects from the UCT student population, where no subject participated in more than one experiment. We provide details of each experiment below.

A. *Experiment 1* (*Baseline*)

Our first experiment served as the baseline for Experiment 2 and Experiment 3 in the sense that the data we collected in Experiment 1 was used in these subsequent experiments. In Experiment 1, subjects took part in a discretised version of the trust game assuming the roles of player 1 (the first mover) *and* player 2 (the second mover), a generalised dictator game, and a risk preference task.

In the trust game, player 1 had to make *five* decisions to send $c \in$ {R0, R20, R40, R60, R80, R100}[14] to player 2, knowing that the amount sent would be tripled.[15] As player 2, subjects had to make return decisions for every possible amount sent, so we used the strategy method to elicit player 2's full strategy; this is an important design choice for the elicitation of beliefs in Experiment 2. In the generalised dictator game, each subject had to make five decisions to send $h \in$ {R0, R10, R20, …, R80, R90, R100} to another, anonymous subject, where player endowments and the multiplier differed across the five decisions, e.g., for one decision, both player 1 (the dictator) and player 2 (the receiver) had the same endowment and the multiplier was 3, thereby mimicking the trust game, as per Cox [2004]. Finally, in the risk preference task, subjects made 100 binary choices between the lottery pairs designed by Wilcox [2018].[16]

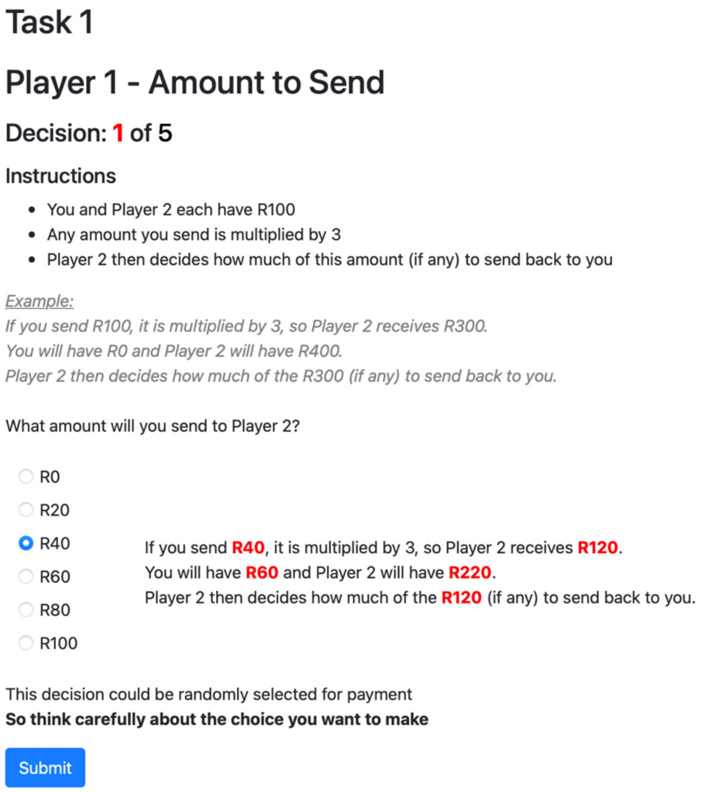
Subjects received written instructions explaining each task before they completed it.[17] When a subject finished reading the instructions for a task, they raised their hand so that a research assistant could play complementary audio-visual instructions on the computer that showed them the decision-making environment in greater detail. Subjects wore on-ear headphones provided by the experimenters, so that they did not disturb other subjects when watching the video instructions.

[14] In 2018, \$1 ≈ R6.40 at purchasing power parity (PPP), implying the stakes in our experiment were slightly higher than BDM. Specifically, the largest amount player 1 could send was approximately \$16. We limited amounts sent to increments of R20, because we used the strategy method to elicit player 2's responses to every possible amount sent.

[15] Subjects were asked to make five decisions given the discretised nature of our trust game. For example, suppose a subject in the role of player 1 wants to send half of their endowment. This is not possible if they make one choice, but by making five choices and oscillating between R40 and R60, their average amount sent would be close to R50. As we discuss later, our subjects did indeed oscillate between amounts sent.

[16] The generalised dictator game and risk preference task are not our focus here, except in so far as we use them to include covariates in our econometric analyses of amounts sent (dictator and risk) and amounts returned (dictator only) in the trust game. In addition, the risk preference data are used to *recover* subjective beliefs to account for our subjects' high levels of risk aversion, which, as a matter of theory, will affect their token allocations in the beliefs task; see the discussion in Section 4.

[17] The written instructions for all tasks are available in Appendix A. The audio-visual instructions can be accessed at: https://osf.io/ypuzd/.



Task 1

Player 1 - Amount to Send

Decision: 1 of 5

Instructions

- You and Player 2 each have R100
- Any amount you send is multiplied by 3
- Player 2 then decides how much of this amount (if any) to send back to you

Example:
If you send R100, it is multiplied by 3, so Player 2 receives R300.
You will have R0 and Player 2 will have R400.
Player 2 then decides how much of the R300 (if any) to send back to you.

What amount will you send to Player 2?

○ R0
○ R20
● R40
○ R60
○ R80
○ R100

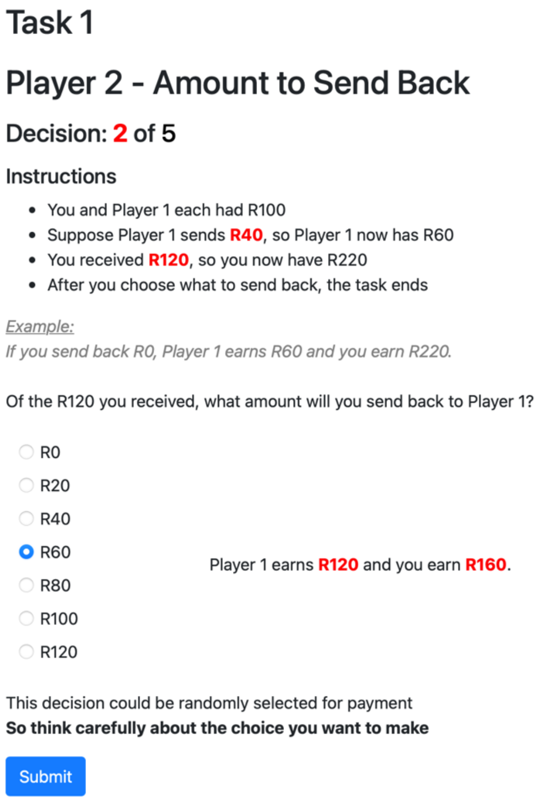
If you send **R40**, it is multiplied by 3, so Player 2 receives **R120**.
You will have **R60** and Player 2 will have **R220**.
Player 2 then decides how much of the **R120** (if any) to send back to you.

This decision could be randomly selected for payment
So think carefully about the choice you want to make

Figure 1: Interface for Player 1 in the Trust Game

We varied the order of the roles in the Trust game, and the order of the Dictator game and Trust game between subjects, while always eliciting Risk preferences at the end of the session. Four task orders were, therefore, used: [T_S, T_R, D, R], [T_R, T_S, D, R], [D, T_S, T_R, R], and [D, T_R, T_S, R], where the subscript S refers to "send" and the subscript R refers to "return."

To promote salience, subjects were shown the payoffs from each possible amount they could choose to send or return in the trust game. Figure 1 shows the interface we designed for player 1 using *oTree*, developed by Chen, Schonger and Wickens [2016]. As subjects clicked on each possible amount to send, they were shown the payoff consequences of this alternative for player 1 *and* player 2. We randomly selected an Example for each subject on each screen so that any potential priming would wash out in the aggregate.



Task 1

Player 2 - Amount to Send Back

Decision: 2 of 5

Instructions

- You and Player 1 each had R100
- Suppose Player 1 sends **R40**, so Player 1 now has R60
- You received **R120**, so you now have R220
- After you choose what to send back, the task ends

Example:
If you send back R0, Player 1 earns R60 and you earn R220.

Of the R120 you received, what amount will you send back to Player 1?

○ R0
○ R20
○ R40
● R60 Player 1 earns **R120** and you earn **R160**.
○ R80
○ R100
○ R120

This decision could be randomly selected for payment
So think carefully about the choice you want to make

Figure 2: Interface for Player 2 in the Trust Game

Figure 2 shows the interface we designed for player 2, where the Example on each screen was randomly selected for each subject. When subjects clicked on each possible amount to send back[18], they were shown the payoff consequences of their choice for player 1 *and* player 2. These amounts incorporated player 2's endowment, as they should. This design directly maps actions to rewards in the experiment, thereby mitigating concerns about whether the salience precept was satisfied. McCabe and Smith [2000] also clearly point out the payoff consequences of player 1 and player 2's choices in the trust game, inclusive of player 2's endowment, but in a far more restrictive environment: a binary trust game where player 1 sends all ($10) or nothing, and player 2, conditional on receiving $30, returns $15 or nothing.[19]

[18] We specifically avoided the word "return," because we did not want to prime subjects to reciprocate.

[19] McCabe and Smith [2000] prevent player 2 from equalising the players' earnings by returning $20, because they want to rule out concerns for equity or fairness in their analysis.

Payment Protocol

We were sensitive to the possibility of cross-role and cross-task contamination in the experiment.[20] For example, if subjects are paid for both roles in the trust game they can hedge against risk by sending a large amount as player 1 and returning nothing as player 2. In addition, if subjects are paid for both tasks, then sending a large amount in the dictator game may crowd out other-regarding motives in the trust game. Consequently, we only randomly paired subjects at the end of the experiment and paid them for *one* role from *one* of the trust game and dictator game tasks. Subjects received a participation fee of R40, and were also paid for one randomly selected choice from the risk preference task, earning R410 ≈ $65 (at PPP), on average.

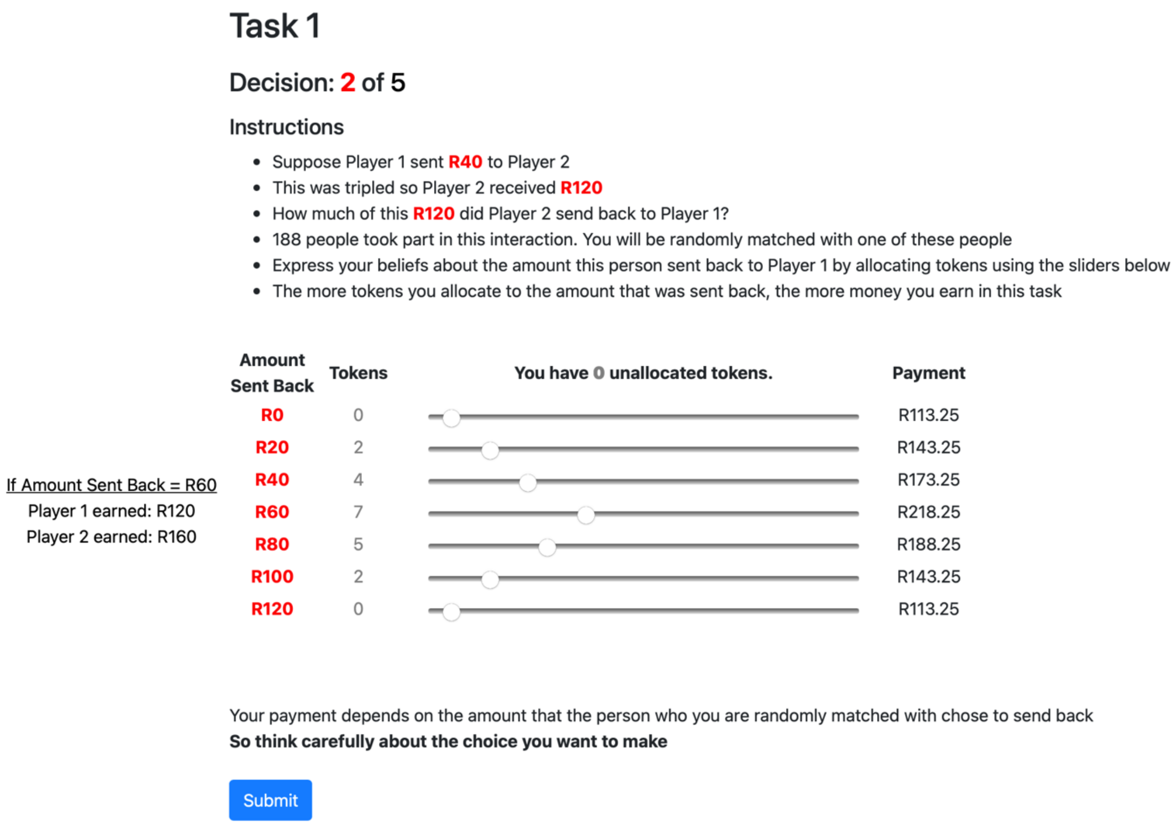
B. *Experiment 2 (Beliefs)*

In Experiment 2 we elicited subjects' risk preferences and subjective beliefs about amounts returned by the 188 participants in Experiment 1 for the five positive amounts $c^* \in$ {R20, R40, R60, R80, R100} that player 1 could send. Experiment 2 provided the rationale for using a discretised version of the trust game, along with the strategy method for player 2, in Experiment 1. For this experiment, we varied the order of the subjective beliefs task and risk preference task across subjects.

For the beliefs task, subjects were given written and audio-visual instructions explaining the trust game that was run in Experiment 1, which included screenshots of player 2's interface so that it was clear that player 2 knew the payoff consequences of their returns. In the task itself, subjects had to allocate 20 tokens to bins, representing the possible amounts returned for each amount sent, to express their beliefs about the likelihood that a particular amount was returned

[20] See Cox, Sadiraj and Schmidt [2014, 2015], Azrieli, Chambers and Healy [2018, 2020], and Brown and Healy [2018] for detailed discussions.

for that amount sent.[21] Subjects could allocate their tokens in any way, e.g., bimodally, and were incentivised using the QSR.



Task 1

Decision: 2 of 5

Instructions

- Suppose Player 1 sent **R40** to Player 2
- This was tripled so Player 2 received **R120**
- How much of this **R120** did Player 2 send back to Player 1?
- 188 people took part in this interaction. You will be randomly matched with one of these people
- Express your beliefs about the amount this person sent back to Player 1 by allocating tokens using the sliders below
- The more tokens you allocate to the amount that was sent back, the more money you earn in this task

If Amount Sent Back = R60
Player 1 earned: R120
Player 2 earned: R160

Amount Sent Back	Tokens	You have 0 unallocated tokens.	Payment
R0	0		R113.25
R20	2		R143.25
R40	4		R173.25
R60	7		R218.25
R80	5		R188.25
R100	2		R143.25
R120	0		R113.25

Your payment depends on the amount that the person who you are randomly matched with chose to send back
So think carefully about the choice you want to make

Figure 3: Interface for Subjective Beliefs Task

Figure 3 shows a screenshot of the subjective beliefs task for an amount sent of R40. When subjects hovered over a bin representing an Amount Sent Back, e.g., R60, a pop-up appeared showing what player 1 and player 2 would have earned from this particular return, e.g., (R120, R160). Our intention, again, was to make the experiment as simple and transparent as possible to enhance salience. Subjects had to report their beliefs about amounts returned for the five positive amounts that could be sent in our trust game. After completing the subjective beliefs task, subjects received written and audio-visual instructions for the risk preference task, which was identical to the task used in Experiment 1, and then made 100 lottery choices.

[21] We have used 100 tokens in other experimental designs, e.g., Harrison et al. [2021, 2022], but chose to limit the number of tokens to 20, so that moving a token from one bin to another had relatively large payoff consequences for subjects. This choice was informed by our desire to avoid payoff dominance concerns (Harrison [1989, 1992, 1994]).

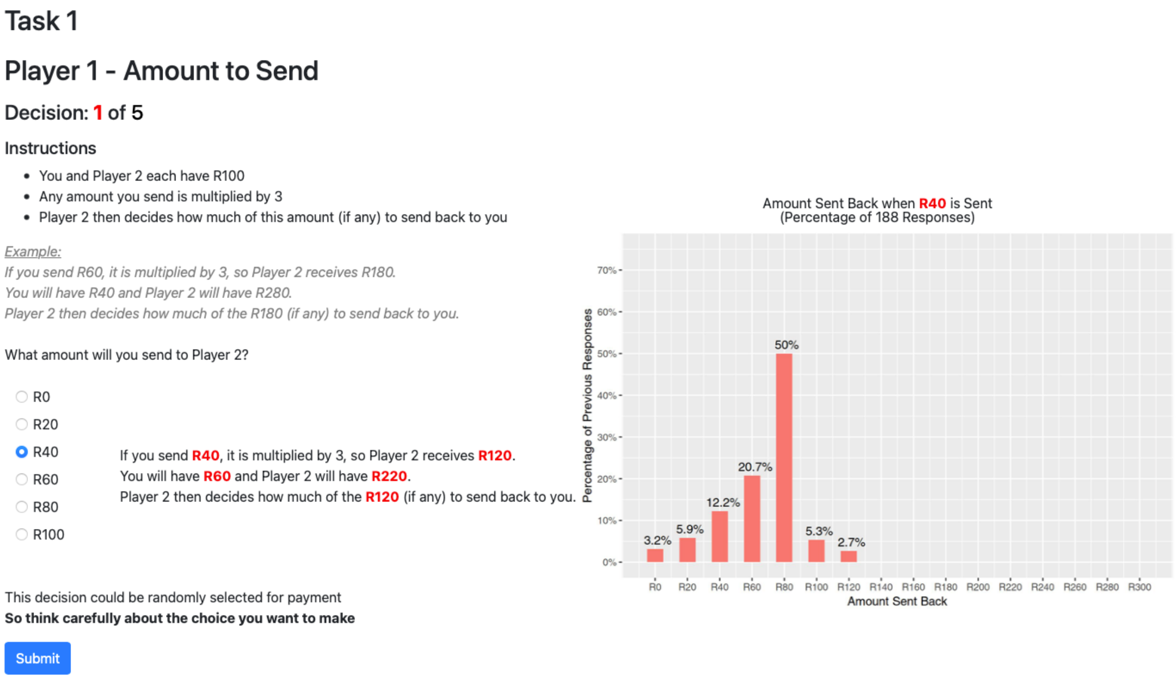
Payment Protocol

After subjects had completed both tasks, they were paid for the subjective beliefs task in the following way: each subject was randomly paired with the data from one of the 188 participants in Experiment 1; one of the subject's five belief questions (for amounts sent of R20, R40, R60, R80, and R100) was randomly selected for payment; and they were paid out on the basis of their token allocation to the bin containing the participant's choice from Experiment 1 with whom they were paired. Using the example in Figure 3, suppose this (hypothetical) subject was paired with a participant in Experiment 1 who decided to return R20 after being sent R40. Given that the subject in Experiment 2 allocated 2 tokens to the Amount Sent Back bin of R20, they would have been paid R143.25.

Subjects received a participation fee of R40 and were paid for one randomly selected risk preference choice, earning R480 ≈ $75 (at PPP), on average.

C. *Experiment 3* (*Social History*)

In Experiment 3 we provided social history information about return behaviour in Experiment 1, and subjects (n = 94) completed the discretised version of the trust game from Experiment 1, assuming the roles of player 1 *and* player 2, the generalised dictator game, and the risk preference task. In the role of player 1, as subjects clicked on each possible amount to send they were shown how this alternative affected the income distribution of player 1 *and* player 2 at this stage of the game. Crucially, they were also shown Amount Sent Bank histograms, using the data from Experiment 1, when they clicked on each alternative. Figure 4 shows the interface we designed for our social history treatment. The histograms automatically updated when subjects clicked on each possible amount, but using the same set of axes to (hopefully) aid comprehension, and to emphasise that amounts returned are necessarily conditional on amounts sent. To avoid priming subjects with social history information prior to making their return decisions, they *always* assumed the role of player 2 first in this experiment, implying we used two task orders: [T_R, T_S, D, R], and [D, T_R, T_S, R].[22]



Task 1

Player 1 - Amount to Send

Decision: 1 of 5

Instructions

- You and Player 2 each have R100
- Any amount you send is multiplied by 3
- Player 2 then decides how much of this amount (if any) to send back to you

Example:
If you send R60, it is multiplied by 3, so Player 2 receives R180.
You will have R40 and Player 2 will have R280.
Player 2 then decides how much of the R180 (if any) to send back to you.

What amount will you send to Player 2?

○ R0
○ R20
● R40
○ R60
○ R80
○ R100

If you send **R40**, it is multiplied by 3, so Player 2 receives **R120**.
You will have **R60** and Player 2 will have **R220**.
Player 2 then decides how much of the **R120** (if any) to send back to you.

This decision could be randomly selected for payment
So think carefully about the choice you want to make

Figure 4: Interface for Player 1 in the Social History Treatment

Payment Protocol

The payment protocol for Experiment 3 was identical to Experiment 1. Thus, subjects were randomly paired at the end of the experiment and paid for *one* role from *one* of the trust game and dictator game tasks. Subjects received a participation fee of R40, and were also paid for one randomly selected choice from the risk preference task, earning R425 ≈ $67 (at PPP), on average.

D. *Discussion*

To the best of our knowledge, the experimental design of STZ is the closest to ours in the following ways: 1) subjects assumed the roles of the first mover *and* second mover; 2) they elicited beliefs about amounts returned; 3) subjects were paid for *one* role in the trust game *or* the

[22] In subsequent discussions, we omit the "R" for brevity when referring to different task orders, because the risk preference task was completed at the end of the experimental session.

beliefs task; and 4) the software allowed subjects to calculate the payoffs of both players. It differs in some fundamental ways, though. First, subjects in the role of player 2 had to click a "Calculate" button to see the earnings of both players, whereas we presented this information automatically when subjects clicked on an amount to return. Second, STZ did not endow player 2; in principle this should make it easier for subjects to determine the payoffs to both players for a particular amount returned, in comparison to the BDM design. Third, we used a between-subject design to elicit beliefs as opposed to the within-subject design of STZ. Fourth, they elicited beliefs using the interval elicitation method, which approximates the mode of the belief distribution, in comparison to our approach that elicits the full distribution. Fifth, subjects were not shown the payoffs of both players during the beliefs task for each possible amount returned. Finally, STZ used a risk preference task where subjects had to make 15 choices between a two-outcome lottery and a certain reward, whereas we presented subjects with 100 lottery choices with up to three prizes per lottery.

In sum, our experimental design arguably elicits richer information about preferences and beliefs than STZ, while making the payoffs from various choices salient to subjects.

4. Econometrics

The amounts sent in our trust game generate discrete, bounded, ordered dependent variables given that R0 < R20 < R40, etc. Thus, we analyse amounts sent using ordered logit regression models. Following Long and Freese [2014, ch. 8] we also test the robustness of our results by estimating multinomial logit models of amounts sent, because the multinomial logit does not incorporate the parallel regression assumption of the ordered logit. For amounts returned, we pool the data for each subject and each possible amount sent, and estimate a fractional response (logistic) regression model of the *proportion* of the amount returned. Given that subjects made five amount sent choices and five amount return choices in our trust game, we cluster the standard errors of the estimates by subject identifier to allow for heteroscedasticity across subjects.

Our econometric approach to the estimation of subjective beliefs about amounts returned in the trust game is relatively complex and requires more explanation. To fix ideas first, recall that subjective beliefs are defined as subjective probabilities over possible events. Thus, belief distributions obey the same rules as probability distributions: probabilities are non-negative and sum (or integrate) to 1. As Savage [1971] recognises, subjective beliefs and risk preferences jointly rationalise observed choices in subjectively risky environments.

Winkler [1969] and Matheson and Winkler [1976] discuss methods for eliciting subjective beliefs by paying subjects for their reported probabilities according to a "scoring rule." They proposed several scoring rules, but only a proper scoring rule, such as the QSR, is incentive compatible for risk neutral agents that obey Subjective Expected Utility (SEU) theory. If these assumptions hold, beliefs can be inferred directly from subjective probability *reports*, viz., token allocations in our setting. There is an extensive literature, though, which suggests that experimental subjects exhibit risk aversion, even over lotteries with small stakes.[23] If subjects are not risk neutral, proper scoring rules do not incentivise subjects to directly reveal their beliefs.

For example, if a subject is (sufficiently) risk averse then they will hedge in a subjective beliefs task by allocating tokens to a number of bins, even though they may not be particularly *confident*[24] that the answer to the belief question falls into these bins. For example, the belief distribution of a risk averse subject would be more highly peaked than the distribution of tokens in Figure 3. In addition, Rank-Dependent Utility (RDU) agents with pronounced probability weighting can allocate *more* tokens to a bin they consider *less* likely to contain the true answer than to the bin they consider most likely to contain the true answer. In Figure 3, a RDU agent

[23] Savage [1971, p. 786] argues, "Within sufficiently narrow limits, any person's utilities can be expected to be practically linear." The experimental evidence, reviewed by Harrison and Rutström [2008], has *not* provided support for this conjecture.

[24] Harrison et al. [2021, 2022] explain the distinction between *bias* and *confidence* of subjective belief distributions.

may believe the answer is more likely to fall into the R80 bin, but nevertheless allocate more tokens to the R60 bin than the R80 bin.[25] Researchers have therefore sought to relax the assumptions of risk neutrality and conformance with SEU.

For example, Offerman et al. [2009] develop a method to correct subjective probability reports to account for risk aversion. However, their approach can produce "corrected" subjective probability reports that sum to greater than 1, and therefore violate a rule of probability distributions. Harrison et al. [2017] also relax the assumption of risk neutrality, while still assuming SEU, and derive subjective probability reports, viz., beliefs, that do indeed sum to 1. Harrison, Monroe and Ulm (HMU) [2022] further relax the requirement of SEU by allowing risk preferences to be consistent with RDU (and probability weighting functions that are invertible, continuous, and differentiable).[26]

We use the theory in HMU, which links RDU risk preferences with the reports elicited using the QSR, to *recover* subjective beliefs from subject token allocations. Adopting this approach we first specify the likelihood function to estimate risk preferences, use the candidate risk preference estimates to recover subjective beliefs, and then fit a probability distribution (in our case, a probability mass function) to the recovered beliefs. We provide a brief explanation of our econometric approach below, and include the details in Appendix B.

The RDU model, due to Quiggin [1982], nests Expected Utility Theory (EUT) when agents do not subjectively distort objective probabilities, so we use the RDU model for the sake of generality. We form an index of the RDU difference between the two lotteries in a pair, by

[25] For example, a RDU agent with a strictly convex probability weighting function will assign higher decision weights to worse prizes and lower decision weights to better prizes, relative to the prizes' objective probabilities. They may, therefore, allocate more tokens to an outcome they consider less likely, in comparison to an outcome they consider more likely.

[26] Another approach is to risk neutralise subjects by using the experimental payment mechanism referred to as the binary lottery procedure (BLP) developed by Smith [1961]. With this approach, experimenters elicit beliefs using, say, the QSR but pay subjects using the BLP. Allen [1987] appears to be the first statement of this approach, and McKelvey and Page [1990] apply this method experimentally. Hossain and Okui [2013] and Schlag and van der Weel [2013] investigate extensions of this approach and refer to them as the "binarized scoring rule," and "randomized QSR," respectively. Harrison, Martínez-Correa, Swarthout and Ulm [2015] also used the QSR together with the BLP to elicit subjective belief *distributions*, as opposed to a moment of that distribution.

assuming a constant relative risk aversion (CRRA) utility function, and the Prelec [1998] two-parameter probability weighting function (PWF). We also adopt the Contextual Utility (CU) behavioural error specification of Wilcox [2011]. The CU model normalises the utility difference to lie within the interval [-1, 1] and includes a Fechner [1966/1860] error term to allow choices in the risk preference task to deviate from the deterministic predictions of RDU. The RDU difference index is passed through the logistic cumulative distribution function to determine the likelihood of a particular choice in each lottery pair. The resulting log-likelihood function can be maximised to estimate the CRRA parameter, the parameters of the PWF, and the Fechner error term.

Now consider our subjective beliefs task where subjects had to allocate 20 tokens to n bins to express the likelihoods they assigned to particular amounts returned, conditional on an amount sent, in the trust game in Experiment 1. Each question in the task elicits beliefs about the amount returned for each positive amount sent c. Thus, the amount sent determines the number of bins n for a particular subjective beliefs question, where $n \in \{4, 7, 10, 13, 16\}$. For example, suppose c = R40 (see Figure 3). This implies $n = 7$, because the amount returned $d \in$ {R0, R20, R40, R60, R80, R100, R120}. Thus, we elicited beliefs about amounts returned for the five positive amounts sent $c^* \in$ {R20, R40, R60, R80, R100}.

For simplicity, focus on the case were c = R20, implying $n = 4$, and $d \in$ {R0, R20, R40, R60}. A subject must allocate all 20 tokens across the 4 bins, but there are no constraints in terms of the allocation, e.g., a subject can allocate all of their tokens to one bin. We summarise a token allocation by dividing the allocation to each bin by 20 (the number of tokens), and refer to this as the subject's observed *report*.

We used the QSR to reward subjects for their reports. Combining the QSR with the RDU model allows us to calculate the utility of a subject's report for a particular amount sent (R20 in our example). Again, we use the CRRA utility function and Prelec [2008] PWF to determine the utility of the observed report. We employ the results in HMU to map the subject's observed report to their *beliefs*, which are conditional on the subject's risk preferences. Weighted maximum likelihood estimation is then used to fit a probability mass function to "recover" these beliefs. By combining the log-likelihood for risk preferences and the (weighted) log-likelihood for beliefs, we estimate subjective beliefs, adjusted for risk preferences, about amounts returned in Experiment 1.

5. Results

Johnson and Mislin (JM) [2011] conducted a meta-analysis of trust game experiments, involving more than 23,000 subjects, by pooling the results from 162 replications of the game up until 2011. They found that the average amount sent, as a proportion of the first mover's endowment, was 50%, whereas the average amount returned, as a proportion of the amount received by the second mover, was 37%. Thus, the stylised "facts" of the trust game are that first movers send about half of their endowment, and trust just pays off, in the sense that first movers tend to break even after receiving approximately the same amount that they sent.[27]

A. *Experiment 1* (*Baseline*)

Figure 5 shows the distribution of amounts sent in the trust game by the 188 subjects in the Baseline Treatment. Recall that each subject made *five* amount sent choices in the task, so the figure is based on 940 data points. The three most prevalent amounts sent are R20 (20%), R40 (24%), and R60 (18%). The average amount sent, pooling across all subjects and choices, is R48, with a standard deviation of R31, which is very close to the result identified by JM.

[27] Of the 162 trust game replications, 91% used a multiplier of 3, as per BDM, while the remaining studies used a multiplier of 2. JM find that a multiplier of 3, as opposed to 2, is associated with a statistically significant *decrease* in the proportion of the amount returned by second movers. On the other hand, the proportion of the amount returned is positively and significantly related to the amount sent by first movers. Thus, there is nuance to the stylised results discussed above.

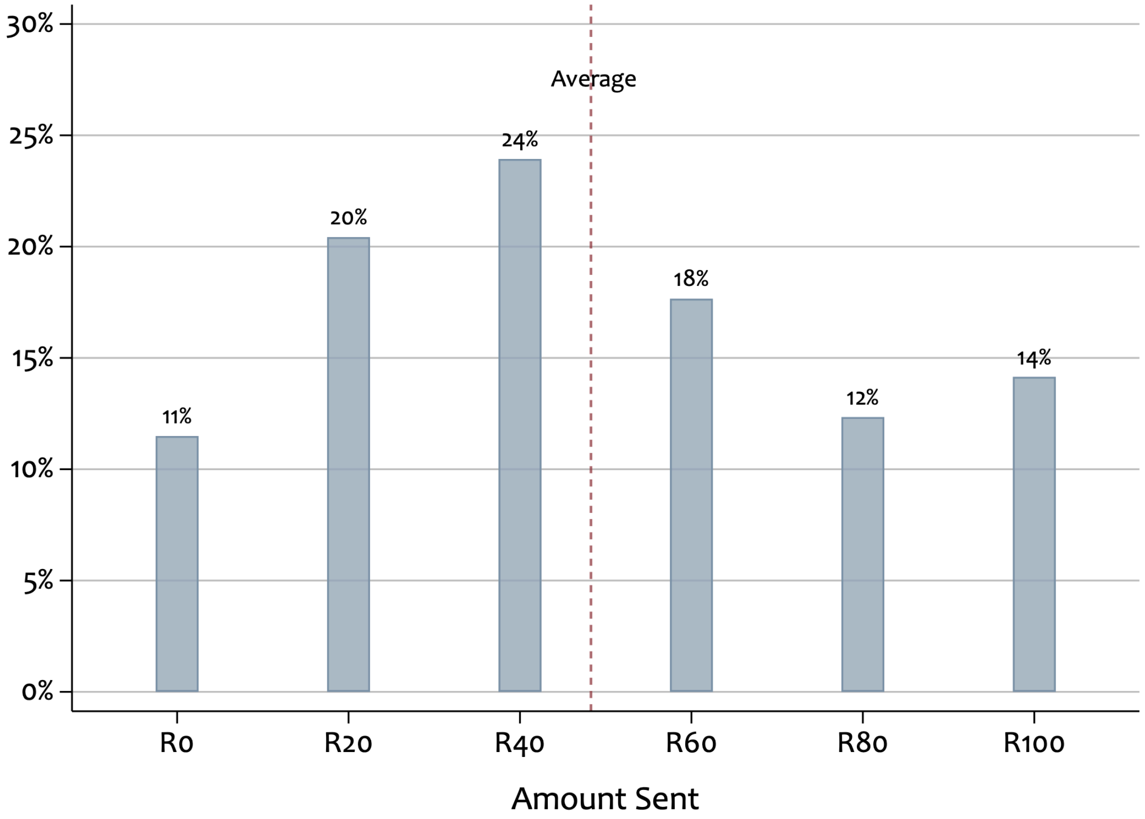


Figure 5: Amount Sent in the Baseline Treatment

We estimate an ordered logit model of amounts sent as a function of: age; whether the subject identifies as male; whether the subject's ethnicity is Black or African[28]; a categorical variable capturing a subject's financial situation on the day of the experiment; a categorical variable for task order; a quadratic of the number of risky choices in the risk preference task; and a quadratic of the amount sent in the dictator game when both the dictator and receiver had endowments of R100 and the multiplier was 3.[29] We cluster the standard errors of the estimates

[28] In terms of ethnicity, 55% of the sample is Black or African, 18% is Coloured, 11% is Indian, 16% is White, approximately 1% is Asian, and the rest of the sample responded with "Other" to the ethnicity question. In South Africa, "Coloured" is an official population group of individuals primarily of Malaysian and Indonesian descent who speak Afrikaans as a first language. "Indian" is also an official population group of people either from India or descendants of people from India.

[29] The average age of the sample is approximately 21 years old, with a minimum of 18 and a maximum of 39. Men comprise 60% of the sample. A plurality of the subjects (41%) reported being "Broke" or "Very Broke" on the day of the experiment, while 39% reported being "Neither Broke nor in Good Shape," and 20% reported being in "Good Shape" or "Very Good Shape." Subjects were split quite evenly across task orders, with 46 subjects in the T_S, T_R, D task order, 46 subjects in the T_R, T_S, D task order, 42 subjects in the D, T_S, T_R task order, and 54 subjects in the D, T_R, T_S task order. The average number of risky choices was approximately 49 (out of 100), with a standard deviation of 14. The average amount sent in the dictator game was R27.50, with a standard deviation of R27.

by subject identifier to take into account the five amount sent choices that each subject made in the task, and focus on average marginal effects (AMEs) for the interpretation of our results.[30]

We find that a standard deviation increase in age, approximately 2.7 years, is associated with an *increase* in the probability of sending R0 or R20 by 3.1 and 2.7 percentage points, respectively, and a *decrease* in the probability of sending R60, R80, or R100 by 1.4, 1.7, and 2.8 percentage points, respectively ($p < 0.01$ for all results). Thus, increases in age shift probability mass to lower amounts sent. Similarly, men, relative to other genders, are significantly *more* likely to send R0 or R20 by 4.9 and 5.3 percentage points, respectively, and significantly *less* likely to send R60, R80, or R100 by 2.3, 3.2, and 5.8 percentage points, respectively ($p < 0.05$ for all results). We find no statistically significant differences in the probabilities of amounts sent according to ethnicity. While subjects who are in "Good Shape" or "Very Good Shape," relative to subjects who are "Broke" or "Very Broke" *and* "Neutral," are significantly less likely to send R0 ($p < 0.1$), there are no other statistically significant differences in the probabilities of sending amounts greater than R0.

In contrast to our previous research (Chetty et al. [2021]), there is no statistically significant relationship between number of risky choices and the probabilities of amounts sent. There is also no statistically significant relationship between task order and the probabilities of amounts sent. However, a standard deviation increase in the amount sent in the dictator game, approximately R27, is associated with large changes in the probabilities of amounts sent: the likelihood of sending R0, R20, or R40 *decreases* by 5.2, 6.7, and 3.2 percentage points, respectively ($p < 0.001$ for all results), while the likelihood of sending R60 ($p < 0.1$), R80 ($p < 0.001$), or R100 ($p < 0.001$) *increases* by 1.5, 3.7, and 9.9 percentage points, respectively. This result suggests that subjects who send more in the dictator game also tend to send more in the trust game, which is

[30] Appendix C includes the full set of estimates, and our data and econometric code are available at: https://osf.io/ypuzd/.

similar to Cox [2004] who found that amounts sent in the dictator game account for a relatively large proportion of amounts sent in the trust game.

