Measuring the Completeness of Economic Models*

Drew Fudenberg† Jon Kleinberg‡ Annie Liang§ Sendhil Mullainathan¶

January 17, 2021

Abstract

Economic models are evaluated by testing the correctness of their predictions. We suggest an additional measure, "completeness": the fraction of the predictable variation in the data that the model captures. We calculate the completeness of prominent models in three problems from experimental economics: assigning certainty equivalents to lotteries, predicting initial play in games, and predicting human generation of random sequences. The completeness measure reveals new insights about these models, including how much room there is for improving their predictions.

*A two-page abstract of an early version of this project appeared in ACM:EC as "The Theory is Predictive, but is it Complete?" We thank Alberto Abadie, Amy Finkelstein, Indira Puri, Marciano Siniscalchi, and Johan Ugander for helpful comments. We are also grateful to Adrian Bruhin, Helga Fehr-Duda, Thomas Epper, Kevin Leyton-Brown, and James Wright for sharing data with us, and National Science Foundation Grants SES 1643517 for financial support.

†MIT

‡Cornell University

§Northwestern University

¶University of Chicago

1 Introduction

There is more reason to look for ways to improve a model that predicts poorly than one that predicts well. But what constitutes "good" performance? Our view is that the answer depends on how well the outcome could possibly be predicted given the specified "features" (i.e. the explanatory variables). To fix ideas, suppose we have data on whether customers agreed to a particular loan offer. Loan offers differ on characteristics such as the interest rate or the term of the loan. One model (the "NPV model") of how these characteristics relate to demand might posit that customers view loans through the lens of expected cost of capital over the duration of the loan. The expected cost of capital is a specific function of the available features. We could test the predictions of this model by evaluating it on data, e.g. by seeing if demand increases when the effective interest rate drops. These tests allow us to reject wrong models, but they do not tell us how much better a different model could do.

To get at this, we propose comparing the model's accuracy to that of the best prediction of demand that could be made using the features we have that describe each loan. Comparing the benchmark's predictive accuracy to that of the NPV model would tell us how much of the predictable signal in the outcome (given the baseline features) is captured by the NPV model. If the best predictions are much better than those of NPV model, there may be another model built on the same features that substantially improves predictive accuracy. For example, another model might postulate that customers ignore future interest rates and focus only on the initial interest rate, or that 2.99% is viewed differently from 2.95%. On the other hand, if the best predictions are not much better than those of the NPV model, then alternative models built on the same features cannot possibly do much better on these data. New models could help but they must do so by identifying new variables that are not currently measured. For example, models that emphasize framing and persuasion would point to expanding our dataset to include the vocabulary used in the loan descriptions.

Moving beyond this specific example, any model's prediction error can generally be decomposed into two sources: (1) intrinsic noise in the outcome due to limitations of the features we have measured, i.e. the *irreducible error*, and (2) regularities in the

data that the model does not capture. The irreducible error provides an upper bound benchmark for how well any model (based on the measured features) could possibly do.

A benchmark at the other end is the performance of a naive model, such as "guess at the outcome at random".[1] We use these extremes to measure what we call the "completeness" of any behavioral model:

$$\frac{\mathcal{E}_{\text{naive}} - \mathcal{E}_{\text{model}}}{\mathcal{E}_{\text{naive}} - \mathcal{E}_{\text{irreducible}}}$$

where $\mathcal{E}_{\text{naive}}$ is the out-of-sample prediction error under a naive baseline (e.g. "guess at random"), $\mathcal{E}_{\text{model}}$ is the out-of-sample error of the model, and $\mathcal{E}_{\text{irreducible}}$ is the irreducible error. That is, completeness is the model's reduction in prediction error (relative to the naive baseline), divided by the achievable reduction in prediction error. A model with a completeness of 0 does not improve upon the naive baseline, while a model with a completeness of 1 eliminates all but the irreducible error. Crucially, a model can be complete for the given measured features even if the it predicts poorly and its R^2 is low. The distinction between a complete model and a perfectly predictive model as especially relevant for the social sciences, where we expect there to be substantial irreducible noise in most outcomes of interest given the measurable features. Economists can rarely hope for our models to be perfectly predictive, but we can hope for them to be relatively complete.

