

# Manipulation-Proof Machine Learning\*

Daniel Björkegren<sup>†</sup>

Brown University

Joshua E. Blumenstock<sup>‡</sup>

U.C. Berkeley

Samsun Knight<sup>§</sup>

Brown University

This version: August 21, 2021

First version: November 30, 2018

## Abstract

An increasing number of decisions are guided by machine learning algorithms. In many settings, from consumer credit to criminal justice, those decisions are made by applying an estimator to data on an individual’s observed behavior. But when consequential decisions are encoded in rules, individuals may strategically alter their behavior to achieve desired outcomes. This paper develops a class of estimator that is stable under manipulation, even when the decision rule is fully transparent. We explicitly model the costs of manipulating different behaviors, and identify decision rules that are stable in equilibrium. This approach also makes it possible to quantify the performance cost of making a decision algorithm transparent. Through a large field experiment in Kenya, we show that decision rules estimated with our strategy-robust method outperform those based on standard supervised learning approaches.

*Keywords:* machine learning, manipulation, decisionmaking, targeting

---

\*We are grateful for helpful conversations with Susan Athey, Jon Bittner, John Friedman, Greg Lewis, and Jesse Shapiro. This project would not have been possible without the creative work of Channing Jang, Simon Muthusi, Nicholas Owsley, and the rest of the team at the Busara Center for Behavioral Economics. We thank Jolie Wei for excellent research assistance, and numerous audiences for helpful feedback. We are grateful for funding from the Brown University Seed Fund, the Bill and Melinda Gates Foundation, and the Center for Effective Global Action. Björkegren thanks the W. Glenn Campbell and Rita Ricardo-Campbell National Fellowship at Stanford University, and Microsoft Research for support. Blumenstock thanks the National Science Foundation for support under CAREER Grant IIS-1942702. This study was pre-registered with the AEA RCT Registry (AEARCTR-0004649), and approved by the IRBs of UC Berkeley, Brown University, and the Kenya Medical Research Institute.

<sup>†</sup>dan@bjorkegren.com

<sup>‡</sup>jblumenstock@berkeley.edu

<sup>§</sup>samsun\_knight@brown.edu

# 1 Introduction

An increasing number of important decisions are being made by machine learning algorithms. Algorithms determine what information we see online; who is hired, fired, and promoted; who gets a loan, and whether to give bail and parole. In the typical machine learning deployment, an individual’s observed behavior is used as input to an estimator that determines future decisions.

These applications of machine intelligence raise two related problems. First, when algorithms are used to make consequential decisions, they create incentives for people to ‘game’ the rule: when agents understand how their behavior affects decisions, they may alter their behavior to achieve the outcome they desire. Second, society increasingly demands a ‘right to explanation’ about how algorithmic decisions are made (Goodman and Flaxman, 2016; Barocas et al., 2018). For instance, the European Union’s General Data Protection Regulation mandates that “meaningful information about the logic” of automated systems be available to data subjects (European Union, 2016). However, such transparency increases the scope for gaming: the more clearly that agents know how their behavior affects a decision, the easier it is to manipulate.

The problem of manipulation stems from the fact that the standard estimators used to construct decision rules assume that the relationship between the outcome of interest and human behaviors is stable. But this assumption tends to be violated as soon as a decision rule is implemented, and agents have incentives to change their behavior to achieve more favored outcomes. When decision rules are gamed, they can produce decisions that are arbitrarily poor or unsafe, which can undermine the use of machine learning in critical applications.

There are two common approaches to deal with this problem. The first, familiar to economists, restricts models to predictors that are presumed to have a theoretical relationship to the outcome of interest.<sup>1</sup> This simple intuitive approach amounts to having a dogmatic prior that the cost of manipulation is either infinite (for included predictors) or zero (for excluded predictors). However, most behaviors are manipulable at some cost, and it may be difficult to assess manipulability in new domains, or

---

<sup>1</sup>An extreme version may restrict to predictors that causally affect the outcome of interest (Kleinberg and Raghavan, 2019; Milli et al., 2019). This may make manipulation desirable (for example, an exam may induce students to study and learn general knowledge) but can reduce predictive performance.

in modern contexts that can have thousands of predictors. Thus in practice many implementations use a second approach, which we refer to as the ‘industry approach.’ This keeps decision rules secret, and periodically updates the model to account for changes in the relationship between features and outcomes (Bruckner and Scheffer, 2011). However, such ‘security through obscurity’ exposes current applications to substantial risk (NIST 2008). If the stakes are high enough, people eventually learn—and exploit—a system’s weaknesses, and may cause great harm at unanticipated times. This approach also limits the use of machine learning in settings where secrecy cannot be maintained (e.g., when regulations mandate transparency) or feedback is noisy or delayed (e.g., it may take years for a social media platform to learn that its recommendation algorithm was gamed by foreign actors). There is also no guarantee that the back-and-forth between estimation and agents will reach equilibrium, or if it does, that such an equilibrium will be desirable.

This paper develops a different approach. We explicitly model the costs that agents incur to manipulate their behavior, and embed the resulting game theoretic model within a machine learning estimator. This allows us to construct decision rules that anticipate strategic agents, and which make good decisions even when the rule is fully transparent. We demonstrate, using Monte Carlo simulations, that our ‘strategy-robust’ estimator performs better than standard models when these costs are known, even if costs are misspecified. We then test the theory in a real world environment, through an incentivized field experiment with 1,557 people in Kenya. We use the experiment to elicit costs of manipulating behavior, and to show that the strategy-robust approach leads to more robust machine decisions.

The paper is organized into two main parts. The first part develops a method to estimate strategy-robust decision rules that are stable under manipulation. We consider a supervised machine learning framework for a policymaker making a decision  $y_i$  for each individual  $i$ . Each individual prefers a larger decision  $y_i$ . We observe a *training* subset of cases that possess both features  $\mathbf{x}_i$  and optimal decisions  $y_i$ . The policymaker seeks to estimate a decision rule  $\hat{y}(\mathbf{x}_i)$  for cases in an *implementation* subset where only features  $\mathbf{x}_i$  are observed. Standard methods assume that  $\mathbf{x}_i$ ’s are fixed: training and implementation samples of  $(\mathbf{x}_i, y_i)$  are drawn from same distribution. Our method allows individuals to adjust behavior in response to the incentives generated by the

decision rule; that is,  $\mathbf{x}_i(\hat{y}(\cdot))$  is a function of the decision rule. Thus, while the training samples come from an unincentivized distribution  $(\mathbf{x}_i(0), y_i)$ ; implementation samples come from  $(\mathbf{x}_i(\hat{y}(\cdot)), y_i)$ . Characterizing this latter distribution requires a new object: the distribution of the elasticity of behaviors  $\mathbf{x}_i$  when incentivized. We assume individuals incur quadratic costs for manipulating behavior, and that these costs can be parametrized by a cost distribution  $\mathbb{C}$ . We describe several methods to estimate manipulation costs.

To sharpen intuition, we derive results for linear decision rules of the form  $\hat{y}(\mathbf{x}) = \beta\mathbf{x}$ . The resulting estimator takes a simple nonlinear least squares form.<sup>2</sup> Our method introduces a new notion of fit, which has analogues to other common linear regression approaches. Ordinary least squares (OLS) maximizes fit within sample; two stage least squares (2SLS) sacrifices fit within sample to estimate coefficients that have causal interpretations; penalized least squares (such as LASSO and ridge) sacrifice within-sample fit to better generalize to other samples drawn from the same population. Our method sacrifices fit within sample to maximize fit in the counterfactual where the decision rule is used to allocate resources, and agents manipulate against it. Our estimator is an example of a class of estimator that maximizes *counterfactual fit*—predictive fit in a counterfactual state of the world.<sup>3</sup> Our method nests standard linear estimators (OLS and penalized least squares), and we regularize towards them based on the amount of manipulation observed in the data.

We use Monte Carlo simulations to compare this strategy-robust approach to common alternatives. OLS can perform extremely poorly when agents behave strategically. The industry approach, which periodically retrains the model, may not converge, or if it does, may do so slowly or to an undesirable equilibrium. By contrast, our method adjusts the model to anticipate manipulation, in each subgame. In simulations where agents respond to the decision rule and manipulation costs are known, our approach exceeds the performance of other estimators. Our approach can exceed the performance of others even if manipulation costs are misspecified for some cases. Under certain parameters, the presence of manipulation can *improve* predictive performance, if it signals unobservables associated with the outcome of interest (in the spirit of

---

<sup>2</sup>Although many machine learning implementations use complex nonlinear models, peoples’ beliefs about these models tend to be simple, and may be well approximated by linear functions.

<sup>3</sup>This is analogous to the concept of performative prediction suggested by [Perdomo et al. \(2020\)](#).

Spence, 1973). In these cases, our method emphasizes features that are manipulable by the types to be screened in, but not by those to be screened out.

In the second part of the paper, we implement and test our method in the context of a field experiment in Kenya. This experiment allows us to compare the performance of the strategy-robust estimator to standard machine learning algorithms in a real-world environment. Specifically, we built a new smartphone app that passively collects data on how people use their phones, and disburses monetary rewards to users based on the data collected. The app is designed to mimic ‘digital credit’ products that are spreading dramatically throughout the developing world (Bharadwaj et al., 2019; Brailovskaya et al., 2021). Digital credit products similarly collect user data, and convert it into a credit score using machine learning, based on the insight that historical patterns of mobile phone use can predict loan repayment (Björkegren, 2010; Francis et al., 2017; Björkegren and Grissen, 2019). However, as these systems have scaled, manipulation has become commonplace as borrowers learn what behaviors will increase their credit limits (McCaffrey et al., 2013; Bloomberg, 2015).<sup>4</sup>

This field experiment produces several results. First, consistent with prior work, we find that a person’s mobile phone usage behaviors ( $\mathbf{x}_i(0)$ ) have a predictive relationship with their characteristics, such as income and intelligence (Raven’s matrices).<sup>5</sup>

Second, we structurally estimate  $\mathbb{C}$  in our model; that is, the distribution of costs of manipulating a variety of observed behaviors  $\mathbf{x}_i$ . These estimates are identified through a series of randomly assigned experiments that offer financial incentives to alter behaviors observed through the app. For example, participants may be paid to increase the frequency of outgoing calls in a given week, or decrease the number text messages they receive. The average weekly payouts are designed to be similar in magnitude to the typical digital credit loans in Kenya at the time (\$4.80 in Bharadwaj et al. (2019)). The pattern of costs we estimate is intuitive: for instance, outgoing communications are less costly to manipulate than incoming communications, and text messages, which are relatively cheap to send, are more easily manipulated than calls. We also find that complex behaviors (such as the standard deviation of talk

---

<sup>4</sup>A recent survey in Kenya and Tanzania found that one of the top five reasons people report saving money in digital accounts is to increase the loan amount qualified for (FSD Kenya, 2018).

<sup>5</sup>Related work has used mobile phone data to predict income and wealth (Blumenstock et al., 2015; Blumenstock, 2018), gender (Blumenstock et al., 2010), and employment (Sundsøy et al., 2016).

time) are less manipulable than simpler behaviors (such as the average duration of talk time).

Third, we find that ‘strategy-robust’ decision rules, which account for the costs of manipulation, perform substantially better than standard machine learning algorithms. We make this comparison by providing participants financial incentives to use their phones like a person of a particular type. For instance, some people receive a message that says, “Earn up to 1000 Ksh if the app guesses that you are a high income earner, based on how you use your phone,” while others receive messages that offer rewards for acting like an ‘intelligent’ person, and so forth. Across a variety of such decision rules, we show that classifications made with the strategy-robust algorithm are more accurate than classifications from standard algorithms.

Finally, we use our method to estimate the performance cost of algorithmic transparency, incurred to the policymaker for disclosing the details of the decision rule. In the experiment, we experimentally vary the amount of information subjects have about the decision rule, and show that the relative performance of the strategy-robust estimator increases with transparency. Transparency reduces the predictive performance of standard estimators by 17% on average, but reduces the strategy-robust estimator’s performance by only 6%. In our setting, the performance cost of moving from an equilibrium where decision rules are secret to an equilibrium where they are disclosed is less than 8%. Our model also allows policymakers to bound this equilibrium cost of transparency even without disclosing decision rules to the world.

Thus, our paper provides a framework for designing decision rules that are robust to manipulation. The empirical approach we take is similar to how organizations hire ‘white hat’ hackers to uncover and repair security weaknesses before they are exploited. It can add the most value in settings where stakes are high; where policymakers have limited evidence on historical manipulation; where decision rules cannot be kept secret; or where updating decision rules is costly or slow. That is, it may be useful for applications beyond the dominant tech firms, such as governments. It may be less successful in contexts where manipulation costs are difficult to model due to unpredictable shifts, or where models cannot be decomposed into manipulable components. We consider the linear case, and describe how our approach could be applied to nonlinear models.

## 1.1 Connection to Literature

Agents game decision rules in a wide variety of empirical settings. Manipulation has been documented in contexts ranging from New York high school exit exams (Dee et al., 2019) and health provider report cards (Dranove et al., 2003), to pollution monitoring in China (Greenstone et al., 2019), to fish vendors in Chile (Gonzalez-Lira and Mobarak, 2019). In the online advertising industry, firms spend many millions of dollars each year on search engine optimization, manipulating their websites in order to be ranked higher by search engine algorithms (Borrell Associates, 2016). A quick Google search suggests over 50 thousand different websites (and 3,000 YouTube videos) contain the phrase “hack your credit score.”

Indeed, the dilemma of manipulation is not new. Goodhart (1975), in what has since become referred to as ‘Goodhart’s Law’, noted that once a measure becomes a target, it ceases to be a good measure. Lucas (1976) also famously observed that historical patterns can deviate when economic policy changes. More broadly, our approach connects with literatures in both economics and computer science.

Our problem can be viewed as a mechanism design problem. Canonical signaling models (Spence, 1973) rely on a single crossing condition to allow full revelation of individual types. In our setting, like the settings of Frankel and Kartik (2019, 2020) and Ball (2019), there are two forms of heterogeneity: types  $\underline{x}_i$  and the costs of manipulating behavior  $C_i$ . Frankel and Kartik (2019) show that unobserved heterogeneity in manipulation costs  $c_i$  ‘muddles’ the relationship between a behavior  $x_i$  and type  $\underline{x}_i$ , causing the single crossing condition to fail. That paper shows that muddling reduces the information available in a market. Ball (2019) extends that framework to multiple dimensions of behavior, and in a theoretical model similar to ours characterizes and proves the existence of an equilibrium. That paper, as well as Frankel and Kartik (2020), suggest that committing to a subgame perfect solution like ours can lead to better outcomes than repeated best responses. Also related, Eliaz and Spiegler (2019) show that incentive problems can theoretically arise even in a setting where agents and policymaker have identical objective functions, if the policymaker adjusts their objective function with penalization. Relative to this work, our paper builds a model that can be empirically estimated, which allows us to probabilistically separate types and costs.