Figure 6 shows the distribution of *average* amounts sent by subjects in the Baseline Treatment with a kernel-weighted local polynomial smoothing overlay. We averaged the amounts sent by each subject across their five decisions in the trust game to construct the figure, which shows, given the probability mass between each R20 interval, that subjects tended to oscillate between amounts sent. Indeed, the modal (average) amount sent is R44 and the second mode is R36. The figure also shows that 4% of subjects sent R0 for all of their five choices, whereas 6% sent R100 for all five choices.

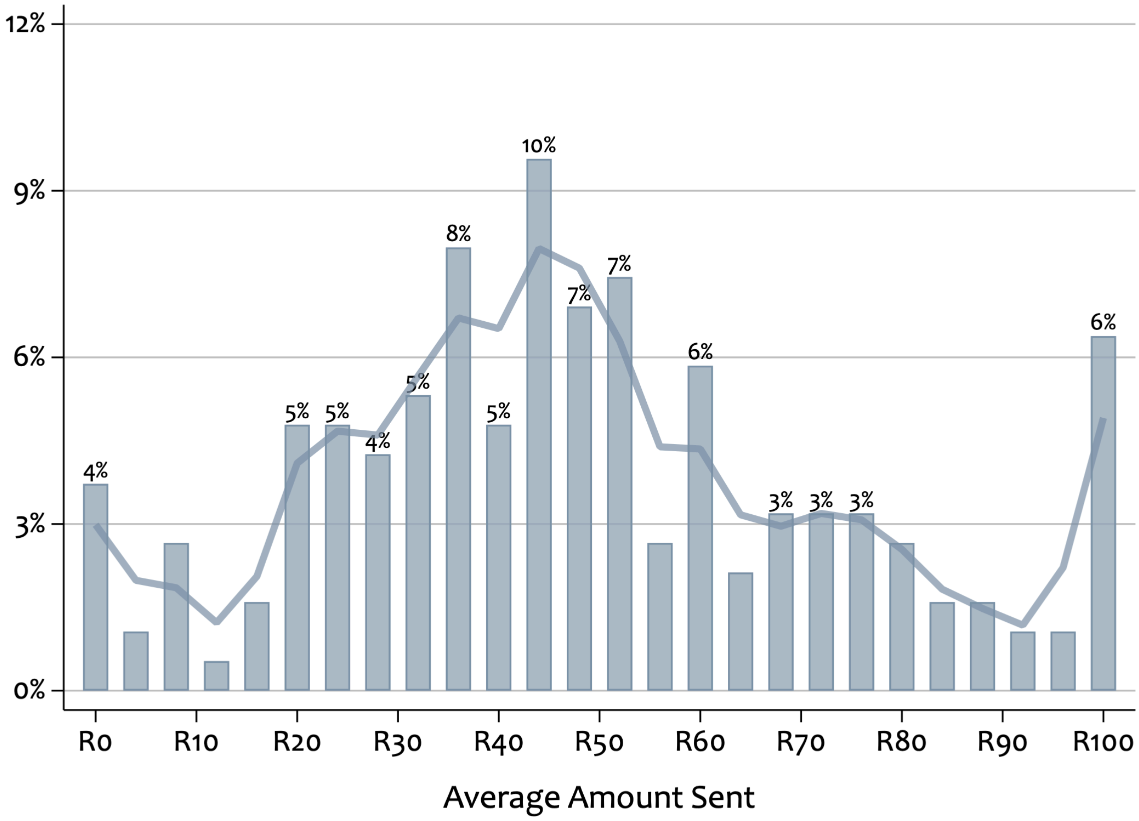


Figure 6: Average Amount Sent by Subjects in the Baseline Treatment

The average amount sent in the Baseline Treatment tracks the meta-analytic result in JM despite the information we provided to subjects to promote salience, the discretised nature of our trust game, and the fact that subjects made five amount sent choices. *The amounts returned, by* *contrast, differ markedly to the extant literature*. Specifically, for *every* positive amount sent $c^* \in$ {R20, R40, R60, R80, R100}, the modal amount returned **equalised** the first and second movers' earnings.

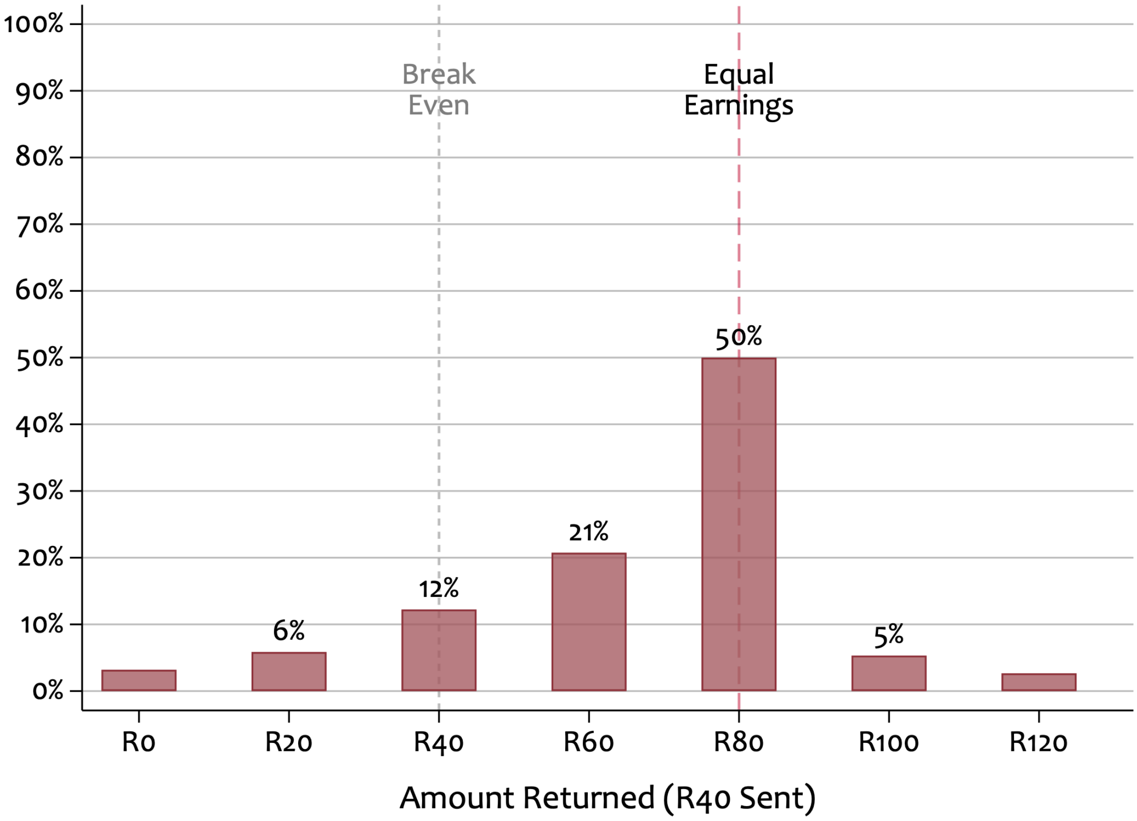


Figure 7: Amount Returned for R40 Sent

Figure 7 shows the distribution of amounts returned for R40 sent. The dashed "Break Even" line shows what the second mover would have to return for the first mover to break even given the amount sent. This is the "standard" result identified by JM: first movers tend to break even on the amount sent. However, only 12% of subjects returned this break even amount for the amount sent of R40. By contrast, 50% of subjects sent back double the amount sent, which equalises the first and second movers' earnings, whereas 21% of subjects returned R60. Appendix D includes Figures for every possible amount sent, and confirms that the modal amount returned equalised the players' earnings.

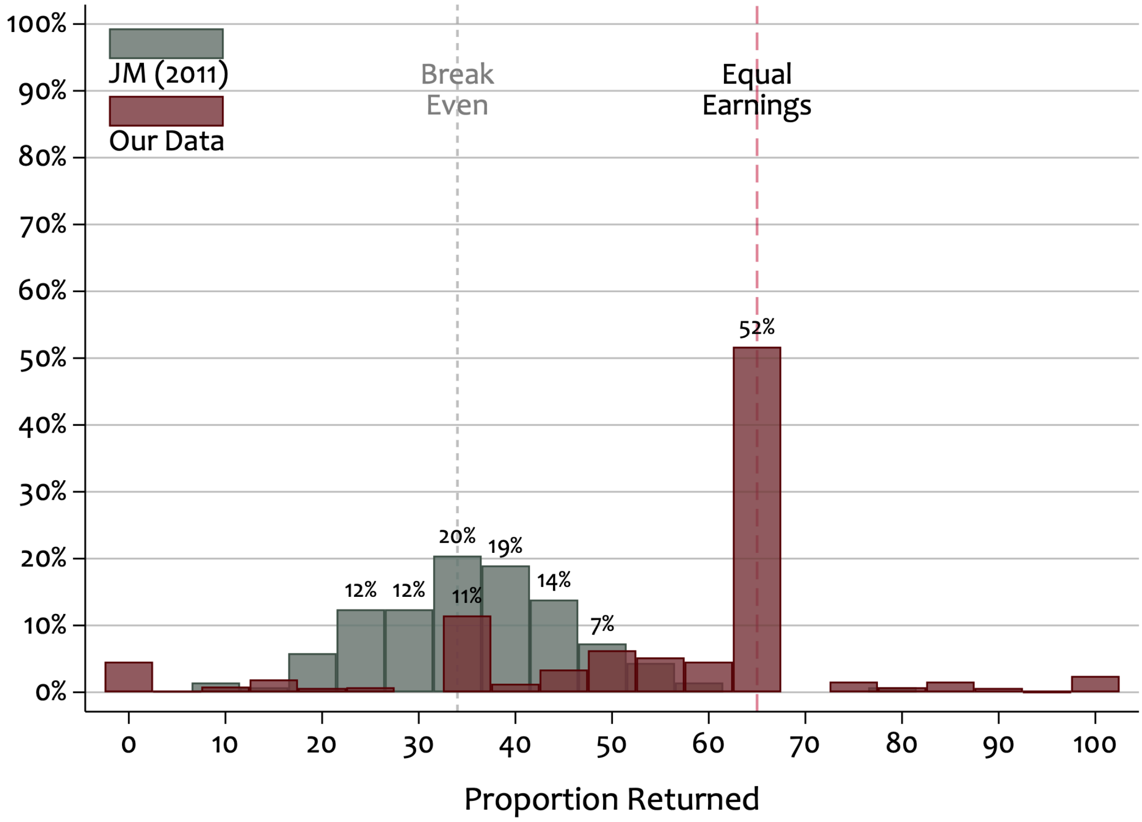


Figure 8: Proportion Retuned in JM (2011) Compared to Our Data

Figure 8 combines the proportion returned histogram from the meta-analysis in JM [p. 872] with the data we elicited in Experiment 1, pooling across all amounts sent. The difference in the distributions is dramatic: while the proportion returned is clustered tightly around the mean of 37% in JM, the median and modal amount returned in our data is 67%, and the mean is 56%. Although an eyeball test clearly shows the distributions are different, we nevertheless conduct Kolmogorov-Smirnov and Epps-Singleton tests, and find that the distribution functions of the two independent samples are **not** identical (p < 0.001 in both cases).[31]

We estimate a fractional response model of the proportion of the amount returned for each possible amount sent and include the same set of variables for our model of amounts sent,

[31] The JM data in Figure 8 is the distribution of *average* proportions returned for each study in their meta analysis, whereas we show the full distribution of proportions returned in our experiment. The JM data is not, therefore, directly comparable to ours. Nevertheless, there is no probably mass in the JM data on the mode of our distribution (67%) of proportions returned.

together with a variable indexing the magnitude of the amount sent: R20, R40, R60, etc.[32] We find no statistically significant differences in the proportion of the amount returned according to age, gender, ethnicity, and financial situation ($p > 0.12$ in all tests). Unlike JM, who find that the proportion of the amount returned tends to increase with the amount sent, there are no statistically significant differences in proportions returned by amount sent ($p > 0.28$ in all tests). By contrast, the proportion of the amount returned tends to be higher for task orders that start with the generalized dictator game (3 out of 4 comparisons, $p < 0.1$) as opposed to the trust game. In addition, a standard deviation increase in the amount sent in the dictator game, approximately R27, is associated with a 3.7 percentage point increase in the proportion of the amount returned.

We contend that the striking amounts returned are a direct product of the institution we designed with our experimental software. In standard implementations of the trust game, the terminal payoff consequences from amounts returned are not clear, because experiment instructions and software interfaces emphasise the amount that player 2 receives and the amount, therefore, they can return, without linking these choices to the players' final earnings. In analyses of a game, preferences are defined over *outcomes*, so the argument of the utility function in the trust game should be final earnings. By focussing attention on *actions*, trust game experiment instructions undermine salience, particularly given the non-trivial calculations required to determine each player's final earnings. Our simple experimental design makes the mapping between actions and final rewards clear, and obviates the need to do any mental arithmetic.

[32] We exclude the number of risky choices in the risk preference task from our model, because there is no stochastic element to the amount return decision.

B. *Experiment 2* (*Beliefs*)

In Experiment 2, we elicited subjective beliefs about amounts returned in the Baseline Treatment. Our sample consists of 106 subjects who responded to five belief questions about amounts returned for the five positive amounts sent in the trust game. When subjects hovered over an amount returned in the beliefs task, they were shown the payoff consequences for player 1 and player 2, thereby insuring they had the same information as subjects in the Baseline Treatment. Subjects in Experiment 2 were also informed that 188 people took part in the Baseline Treatment, and their earnings in the beliefs task would be determined by how one of these 188 subjects actually responded in the Baseline Treatment.

We use the econometric approach outlined in Section 4 to estimate subjective belief probability mass functions with a RDU model of risk preferences, assuming homogenous risk preferences over all individuals in the form of a representative agent. We find no statistically significant evidence of probability weighting. Specifically, our PWF estimates of $\phi = 1.025$ and $\eta = 0.987$ are not significantly different to 1: $\phi = 1$ ($p = 0.670$); $\eta = 1$ ($p = 0.782$); and $\phi = \eta = 1$ ($p = 0.901$). Thus, our (pooled) sample is best characterised by EUT.[33] We find a high level of risk aversion in our sample, with the coefficient of relative risk aversion $r = 1.333$, which is significantly greater than 1 ($p < 0.001$).[34] This implies that *token allocations* in the subjective beliefs task will be less peaked than estimated *belief distributions*.

Figure 9 shows the amounts returned for R40 sent from Figure 7, together with the estimated probability mass function from the subjective beliefs task. We see that the modal belief tracks the result in JM, in the sense that subjects believe that the R40 break-even amount is the

[33] Although EUT best represents the risk preferences of the sample in Experiment 2, we know from prior experience that some individual subjects are better characterised by RDU, while others are better characterised by EUT. Given the issues that Monroe [2020] identifies with individual-level maximum likelihood estimation, to draw inferences about the *type* of risk preferences prevalent in a sample one should estimate pooled mixture models, pooled maximum simulated likelihood models, or Bayesian Hierarchical Models, as discussed by Harrison et al. [2022] and implemented by Gao, Harrison and Tchernis [2023] and HMU.

[34] These estimates of r, ϕ and η are from the model for beliefs about the amount sent of R100. The estimates of these parameters are very similar across all models, and differ only by the second or third decimal point.

most likely. However, there is significant probability mass for amounts returned both above and below the break-even amount.

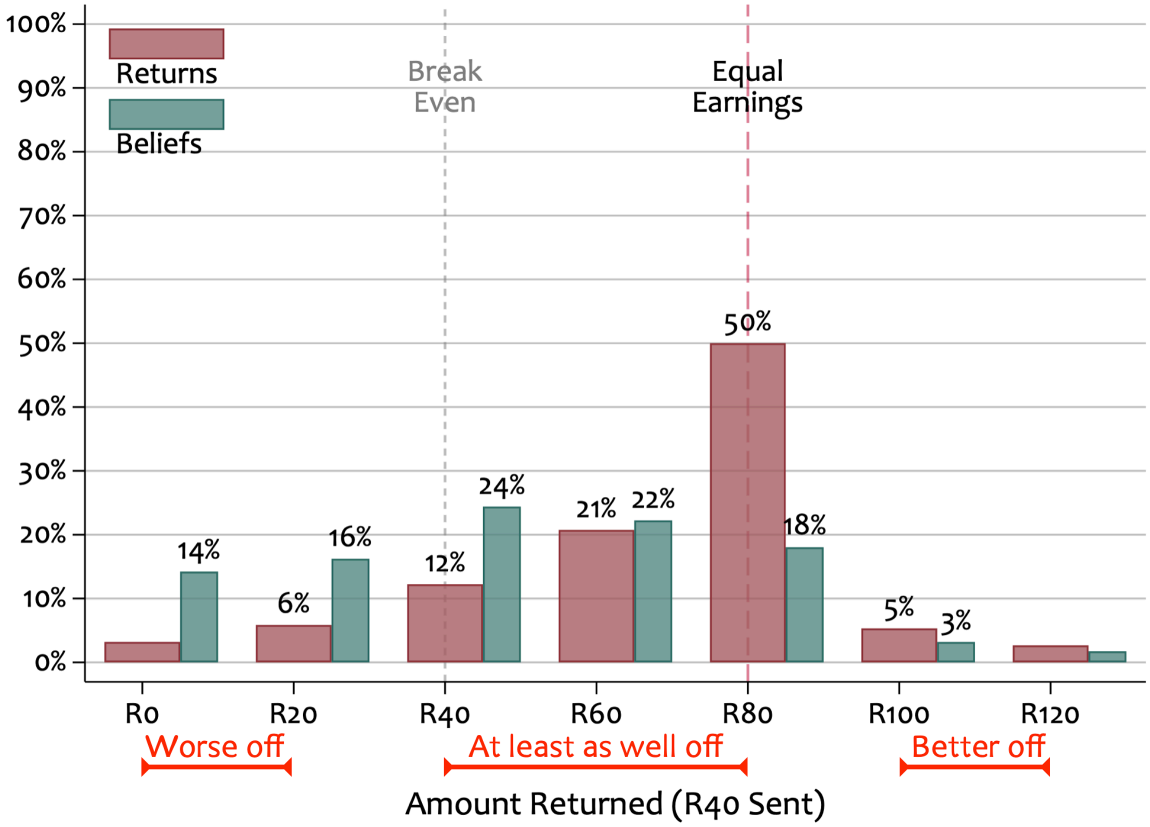


Figure 9: Amounts Returned and Beliefs about Amounts Returned for R40 Sent

An informative way to compare actual returns with beliefs about amounts returned is by dividing the amount returned region into three parts: 1) returns that make player 1 worse off (returns of R0 and R20 for R40 sent); 2) returns that make player 1 at least as well off for the amount sent, but not better off than player 2 (returns of R40, R60 and R80)[35]; and 3) returns that make player 1 better off than player 2 (returns of R100 and R120 for R40 sent). When aggregating amounts returned (and beliefs about amounts returned) in this way, the results are striking: subjects in Experiment 2 believed returns were 21 percentage points *more* likely to leave player 1 worse off relative to the actual amounts returned. Figure 9 shows why this is the case.

[35] For brevity, we refer to this as the "at least as well off" region.

The sum of the probability mass for beliefs about amounts returned of R0 and R20 is 30%, but the sum of the probability mass for the actual amounts returned is only 9%, leading to the difference of 21 percentage points. By contrast, subjects in Experiment 2 believed returns were 18 percentage points *less* likely to leave player 1 at least as well off in comparison to the actual amounts returned. Finally, subjects believed returns were 3 percentage points *less* likely to leave player 1 better off relative to actual returns. All of these differences are statistically significant ($p < 0.001$).

Appendix E includes complementary figures for all subjective belief questions, along with figures that include the raw token allocations as opposed to the estimated beliefs. Tests for whether beliefs about amounts returned are significantly different to actual amounts returned, focussing on the three regions define above, are included Appendix C. The results for other amounts sent are just as stark as those presented above for R40 sent.

C. *Experiment 3* (*Social History*)

In our Social History Treatment (Experiment 3), subjects (n = 94) in the role of first mover were shown distributions of the actual amounts returned for each amount sent in the Baseline Treatment. We hypothesised that presenting this information to subjects in an easily intelligible way would increase amounts sent, because subjects in the Baseline Treatment tended to equalise the earnings of both players, implying there were large potential gains from sending positive amounts in the trust game.

Figure 10 shows the distributions of amounts sent in the Baseline and Social History treatments. The conspicuous differences occur at the end points: amounts sent of R0 and R100. In the Baseline Treatment, 11% of all amounts sent were R0, compared to only 6% in the Social History Treatment. On the other hand, fully 23% of amounts sent in the Social History Treatment were R100, compared to only 14% in the Baseline Treatment. Thus, probability mass shifted dramatically from R0 to R100 across the Baseline and Social History treatments. Indeed, the modal amount sent in the Social History Treatment is R100, whereas the mode is R40 in the Baseline Treatment.

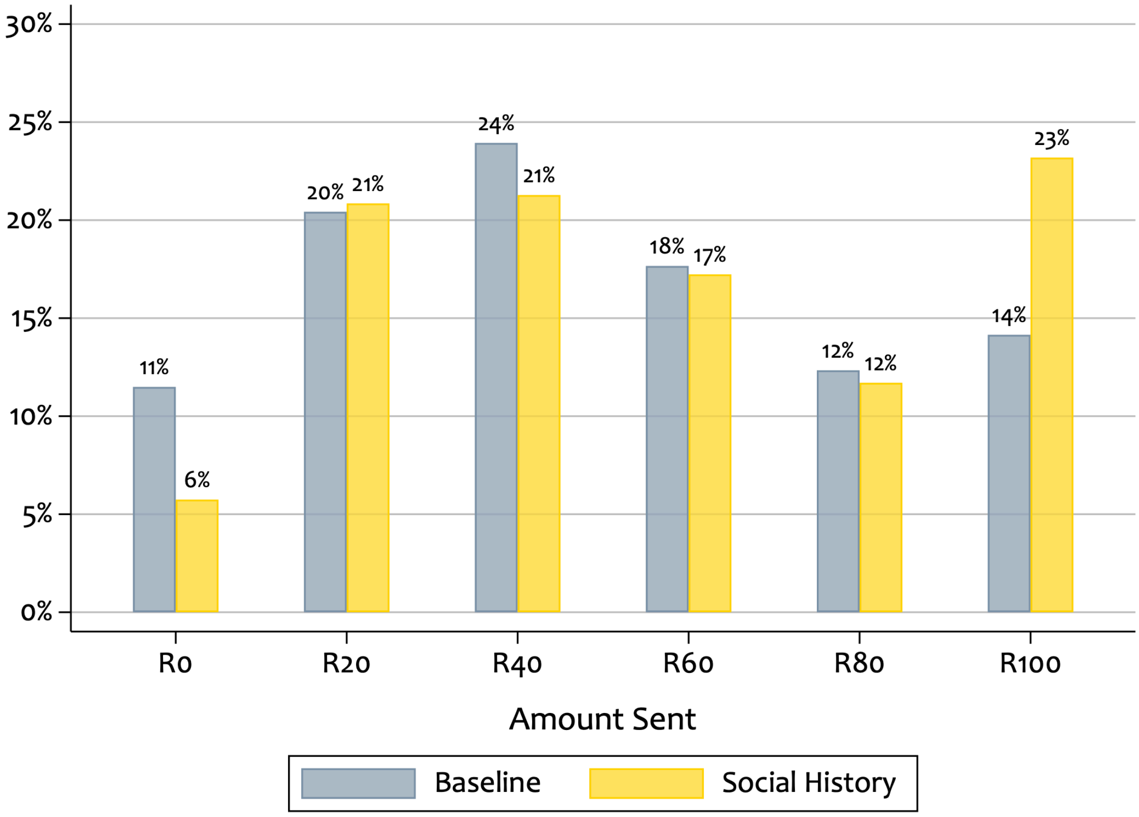


Figure 10: Amounts Sent in Baseline and Social History Treatments

We estimate an ordered logit model of amounts sent, pooling data across the Baseline and Social History treatments, which includes all of the variables from our original specification, e.g., age, gender, etc., along with a dummy variable for the Social History Treatment. Estimation of AMEs reveal a dramatic shift in probability mass across the two treatments. Specifically, the likelihood of sending R0, R20, or R40 *decreased* by 5.1, 7.1, and 2.2 percentage points, respectively, whereas the probabilities of sending R60, R80, or R100 *increased* by 2.3, 3.7 and 8.5 percentage points, respectively, in the Social History Treatment relative to the Baseline Treatment ($p < 0.05$ in all tests).

We also estimate a multinomial logit of amounts sent to test the robustness of our results to the parallel regression assumption (Long and Freese [2014, ch. 8]). The Social History

Treatment is associated with 5.2 (p = 0.045) and 6.4 (p = 0.003) percentage point *decreases* in the probability of sending R0 and R40, respectively, and a 14.4 percentage point *increase* in the probability of sending R100 (p = 0.002) relative to the Baseline Treatment. These results clearly track those from the ordered logit regression model and the histograms in Figure 10.[36]

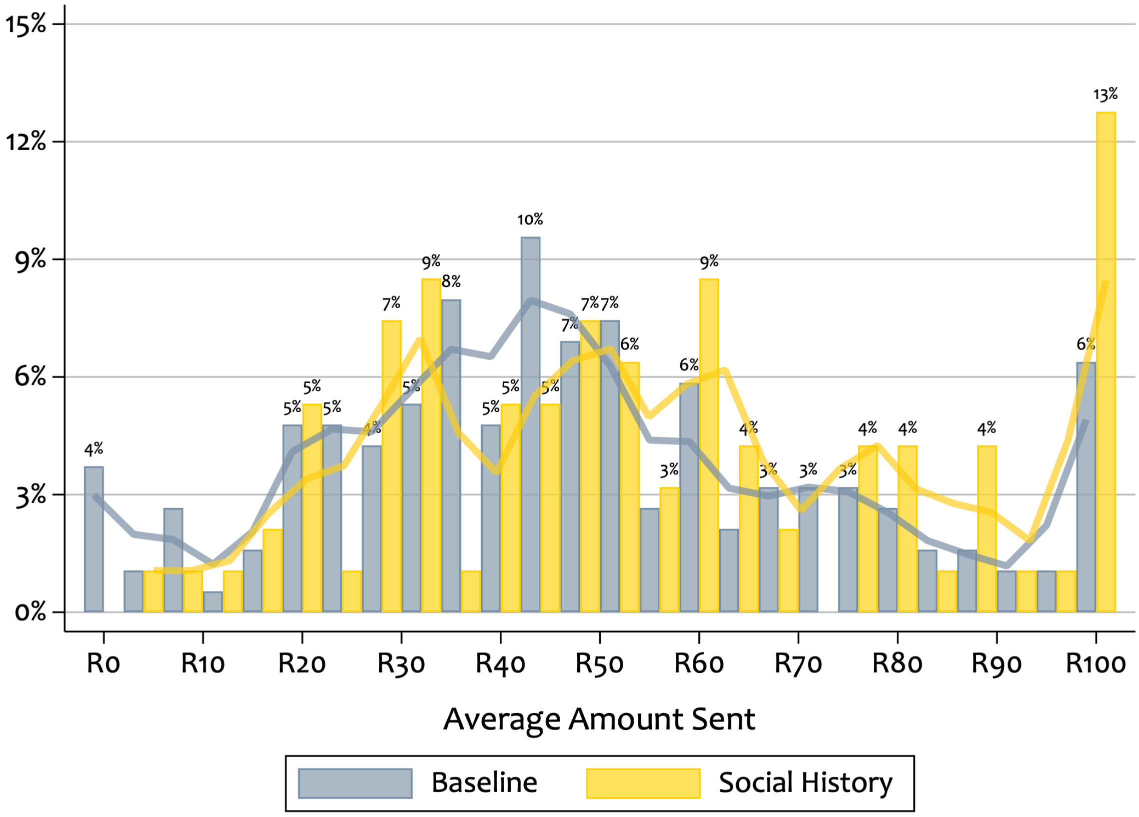


Figure 11: Average Amount Sent by Subjects in the Baseline and Social History Treatments

Figure 11 shows distributions of the average amount sent by each subject across the Baseline and Social History treatments with a kernel-weighted local polynomial smoothing overlay. The most interesting feature of the figure is that 4% of subjects sent R0 for all their five choices in the Baseline Treatment, whereas *no* subjects sent R0 for all their five choices in the Social History Treatment. By contrast, 13% of subjects sent R100 for all their choices in the Social History Treatment, and only 6% of subjects sent R100 for all their choices in the Baseline

[36] Appendix F includes additional robustness checks by focusing on the task orders common to the Baseline and Social History treatments, and amounts returned across the Baseline and Social History treatments. In addition, we focus purely on the first and last of the five amount sent choices across the Baseline and Social History treatments. Our results are robust to these different specifications.

Treatment. Figure 11 therefore emphasises the effect of the provision of social history information on amounts sent in our trust game experiments.

D. *Bayesian Updating*

Through what channel did the provision of social history information affect amounts sent in Experiment 3? The natural candidate, particularly given the robustness checks we conducted, is subjective beliefs. Bayes' Rule implies that subjects update their prior beliefs after receiving new information. This means that our subjects' posterior beliefs are a "compromise" between their prior beliefs and the likelihood of the new data, where the new data in our case is the information we presented in the Social History Treatment about amounts returned in Experiment 1.

The data presented to subjects in the Social History Treatment is categorical, because every observation is an amount returned, such as R20, from a set of possible returns, such as {R0, R20, R40, R60}. Consequently, we use the Dirichlet-Multinomial conjugate family to define the Bayesian updating process.[37] With uninformative prior beliefs, the distribution of posterior beliefs will largely reflect the likelihood of the data presented to subjects in the Social History Treatment. We therefore investigate how the "strength" of informative priors, combined with the likelihood of the data, affect the distribution of posterior beliefs. Specifically, we assume that the beliefs elicited in Experiment 2 represent the most likely draw from the Dirichlet prior distribution. We then assume that the distribution was characterised by either 25 previous observations (a weakly informative prior) or 250 previous observations (an informative prior) of amounts returned in the trust game.

[37] The Dirichlet distribution is multivariate, and defined such that each draw from the distribution is a *vector* of values in [0, 1] and the sum of these values equals 1. The Dirichlet distribution defines the prior for the probabilities of the possible amounts returned in the trust game. The Multinomial likelihood function is a generalisation of the Binomial likelihood function to events with more than two possible values. In our context, the Multinomial likelihood function models the probability of every possible amount returned, conditional on the amount sent, across the number of subjects: 188 in the Baseline Treatment.

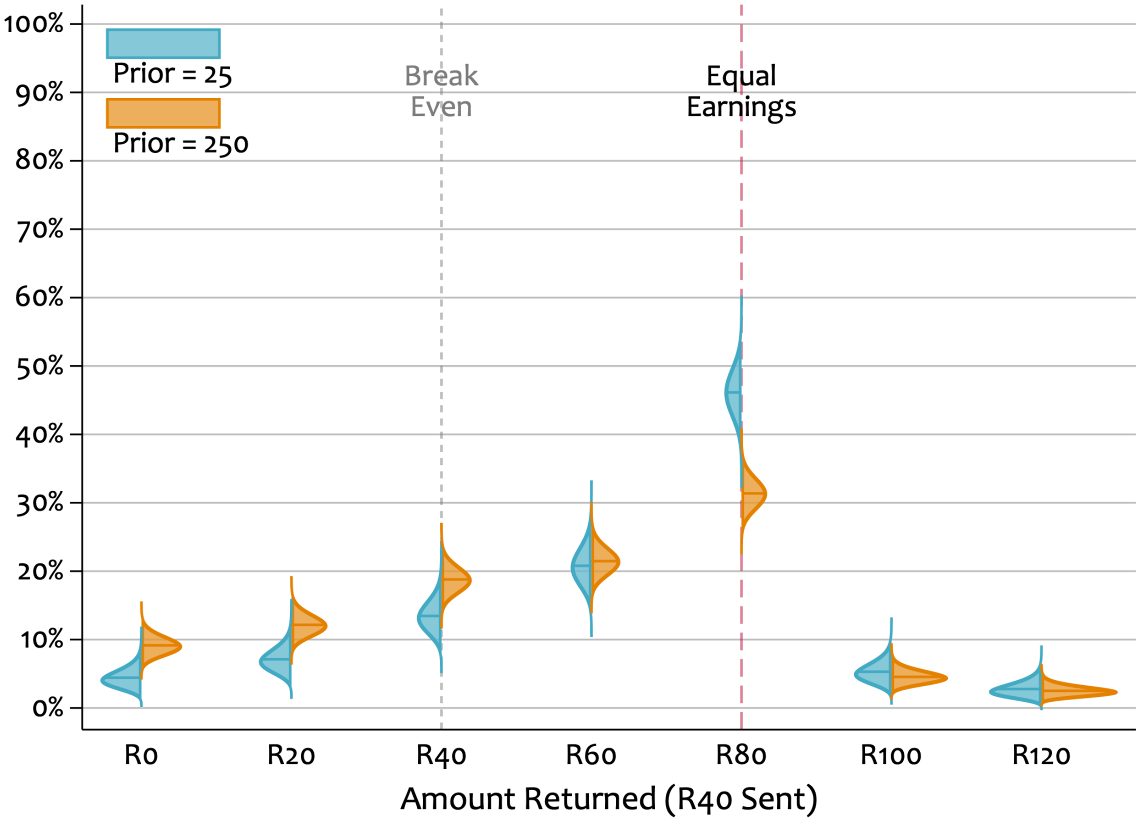


Figure 12: Posterior Beliefs About Amounts Returned

Figure 12 is a split violin plot of the posterior belief distributions assuming prior beliefs informed by 25 (teal) or 250 (orange) previous observations of amounts returned for R40 sent. Assuming 25 observations, the likelihood of the new data overwhelms the prior beliefs, and shifts posterior beliefs away from the pessimistic belief distribution elicited in Experiment 2 towards the actual returns in Experiment 1 (see Figure 9). In other words, the data shown to subjects shift probability mass from the break-even amount of R40 towards R80 (the amount that equalises the players' earnings). Assuming 250 observations, the data has a less pronounced effect on posterior beliefs, but the greatest probability mass is *still* assigned to the R80 equalise-earnings amount. Thus, the large change in amounts sent across Experiment 1 and Experiment 3 can be rationalised by applying Bayesian updating in response to the social history information presented to subjects, even if subjects held strong priors informed by 250 previous observations of amounts returned. Appendix G includes split violin plots for all amounts sent, and the same qualitative pattern in Figure 12 is reproduced in each figure.

6. Conclusion

The investment game is ubiquitous in social science investigations of trust, and the stylised results suggest that behaviour is remarkably robust across different populations. We adopted a new experimental design to enhance salience so that experimental subjects understood how the choices they made affected the final earnings of both players. We replicate the result that player 1 tends to send about 50% of their endowment to player 2, but observe a dramatic increase in amounts returned. Instead of the break-even return observed in other experiments, player 2 tends to equalise earnings, thereby making the decision to "trust" welfare enhancing for *both* players (on average). We find that subjects tend to be pessimistic about amounts returned, even when they were provided with the same information that player 2 had when making their decisions, which is clearly at odds with what we observed. However, after providing information on actual return behaviour, player 1 sends significantly more to player 2, thereby increasing the gains from trade and the welfare of both players. Finally, we demonstrate that this increase in amounts sent can be explained through belief updating.

One could argue that our experimental design may prime subjects to equalise earnings, because it shows them the amount they would need to return so that both players receive the same payoff. We do not agree with this supposition. Subjects were shown the earnings implications for each amount returned, without any particular emphasis on the amount that would equalise earnings. Furthermore, the trust game is one of complete information, in the sense that players (at least ordinally) know the preferences of their partner. Showing the amounts both players would earn from amount return choices therefore instantiates a core information property of the trust game, and promotes salience.

We acknowledge that there are settings in which using our experimental design may not be practical, such as field experiments or with subjects who are not computer literate. In these cases, experimenters could use a quiz that asks subjects to calculate the earnings of both players for different amounts returned, and only allow them to progress to the experiment if their answers are correct. We believe this is a second-best approach because there will be heterogeneity in subjects' willingness and ability to perform these calculations in the trust game itself.

Our results support the findings of Cox [2004] and Chetty et al. [2021], and suggest that the trust game does not (only) measure trust and reciprocity. With minor tweaks to the standard experimental design, we observed marked increases in amounts returned. Indeed, the modal amount returned for every amount sent equalised the players' earnings in our experiment, which is a result that, to the best of our knowledge, does not appear in the extant literature. Similarly, providing social history information can affect amounts sent in the trust game, if this information is presented in a simple, graphical manner. As Smith [1982, p. 936] argues, "…if institutions make a difference, it is because the rules make a difference, and if the rules make a difference, it is because incentives make a difference." The institution we designed for our experiments made the rules of the game clear, and mapped actions directly to rewards. We therefore encourage other researchers to adopt our experimental design to determine whether the results they typically find stand up to minor changes in the way subjects are shown the consequences of their choices.

References

ALLEN, F. (1987): "Discovering Personal Probabilities When Utility Functions Are Unknown," *Management Science*, 33, 452-454.