In addition to proposing the completeness measure, we demonstrate that completeness can be accurately estimated for a diverse range of experimental data sets. The challenge is estimating irreducible error. In general, the performance of black box machine learning methods can be used as a stand-in for irreducible error. When the data consists of a large number of outcome observations for each vector of features, then it is possible to obtain a fairly precise estimate of irreducible error using a simple "lookup table," which nonparametrically searches the space of possible models, and finds the model that maximizes out-of-sample predictive accuracy for the set of available features. Many lab data sets have this property: for example, a large number of observations of game play for each of a small set of games, or a large number of observations of certainty equivalents for each of a small set of lotteries.

[1] Diebold and Kilian (2001) proposes benchmarking the accuracy of time series forecasts relative to that of a bad forecast. This is in the spirit of our comparison against to a naive model.

Our applications use data sets like this to evaluate the completeness of prominent models from three experimental domains: Cumulative Prospect Theory (Tversky and Kahneman, 1992) for prediction of certainty equivalents, the Poisson Cognitive Hierarchy Model (Camerer et al., 2004) for prediction of initial play in games, and Rabin and Vayanos (2010) for prediction of human perception of randomness.

We find that Cumulative Prospect Theory is 92% complete for predicting certainty equivalents; that is, it almost achieves the best possible error given the explanatory variables that we use. Simply evaluating the absolute level of the model's error would not have revealed this, as the mean-squared error is seemingly high, 68.43. In contrast, the Rabin and Vayanos (2010) model is only 14% complete, suggesting that there is further room for improvements in predictive accuracy using other regularities in behavior. In the setting of initial play, we find that the predictive limits vary substantially across different kinds of games, and so the same level of prediction error can mean different things.

In all of these prediction problems, we find that the irreducible error is substantial, so simply looking at the absolute level of a model's predictive error would paint a misleading picture. This is starkest in our application of predicting human generation of random outcomes, where the Rabin and Vayanos (2010) model reduces prediction error (relative to the naive benchmark of guessing at random) by only 0.0008, but is nevertheless 14% complete. These, and the subsequent observations we make in Sections 5.1–5.3, are informative about the problem domains and how much room there is for improving the predictions of their leading models without obtaining new sorts of data.

Our completeness measure depends on a specified set of features and is evaluated on a given data set. If we change either the underlying feature set or the data, we would expect the measurement of completeness to change, as we discuss in Section 6.2. Moreover, a model's completeness depends on which experiments are run, e.g. which lotteries or games are used in testing. As we show in Sections 5.2, the way that the completeness of a model varies across test cases is of independent interest, as it can shed light on the domains in which the model performs well or performs poorly.

1.1 Related Work

Irreducible error is an old concept in statistics and machine learning. A large literature has studied the decomposition of this error into *bias* (reflecting error due to the specification of the model class) and *variance* (reflecting sensitivity of the estimated rule to the randomness in the training data). Depending on the quantity of data available to the analyst, it may be preferable to trade off bias for variance or vice versa.[2] This paper abstracts from these concerns, as well as the related concern of overfitting. We work exclusively with data sets where there is enough data that the best feasible out-of-sample prediction accuracy is well approximated by searching across the unrestricted space of mappings from $\mathcal{X}$ into $\mathcal{Y}$ (see Appendix A).

The only previous measures of predictive success for economic models in experimental work that we know of are Selten (1991)'s measure of the relative frequency of successful predictions, Erev et al. (2007)'s definition of the *equivalent number of observations*, and Apesteguia and Ballester (2020)'s measure of goodness-of-fit for stochastic choice models. Our work differs in that we focus on understanding the best possible prediction in a given problem, and evaluate performance relative to that benchmark.