Our paper is also related to the problem in public finance of setting taxes in environments where agents adapt their behaviors. [Mirrlees \(1971\)](#) considers taxes as a function of earnings, and faces the central problem that taxation induces a behavioral response. [Akerlof \(1978\)](#) suggests that conditioning on additional attributes that are harder to manipulate (‘tags’) can improve efficiency. Our method evaluates predictors with the inverse of the matrix of the costs of manipulating them, in a manner similar to [Ramsey \(1927\)](#). The market design literature has also considered designing allocation algorithms in the face of strategic reporting ([Agarwal and Budish, 2021](#)).

Our experimental application also connects to work on poverty targeting. In developing countries and other settings where income is difficult to observe, policymakers commonly determine program eligibility ( $y_i$ ) based on easily observable characteristics or behaviors ( $\mathbf{x}_i$ ) ([Hanna and Olken, 2018](#)), and more recently, based on how people use their phones ([Aiken et al., 2021](#)). The policymaker may infer a household’s type based on the levels of these variables, or, implicitly, on how they change in response to incentives.<sup>6</sup> There is evidence that such decision rules induce households to manipulate their observable features. For instance, [Banerjee et al. \(2018\)](#) find that adding a question about flat screen TV ownership to a census caused people to underreport ownership by 16% on a follow-up survey, in order to appear less wealthy.<sup>7</sup>

Finally, our approach relates to existing strands in the computer science literature. The theoretical computer science community has recently considered this problem as one of ‘strategic classification’ ([Hardt et al., 2016](#); [Dong et al., 2018](#)). This literature is focused primarily on obtaining computationally efficient learning algorithms, and how strategic behavior can affect statistical definitions of fairness ([Hu et al., 2019](#); [Milli et al., 2019](#)). In computer security, ‘adversarial machine learning’ considers how strategic adversaries can systematically undermine supervised learning algorithms.<sup>8</sup> Also related is the concept of ‘covariate shift’, which considers scenarios where a test

---

<sup>6</sup>Our method thus includes this latter case of self-targeting ([Nichols and Zeckhauser, 1982](#); [Alatas et al., 2016](#)), which identifies beneficiaries based on willingness to engage with a costly “ordeal.”

<sup>7</sup>In other examples from the development literature, [Camacho and Conover \(2011\)](#) find that after a program eligibility decision rule was made transparent, it was manipulated by an amount corresponding to 7% of the National Health and Social Security budget. They note, “there is anecdotal evidence of people moving or hiding their assets, or of borrowing and lending children.” And [Niehaus et al. \(2013\)](#) find that when implementing agents can be corrupted, considering additional poverty indicators can worsen the targeting of benefits, by making it more difficult to verify eligibility.

<sup>8</sup>For instance, [Bruckner and Scheffer \(2011\)](#) study adversarial prediction when the agent acts in response to an observed predictive model, with an application to spam filtering.

distribution differs from the training distribution (Sayed-Mouchaweh and Lughofer, 2012). The manipulation we consider induces the conditional distribution  $y|\mathbf{x}$  to change endogenously when action is taken based on the estimated relationship.

Thus, papers from a variety of sub-literatures have confronted the notion that agents will act strategically when their actions are used to determine allocations. Relative to prior work, our paper makes two main contributions. First, we develop an equilibrium model of decision rule manipulation that can be estimated, which yields rules that function well under manipulation even when fully transparent. And second, to our knowledge for the first time in any literature, we design and implement a field experiment that stress-tests such decision rules in a real-world setting with incentivized agents.

## 2 Theory

This section introduces the model underlying our estimator, and demonstrates its intuition with simulations.

### 2.1 Model

A policymaker observes a *training sample*, i.e., a subset of cases that possess both features  $\mathbf{x}_i$  and optimal decisions  $y_i$ . The policymaker also obtains information on the costs of manipulating features, which will be detailed later. The policymaker would like to estimate the parameters of a decision rule  $\hat{y}(\mathbf{x}_i)$  for cases in an *implementation* subset where only features  $\mathbf{x}_i$  are observed, and may be manipulated.

The policymaker has a preferred action  $y_i$  for each individual  $i$ , denominated in units of individuals' utility. The action  $y_i$  can be projected onto  $i$ 's bliss behavior  $\underline{\mathbf{x}}_i$  by the equation  $y_i = a + \mathbf{b}'\underline{\mathbf{x}}_i + e_i$ , with  $e_i \perp \underline{\mathbf{x}}_i$  representing idiosyncratic preference.

However, the policymaker only observes the individual's actual behavior  $\mathbf{x}_i$ , which may differ from their bliss level  $\underline{\mathbf{x}}_i$ . They select a deterministic decision rule:

$$\hat{y}(\mathbf{x}_i) = \alpha + \beta' \mathbf{x}_i$$

Individuals can manipulate their behavior  $\mathbf{x}_i$  away from their bliss level  $\underline{\mathbf{x}}_i$  at some

cost. Each individual earns utility from the policymaker's decision, minus this cost:

$$u_i(\hat{y}, \mathbf{x}_i) = \hat{y}(\mathbf{x}_i) - c_i(\mathbf{x}_i, \underline{\mathbf{x}}_i)$$

For simplicity, we consider the case where the utility from the decision exactly coincides with the policymaker's prediction, though this approach could be extended to more general utility functions.<sup>9</sup>

Individuals  $i$  are heterogeneous in two respects: bliss behaviors  $\underline{\mathbf{x}}_i$  and gaming ability  $\gamma_i$ . Manipulation costs are quadratic:

$$c_i(\mathbf{x}_i, \underline{\mathbf{x}}_i) = \frac{1}{2}(\mathbf{x}_i - \underline{\mathbf{x}}_i)' C_i (\mathbf{x}_i - \underline{\mathbf{x}}_i)$$

for matrix  $C_i$ :

$$C_i = \frac{1}{\gamma_i} \begin{bmatrix} c_{11} & \cdots & c_{1K} \\ \vdots & \ddots & \vdots \\ c_{K1} & \cdots & c_{KK} \end{bmatrix}$$

Some behaviors may be harder to manipulate than others, either by themselves (the diagonal  $c_{kk}$ ) or in conjunction with other behaviors (the off-diagonals  $c_{kj}$ ). Different people may also find it easier or harder to manipulate ( $\gamma_i$ ); for example, based on a person's tech savviness or opportunity cost of time.

When  $i$  knows the decision rule  $\hat{y}(\mathbf{x}_i)$  and receives benefits according to it, first order conditions imply they will manipulate behavior to level:

$$\mathbf{x}_i^*(\boldsymbol{\beta}) = \arg \max_{\mathbf{x}_i} u_i(\hat{y}, \mathbf{x}_i) = \underline{\mathbf{x}}_i + C_i^{-1} \boldsymbol{\beta}$$

When behavior is not incentivized ( $\boldsymbol{\beta} = \mathbf{0}$ ), optimal behavior equals the bliss level ( $\mathbf{x}_i^*(\mathbf{0}) = \underline{\mathbf{x}}_i$ ). However, as  $\boldsymbol{\beta}$  moves away from zero, behavior moves in the same direction, down-weighted by the cost of manipulation (highlighted in blue).

---

<sup>9</sup>That is, we consider the case where the utility of the decision  $u(\hat{y}) = \hat{y}$ , to match our experimental setting. Under more general utility functions, our model could be considered a linear approximation.

## Decision rules

If, during implementation, the policymaker knew each individual  $i$ 's cost matrix  $C_i$ , they could invert any manipulation to infer the individual's type. However, each individual's cost matrix is not typically known, which leads to information loss (Frankel and Kartik, 2019). We assume that during implementation the policymaker only observes behavior  $\mathbf{x}_i$ , but during model training the policymaker also obtains some knowledge about costs, believing  $i$ 's costs are distributed  $C_{iq} \sim \mathbb{C}_i$  for a random draw  $q$ . We demonstrate a way to recover these beliefs using experiments, and in later sections consider alternate approaches to estimating these costs (such as polling domain experts or deriving from first principles).

The policymaker faces expected squared loss:

$$L(\alpha, \boldsymbol{\beta}) = \mathbb{E}_{i,q} \left[ \left( y_i - \hat{y}(\hat{\mathbf{x}}_{iq}(\hat{y}(\cdot))) \right)^2 + M(\cdot) \right]$$

where  $\hat{\mathbf{x}}_{iq}(\boldsymbol{\beta}) = \underline{\mathbf{x}}_i + C_{iq}^{-1}\boldsymbol{\beta}$ . The first term in the above expectation represents the squared loss *in the counterfactual* where  $\boldsymbol{\beta}$  is implemented and agents manipulate behavior. This accounts for the fact that implementing  $\boldsymbol{\beta}$  will induce behavior to shift away from its distribution in the training data. If the policymaker cares about the costs that individuals incur manipulating behavior, they may include additional penalty  $M(\cdot)$ .

Our **strategy-robust decision rule** is given by:

$$\boldsymbol{\beta}^{SR} = \arg \min_{\alpha, \boldsymbol{\beta}} \mathbb{E}_q \left[ \frac{1}{N} \sum_i (y_i - \alpha - \boldsymbol{\beta}'(\underline{\mathbf{x}}_i + C_{iq}^{-1}\boldsymbol{\beta}))^2 + \dots \right] \quad (1)$$

which deviates from ordinary least squares by the manipulation term  $C_{iq}^{-1}\boldsymbol{\beta}$ . Additional terms ' $\dots$ ' may include  $M(\cdot)$  or regularization terms  $R_{\lambda^{decision}}(\cdot)$ .

## Discussion

Our estimator coincides with nonlinear least squares in the simple case where the policymaker knows costs in the training sample, only cares about targeting performance ( $M(\cdot) \equiv 0$ ), and there are no additional regularization terms ( $R(\cdot) \equiv 0$ ). First order

conditions for  $\beta$  are then given by:

$$\mathbb{E}_{i,q} \left[ \varepsilon_i(\beta, \hat{\mathbf{x}}_{iq}(\beta)) \left( \frac{\partial \varepsilon_i'}{\partial \beta} + \frac{\partial \varepsilon_i'}{\partial \mathbf{x}} \frac{\partial \hat{\mathbf{x}}_{iq}}{\partial \beta} \right) \right] = \mathbf{0}$$

$$\varepsilon_i(\beta, \mathbf{x}) = y_i - \alpha - \beta' \mathbf{x}$$

where the first term captures how  $\beta$  affects fit holding  $\mathbf{x}_i$  constant, and the second term accounts for manipulation: the influence of  $\beta$  on  $\mathbf{x}_i$ . This results in moment condition:

$$\mathbb{E}_{i,q} [\hat{\mathbf{x}}_{iq}(\beta) \cdot \varepsilon_i(\beta, \hat{\mathbf{x}}_{iq}(\beta))] = -\mathbb{E}_{i,q} [C_{iq}^{-1} \beta \cdot \varepsilon_i(\beta, \hat{\mathbf{x}}_{iq}(\beta))] \quad (2)$$

This contrasts with standard estimators, which do not account for manipulation. For example, ordinary least squares (OLS) selects  $\beta$  such that errors are orthogonal to observed features in the training data:  $\mathbb{E}_i [\mathbf{x}_i \cdot \varepsilon_i(\beta, \mathbf{x}_i)] = 0$ . Our estimator differs in three ways.

First, it anticipates the best response *levels* of behaviors—the left hand side is akin to OLS except with counterfactual behaviors  $\hat{\mathbf{x}}_{iq}(\beta)$ . When these behaviors cannot be manipulated ( $C_i \rightarrow \infty$ ), our estimator corresponds to OLS. If all people have the same gaming ability, manipulation may simply shift behaviors, without damaging their information content. Alternately, if desired targets find it easier to game, their shift in behavior can make them more distinguishable, and manipulation itself becomes a signal. Or if desired targets find it harder to game, manipulation may confound the desired targets with people who have high ability to game.

Second, it anticipates the best response *spread* of behaviors. In practice, there will be uncertainty about each  $i$ 's gaming ability and thus  $\hat{\mathbf{x}}_i(\beta)$ . The moment expectations are taken over the distribution of gaming ability, so this variance will tend to attenuate coefficients, accounting for how manipulation ‘muddles’ information.<sup>10</sup>

Third, it anticipates the *gradient* of those behaviors: how they would shift in the subgame if an alternate  $\beta$  were selected. This is captured in the right hand side, which

---

<sup>10</sup>To obtain a simple, estimable model, our parametrization allows the variance in manipulability of behavior  $k$  to scale with its manipulation cost.

deviates from orthogonality. OLS assumes that  $\mathbf{x}_i$  will remain fixed, and in that sense computes a one-step best response. Even if one obtained data from a strategy-robust equilibrium  $(y_i, \mathbf{x}_i^*(\boldsymbol{\beta}^{SR}))$ , OLS will not generally yield the strategy-robust estimate.  $\boldsymbol{\beta} = \boldsymbol{\beta}^{SR}$  is a solution to  $\mathbb{E}_i [\mathbf{x}_i^*(\boldsymbol{\beta}^{SR})\varepsilon_i(\boldsymbol{\beta}, \mathbf{x}_i^*(\boldsymbol{\beta}^{SR}))] = \mathbf{0}$  only if the right hand side of Equation (2) is zero. In contrast, our estimator anticipates that a change in  $\boldsymbol{\beta}$  will induce behavioral responses. This results in a subgame-perfect equilibrium, which maximizes performance in the subgame where the resulting rule is manipulated. This relies on a commitment to not exploit all of the partial equilibrium correlations in the observed data.<sup>11</sup>

When the policymaker cares about not only the resulting allocation, but also the manipulation costs that individuals incur, this is captured by the term  $M(\cdot)$ . A policymaker that is narrowly concerned with its own objective may thus select different decision rules from one that maximizes social welfare (a profit maximizing firm may be satisfied with an equilibrium where all individuals expend welfare gaming a test; a social planner may not be).