ASHRAF, N., I. BOHNET, AND N. PIANKOV (2006): "Decomposing Trust and Trustworthiness," *Experimental Economics*, 9

AZRIELI, Y., C. P. CHAMBERS, AND P. J. HEALY (2018): "Incentives in Experiments: A Theoretical Analysis," *Journal of Political Economy*, 126, 1472-1503.

AZRIELI, Y., C. P. CHAMBERS, AND P. J. HEALY (2020): "Incentives in Experiments with Objective Loteries," *Experimental Economics*, 23, 1-29.

BERG, J., J. DICKHAUT, AND K. A. MCCABE (1995): "Trust, Reciprocity, and Social History," *Games and Economic Behavior*, 10, 122-142.

BLANCO, M., D. ENGELMANN, A. K. KOCH, AND H.-T. NORMANN (2010): "Belief Elicitation in Experiments: Is There a Hedging Problem?," *Experimental economics*, 13, 412-438.

BROWN, A. L., AND P. J. HEALY (2018): "Separated Decisions," *European Economic Review*, 101, 20-34.

BUCHAN, N. R., R. T. A. CROSON, AND S. SOLNICK (2008): "Trust and Gender: An Examination of Behavior and Beliefs in the Investment Game," *Journal of economic behavior & organization*, 68, 466-476.

CHAUDHURI, A., AND L. GANGADHARAN (2007): "An Experimental Analysis of Trust and Trustworthiness," *Southern economic journal*, 73, 959-985.

CHEN, D. L., M. SCHONGER, AND S. WICKENS (2016): "Otree—an Open-Source Platform for Laboratory, Online, and Field Experiments," *Journal of Behavioral and Experimental Finance*, 9, 88-97.

CHETTY, R., A. HOFMEYR, H. KINCAID, AND B. MONROE (2021): "The Trust Game Does Not (Only) Measure Trust: The Risk-Trust Confound Revisited," *Journal of Behavioral and Experimental Economics*, 90, 1-14.

COSTA-GOMES, M. A., S. HUCK, AND G. WEIZSÄCKER (2010): "Beliefs and Actions in the Trust Game: Creating Instrumental Variables to Estimate the Causal Effect," ESRC Centre for Economic Learning and Social Evolution.

COX, J. C. (2004): "How to Identify Trust and Reciprocity," *Games and Economic Behavior*, 46, 260-281.

COX, J. C., V. SADIRAJ, AND U. SCHMIDT (2014): "Asymmetrically Dominated Choice Problems, the Isolation Hypothesis and Random Incentive Mechanisms," *PLoS ONE*, 9 (e90742), 1-3.

COX, J. C., V. SADIRAJ, AND U. SCHMIDT (2015): "Paradoxes and Mechanisms for Choice under Risk," *Experimental Economics*, 18, 215-250.

ECKEL, C., AND R. K. WILSON (2004): "Is Trust a Risky Decision?," *Journal of Economic Behavior & Organization*, 55, 447-465.

FECHNER, G. T. (1966/1860): *Elements of Psychophysics*. New York, NY: Holt, Rinehart and Winston 1.

GAO, X. S., G. W. HARRISON, AND R. TCHERNIS (2023): "Behavioral Welfare Economics and Risk Preferences: A Bayesian Approach," *Experimental Economics*, 26, 273-303.

HAERPFER, C., R. INGLEHARD, A. MORENO, C. WELZEL, K. KIZILOVA, J. DIEZ-MEDRANO, P. M. LAGOS, E. NORRIS, E. PONARIN, AND B. PURANEN (2022): *World Values Survey: Round Seven - Country-Pooled Datafile Version 4.0*. Madrid, Spain and Vienna, Austria: JD Systems Institute & WVSA Secretariat.

HARRISON, G. W. (1989): "Theory and Misbehavior of First-Price Auctions," *American Economic Review*, 79, 749-762.

HARRISON, G. W. (1992): "Theory and Misbehavior of First-Price Auctions: Reply," *American Economic Review*, 82, 1426-1443.

HARRISON, G. W. (1994): "Expected Utility Theory and the Experimentalists," *Empirical Economics*, 19, 223-253.

HARRISON, G. W., A. HOFMEYR, H. KINCAID, B. MONROE, D. ROSS, M. SCHNEIDER, AND J. T. SWARTHOUT (2021): "Eliciting Beliefs About Covid-19 Prevalence and Mortality: Epidemiological Models Compared with the Street," *Methods*, 195, 103-112.

HARRISON, G. W., A. HOFMEYR, H. KINCAID, B. MONROE, D. ROSS, M. SCHNEIDER, AND J. T. SWARTHOUT (2022): "Subjective Beliefs and Economic Preferences During the Covid-19 Pandemic," *Experimental Economics*, 25, 795-823.

HARRISON, G. W., J. MARTÍNEZ-CORREA, J. T. SWARTHOUT, AND E. R. ULM (2015): "Eliciting Subjective Probability Distributions with Binary Lotteries," *Economics Letters*, 127, 68-71.

HARRISON, G. W., J. MARTÍNEZ-CORREA, J. T. SWARTHOUT, AND E. R. ULM (2017): "Scoring Rules for Subjective Probability Distributions," *Journal of Economic Behavior & Organization*, 134, 430-448.

HARRISON, G. W., B. MONROE, AND E. R. ULM (2022): "Recovering Subjective Probability Distributions: A Bayesian Approach," *Working Paper WP 2022-03*, Center for the Economic Analysis of Risk (CEAR), Georgia State University.

HARRISON, G. W., AND E. E. RUTSTRÖM (2008): "Risk Aversion in the Laboratory," in *Research in Experimental Economics: Volume 12. Risk Aversion in Experiments*, ed. by J. C. Cox, and G. W. Harrison. Bingley: Emerald, 41-196.

HOLT, C. A., AND S. K. LAURY (2002): "Risk Aversion and Incentive Effects," *American Economic Review*, 92, 1644-1655.

HOSSAIN, T., AND R. OKUI (2013): "The Binarized Scoring Rule," *The Review of Economic Studies*, 80, 984-1001.

HOUSER, D., D. SCHUNK, AND J. WINTER (2006): "Trust Games Measure Trust," *Working Paper* 06-14, Universität Mannheim, Rationalitätskonzepte, Entscheidungsverhalten und ökonomische Modellierung.

HOUSER, D., D. SCHUNK, AND J. WINTER (2010): "Distinguishing Trust from Risk: An Anatomy of the Investment Game," *Journal of Economic Behavior & Organization*, 74, 72-81.

JOHNSON, N. D., AND A. A. MISLIN (2011): "Trust Games: A Meta-Analysis," *Journal of Economic Psychology*, 32, 865-889.

LONG, J. S., AND J. FREESE (2014): *Regression Models for Categorical Dependent Variables Using Stata (Third Edition).* College Station, TX: Stata Press.

MANSKI, C. F. (2004): "Measuring Expectations," *Econometrica*, 72, 1329-1376.

MATHESON, J. E., AND R. L. WINKLER (1976): "Scoring Rules for Continuous Probability Distributions," *Management Science*, 22, 1087-1096.

MCCABE, K. A., AND V. L. SMITH (2000): "A Comparison of Naïve and Sophisticated Subject Behavior with Game Theoretic Predictions," *Proceedings of the National Academy of Sciences*, 97, 3777-3781.

MCKELVEY, R. D., AND T. PAGE (1990): "Public and Private Information: An Experimental Study of Information Pooling," *Econometrica*, 58, 1321-1339.

MONROE, B. A. (2020): "The Statistical Power of Individual-Level Risk Preference Estimation," *Journal of the Economic Science Association*, 6, 168-188.

OFFERMAN, T., J. SONNEMANS, G. VAN DE KUILEN, AND P. P. WAKKER (2009): "A Truth Serum for Non-Bayesians: Correcting Proper Scoring Rules for Risk Attitudes," *Review of Economic Studies*, 76, 1461-1489.

ORTMANN, A., J. FITZGERALD, AND C. BOEING (2000): "Trust, Reciprocity, and Social History: A Re-Examination," *Experimental Economics*, 3, 81-100.

PRELEC, D. (1998): "The Probability Weighting Function," *Econometrica*, 66, 497-527.

QUIGGIN, J. (1982): "A Theory of Anticipated Utility," *Journal of Economic Behavior & Organization*, 3, 323-343.

SAPIENZA, P., A. TOLDRA-SIMATS, AND L. ZINGALES (2013): "Understanding Trust," *The Economic Journal*, 123, 1313-1332.

SAVAGE, L. J. (1971): "Elicitation of Personal Probabilities and Expectations," *Journal of the American Statistical Association*, 66, 783-801.

SCHLAG, K. H., AND J. VAN DER WEELE (2013): "Eliciting Probabilities, Means, Medians, Variances and Covariances without Assuming Risk-Neutrality," *Theoretical Economics Letters*, 3, 38-42.

SMITH, C. A. B. (1961): "Consistency in Statistical Inference and Decision," *Journal of the Royal Statistical Society: Series B (Methodological)*, 23, 1-25.

SMITH, V. L. (1982): "Microeconomic Systems as an Experimental Science," *American Economic Review*, 72, 923-955.

VYRASTEKOVA, J., AND S. GARIKIPATI (2005): "Beliefs and Trust: An Experiment," *Working Paper*, CentER Discussion Paper No. 2005-88.

WILCOX, N. T. (2011): "'Stochastically More Risk Averse:' a Contextual Theory of Stochastic Discrete Choice under Risk," *Journal of Econometrics*, 162, 89-104.

WILCOX, N. T. (2018): "Quasi-Optimal Design of a Hey and Orme Experiment," Conference Presentation: "Estimated Risk Preferences: Power and Normative Considerations," University of Cape Town, Cape Town, South Africa.

WINKLER, R. L. (1969): "Scoring Rules and Evaluation of Probability Assessors," *Journal of the American Statistical Association*, 64, 1073-1078.

Appendix A: Experiment Instructions

[Online Only]

This appendix includes the written experiment instructions for the Trust game, generalised Dictator game, Risk preference task, and subjective Beliefs task, respectively. The headers on the right of the subsequent pages (-T-, -D-, -R-, -B-) make this clear. We used identical written experiment instructions for the trust game in Experiment 1 and the trust game with Social History information in Experiment 3. The audio-visual instructions for the trust game differed across these two experiments though: subjects were shown the amount sent histograms in Experiment 3 *prior* to making their amount sent choices, but *after* making their amount return choices. In other words, we wanted Experiment 3 to be as similar to Experiment 1 as possible, except for the provision of social history information. The audio-visual experiment instructions are available at: https://osf.io/ypuzd/.

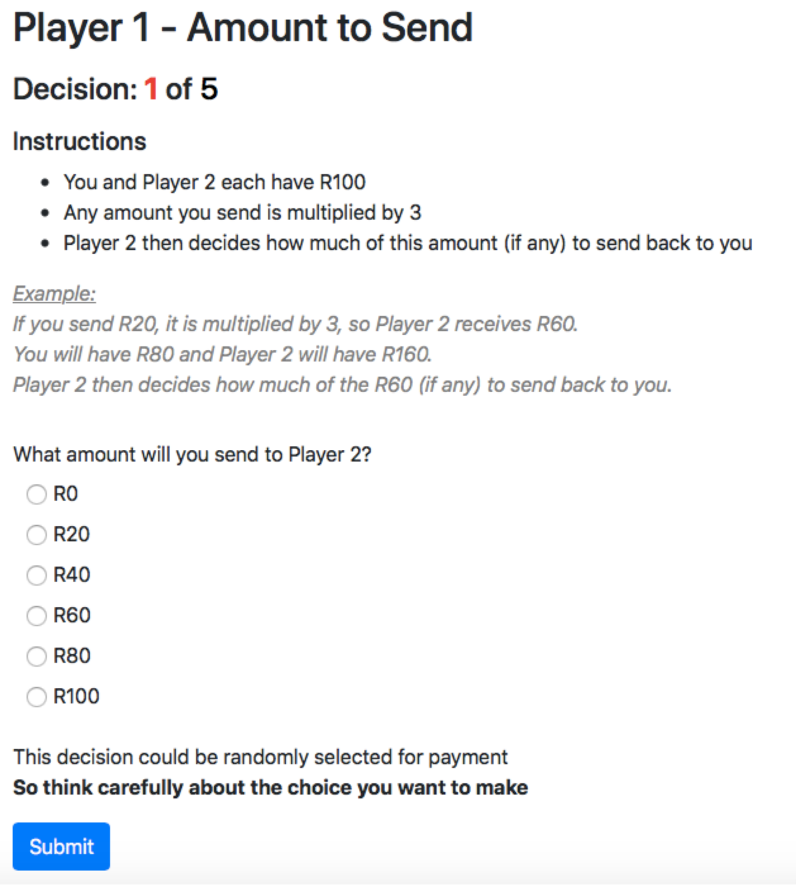
Task Instructions

This is a task where the decisions that you and another person make will determine the amounts of money that each of you earn. In this task, there are two roles, which we can refer to as Player 1 and Player 2. You will be asked to make decisions in each of these roles: as Player 1 **and** as Player 2.

At the end of the session today, you will be randomly and anonymously paired up with another person in the room. If this task is selected for payment, you and your partner will be randomly assigned to one of the two roles: either you are Player 1 and your partner is Player 2, or you are Player 2 and your partner is Player 1. Once these roles have been randomly assigned, a choice that you and your partner made will determine the earnings that each of you receive.

The task works as follows: Player 1 and Player 2 are both given R100. Player 1 needs to decide how much of the R100 (if any) to send to Player 2. Player 1 can send amounts in R20 increments: R0, R20, R40, R60, R80, or R100. The amount that Player 1 sends is automatically tripled before it is received by Player 2. So, if Player 1 sends R40 then Player 2 receives R120. Player 2 then decides how much of the R120 (if any) to send back to Player 1 and, therefore, how much to keep for himself/herself. Player 2 can send amounts in R20 increments.

These decisions will be made on a computer. This is what the computer display will look like for Player 1:



Player 1 - Amount to Send

Decision: 1 of 5

Instructions

- You and Player 2 each have R100
- Any amount you send is multiplied by 3
- Player 2 then decides how much of this amount (if any) to send back to you

Example:
If you send R20, it is multiplied by 3, so Player 2 receives R60.
You will have R80 and Player 2 will have R160.
Player 2 then decides how much of the R60 (if any) to send back to you.

What amount will you send to Player 2?

○ R0
○ R20
○ R40
○ R60
○ R80
○ R100

This decision could be randomly selected for payment
So think carefully about the choice you want to make

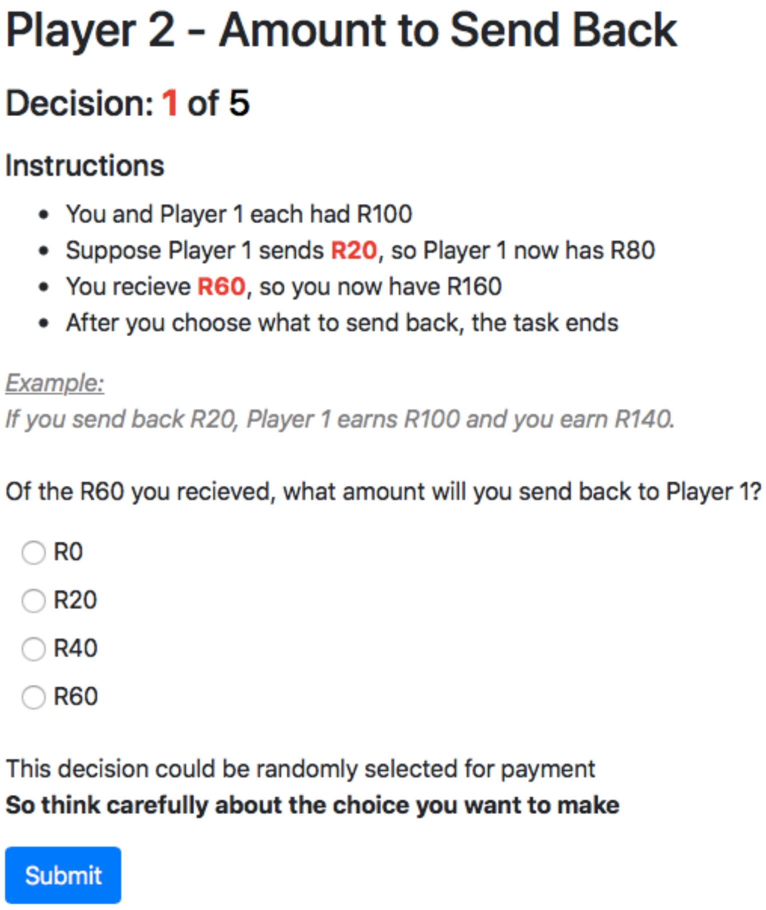
As you can see, Player 1 has to choose whether to send R0, R20, R40, R60, R80, or R100 to Player 2, knowing that any amount that is sent will be tripled and received by Player 2. Player 1 has to make this decision 5 times, on 5 separate computer screens. As any of these 5 choices could be randomly selected for payment, you should approach each choice as if it is the one that you will be paid for.

Now, when you are in the role of Player 2, you will not know how much money has been sent to you by Player 1 because we only randomly and anonymously pair up people at the end of the session today. So, you will choose how much to send back to Player 1 for every possible amount that Player 1 can send you, except if Player 1 sends R0 because then there is nothing for you to send back.

So, you will decide how much to send back to Player 1 for every possible amount Player 1 can send:

- If Player 1 sends R20, which is then tripled and becomes R60
- If Player 1 sends R40, which is then tripled and becomes R120
- If Player 1 sends R60, which is then tripled and becomes R180
- If Player 1 sends R80, which is then tripled and becomes R240
- If Player 1 sends R100, which is then tripled and becomes R300

This is what the computer display will look like for Player 2:



Player 2 - Amount to Send Back

Decision: 1 of 5

Instructions

- You and Player 1 each had R100
- Suppose Player 1 sends **R20**, so Player 1 now has R80
- You recieve **R60**, so you now have R160
- After you choose what to send back, the task ends

Example:
If you send back R20, Player 1 earns R100 and you earn R140.

Of the R60 you recieved, what amount will you send back to Player 1?

○ R0
○ R20
○ R40
○ R60

This decision could be randomly selected for payment
So think carefully about the choice you want to make

Once you have made your choices in the roles of Player 1 and Player 2 you will move on to the next task. At the end of the session today, we will determine your earnings for the first 2 tasks in the following way:

- You will be randomly and anonymously paired with another person in the room
- One of the first 2 tasks in today's session will then be randomly selected for payment
- If this task is randomly selected for payment, you and your partner will be randomly assigned to one of the two roles: either you are Player 1 and your partner is Player 2, or you are Player 2 and your partner is Player 1.
- Once these roles have been randomly assigned, one of the choices Player 1 made will be randomly selected
- Given the amount sent by Player 1, the amount that Player 2 chose to send back to Player 1 will determine the earnings that each of you receive

For example, suppose that you are randomly selected as Player 1. One of the 5 choices you made in this role will be randomly selected to determine payment. Suppose you chose to send R60 to Player 2. This amount is tripled so that Player 2 receives R180. Player 2 would have chosen what amount to send back to Player 1 for every possible amount that Player 1 could send. Suppose that when Player 1 sends R60, which is tripled to become R180, Player 2 chose to send R100 back to Player 1. Then, as Player 1, you earn the R100 that you were given at the start, minus the R60 you sent to Player 2, plus the R100 that Player 2 returned to you = R100 – R60 + R100 = **R140**. Player 2 earns the R100 that he/she was given at the start, plus the R180 that you sent, minus the R100 that Player 2 sent back to you = R100 + R180 – R100 = **R180**.

As another example, suppose that you are randomly selected as Player 2 and that Player 1 chose to send R20. This amount is tripled so that you receive R60. You would have chosen what to send back to Player 1 if Player 1 sends R20. Assume that you chose to send back R20 out of the R60 you received. Then you, as Player 2, earn the R100 that you were given at the start, plus the R60 you received from Player 1, minus the R20 that you sent back = R100 + R60 – R20 = **R140**. Player 1 earns the R100 that he/she was given at the start, minus the R20 that was sent to you, plus the R20 that you sent back to Player 1 = R100 – R20 + R20 = **R100**.

There are no right or wrong answers in this task. Please work silently and make your choices by thinking carefully about the different options. When you have finished the task, please raise your hand and a research assistant will come to you to prepare you for the next task.

Please raise your hand now.

Task Instructions

This is a task where the decisions that either you or another person make will determine the amounts of money that each of you earn. In this task, there are two roles, which we can refer to as Player 1 and Player 2. Player 2 is a passive player and does not have any choices to make. Player 1, on the other hand, has to make choices and these choices will determine the amounts of money that Player 1 and Player 2 earn.

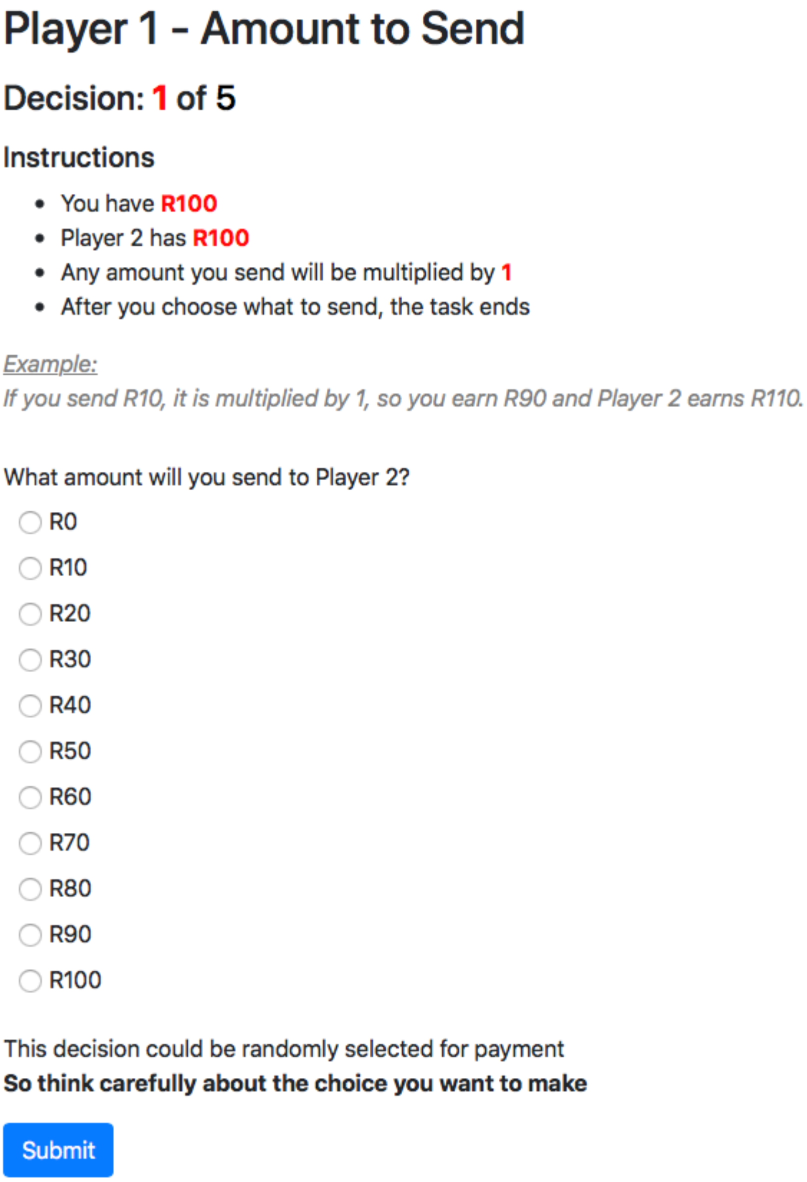
At the end of the session today, you will be randomly and anonymously paired up with another person in the room. If this task is selected for payment, you and your partner will be randomly assigned to one of the two roles: either you are Player 1 and your partner is Player 2, or you are Player 2 and your partner is Player 1. Once these roles have been randomly assigned, a choice that you or your partner made will determine the earnings that each of you receive.

The task works as follows: Player 1 is given an amount of money, e.g., R100. Player 1 needs to decide how much of this amount (if any) to send to Player 2. Player 1 can send amounts in R10 increments: R0, R10, R20, R30, R40, R50, R60, R70, R80, R90, or R100. The money that is sent is then multiplied by a number, e.g., 3, before it is received by Player 2. After Player 2 has received the amount sent by Player 1, the task ends.

Player 1 needs to make 5 of these decisions on 5 separate computer screens. While the basic structure of the task is the same for each decision, some of the details change across the decisions. For example, for one of the decisions, Player 1 will be given R100 and Player 2 will also be given R100. For another decision, Player 1 will be given R80 and Player 2 will be given R0. Thus, the amounts that Player 1 and Player 2 are given differs across the decisions.

In addition, the money that Player 1 sends to Player 2 will be multiplied by different numbers for different decisions. For example, for one of the decisions, any money that Player 1 sends will be multiplied by 3 before it is received by Player 2 (i.e., the multiplier is 3). So, if Player 1 sends R10 then Player 2 will receive R30. For another decision, any money that Player 1 sends will be multiplied by 1 before it is received by Player 2 (i.e., the multiplier is 1). So, if Player 1 sends R40, then Player 2 receives R40 in this case. Finally, for another decision, any money that Player 1 sends will be multiplied by 5 before it is received by Player 2 (i.e., the multiplier is 5). So, if Player 1 sends R20 then Player 2 receives R100.

This is what the computer display will look like:



Player 1 - Amount to Send

Decision: 1 of 5

Instructions

- You have R100
- Player 2 has R100
- Any amount you send will be multiplied by 1
- After you choose what to send, the task ends

Example:
If you send R10, it is multiplied by 1, so you earn R90 and Player 2 earns R110.

What amount will you send to Player 2?

○ R0
○ R10
○ R20
○ R30
○ R40
○ R50
○ R60
○ R70
○ R80
○ R90
○ R100

This decision could be randomly selected for payment
So think carefully about the choice you want to make

Once you have made your choices as Player 1 you will move on to the next task. At the end of the session today, we will determine your earnings for the first 2 tasks in the following way:

- You will be randomly and anonymously paired with another person in the room
- One of the first 2 tasks in today's session will then be randomly selected for payment
- If this task is randomly selected for payment, you and your partner will be randomly assigned to one of the two roles: either you are Player 1 and your partner is Player 2, or you are Player 2 and your partner is Player 1.
- Once these roles have been randomly assigned, one of the choices that Player 1 made will be randomly selected to determine the earnings that each of you receive.

Note that as any of the 5 choices that you make as Player 1 could be randomly selected for payment, you should approach each choice as if it is the one that you will be paid for. In addition, please pay careful attention to the information that is provided on every screen because the amounts of money that Player 1 and Player 2 are given and the amount by which sent money is multiplied changes across the screens.

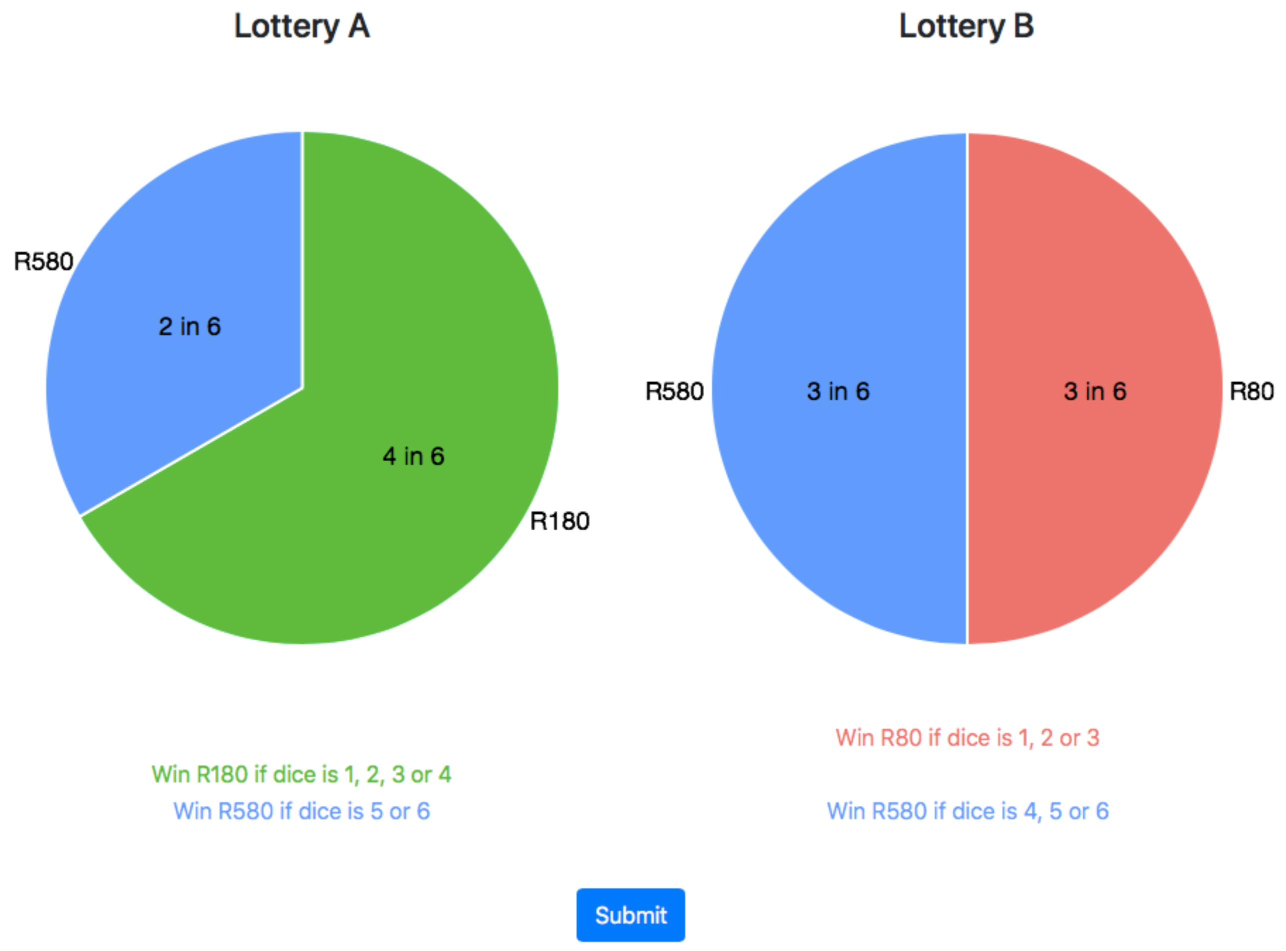
There are no right or wrong answers in this task. Please work silently and make your choices by thinking carefully about the different options, particularly because they vary across the different decisions. When you have finished the task, please raise your hand and a research assistant will come to you to prepare you for the next task.

Please raise your hand now.

Task Instructions

This is a task where you will choose between lotteries with varying prizes and chances of winning. On each computer screen you will be presented with a pair of lotteries and you will need to choose one of them. There are 100 pairs of lotteries in this task. For each pair of lotteries, you should choose the lottery you would prefer to play. You will actually get the chance to play **one** of the lotteries you choose, and you will be paid according to the outcome of that lottery, so you should think carefully about which lottery you prefer. Note that this is an individual decision-making task so you are not paired with anyone else.

Here is an example of what the computer display of a pair of lotteries might look like:



Win R180 if dice is 1, 2, 3 or 4
Win R580 if dice is 5 or 6

Win R80 if dice is 1, 2 or 3

Win R580 if dice is 4, 5 or 6

The outcome of the lotteries will be determined by rolling a regular 6-sided dice. And you will get to roll this 6-sided dice yourself at the end of the session today.

In the above example, Lottery A pays R180 with a 4-in-6 chance and R580 with a 2-in-6 chance. So when you roll the 6-sided dice, if it lands on 1, 2, 3 or 4 you will be paid R180, and if it lands on 5 or 6 you will be paid R580. The green colour in the pie chart corresponds to 4/6 of the area and illustrates the chance that the dice lands on 1, 2, 3 or 4 and your prize is R180. The blue colour in the pie chart corresponds to 2/6 of the area and illustrates the chance that the dice lands on 5 or 6 and your prize is R580.

Now look at Lottery B in the example. It pays R80 with a 3-in-6 chance, and R580 with a 3-in-6 chance. So when you roll the 6-sided dice, if it lands on 1, 2 or 3 you will be paid R80, and if it lands on 4, 5 or 6 you will be paid R580. The red colour in the pie chart corresponds to 3/6 of the area and illustrates the chance that the dice lands on 1, 2 or 3 and your prize is R80. The blue colour in the pie chart corresponds to 3/6 of the area and illustrates the chance that the dice lands on 4, 5 or 6 and your prize is R580.

Each pair of lotteries is shown on a new screen on the computer. On each screen, you should indicate which lottery you would prefer to play by clicking on the pie chart that represents the lottery. You will then click the “Submit” button to move on to the next screen with a new set of lotteries.

After you have worked through all of the 100 pairs of lotteries, raise your hand and a research assistant will come to you to determine your payment for this task. You will roll two 10-sided dice to pick a number between 1 and 100 to determine which pair of lotteries will be played out. Since there is a chance that any of your 100 choices could be played out for real, you should approach each pair of lotteries as if it is the one that you will play out.

Therefore, your earnings for this task are determined by three things:

- by which lottery you selected, Lottery A or Lottery B, for each of the 100 pairs;
- by which lottery pair is chosen to be played out in the set of 100 pairs using the two 10-sided dice; and
- by the outcome of that lottery when you roll the regular 6-sided dice.

Which lotteries you prefer is a matter of personal taste. The people next to you may be presented with different lotteries, and may have different preferences, so their responses should not matter to you. Please work silently and make your choices by thinking carefully about each lottery.

Payment for this task is in cash and is in addition to the R40 show-up fee that you receive just for being here. When you have finished the task, please raise your hand and a research assistant will come to you to determine your payment for this task and for the first two tasks that you completed.

Please raise your hand now.

Task Instructions

This is a task where you will be paid according to how accurate your beliefs are about the outcome of an interaction between two people. When you have made your choices in this task, one of them will be randomly selected to determine your payment. You will then be randomly matched with one of the 188 people who took part in this interaction in the last month and your earnings will be based on what this person actually chose to do.

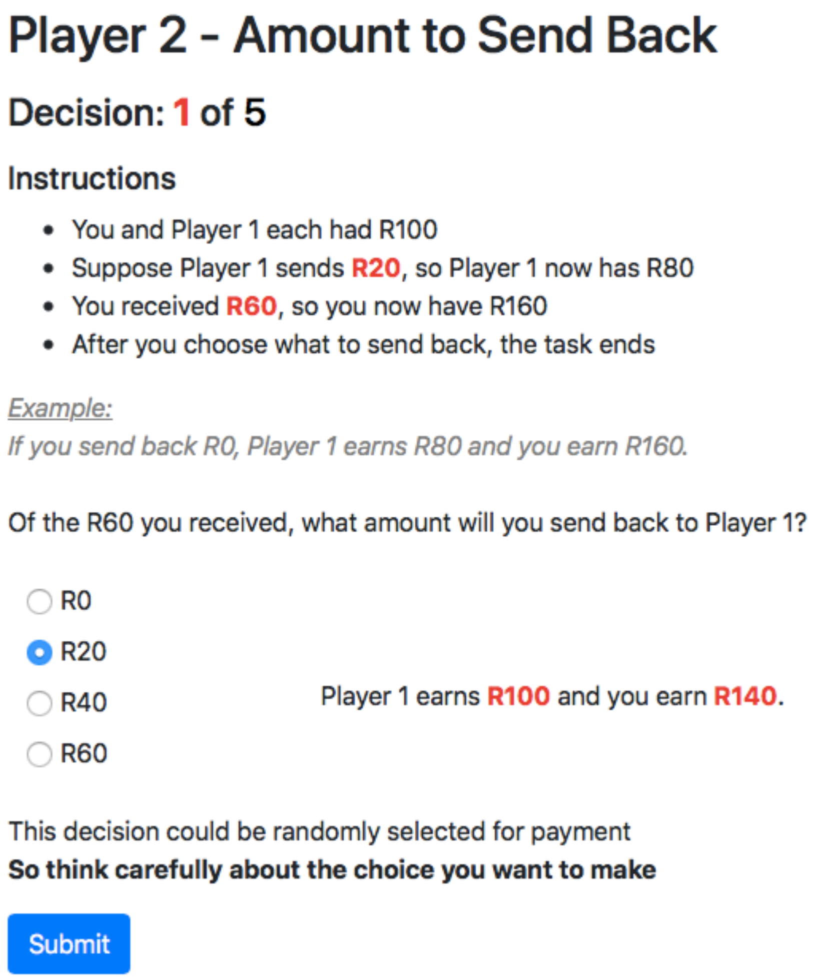
The interaction between the two people works as follows. Player 1 and Player 2 are each given R100. Player 1 decides how much of the R100 (if any) to send to Player 2. Player 1 can send amounts in R20 increments: R0, R20, R40, R60, R80, or R100. The amount that Player 1 sends is automatically tripled before it is received by Player 2. So, if Player 1 sends R20 then Player 2 receives R60. Player 2 then decides how much of the R60 (if any) to send back to Player 1 and, therefore, how much to keep for himself/herself. Player 2 can send back amounts in R20 increments. After Player 2 chooses what amount to send back, the interaction ends.

