The norm ultimatum: when social norms impede optimal behavior

Introduction

While norms often act as a useful heuristic facilitating cohesive interpersonal dynamics, norms sometimes prevent the exploration of valuable courses of action. Under what circumstances do social norms impede optimal decision making? The ultimatum game is a useful tool with which to examine this issue. The ultimatum game is a dyadic paradigm in which one player, the proposer, decides how much of an endowment (e.g., €10) to offer to a responder, who decides whether to accept the offer, in which case the endowment is divided as proposed, or to reject the offer, in which case both parties receive nothing for that trial. While the canonical model of self-interest in economics predicts that proposers offer the smallest possible division of the endowment, and that responders accept any amount offered (Fehr & Schmidt, 2006), empirical studies find that the majority of proposers make offers between 40% and 60% (Oosterbeek, Sloof, & van de Kuilen, 2004), and that proposers commonly reject offers below 20% (Camerer, 2003). Two psychological phenomena have been offered as competing explanations of this deviation from pure self-interest: adherence to a fairness norm, and fear that an “unfair” offer will be rejected (Oosterbeek et al., 2004; Thaler, 1988). In the current experiment, we will be able to tease apart these two phenomena by elucidating how quickly proposers learn the range of offers that will be accepted by receivers, and by determining whether or not this learning rate is selectively impeded by the fairness norm.

To this end, we propose a design in which optimal behavior of proposers (optimal here defined as maximizing possible monetary payoff) requires violation of the fairness norm. Specifically, we propose a design in which proposers play multiple rounds against different groups of receivers, each group having their own acceptance thresholds. Crucially, these thresholds are not made known to the proposers, but must be learned through trial and error. In order to learn a threshold below the “fair” 50/50 split, a proposer must violate the fairness norm and actually make such an offer. We predict (prediction 1) that proposers will in fact avoid exploring the full acceptance range of the responders, and thus effectively leave money on the table, and behave suboptimally.

If this failure to learn is indeed cause by reliance on the fairness norm and not simply on fear that an offer will be rejected, then in circumstances where the fairness norm is not applicable (yet rejections are still possible), learning should take place and lead to optimal behavior. Following from this, in our design subjects will play in two different conditions: one in which the decisions made by responders are indeed made by real individuals who are actually accepting or rejecting the offers of the proposers, and one in which proposers are explicitly playing against the responses of a computer. In the computerized condition, the fairness norm does not apply, and therefore we predict (prediction 2) that subjects will explore the full acceptance range of the computerized receivers, and thus behave optimally.

Design

Each proposer will play in two different conditions: social (in which they are playing against other individuals), and computer (in which they are explicitly playing against a computerized algorithm programmed to mimic the acceptance threshold distributions of the real individuals). In both of these conditions proposers will play against 3 distinct groups of receivers, each demarcated with a neutral symbol, and each exhibiting a different average acceptance threshold which must be learned by the proposer in order to facilitate optimal behavior. The three distinct groups of receivers will be incentivized to exhibit different acceptance thresholds by manipulating their starting endowments and thus creating payout asymmetries of which the proposers are unaware. Specifically, the three receiver groups will start each trial as follows: group A will start each trial with a €0 endowment, group B will start each trial with a €5 endowment, and group C will start each trial with a €10 endowment. This means that offers made to group A are offers in which a total of €10 are at stake, offers made to group B are offers in which a total of €15 are at stake, and offers made to group C are offers in which a total of €20 are at stake. This means that a “fair” offer to a responder in group A is €5, a “fair” offer to a responder in group B is €2/3, and a “fair” offer to a responder in group C is €0. Therefore, if our responders rely on the fairness norm, then they should, on average, exhibit consistently different acceptance thresholds which our proposers must learn.