To reduce overfitting in small samples, one may also include common forms of regularization; for example,  $R_{\lambda^{decision}}^{LASSO}(\boldsymbol{\beta}) = \lambda^{decision} \sum_k |\beta_k|$  or  $R_{\lambda^{decision}}^{ridge}(\boldsymbol{\beta}) = \lambda^{decision} \sum_k \beta_k^2$ , for regularization hyperparameter  $\lambda^{decision}$ . Under these regularization terms, when  $M(\cdot) \equiv 0$  and  $C_i \rightarrow \infty$  the resulting estimator corresponds to LASSO or ridge, respectively.

## 2.2 Intuition

We demonstrate the method with Monte Carlo simulations. This involves deriving desired payments  $\mathbf{y} = a + \mathbf{b}'\mathbf{x} + \mathbf{e}$ , then assessing the decision rules  $\hat{y}(\mathbf{x})$  generated with different estimators. To focus on the intuition, this section assumes that the policymaker is able to recover the manipulation costs of each individual in its training sample ( $C_{iq} \equiv C_i$ ), but not in implementation.

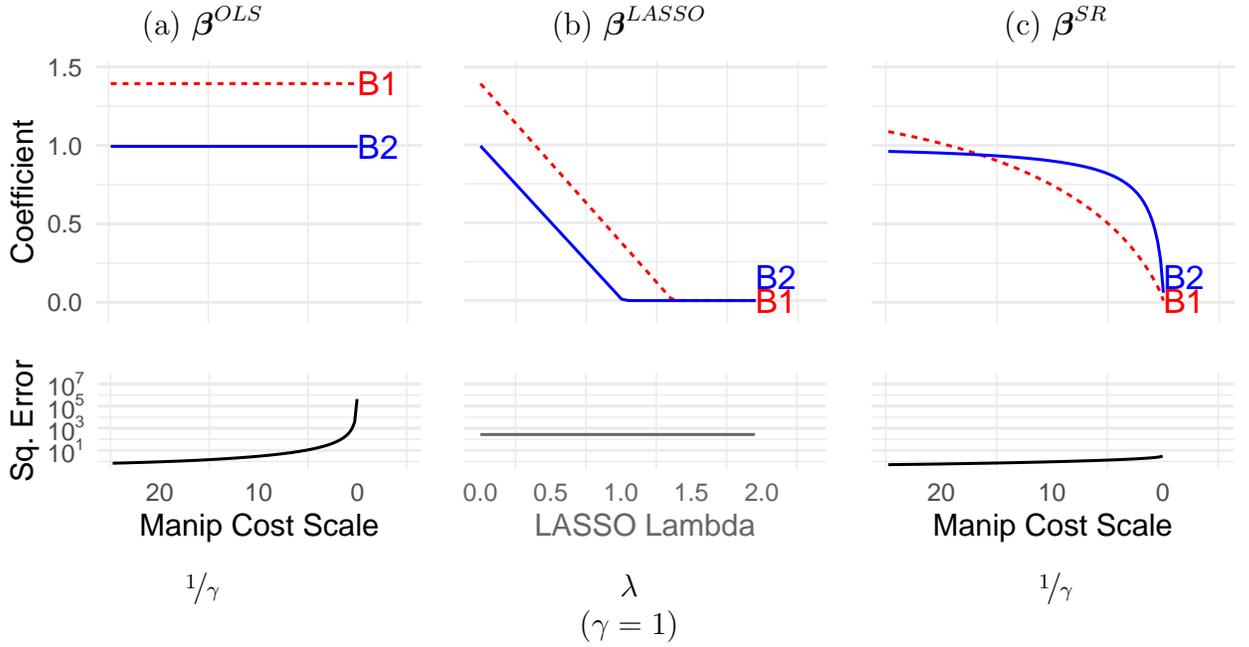
### Comparative statics

We consider a case where  $x_1$  is more predictive than  $x_2$  in baseline behavior, but would be easily manipulated if used in a decision rule ( $b_1 > b_2$  but  $c_{11} \ll c_{22}$ ).

---

<sup>11</sup>See Ball (2019) and Frankel and Kartik (2020) for a theoretical discussion of this distinction.

Figure 1: Common vs. Strategy Robust Estimators



*Note:* The first behavior is more predictive ( $b_1 > b_2$ ), but is easily manipulable ( $c_{11} \ll c_{22}$ ). **(a)** OLS performance deteriorates when behavior can be manipulated. **(b)** LASSO penalization favors  $x_1$ , which will be manipulated as soon as the decision rule is implemented. **(c)** Our method anticipates that  $x_1$  will be manipulated, and shifts weight to  $x_2$  as behavior becomes manipulable.

---


$$\underline{x}_i \stackrel{iid}{\sim} N\left(\mathbf{0}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right), \mathbf{b} = \begin{bmatrix} 1.4 \\ 1 \end{bmatrix}, \mathbf{C}_i = \frac{1}{\gamma_i^{het}} \begin{bmatrix} 4 & 0 \\ 0 & 32 \end{bmatrix}, \frac{1}{\gamma_i^{het}} \stackrel{iid}{\sim} Uniform[0, 10],$$

$e_i \stackrel{iid}{\sim} N(0, 0.25)$ . Squared error measured on an out of sample draw from the same population, incentivized to that decision rule.

Figure 1 compares our method to OLS and LASSO, which both place most weight on  $x_1$ . OLS maximizes predicted performance within the unincentivized sample  $(\mathbf{x}_i(\mathbf{0}), y_i)$ ; as shown in Figure 1a, it performs poorly as manipulation becomes easier. Figure 1b shows that for a given cost of manipulation, LASSO shrinks these coefficients. However, standard regularization does not consider manipulation, and LASSO selection does the wrong thing: it kicks  $x_2$  out of the regression first. In contrast, our method considers how predictive features will be in equilibrium when the decision rule is implemented:  $(\mathbf{x}_i(\boldsymbol{\beta}), y_i)$ . As shown in Figure 1c, when manipulation costs are high, our method approaches OLS; as manipulation becomes easier, our method substantially penalizes  $x_1$ .

The Supplemental Appendix (section 4.1) presents additional comparative statics. If each feature is equally costly to manipulate ( $c_{jl} \equiv c_{kl}$ ), our method shrinks them together, similar to ridge regression. If all individuals have the same gaming ability ( $\gamma_{iq} \equiv \gamma$ ), then manipulation shifts behavior uniformly; although this does not affect predictive performance, individuals may incur substantial costs manipulating.<sup>12</sup>

## Performance

Table 1 shows the results of an example Monte Carlo simulation, chosen to contrast our method with standard approaches. In this simulation, type  $\underline{x}_1$  has a large weight in the desired payment ( $b_1 = 3$ ) relative to the other two dimensions ( $b_2 = b_3 = 0.1$ ); however, behavior  $x_1$  is much easier to manipulate ( $c_{11} = 1$  vs.  $c_{22} = 2$  and  $c_{33} = 4$ ). In this environment, OLS considers the static relationship in the unmanipulated data. This rule would perform well if behavior were fixed (the ‘no manipulation’ column); however, once consumers adjust to the rule, it makes terrible decisions (the ‘manipulation’ column).

A common ‘industry approach’ involves retraining the model periodically. Thus, as shown in Panel B of Table 1, if we observe consumers’ new behavior and reestimate OLS, we obtain  $\boldsymbol{\beta}^{OLS(2)}$ , which places negative weight on the manipulated  $x_1$ . However, once consumers respond, the decision rule performs poorly. If we repeatedly allow

---

<sup>12</sup>As the cost of manipulating one particular behavior ( $c_{22}$ ) decreases, it is penalized, and weight is shifted to other behaviors. The method also can exploit cost interactions, penalizing behaviors that make it easier to shift other predictive behaviors (akin to Ramsey (1927) taxation). When manipulating  $x_1$  makes it easier to manipulate  $x_2$  ( $c_{12}$  sufficiently negative), our method further reduces weight on  $x_1$ .

Table 1: Manipulation Can Harm Prediction (Monte Carlo)

	Decision Rule				Performance (squared loss)	
	$\beta_1$	$\beta_2$	$\beta_3$	$\alpha$	No manip.	Manipulation
<i>Panel A: Data Generating Process</i>						
$\mathbf{b}^{DGP}$	3.00	0.10	0.10	0.20	0.27	3745.05
<i>Panel B: Standard Approaches</i>						
$\beta^{OLS}$	3.04	0.06	0.12	0.21	0.27	3961.23
<i>Industry Approach</i>						
$\beta^{OLS(2)}$	0.06	2.09	-1.68	-0.80	3.28	625.76
$\beta^{OLS(3)}$	3.11	-0.04	0.22	0.17	0.27	4332.21
$\beta^{OLS(4)}$	0.12	2.08	-1.67	-0.76	3.07	619.06
$\vdots$						
$\beta^{OLS(1001)}$	3.74	-1.34	1.57	-0.39	1.38	11 611.88
$\beta^{OLS(1002)}$	0.70	1.86	-1.53	-0.40	1.67	565.38
<i>Panel C: Strategy-Robust Method</i>						
$\beta^{SR}$	0.50	0.54	-0.10	-1.81	9.16	1.94
<i>If policymaker knows only the distribution of costs between individuals:</i>						
$\beta_{C_{iq}=\text{bootstrap}_i(C_i)}^{SR}$	0.31	0.49	0.15	-0.74	7.00	3.38
<i>If costs are misestimated:</i>						
$\beta_{C_{iq}\equiv 2 \cdot \text{diag}(C_i)}^{SR}$	0.66	0.72	-0.35	-1.57	6.89	10.83

*Notes:* Monte Carlo simulation results. Panel A shows the coefficients that relate the outcome ( $y$ ) to behaviors ( $\mathbf{x}$ ) under the data generating process (DGP). Panel B shows coefficients from OLS. Panel C shows coefficients estimated with the strategy-robust method with costs known during training ( $C_{iq} \equiv C_i$ ); with heterogeneous costs bootstrapped between individuals over 10 draws; and with costs mis-estimated to be double and to omit off-diagonals. Performance is assessed on the same sample of individuals under behavior with and without manipulation. Parameters:

$$C_i = \frac{1}{\gamma_i} \begin{bmatrix} 1.0 & 0.1 & 0.2 \\ 0.1 & 2.0 & 0.8 \\ 0.2 & 0.8 & 4.0 \end{bmatrix}, \mathbf{x} \stackrel{iid}{\sim} N \left( \mathbf{0}, \begin{bmatrix} 1.0 & 1.0 & 0.1 \\ 1.0 & 2.0 & 1.0 \\ 0.1 & 1.0 & 1.0 \end{bmatrix} \right), \gamma_i = \begin{cases} 1 & \mathbf{x}_{i1} \leq 0.2 \\ 10 & \mathbf{x}_{i1} > 0.2 \end{cases}, e_i \stackrel{iid}{\sim} N(0, 0.25)$$

individuals to best respond, and then estimate the decision rule  $\beta^{OLS(r)}$  using data from the prior period ( $r - 1$ ) (iterative best response), this process continues to make poor decisions. In this case, the process does not converge; it alternates between decision rules that place high and low weight on  $x_1$ .<sup>13</sup> Thus, standard approaches can perform poorly even in stable settings with perfect information. In settings with noise or frictions in learning, a system might unexpectedly and catastrophically fail when the other side discovers how to exploit it.<sup>14</sup>

In contrast, our strategy-robust estimator ( $\beta^{SR}$ ) penalizes the easily manipulable behavior  $x_1$  and shifts weight to behaviors that are harder to manipulate ( $x_2$  and  $x_3$ ). It anticipates manipulation off-path, sacrificing performance in the environment in which it is trained (in-sample, no manipulation) for performance in the counterfactual where there is manipulation. When individuals manipulate as described in the model, our estimator exceeds the performance of other estimators.<sup>15</sup>

Our method performs similarly well when the policymaker knows only the distribution of costs  $\mathbb{C}$  and not the cost of each individual in its training sample (Panel C of Table 1). The method can also reduce risk even if manipulation costs are misestimated. For instance, the last row considers the case where all off diagonal elements are erroneously set to zero, and the estimated costs of manipulation are two times too large. Performance deteriorates relative to the case where we know the true cost matrix, but our method still outperforms OLS in the presence of manipulation.

## Manipulation can improve performance

Manipulation can *improve* performance, if ease of manipulation ( $\gamma_i$ ) is correlated with the outcome ( $y_i$ ). In that case, manipulation represents a signal of the underlying type, as in [Spence \(1973\)](#), and applications of self-targeting ([Nichols and Zeckhauser, 1982](#)). We provide an example in the Supplemental Appendix (Section 4.2) where manipulation improves the performance even of naïve estimators. Our method *increases*

---

<sup>13</sup>These oscillations can be dampened by using cumulative data from all prior periods, as shown in the Supplemental Appendix (section 4). That still takes several iterations to converge to a less performant equilibrium.

<sup>14</sup>For example, [Gonzalez-Lira and Mobarak \(2019\)](#) find that increased enforcement of a ban on selling endangered fish led vendors to learn about, and more effectively undermine, the decision rule.

<sup>15</sup>The strategy-robust estimator can also be combined with industry approach by using the strategy-robust approach first, then iteratively retraining, as shown in the Supplemental Appendix (section 4).

the coefficient on the manipulated behaviors to better exploit the information contained in manipulation, and thus further improves performance. In that sense, if different subgroups in a population (like the rich or poor) are differentially able to manipulate behavior (Hu et al., 2019), our method can utilize that correlation to bring allocations closer to the policymaker’s objective.

### 3 Estimation

This section describes how the strategy-robust model can be estimated with experimental data. We will assume, for now, that it is possible to experimentally incentivize individuals to manipulate different behaviors.<sup>16</sup> Specifically, for each individual  $i$ , we observe multiple time periods  $t \in \mathbb{T}_i$ . Each period we assign  $i$  to a decision rule which pays out based on their behavior that period:  $\hat{y}_{it}(\mathbf{x}_{it}) = \alpha_{it} + \boldsymbol{\beta}'_{it}\mathbf{x}_{it}$ . We designate ‘control’ periods  $\mathbb{T}_i^{\text{control}}$  during which behaviors are not incentivized:  $\boldsymbol{\beta}_{it} \equiv \mathbf{0}$ . In ‘treatment’ periods  $\mathbb{T}_i^{\text{treatment}}$  one behavior  $k \in 1 \dots K$  is drawn at random and incentivized by disclosing a rule that pays  $\beta_{itk} \neq 0$  for each unit of behavior  $k$  but not for other behaviors:  $\beta_{itj} = 0$  for  $j \neq k$ .