In this interaction, Player 2 had to choose what amount he/she would send back to Player 1 for every amount of money that Player 1 could send to Player 2.

So, Player 2 had to make 5 choices, one for each of the 5 amounts of money that Player 1 could send:

- If Player 1 sends R20, which is then tripled and becomes R60
- If Player 1 sends R40, which is then tripled and becomes R120
- If Player 1 sends R60, which is then tripled and becomes R180
- If Player 1 sends R80, which is then tripled and becomes R240
- If Player 1 sends R100, which is then tripled and becomes R300

The screenshot below shows you the computer display for Player 2 for the case where Player 1 sends R20, which is tripled and becomes R60. The software was designed so that whenever Player 2 clicked an amount to send back to Player 1, it told Player 2 the amount that Player 1 would earn and the amount that Player 2 would earn from this choice. So, in the screenshot below, if Player 1 sent R20 and Player 2 chose to send back R20 to Player 1 then Player 1 would earn R100 and Player 2 would earn R140 in this interaction. In other words, Player 2 was completely aware of how his/her choices would affect the earnings of both players when the choice was made about the amount to send back.



Player 2 - Amount to Send Back

Decision: 1 of 5

Instructions

- You and Player 1 each had R100
- Suppose Player 1 sends **R20**, so Player 1 now has R80
- You received **R60**, so you now have R160
- After you choose what to send back, the task ends

Example:
If you send back R0, Player 1 earns R80 and you earn R160.

Of the R60 you received, what amount will you send back to Player 1?

○ R0
● R20
○ R40
○ R60

Player 1 earns **R100** and you earn **R140**.

This decision could be randomly selected for payment
So think carefully about the choice you want to make

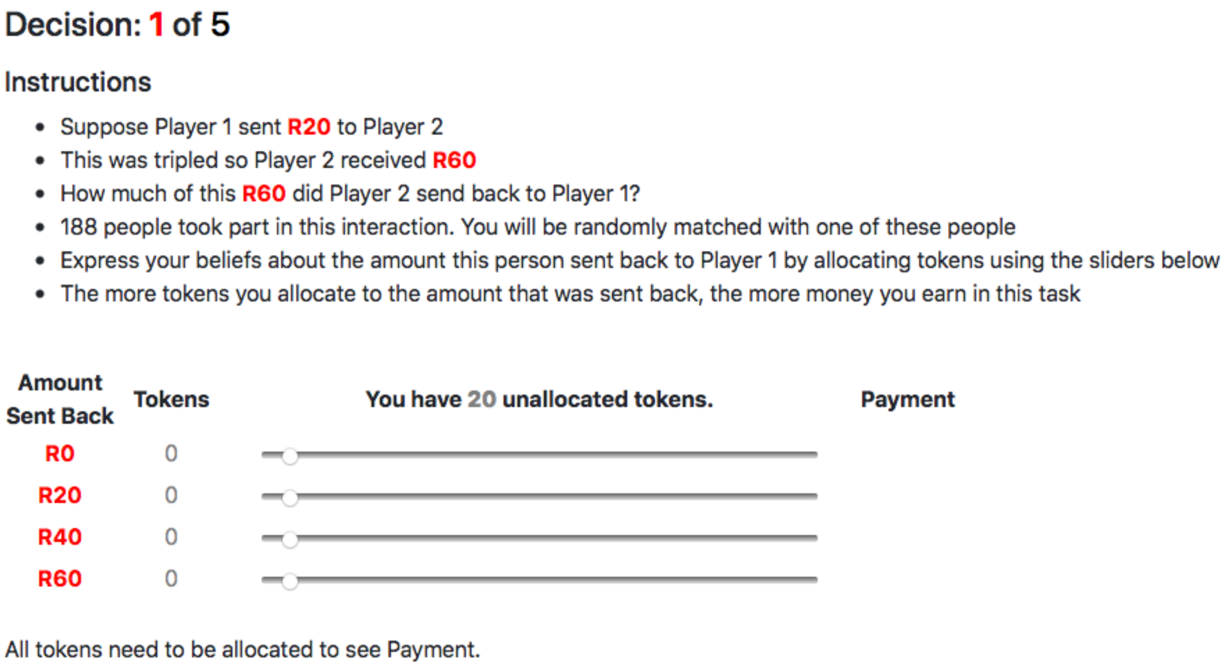
In this task, you will need to express your beliefs about the amount of money that Player 2 actually chose to send back to Player 1 for each possible amount of money that Player 1 could send to Player 2.

So, you will need to express your beliefs about how much Player 2 sent back to Player 1 in the following 5 situations:

- If Player 1 sent R20, which was tripled and became R60: How much of the R60 did Player 2 send back?
- If Player 1 sent R40, which was tripled and became R120: How much of the R120 did Player 2 send back?
- If Player 1 sent R60, which was tripled and became R180: How much of the R180 did Player 2 send back?
- If Player 1 sent R80, which was tripled and became R240: How much of the R240 did Player 2 send back?
- If Player 1 sent R100, which was tripled and became R300: How much of the R300 did Player 2 send back?

You will express your beliefs by allocating tokens to the possible amounts that Player 2 could send back to Player 1. The screenshot below shows you the computer display for the case where Player 1 sent R20 which was tripled and became R60. Player 2 could send back R0, R20, R40, or R60 and you need to express your beliefs about the amount of money that Player 2 actually chose to send back.

You express your beliefs about how much money Player 2 sent back to Player 1 by allocating tokens to the amounts of money (R0, R20, R40, and R60) in the "Amount Sent Back" column. You allocate these tokens using the sliders next to each Amount Sent Back. For each of the 5 decisions you need to make, you have 20 unallocated tokens to begin with and you must allocate all of these tokens before the payments you will receive for this particular allocation are displayed on screen.



Decision: 1 of 5

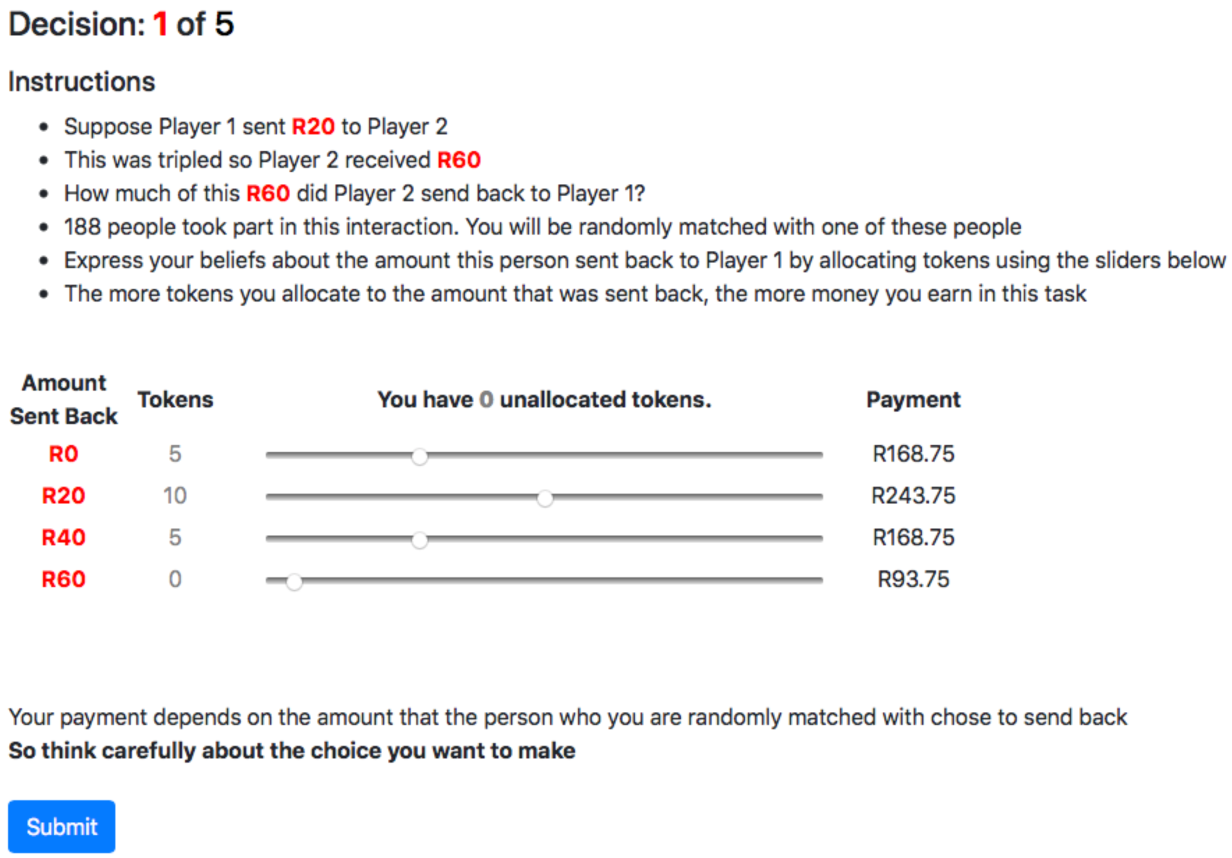
Instructions

- Suppose Player 1 sent **R20** to Player 2
- This was tripled so Player 2 received **R60**
- How much of this **R60** did Player 2 send back to Player 1?
- 188 people took part in this interaction. You will be randomly matched with one of these people
- Express your beliefs about the amount this person sent back to Player 1 by allocating tokens using the sliders below
- The more tokens you allocate to the amount that was sent back, the more money you earn in this task

Amount Sent Back	Tokens	You have 20 unallocated tokens.	Payment
R0	0		
R20	0		
R40	0		
R60	0		

All tokens need to be allocated to see Payment.

Suppose you think there is a good chance that Player 2 chose to send back R20. Then you might allocate 10 tokens to the Amount Sent Back of R20. Suppose you also think there is a pretty good chance that Player 2 sent back R0, a pretty good chance that Player 2 sent back R40, and no chance that Player 2 sent back R60. Then you might allocate 5 tokens to the Amount Sent Back of R0, 5 tokens to the Amount Sent Back of R40, and 0 tokens to the Amount Sent Back of R60. This is what the computer display will look like in this case:



Decision: 1 of 5

Instructions

- Suppose Player 1 sent **R20** to Player 2
- This was tripled so Player 2 received **R60**
- How much of this **R60** did Player 2 send back to Player 1?
- 188 people took part in this interaction. You will be randomly matched with one of these people
- Express your beliefs about the amount this person sent back to Player 1 by allocating tokens using the sliders below
- The more tokens you allocate to the amount that was sent back, the more money you earn in this task

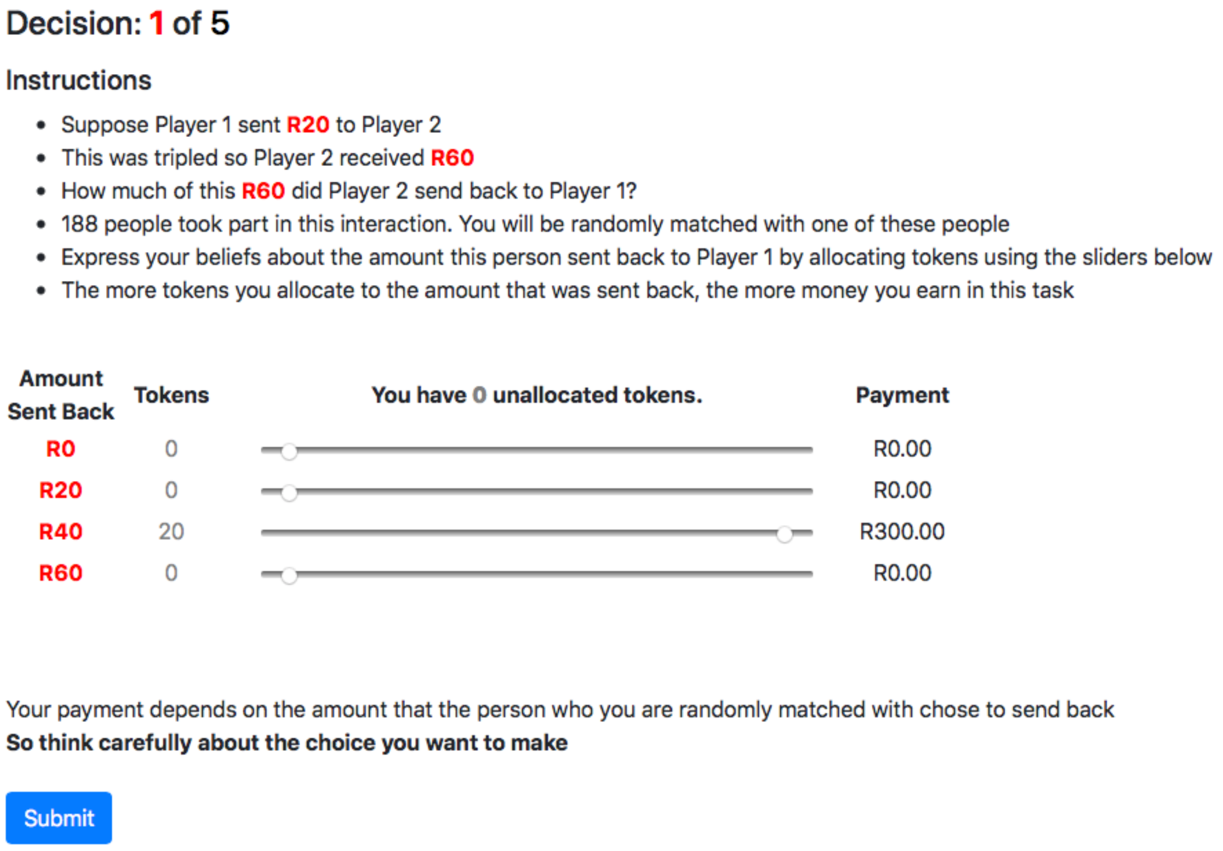
Amount Sent Back	Tokens	You have 0 unallocated tokens.	Payment
R0	5		R168.75
R20	10		R243.75
R40	5		R168.75
R60	0		R93.75

Your payment depends on the amount that the person who you are randomly matched with chose to send back
So think carefully about the choice you want to make

So here we show 5 tokens allocated to R0, 10 tokens allocated to R20, 5 tokens allocated to R40, and 0 tokens allocated to R60. Because you have allocated all of your 20 tokens, the "Payment" column is now visible and a "Submit" button appears on screen so that you can submit your choice and move on to the next decision. If you would like to change your token allocation then just use the sliders to make any adjustments.

At the end of the session today one of your five choices will be randomly selected for payment and you will be randomly matched with one of the 188 people who took part in this interaction in the last month. Note that with the token allocation above, if the person who took part in this interaction actually chose to send back R40 then your payment for this task will be R168.75.

What if you had allocated all of your tokens to the Amount Sent Back of R40? Then you would have faced the earnings outcomes shown below:



Decision: 1 of 5

Instructions

- Suppose Player 1 sent **R20** to Player 2
- This was tripled so Player 2 received **R60**
- How much of this **R60** did Player 2 send back to Player 1?
- 188 people took part in this interaction. You will be randomly matched with one of these people
- Express your beliefs about the amount this person sent back to Player 1 by allocating tokens using the sliders below
- The more tokens you allocate to the amount that was sent back, the more money you earn in this task

Amount Sent Back	Tokens	You have 0 unallocated tokens.	Payment
R0	0		R0.00
R20	0		R0.00
R40	20		R300.00
R60	0		R0.00

Your payment depends on the amount that the person who you are randomly matched with chose to send back
So think carefully about the choice you want to make

Note the "good news" and the "bad news" here. If the person who you are randomly matched with chose to send back R40, you earn the maximum payoff, shown here as R300.00. But if the person who you are randomly matched with sent back some other amount (R0, R20, or R60), then you would have earned nothing.

It is up to you to balance the strength of your personal beliefs with the risk of them being wrong. There are two important points for you to keep in mind when allocating tokens to the different amounts sent back in each decision:

1. **Your belief about the amount of money that Player 2 chose to send back to Player 1 is a personal judgement**. You may think that someone will always send back R0 or that someone will always send back R60 and your token allocation will reflect this.
2. **Your choices might also depend on your willingness to take risks or to gamble**. There is no right choice for everyone. For example, in a horse race you might want to bet on the longshot since it will bring you more money if it wins. On the other hand, you might want to bet on the favourite since it is more likely to win something.

For each decision, your choice will depend on two things: your judgement about how likely it is that each possible amount was actually sent back, and how much you like to gamble or take risks.

When you are happy with your token allocation, you should click the "Submit" button to confirm your choice and move on to the next decision. When you are finished the task, please raise your hand and a research assistant will come to you to prepare you for the next part of the session.

Please raise your hand now.

Appendix B: Econometrics
[Online Only]

Adopting the HMU approach we first specify the likelihood function to estimate risk preferences, use the candidate risk preference estimates to recover subjective beliefs, and then fit a probability distribution (in our case, a probability mass function) to the recovered beliefs. We start, therefore, by specifying our likelihood function for risk preferences.

The RDU model, due to Quiggin [1982], nests expected utility theory (EUT) when agents do not subjectively distort objective probabilities, so we use the RDU model below for the sake of generality. Let x_j represent prize j in lottery L, $p\left(x_j\right) = p_j$ the probability assigned to prize j, $w\left(x_j\right) = w_j$ the decision weight applied to prize j, and $u\left(x_j\right)$ the utility of prize j. The RDU of lottery L is:

$$\text{RDU (L)} = \sum_{j=1,\, \dots,\, m} \left[w_j \times u\left(x_j\right)\right], \tag{1}$$

where

$$w_j = \pi\left(p_j + \dots + p_m\right) - \pi\left(p_{j+1} + \dots + p_m\right), \tag{2}$$

for $j = 1, \dots, m-1$, and

$$w_j = \pi\left(p_j\right), \tag{3}$$

for $j = m$. The subscript j represents outcomes ranked from worst to best, and $\pi\left(p\right)$ is a specific probability weighting function (PWF). We use the Prelec [1998] PWF, because it allows independent specification of location and curvature in probability weighting:

$$\pi\left(p\right) = \exp\left[-\eta\left(-\ln\left(p\right)\right)^{\phi}\right] \tag{4}$$

where $1 > p > 0$, $\phi > 0$ and $\eta > 0$.

We also assume that the utility function over income $u\left(x\right)$ exhibits constant relative risk aversion (CRRA):

$$u\left(x\right) = \frac{x^{1-r}}{1-r} \tag{5}$$

where r is the coefficient of relative risk aversion.

Given two lotteries presented to subjects on a computer screen, the Left lottery L and the right lottery R, we can calculate the difference in the RDU of these two lotteries given their associated prizes, x_{jL} and x_{jR}, and probabilities, p_{jL} and p_{jR}, along with candidate values of r, ϕ, and η:

$$\nabla\mathrm{RDU} = \mathrm{RDU_R} - \mathrm{RDU_L} \tag{6}$$

To link this index to a subject's observed choices in the risk preference task we use the logistic cumulative distribution function , which yields the "logit" link function:

$$Pr\,(\mathrm{R}) = \Lambda\,(\nabla\mathrm{RDU}) \tag{7}$$

The index in (6) is linked to subject's choices by specifying that the Right lottery is chosen when $\Lambda\,(\nabla\mathrm{RDU}) > \frac{1}{2}$.

Let z denote a binary indicator of whether the subject chose lottery R (z=1) or lottery L (z=0). Then the likelihood of the observed responses, conditional on the RDU, Prelec [1998] and CRRA assumptions, depends on the estimates of r, ϕ, and η, given the statistical model in (7) and subject choices in the risk preference task. Using ι to index observations, the conditional log-likelihood for the risk preference responses is:

$$\ln L_\iota^{\mathrm{RP}}\left(r, \phi, \eta; z\right) = \sum_\iota z_\iota \ln\left(\Lambda\,(\nabla\mathrm{RDU})\right) + (1 - z_\iota)\ln\left(1 - \Lambda\,(\nabla\mathrm{RDU})\right) \tag{8}$$

Finally, we adopt the Contextual Utility behavioural error specification of Wilcox [2011] to allow choices in the risk preference task to deviate from the deterministic predictions of RDU. We therefore change the index in (6) to incorporate a Fechner [1966/1860] error term μ and a term $\lambda = u_{\max} - u_{\min}$, where $u_{\max}$ represents the utility of the highest prize in a lottery pair and $u_{\min}$ represents the utility of the lowest prize in a lottery pair, which normalises the RDU difference to lie in the interval [-1, 1]. Our new index function is therefore:

$$\nabla\mathrm{RDU} = \frac{\left(\mathrm{RDU_R} - \mathrm{RDU_L}\right)/\lambda}{\mu} \tag{9}$$

The conditional log-likelihood for the model now includes the Fechner error term μ in addition to r, ϕ, and η:

$$\ln L_{\iota}^{\mathrm{RP}}\left(r,\phi,\eta,\mu;z\right)=\sum_{\iota} z_{\iota}\ln\left(\Lambda\left(\nabla \mathrm{RDU}\right)\right)+\left(1-z_{\iota}\right)\ln\left(1-\Lambda\left(\nabla \mathrm{RDU}\right)\right). \tag{10}$$

This expression can be maximised using standard numerical optimisers to estimate the parameters r, ϕ, and η, which define risk preferences under RDU, together with the Fechner error term μ.

Now consider our subjective beliefs task where subjects had to allocate 20 tokens to n bins to express the likelihoods they assigned to particular amounts returned, conditional on an amount sent, in the trust game in Experiment 1. Each positive amount sent $c > 0$ determines the number of bins n for a particular subjective beliefs question, where $n \in \{4, 7, 10, 13, 16\}$. For example, suppose c = R40. This implies $n = 7$, because the amount returned $d \in$ {R20, R40, R60, R80, R100}. Thus, we elicited beliefs about amounts returned for the five positive amounts sent $c^{*} \in$ {R20, R40, R60, R80, R100}.

For simplicity, focus on the case were c = R20, implying $n = 4$, and $d \in$ {R0, R20, R40, R60}. A subject must allocate all 20 tokens across the 4 bins, but there are no constraints in terms of the allocation, e.g., a subject can allocate all of their tokens to one bin. We summarise a token allocation by dividing the allocation to each bin by 20 (the number of tokens), and refer to this as the subject's observed *report* $\mathbf{s}=\left(s_{\mathrm{R0}},s_{\mathrm{R20}},s_{\mathrm{R40}},s_{\mathrm{R60}}\right)=\left(s_1,s_2,s_3,s_4\right)$, where $s_i \geq 0 \forall i$ and $\sum_i s_i = 1$.[1]

We used the QSR to reward subjects for their reports. If i is the bin in which the actual amount returned lies, then the payoff is defined by Matheson and Winkler [1976, p. 1088, equation (6)] as: $Q\left(s_i|s\right)=\left(2\times s_i\right)-\sum_i\left(s_i\right)^2$. This means that the payoff is determined by doubling the report s_i to the correct bin i, and penalising the subject depending on the full report $\mathbf{s}$ across all n bins. We used an endowment, α, and scaling parameter, β, to prevent subjects from incurring losses. In this case, the payoff from the QSR is: $Q\left(s_i|s\right)=\alpha+\beta\left[\left(2\times s_i\right)-\sum_i\left(s_i\right)^2\right]$. We set α = R150 and β = R150 in our experiments. We can now define the RDU of a report $\mathbf{s}$:

[1] The ordered list $\mathbf{s}$ is an n-tuple, where n represents the number of bins for a subjective beliefs question.

$$\mathrm{RDU}(\mathbf{s}) = \sum_{i=1}^{n} w_i(b_i) \times u\left(Q(s_i|\mathbf{s})\right), \tag{11}$$

where b_i represents an agent's belief that the amount returned lies in bin i. In the case where a subject is risk neutral and obeys SEU, $b_i = s_i \forall i$, but if either of these assumptions do not hold then $b_i \neq s_i$.[2] We use the Prelec [1998] PWF (4) to calculate decision weights w_i, and the CRRA function (5) to determine the utility of the payoff $Q(s_i|\mathbf{s})$.

HMU prove the existence of a function $g(\mathbf{s}|\psi)$ that maps the observed report **s** to beliefs **b** given risk preferences ψ.[3] Lemma 2 of HMU [p. 11] explains the assumptions that must hold for this function to exist: the utility function $u(\cdot)$ is increasing, continuous, twice differentiable and concave, and the report **s** maximises RDU. If these assumptions hold, a report **s** maps to unique "recovered" beliefs **b** given risk preferences ψ. We use weighted maximum likelihood estimation to fit a probability mass function to the recovered beliefs. Candidate beliefs $\widetilde{b}_i$ are logged and weighted by recovered beliefs $b_i = g_i(s|\widetilde{\psi})$, and then summed across the n bins. The recovered beliefs are determined by candidate risk preferences $\widetilde{\psi}$ and the observed report **s**. The weighted log-likelihood for recovered beliefs **b** is therefore:

$$\ln L^B\left(\mathbf{b}|\ \mathbf{s}, \widetilde{\psi}\right) = \sum_{i}^{n} b_i \times \ln\left(\widetilde{b}_i\right). \tag{12}$$

The log-likelihood for risk preferences (10) is combined with the weighted log-likelihood for recovered beliefs (12) to form the joint log-likelihood of risk preferences and beliefs:

$$\ln L^{\text{Joint}} = \ln L^{\text{RP}} + \ln L^{\text{B}}. \tag{13}$$

The specification in (13) is maximised to estimate the risk preference parameters r, ϕ, and η, the Fechner error term μ, and the recovered beliefs **b**.

[2] Harrison et al. [2017] show that the observed *reports* of risk averse SEU agents, with coefficients of relative risk aversion in the range often observed in lab experiments, are "close" to their *beliefs*. This is not the case, however, with RDU agents, because there can be first order differences between observed reports and beliefs.

[3] Under RDU, risk preferences are determined by the curvature of the utility function $u(\cdot)$ along with the probability weighting parameters ϕ and η. We therefore use $\psi = \{u(\cdot), \phi, \eta\}$ to represent risk preferences under RDU.

Appendix C: Estimates

[Online Only]

We include all of the estimates referenced in the main text in the order that they appear there. The covariates in the statistical models are explained in the text. The notation for the subjective beliefs estimation is:

• The parameter of the utility function is *r* in the estimates, and r in Appendix B.

• The parameters of the Prelec [1998] probability weighting function are phi and eta, and ϕ and η in Appendix B.

• The Fechner error term is noise, and μ in Appendix B.