Several recent papers compare a model's predictive performance to that of specific machine learning algorithms. These algorithms sometimes approximate the best possible predictions. For example, Peysakhovich and Naecker (2017a) compares the performance of economic models for the willingness to pay for three-outcome lotteries to the performance of regularized regression algorithms, and Bodoh-Creed et al. (2019) compares the performance of simple OLS models using known regressors against the performance of random forests built on a rich feature set, for the problem of predicting pricing variation. The algorithms used in these papers need not achieve the irreducible error, but they do provide a lower bound for the best achievable accuracy. We show that in experimental contexts, it can be possible to directly estimate the best achievable accuracy and use that as a benchmark.

Other papers directly use an algorithmic approach to predict economic behavior, e.g. Plonsky et al. (2017), Noti et al. (2016), and Plonsky et al. (2019) for prediction of choice, and Camerer et al. (2019) for prediction of disagreements in bargaining. The

[2]For example, given small quantities of data, we may prefer to work with models that have fewer free parameters, leading to higher bias but potentially lower variance.

improvements achieved by these more complex algorithms over the existing economic models are sometimes modest. One reason for this might be intrinsic noise, as Bourgin et al. (2019) points out. We show how this noise can be quantified.

Finally, we note that in the special case where performance is measured by mean-squared error, and the naive rule is an unconditional mean, then our completeness measure can be seen as a ratio of the model's R^2 and the nonparametric R^2, as we explain in Appendix B. Our approach is not special to this loss function, however, and can be implemented with any metric of accuracy.

2 Example

We begin with a simple example that illustrates the need for a measure such as completeness. Let y be a binary outcome of interest, which is related to two binary features x_1 and x_2. Specific theories make predictions about how the given features relate to the outcome. Suppose that our model posits that the features enter linearly according to $y = \beta \cdot (x_1 + x_2)$ for some $\beta \in \mathbb{R}$. We can test this model by acquiring observations of (x_1, x_2, y) drawn from their true joint distribution, estimating β, and using the estimated model to predict outcomes in a new data set. The (out-of-sample) prediction error would then be the error of the model's predictions according to i.e. the (average) squared difference between the predicted and true outcomes. But it is hard to interpret the magnitude of this error without additional information. To see the problem, consider Table 1, which describes two data-generating processes for the value of y given x_1 and x_2.

x_1	x_2	$\mathbb{E}(y\|x)$
0	0	0
0	1	0.5
1	0	0.5
1	1	1

x_1	x_2	$\mathbb{E}(y\|x)$
0	0	0
0	1	0.1
1	0	0.9
1	1	1

Table 1: Two processes specifying the expected value of y given the values of x_1 and x_2. In both cases, the distribution over features is uniform.

For both data-generating processes, the estimated value of the parameter β (given

sufficient data) is $\beta = 0.5$, and so the estimated model is $f(x) = 0.5 \cdot (x_1 + x_2)$. The expected prediction error of this model is 0.125 under both of the data generating processes in Table 1. Taking the absolute level of prediction error at face value would suggest that the model is equally predictive for both versions of ground truth. But the equality of the prediction errors obscures an important difference, which is that in the first case there is no alternative model built on x_1 and x_2 that can make more accurate predictions, while in the second case the model $y = \beta_1 x_1 + \beta_2 x_2$ (with β_1 estimated to 0.1 and β_2 estimated to 0.9) achieves a prediction error of 0.045. That is, the proposed model is complete given the first data-generating process but incomplete given the second. Our subsequent approach formalizes this notion of completeness.

3 Completeness

Section 3.1 introduces the setting of prediction problems and Section 3.2 defines completeness.

3.1 Preliminaries

In a prediction problem, there is an *outcome* Y whose realization is of interest, and *features* X that are statistically related to the outcome. The goal is to predict the outcome given the observed features. Some examples include predicting an individual's future wage based on childhood covariates (city of birth, family income, quality of education, etc.), or predicting a criminal defendant's flight risk based on their past record and properties of the crime (Kleinberg et al., 2018). We focus on three prediction problems that emerge from experimental economics:

Example 1 (Risk Preferences). Can we predict the valuations that people will assign to various money lotteries?

Example 2 (Predicting Play in Games). Can we predict how people will play the first time they encounter a new simultaneous-move game?