In period  $t$ , we allow an individual to deviate from bliss behavior due to manipulation, or shocks that are common ( $\boldsymbol{\mu}_t$ ) or individual-specific ( $\boldsymbol{\epsilon}_{it}$ ):

$$\mathbf{x}_{it}^*(\boldsymbol{\beta}_{it}) = \underline{\mathbf{x}}_i + C_i^{-1}\boldsymbol{\beta}_{it} + \boldsymbol{\mu}_t + \boldsymbol{\epsilon}_{it} \tag{3}$$

where both components are mean zero:  $\mathbb{E}\boldsymbol{\mu}_t = \mathbf{0}$  and  $\mathbb{E}\boldsymbol{\epsilon}_{it} = \mathbf{0}$ .<sup>17</sup>

We estimate strategy-robust decision rules in two steps.

---

<sup>16</sup>Note that such a procedure could still induce meta-manipulation: if training users had preferences over the algorithm that implementation users ultimately faced, they could pretend to make some behaviors more or less manipulable. We expect incentives to do this to be small, and that these correlations are sufficiently complex that it is unlikely that users would know which direction to game their training behavior. In extensions we discuss alternate approaches to measuring costs that would be robust to this concern.

<sup>17</sup>This arises from the utility function  $u_{it} = \hat{y}_{it}(\mathbf{x}_{it}) - c_i(\mathbf{x}_{it}, \underline{\mathbf{x}}_i) + (\boldsymbol{\mu}_t + \boldsymbol{\epsilon}_{it})'C_i(\mathbf{x}_{it} - \underline{\mathbf{x}}_i)$ .

### 3.1 Primitives

We first estimate primitives: types  $\underline{\mathbf{x}}$ , cost parameters  $\boldsymbol{\omega}$  and  $C^{-1}$ , and the distribution of unobserved gaming ability  $V$ .

#### Types

We estimate types ( $\underline{\mathbf{x}}$ ) and time period fixed effects ( $\boldsymbol{\mu}$ ) from control periods, using the ordinary least squares regression:

$$\mathbf{x}_{it} = \underline{\mathbf{x}}_i + \boldsymbol{\mu}_t + \boldsymbol{\epsilon}_{it} \tag{4}$$

including only time periods where  $\boldsymbol{\beta} = \mathbf{0}$  (in which people act as their types).

#### Costs

We parameterize the cost matrix as:

$$C_{iq} = \frac{1}{\gamma_{iq}} \cdot C$$

allowing heterogeneity by behavior, and by individual:

$$\gamma_{iq} = e^{-\boldsymbol{\omega}'\mathbf{z}_i} + v_q$$

Individual gaming ability can vary with characteristics  $\mathbf{z}_i$  (observed in the training sample but not in implementation), and due to unobserved heterogeneity  $v_q \sim V$  (where  $\mathbb{E}v_q = 0$ ).

We estimate  $C$  and  $\boldsymbol{\omega}$  using experimental variation in incentives and a general method of moments (GMM) loss function. We impose  $\underline{\mathbf{x}}$  and  $\boldsymbol{\mu}$ . We limit the potential to overfit using two adjustments. First, we impose the constraint that behaviors move in the direction they are incentivized:  $c_{jj} > 0$ . Second, we penalize the ease of manipulation towards zero (cost towards infinity), which regularizes towards standard methods (OLS/LASSO/etc). We use ridge style penalization, allowing separate hyperparameters for diagonal and off-diagonal costs ( $\boldsymbol{\lambda}^{costs} = \{\lambda_{diagonal}^{costs}, \lambda_{of\,fdiagonal}^{costs}\}$ ). In our application we will penalize off diagonal elements to zero because of noise in estimating them, and use three-fold cross validation to select  $\lambda_{diagonal}^{costs}$ .

We recover the distribution of unobserved gaming ability  $V$  by shrinking and then shuffling the gaming ability residuals. For more details, see Appendix A1.

### 3.2 Decision Rules

Given these primitives, a strategy robust decision rule is given by:

$$\boldsymbol{\beta}^{SR} = \arg \min_{\alpha, \boldsymbol{\beta}} \left[ \frac{1}{N} \sum_i \left[ \frac{1}{Q} \sum_q [y_i - \alpha - \boldsymbol{\beta}'(\mathbf{x}_i + C_{iq}^{-1}\boldsymbol{\beta})]^2 + R_{decision}^{\lambda}(\boldsymbol{\beta}, \mathbf{y}, \mathbf{C}_q) \right] \right]$$

for  $Q$  random draws of the cost matrix  $C_{iq}$ , where draws  $v_q \sim V$  are treated as random effects. This loss includes any regularization term added to the decision rule itself  $R^{\lambda^{decision}}(\cdot)$ ; we set the regularization hyperparameter  $\lambda^{decision}$  with cross validation in the unmanipulated baseline sample, where we have more data.

## 4 Experiment

We designed a field experiment to test the performance of our strategy-robust estimator in a real-world setting. Design started in 2017. Working with the Busara Center for Behavioral Economics in Nairobi, we developed and deployed a new smartphone-based application (‘app’) to 1,557 research subjects. The app was designed to mimic key features of ‘digital credit’ apps that are quickly transforming consumer credit in developing countries (Francis et al., 2017). In Kenya, at the time of our study, CGAP (2018) estimates that 27% of all adults had an outstanding ‘digital credit’ loan.

This section describes the app and experimental design; estimates costs of manipulation and derives strategy-robust decision rules using our method; and compares the performance of these new estimators to traditional learning algorithms. Our design was pre-specified in a pre-analysis plan registered in the AEA RCT registry under AEARCTR-0004649.

## 4.1 Experimental design and smartphone app

Our experiment is intended to create incentives similar to those of a digital credit lending app. These apps run in the background on a smartphone, and collect data on phone use (including data on communications, mobility, social media behavior, and much more). Digital credit apps use this information to allocate loans to people who appear creditworthy (i.e., for whom  $\hat{y}_i$  exceeds some threshold; Björkegren (2010); Björkegren and Grissen (2019)). Since financial regulations prevented us from actually underwriting loans to research subjects, we instead focused on analogous problems where a decisionmaker wishes to allocate resources to individuals with specific characteristics—for instance, by paying individuals who have a certain income level, or other characteristic (e.g., intelligence, level of activity, education).<sup>18</sup>

### Smartphone app

The ‘Smart Sensing’ app we worked with Busara to build has two key features. First, it runs in the background to capture anonymized metadata on how individuals use their phones, such as when calls or texts are placed, which apps are installed and used, geolocation, battery usage, wifi connections, and when the screen was on. In total, we extract over  $\bar{K} > 1,000$  behavioral indicators (“features”). Second, it delivers weekly “challenges” to users (see Figure 2). These challenges appear on the user’s phone, and provide financial rewards based on the user’s behavior. The challenges can be very simple (‘You will receive 12 Ksh. for every incoming call you receive this week’) or more complex (‘Earn up to 1000 Ksh. if the Sensing app guesses you are a high-income earner’). Users are paid a base amount of 100 Ksh. for uploading data, plus any challenge winnings, directly via mobile money at the conclusion of each week.

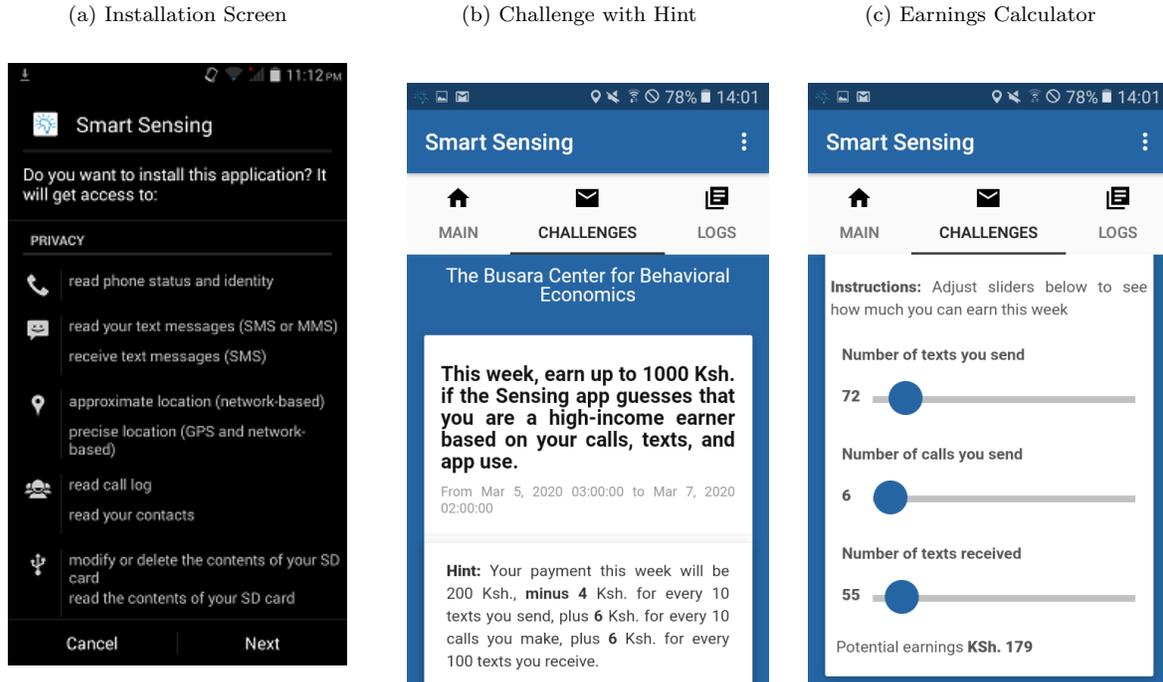
### Study population and recruitment

The subject population consists of Kenyans aged 18 years or older who own a smartphone and were able to travel to the Busara center in Nairobi. Participants were recruited in person in public spaces in Nairobi, and were sequentially invited for an

---

<sup>18</sup>While these prediction targets differ from credit-worthiness, there are many settings where similar characteristics are inferred by digital traces (for example, social assistance programs that target the poor (Aiken et al., 2021), or digital advertisers who target college students).

Figure 2: Smart Sensing App



enrollment session at the Busara center. During enrollment, participants completed a survey, which captured ground truth characteristics that we later seek to infer based on phone usage behavior.

Prospective participants were asked to keep the Sensing App on their phones for about 16 weeks. During the informed consent process, participants were told the dimensions of behavior that would be recorded by the app, and were given the opportunity to ask questions. Participants had the opportunity to view the Android permissions required for the app to function properly, and generally appeared to understand the privacy tradeoffs involved in participation. Our sample includes only participants who opted in. 83% of participants elected to receive challenges in English, 16% in Swahili, and 1% in both languages.

During onboarding, we discussed with participants strategies for altering different types of phone behavior, surfaced from prior focus groups (Musya and Kamau, 2018). We included this discussion to mimic what might be observed in the long run after individuals discover the easiest ways to manipulate these indicators.

## Weekly rhythm

The study followed a weekly rhythm. Each Wednesday at noon, each participant received a generic notification on their phone that said, ‘Opt in to see this week’s challenge!’ If the participant opened the app and opted in, they were shown the details of that week’s challenge (see Figure 2).<sup>19</sup> Challenge incentives were valid until 1pm Tuesday. At the conclusion of the challenge, participants had 21 hours to ensure that their data was uploaded (until 10am Wednesday). Busara then determined how much each participant should be paid, and payments were sent via mobile money by noon Wednesday, at which point the next week’s cycle would begin.

Each week, participants could attrite in two ways: by not uploading their data, or by not opting in to the challenge. These participants were sent text message reminders or called by Busara staff, following an attrition protocol detailed in the Supplemental Appendix (Section 1.4). We include in our analysis only participant-weeks where the participant opted in and uploaded during the end-of-week upload window.

## 4.2 Baseline predictions and model estimation

### Predicting user characteristics

During the first several weeks of the experiment, participants were observed but were not incentivized to change behavior.<sup>20</sup> Using data from these ‘control’ weeks, we find that baseline phone behaviors have a (weak) predictive relationship with participant characteristics. We focus on two primary characteristics ( $\tilde{y}_i$ ): monthly income, and intelligence (above-median performance on Raven’s matrices).<sup>21</sup> Results for these outcomes, based on OLS, are shown in Table 2:  $R^2$  are approximately 0.03.<sup>22</sup> Because models with many coefficients can be difficult for participants to interpret, we will use

---

<sup>19</sup>To minimize the possibility of differential attrition, the pre-opt-in notification was the same for all participants regardless of their assigned challenge.

<sup>20</sup>In these weeks, the subject received a challenge of the form, ‘Dear user, you do not have to do anything for this week’s challenge. You will receive an extra Ksh 50 for accepting this challenge.’

<sup>21</sup>In addition to monthly income and intelligence, we conducted experiments to predict whether a person is married, has advanced technology skills, has many friends, is below median age, or communicates a lot (several different measures of total phone activity). Pooled results for all characteristics are provided in the Supplemental Appendix (SA Table S1).

<sup>22</sup>These  $R^2$ ’s are fairly low, likely due to the fact that we have a small sample of relatively homogeneous users, observed for short time spans, and that our probes are relatively limited.

Table 2: Behavior Predicts Individual Characteristics

	Monthly Income		Intelligence (Above Median Ravens)	
Mean Duration of Evening Calls	-0.559	(3.702)	0.0001	(0.0002)
Mean Duration of Outgoing Calls	-1.770	(8.965)	-0.0007	(0.0004)*
Calls with Non-Contacts	-42.023	(14.033)***	••	-0.002 (0.0006)***
Outgoing Text Count	••	10.211 (12.396)	0.0004	(0.0006)
Incoming Text Count	•	3.888 (7.974)	••	-0.0002 (0.0004)
Evening Text Count	•	-9.029 (7.815)	-0.0002	(0.0003)
Outgoing Call Count	••	76.752 (18.133)***	0.002	(0.0008)*
Missed Outgoing Call Count	-84.533	(31.636)***	•	-0.003 (0.0014)**
Outgoing Texts on Weekdays	-15.023	(15.210)	-0.001	(0.0007)
Max Daily Incoming Text Count	2.901	(21.212)	•	0.003 (0.0009)***
Intercept	5651.04	(430.141)***	0.480	(0.019)***
N (individuals)	1539		1557	
$R^2$	0.026		0.027	

*Notes:* Each column represents a regression of the outcome characteristics (column header) on behaviors measured through the Sensing app (rows) Observations include data collected during the first week the participant used the sensing app. Standard errors in parentheses. \* = 10 percent significance, \*\* = 5 percent significance, \*\*\* = 1 percent significance. • : included in incentivized naive LASSO model, •• : included in incentivized strategy-robust (SR) model.