• After the subjective belief estimates for every amount sent, e.g., R20, R40, etc., we calculate the probability mass of each bin, e.g., bin1, bin2, etc., along with the mean (mean) and standard deviation (sd) of the probability mass function.

```
. /**********************************************************************/
. /*  SECTION 0: Some simple results first
>     Notes: This DO file performs the analyses of the TG data from the
>     study conducted by Hofmeyr, Kincaid and Monroe at UCT in 2018. */
. /**********************************************************************/

. /*----------------------------------------------------*/
.    /* [>   Summary Statistics   <] */
. /*----------------------------------------------------*/
.
. /* [> Baseline treatment <] */
.
. * Summarise the continuous and binary variables
. summarize age male no_risky_choices if period == 1 & treatment == 1

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
         age |        187    21.26738    2.755865         18         39
        male |        188    .6010638    .4909871          0          1
no_risky_c~s |        188    49.32447    14.28009         13         93

.
. * Categorical variables
. tab task_order if period == 1 & treatment == 1

  Order in which |
        subjects |
completed the TG |
   and DG, and R |
           and B |      Freq.     Percent        Cum.
-----------------+-----------------------------------
 1. Ts, Tr, D, R |         46       24.47       24.47
 2. Tr, Ts, D, R |         46       24.47       48.94
 3. D, Ts, Tr, R |         42       22.34       71.28
 4. D, Tr, Ts, R |         54       28.72      100.00
-----------------+-----------------------------------
           Total |        188      100.00

. tab financial_situation_3_cat if period == 1 & treatment == 1

      Financial |
  Situation - 3 |
     categories |      Freq.     Percent        Cum.
----------------+-----------------------------------
V Broke / Broke |         77       40.96       40.96
        Neutral |         73       38.83       79.79
  Good / V Good |         38       20.21      100.00
----------------+-----------------------------------
          Total |        188      100.00

. tab black_african if period == 1 & treatment == 1

Black / African |      Freq.     Percent        Cum.
----------------+-----------------------------------
          Other |         85       45.21       45.21
Black / African |        103       54.79      100.00
----------------+-----------------------------------
          Total |        188      100.00
```

```
. tab ethnicity if period == 1 & treatment == 1

              Ethnicity |      Freq.     Percent        Cum.
------------------------+-----------------------------------
     1. Black / African |        103       54.79       54.79
            2. Coloured |         33       17.55       72.34
              3. Indian |         20       10.64       82.98
               4. White |         30       15.96       98.94
               5. Asian |          1        0.53       99.47
               6. Other |          1        0.53      100.00
------------------------+-----------------------------------
                  Total |        188      100.00

. 
. * Summarize DG data for multiplier of 3, and R100 endowments for both players
. * in baseline treatment
. summarize amount_sent_dg if dictator == 1 & dictator_multiplier == 3 ///
> & dictator_receiver_endowment == 100 & dictator_sender_endowment == 100 ///
> & treatment == 1

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
amount_sen~g |        188        27.5    27.03978          0        100

. 
. * Tabulate average amount sent in baseline treatment
. tab avg_amount_sent_tg if treatment == 1 & period == 1

    Average |
amount sent |
   in Trust |
       game |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |          7        3.72        3.72
          4 |          2        1.06        4.79
          8 |          5        2.66        7.45
         12 |          1        0.53        7.98
         16 |          3        1.60        9.57
         20 |          9        4.79       14.36
         24 |          9        4.79       19.15
         28 |          8        4.26       23.40
         32 |         10        5.32       28.72
         36 |         15        7.98       36.70
         40 |          9        4.79       41.49
         44 |         18        9.57       51.06
         48 |         13        6.91       57.98
         52 |         14        7.45       65.43
         56 |          5        2.66       68.09
         60 |         11        5.85       73.94
         64 |          4        2.13       76.06
         68 |          6        3.19       79.26
         72 |          6        3.19       82.45
         76 |          6        3.19       85.64
         80 |          5        2.66       88.30
         84 |          3        1.60       89.89
         88 |          3        1.60       91.49
         92 |          2        1.06       92.55
         96 |          2        1.06       93.62
        100 |         12        6.38      100.00
------------+-----------------------------------
      Total |        188      100.00

. 
. * Summarize average amount sent in baseline treatment
. summarize avg_amount_sent_tg if treatment == 1 & period == 1

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
avg_amount~g |        188     48.2766    25.52399          0        100

. 
```

```
. * Summarize amount sent and proportion of amount returned in baseline treatment
. summarize choice_logit_send choice_logit_return_prop if treatment == 1, detail

                         Amount Sent
-------------------------------------------------------------
      Percentiles      Smallest
 1%            0              0
 5%            0              0
10%            0              0       Obs                 940
25%           20              0       Sum of wgt.         940

50%           40                      Mean            48.2766
                        Largest       Std. dev.      31.23323
75%           80            100
90%          100            100       Variance       975.5145
95%          100            100       Skewness       .2012532
99%          100            100       Kurtosis       1.997448

                  choice_logit_return_prop
-------------------------------------------------------------
      Percentiles      Smallest
 1%            0              0
 5%     .0833333              0
10%     .3333333              0       Obs                 940
25%     .4666667              0       Sum of wgt.         940

50%     .6666667                      Mean           .5601773
                        Largest       Std. dev.      .2043327
75%     .6666667              1
90%     .6666667              1       Variance       .0417518
95%     .7888889              1       Skewness      -1.027394
99%            1              1       Kurtosis       4.140436

.
. * Tabulate amount returned as a function of amount sent in baseline treatment
. tab choice_logit_return choice_order if treatment == 1

    Amount |
  Returned |
 (for each |
    Amount |                      choice_order
     Sent) |         1          2          3          4          5 |     Total
-----------+-------------------------------------------------------+----------
        R0 |         9          6          8          8         12 |        43
       R20 |        44         11          3          5          2 |        65
       R40 |       130         23          5          5          2 |       165
       R60 |         5         39         18          6          1 |        69
       R80 |         0         94         24          8          1 |       127
      R100 |         0         10         35          8         15 |        68
      R120 |         0          5         82         20          4 |       111
      R140 |         0          0          5         24          8 |        37
      R160 |         0          0          4         82         14 |       100
      R180 |         0          0          4         11         19 |        34
      R200 |         0          0          0          4         98 |       102
      R220 |         0          0          0          2          4 |         6
      R240 |         0          0          0          5          2 |         7
      R260 |         0          0          0          0          1 |         1
      R280 |         0          0          0          0          1 |         1
      R300 |         0          0          0          0          4 |         4
-----------+-------------------------------------------------------+----------
     Total |       188        188        188        188        188 |       940

.
.
. /*------------------------------------ End of SECTION 0 ------------------------------------*/
.
.
```

```
. /**********************************************************************/
. /*  SECTION 1: Ordered Logit analyses of the TG Amount Sent data
>     Notes: Estimates for Baseline treatment. */
. /**********************************************************************/
.
. /*----------------------------------------------------*/
.    /* [>   1.1.  Baseline Treatment   <] */
. /*----------------------------------------------------*/
.
. /* [> Primary ordered logit with task order (to) variables <] */
. * Set estimation variables
. local est_vars "c.age i.male i.black_african i.financial_situation_3_cat
c.no_risky_choices##c.no_risky_choices c.amount_
> sent_dg##c.amount_sent_dg i.base_info_to"

.
. ologit choice_logit_send `est_vars' if treatment == 1, cluster(subjectid)

Iteration 0:   log pseudolikelihood = -1642.6927
Iteration 1:   log pseudolikelihood = -1534.6538
Iteration 2:   log pseudolikelihood = -1530.4609
Iteration 3:   log pseudolikelihood = -1530.4345
Iteration 4:   log pseudolikelihood = -1530.4345

Ordered logistic regression                             Number of obs =    935
                                                        Wald chi2(12) =  75.90
                                                        Prob > chi2   = 0.0000
Log pseudolikelihood = -1530.4345                       Pseudo R2     = 0.0683

                                                     (Std. err. adjusted for 187 clusters in subjectid)
-------------------------------------------------------------------------------------------------------
                                      |               Robust
                    choice_logit_send | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
--------------------------------------+----------------------------------------------------------------
                                  age |  -.1059919   .0322252    -3.29   0.001    -.1691521   -.0428317
                               1.male |  -.5356529   .2385929    -2.25   0.025    -1.003286   -.0680194
                                      |
                        black_african |
                     Black / African  |   .0980217   .2205613     0.44   0.657    -.3342705    .5303139
                                      |
            financial_situation_3_cat |
                             Neutral  |  -.0438034   .2337783    -0.19   0.851    -.5020004    .4143936
                       Good / V Good  |   .5201557    .330076     1.58   0.115    -.1267815    1.167093
                                      |
                     no_risky_choices |  -.0302052   .0375065    -0.81   0.421    -.1037166    .0433063
                                      |
c.no_risky_choices#c.no_risky_choices |   .0003205   .0003658     0.88   0.381    -.0003966    .0010375
                                      |
                       amount_sent_dg |   .0211397   .0118608     1.78   0.075     -.002107    .0443863
                                      |
    c.amount_sent_dg#c.amount_sent_dg |   .0000778   .0001291     0.60   0.547    -.0001752    .0003309
                                      |
                         base_info_to |
                     2. Tr, Ts, D, R  |  -.2863928   .2605631    -1.10   0.272    -.7970871    .2243016
                     3. D, Ts, Tr, R  |   .0073288   .3106459     0.02   0.981     -.601526    .6161836
                     4. D, Tr, Ts, R  |  -.4459578   .2890196    -1.54   0.123    -1.012426    .1205103
--------------------------------------+----------------------------------------------------------------
                                /cut1 |  -4.916383   1.264429                     -7.394618   -2.438148
                                /cut2 |  -3.496662   1.247369                     -5.941461   -1.051863
                                /cut3 |  -2.334064   1.239945                     -4.764311    .0961832
                                /cut4 |  -1.412596   1.235743                     -3.834607    1.009415
                                /cut5 |  -.4925649   1.238544                     -2.920066    1.934937
-------------------------------------------------------------------------------------------------------

.
.
```

```
. /* [> Average Marginal Effects (AMEs) <] */
. local est_vars_no_factor "age male black_african financial_situation_3_cat no_risky_choices
amount_sent_dg base_info_to"

. foreach var of varlist `est_vars_no_factor' {
  2.     mchange `var', amount(sd) brief
  3. }

ologit: Changes in Pr(y) | Number of obs = 935

Expression: Pr(choice_logit_send), predict(outcome())

             |        R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
age          |
         +SD |     0.031      0.027      0.001     -0.014     -0.017     -0.028
     p-value |     0.005      0.000      0.625      0.002      0.001      0.001

ologit: Changes in Pr(y) | Number of obs = 935

Expression: Pr(choice_logit_send), predict(outcome())

             |        R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
male         |
      1 vs 0 |     0.049      0.053      0.011     -0.023     -0.032     -0.058
     p-value |     0.012      0.034      0.201      0.021      0.028      0.037

ologit: Changes in Pr(y) | Number of obs = 935

Expression: Pr(choice_logit_send), predict(outcome())

                          |        R0        R20        R40        R60        R80       R100
--------------------------+------------------------------------------------------------------
black african             |
 Black / African vs Other |    -0.009     -0.009     -0.001      0.004      0.006      0.010
                  p-value |     0.661      0.653      0.649      0.658      0.659      0.654

ologit: Changes in Pr(y) | Number of obs = 935

Expression: Pr(choice_logit_send), predict(outcome())

                                 |        R0        R20        R40        R60        R80       R100
---------------------------------+------------------------------------------------------------------
financial situation 3 cat        |
      Neutral vs V Broke / Broke |     0.004      0.004      0.000     -0.002     -0.003     -0.004
                         p-value |     0.851      0.852      0.857      0.852      0.852      0.851
 Good / V Good vs V Broke / Br~e |    -0.044     -0.051     -0.015      0.020      0.031      0.059
                         p-value |     0.081      0.125      0.306      0.083      0.114      0.150
        Good / V Good vs Neutral |    -0.048     -0.056     -0.016      0.022      0.033      0.064
                         p-value |     0.062      0.114      0.300      0.073      0.102      0.132

ologit: Changes in Pr(y) | Number of obs = 935

Expression: Pr(choice_logit_send), predict(outcome())

             |        R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
no risky c~s |
         +SD |    -0.005     -0.008     -0.005      0.001      0.005      0.012
     p-value |     0.592      0.523      0.427      0.729      0.499      0.490

ologit: Changes in Pr(y) | Number of obs = 935

Expression: Pr(choice_logit_send), predict(outcome())

             |        R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
amount sen~g |
         +SD |    -0.052     -0.067     -0.032      0.015      0.037      0.099
     p-value |     0.000      0.000      0.000      0.062      0.000      0.000
```

```
ologit: Changes in Pr(y) | Number of obs = 935

Expression: Pr(choice_logit_send), predict(outcome())

                                 |         R0        R20        R40        R60        R80       R100
---------------------------------+------------------------------------------------------------------
base info to                     |
 2. Tr, Ts, D, R vs 1. Ts, Tr,~R |      0.026      0.028      0.006     -0.013     -0.017     -0.031
                         p-value |      0.267      0.278      0.384      0.273      0.275      0.279
 3. D, Ts, Tr, R vs 1. Ts, Tr,~R |     -0.001     -0.001     -0.000      0.000      0.000      0.001
                         p-value |      0.981      0.981      0.981      0.981      0.981      0.981
 4. D, Tr, Ts, R vs 1. Ts, Tr,~R |      0.043      0.044      0.006     -0.021     -0.026     -0.045
                         p-value |      0.138      0.126      0.375      0.143      0.143      0.119
 3. D, Ts, Tr, R vs 2. Tr, Ts,~R |     -0.027     -0.029     -0.006      0.013      0.017      0.031
                         p-value |      0.310      0.326      0.444      0.312      0.316      0.337
 4. D, Tr, Ts, R vs 2. Tr, Ts,~R |      0.017      0.015      0.000     -0.008     -0.009     -0.015
                         p-value |      0.567      0.558      0.907      0.567      0.567      0.554
 4. D, Tr, Ts, R vs 3. D, Ts, ~R |      0.044      0.044      0.006     -0.021     -0.027     -0.046
                         p-value |      0.168      0.164      0.423      0.170      0.172      0.170

.
. /*------------------------------------ End of SECTION 1 ------------------------------------*/
.
.
```

```
. /**********************************************************************/
. /*  SECTION 2: JM comparison
>     Notes: Statistical comparisons of JM data with proportions returned
>     in our experiment. */
. /**********************************************************************/
.
. do JMcomparison.do

.
. // KS test of equality of the two emperical distributions
. // Visually obvious, but nice to have
. ksmirnov percent_returned, by(dat) exact

Two-sample Kolmogorov-Smirnov test for equality of distribution functions

Smaller group             D     p-value      Exact
--------------------------------------------------
0                    0.0536       0.503
1                   -0.6540       0.000
Combined K-S         0.6540       0.000      0.000

Note: Ties exist in combined dataset;
      there are 69 unique values out of 1077 observations.

.
. // ES test that the distribution functions of the two ///
. // independent samples are identical
. escftest percent_returned, group(dat)

Epps-Singleton Two-Sample Empirical Characteristic Function test

Sample sizes: dat = 0          940
              dat = 1          137
              total           1077
t1                           0.400
t2                           0.800

Critical value for W2 at 10%     7.779
                          5%     9.488
                          1%    13.277
Test statistic W2              906.479

Ho: distributions are identical
P-value                    0.00000

.
.
. /*------------------------------------ End of SECTION 2 ------------------------------------*/
.
.
```

```
. /**********************************************************************/
. /*  SECTION 3: Analyses of TG Amount Returned data
>     Notes: Fractional logistic regression of proportion of amounts
>     returned. */
. /**********************************************************************/
.
. /*----------------------------------------------------*/
.    /* [>   3.1.  Baseline Treatment   <] */
. /*----------------------------------------------------*/
.
. * Set estimation variables
. /* [> choice_order represents amount sent where 1 = R20, 2 = R40, etc. <] */
. local est_vars "c.age i.male i.black_african i.financial_situation_3_cat
c.amount_sent_dg##c.amount_sent_dg i.base_info_t
> o i.choice_order"

.
. /* [> Fractional logistic regrestion with clustered standard errors to allow
> for heteroscedasticity across individuals <] */
.
. fracreg logit choice_logit_return_prop `est_vars' if treatment == 1, ///
> vce(cluster subjectid)

Iteration 0:   log pseudolikelihood = -649.31952
Iteration 1:   log pseudolikelihood = -633.27817
Iteration 2:   log pseudolikelihood =   -633.256
Iteration 3:   log pseudolikelihood =   -633.256

Fractional logistic regression                          Number of obs =    935
                                                        Wald chi2(14) =  23.46
                                                        Prob > chi2   = 0.0532
Log pseudolikelihood = -633.256                         Pseudo R2     = 0.0125

                                                 (Std. err. adjusted for 187 clusters in subjectid)
---------------------------------------------------------------------------------------------------
                                  |               Robust
         choice_logit_return_prop | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
----------------------------------+----------------------------------------------------------------
                              age |  -.0367598   .0245853    -1.50   0.135    -.0849462    .0114265
                           1.male |   .0798147   .1045985     0.76   0.445    -.1251945     .284824
                                  |
                    black_african |
                 Black / African  |    .176732   .1132456     1.56   0.119    -.0452252    .3986892
                                  |
        financial_situation_3_cat |
                         Neutral  |  -.0415138   .1298485    -0.32   0.749    -.2960121    .2129845
                   Good / V Good  |   .1239433   .1368335     0.91   0.365    -.1442454     .392132
                                  |
                   amount_sent_dg |   .0010885   .0054305     0.20   0.841    -.0095551    .0117322
                                  |
c.amount_sent_dg#c.amount_sent_dg |   .0000599   .0000611     0.98   0.327      -.00006    .0001797
                                  |
                     base_info_to |
                 2. Tr, Ts, D, R  |   .1146874   .1412293     0.81   0.417     -.162117    .3914918
                 3. D, Ts, Tr, R  |    .356087   .1461253     2.44   0.015     .0696867    .6424874
                 4. D, Tr, Ts, R  |   .2816343   .1413461     1.99   0.046     .0046009    .5586676
                                  |
                     choice_order |
                               2  |  -.0257798     .05258    -0.49   0.624    -.1288347    .0772752
                               3  |  -.0539291   .0509288    -1.06   0.290    -.1537477    .0458895
                               4  |  -.0092162    .061035    -0.15   0.880    -.1288426    .1104102
                               5  |  -.0117949   .0604357    -0.20   0.845    -.1302467     .106657
                                  |
                            _cons |   .5893954   .5612868     1.05   0.294    -.5107065    1.689497
---------------------------------------------------------------------------------------------------

.
```

```
. /* All AMEs in the model with only SD change reported for continuous
> variables - this takes (some) time */
. local est_vars_no_factor "age male black_african financial_situation_3_cat amount_sent_dg
base_info_to choice_order"

. foreach var of varlist `est_vars_no_factor' {
  2.     mchange `var', amount(sd) brief
  3. }

fracreg: Changes in Margin | Number of obs = 935

Expression: Conditional mean of choice_logit_return_prop, predict()

             |    Change    p-value
-------------+----------------------
age          |
         +SD |    -0.025      0.135

fracreg: Changes in Margin | Number of obs = 935

Expression: Conditional mean of choice_logit_return_prop, predict()

             |    Change    p-value
-------------+----------------------
male         |
      1 vs 0 |     0.019      0.446

fracreg: Changes in Margin | Number of obs = 935

Expression: Conditional mean of choice_logit_return_prop, predict()

                          |    Change    p-value
--------------------------+----------------------
black african             |
 Black / African vs Other |     0.043      0.119

fracreg: Changes in Margin | Number of obs = 935

Expression: Conditional mean of choice_logit_return_prop, predict()

                                 |    Change    p-value
---------------------------------+----------------------
financial situation 3 cat        |
      Neutral vs V Broke / Broke |    -0.010      0.749
 Good / V Good vs V Broke / Br~e |     0.030      0.365
        Good / V Good vs Neutral |     0.040      0.190

fracreg: Changes in Margin | Number of obs = 935

Expression: Conditional mean of choice_logit_return_prop, predict()

             |    Change    p-value
-------------+----------------------
amount sen~g |
         +SD |     0.037      0.002

fracreg: Changes in Margin | Number of obs = 935

Expression: Conditional mean of choice_logit_return_prop, predict()

                                 |    Change    p-value
---------------------------------+----------------------
base info to                     |
 2. Tr, Ts, D, R vs 1. Ts, Tr,~R |     0.028      0.416
 3. D, Ts, Tr, R vs 1. Ts, Tr,~R |     0.086      0.014
 4. D, Tr, Ts, R vs 1. Ts, Tr,~R |     0.069      0.045
 3. D, Ts, Tr, R vs 2. Tr, Ts,~R |     0.058      0.092
 4. D, Tr, Ts, R vs 2. Tr, Ts,~R |     0.041      0.216
 4. D, Tr, Ts, R vs 3. D, Ts, ~R |    -0.018      0.590
```

```
fracreg: Changes in Margin | Number of obs = 935

Expression: Conditional mean of choice_logit_return_prop, predict()

             |    Change    p-value
-------------+---------------------
choice order |
      2 vs 1 |    -0.006      0.624
      3 vs 1 |    -0.013      0.290
      4 vs 1 |    -0.002      0.880
      5 vs 1 |    -0.003      0.845
      3 vs 2 |    -0.007      0.532
      4 vs 2 |     0.004      0.736
      5 vs 2 |     0.003      0.788
      4 vs 3 |     0.011      0.290
      5 vs 3 |     0.010      0.401
      5 vs 4 |    -0.001      0.951

.
.
```

```
. /*----------------------------------------------------*/
.    /* [>   3.2.  Baseline and Info Treatments   <] */
. /*----------------------------------------------------*/
.
. /* [> ROBUSTNESS CHECK <] */
. /* [> choice_order represents amount sent where 1 = R20, 2 = R40, etc. <] */
. local est_vars "c.age i.male i.black_african i.financial_situation_3_cat
c.amount_sent_dg##c.amount_sent_dg i.base_info_t
> o i.choice_order i.treatment"

.
. /* [> Fractional logistic regrestion with clustered standard errors to allow
> for heteroscedasticity across individuals <] */
.
. fracreg logit choice_logit_return_prop `est_vars', vce(cluster subjectid)

Iteration 0:   log pseudolikelihood = -968.99706
Iteration 1:   log pseudolikelihood = -940.68221
Iteration 2:   log pseudolikelihood = -940.64033
Iteration 3:   log pseudolikelihood = -940.64033

Fractional logistic regression                          Number of obs =  1,385
                                                        Wald chi2(15) =  31.39
                                                        Prob > chi2   = 0.0078
Log pseudolikelihood = -940.64033                       Pseudo R2     = 0.0134

                                                 (Std. err. adjusted for 277 clusters in subjectid)
---------------------------------------------------------------------------------------------------
                                  |               Robust
         choice_logit_return_prop | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
----------------------------------+----------------------------------------------------------------
                              age |  -.0302432    .020923    -1.45   0.148    -.0712515     .010765
                           1.male |   .0387296   .0913756     0.42   0.672    -.1403633    .2178225
                                  |
                    black_african |
                 Black / African  |   .0740215   .0959812     0.77   0.441    -.1140982    .2621412
                                  |
        financial_situation_3_cat |
                         Neutral  |  -.0738096   .1076457    -0.69   0.493    -.2847913    .1371721
                   Good / V Good  |   .0521192   .1310213     0.40   0.691    -.2046779    .3089162
                                  |
                   amount_sent_dg |    .004472   .0051665     0.87   0.387     -.005654    .0145981
                                  |
c.amount_sent_dg#c.amount_sent_dg |   .0000396   .0000605     0.66   0.512    -.0000789    .0001581
                                  |
                     base_info_to |
                 2. Tr, Ts, D, R  |   .0499298   .1334806     0.37   0.708    -.2116874    .3115471
                 3. D, Ts, Tr, R  |   .3451537   .1470596     2.35   0.019     .0569222    .6333852
                 4. D, Tr, Ts, R  |   .3709723   .1371772     2.70   0.007     .1021099    .6398346
                                  |
                     choice_order |
                               2  |  -.0519349   .0439802    -1.18   0.238    -.1381344    .0342647
                               3  |  -.0445299    .040818    -1.09   0.275    -.1245317     .035472
                               4  |  -.0185762   .0485819    -0.38   0.702     -.113795    .0766426
                               5  |   .0148861   .0497557     0.30   0.765    -.0826333    .1124056
                                  |
                        treatment |
                  3. Information  |   -.154951    .109899    -1.41   0.159    -.3703491    .0604472
                            _cons |   .4902225   .4761534     1.03   0.303    -.4430211    1.423466
---------------------------------------------------------------------------------------------------

.
```

```
. /* All AMEs in the model with only SD change reported for continuous
> variables - this takes (some) time */
. local est_vars_no_factor "age male black_african financial_situation_3_cat amount_sent_dg
base_info_to choice_order"

. foreach var of varlist `est_vars_no_factor' {
  2.     mchange `var', amount(sd) brief
  3. }

fracreg: Changes in Margin | Number of obs = 1385

Expression: Conditional mean of choice_logit_return_prop, predict()

             |    Change    p-value
-------------+----------------------
age          |
         +SD |    -0.019      0.149

fracreg: Changes in Margin | Number of obs = 1385

Expression: Conditional mean of choice_logit_return_prop, predict()

             |    Change    p-value
-------------+----------------------
male         |
      1 vs 0 |     0.009      0.672

fracreg: Changes in Margin | Number of obs = 1385

Expression: Conditional mean of choice_logit_return_prop, predict()

                          |    Change    p-value
--------------------------+----------------------
black african             |
 Black / African vs Other |     0.018      0.441

fracreg: Changes in Margin | Number of obs = 1385

Expression: Conditional mean of choice_logit_return_prop, predict()

                                 |    Change    p-value
---------------------------------+----------------------
financial situation 3 cat        |
      Neutral vs V Broke / Broke |    -0.018      0.492
 Good / V Good vs V Broke / Br~e |     0.013      0.691
        Good / V Good vs Neutral |     0.031      0.280

fracreg: Changes in Margin | Number of obs = 1385

Expression: Conditional mean of choice_logit_return_prop, predict()

             |    Change    p-value
-------------+----------------------
amount sen~g |
         +SD |     0.046      0.000

fracreg: Changes in Margin | Number of obs = 1385

Expression: Conditional mean of choice_logit_return_prop, predict()

                                 |    Change    p-value
---------------------------------+----------------------
base info to                     |
 2. Tr, Ts, D, R vs 1. Ts, Tr,~R |     0.012      0.708
 3. D, Ts, Tr, R vs 1. Ts, Tr,~R |     0.084      0.018
 4. D, Tr, Ts, R vs 1. Ts, Tr,~R |     0.091      0.007
 3. D, Ts, Tr, R vs 2. Tr, Ts,~R |     0.072      0.031
 4. D, Tr, Ts, R vs 2. Tr, Ts,~R |     0.078      0.003
 4. D, Tr, Ts, R vs 3. D, Ts, ~R |     0.006      0.850
```

```
fracreg: Changes in Margin | Number of obs = 1385

Expression: Conditional mean of choice_logit_return_prop, predict()

             |    Change    p-value
-------------+----------------------
choice order |
      2 vs 1 |    -0.013      0.238
      3 vs 1 |    -0.011      0.275
      4 vs 1 |    -0.005      0.702
      5 vs 1 |     0.004      0.765
      3 vs 2 |     0.002      0.845
      4 vs 2 |     0.008      0.393
      5 vs 2 |     0.016      0.123
      4 vs 3 |     0.006      0.454
      5 vs 3 |     0.014      0.131
      5 vs 4 |     0.008      0.337

.
. margins treatment, post

Predictive margins                                      Number of obs = 1,385
Model VCE: Robust

Expression: Conditional mean of choice_logit_return_prop, predict()

---------------------------------------------------------------------------------
                |            Delta-method
                |     Margin   std. err.      z    P>|z|     [95% conf. interval]
----------------+----------------------------------------------------------------
      treatment |
   1. Baseline  |   .5610278   .0125972    44.54   0.000     .5363377    .5857179
3. Information  |   .5232458   .0224407    23.32   0.000     .4792628    .5672288
---------------------------------------------------------------------------------

. test 1.treatment == 3.treatment

 ( 1)  1bn.treatment - 3.treatment = 0

           chi2(  1) =    1.98
         Prob > chi2 =    0.1590

.
.
. /*------------------------------------ End of SECTION 3 ------------------------------------*/
.
.
```

```
. /**********************************************************************/
. /*  SECTION 4: Beliefs
>     Notes: Beliefs estimated with risk preferences and comparisons
>     of beliefs with actual returns. */
. /**********************************************************************/
.

Amount sent: R20

Structural Estimates


                                                        Number of obs = 10,706
                                                        Wald chi2(0)  =      .
Log pseudolikelihood = -6305.7816                       Prob > chi2   =      .

                            (Std. err. adjusted for 106 clusters in subjectid)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
r            |
       _cons |   1.336165   .0698942    19.12   0.000     1.199175    1.473156
-------------+----------------------------------------------------------------
phi          |
       _cons |   1.021261   .0589964    17.31   0.000     .9056299    1.136892
-------------+----------------------------------------------------------------
eta          |
       _cons |   .9828443   .0451896    21.75   0.000     .8942743    1.071414
-------------+----------------------------------------------------------------
noise        |
       _cons |   .1620843   .0100683    16.10   0.000     .1423509    .1818178
-------------+----------------------------------------------------------------
b1           |
       _cons |     1.0975   .1996134     5.50   0.000     .7062652    1.488735
-------------+----------------------------------------------------------------
b2           |
       _cons |   1.829841   .2040076     8.97   0.000     1.429994    2.229689
-------------+----------------------------------------------------------------
b3           |
       _cons |   1.518752   .1897591     8.00   0.000     1.146831    1.890673
------------------------------------------------------------------------------

Beliefs about Amount sent: R20

------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
        bin1 |   .2025309   .0176734    11.46   0.000     .1678918    .2371701
        bin2 |   .4212531   .0187412    22.48   0.000     .3845211    .4579852
        bin3 |   .3086305   .0202485    15.24   0.000     .2689441    .3483169
        bin4 |   .0675854   .0118845     5.69   0.000     .0442922    .0908786
        mean |   24.82541   .8410963    29.52   0.000     23.17689    26.47393
          sd |   17.00931   .4738677    35.89   0.000     16.08055    17.93808
------------------------------------------------------------------------------

Beliefs and Actual Returns for Amount sent: R20

------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       bin1m |   .1546586   .0176734     8.75   0.000     .1200195    .1892977
       bin2m |   .1872106   .0187412     9.99   0.000     .1504785    .2239426
       bin3m |  -.3828589   .0202485   -18.91   0.000    -.4225453   -.3431725
       bin4m |   .0409897   .0118845     3.45   0.001     .0176965    .0642829
       worse |   .1546586   .0176734     8.75   0.000     .1200195    .1892977
      asgood |  -.1956483   .0208725    -9.37   0.000    -.2365577   -.1547389
      better |   .0409897   .0118845     3.45   0.001     .0176965    .0642829
------------------------------------------------------------------------------
```

```
Amount sent: R40

Structural Estimates


                                                        Number of obs = 10,706
                                                        Wald chi2(0)  =      .
Log pseudolikelihood = -6359.5131                       Prob > chi2   =      .

                            (Std. err. adjusted for 106 clusters in subjectid)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
r            |
       _cons |   1.336412   .0698652    19.13   0.000     1.199479    1.473345
-------------+----------------------------------------------------------------
phi          |
       _cons |   1.022043    .058902    17.35   0.000     .9065976    1.137489
-------------+----------------------------------------------------------------
eta          |
       _cons |   .9822725   .0452951    21.69   0.000     .8934956    1.071049
-------------+----------------------------------------------------------------
noise        |
       _cons |   .1621216   .0100614    16.11   0.000     .1424017    .1818416
-------------+----------------------------------------------------------------
b1           |
       _cons |   2.105965   .2617793     8.04   0.000     1.592887    2.619043
-------------+----------------------------------------------------------------
b2           |
       _cons |   2.239249   .2343698     9.55   0.000     1.779893    2.698606
-------------+----------------------------------------------------------------
b3           |
       _cons |   2.646714   .2461756    10.75   0.000     2.164218    3.129209
-------------+----------------------------------------------------------------
b4           |
       _cons |   2.553632     .23914    10.68   0.000     2.084926    3.022338
-------------+----------------------------------------------------------------
b5           |
       _cons |   2.343972   .2345365     9.99   0.000     1.884289    2.803655
-------------+----------------------------------------------------------------
b6           |
       _cons |   .6104992   .1800104     3.39   0.001     .2576854     .963313
------------------------------------------------------------------------------

Beliefs about Amount sent: R40

------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
        bin1 |   .1420633   .0176922     8.03   0.000     .1073872    .1767393
        bin2 |    .162318   .0136958    11.85   0.000     .1354748    .1891612
        bin3 |   .2439642   .0157917    15.45   0.000     .2130131    .2749154
        bin4 |   .2222804   .0157372    14.12   0.000      .191436    .2531248
        bin5 |   .1802383   .0189873     9.49   0.000      .143024    .2174527
        bin6 |   .0318427   .0056032     5.68   0.000     .0208606    .0428247
        bin7 |   .0172931   .0037803     4.57   0.000     .0098839    .0247023
        mean |   46.02026   1.708377    26.94   0.000      42.6719    49.36861
          sd |   29.30167   .7406937    39.56   0.000     27.84994     30.7534
------------------------------------------------------------------------------
```

Beliefs and Actual Returns for Amount sent: R40

```
------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       bin1m |   .1101484   .0176922     6.23   0.000     .0754723    .1448245
       bin2m |   .1038074   .0136958     7.58   0.000     .0769642    .1306506
       bin3m |   .1216238   .0157917     7.70   0.000     .0906726     .152575
       bin4m |   .0148336   .0157372     0.94   0.346    -.0160108     .045678
       bin5m |  -.3197617   .0189873   -16.84   0.000     -.356976   -.2825473
       bin6m |  -.0213488   .0056032    -3.81   0.000    -.0323308   -.0103668
       bin7m |  -.0093026   .0037803    -2.46   0.014    -.0167119   -.0018934
       worse |   .2139557   .0249099     8.59   0.000     .1651332    .2627783
      asgood |  -.1833043   .0254684    -7.20   0.000    -.2332215   -.1333871
      better |  -.0306515   .0084384    -3.63   0.000    -.0471905   -.0141125
------------------------------------------------------------------------------
```

Amount sent: R60

Structural Estimates

```
                                                        Number of obs = 10,706
                                                        Wald chi2(0)  =      .
Log pseudolikelihood = -6397.2897                       Prob > chi2   =      .

                            (Std. err. adjusted for 106 clusters in subjectid)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
r            |
       _cons |   1.337031   .0700236    19.09   0.000     1.199787    1.474274
-------------+----------------------------------------------------------------
phi          |
       _cons |   1.024121   .0593567    17.25   0.000     .9077839    1.140458
-------------+----------------------------------------------------------------
eta          |
       _cons |   .9819407   .0452485    21.70   0.000     .8932552    1.070626
-------------+----------------------------------------------------------------
noise        |
       _cons |   .1622542   .0101016    16.06   0.000     .1424553     .182053
-------------+----------------------------------------------------------------
b1           |
       _cons |   2.264498   .2807323     8.07   0.000     1.714273    2.814724
-------------+----------------------------------------------------------------
b2           |
       _cons |   1.982239    .263651     7.52   0.000     1.465493    2.498986
-------------+----------------------------------------------------------------
b3           |
       _cons |   2.277888   .2656753     8.57   0.000     1.757174    2.798603
-------------+----------------------------------------------------------------
b4           |
       _cons |   2.548679   .2648908     9.62   0.000     2.029503    3.067856
-------------+----------------------------------------------------------------
b5           |
       _cons |   2.505778    .266761     9.39   0.000     1.982936    3.028619
-------------+----------------------------------------------------------------
b6           |
       _cons |    2.42881   .2644271     9.19   0.000     1.910542    2.947077
-------------+----------------------------------------------------------------
b7           |
       _cons |    2.56764   .2642199     9.72   0.000     2.049779    3.085502
-------------+----------------------------------------------------------------
b8           |
       _cons |   .8096963   .2201319     3.68   0.000     .3782457    1.241147
-------------+----------------------------------------------------------------
b9           |
       _cons |   .2671808   .2612627     1.02   0.306    -.2448847    .7792462
------------------------------------------------------------------------------
```

```
Beliefs about Amount sent: R60

------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
        bin1 |   .1194055   .0173904     6.87   0.000      .085321      .15349
        bin2 |   .0900411   .0102403     8.79   0.000     .0699705    .1101117
        bin3 |   .1210151   .0126541     9.56   0.000     .0962135    .1458168
        bin4 |   .1586509   .0131903    12.03   0.000     .1327983    .1845035
        bin5 |   .1519884     .01304    11.66   0.000     .1264304    .1775465
        bin6 |   .1407291   .0132733    10.60   0.000     .1147138    .1667444
        bin7 |   .1616878   .0186595     8.67   0.000     .1251159    .1982597
        bin8 |   .0278748   .0047777     5.83   0.000     .0185106     .037239
        bin9 |   .0162032   .0042618     3.80   0.000     .0078502    .0245562
       bin10 |   .0124041   .0029474     4.21   0.000     .0066272     .018181
        mean |   70.52272   2.645249    26.66   0.000     65.33813    75.70731
          sd |   43.57398   1.179842    36.93   0.000     41.26153    45.88643
------------------------------------------------------------------------------

Beliefs and Actual Returns for Amount sent: R60

------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       bin1m |   .0768523   .0173904     4.42   0.000     .0427678    .1109368
       bin2m |   .0740837   .0102403     7.23   0.000     .0540131    .0941543
       bin3m |   .0944194   .0126541     7.46   0.000     .0696177     .119221
       bin4m |   .0629062   .0131903     4.77   0.000     .0370536    .0887588
       bin5m |   .0243289     .01304     1.87   0.062    -.0012292    .0498869
       bin6m |  -.0454411   .0132733    -3.42   0.001    -.0714564   -.0194258
       bin7m |  -.2744824   .0186595   -14.71   0.000    -.3110543   -.2379105
       bin8m |   .0012791   .0047777     0.27   0.789    -.0080851    .0106433
       bin9m |  -.0050734   .0042618    -1.19   0.234    -.0134264    .0032796
      bin10m |  -.0088725   .0029474    -3.01   0.003    -.0146494   -.0030956
       worse |   .2453553   .0283324     8.66   0.000     .1898248    .3008859
      asgood |  -.2326885   .0288682    -8.06   0.000    -.2892691   -.1761078
      better |  -.0126669   .0092991    -1.36   0.173    -.0308928    .0055591
------------------------------------------------------------------------------
```

```
Amount sent: R80

Structural Estimates


                                                        Number of obs = 10,706
                                                        Wald chi2(0)  =      .
Log pseudolikelihood = -6420.3185                       Prob > chi2   =      .

                            (Std. err. adjusted for 106 clusters in subjectid)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
r            |
       _cons |   1.336672   .0699611    19.11   0.000     1.199551    1.473793
-------------+----------------------------------------------------------------
phi          |
       _cons |   1.024167   .0594638    17.22   0.000     .9076205    1.140714
-------------+----------------------------------------------------------------
eta          |
       _cons |   .9826117   .0458275    21.44   0.000     .8927915    1.072432
-------------+----------------------------------------------------------------
noise        |
       _cons |    .162279   .0101095    16.05   0.000     .1424646    .1820933
-------------+----------------------------------------------------------------
b1           |
       _cons |   2.354716   .3690796     6.38   0.000     1.631333    3.078099
-------------+----------------------------------------------------------------
b2           |
       _cons |   1.929488   .3469506     5.56   0.000     1.249478    2.609499
-------------+----------------------------------------------------------------
b3           |
       _cons |   1.937731   .3467751     5.59   0.000     1.258065    2.617398
-------------+----------------------------------------------------------------
b4           |
       _cons |   1.789776   .3362716     5.32   0.000     1.130695    2.448856
-------------+----------------------------------------------------------------
b5           |
       _cons |   2.703571   .3570993     7.57   0.000     2.003669    3.403473
-------------+----------------------------------------------------------------
b6           |
       _cons |   2.529668   .3456388     7.32   0.000     1.852228    3.207107
-------------+----------------------------------------------------------------
b7           |
       _cons |   2.488342   .3322012     7.49   0.000     1.837239    3.139444
-------------+----------------------------------------------------------------
b8           |
       _cons |   2.322104   .3325289     6.98   0.000     1.670359    2.973848
-------------+----------------------------------------------------------------
b9           |
       _cons |   2.785509   .3499596     7.96   0.000     2.099601    3.471418
-------------+----------------------------------------------------------------
b10          |
       _cons |    .609739   .2533326     2.41   0.016     .1132162    1.106262
-------------+----------------------------------------------------------------
b11          |
       _cons |   .4423715   .3015111     1.47   0.142    -.1485794    1.033322
-------------+----------------------------------------------------------------
b12          |
       _cons |   .1114407   .2317615     0.48   0.631    -.3428036    .5656849
------------------------------------------------------------------------------
```

Beliefs about Amount sent: R80

```
------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
        bin1 |   .1034934   .0160853     6.43   0.000     .0719669      .13502
        bin2 |   .0676455   .0087785     7.71   0.000     .0504399    .0848511
        bin3 |   .0682054   .0094406     7.22   0.000     .0497021    .0867087
        bin4 |   .0588251    .007192     8.18   0.000     .0447291     .072921
        bin5 |   .1466961   .0157023     9.34   0.000     .1159201    .1774721
        bin6 |   .1232802     .01276     9.66   0.000     .0982711    .1482894
        bin7 |   .1182894   .0110905    10.67   0.000     .0965523    .1400264
        bin8 |   .1001727   .0109933     9.11   0.000     .0786263    .1217191
        bin9 |   .1592224   .0190133     8.37   0.000     .1219569    .1964878
       bin10 |    .018075   .0037098     4.87   0.000     .0108039    .0253462
       bin11 |   .0152895   .0044867     3.41   0.001     .0064956    .0240833
       bin12 |   .0109817   .0027724     3.96   0.000      .005548    .0164154
       bin13 |   .0098236   .0031095     3.16   0.002     .0037291    .0159182
        mean |   96.45387   3.630713    26.57   0.000     89.33781    103.5699
          sd |     57.041   1.790752    31.85   0.000     53.53119    60.55081
------------------------------------------------------------------------------
```