Example 3 (Human Generation of Random Sequences). Given a target random process—for example, a Bernoulli random sequence—can we predict the errors that a human will make while mimicking this process?

Formally, suppose that the observable features belong to some space $\mathcal{X}$ and the outcome belongs to $\mathcal{Y}$. There is a true but unknown joint distribution P over $\mathcal{X} \times \mathcal{Y}$. A map $f : \mathcal{X} \to \mathcal{Y}$ from features to outcomes is a *(point) prediction rule*.[3] Many economic models can be described as a family of prediction rules $\mathcal{F}_\Theta$ indexed by an interpretable parameter set Θ. For example, the model class may impose a linear relationship $f(x) = \langle x, \theta \rangle$ between the outcome and a set of features x, in which case the parameter $\theta \in \Theta$ defines a vector of weights applied to each feature. In Section 5.1, one specification of $\mathcal{F}_\Theta$ is a family of utility functions $u(z) = z^\theta$ over dollar amounts, where the parameter θ reflects the degree of risk aversion.

3.2 Definition

We suppose that our prediction problem comes with a *loss function*, $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, where $\ell(y', y)$ is the error assigned to a prediction of y' when the realized outcome is y. The commonly used loss functions mean-squared error and classification error correspond to $\ell(y', y) = (y' - y)^2$ and $\ell(y', y) = \mathbb{1}(y' \neq y)$ respectively.[4]

Definition. *The **expected error** (or **risk**) of prediction rule f on a new observation $(x, y) \sim P$ is*

$$\mathcal{E}_P(f) = \mathbb{E}_P[\ell(f(x), y)]. \tag{1}$$

The prediction rule in the parametric class $\mathcal{F}_\Theta$ that minimizes the expected prediction error is

$$f^*_\Theta = \arg\min_{f \in \mathcal{F}_\Theta} \mathcal{E}_P(f).$$

The expected error of this "best" rule in $\mathcal{F}$ is $\mathcal{E}_P(f^*_\Theta)$. In Section 4, we discuss how to estimate $\mathcal{E}_P(f^*_\Theta)$ on finite data; here we discuss how to interpret it.

To understand a model's error, it is helpful to distinguish between two different sources of error. First, if the conditional distribution $Y \mid X$ is not degenerate, then

[3] Note that prediction of a probability distribution over $\mathcal{Y}$ can be cast as the prediction of a point in the space $Y' = \Delta(\mathcal{Y})$ of distributions on $\mathcal{Y}$; for many loss functions this prediction problem will have an irreducible error of zero.

[4] Different loss functions are typically used when predicting distributions, see e.g. Gneiting and Raftery (2007).

even the ideal prediction rule

$$f^*(x) = \arg\min_{y' \in \mathcal{Y}} \mathbb{E}_P \left[\ell(y', y) \mid x \right]$$

does not predict perfectly.

Definition. *The **irreducible error** in the prediction problem is the expected error*

$$\mathcal{E}_P(f^*) = \mathbb{E}_P \left[\ell(f^*(x), y) \right] \tag{2}$$

of the ideal rule on a new test observation.

The irreducible error is a lower bound on the error when predicting Y using the features in X.

A second source of prediction error is the specification of which prediction rules are in the class $\mathcal{F}_\Theta$. Typically the best possible model will not be an element of $\mathcal{F}_\Theta$, as most model classes are at least slightly misspecified. If $\mathcal{F}_\Theta$ leaves out an important regularity in the data, then there may be models outside of $\mathcal{F}_\Theta$ that yield much better predictions.[5]

These two sources of prediction error have very different implications for how to generate better predictions. If the model's prediction error is substantially higher than the irreducible error, it may be possible to identify new regularities and incorporate them into models that improve prediction given the same feature set. These new models might be preferable if they do not involve too great an increase in complexity or in the number of parameters. Conversely, if the model's prediction error is close to the irreducible error for the current feature set, the priority should be to identify additional features that will allow for better predictions.