LASSO penalization to restrict to three-variable decision rules.

We use these control weeks to estimate types  $\underline{x}$  for each participant using Equation (4), with week fixed effects to absorb idiosyncratic weekly shocks.

### Evidence that app-based challenges induce manipulation

During the main phase of the experiment, we randomized participants into groups that received incentives to change specific behaviors. The ‘simple’ challenges were of the form, ‘We’ll pay you  $\beta_j$  for each additional  $x_j$  you do’, where behavior  $j$  and amount  $\beta_j$  are assigned randomly. For example, one challenge was, “You will receive 3 Ksh. for each text you send this week, up to Ksh. 250.” We restricted consideration to behaviors  $j$  that had some predictive relationship to participant types at baseline (such as those shown in Table 2).<sup>23</sup> Payout levels  $\beta_j$  were drawn for each participant

<sup>23</sup>Specifically, we run LASSO regressions for each characteristic to select models with three behavioral predictors. We included all selected behaviors and similar behaviors (correlates, different measures of the same concept, or behaviors selected by LASSO if the original behavior was omitted). For example, if the original regression selects outgoing calls, we also include incoming calls. Note that by including only a subset of variables, our procedure implicitly assumes that omitted variables

at random; most incentives were positive but some were negative (participants were incentivized to reduce behavior).<sup>24</sup>

Participants’ behavior changed in response to these simple challenges. Table 3 presents a regression of each participant’s weekly level of different behaviors (columns) on randomly assigned incentives to change them (rows). There are three takeaways. First, individuals manipulated the behaviors that were incentivized, as shown by the diagonal, which is positive and significant for most behaviors. Second, some behaviors were more manipulable than others. For example, the number of texts sent was 49 times more responsive to incentives than the number of people called during the workday. And finally, incentivizing one behavior can affect others, as shown in the off diagonal elements. For example, incentivizing missed incoming calls also increased the number of texts sent (possibly because people sent messages to ask their contacts to call them back). In theory, our method can exploit these cross-elasticities, though many are noisily estimated in our data.

In Section 5.3, we evaluate other methods for measuring manipulation costs. In the Supplemental Appendix (section 2.1), we show that the quadratic cost assumption is a reasonable (if imperfect) approximation of how people respond to variable incentive amounts.

## Estimation

We next use the data from these first parts of the experiment to estimate manipulation costs. We allow the distribution of manipulation cost  $\mathbb{C}(z_i)$  to differ by whether a person reports having high tech skills ( $z_i \in \{0, 1\}$ ), and by an unobserved random effect  $v_q$ .<sup>25</sup> Table 4 summarizes these estimated costs for behaviors selected by our models (for all behaviors, see Appendix Table A1). With our sample size, we found that off-diagonal elements were noisily estimated, so we penalized them to zero ( $\lambda_{offdiagonal}^{costs} \rightarrow \infty$ ); this results in a diagonal cost matrix  $C$ .

Several intuitive patterns to the costs of manipulation can be seen in the top

---

are costless to manipulate.

<sup>24</sup>Each individual’s payment level for  $j$  was drawn from  $\{-2r_j, -r_j, r_j, 2r_j, 4r_j, 8r_j\}$ , for scalar  $r_j$ . We scaled the payout for each behavior so that the maximum payout could be achieved by someone reaching the 90th percentile of baseline behavior.

<sup>25</sup>Tech skills explained the most heterogeneity in preliminary analysis; Spence signaling will only be captured in this dimension of heterogeneity.

Table 3: Behavior Changes when Incentivized

Behavior incentivized	Behavior observed (change per ¢ of incentive)				
	# Texts sent	# Missed calls (outgoing)	# Missed calls (incoming)	# People called (Workdays, i.e. M-F, 9am-5pm)	# Calls w non-contacts (weekends)
# Texts sent	<b>24.51</b> (3.202) <sup>***</sup>	-0.052 (0.588)	-0.836 (0.87)	-0.305 (0.217)	-0.022 (0.368)
# Missed incoming calls	4.16 (2.196) <sup>*</sup>	<b>0.709</b> (0.403) <sup>*</sup>	0.825 (0.597)	0.128 (0.252)	-0.002 (0.995)
# Missed outgoing calls	-0.206 (2.856)	0.324 (0.524)	<b>1.187</b> (0.776)	0.22 (0.194)	0.502 (0.328)
# People called (workday)	2.308 (2.505)	0.156 (0.46)	0.68 (0.681)	<b>0.497</b> (0.17) <sup>***</sup>	0.108 (0.288)
# Calls w non-contacts (weekends)	-2.019 (2.866)	-0.056 (0.526)	1.234 (0.779)	0.015 (0.194)	<b>1.233</b> (0.329) <sup>***</sup>
Individual Fixed Effects	X	X	X	X	X
Week Fixed Effects	X	X	X	X	X
N (person-weeks)	7966	7966	7966	7966	7966
$R^2$	0.704	0.522	0.637	0.604	0.491

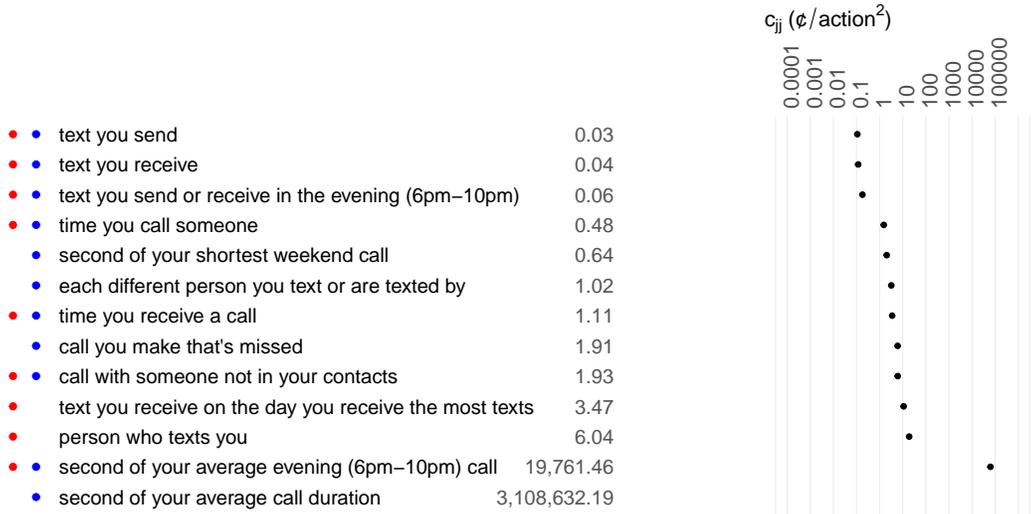
*Notes:* Standard errors in parentheses. Bold indicates diagonal: effect on behavior  $j$  when behavior  $j$  is incentivized. Each column represents a separate regression over the full set of behaviors assigned; only the first five coefficients reported here. N represents person-weeks during which ‘simple’ (single behavior) challenges were issued. \*  $p < 0.1$ , \*\*  $p < 0.05$ , \*\*\*  $p < 0.01$ .

panel of Table 4. Outgoing communications are less costly to manipulate than incoming communications. Text messages, which are relatively cheap to send, are more manipulable than calls, which are relatively expensive. Simpler behaviors (such as the number of texts sent) are more manipulable than complex behaviors (such as the standard deviation of texts sent by day; see Appendix Table A1).

Costs are also heterogeneous across people, as shown in the bottom panel of Table 4. On average it is 9% easier for individuals who report advanced or higher tech skills to manipulate behaviors. Including unobserved heterogeneity, the 90th percentile of gaming ability finds it twice as easy to game as the 10th percentile.

Table 4: Estimated Manipulation Costs

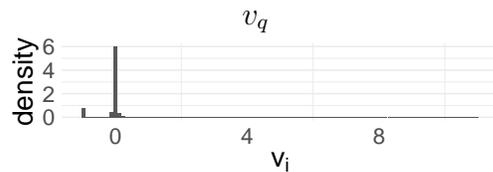
**Heterogeneity by Behavior** (C diagonal; subset of behaviors selected by models)



**Heterogeneity by Person** ( $\gamma_{iq}$ )

$$\gamma_{iq} = e^{-\omega z_i} +$$

Low tech skills	1.000
High tech skills	1.087



Parameters estimated using GMM. Top panel shows only behaviors used in models (• : naive LASSO, • : strategy-robust (SR)); for all behaviors see Appendix Table A1. In cost matrix, off diagonal elements regularized to zero ( $\lambda_{offdiagonal}^{costs} \rightarrow \infty$ ), diagonal elements regularized with  $\lambda_{diagonal}^{costs} = 1.0$ , set via cross validation.  $v_q$  plot omits top 5 percent of observations.

### 4.3 Results: Naïve vs. Robust Decisions

The final and most important stage of the experiment compares decisions made by standard machine learning algorithms to the decisions made by our new strategy-robust estimator that accounts for the costs of manipulating behavior. The robust decision rules can be directly estimated with Equation (1), which relies on  $\underline{\mathbf{x}}_i$  and  $\mathbf{C}_i$  that come from previous stages of the experiment.<sup>26</sup>

In this final stage, subjects received ‘complex’ challenges that rewarded them for their ultimate classification. These challenges were designed to mimic real-world applications of machine learning, where people can receive a desirable benefit as a result of their classification, such as a loan (digital credit) or a grant (targeted aid). In our experiment, the challenges were of the form, ‘We’ll pay you  $m$  if you are classified as  $\hat{y}$ .’ Our analysis highlights responses to the challenge, ‘Earn up to 1000 Ksh. if the Sensing app guesses you are a high-income earner’; pooled results from other complex challenges are provided in the Supplemental Appendix.<sup>27</sup>

#### Estimating Decision Rules

In order to keep decision rules simple and interpretable for our participants, we restricted consideration to decision rules with at most three predictors, using LASSO penalization.<sup>28</sup> The distribution of unobserved gaming ability  $V$  is affected by a shrinkage parameter, which we calibrated based on performance on the first few weeks of decision rules (see Appendix A1).

---

<sup>26</sup>For estimating decision rules we used  $\underline{\mathbf{x}}_i$  equal to the simple average of  $\mathbf{x}_i$  during control weeks (without week fixed effects). Due to the tight experimental timeline, the implemented decision rules were derived from preliminary estimates of  $\mathbf{C}_i$ . The main tables report the decision rules as assessed by final cost estimates (as shown in the Supplemental Appendix, section 2.2, decision rules resulting from preliminary and final cost estimates are similar). The main analysis further omits select weeks when upload servers were offline and there was a mistake in computing the heterogeneity parameter; the Supplemental Appendix (section 2.2) shows that our results are robust to their inclusion.

<sup>27</sup>Full results are available at <https://dan.bjorkegren.com/manipulation-appendix-extra.pdf>. We map characteristics,  $\tilde{y}$ , into payouts,  $y$ , with transformation  $y_i = \max\{0, \min\{1000, \tilde{y}_i \cdot \frac{1000}{\tilde{y}_j^{(90\%)}}\}\}$  given  $\tilde{y}^{(90\%)}$ , the 90th percentile of raw outcome  $\tilde{y}$ .

<sup>28</sup>To do this, we regularized naïve LASSO decision rules with  $\lambda^{decision} = \max(\lambda^{cv}, \underline{\lambda}^{3var})$ , where  $\lambda^{cv}$  is the cross-validated penalty parameter and  $\underline{\lambda}^{3var}$  is the smallest that resulted in a 3-variable model. We used the same  $\lambda^{decision}$  to penalize our strategy-robust decision rule, and also allowed it to select only among three variable models.

## Treatments

Participants were randomly assigned into different target outcomes ( $y$ ), decision rules (standard  $\beta^{LASSO}$ , or robust  $\beta^{SR}$ ), and whether the decision rule was opaque or transparent to the user. Under the opaque treatment, users were told only the target outcome and the reward. Under the transparent treatment, users saw the coefficients of the decision rule, which revealed how much they are rewarded for each behavior. In the transparent treatment, we also provided an interactive interface that showed participants how their payments would be calculated from different behaviors (see Figure 2c). Because the transparent treatment revealed information about potential decision rules, after a person had seen a transparent challenge for  $\hat{y}$ , we did not assign them to an opaque challenge for the same outcome.

Table 5 provides suggestive evidence of how decision rule incentives affect behavior. The first panel simply indicates the estimated decision rule: high-income people make more outgoing calls, send fewer texts, and receive more texts. In the second panel, we see that if we pay people to ‘act like a high-income earner’ without revealing the decision rule, the response is not statistically significant and often in the wrong direction on average (i.e., participants place fewer calls and send more texts). However, participants assigned the transparent treatment change their behavior broadly in the direction incentivized by the algorithm, though the response is measured with noise.

## Performance of decision rules

Our main empirical results, shown in Table 6, compare the performance of naïve and strategy-robust decision rules. The first two columns (under ‘Income’) show results for the challenge that incentivized participants to use their phones like a high-income earner; the last two columns show the performance averaged across both the income and intelligence challenges. The decision rules and associated manipulation costs are shown in the top panel (“Decision Rules”); the relative performance of the different estimators is shown below (under “Prediction Error”). We note several results.