We assume that a proposer (𝑷) learns a responder’s (𝑹) acceptance logistic function 𝝋, which maps the proposal amount into a decision to accept or reject the offer. This function is governed by two parameters: 𝜷𝑹𝟎 (intercept) and 𝜷𝑹𝟏 (slope). At each trial 𝒕, from his/her endowment 𝑬𝒕, 𝑷 has to consider each possible offer 𝑨𝒕 and compute the *expected payoff* 𝑽𝒕. This expected payoff depends on his/her estimate of how likely it is that 𝑹 will accept this offer 𝑷𝑹,𝒕 (𝑫𝒕 = 𝟏|𝑨𝒕). For simplicity, this will be referred to as 𝑷𝑹,𝒕(𝑨𝒕). 𝑷’s expected payoff 𝑽𝒕 for the offer 𝑨𝒕 is:

𝑽𝒕=[𝑷𝑹,𝒕(𝑨𝒕)×(𝑬𝒕−𝑨𝒕)]+[(𝟏–𝑷𝑹,𝒕(𝑨𝒕))×𝟎] (1)

With the two square brackets representing cases where 𝑹 accepts and 𝑹 rejects, respectively.

Critically, 𝑷 estimates 𝑷𝑹,𝒕(𝑨𝒕) thanks to his/her estimate of 𝑹’s acceptance logistic function 𝝋:

𝑷𝑹,𝒕(𝑨𝒕)=𝟏/(𝟏+𝒆𝒙𝒑(−(𝜷𝑹𝟎,𝒕+𝜷𝑹𝟏,𝒕×𝑨𝒕))) (2)

𝑷 can choose the offer value 𝑨𝒕 which truly maximizes his expected payoff (no parameter)

𝑨𝒕=𝐚𝐫𝐠𝒎𝒂𝒙𝑽𝒕 (3)

Alternatively, 𝑷 can choose 𝑨𝒕 according to a softmax function (multinomial), with a temperature parameter 𝜷

𝑷(𝑨𝒕=𝒊) = (4)

After the offer is made, 𝑹 will make the decision 𝑫𝒕 to accept or reject the offer. This decision can be used by 𝑷 to update his/her representation of 𝑹’s acceptance logistic function ϑ**.** Learning could be done using a simple delta rule on ϑ’s two parameters 𝜷𝑹𝟎,𝒕 and 𝜷𝑹𝟏,𝒕.

This means that 𝑹 first computes a choice prediction error 𝜹𝒕

𝜹𝒕=𝑫𝒕−𝑷𝑹,𝒕(𝑨𝒕) (5)

Importantly, the model needs initial estimates of 𝜷𝑹𝟎,𝒕+𝟏 and 𝜷𝑹𝟏,𝒕+𝟏. These could be fitted (but those would be additional free-parameters) or we could use empirical values (e.g. estimated from multiple one-shot games, performed before the repeated game of interest).

In addition to the parameters included in the above equations, we can test alternative models against one another containing additional free parameters capturing such processes as risk propensity/aversion, inequity aversion, and collective utility (Fehr & Schmidt, 2001; Niv et al., 2012; etc.).

References

Camerer, C. (2003). Behavioral Game Theory: Experiments in Strategic Interaction. *Insights in Decision Making: A Tribute to Hillel J. Einhorn*, 544. https://doi.org/10.1024/0301-1526.32.1.54

Fehr, E., & Schmidt, K. M. (1999). A Theory of Fairness, Competition, and Cooperation. *The Quarterly Journal of Economics*, *114*(3), 817–868. https://doi.org/10.1162/003355399556151

Fehr, E., & Schmidt, K. M. (2006). Chapter 8 The Economics of Fairness, Reciprocity and Altruism - Experimental Evidence and New Theories. *Handbook of the Economics of Giving, Altruism and Reciprocity*, *1*(6), 615–691. https://doi.org/10.1016/S1574-0714(06)01008-6

Henrich, J., Boyd, R., Bowles, S., Camerer, C., Fehr, E., Gintis, H., … Tracer, D. (2005). “Economic man” in cross-cultural perspective: Behavioral experiments in 15 small-scale societies. *Behavioral and Brain Sciences*, *28*(6), 795–855. https://doi.org/10.1017/S0140525X05000142

Niv, Y., Edlund, J. A., Dayan, P., & O’Doherty, J. P. (2012). Neural Prediction Errors Reveal a Risk-Sensitive Reinforcement-Learning Process in the Human Brain. *Journal of Neuroscience*, *32*(2), 551–562. https://doi.org/10.1523/JNEUROSCI.5498-10.2012

Oosterbeek, H., Sloof, R., & van de Kuilen, G. (2004). Cultural differences in ultimatum experiments: Evidence from a meta-analysis. *Experimental Economics*, *7*(2), 171–188.