Beliefs and Actual Returns for Amount sent: R80

```
------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       bin1m |   .0609402   .0160853     3.79   0.000     .0294137    .0924668
       bin2m |   .0410497   .0087785     4.68   0.000     .0238441    .0582554
       bin3m |   .0416096   .0094406     4.41   0.000     .0231063    .0601129
       bin4m |   .0269102    .007192     3.74   0.000     .0128142    .0410061
       bin5m |   .1041429   .0157023     6.63   0.000     .0733669    .1349189
       bin6m |    .080727     .01276     6.33   0.000     .0557179    .1057362
       bin7m |   .0119064   .0110905     1.07   0.283    -.0098306    .0336434
       bin8m |  -.0274869   .0109933    -2.50   0.012    -.0490333   -.0059405
       bin9m |  -.2769479   .0190133   -14.57   0.000    -.3142133   -.2396824
      bin10m |  -.0404356   .0037098   -10.90   0.000    -.0477068   -.0331645
      bin11m |  -.0059871   .0044867    -1.33   0.182     -.014781    .0028067
      bin12m |   .0003434   .0027724     0.12   0.901    -.0050903    .0057771
      bin13m |  -.0167721   .0031095    -5.39   0.000    -.0228667   -.0106775
       worse |   .1705098   .0289408     5.89   0.000     .1137868    .2272328
      asgood |  -.1076584   .0296426    -3.63   0.000    -.1657569   -.0495599
      better |  -.0628514   .0115021    -5.46   0.000    -.0853951   -.0403077
------------------------------------------------------------------------------
```

```
Amount sent: R100

Structural Estimates


                                                        Number of obs = 10,706
                                                        Wald chi2(0)  =      .
Log pseudolikelihood = -6438.2444                       Prob > chi2   =      .

                            (Std. err. adjusted for 106 clusters in subjectid)
------------------------------------------------------------------------------
             |               Robust
             | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
r            |
       _cons |   1.332628   .0689828    19.32   0.000     1.197424    1.467832
-------------+----------------------------------------------------------------
phi          |
       _cons |   1.025481   .0597086    17.17   0.000     .9084541    1.142507
-------------+----------------------------------------------------------------
eta          |
       _cons |   .9871497   .0464913    21.23   0.000     .8960284    1.078271
-------------+----------------------------------------------------------------
noise        |
       _cons |   .1625325   .0101395    16.03   0.000     .1426594    .1824055
-------------+----------------------------------------------------------------
b1           |
       _cons |   2.322042   .3484999     6.66   0.000     1.638995    3.005089
-------------+----------------------------------------------------------------
b2           |
       _cons |   1.454355   .3478892     4.18   0.000     .7725046    2.136205
-------------+----------------------------------------------------------------
b3           |
       _cons |   1.438206   .3657868     3.93   0.000     .7212768    2.155135
-------------+----------------------------------------------------------------
b4           |
       _cons |   1.526805   .3576052     4.27   0.000     .8259117    2.227698
-------------+----------------------------------------------------------------
b5           |
       _cons |    1.59729   .3475209     4.60   0.000     .9161611    2.278418
-------------+----------------------------------------------------------------
b6           |
       _cons |   2.614859   .3331306     7.85   0.000     1.961935    3.267782
-------------+----------------------------------------------------------------
b7           |
       _cons |   2.019497   .3330993     6.06   0.000     1.366635     2.67236
-------------+----------------------------------------------------------------
b8           |
       _cons |   2.141705     .34631     6.18   0.000     1.462949     2.82046
-------------+----------------------------------------------------------------
b9           |
       _cons |   1.814094   .3325929     5.45   0.000     1.162224    2.465964
-------------+----------------------------------------------------------------
b10          |
       _cons |   1.803504   .3327832     5.42   0.000     1.151261    2.455747
-------------+----------------------------------------------------------------
b11          |
       _cons |   2.691968    .318909     8.44   0.000     2.066918    3.317018
-------------+----------------------------------------------------------------
b12          |
       _cons |   .4978091   .4010207     1.24   0.214    -.2881771    1.283795
-------------+----------------------------------------------------------------
b13          |
       _cons |   .2302355   .5256217     0.44   0.661    -.7999641    1.260435
-------------+----------------------------------------------------------------
b14          |
       _cons |  -.0095105   .6012059    -0.02   0.987    -1.187852    1.168831
-------------+----------------------------------------------------------------
b15          |
       _cons |   -.616978    .247343    -2.49   0.013    -1.101761   -.1321946
------------------------------------------------------------------------------
```

```
Beliefs about Amount sent: R100

------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
        bin1 |   .1128561   .0171836     6.57   0.000     .0791768    .1465353
        bin2 |   .0473907   .0067226     7.05   0.000     .0342146    .0605668
        bin3 |   .0466315   .0080009     5.83   0.000     .0309501     .062313
        bin4 |   .0509516    .007727     6.59   0.000     .0358069    .0660963
        bin5 |   .0546725   .0068736     7.95   0.000     .0412006    .0681445
        bin6 |   .1512494   .0137179    11.03   0.000     .1243628    .1781359
        bin7 |   .0833934   .0084982     9.81   0.000     .0667372    .1000496
        bin8 |   .0942335    .011678     8.07   0.000      .071345    .1171221
        bin9 |   .0679088   .0077174     8.80   0.000      .052783    .0830345
       bin10 |   .0671934   .0084413     7.96   0.000     .0506488     .083738
       bin11 |   .1633735   .0191073     8.55   0.000     .1259239    .2008231
       bin12 |   .0182084   .0051232     3.55   0.000     .0081671    .0282496
       bin13 |   .0139337   .0065543     2.13   0.034     .0010875    .0267798
       bin14 |   .0109634   .0059025     1.86   0.063    -.0006054    .0225321
       bin15 |   .0059721   .0015317     3.90   0.000     .0029699    .0089742
       bin16 |   .0110681   .0034107     3.25   0.001     .0043833     .017753
        mean |   119.3967   4.828596    24.73   0.000     109.9329    128.8606
          sd |   72.60476   2.245172    32.34   0.000     68.20431    77.00522
------------------------------------------------------------------------------

Beliefs and Actual Returns for Amount sent: R100

------------------------------------------------------------------------------
             | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       bin1m |   .0490263   .0171836     2.85   0.004      .015347    .0827055
       bin2m |   .0367524   .0067226     5.47   0.000     .0235763    .0499285
       bin3m |   .0359932   .0080009     4.50   0.000     .0203118    .0516747
       bin4m |   .0456325    .007727     5.91   0.000     .0304878    .0607771
       bin5m |   .0493534   .0068736     7.18   0.000     .0358814    .0628253
       bin6m |   .0714621   .0137179     5.21   0.000     .0445756    .0983487
       bin7m |   .0621168   .0084982     7.31   0.000     .0454606     .078773
       bin8m |   .0516803    .011678     4.43   0.000     .0287918    .0745689
       bin9m |  -.0065593   .0077174    -0.85   0.395    -.0216851    .0085664
      bin10m |  -.0338704   .0084413    -4.01   0.000     -.050415   -.0173258
      bin11m |  -.3579031   .0191073   -18.73   0.000    -.3953526   -.3204535
      bin12m |  -.0030682   .0051232    -0.60   0.549    -.0131095     .006973
      bin13m |   .0032954   .0065543     0.50   0.615    -.0095508    .0161415
      bin14m |   .0056442   .0059025     0.96   0.339    -.0059245     .017213
      bin15m |   .0006529   .0015317     0.43   0.670    -.0023493    .0036551
      bin16m |  -.0102084   .0034107    -2.99   0.003    -.0168933   -.0035236
       worse |   .2167578   .0300988     7.20   0.000     .1577651    .2757504
      asgood |  -.2130736    .029902    -7.13   0.000    -.2716805   -.1544667
      better |  -.0036842    .013479    -0.27   0.785    -.0301026    .0227343
------------------------------------------------------------------------------

. 
. /*------------------------------------ End of SECTION 4 ------------------------------------*/
. 
. 
. 
```

```
. /*********************************************************************/
. /*  SECTION 5: Ordered and Multinomial logit analysis of amount sent
>     Notes: These models compare the Baseline Treatment with the
>     Social History Treatment. */
. /*********************************************************************/
.
. /*---------------------------------------------------*/
.    /* [>   5.1.  Ordered Logit   <] */
. /*---------------------------------------------------*/
.
. /* [> Primary ordered logit with treatment and task order (to) variables <] */
.
. * Set estimation variables
. local est_vars "c.age i.male i.black_african i.financial_situation_3_cat
c.no_risky_choices##c.no_risky_choices c.amount_
> sent_dg##c.amount_sent_dg i.base_info_to i.treatment"

.
. * Estimate ordered logit
. ologit choice_logit_send `est_vars', cluster(subjectid)

Iteration 0:   log pseudolikelihood = -2428.2301
Iteration 1:   log pseudolikelihood = -2287.3423
Iteration 2:   log pseudolikelihood = -2283.6976
Iteration 3:   log pseudolikelihood = -2283.6733
Iteration 4:   log pseudolikelihood = -2283.6733

Ordered logistic regression                            Number of obs =  1,385
                                                       Wald chi2(13) =  93.36
                                                       Prob > chi2   = 0.0000
Log pseudolikelihood = -2283.6733                      Pseudo R2     = 0.0595

                                                     (Std. err. adjusted for 277 clusters in subjectid)
-------------------------------------------------------------------------------------------------------
                                      |               Robust
                    choice_logit_send | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
--------------------------------------+----------------------------------------------------------------
                                  age |  -.0969325   .0312383    -3.10   0.002    -.1581584   -.0357066
                               1.male |  -.5104467   .1932314    -2.64   0.008    -.8891733     -.13172
                                      |
                        black_african |
                     Black / African  |  -.0886744   .1887613    -0.47   0.639    -.4586397    .2812909
                                      |
            financial_situation_3_cat |
                             Neutral  |  -.0209983   .1986815    -0.11   0.916     -.410407    .3684103
                       Good / V Good  |   .3480058   .2546673     1.37   0.172    -.1511328    .8471445
                                      |
                     no_risky_choices |   .0109884   .0250476     0.44   0.661     -.038104    .0600809
                                      |
c.no_risky_choices#c.no_risky_choices |     -.0001   .0002322    -0.43   0.667    -.0005552    .0003551
                                      |
                       amount_sent_dg |   .0137265    .010429     1.32   0.188    -.0067139     .034167
                                      |
    c.amount_sent_dg#c.amount_sent_dg |    .000163   .0001167     1.40   0.163    -.0000658    .0003918
                                      |
                         base_info_to |
                     2. Tr, Ts, D, R  |  -.4822832   .2467751    -1.95   0.051    -.9659534    .0013871
                     3. D, Ts, Tr, R  |   .0148267   .2994562     0.05   0.961    -.5720967    .6017501
                     4. D, Tr, Ts, R  |  -.2205951   .2641105    -0.84   0.404    -.7382423     .297052
                                      |
                            treatment |
                      3. Information  |   .6634948   .2073232     3.20   0.001     .2571487    1.069841
--------------------------------------+----------------------------------------------------------------
                                /cut1 |  -3.983759   1.072785                     -6.086379   -1.881139
                                /cut2 |  -2.486452   1.063731                     -4.571325   -.4015777
                                /cut3 |  -1.388417   1.058001                     -3.462061    .6852275
                                /cut4 |  -.5311779   1.055301                      -2.59953    1.537174
                                /cut5 |   .2919157   1.051342                     -1.768677    2.352508
-------------------------------------------------------------------------------------------------------

.
```

```
.  /* All AMEs in the model with only SD change reported for continuous
> variables - this takes (some) time */
. local est_vars_no_factor "age male black_african financial_situation_3_cat no_risky_choices
amount_sent_dg base_info_to t
> reatment"

. foreach var of varlist `est_vars_no_factor' {
  2.     mchange `var', amount(sd) brief
  3. }

ologit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
age          |
         +SD |      0.023      0.026      0.004     -0.011     -0.014     -0.028
     p-value |      0.008      0.001      0.032      0.006      0.002      0.001

ologit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
male         |
      1 vs 0 |      0.040      0.055      0.017     -0.019     -0.030     -0.064
     p-value |      0.004      0.010      0.066      0.007      0.010      0.014

ologit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

                          |         R0        R20        R40        R60        R80       R100
--------------------------+------------------------------------------------------------------
black african             |
 Black / African vs Other |      0.007      0.009      0.002     -0.003     -0.005     -0.011
                  p-value |      0.634      0.641      0.651      0.636      0.638      0.641

ologit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

                                 |         R0        R20        R40        R60        R80       R100
---------------------------------+------------------------------------------------------------------
financial situation 3 cat        |
      Neutral vs V Broke / Broke |      0.002      0.002      0.000     -0.001     -0.001     -0.002
                         p-value |      0.916      0.916      0.916      0.916      0.916      0.916
 Good / V Good vs V Broke / Br~e |     -0.027     -0.037     -0.012      0.012      0.020      0.045
                         p-value |      0.145      0.173      0.276      0.139      0.172      0.193
        Good / V Good vs Neutral |     -0.029     -0.039     -0.013      0.012      0.021      0.047
                         p-value |      0.138      0.170      0.265      0.140      0.167      0.183

ologit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
no risky c~s |
         +SD |      0.000      0.001      0.001      0.000     -0.000     -0.001
     p-value |      0.992      0.956      0.828      0.988      0.963      0.929

ologit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
amount sen~g |
         +SD |     -0.039     -0.063     -0.035      0.006      0.028      0.103
     p-value |      0.000      0.000      0.000      0.243      0.000      0.000
```

```
ologit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

                                 |        R0         R20         R40         R60         R80        R100
---------------------------------+-----------------------------------------------------------------
base info to                     |
 2. Tr, Ts, D, R vs 1. Ts, Tr,~R |     0.041       0.051       0.012      -0.018      -0.027      -0.059
                         p-value |     0.044       0.049       0.171       0.030       0.047       0.067
 3. D, Ts, Tr, R vs 1. Ts, Tr,~R |    -0.001      -0.002      -0.001       0.000       0.001       0.002
                         p-value |     0.960       0.961       0.961       0.960       0.960       0.961
 4. D, Tr, Ts, R vs 1. Ts, Tr,~R |     0.017       0.023       0.008      -0.007      -0.012      -0.029
                         p-value |     0.400       0.403       0.435       0.395       0.407       0.410
 3. D, Ts, Tr, R vs 2. Tr, Ts,~R |    -0.042      -0.052      -0.013       0.019       0.028       0.061
                         p-value |     0.066       0.080       0.230       0.044       0.072       0.109
 4. D, Tr, Ts, R vs 2. Tr, Ts,~R |    -0.024      -0.027      -0.004       0.011       0.015       0.030
                         p-value |     0.215       0.217       0.330       0.202       0.208       0.232
 4. D, Tr, Ts, R vs 3. D, Ts, ~R |     0.018       0.025       0.008      -0.007      -0.013      -0.031
                         p-value |     0.417       0.425       0.470       0.404       0.425       0.438

ologit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

                               |        R0         R20         R40         R60         R80        R100
-------------------------------+-----------------------------------------------------------------
treatment                      |
 3. Information vs 1. Baseline |    -0.051      -0.071      -0.022       0.023       0.037       0.085
                       p-value |     0.001       0.001       0.031       0.001       0.001       0.004

.
.
```

```
. /* [> ROBUSTNESS CHECK <] */
. /* [> Secondary ordered logit with with the interaction of treatment and task
> order, but only using the task orders common across the two treatments. This
> can only estimate with common task orders or there are empty interactions <] */
.
. * Set estimation variables
. local est_vars "c.age i.male i.black_african i.financial_situation_3_cat
c.no_risky_choices##c.no_risky_choices c.amount_
> sent_dg##c.amount_sent_dg i.base_info_to##i.treatment "


.
. * Estimate ordered logit
. ologit choice_logit_send `est_vars' if (base_info_to == 2 | base_info_to == 4), ///
> cluster(subjectid)

Iteration 0:   log pseudolikelihood = -1643.5395
Iteration 1:   log pseudolikelihood = -1530.7341
Iteration 2:   log pseudolikelihood = -1527.6031
Iteration 3:   log pseudolikelihood = -1527.5618
Iteration 4:   log pseudolikelihood = -1527.5618

Ordered logistic regression                         Number of obs =    945
                                                    Wald chi2(12) =  64.49
                                                    Prob > chi2   = 0.0000
Log pseudolikelihood = -1527.5618                   Pseudo R2     = 0.0706

                                                (Std. err. adjusted for 189 clusters in subjectid)
-----------------------------------------------------------------------------------------------------
                                     |               Robust
                   choice_logit_send | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------------------------------+---------------------------------------------------------------
                                 age |  -.1047518   .0402853    -2.60   0.009    -.1837094   -.0257941
                              1.male |  -.6403433   .2485312    -2.58   0.010    -1.127456    -.153231
                                     |
                       black_african |
                     Black / African |  -.1435703   .2340778    -0.61   0.540    -.6023544    .3152138
                                     |
           financial_situation_3_cat |
                             Neutral |  -.0601629   .2355725    -0.26   0.798    -.5218765    .4015507
                         Good / V Good |   .1472504   .3112315     0.47   0.636    -.4627522     .757253
                                     |
                    no_risky_choices |   .0208475   .0273528     0.76   0.446     -.032763    .0744581
                                     |
c.no_risky_choices#c.no_risky_choices |  -.0002068   .0002346    -0.88   0.378    -.0006667    .0002531
                                     |
                      amount_sent_dg |   .0072798   .0136072     0.53   0.593    -.0193898    .0339494
                                     |
    c.amount_sent_dg#c.amount_sent_dg |   .0002703   .0001592     1.70   0.090    -.0000417    .0005824
                                     |
                        base_info_to |
                     4. D, Tr, Ts, R |  -.2619538    .275665    -0.95   0.342    -.8022473    .2783398
                                     |
                           treatment |
                      3. Information |   .1904875   .2482777     0.77   0.443    -.2961278    .6771028
                                     |
              base_info_to#treatment |
     4. D, Tr, Ts, R#3. Information |   1.070433   .4456259     2.40   0.016      .197022    1.943843
-------------------------------------+---------------------------------------------------------------
                               /cut1 |  -4.058513   1.312676                      -6.631311   -1.485715
                               /cut2 |  -2.400168   1.296649                      -4.941554    .1412175
                               /cut3 |  -1.287008   1.291701                      -3.818695    1.244679
                               /cut4 |  -.4726767    1.28902                      -2.999109    2.053756
                               /cut5 |   .3242032   1.279621                      -2.183807    2.832214
-----------------------------------------------------------------------------------------------------

.
```

```
.  /* All AMEs in the model with only SD change reported for continuous
> variables - this takes (some) time */
. local est_vars_no_factor "age male black_african financial_situation_3_cat no_risky_choices
amount_sent_dg base_info_to t
> reatment"

. foreach var of varlist `est_vars_no_factor' {
  2.     mchange `var', amount(sd) brief
  3. }

ologit: Changes in Pr(y) | Number of obs = 945

Expression: Pr(choice_logit_send), predict(outcome())

             |        R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
age          |
         +SD |     0.022      0.028      0.002     -0.010     -0.013     -0.028
     p-value |     0.027      0.006      0.345      0.019      0.011      0.007

ologit: Changes in Pr(y) | Number of obs = 945

Expression: Pr(choice_logit_send), predict(outcome())

             |        R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
male         |
      1 vs 0 |     0.047      0.073      0.017     -0.022     -0.034     -0.081
     p-value |     0.004      0.012      0.141      0.007      0.014      0.019

ologit: Changes in Pr(y) | Number of obs = 945

Expression: Pr(choice_logit_send), predict(outcome())

                          |        R0        R20        R40        R60        R80       R100
--------------------------+------------------------------------------------------------------
black african             |
 Black / African vs Other |     0.011      0.016      0.003     -0.005     -0.007     -0.017
                  p-value |     0.529      0.545      0.586      0.539      0.542      0.544

ologit: Changes in Pr(y) | Number of obs = 945

Expression: Pr(choice_logit_send), predict(outcome())

                                 |        R0        R20        R40        R60        R80       R100
---------------------------------+------------------------------------------------------------------
financial situation 3 cat        |
      Neutral vs V Broke / Broke |     0.005      0.007      0.001     -0.002     -0.003     -0.007
                         p-value |     0.798      0.799      0.802      0.798      0.799      0.799
 Good / V Good vs V Broke / Br~e |    -0.011     -0.016     -0.003      0.005      0.008      0.018
                         p-value |     0.628      0.638      0.680      0.627      0.636      0.643
        Good / V Good vs Neutral |    -0.016     -0.023     -0.004      0.007      0.011      0.025
                         p-value |     0.496      0.518      0.597      0.496      0.516      0.525

ologit: Changes in Pr(y) | Number of obs = 945

Expression: Pr(choice_logit_send), predict(outcome())

             |        R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
no risky c~s |
         +SD |     0.002      0.004      0.001     -0.001     -0.002     -0.005
     p-value |     0.756      0.662      0.264      0.742      0.691      0.637

ologit: Changes in Pr(y) | Number of obs = 945

Expression: Pr(choice_logit_send), predict(outcome())

             |        R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
amount sen~g |
         +SD |    -0.037     -0.067     -0.032      0.005      0.023      0.107
     p-value |     0.001      0.000      0.000      0.389      0.005      0.000
```

```
ologit: Changes in Pr(y) | Number of obs = 945

Expression: Pr(choice_logit_send), predict(outcome())

                                 |         R0        R20        R40        R60        R80       R100
---------------------------------+------------------------------------------------------------------
base info to                     |
 4. D, Tr, Ts, R vs 2. Tr, Ts,~R |     -0.007     -0.030     -0.023      0.002      0.016      0.043
                         p-value |      0.673      0.214      0.042      0.784      0.158      0.148

ologit: Changes in Pr(y) | Number of obs = 945

Expression: Pr(choice_logit_send), predict(outcome())

                               |         R0        R20        R40        R60        R80       R100
-------------------------------+------------------------------------------------------------------
treatment                      |
 3. Information vs 1. Baseline |     -0.054     -0.086     -0.017      0.028      0.039      0.090
                       p-value |      0.002      0.000      0.112      0.004      0.000      0.002

.
.
```

```
. /* [> ROBUSTNESS CHECK <] */
. /* [> Primary ordered logit but only focussing on FIRST of the five choices <] */
. 
. * Set estimation variables
. local est_vars "c.age i.male i.black_african i.financial_situation_3_cat
c.no_risky_choices##c.no_risky_choices c.amount_
> sent_dg##c.amount_sent_dg i.base_info_to i.treatment"

. 
. * Estimate ordered logit
. ologit choice_logit_send `est_vars' if choice_order == 1, cluster(subjectid)

Iteration 0:   log pseudolikelihood = -467.37259
Iteration 1:   log pseudolikelihood = -428.85248
Iteration 2:   log pseudolikelihood = -427.51837
Iteration 3:   log pseudolikelihood = -427.50587
Iteration 4:   log pseudolikelihood = -427.50587

Ordered logistic regression                             Number of obs =    277
                                                        Wald chi2(13) =  67.66
                                                        Prob > chi2   = 0.0000
Log pseudolikelihood = -427.50587                       Pseudo R2     = 0.0853

                                                     (Std. err. adjusted for 277 clusters in subjectid)
-------------------------------------------------------------------------------------------------------
                                      |               Robust
                    choice_logit_send | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
--------------------------------------+----------------------------------------------------------------
                                  age |  -.1640375   .0511434    -3.21   0.001    -.2642767   -.0637984
                               1.male |  -.5215235   .2549772    -2.05   0.041     -1.02127   -.0217774
                                      |
                        black_african |
                     Black / African  |  -.1159646   .2494423    -0.46   0.642    -.6048626    .3729334
                                      |
            financial_situation_3_cat |
                             Neutral  |  -.1041595   .2576113    -0.40   0.686    -.6090683    .4007494
                       Good / V Good  |   .4643375   .3318563     1.40   0.162     -.186089    1.114764
                                      |
                     no_risky_choices |   .0334388   .0357551     0.94   0.350      -.03664    .1035175
                                      |
c.no_risky_choices#c.no_risky_choices |  -.0003367   .0003203    -1.05   0.293    -.0009645    .0002911
                                      |
                       amount_sent_dg |   .0188687   .0127575     1.48   0.139    -.0061356    .0438729
                                      |
    c.amount_sent_dg#c.amount_sent_dg |   .0001444   .0001371     1.05   0.292    -.0001244    .0004131
                                      |
                         base_info_to |
                     2. Tr, Ts, D, R  |  -.4917522   .3267391    -1.51   0.132    -1.132149    .1486447
                     3. D, Ts, Tr, R  |   .1620627    .359852     0.45   0.652    -.5432343    .8673597
                     4. D, Tr, Ts, R  |    -.32711   .3541723    -0.92   0.356    -1.021275    .3670549
                                      |
                            treatment |
                      3. Information  |    1.01706    .279843     3.63   0.000     .4685777    1.565542
--------------------------------------+----------------------------------------------------------------
                                /cut1 |  -5.122586    1.50642                     -8.075115   -2.170057
                                /cut2 |  -3.098253   1.475698                     -5.990567    -.205939
                                /cut3 |   -1.78348   1.465997                     -4.656781    1.089821
                                /cut4 |  -.8242146    1.45042                     -3.666986    2.018557
                                /cut5 |  -.2318023   1.445702                     -3.065327    2.601722
-------------------------------------------------------------------------------------------------------

. 
. 
```

```
.  /* All AMEs in the model with only SD change reported for continuous
> variables - this takes (some) time */
. local est_vars_no_factor "age male black_african financial_situation_3_cat no_risky_choices
amount_sent_dg base_info_to t
> reatment"

. foreach var of varlist `est_vars_no_factor' {
  2.     mchange `var', amount(sd) brief
  3. }

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
age          |
         +SD |     0.033      0.050     -0.003     -0.023     -0.015     -0.040
     p-value |     0.008      0.001      0.521      0.005      0.004      0.000

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
male         |
      1 vs 0 |     0.033      0.065      0.008     -0.026     -0.020     -0.059
     p-value |     0.030      0.046      0.359      0.051      0.067      0.050

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

                          |         R0        R20        R40        R60        R80       R100
--------------------------+------------------------------------------------------------------
black african             |
 Black / African vs Other |     0.008      0.014      0.001     -0.006     -0.004     -0.013
                  p-value |     0.636      0.643      0.711      0.640      0.642      0.645

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

                                 |         R0        R20        R40        R60        R80       R100
---------------------------------+------------------------------------------------------------------
financial situation 3 cat        |
      Neutral vs V Broke / Broke |     0.008      0.013     -0.000     -0.006     -0.004     -0.011
                         p-value |     0.687      0.686      0.939      0.686      0.688      0.686
 Good / V Good vs V Broke / Br~e |    -0.027     -0.057     -0.010      0.021      0.018      0.055
                         p-value |     0.135      0.160      0.399      0.146      0.173      0.190
        Good / V Good vs Neutral |    -0.035     -0.070     -0.010      0.027      0.022      0.066
                         p-value |     0.080      0.097      0.410      0.085      0.115      0.123

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
no risky c~s |
         +SD |     0.004      0.009      0.002     -0.003     -0.003     -0.008
     p-value |     0.646      0.531      0.203      0.593      0.536      0.470

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
amount sen~g |
         +SD |    -0.035     -0.084     -0.032      0.020      0.024      0.108
     p-value |     0.000      0.000      0.000      0.025      0.002      0.000
```

```
ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

                                  |        R0         R20         R40         R60         R80        R100
----------------------------------+--------------------------------------------------------------------
base info to                      |
 2. Tr, Ts, D, R vs 1. Ts, Tr,~R |     0.033       0.059       0.004      -0.024      -0.018      -0.054
                          p-value |     0.129       0.127       0.594       0.093       0.140       0.164
 3. D, Ts, Tr, R vs 1. Ts, Tr,~R |    -0.008      -0.019      -0.005       0.006       0.006       0.021
                          p-value |     0.650       0.653       0.662       0.654       0.652       0.654
 4. D, Tr, Ts, R vs 1. Ts, Tr,~R |     0.021       0.039       0.005      -0.015      -0.012      -0.038
                          p-value |     0.362       0.353       0.504       0.347       0.356       0.367
 3. D, Ts, Tr, R vs 2. Tr, Ts,~R |    -0.041      -0.078      -0.009       0.030       0.024       0.075
                          p-value |     0.054       0.062       0.394       0.032       0.066       0.093
 4. D, Tr, Ts, R vs 2. Tr, Ts,~R |    -0.012      -0.020       0.001       0.009       0.006       0.016
                          p-value |     0.563       0.568       0.731       0.559       0.572       0.575
 4. D, Tr, Ts, R vs 3. D, Ts, ~R |     0.029       0.059       0.010      -0.021      -0.018      -0.059
                          p-value |     0.190       0.189       0.352       0.180       0.187       0.209

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

                                |        R0         R20         R40         R60         R80        R100
--------------------------------+--------------------------------------------------------------------
treatment                       |
 3. Information vs 1. Baseline |    -0.059      -0.128      -0.018       0.048       0.038       0.119
                        p-value |     0.001       0.000       0.214       0.000       0.002       0.002

.
.
```

```
. /* [> ROBUSTNESS CHECK <] */
. /* [> Primary ordered logit but only focussing on LAST of the five choices <] */
. 
. * Set estimation variables
. local est_vars "c.age i.male i.black_african i.financial_situation_3_cat
c.no_risky_choices##c.no_risky_choices c.amount_
> sent_dg##c.amount_sent_dg i.base_info_to i.treatment"

. 
. * Estimate ordered logit
. ologit choice_logit_send `est_vars' if choice_order == 5, cluster(subjectid)

Iteration 0:   log pseudolikelihood = -489.01922
Iteration 1:   log pseudolikelihood = -454.42541
Iteration 2:   log pseudolikelihood = -453.72559
Iteration 3:   log pseudolikelihood = -453.72318
Iteration 4:   log pseudolikelihood = -453.72318

Ordered logistic regression                         Number of obs =    277
                                                    Wald chi2(13) =  61.86
                                                    Prob > chi2   = 0.0000
Log pseudolikelihood = -453.72318                   Pseudo R2     = 0.0722

                                                     (Std. err. adjusted for 277 clusters in subjectid)
-------------------------------------------------------------------------------------------------------
                                      |               Robust
                    choice_logit_send | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
--------------------------------------+----------------------------------------------------------------
                                  age |  -.0932191   .0467593    -1.99   0.046    -.1848655   -.0015726
                               1.male |  -.7345577   .2476473    -2.97   0.003    -1.219938   -.2491779
                                      |
                        black_african |
                     Black / African  |   .1040748   .2480563     0.42   0.675    -.3821066    .5902562
                                      |
            financial_situation_3_cat |
                             Neutral  |   .0176789   .2632877     0.07   0.946    -.4983555    .5337132
                         Good / V Good  |   .2366302    .333116     0.71   0.477    -.4162651    .8895256
                                      |
                      no_risky_choices |  -.0171493   .0260142    -0.66   0.510    -.0681362    .0338376
                                      |
c.no_risky_choices#c.no_risky_choices |   .0001519   .0002469     0.62   0.539    -.0003321    .0006358
                                      |
                       amount_sent_dg |   .0164301   .0125567     1.31   0.191    -.0081805    .0410408
                                      |
    c.amount_sent_dg#c.amount_sent_dg |   .0001732   .0001508     1.15   0.251    -.0001223    .0004688
                                      |
                         base_info_to |
                    2. Tr, Ts, D, R  |  -.8555034   .3125367    -2.74   0.006    -1.468064   -.2429427
                    3. D, Ts, Tr, R  |  -.2358944   .3845668    -0.61   0.540    -.9896315    .5178427
                    4. D, Tr, Ts, R  |  -.5888799   .3327333    -1.77   0.077    -1.241025    .0632655
                                      |
                            treatment |
                      3. Information  |   .8508558   .2720151     3.13   0.002     .3177161    1.383996
--------------------------------------+----------------------------------------------------------------
                                /cut1 |  -4.757263   1.439158                     -7.577961   -1.936566
                                /cut2 |  -3.615136    1.41671                     -6.391837   -.8384343
                                /cut3 |  -2.561386   1.404586                     -5.314324    .1915509
                                /cut4 |  -1.554718   1.396687                     -4.292174    1.182738
                                /cut5 |  -.6765126   1.394884                     -3.410434    2.057409
-------------------------------------------------------------------------------------------------------
```

```
.  /* All AMEs in the model with only SD change reported for continuous
> variables - this takes (some) time */
. local est_vars_no_factor "age male black_african financial_situation_3_cat no_risky_choices
amount_sent_dg base_info_to t
> reatment"

. foreach var of varlist `est_vars_no_factor' {
  2.     mchange `var', amount(sd) brief
  3. }

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
age          |
         +SD |      0.023      0.019      0.008     -0.008     -0.013     -0.029
     p-value |      0.067      0.045      0.034      0.106      0.049      0.036

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
male         |
      1 vs 0 |      0.061      0.059      0.037     -0.016     -0.041     -0.100
     p-value |      0.003      0.005      0.017      0.039      0.004      0.007

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

                          |         R0        R20        R40        R60        R80       R100
--------------------------+------------------------------------------------------------------
black african             |
 Black / African vs Other |     -0.009     -0.008     -0.004      0.003      0.006      0.013
                  p-value |      0.678      0.675      0.671      0.684      0.676      0.673

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

                                 |         R0        R20        R40        R60        R80       R100
---------------------------------+------------------------------------------------------------------
financial situation 3 cat        |
      Neutral vs V Broke / Broke |     -0.002     -0.001     -0.001      0.001      0.001      0.002
                         p-value |      0.946      0.946      0.946      0.947      0.946      0.946
 Good / V Good vs V Broke / Br~e |     -0.020     -0.019     -0.011      0.005      0.013      0.032
                         p-value |      0.456      0.484      0.512      0.450      0.472      0.489
        Good / V Good vs Neutral |     -0.019     -0.017     -0.010      0.005      0.012      0.029
                         p-value |      0.504      0.531      0.554      0.489      0.521      0.535

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
no risky c~s |
         +SD |      0.001     -0.000     -0.001     -0.001     -0.000      0.001
     p-value |      0.949      0.999      0.900      0.771      0.989      0.956

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
amount sen~g |
         +SD |     -0.047     -0.051     -0.044     -0.005      0.027      0.121
     p-value |      0.000      0.000      0.000      0.447      0.002      0.000
```

```
ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

                                  |        R0         R20         R40         R60         R80        R100
----------------------------------+--------------------------------------------------------------------
base info to                      |
 2. Tr, Ts, D, R vs 1. Ts, Tr,~R |     0.073       0.066       0.039      -0.017      -0.044      -0.116
                          p-value |     0.005       0.008       0.029       0.063       0.008       0.010
 3. D, Ts, Tr, R vs 1. Ts, Tr,~R |     0.016       0.018       0.014      -0.000      -0.011      -0.037
                          p-value |     0.556       0.543       0.529       0.962       0.552       0.533
 4. D, Tr, Ts, R vs 1. Ts, Tr,~R |     0.045       0.045       0.031      -0.007      -0.029      -0.085
                          p-value |     0.080       0.077       0.100       0.314       0.083       0.084
 3. D, Ts, Tr, R vs 2. Tr, Ts,~R |    -0.057      -0.048      -0.024       0.017       0.033       0.079
                          p-value |     0.077       0.113       0.214       0.069       0.100       0.138
 4. D, Tr, Ts, R vs 2. Tr, Ts,~R |    -0.027      -0.021      -0.008       0.010       0.015       0.031
                          p-value |     0.316       0.327       0.365       0.317       0.323       0.327
 4. D, Tr, Ts, R vs 3. D, Ts, ~R |     0.029       0.028       0.017      -0.007      -0.018      -0.048
                          p-value |     0.355       0.375       0.421       0.345       0.367       0.393

ologit: Changes in Pr(y) | Number of obs = 277

Expression: Pr(choice_logit_send), predict(outcome())

                                |        R0         R20         R40         R60         R80        R100
--------------------------------+--------------------------------------------------------------------
treatment                       |
 3. Information vs 1. Baseline |    -0.069      -0.068      -0.042       0.017       0.045       0.116
                        p-value |     0.001       0.002       0.020       0.024       0.003       0.005

.
.
```