First, in Panel A, we observe important differences in the decision rules. LASSO places weight on the behaviors that were most correlated at baseline: outgoing calls, outgoing texts, and incoming texts. However, some of these behaviors, particularly text messaging, are easy to manipulate (as shown in the ‘Costs’ column). Our

Table 5: Agents Game Algorithms

	Calls (outgoing)	Texts (outgoing)	Texts (incoming)	Calls w con-contacts (incoming + outgoing)	Avg call length (evening, seconds)
<b>Panel A: Incentives generated by algorithm (¢/action)</b>					
$\beta^{LASSO}$	0.625	-0.395	0.065	0	0
<b>Panel B: Regression of <math>x_{it}</math> (column label) on treatment assignment (row label)</b>					
Opaque challenge	-4.7 (8.6)	12.5 (17.2)	11.1 (20.7)	0.8 (3.4)	-4.3 (7.1)
Transparent challenge	13.7 (7.9)*	-17.5 (15.7)	-6.5 (19.0)	0.3 (3.1)	-2.1 (6.5)
N (Person-weeks)	1651	1651	1651	1651	1651

*Notes:* Panel A reports the decision rule associated with the challenge, ‘Earn up to 1000 Ksh. if the Sensing app guesses you are a high-income earner!’. Panel B reports how behaviors (indicated by columns) changed when participants were randomly assigned to the opaque challenge (which provided no information about the decision rule) or the transparent challenge (which revealed the details of the decision rule). The sample includes all people who were assigned the income challenge (either opaque, or the transparent LASSO model), in control weeks and the week they were assigned that challenge. Standard errors in parentheses. \*  $p < 0.1$ .

strategy-robust decision rule both selects behaviors that are harder to manipulate (i.e., evening texts rather than incoming texts), and shrinks the importance of more easily manipulated behaviors (especially outgoing texts).

We evaluate predictive performance using root mean squared error (RMSE), in units of US dollars, in Panel B. This measures how far off the payments we gave to people (based on the model and their behavior that week) were from what we desired to give to them (based on their fixed characteristic that we targeted). The first pair of rows report the prediction error that was expected ex ante, based on behavior observed during the control weeks. The first row shows that when there is no manipulation, LASSO is expected to perform marginally better than our strategy-robust estimator (by \$0.01 for income; \$0.005 for income and intelligence pooled). The second row shows the error predicted by our model if the rule were made transparent and people were manipulating behavior: here, the strategy-robust method is expected to perform better (by \$0.09 for income; \$0.05 pooled).

The next pair of rows report the prediction error that we actually obtained when

Table 6: Strategy-Robust vs. Standard Decision Rules

	Income		Costs	Pooled: Income & Intelligence	
	$\beta^{LASSO}$ ¢/action	$\beta^{SR}$	$c_{jj}$ ¢/action <sup>2</sup>	$\beta^{LASSO}$	$\beta^{SR}$
<i>Panel A: Decision Rule</i>					
# Texts (outgoing)	-0.395	-0.107	0.035	.	.
# Texts (incoming)	0.065	0	0.037	.	.
# Texts (6pm-10pm)	0	-0.121	0.057	.	.
# Calls (outgoing)	0.625	0.542	0.480	.	.
Intercept ( $\alpha$ )	301.071	304.622		.	.
<i>Panel B: Prediction Error</i>					
	RMSE (\$)			RMSE (\$)	
Baseline Data: Control	3.574	3.583		4.273	4.278
Baseline Data: Predicted Transparent	3.672	3.585		4.328	4.279
Implemented: Opaque	3.549	3.525		4.224	4.216
Implemented: Transparent	3.675	3.484		4.356	4.189
Average Payout (\$)	3.34	3.25		4.21	4.18
N (Control Individuals)	1376	1376		1391	1391
N (Treatment Person-Weeks, Opaque)	75	75		156	156
N (Treatment Person-Weeks, Trans.)	90	74		166	154

*Notes:* Panel A reports the decision rule associated with the challenge, and the costs associated with manipulating these behaviors. Panel B reports the performance of each decision rule by outcome, root mean squared error (RMSE) at the week-model level. Pooled metrics present the mean RMSE across models. Predicted Transparent represents the average expected performance of models given the theoretical model, behavior incentives, and estimated costs. Implemented Transparent/Opaque represents the average performance of models when assigned with/without transparency hints. Average payout represents the average payout to recipients based on model coefficients, given observed behavior. SR model estimated using preliminary costs estimates. Full results reported in appendix.

the decision rules were implemented experimentally. These will differ from the expected prediction error if people respond differently than anticipated by our model. Here, we find that the strategy-robust (SR) method performs better than LASSO when participants are given full information about the decision rule (by \$0.19 / 5% for income; \$0.17 / 4% pooled). The strategy-robust method also performs slightly better when the decision rule is opaque (by \$0.02 / 0.6% for income; \$0.01 / 0.2% pooled) — possibly because of increased shrinkage relative to standard LASSO. Table A2 shows detailed results for both the income and intelligence outcomes, and the Supplemental Appendix shows that the performance improvements are even larger when all outcomes are considered: under full information, SR outperforms LASSO by 12%; under opacity, SR outperforms LASSO by 1% (see SA Table 1).

Even if a policymaker intended to keep the decision rule opaque, using our robust method can reduce systematic risk in the chance that agents discover the decision rule. In practical implementations, policymakers could adaptively tweak the level of robustness to match the level of manipulation.

## 5 Discussion and Extensions

### 5.1 Contrast to standard estimators

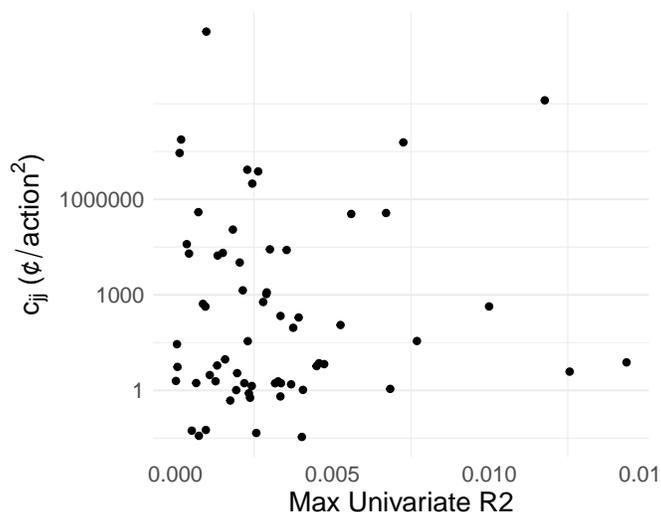
Standard supervised machine learning estimators evaluate each predictor based on its correlation with the outcome within a training dataset. However, we find that features that appear equally predictive in a training dataset have wildly different manipulation costs, and thus will be differentially effective if used in a decision rule. We illustrate this in Figure 3, which compares the cost of manipulation to the baseline predictive power of several dozen features from our experiment.

We next compare our method to two common approaches to manipulation, simulating performance using our experimentally estimated model of behavior.

#### Contrast with the ‘intuitive’ approach

An approach that is intuitive to economists would be to train a standard estimator but simply omit behaviors that are most manipulable (e.g., by only considering features

Figure 3: Manipulation Costs vs. Baseline Predictive Power



Each dot is a feature. The x-axis indicates the highest  $R^2$  across income and intelligence; the y-axis indicates the estimated manipulation cost.

above some y-axis threshold on Figure 3). We assess this approach in the Supplemental Appendix (section 4.3). This intuitive approach reduces the predicted manipulability of models, but as suggested by Figure 3 also removes from consideration useful predictors, in some cases by so much that it decreases the predicted performance. When the model is allowed to select from only the least manipulable indicators, in some cases LASSO is left with no behaviors that are predictive enough to include in the regression. In contrast, our approach can extract signal even from manipulable behaviors.

### Contrast with the ‘industry’ approach

A second approach involves iteratively re-training a naïve machine learning estimator after people have responded to the previous decision rule. With both income and intelligence, we observe that the performance of this method approaches the strategy-robust method after approximately 4 iterations of consumers being made perfectly aware of a new rule, adapting behavior, and then the policymaker retraining the algorithm (see Supplemental Appendix, section 4.3). However, the performance of this iterative approach then begins to deteriorate. When predicting income this deterioration is small, but for intelligence, performance eventually falls below the performance obtained before any retraining. This is foreshadowed by the difference

in moment conditions between the methods (Equation (2)); even when trained on data from a strategy-robust equilibrium, standard methods may leave an equilibrium because they do not anticipate that agents will respond.

## 5.2 Performance cost of transparency

While society increasingly demands transparency in machine decisions, transparency can facilitate manipulation, which may reduce the quality of those decisions. Our setting allows us to estimate this performance cost of transparency by comparing the performance of the optimal opaque rule (under the assumption that opacity will prevent it from being manipulated) to the optimal strategy-robust transparent rule (factoring in equilibrium manipulation). Because the opaque rule also faces the threat of manipulation, this difference represents an upper bound of the true performance cost. Crucially, under the assumptions of our model, this quantity can be estimated without revealing the decision rule: it only requires the estimation of types and costs (the first part of our experiment).<sup>29</sup>

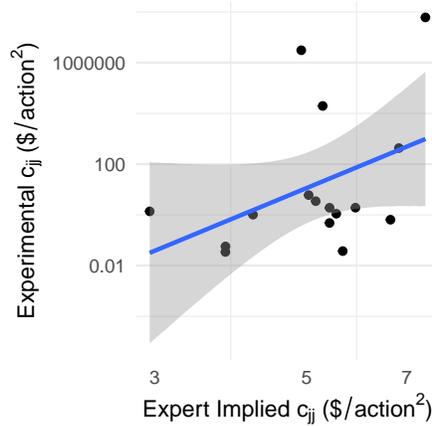
We estimate this cost of transparency in two ways: with our model and with our experiment, shown in the final rows of Panel B of Table 6. Our model predicts that transparency will reduce the performance of naive models by  $\$4.328 - \$4.273 = \$0.055$  (1.2%) on average across income and intelligence, but that strategy-robust models will perform similarly regardless if transparent. These predictions are similar to the actual change in performance due to transparency that we find in our experiment:  $\$4.356 - \$4.224 = \$1.32$  (3%) for naive models, and indeed negligible for our strategy-robust models.<sup>30</sup> These two outcomes had a lower cost of transparency than other outcomes; when we pool all outcomes together we find that transparency reduced performance of naive models by 17% and strategy-robust models by only 6%.

---

<sup>29</sup>Our method of estimating costs does require revealing the existence of features to users, but does not require specifying whether those features are included in the model, or with what weights.

<sup>30</sup>Our results suggest that the cost of transparency is actually negative when the decision rule targets high-income individuals, which is theoretically possible.

Figure 4: Costs Elicited from Experts vs. Costs Measured in Experiment



*Notes:* Each dot represents a behavior captured by the Sensing App. Y-axis indicates the cost of manipulating that behavior, estimated through our experiment (Table 4). X-axis indicates costs elicited from expert surveys, inferred as  $\hat{c}_{jj} = \frac{1}{N_{survey}} \sum_i \frac{\beta_j}{\max(0.001, \Delta_{jji})}$  for each  $i$  surveyed.

### 5.3 Alternate methods to estimate manipulation costs

The strategy-robust estimator requires beliefs about the costs of manipulating different behaviors. This paper demonstrates an experimental approach to eliciting those costs, but alternative approaches may be better suited to other settings:

*Expert elicitations.* We evaluate how well experts can predict the costs of manipulating different behaviors, in the spirit of DellaVigna and Pope (2016). We surveyed experts with different backgrounds (PhDs from different fields, research assistants, Busara staff who had not worked on the experiment, and Mechanical Turk workers in the US) to predict how Kenyans would manipulate different phone behaviors when incentivized. We then infer the structural cost parameters implied by the predictions of the 171 respondents. Results are shown in Figure 4. Although experts generally predict that costs are too low, the correlation is 0.30. If we use expert predictions of manipulation costs to train our model, and then assess predicted performance with the experimentally estimated model, even these noisy estimates improve performance substantially for one outcome, and have an inconsequential negative effect on the other, as shown in Table A3. This suggests that expert elicitations show promise as a low-cost way to estimate manipulation costs. See Supplemental Appendix section 3.

*First principles/structural approach.* In some cases, it may be possible to build up the cost of underlying manipulations from market prices and first principles.<sup>31</sup> A structural model of costs would allow an implementer to account for changes in these underlying parameters, suggesting how manipulation will change if, for example, the phone company changed the price of calls, or a service emerged that made it easy to generate incoming calls.

## 5.4 Nonlinear decision rules

This paper focuses on linear decision rules to sharpen intuition, but the core insight is also relevant in nonlinear settings. If outcomes are binary or discrete, agents near the classification threshold have higher incentives to manipulate behavior. Agents must have beliefs about how close they are to the threshold. More generally, many modern machine-learned decision rules are complex and nonlinear. In such settings, if agents' beliefs about those rules can be approximated by linear functions, our approach could be viewed as a linear approximation of those beliefs, as well as the actual functions.<sup>32</sup>

## 5.5 Social costs of manipulation

Our main specifications consider a narrow-minded policymaker who considers only predictive accuracy ( $M(\cdot) \equiv 0$ ). A socially-minded policymaker may also weigh the costs that agents incur manipulating behavior. Appendix Table A4 shows that as the loss function places more weight on the welfare costs that agents incur manipulating, our estimator adjusts models, typically towards even less manipulable behaviors.

---

<sup>31</sup>For example, the dark net price index (Gomez, 2020) reports the going price for online manipulations from an investigation on web forums: the average rate for 1,000 Instagram likes is \$6; 1,000 Twitter retweets go for \$25, suggesting they are more costly to manipulate. One can also cost out manipulation strategies: one can increase the number of noncontacts spoken with by randomly dialing 10 digit numbers and hanging up after the recipient picks up. That costs the call price of \$0.04/minute plus the value of the time to dial a 10 digit number, divided by the fraction of such numbers that are valid and pick up, which can be valued at the going wage.

<sup>32</sup>The benefits of extreme nonlinearities in modern machine learning may be lessened when manipulation is taken into account; linear decision rules can be more robust (Holmstrom and Milgrom, 1987; Carroll, 2015). Nonlinear environments may also have many more equilibria. In such settings, if iterative learning converges, it may converge to an undesirable equilibrium, whereas an approach like ours could be used to select a global optimum.

## 5.6 Nondeterministic or imperfectly known decision rules

Our model considers the case where agents know the decision rule perfectly. In practice, agents are likely to have noisy beliefs: even if a policymaker keeps a decision rule secret, agents may still be able to guess some of its properties. And even if a rule is revealed, it may be difficult to interpret (Freitas, 2014). How individuals respond will also ultimately depend on how beliefs are transmitted, including the ability for middlemen to capitalize on exploits and defraud at scale. Such considerations suggest extensions to our approach that incorporate a model of belief formation.

Another option for addressing manipulation is to make the decision rule less predictable. Although making decision rules nondeterministic may make them harder to manipulate, it undermines a major goal of transparency: that people know how they are evaluated. It may be appropriate in some settings (as with the drunk driving checkpoints described in Banerjee et al. (2019)).

## 5.7 Alternate forms of costs

For simplicity, we have modeled the cost of manipulation as having a quadratic form, which implies that behavior shifts linearly with incentives. In general, manipulation costs may include fixed components (e.g., the cost of setting up a spoofing app), asymmetries (e.g., the cost of installing an app differs from that of deleting it), and dynamic elements (such as seasonality or changes in the price of calls). There may also be costs associated with learning the decision rule.<sup>33</sup> In the Supplemental Appendix (section 2.1), we analyze how behavior responds to random variation in financial incentives. We find that linearity is a reasonable first approximation, though there is some evidence of diminishing returns, and less response for negative incentives.

## 5.8 Greenfield vs. brownfield implementations

Like our study, new applications of machine learning are typically trained in *greenfield* settings, using baseline data that was not incentivized, and not manipulated. Models

---

<sup>33</sup>As suggested by Ball (2019), there may also be particular features that have more heterogeneity in cost between individuals. We treat these two dimensions of heterogeneity as independent; if our approach were extended to allow for this interdependence, it would downweight indicators that have a particular spread in manipulability.

trained in new settings can be acutely susceptible to manipulation, as baseline data does not expose evidence of manipulation. In greenfield settings, during training it is possible to infer individual types directly from baseline data (Equation (4)). Our method can also be applied in *brownfield* settings, where an implementation has already been implemented and baseline behavior is already manipulated, by inverting observed behavior under current incentives using joint moment conditions.

## 6 Conclusion

This paper considers the possibility that machine decisions change the world in which they are deployed. We focus on the case where individuals manipulate their behavior in order to game decision rules. We derive decision rules that anticipate this manipulation, by embedding a behavioral model of how individuals will respond. This structural approach makes it possible to decompose decision rules into constituent components, and to gather data on how those components can be manipulated. From these components, our structural model allows us to understand how *any* proposed decision rule of a given form would be manipulated. This allows us to compute decision rules that are optimal in equilibrium.

We demonstrate our method in a field experiment in Kenya, by deploying a tailor-made smartphone app that mimics the ‘digital credit’ loan products that are now commonplace in sub-Saharan Africa. We find that even some of the world’s poorest users of technology – who are relatively recent adopters of smartphones and to whom whom the concept of an ‘algorithm’ is quite foreign (Musya and Kamau, 2018) – are savvy enough to change their behavior to game machine decisions. In this setting, we show that our strategy-robust estimator outperforms standard estimators on average by 12% when individuals are given information about the scoring rule. This framework also allows us to quantify the “cost of transparency”, i.e., the loss in predictive performance associated with moving from “security through obscurity” (with a naïve decision rule) to a regime of full algorithmic transparency (with our strategy-robust rule). We estimate this loss to be roughly 6% in equilibrium—substantially less than the 17% loss associated with making the naïve rule transparent.

Our discussion focuses on the simple case of linear models with a small number

of predictor variables, where subjects have either no information or full information about the decision rule. We envision useful extensions to more complex models and more nuanced beliefs. More generally, our approach of embedding a model of behavior within a machine learning estimator may be relevant to a wide range of contexts where machine learning systems face a changing human environment. In this sense, it offers a machine learning interpretation of [Lucas \(1976\)](#), where algorithmic decisions change the context of the systems they model. In our setting,  $\beta$  determines not just predictive performance within a given world, but also which counterfactual world occurs.

## References

- Agarwal, Nikhil and Eric Budish**, “Market Design,” *Handbook of Industrial Organization*, 2021.
- Aiken, Emily, Suzanne Bellue, Dean Karlan, Christopher Udry, and Joshua E Blumenstock**, “Machine Learning and Mobile Phone Data Can Improve the Targeting of Humanitarian Assistance,” *Working Paper*, July 2021.
- Akerlof, George A.**, “The economics of ”tagging” as applied to the optimal income tax, welfare programs, and manpower planning,” *The American economic review*, 1978, *68* (1), 8–19.
- Alatas, Vivi, Abhijit Banerjee, Rema Hanna, Benjamin A. Olken, Ririn Purnamasari, and Matthew Wai-Poi**, “Self-Targeting: Evidence from a Field Experiment in Indonesia,” *Journal of Political Economy*, March 2016, *124* (2), 371–427.
- Ball, Ian**, “Scoring Strategic Agents,” *arXiv:1909.01888 [econ]*, November 2019. arXiv: 1909.01888.
- Banerjee, Abhijit, Esther Duflo, Daniel Keniston, and Nina Singh**, “The Efficient Deployment of Police Resources: Theory and New Evidence from a Randomized Drunk Driving Crackdown in India,” Working Paper 26224, National Bureau of Economic Research September 2019. Series: Working Paper Series.
- , **Rema Hanna, Benjamin A Olken, and Sudarno Sumarto**, “The (lack of) Distortionary Effects of Proxy-Means Tests: Results from a Nationwide Experiment in Indonesia,” Working Paper 25362, National Bureau of Economic Research December 2018.
- Barocas, Solon, Moritz Hardt, and Arvind Narayanan**, *Fairness and Machine Learning*, fairmlbook.org, 2018.

- Bharadwaj, Prashant, William Jack, and Tavneet Suri**, “Fintech and Household Resilience to Shocks: Evidence from Digital Loans in Kenya,” Working Paper 25604, National Bureau of Economic Research February 2019.
- Björkegren, Daniel**, “‘Big data’ for development,” 2010.
- **and Darrell Grissen**, “Behavior Revealed in Mobile Phone Usage Predicts Credit Repayment,” *The World Bank Economic Review*, 2019.
- Bloomberg**, “Phone Stats Unlock a Million Loans a Month for Africa Lender,” *Bloomberg.com*, September 2015.
- Blumenstock, Joshua E.**, “Estimating Economic Characteristics with Phone Data,” *AEA Papers and Proceedings*, 2018, 108, 72–76.
- Blumenstock, Joshua Evan, Dan Gillick, and Nathan Eagle**, “Who’s Calling? Demographics of Mobile Phone Use in Rwanda,” in “2010 AAAI Spring Symposium Series” March 2010.
- **, Gabriel Cadamuro, and Robert On**, “Predicting poverty and wealth from mobile phone metadata,” *Science*, November 2015, 350 (6264), 1073–1076.
- Borrell Associates**, “Trends in Digital Marketing Services,” 2016.
- Brailovskaya, Valentina, Pascaline Dupas, Jonathan Robinson, and Jonathan Robinson**, “Digital Credit: Filling a hole, or digging a hole? Evidence from Malawi,” Working Paper May 2021.
- Bruckner, Michael and Tobias Scheffer**, “Stackelberg Games for Adversarial Prediction Problems,” in “Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining” KDD ’11 ACM New York, NY, USA 2011, pp. 547–555.
- Camacho, Adriana and Emily Conover**, “Manipulation of Social Program Eligibility,” *American Economic Journal: Economic Policy*, May 2011, 3 (2), 41–65.
- Carroll, Gabriel**, “Robustness and Linear Contracts,” *American Economic Review*, February 2015, 105 (2), 536–563.
- CGAP**, “Kenya’s Digital Credit Revolution Five Years On,” *CGAP*, March 2018.
- Dee, Thomas S., Will Dobbie, Brian A. Jacob, and Jonah Rockoff**, “The Causes and Consequences of Test Score Manipulation: Evidence from the New York Regents Examinations,” *American Economic Journal: Applied Economics*, July 2019, 11 (3), 382–423.

- DellaVigna, Stefano and Devin Pope**, “Predicting Experimental Results: Who Knows What?,” Working Paper 22566, National Bureau of Economic Research August 2016.
- Dong, Jinshuo, Aaron Roth, Zachary Schutzman, Bo Waggoner, and Zhiwei Steven Wu**, “Strategic Classification from Revealed Preferences,” in “Proceedings of the 2018 ACM Conference on Economics and Computation” EC ’18 ACM New York, NY, USA 2018, pp. 55–70.
- Dranove, David, Daniel Kessler, Mark McClellan, and Mark Satterthwaite**, “Is More Information Better? The Effects of “Report Cards” on Health Care Providers,” *Journal of Political Economy*, June 2003, 111 (3), 555–588.
- Eliaz, Kfir and Ran Spiegler**, “The Model Selection Curse,” *American Economic Review: Insights*, September 2019, 1 (2), 127–140.
- European Union**, “EU General Data Protection Regulation (GDPR),” 2016.
- Francis, Eilin, Joshua Blumenstock, and Jonathan Robinson**, “Digital Credit: A Snapshot of the Current Landscape and Open Research Questions,” *CEGA White Paper*, 2017.
- Frankel, Alex and Navin Kartik**, “Muddled Information,” *Journal of Political Economy*, August 2019, 127 (4), 1739–1776.
- and – , “Improving Information from Manipulable Data,” *arXiv:1908.10330 [econ]*, April 2020. arXiv: 1908.10330.
- Freitas, Alex A.**, “Comprehensible Classification Models: A Position Paper,” *SIGKDD Explor. Newsl.*, March 2014, 15 (1), 1–10.
- FSD Kenya**, “Tech-enabled lending in Africa,” 2018.
- Gomez, Miguel**, “Dark Web Price Index,” 2020. Section: SECURITY.
- Gonzalez-Lira, Andres and Ahmed Mobarak**, “Slippery Fish: Enforcing Regulation under Subversive Adaptation,” IZA Discussion Paper 12179, Institute of Labor Economics (IZA) February 2019.
- Goodhart, Charles**, *Monetary Relationships: A View from Threadneedle Street*, University of Warwick, 1975. Google-Books-ID: GKwJMwEACAAJ.
- Goodman, Bryce and Seth Flaxman**, “European Union regulations on algorithmic decision-making and a ”right to explanation”,” *arXiv:1606.08813 [cs, stat]*, June 2016. arXiv: 1606.08813.

- Greenstone, Michael, Guojun He, Ruixue Jia, and Tong Liu**, “Can Technology Solve the Principal-Agent Problem? Evidence from Pollution Monitoring in China,” 2019.
- Hanna, Rema and Benjamin A. Olken**, “Universal Basic Incomes versus Targeted Transfers: Anti-Poverty Programs in Developing Countries,” *Journal of Economic Perspectives*, November 2018, 32 (4), 201–226.
- Hardt, Moritz, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters**, “Strategic Classification,” in “Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science” ITCS ’16 ACM New York, NY, USA 2016, pp. 111–122.
- Holmstrom, Bengt and Paul Milgrom**, “Aggregation and Linearity in the Provision of Intertemporal Incentives,” *Econometrica*, 1987, 55 (2), 303–328.
- Hu, Lily, Nicole Immorlica, and Jennifer Wortman Vaughan**, “The Disparate Effects of Strategic Manipulation,” *Proceedings of the Conference on Fairness, Accountability, and Transparency - FAT\* ’19*, 2019, pp. 259–268. arXiv: 1808.08646.
- Kleinberg, Jon and Manish Raghavan**, “How Do Classifiers Induce Agents to Invest Effort Strategically?,” in “Proceedings of the 2019 ACM Conference on Economics and Computation” EC ’19 ACM New York, NY, USA 2019, pp. 825–844. event-place: Phoenix, AZ, USA.
- Lucas, Robert E.**, “Econometric policy evaluation: A critique,” *Carnegie-Rochester Conference Series on Public Policy*, January 1976, 1 (Supplement C), 19–46.
- McCaffrey, Mike, Olivia Obiero, and George Mugweru**, “M-Shwari: Market Reactions and Potential Improvements,” Technical Report 139 2013.
- Milli, Smitha, John Miller, Anca D. Dragan, and Moritz Hardt**, “The Social Cost of Strategic Classification,” in “Proceedings of the Conference on Fairness, Accountability, and Transparency” FAT\* ’19 ACM New York, NY, USA 2019, pp. 230–239. event-place: Atlanta, GA, USA.
- Mirrlees, J. A.**, “An Exploration in the Theory of Optimum Income Taxation,” *The Review of Economic Studies*, 1971, 38 (2), 175–208.
- Musya, Mercy and Grace Kamau**, “How do you say “algorithm” in Kiswahili?,” December 2018. Library Catalog: medium.com.
- National Institute of Standards and Technology**, “Guide to General Server Security,” *NIST Special Publication*, July 2008, (800-123).
- Nichols, Albert L. and Richard J. Zeckhauser**, “Targeting Transfers through Restrictions on Recipients,” *The American Economic Review*, 1982, 72 (2), 372–377.

- Niehaus, Paul, Antonia Atanassova, Marianne Bertrand, and Sendhil Mullainathan**, “Targeting with Agents,” *American Economic Journal: Economic Policy*, 2013, 5 (1), 206–238.
- Perdomo, Juan C., Tijana Zrnic, Celestine Mender-Dünner, and Moritz Hardt**, “Performative Prediction,” *arXiv:2002.06673 [cs, stat]*, June 2020. arXiv: 2002.06673.
- Ramsey, F. P.**, “A Contribution to the Theory of Taxation,” *The Economic Journal*, 1927, 37 (145), 47–61.
- Sayed-Mouchaweh, Moamar and Edwin Lughofer**, *Learning in Non-Stationary Environments: Methods and Applications*, Springer Science & Business Media, April 2012. Google-Books-ID: qFWM2nva7xQC.
- Spence, Michael**, “Job Market Signaling,” *The Quarterly Journal of Economics*, 1973, 87 (3), 355–374.
- Sundsøy, Pål, Johannes Bjelland, Bjørn-Atle Reme, Eaman Jahani, Erik Wetter, and Linus Bengtsson**, “Estimating individual employment status using mobile phone network data,” *arXiv:1612.03870 [cs]*, December 2016. arXiv: 1612.03870.

## Appendices

### A1 Estimation Details

#### Moment Conditions

The following moment conditions jointly identify  $C$  and  $\omega$ .

Incentives are orthogonal to idiosyncratic behavior shocks ( $\mathbb{E}[\beta_{itk}\epsilon_{itj}] = 0$ ). For each pair of behaviors  $jk$  (including  $j = k$ ) this yields sample moment condition:

$$\frac{1}{N} \sum_{i=1}^N \sum_{t \in \mathbb{T}_i} \beta_{itk} \left[ x_{ijt} - \mathbf{x}_{ij} - \mu_{jt} - e^{-\omega' \mathbf{z}_i} \cdot [C^{-1} \beta_{it}]_j \right] = 0$$

where  $[\mathbf{a}]_k$  indicates the  $k$ th element of  $\mathbf{a}$ .