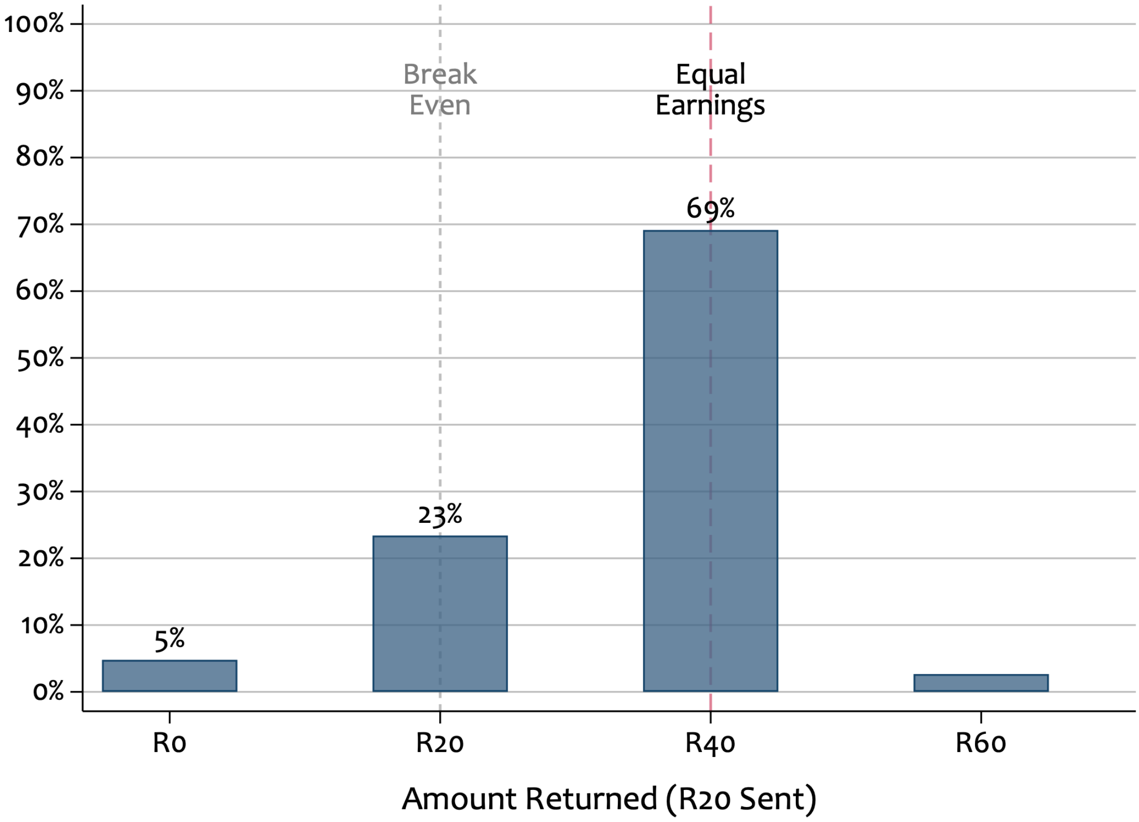
```
. /*---------------------------------------------------*/
.    /* [>   5.2.  Multinomial logit   <] */
. /*---------------------------------------------------*/
.
. /* [> ROBUSTNESS CHECK <] */
. /* [> Primary multinomial logit with treatment and task order (to) variables <] */
.
. * Set estimation variables
. local est_vars "c.age i.male i.black_african i.financial_situation_3_cat
c.no_risky_choices##c.no_risky_choices c.amount_
> sent_dg##c.amount_sent_dg i.base_info_to i.treatment"


.
. * Estimate multinomial logit
. mlogit choice_logit_send `est_vars', base(0) cluster(subjectid)

Iteration 0:   log pseudolikelihood = -2428.2301
Iteration 1:   log pseudolikelihood = -2211.9237
Iteration 2:   log pseudolikelihood = -2194.5178
Iteration 3:   log pseudolikelihood = -2192.9516
Iteration 4:   log pseudolikelihood = -2192.9474
Iteration 5:   log pseudolikelihood = -2192.9474

Multinomial logistic regression                        Number of obs =  1,385
                                                       Wald chi2(65) = 235.12
                                                       Prob > chi2   = 0.0000
Log pseudolikelihood = -2192.9474                      Pseudo R2     = 0.0969

                                                    (Std. err. adjusted for 277 clusters in subjectid)
------------------------------------------------------------------------------------------------------
                                      |               Robust
                    choice_logit_send | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
--------------------------------------+---------------------------------------------------------------
R0                                    |  (base outcome)
--------------------------------------+---------------------------------------------------------------
R20                                   |
                                  age |  -.0344472   .0472814    -0.73   0.466     -.127117    .0582226
                               1.male |   .4233936   .3435768     1.23   0.218    -.2500045    1.096792
                                      |
                        black_african |
                     Black / African  |   .5090397   .3179201     1.60   0.109    -.1140723    1.132152
                                      |
            financial_situation_3_cat |
                             Neutral  |   .2124445   .3760777     0.56   0.572    -.5246543    .9495434
                         Good / V Good  |  -.450396   .3903721    -1.15   0.249    -1.215511    .3147192
                                      |
                     no_risky_choices |  -.0205336   .0415001    -0.49   0.621    -.1018723    .0608051
                                      |
c.no_risky_choices#c.no_risky_choices |    .000236   .0004216     0.56   0.576    -.0005902    .0010623
                                      |
                      amount_sent_dg |   .0623255    .018945     3.29   0.001      .025194     .099457
                                      |
    c.amount_sent_dg#c.amount_sent_dg |  -.0007954   .0002975    -2.67   0.008    -.0013784   -.0002123
                                      |
                         base_info_to |
                     2. Tr, Ts, D, R  |   .4444745   .4997021     0.89   0.374    -.5349235    1.423873
                     3. D, Ts, Tr, R  |  -.0845068   .5610873    -0.15   0.880    -1.184218    1.015204
                     4. D, Tr, Ts, R  |   .1267313   .5206368     0.24   0.808     -.893698    1.147161
                                      |
                            treatment |
                      3. Information  |   .3888047   .3698068     1.05   0.293    -.3360034    1.113613
                                _cons |   .5180218    1.58888     0.33   0.744    -2.596125    3.632169
--------------------------------------+---------------------------------------------------------------
R40                                   |
                                  age |  -.1033599   .0565472    -1.83   0.068    -.2141903    .0074706
                               1.male |   .1872206   .3632476     0.52   0.606    -.5247317    .8991728
                                      |
                        black_african |
                     Black / African  |    .162878   .3130198     0.52   0.603    -.4506296    .7763856
                                      |
            financial_situation_3_cat |
                             Neutral  |  -.1036917   .3627854    -0.29   0.775     -.814738    .6073546
                         Good / V Good  |  -.5409395   .4310451    -1.25   0.209    -1.385772    .3038934
                                      |
                     no_risky_choices |  -.0039439   .0468211    -0.08   0.933    -.0957116    .0878238
                                      |
c.no_risky_choices#c.no_risky_choices |   .0001374   .0004837     0.28   0.776    -.0008107    .0010854
                                      |
                       amount_sent_dg |   .0874581    .016813     5.20   0.000     .0545053    .1204109
                                      |
    c.amount_sent_dg#c.amount_sent_dg |  -.0008512   .0001868    -4.56   0.000    -.0012173   -.0004852
                                      |
                         base_info_to |
                     2. Tr, Ts, D, R  |   .2429378   .4926509     0.49   0.622    -.7226401    1.208516
                     3. D, Ts, Tr, R  |  -.0022487   .5582515    -0.00   0.997    -1.096402    1.091904
                     4. D, Tr, Ts, R  |  -.1660075   .5288546    -0.31   0.754    -1.202543    .8705285
                                      |
                            treatment |
                      3. Information  |   .3465234   .3966929     0.87   0.382    -.4309803    1.124027
                                _cons |   1.748377   1.805191     0.97   0.333    -1.789733    5.286486
--------------------------------------+----------------------------------------------------------------
R60                                   |
                                  age |  -.1457545   .0638177    -2.28   0.022    -.2708348   -.0206742
                               1.male |  -.1601335   .3867743    -0.41   0.679    -.9181972    .5979302
                                      |
                        black_african |
                     Black / African  |   .2669878   .3478154     0.77   0.443    -.4147179    .9486934
                                      |
             financial_situation_3_cat |
                             Neutral  |   .3313576   .3997957     0.83   0.407    -.4522276    1.114943
                       Good / V Good  |   .0887867   .4591907     0.19   0.847    -.8112106    .9887839
                                      |
                     no_risky_choices |   .0167185   .0490987     0.34   0.733    -.0795133    .1129502
                                      |
c.no_risky_choices#c.no_risky_choices |   -.000162   .0004897    -0.33   0.741    -.0011218    .0007979
                                      |
                       amount_sent_dg |   .0783733   .0188851     4.15   0.000     .0413591    .1153874
                                      |
    c.amount_sent_dg#c.amount_sent_dg |  -.0006507   .0002305    -2.82   0.005    -.0011025   -.0001989
                                      |
                         base_info_to |
                     2. Tr, Ts, D, R  |  -.0156624   .5187795    -0.03   0.976    -1.032452    1.001127
                     3. D, Ts, Tr, R  |   .1125988   .5915508     0.19   0.849    -1.046819    1.272017
                     4. D, Tr, Ts, R  |  -.8709779   .5526931    -1.58   0.115    -1.954237    .2122808
                                      |
                            treatment |
                      3. Information  |   .6946655   .4384538     1.58   0.113    -.1646881    1.554019
                                _cons |   2.169555   1.953774     1.11   0.267    -1.659771    5.998882
--------------------------------------+----------------------------------------------------------------
R80                                   |
                                  age |  -.1059953   .0513782    -2.06   0.039    -.2066948   -.0052958
                               1.male |  -.3266801   .3928762    -0.83   0.406    -1.096703    .4433432
                                      |
                        black_african |
                     Black / African  |   .8641287    .368979     2.34   0.019     .1409432    1.587314
                                      |
             financial_situation_3_cat |
                             Neutral  |   .2188562   .4273738     0.51   0.609    -.6187811    1.056494
                       Good / V Good  |   .1800764   .4628014     0.39   0.697    -.7269977     1.08715
                                      |
                     no_risky_choices |   .0129007    .046409     0.28   0.781    -.0780593    .1038608
                                      |
c.no_risky_choices#c.no_risky_choices |  -.0001973   .0004691    -0.42   0.674    -.0011167    .0007221
                                      |
                       amount_sent_dg |    .081435   .0192826     4.22   0.000     .0436418    .1192282
                                      |
    c.amount_sent_dg#c.amount_sent_dg |  -.0006102   .0002049    -2.98   0.003    -.0010117   -.0002086
                                      |
                         base_info_to |
                     2. Tr, Ts, D, R  |  -.6058731   .5677528    -1.07   0.286    -1.718648     .506902
                     3. D, Ts, Tr, R  |  -.6283696   .6074643    -1.03   0.301    -1.818978    .5622386
                     4. D, Tr, Ts, R  |  -.9867562    .535957    -1.84   0.066    -2.037213    .0637003
                                      |
                            treatment |
                      3. Information  |   .7766721   .4610682     1.68   0.092     -.127005    1.680349
                                _cons |   1.205965   1.719638     0.70   0.483    -2.164464    4.576395
--------------------------------------+----------------------------------------------------------------
R100                                  |
                                  age |  -.1921075   .0840642    -2.29   0.022    -.3568704   -.0273446
                               1.male |  -.6077745   .4217054    -1.44   0.150    -1.434302     .218753
                                      |
                        black_african |
                     Black / African  |  -.1087048   .4074371    -0.27   0.790    -.9072669    .6898573
                                      |
            financial_situation_3_cat |
                             Neutral  |  -.0429921   .4665015    -0.09   0.927    -.9573182     .871334
                       Good / V Good  |   .2567678   .5243791     0.49   0.624    -.7709963    1.284532
                                      |
                     no_risky_choices |   .0132514   .0542078     0.24   0.807     -.092994    .1194968
                                      |
c.no_risky_choices#c.no_risky_choices |  -.0000136   .0005416    -0.03   0.980    -.0010751    .0010479
                                      |
                       amount_sent_dg |    .042476   .0211703     2.01   0.045      .000983     .083969
                                      |
    c.amount_sent_dg#c.amount_sent_dg |  -.0000554   .0002216    -0.25   0.802    -.0004899     .000379
                                      |
                         base_info_to |
                     2. Tr, Ts, D, R  |  -.8724918   .6043197    -1.44   0.149    -2.056937     .311953
                     3. D, Ts, Tr, R  |   .2318691   .6627733     0.35   0.726    -1.067143    1.530881
                     4. D, Tr, Ts, R  |  -.2250371    .616841    -0.36   0.715    -1.434023    .9839491
                                      |
                            treatment |
                      3. Information  |    1.64498   .5070404     3.24   0.001     .6511996    2.638761
                                _cons |   3.072127   2.536703     1.21   0.226     -1.89972    8.043975
-------------------------------------------------------------------------------------------------------

.
.  /* All AMEs in the model with only SD change reported for continuous
> variables - this takes (some) time */
. local est_vars_no_factor "age male black_african financial_situation_3_cat no_risky_choices
amount_sent_dg base_info_to t
> reatment"

. foreach var of varlist `est_vars_no_factor' {
  2.     mchange `var', amount(sd) brief
  3. }

mlogit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
age          |
         +SD |      0.023      0.030     -0.004     -0.019     -0.001     -0.028
     p-value |      0.049      0.016      0.773      0.166      0.901      0.101

mlogit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
male         |
      1 vs 0 |      0.001      0.081      0.048     -0.020     -0.033     -0.076
     p-value |      0.979      0.005      0.106      0.465      0.171      0.023

mlogit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

                          |         R0        R20        R40        R60        R80       R100
--------------------------+------------------------------------------------------------------
black african             |
 Black / African vs Other |     -0.026      0.044     -0.029     -0.003      0.068     -0.054
                  p-value |      0.284      0.163      0.319      0.911      0.003      0.109
```

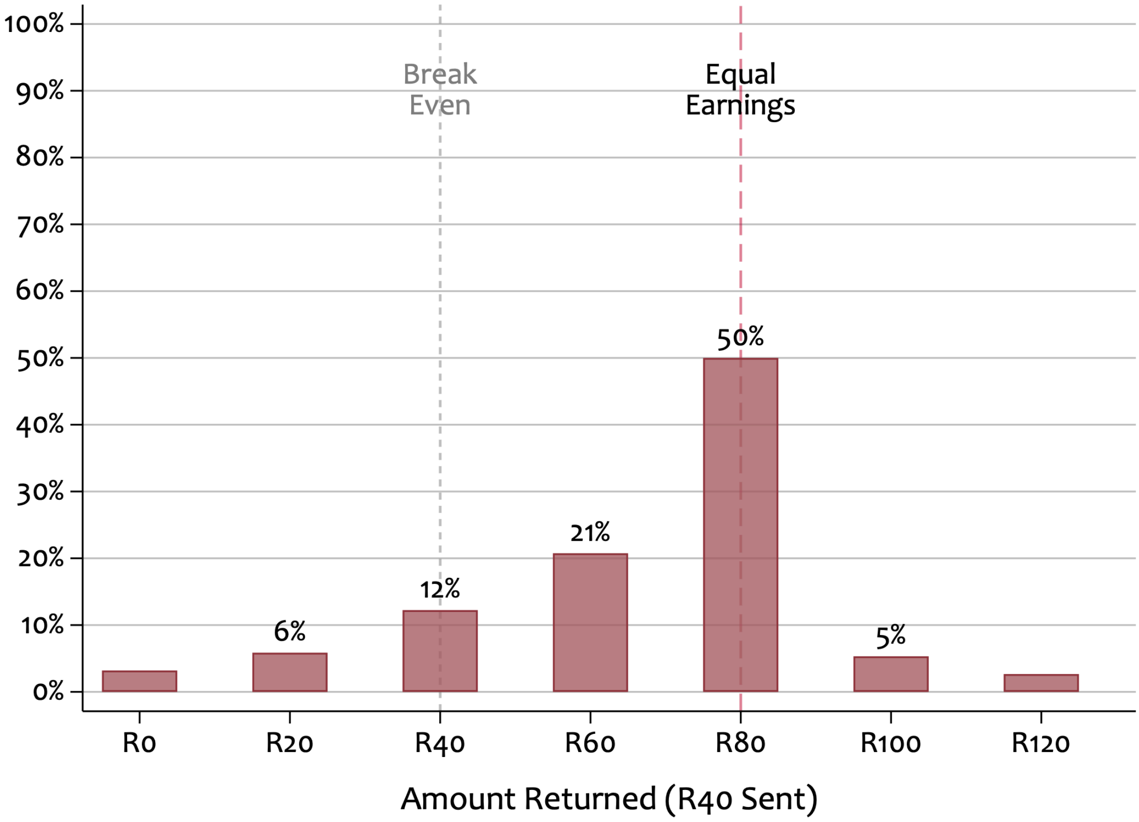
```
mlogit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

                                 |         R0        R20        R40        R60        R80       R100
---------------------------------+------------------------------------------------------------------
financial situation 3 cat        |
      Neutral vs V Broke / Broke |     -0.009      0.024     -0.050      0.039      0.013     -0.018
                         p-value |      0.740      0.517      0.109      0.215      0.621      0.614
 Good / V Good vs V Broke / Br~e |      0.015     -0.052     -0.088      0.035      0.037      0.054
                         p-value |      0.670      0.142      0.019      0.286      0.178      0.229
        Good / V Good vs Neutral |      0.024     -0.076     -0.039     -0.004      0.023      0.072
                         p-value |      0.519      0.045      0.307      0.922      0.432      0.123

mlogit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
no risky c~s |
         +SD |     -0.006      0.003      0.025     -0.015     -0.022      0.015
     p-value |      0.645      0.858      0.281      0.172      0.010      0.463

mlogit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

             |         R0        R20        R40        R60        R80       R100
-------------+------------------------------------------------------------------
amount sen~g |
         +SD |     -0.055     -0.066      0.009      0.018      0.028      0.067
     p-value |      0.000      0.000      0.564      0.187      0.015      0.000

mlogit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

                                 |         R0        R20        R40        R60        R80       R100
---------------------------------+------------------------------------------------------------------
base info to                     |
 2. Tr, Ts, D, R vs 1. Ts, Tr,~R |      0.001      0.084      0.063      0.012     -0.075     -0.085
                         p-value |      0.969      0.034      0.098      0.759      0.078      0.057
 3. D, Ts, Tr, R vs 1. Ts, Tr,~R |      0.003     -0.005      0.008      0.031     -0.084      0.047
                         p-value |      0.933      0.895      0.871      0.556      0.047      0.450
 4. D, Tr, Ts, R vs 1. Ts, Tr,~R |      0.027      0.079      0.035     -0.077     -0.089      0.025
                         p-value |      0.492      0.045      0.397      0.030      0.013      0.630
 3. D, Ts, Tr, R vs 2. Tr, Ts,~R |      0.002     -0.089     -0.055      0.019     -0.009      0.132
                         p-value |      0.956      0.027      0.226      0.719      0.828      0.037
 4. D, Tr, Ts, R vs 2. Tr, Ts,~R |      0.026     -0.005     -0.029     -0.089     -0.013      0.110
                         p-value |      0.421      0.889      0.359      0.001      0.613      0.002
 4. D, Tr, Ts, R vs 3. D, Ts, ~R |      0.024      0.084      0.027     -0.108     -0.005     -0.022
                         p-value |      0.550      0.036      0.589      0.032      0.890      0.747

mlogit: Changes in Pr(y) | Number of obs = 1385

Expression: Pr(choice_logit_send), predict(outcome())

                               |         R0        R20        R40        R60        R80       R100
-------------------------------+------------------------------------------------------------------
treatment                      |
 3. Information vs 1. Baseline |     -0.052     -0.040     -0.064      0.003      0.009      0.144
                       p-value |      0.045      0.190      0.030      0.929      0.734      0.002

.
.
. /*------------------------------------ End of SECTION 5 ------------------------------------*/
```

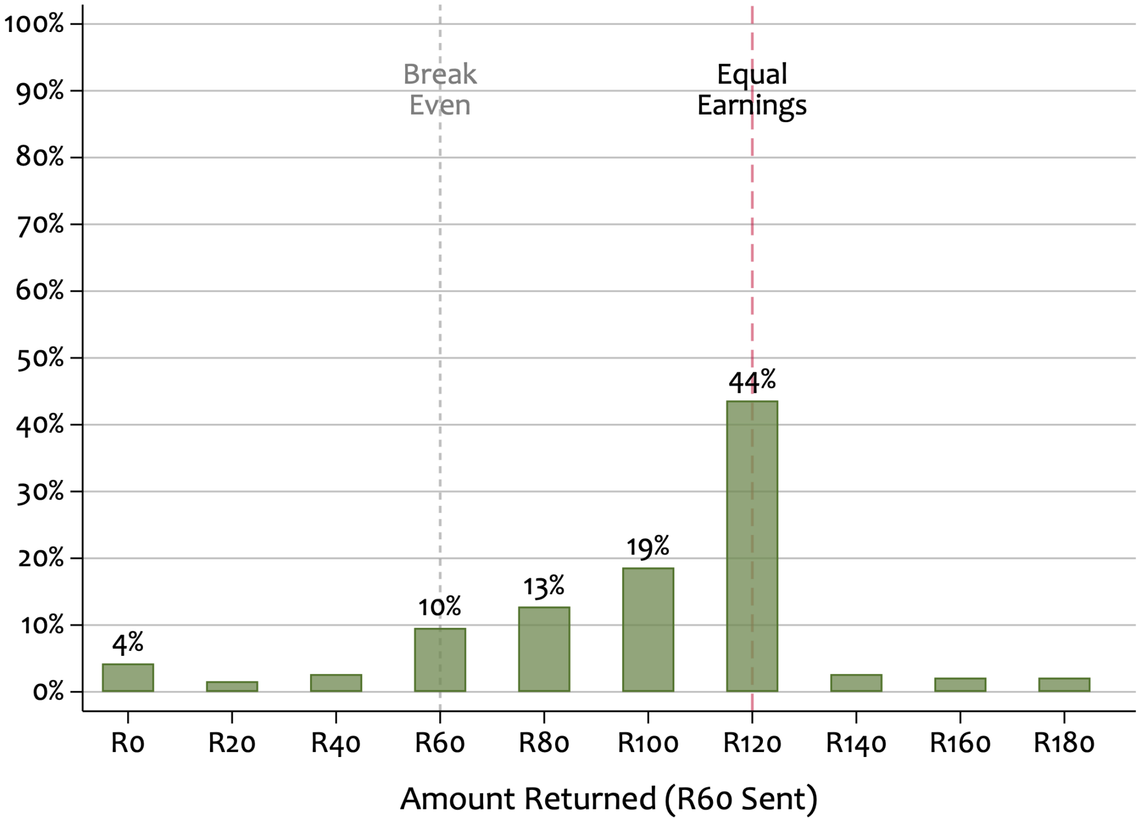
Appendix D: Additional Baseline Treatment Figures

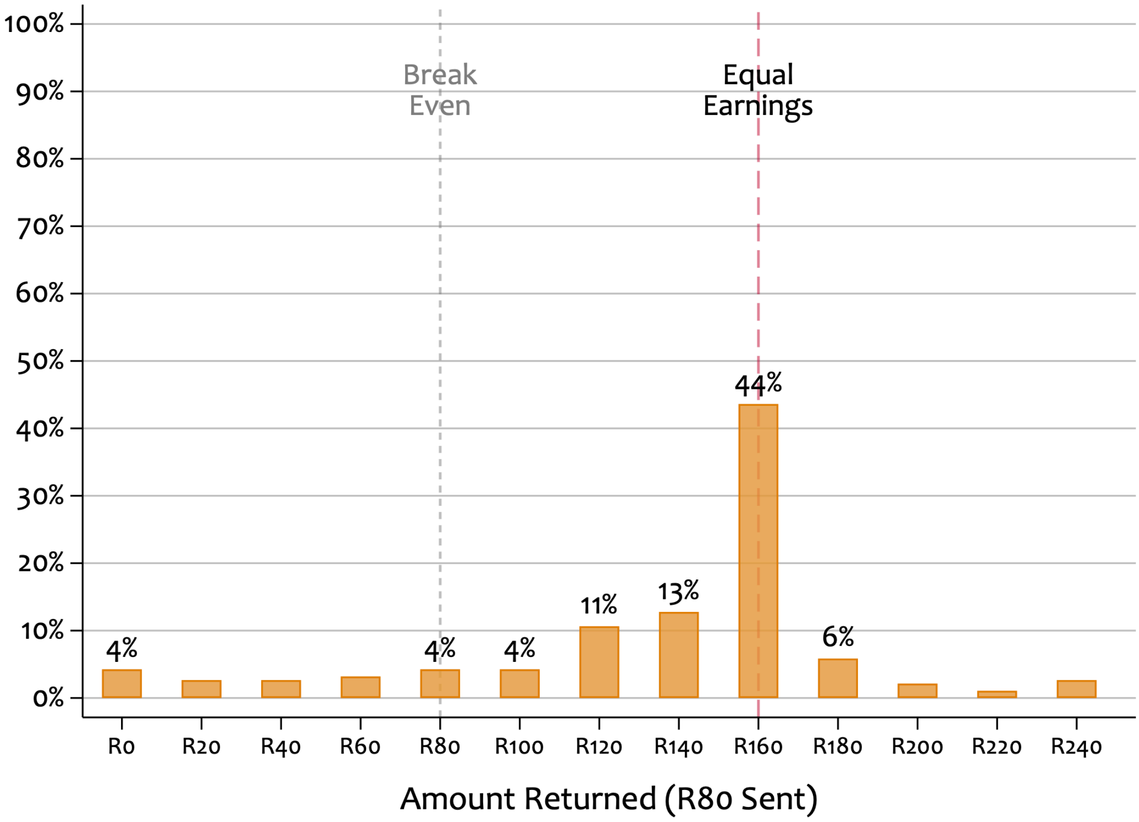
[Online Only]

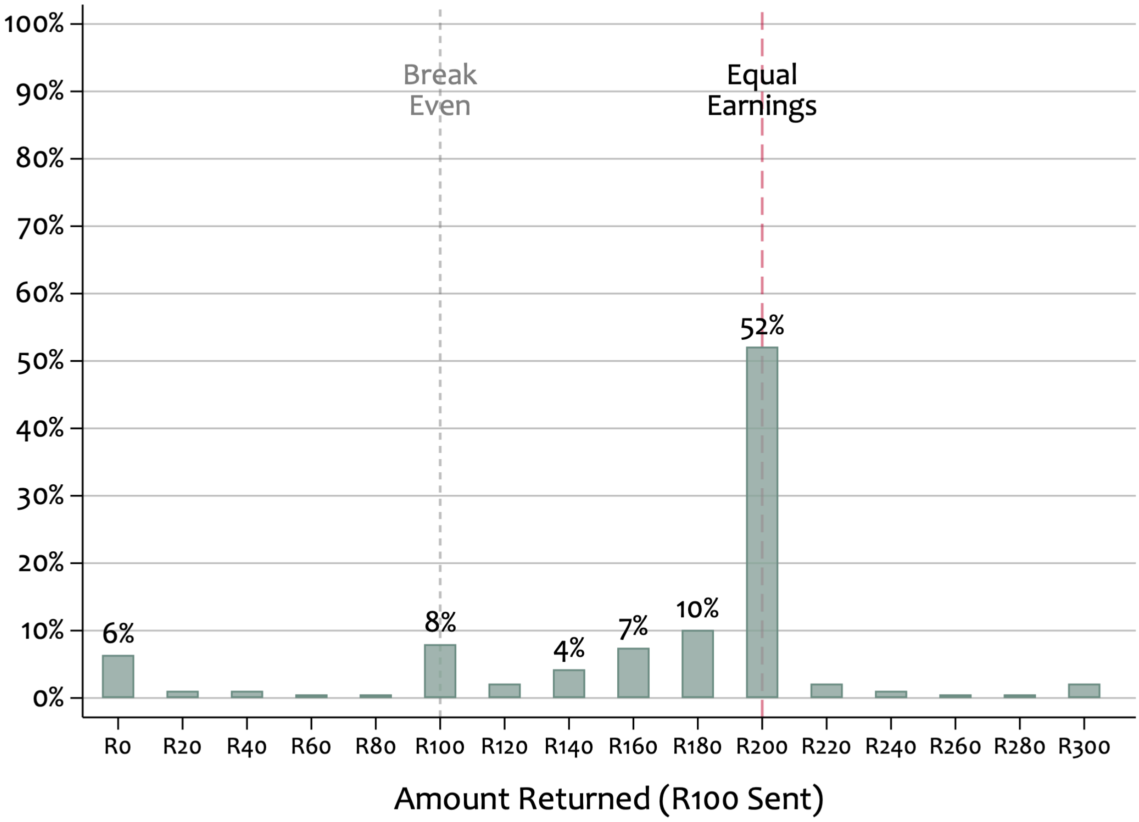
In this appendix, we include figures of amounts returned for every positive amount sent in the trust game. We used the strategy method to elicit amounts returned, so each figure is based on N = 188 observations.




100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Break Even
Equal Earnings
50%
21%
12%
6%
5%
R0
R20
R40
R60
R80
R100
R120
Amount Returned (R40 Sent)


100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Break Even
Equal Earnings
44%
19%
13%
10%
4%
R0
R20
R40
R60
R80
R100
R120
R140
R160
R180
Amount Returned (R60 Sent)

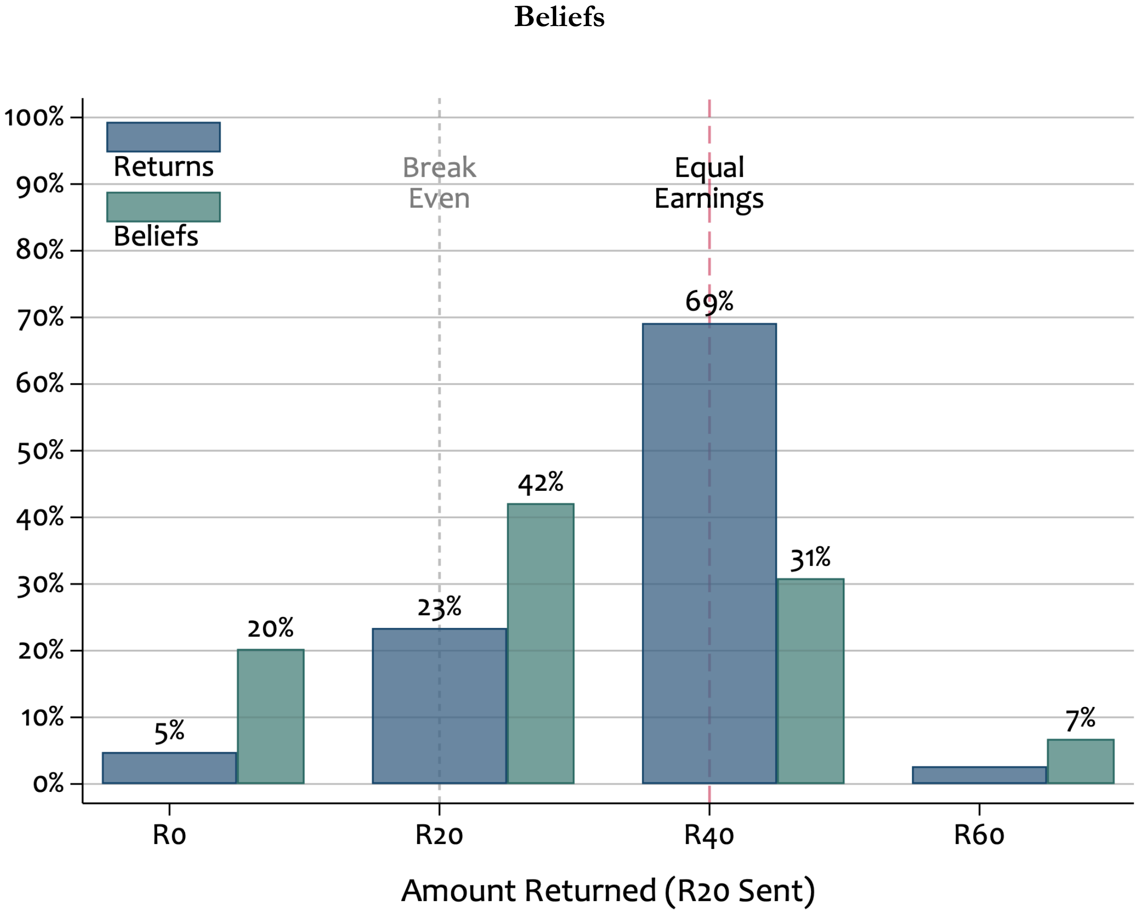

100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Break Even
Equal Earnings
4%
4%
4%
11%
13%
44%
6%
R0
R20
R40
R60
R80
R100
R120
R140
R160
R180
R200
R220
R240
Amount Returned (R80 Sent)

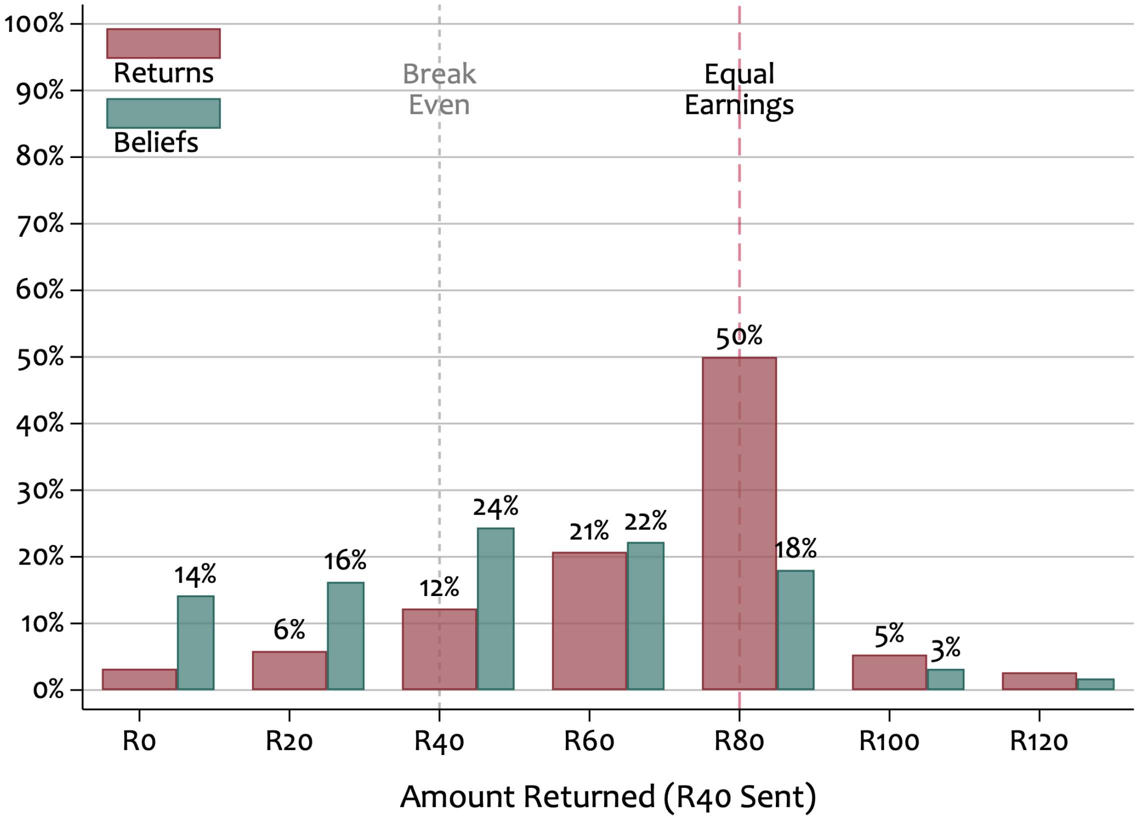

100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Break Even
Equal Earnings
6%
8%
4%
7%
10%
52%
R0
R20
R40
R60
R80
R100
R120
R140
R160
R180
R200
R220
R240
R260
R280
R300
Amount Returned (R100 Sent)

Appendix E: Additional Belief Figures

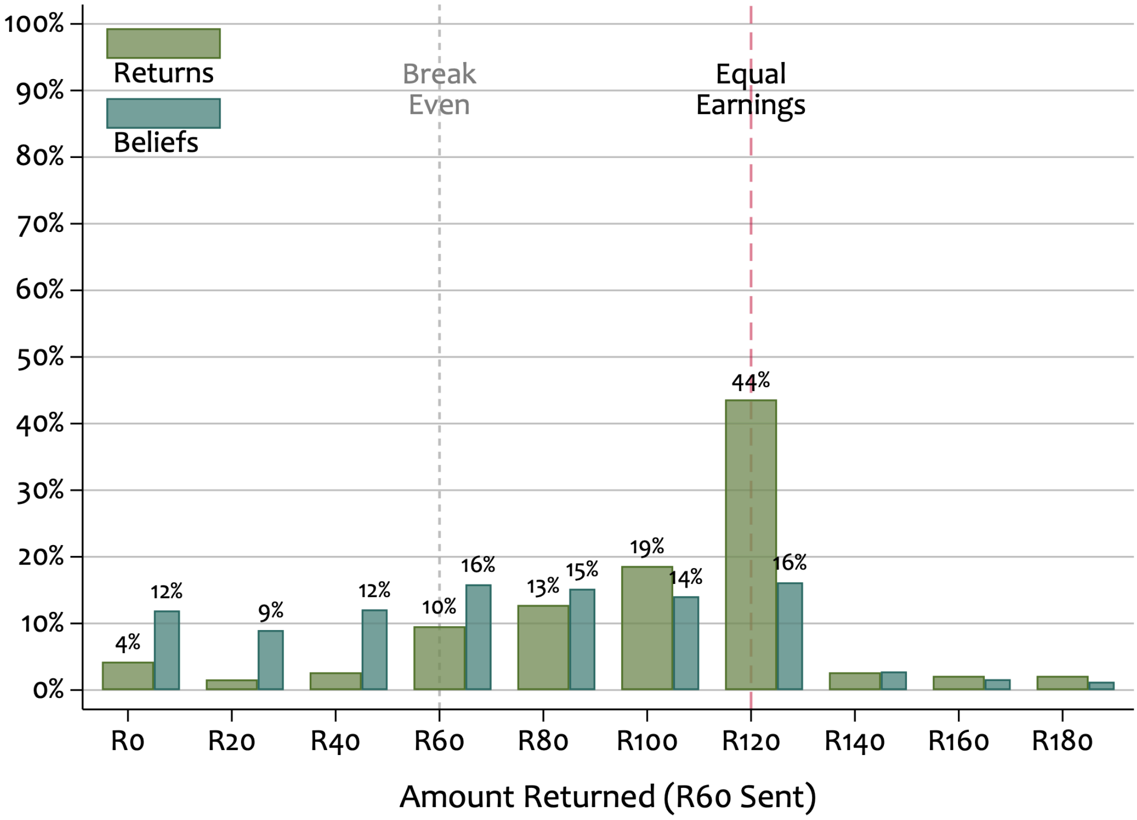
[Online Only]

In this appendix, we include figures of amounts returned for every positive amount sent in the trust game, together with estimated belief distributions. We also include figures that include raw token allocations as opposed to estimated beliefs.