Implied unobserved heterogeneity  $\tilde{v}_i$  is given by:

$$\tilde{v}_i = \frac{1}{\sum_{t \in \mathbb{T}_i^{treatment}} |K_{it}^{eval}|} \sum_{t \in \mathbb{T}_i^{treatment}} \sum_{k \in K_{it}^{eval}} \left[ \frac{x_{ikt} - \mathbf{x}_{ik} - \mu_{kt}}{[C^{-1}\boldsymbol{\beta}_{it}]_k} - e^{-\boldsymbol{\omega}'\mathbf{z}_i} \right] \quad (5)$$

where  $K_{it}^{eval}$  is the set of behaviors to be evaluated for  $i$  in period  $t$ .<sup>34</sup> Unobserved heterogeneity is mean zero, yielding moment condition,  $\frac{1}{N} \sum_i \tilde{v}_i = 0$ , and orthogonal to each heterogeneity characteristic  $z_l$ , yielding moment condition(s)  $\frac{1}{N} \sum_i z_{li} \cdot \tilde{v}_i = 0$ .

### Manipulation Cost Regularization

We add to our GMM loss function the regularization term:

$$R_{costs}^{\lambda_{costs}}(\cdot) = \left[ \lambda_{diagonal}^{costs} \sum_k \theta_{kk}^2 + \lambda_{offdiagonal}^{costs} \sum_{j \neq k} \theta_{jk}^2 \right] \left[ \frac{1}{N} \sum_i e^{-2\boldsymbol{\omega}'\mathbf{z}_i} \right]$$

where  $\theta_{jk}$  represents the elements of inverse costs  $C^{-1}$ .

### Unobserved Gaming Ability

We recover the distribution of unobserved gaming ability  $V$  in two steps. We compute gaming ability residuals  $\tilde{v}_i$  as in Equation (5), which capture whether each individual manipulates more or less than predicted during incentivized periods. Then, to reduce the impact of noise and outliers, we shrink and winsorize these inferred shocks. We form the empirical distribution  $V = \{\max(\phi \cdot \tilde{v}_i, \underline{v})\}_i$ , where  $\underline{v}$  is the lowest value of  $\tilde{v}$  that leads to a nonnegative implied gaming ability, and  $\phi$  is a shrinkage parameter calibrated to minimize overall error in observed incentivized periods (that is,  $\underline{v} = \min_i(\tilde{v}_i | \phi \cdot \tilde{v}_i \geq -\min_j(e^{-\boldsymbol{\omega}'\mathbf{z}_j}))$ ).

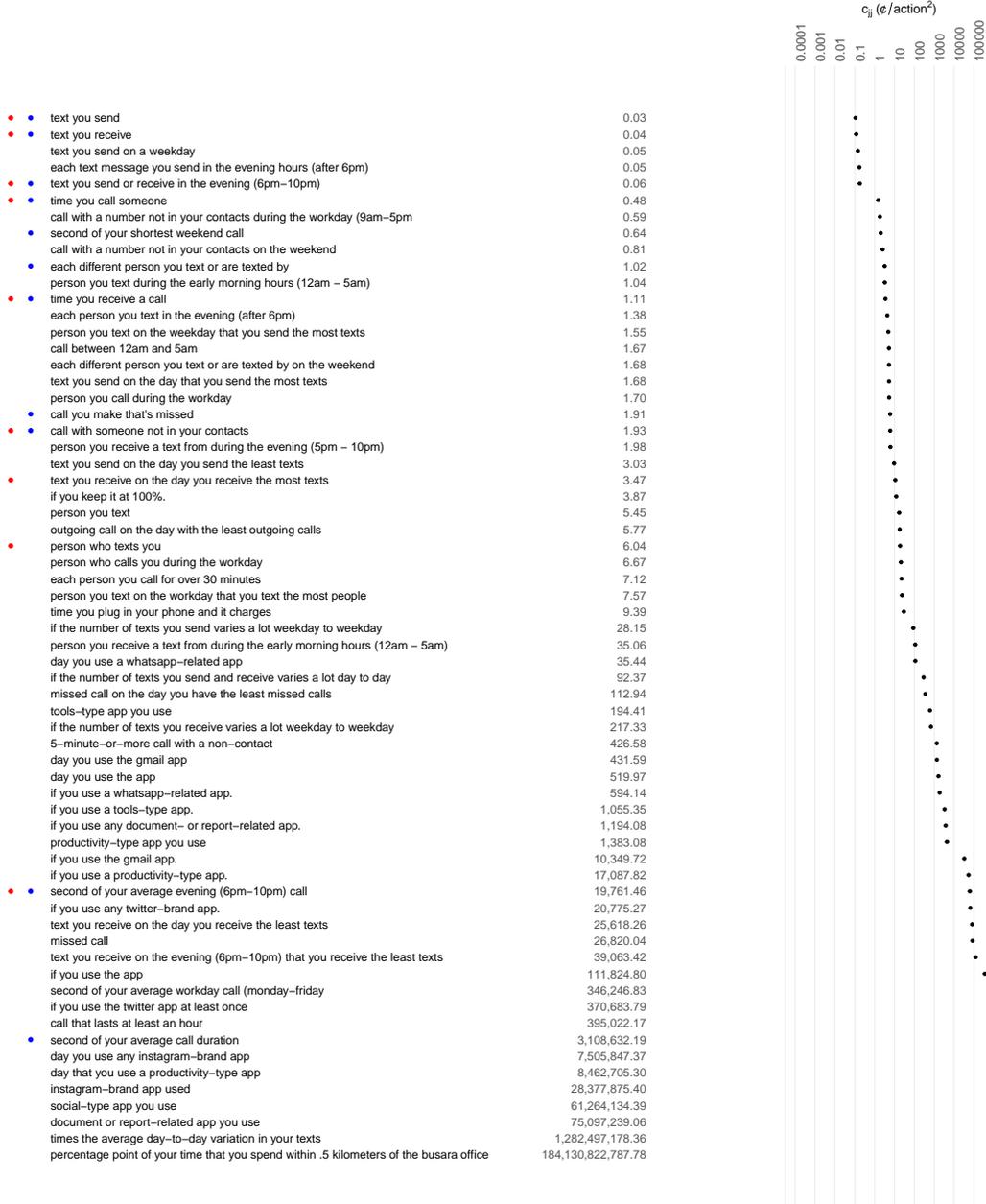
We calibrated  $\phi$  to 1e-6; For details, see Supplemental Appendix 2.2.2.

---

<sup>34</sup>We set  $K_{it}^{eval} = \{k \text{ s.t. } \beta_{itk} \neq 0\}$ , so that  $\tilde{v}_i$  is evaluated only off shifts in the incentivized behavior. One could alternately evaluate how each incentive shifts all behaviors.

Table A1: Estimated Manipulation Costs for All Behaviors

Heterogeneity by Behavior (C diagonal; all incentivized behaviors)



Parameters estimated using GMM. Red dot indicates used in a LASSO model; blue indicates used in SR model. In cost matrix, off diagonal elements  $c_{jk}; j \neq k$  regularized to zero ( $\lambda_{offdiagonal}^{costs} \rightarrow \infty$ ), diagonal elements regularized with  $\lambda_{diagonal}^{costs} = 1.0$ , set via 3-fold cross validation.

Table A2: Performance of Decision Rules

	<i>Costs</i>	<b>Income &amp; Intelligence</b> (Pooled)		<b>Income</b>		<b>Intelligence</b> (Ravens above median)	
	$c_{jj}$ ¢/action <sup>2</sup>	$\beta^{LASSO}$	$\beta^{SR}$	$\beta^{LASSO}$	$\beta^{SR}$	$\beta^{LASSO}$	$\beta^{SR}$
				¢/action		¢/action	
<b>Panel A: Decision Rule</b>							
text_count_out	0.035	-	-	-0.395	-0.107		
text_count_incoming	0.037	-	-	0.065		0.278	0.145
text_count_evening	0.057	-	-		-0.121		
call_count_out	0.480	-	-	0.625	0.542		
call_count_outgoing_missed	1.91	-	-			-0.208	
calls_noncontacts	1.929	-	-			-0.606	-0.575
max_daily_texts_incoming	3.471	-	-				0.324
intercept	.	-	-	301.071	304.622	490.727	488.441
<b>Panel B: Prediction Error</b>							
		RMSE (\$)		RMSE (\$)		RMSE (\$)	
Baseline Data: Control		4.273	4.278	3.574	3.583	4.971	4.973
Baseline Data: Predicted Transparent		4.328	4.279	3.672	3.585	4.984	4.974
Implemented: Opaque		4.224	4.216	3.549	3.525	4.898	4.906
Implemented: Transparent		4.356	4.189	3.675	3.484	5.037	4.894
Average Payout (\$)		4.21	4.18	3.34	3.25	5.11	5.07
N (Control Individuals)		1391	1391	1376	1376	1391	1391
N (Treatment person-weeks, Opaque)		156	156	75	75	81	81
N (Treatment person-weeks, Transparent)		166	154	90	74	76	80

*Notes:* Panel A reports the decision rule associated with the challenge, and the costs associated with manipulating these behaviors. Panel B reports the performance of each decision rule by outcome, root mean squared error (RMSE) at the week-model level. Pooled metrics present the mean RMSE across models. Predicted Transparent represents the average expected performance of models given the theoretical model, behavior incentives, and estimated costs. Implemented Transparent/Opaque represents the average performance of models when assigned with/without transparency hints. SR model estimated using preliminary cost estimates.

Table A3: SR Models Based on Expert-Estimated Costs

	<i>Costs</i> (Actual)	<i>Costs</i> (From Experts)	$\beta^{LASSO_{final}}$	<b>Income</b> $\beta^{SR_{final}}_{ExpertCost}$	$\beta^{SR_{final}}$	<b>Intelligence</b> (above median Ravens)		
						$\beta^{LASSO_{final}}$	$\beta^{SR_{final}}_{ExpertCosts}$	$\beta^{SR_{final}}$
<i>Panel A: Decision Rule</i>								
text_count_out	0.035	3.804	-0.499	-0.329	-0.093			
text_count_incoming	0.037	5.645	0.141	0.014		0.270	0.223	0.114
text_count_evening	0.057	3.805			-0.115			
call_count_out	0.480	5.4	0.657	0.591	0.501		-0.058	
call_count_outgoing_missed	1.914	5.4				-0.156		
calls_noncontacts	1.929	5.891				-0.547		-0.518
max_daily_texts_incoming	3.471	5.155						0.421
Intercept			296.342	305.309	303.456	489.686	483.529	487.049
$\lambda^{decision}$			759.295	759.295	759.296	1032.37	1032.37	1032.37
<i>Panel B: Prediction Error</i>								
				RMSE (\$)			RMSE (\$)	
Predicted Opaque			3.572	3.577	3.586	4.972	4.982	4.973
Predicted Transparent			3.831	3.64	3.586	4.983	4.989	4.973

*Notes:* Panel A reports the decision rules derived from naive LASSO and our strategy-robust model, as well as strategy-robust models that use only control weeks and costs estimated from expert surveys. It also reports the costs associated with these behaviors. Panel B reports the predicted performance of these decision rules, using the experimentally estimated model.  $\beta^{LASSO_{final}}$  presented in this table differs slightly from the  $\beta^{LASSO}$  which was implemented. The regularization protocol was updated to select penalization closer to the boundary of 3 coefficients and the sample was changed to coincide with that used for the SR model (it includes only individuals with nonmissing tech skills, dropping approximately 1.5 percent of the sample). For expert survey costs, we infer heterogeneity in gaming ability using variation in participant responses (see Supplemental Appendix).

Table A4: Models Adjusted for Welfare Costs of Manipulation

	<i>Costs</i>	<b>Income</b>					<b>Intelligence</b> (above median Ravens)				
		$c_{jj}$	$\beta^{LASSO_{final}}$	$\beta_{w=0}^{SR_{final}}$	$\beta_{w=0.1}^{SR_{final}}$	$\beta_{w=0.5}^{SR_{final}}$	$\beta_{w=1}^{SR_{final}}$	$\beta^{LASSO_{final}}$	$\beta_{w=0}^{SR_{final}}$	$\beta_{w=0.1}^{SR_{final}}$	$\beta_{w=0.5}^{SR_{final}}$
<b>Panel A: Decision Rule</b>											
text_count_out	0.035	-0.499	-0.093	-0.092							
text_count_incoming	0.037	0.141					0.270	0.114	0.067	0.030	0.019
text_count_out_evening	0.054										
text_count_evening	0.057		-0.115	-0.115	-0.055	-0.037					0.023
call_count_out	0.480	0.657	0.501	0.494	0.278	0.179					
max_daily_texts_out	1.683				-0.294	-0.222					
call_count_outgoing_missed	1.914						-0.156				
calls_noncontacts	1.929						-0.547	-0.518	-0.422	-0.204	
max_daily_texts_in	3.471							0.421	0.541	0.518	0.387
call_count_over_1_minute	395022										
Intercept		296.342	303.456	303.669	312.514	314.717	489.686	487.049	489.071	488.921	489.317
$\lambda^{decision}$		759.296	759.296	759.296	759.296	759.296	1032.37	1032.37	1032.37	1032.37	1032.37
<b>Panel B: Prediction Error</b>											
			RMSE (\$)					RMSE (\$)			
Predicted Opaque		3.572	3.586	3.586	3.599	3.607	4.972	4.973	4.974	4.979	4.984
Predicted Transparent		3.831	3.586	3.586	3.598	3.607	4.983	4.973	4.974	4.979	4.984

*Notes:* Panel A reports the decision rules derived from naive LASSO and our strategy-robust model, with varying social welfare weight  $w$  placed on the costs agents incur manipulating. Panel B reports performance, measured as root mean squared error (RMSE).  $\beta^{LASSO_{final}}$  presented in this table differs slightly from the  $\beta^{LASSO}$  which was implemented. The regularization protocol was updated to select penalization closer to the boundary of 3 coefficients and the sample was changed to coincide with that used for the SR model (it includes only individuals with nonmissing tech skills, dropping approximately 1.5 percent of the sample). Manipulation costs included in policymaker's objective as  $M(\cdot) = w \cdot \mathbb{E}_i [c_i(\mathbf{x}_i^*(\beta), \underline{x}_i)] = w \cdot \mathbb{E}_{i,q} \left[ \frac{1}{2} \beta' C_{iq}^{-1} \beta \right]$ , for a weight  $w$  on consumer welfare.