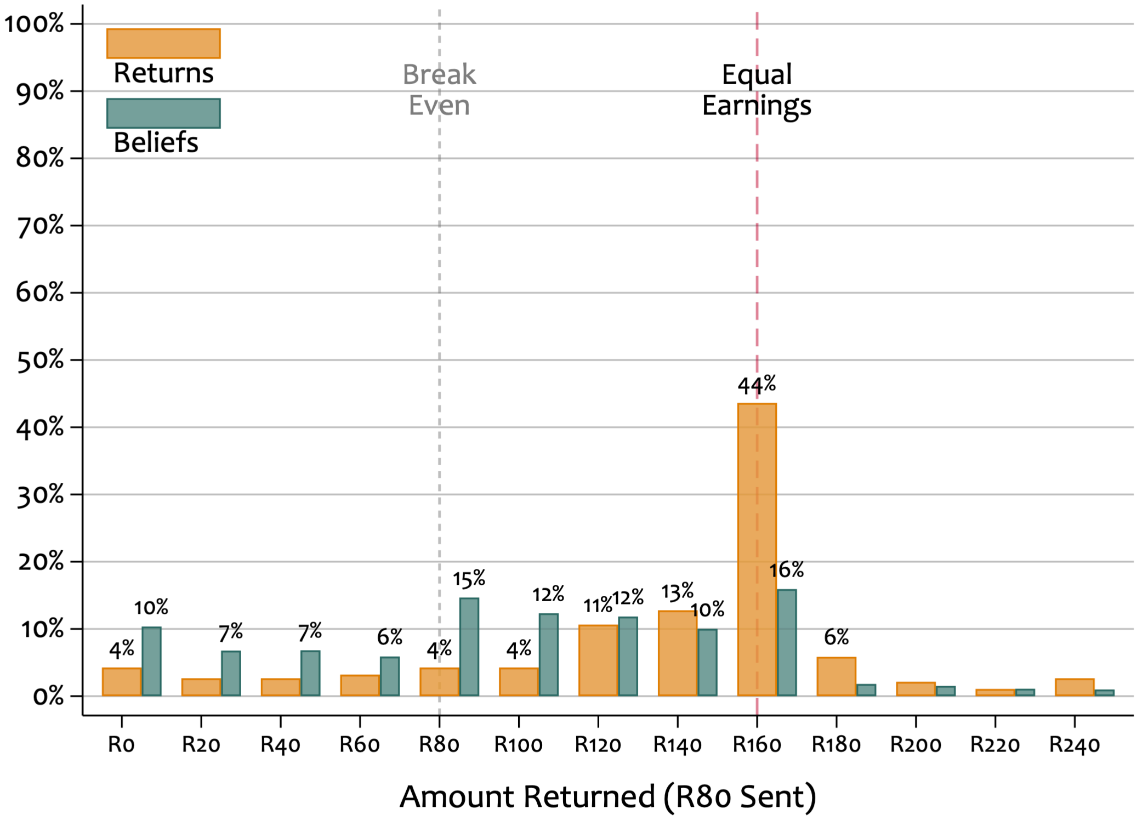




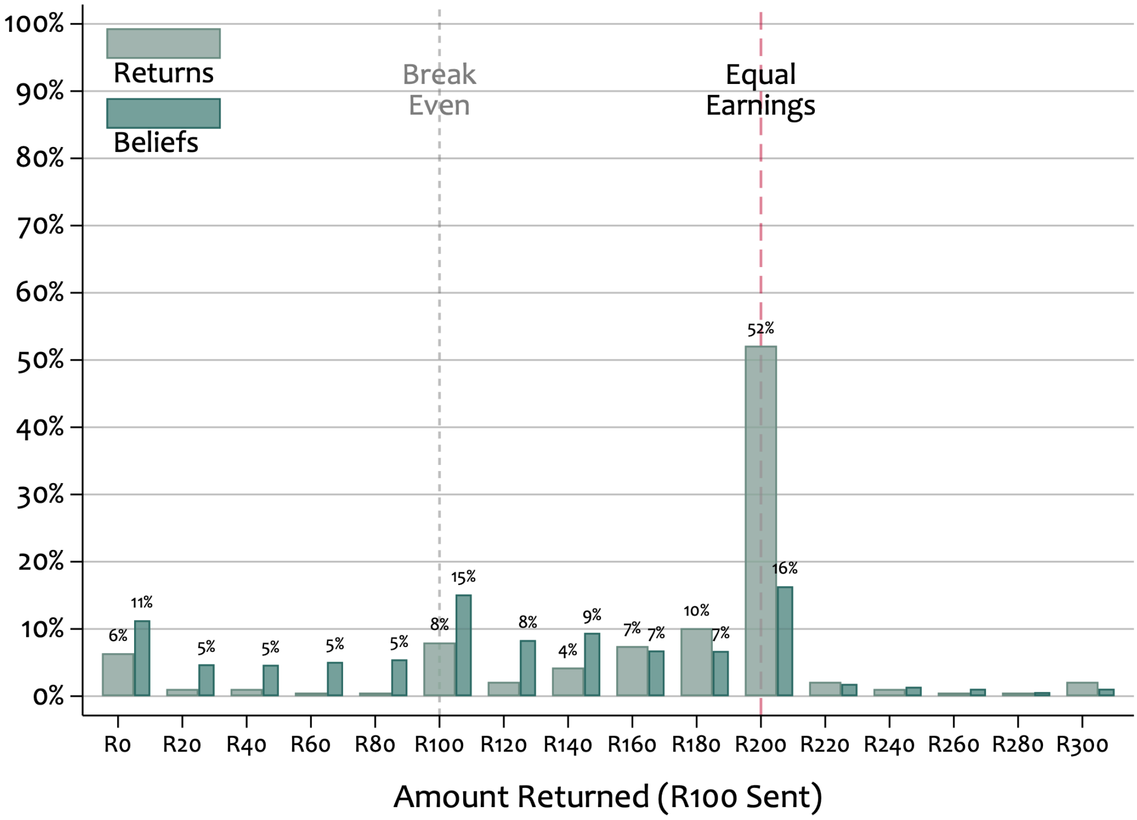
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Returns
Beliefs
Break Even
Equal Earnings
14%
6%
16%
12%
24%
21%
22%
50%
18%
5%
3%
R0
R20
R40
R60
R80
R100
R120
Amount Returned (R40 Sent)



100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Returns
Beliefs
Break Even
Equal Earnings
4%
12%
9%
12%
10%
16%
13%
15%
19%
14%
44%
16%
R0
R20
R40
R60
R80
R100
R120
R140
R160
R180
Amount Returned (R60 Sent)

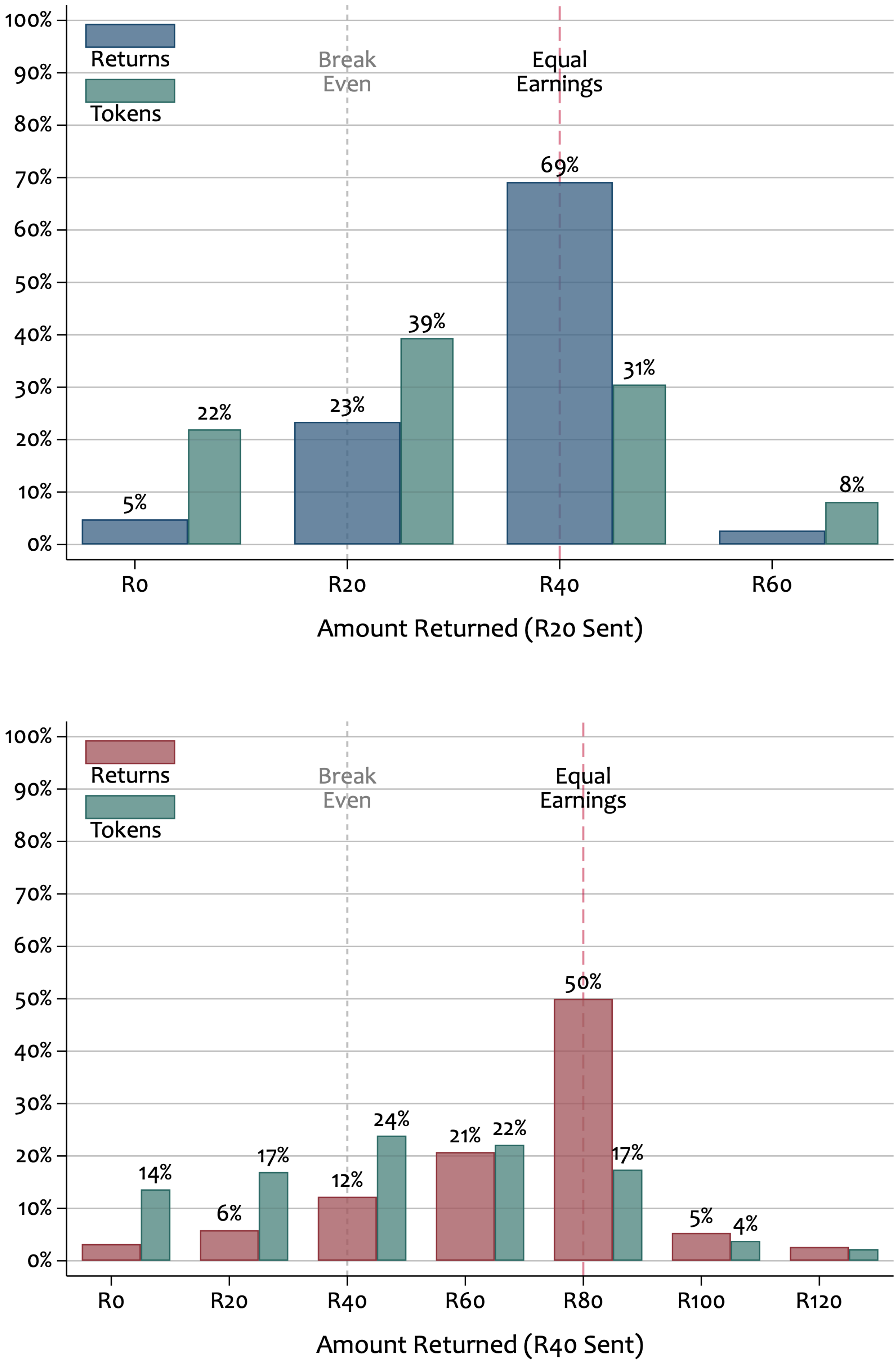


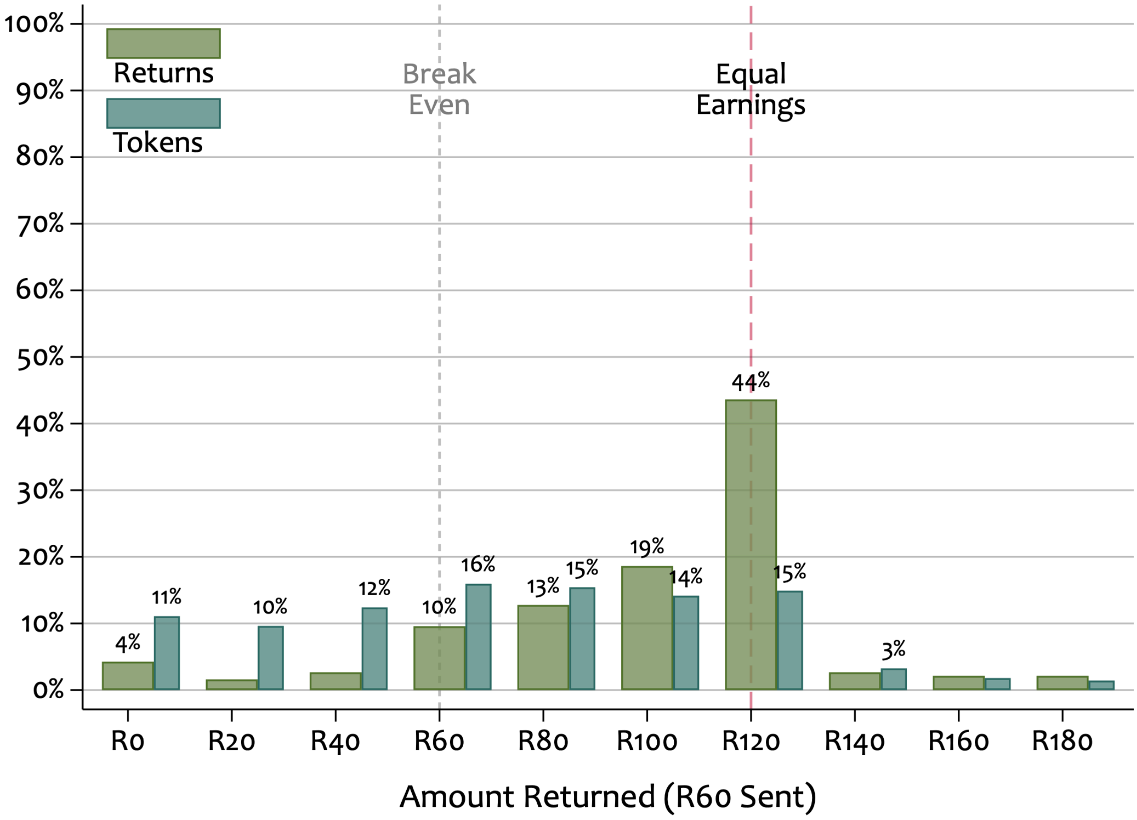
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Returns
Beliefs
Break Even
Equal Earnings
44%
16%
4%
10%
7%
7%
6%
4%
15%
4%
12%
11%
12%
13%
10%
6%
R0
R20
R40
R60
R80
R100
R120
R140
R160
R180
R200
R220
R240
Amount Returned (R80 Sent)

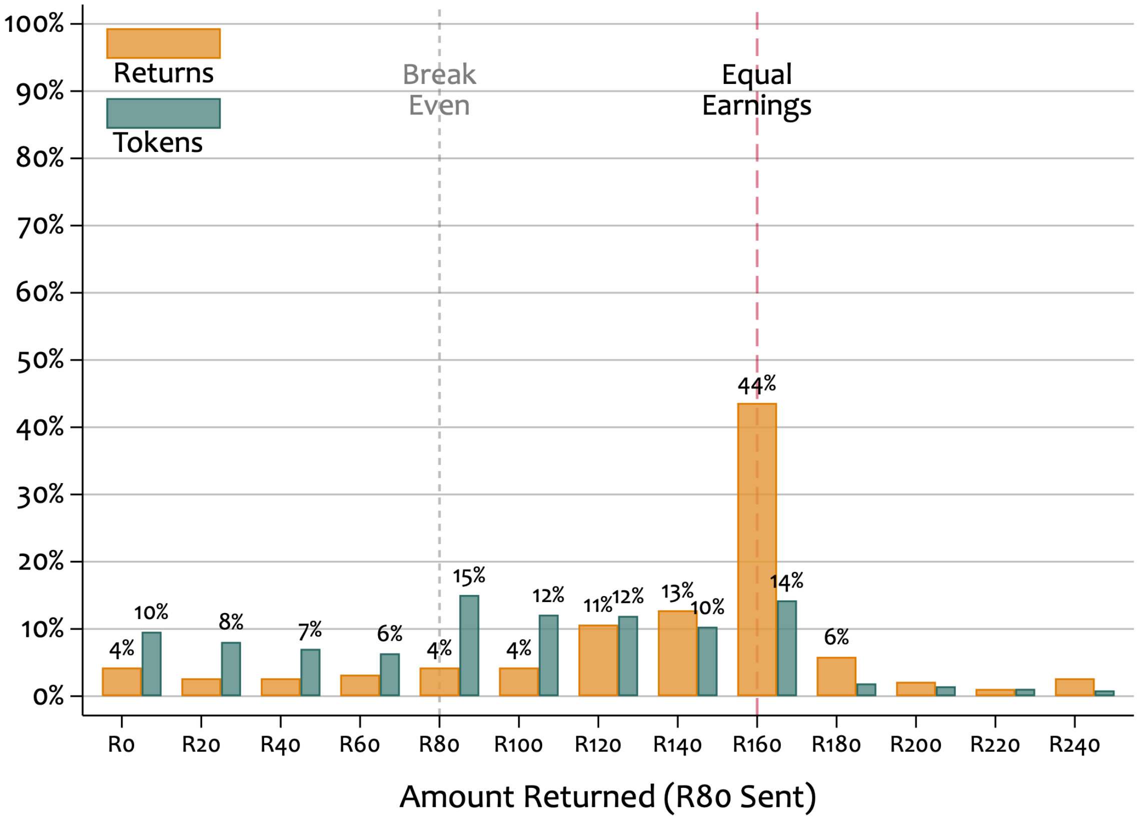


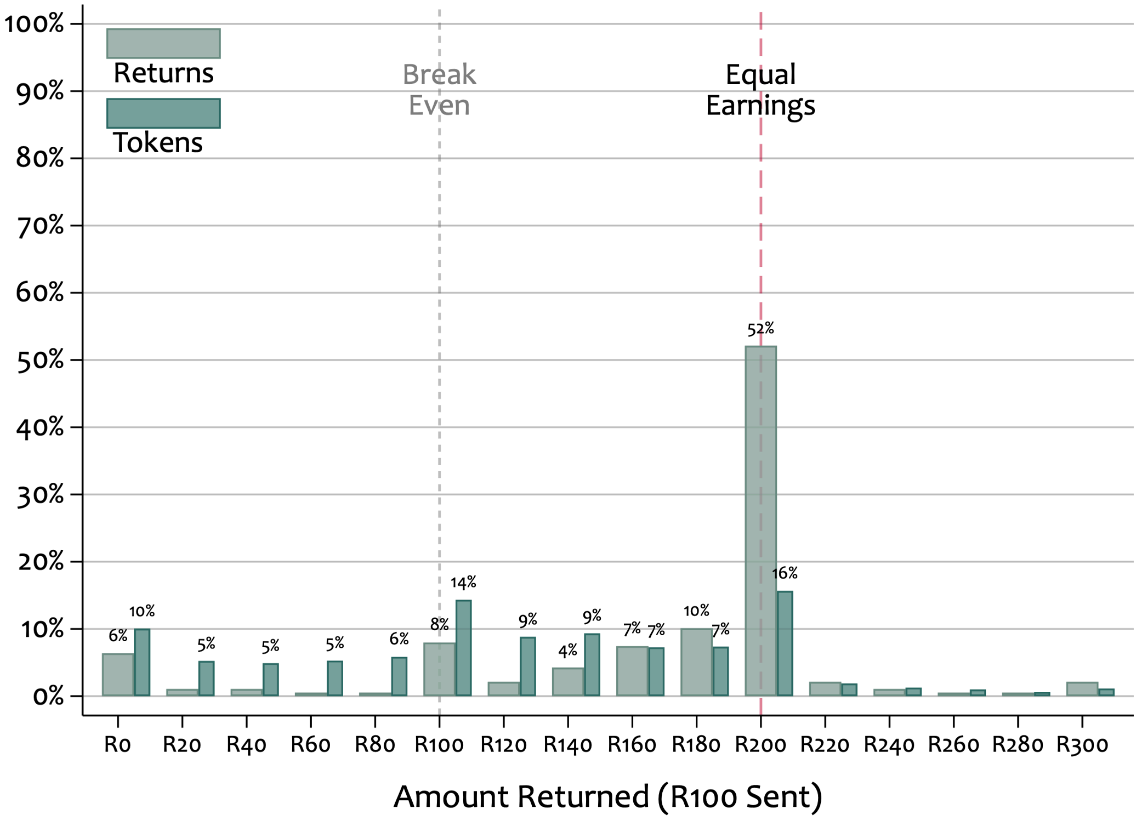
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Returns
Beliefs
Break Even
Equal Earnings
52%
16%
6%
11%
5%
5%
5%
5%
8%
15%
8%
4%
9%
7%
7%
10%
7%
R0
R20
R40
R60
R80
R100
R120
R140
R160
R180
R200
R220
R240
R260
R280
R300
Amount Returned (R100 Sent)

Token Allocations




100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Returns
Tokens
Break
Even
Equal
Earnings
4%
11%
10%
12%
10%
16%
13%
15%
19%
14%
44%
15%
3%
R0
R20
R40
R60
R80
R100
R120
R140
R160
R180
Amount Returned (R60 Sent)


100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Returns
Tokens
Break
Even
Equal
Earnings
4%
10%
8%
7%
6%
4%
15%
4%
12%
11%
12%
13%
10%
44%
14%
6%
R0
R20
R40
R60
R80
R100
R120
R140
R160
R180
R200
R220
R240
Amount Returned (R80 Sent)


100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Returns
Tokens
Break Even
Equal Earnings
6%
10%
5%
5%
5%
6%
8%
14%
9%
4%
9%
7%
7%
10%
7%
52%
16%
R0
R20
R40
R60
R80
R100
R120
R140
R160
R180
R200
R220
R240
R260
R280
R300
Amount Returned (R100 Sent)

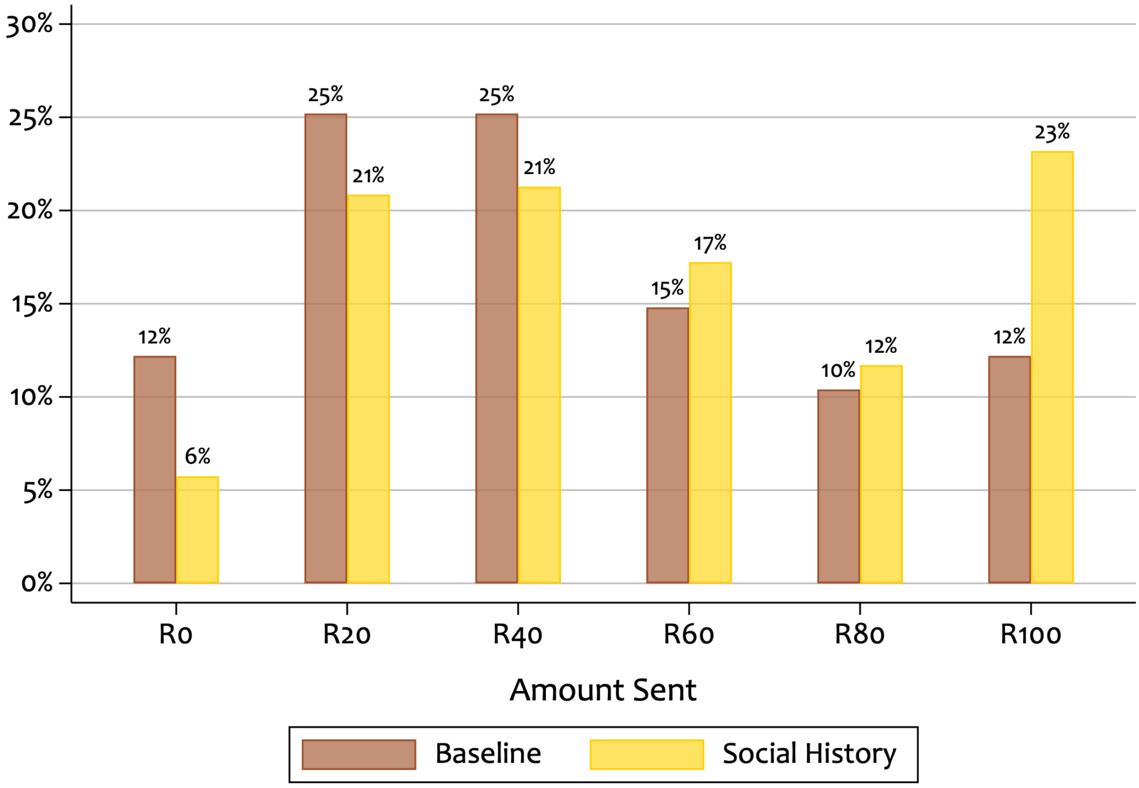
Appendix F: Baseline and Social History Treatment Robustness Checks
[Online Only]

In addition to the robustness checks in the main text, we also investigate whether amounts sent in the Baseline and Social History treatments differ when we only focus on the task orders common to both experiments: T_R, T_S, D and D, T_R, T_S. Figure F1 mimics Figure 10, but only includes data for task orders common to both experiments. The figure shows, again, that the marked differences occur at the end points of the distributions: amounts sent of R0 and R100. In the Baseline Treatment with common task orders, 12% of all amounts sent were R0, compared to only 6% in the Social History Treatment. On the other hand, 23% of amounts sent in the Social History Treatment were R100, compared to only 12% in the Baseline Treatment with common task orders. Thus, probability mass shifted from R0 to R100 across the Social History Treatment and Baseline Treatment with common task orders. AMEs from an ordered logit regression model confirm that probability mass shifted significantly from amounts sent of R0 and R20 to amounts sent of R60, R80, and R100 ($p < 0.01$ in all comparisons) across the Baseline Treatment (with common task orders) and the Social History Treatment; see Appendix C for results.

We also estimate a fractional response model of the proportion of the amounts returned across the Baseline and Social History treatments that includes the same set of covariates from our earlier specification. The logic of estimating this model is that perhaps subjects in the Social History Treatment were just more generous, both in terms of amounts sent and amounts returned, than subjects in the Baseline Treatment. Then the observed difference in amounts sent may be driven by differences in the samples in the two experiments as opposed to the provision of information.[1] The predicted probability of the proportion of the amount returned is actually *lower* for subjects in the Social History Treatment than the estimate for subjects in the Baseline

[1] Given that we include a number of variables to account for observed individual heterogeneity in our amount sent models this is unlikely, but we conduct this test nevertheless.

Treatment, but the difference is not statistically significant ($p = 0.16$). This result lends further credence to our assertion that presenting information about return behaviour in an easily intelligible way, and based on the behaviour of a relatively large sample of individuals, has an impact on amount sent behaviour in the trust game.



Note: Baseline and Social History treatments with common task orders.

Figure F1: Amounts Sent in Baseline and Social History Treatments with Common Task Orders

As final robustness checks, we estimate ordered logit models of the *first* and *last* amount sent choices to determine whether they differ across the Baseline and Social History treatments. The rationale for these tests is to ensure that the differences we find across treatments are not a function of the five amount sent choices in our experimental design. Focussing on the first amount sent choice, probability mass shifted significantly from amounts sent of R0 and R20 to amounts sent of R60, R80, and R100 ($p < 0.01$ in all comparisons) across the Baseline Treatment and the Social History Treatment. For the last amount sent choice, the probability of sending R0, R20, **and** R40 *decreases*, whereas the probability of sending R60, R80, and R100 *increases* across the

Baseline and Social History treatments (*p* < 0.05 in all comparisons). Thus, information provision has a consistent effect irrespective of whether we pool all five amount sent choices, or focus purely on the first and last amount sent choices.

Appendix G: Bayesian Updating of Beliefs

[Online Only]

This appendix includes split violin plots of the posterior belief distributions assuming priors of 25 (teal) or 250 (orange) previous observations of amounts returned for all possible amounts sent.

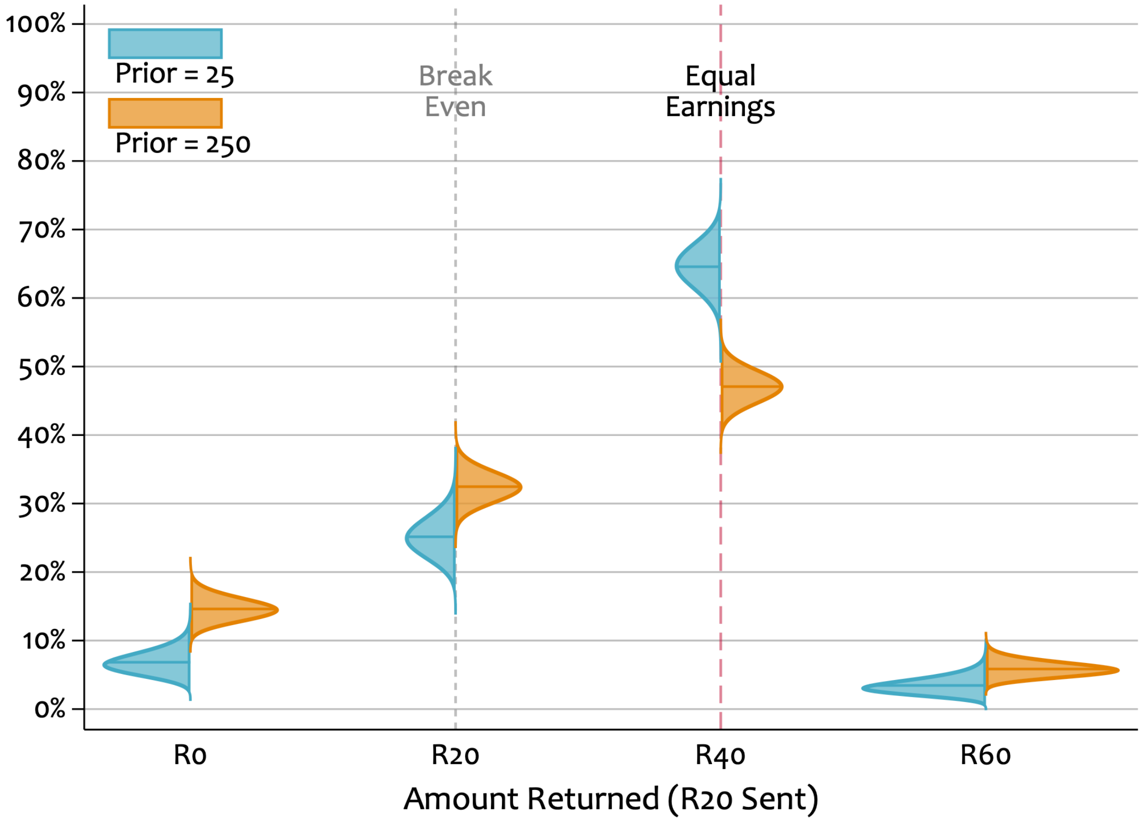


Figure G1: Posterior Beliefs About Amounts Returned

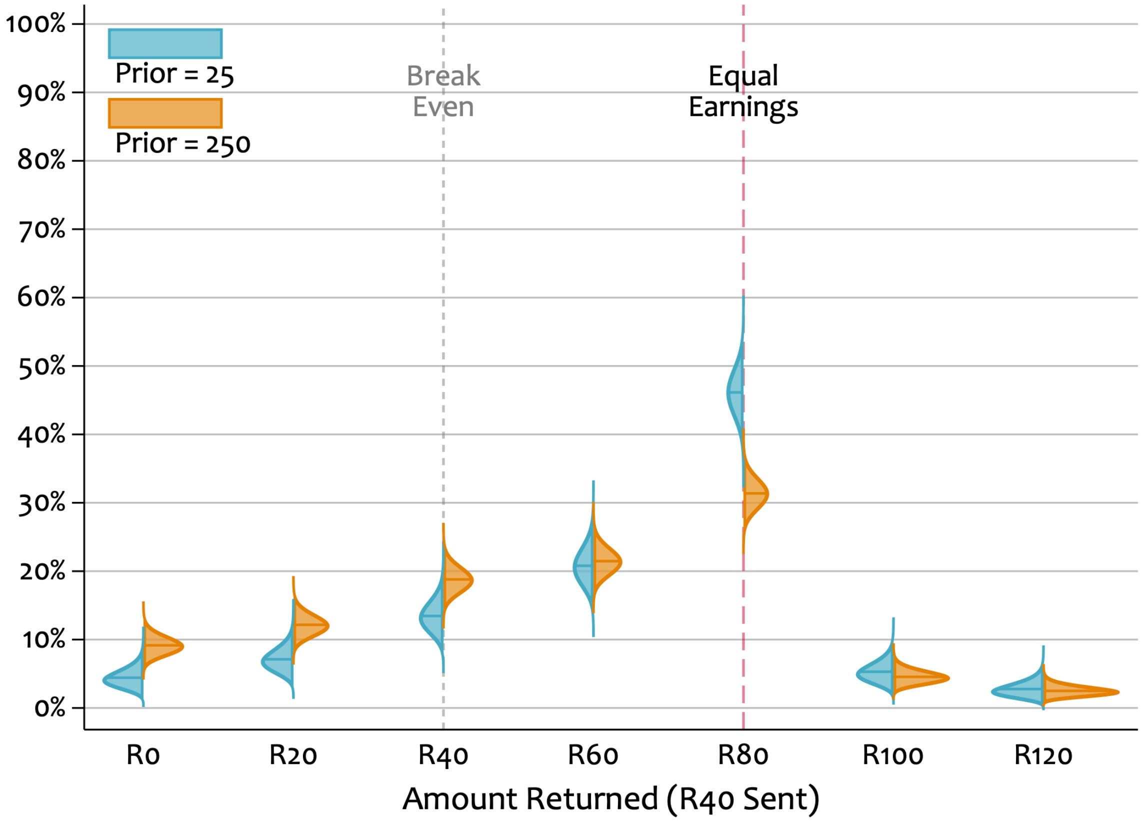


Figure G2: Posterior Beliefs About Amounts Returned

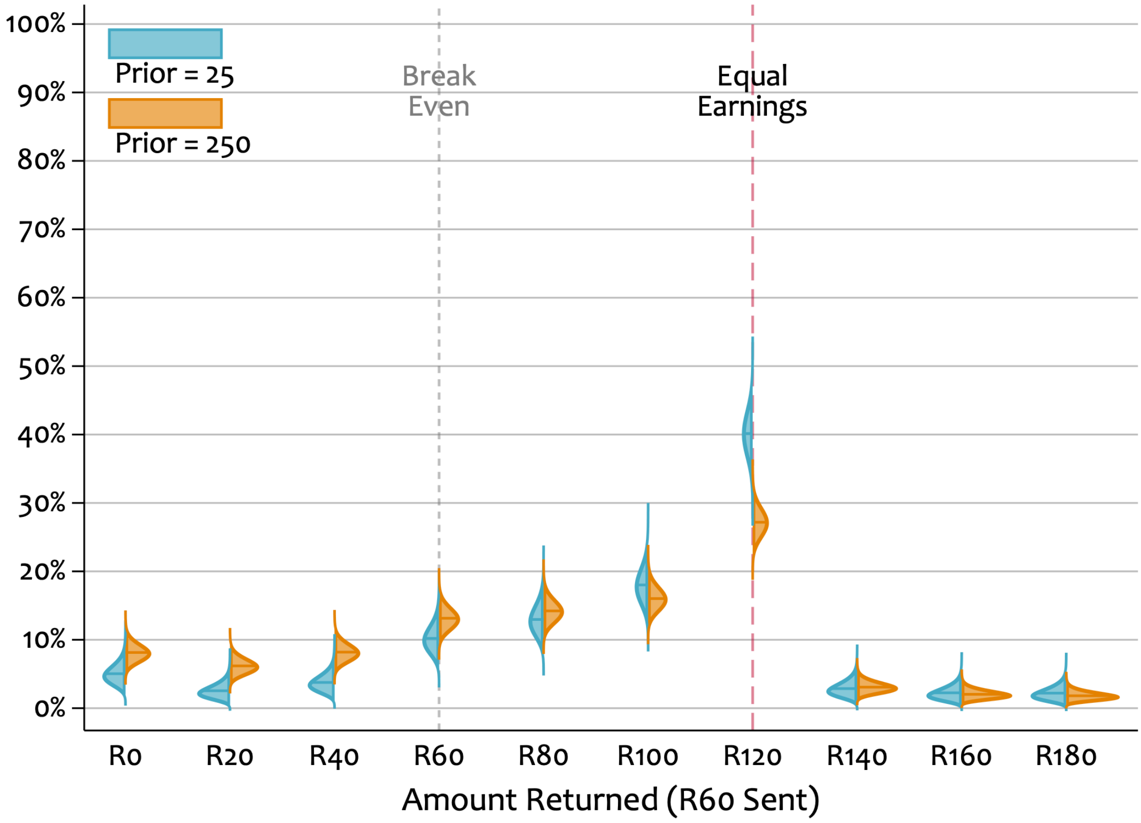


Figure G3: Posterior Beliefs About Amounts Returned

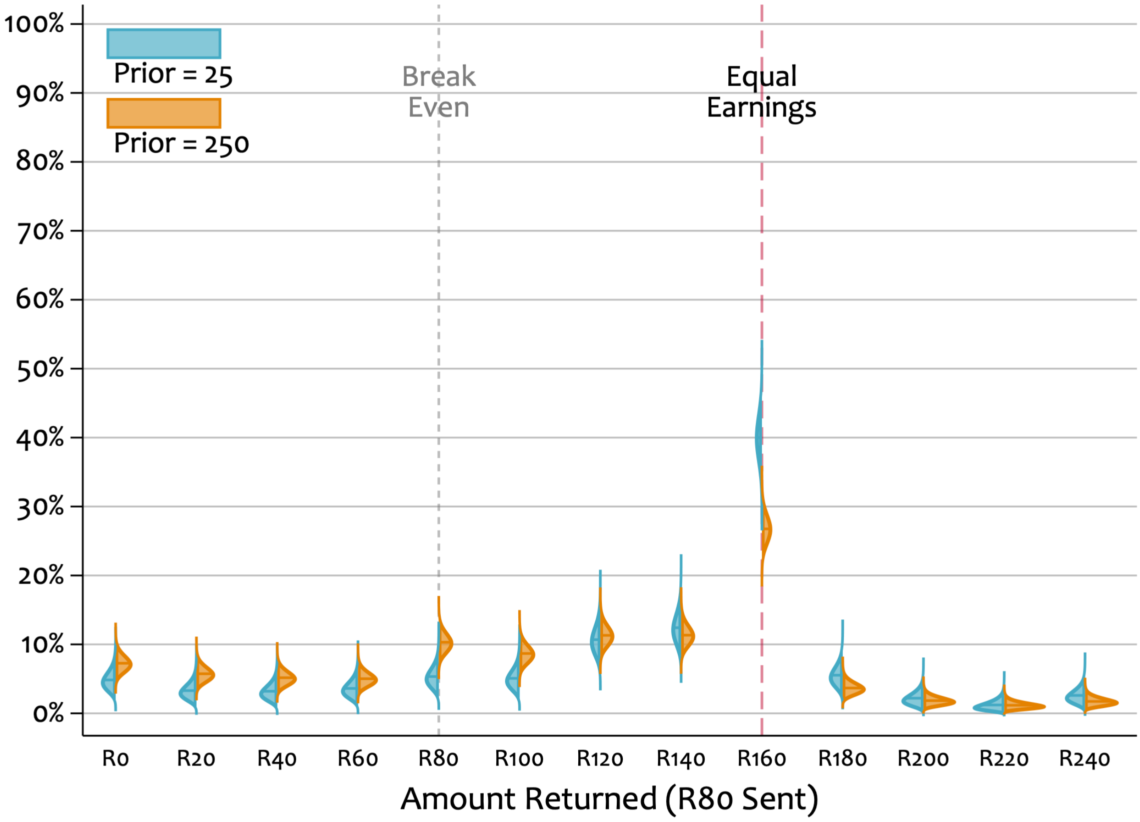


Figure G4: Posterior Beliefs About Amounts Returned

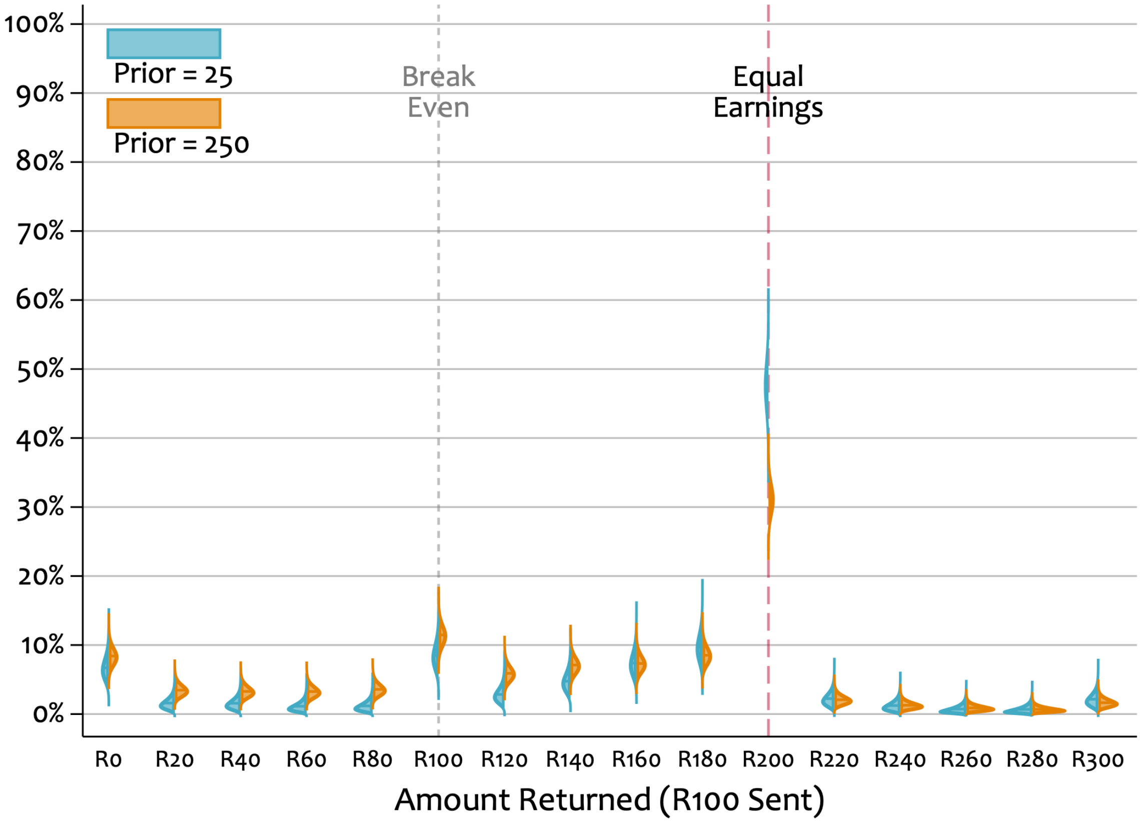


Figure G5: Posterior Beliefs About Amounts Returned