We propose the ratio of the reduction in prediction error achieved by the model, compared to the achievable reduction, as a measure of how close the model comes to the best achievable performance. We call this ratio the model's *completeness*. To operationalize this measure, we select a naive rule $f_{\text{naive}} : \mathcal{X} \to \mathcal{Y}$ suited to the prediction problem, e.g. "predict uniformly at random." The performance of this naive rule is interpreted as a "worst case" prediction accuracy. We assume throughout that f_{naive} belongs to $\mathcal{F}_\Theta$, as is the case in our subsequent applications.

[5] On the other hand, expanding the model class risks overfitting, so more parsimonious model classes can lead to more accurate predictions when data is scarce (Hastie et al., 2009). As we discuss in Sections 1.1 and 4, all of the data sets we consider here are large relative to the number of features.

Definition. *The* **completeness of the parametric model class $\mathcal{F}_\Theta$ is**

$$\frac{\mathcal{E}_P(f_{\textbf{naive}}) - \mathcal{E}_P(f^*_\Theta)}{\mathcal{E}_P(f_{\textbf{naive}}) - \mathcal{E}_P(f^*)}. \tag{3}$$

Completeness is a normalized measure of the reduction in error. A model with completeness 0 does no better than the naive rule, while a "fully complete" model with completeness 1 removes all but the irreducible error.[6]

3.3 Discussion

User inputs. The proposed measure depends on two user inputs: a naive benchmark f_{naive} and a marginal distribution P_X on an underlying feature space X. In many cases, there is a natural choice for the naive rule, or a range of natural choices. In Appendix C.1, we expand one of our applications by estimating completeness relative to a set of possible naive benchmarks. We show that completeness is stable across these choices.

On the other hand, there generally isn't a corresponding notion for "standard" choices of P_X, although for experimental data this distribution too is typically chosen by the analyst—e.g. which games to ask laboratory participants to play or which lotteries to offer to subjects.[7] Ideally, we would like the chosen distribution over features to be the one that is most "economically relevant," but in practice, it may be difficult to define what this means. We show in Section 5.2 that completeness can vary substantially across different marginal distributions P_X, especially those with different support. This reflects the fact that some models perform better for certain kinds of inputs (e.g. certain kinds of lotteries or certain kinds of games). We view the ability of the completeness measure to capture this variation as a strength of the approach. In particular, one can study how completeness varies across subsets of the domain, as we do in the subsequent application in Section 5.2.

[6] This is one of many possible measures with completeness 1 when the model removes all but irreducible error and 0 when the model coincides with the naive error. Our definition measures "units" of completeness as percentage improvements in prediction error, which facilitates comparison across settings with different loss functions.

[7] For naturally occurring data, the distribution over the feature space X may be governed by an external process.

Expanding $\mathcal{X}$. Completeness is defined for a fixed feature set $\mathcal{X}$, which we generally interpret as the measured features in the data. If we vary $\mathcal{X}$ by expanding it to include new measured features, then the predictive performance of the original model remains the same, but the predictive optimum weakly improves.[8] So a model that is complete for one feature set $\mathcal{X}$ may not be complete for another $\mathcal{X}' \supset \mathcal{X}$. In general, if a model is nearly complete for the measured features, the only way to improve predictive accuracy is to measure new features and develop new models on the larger feature set.

Choice of prediction problem. Here we consider a "model" to be a map from features to the prediction of interest, which differs from other common uses of this word. The completeness of a model depends on the specified prediction problem: With the same features, a model of the effect of a price cut on sales might be able to predict the aggregate effect (e.g. a 5% increase in sales) very well but unable to predict which consumers would increase their purchases.

3.4 Evaluating Models

Predictive accuracy is only one of many criteria that matter for selecting theories. Economists typically also value parsimony, portability, and tractability, and trade them off against accuracy and each other when selecting models. This paper is not designed to be about the tension between these criteria. Rather, our definition of completeness is meant as a tool to facilitate making such tradeoffs.

A high-level analogy is to the idea of polynomial-time approximation algorithms for NP-hard optimization problems. In the theory of approximation algorithms, there is a tension between efficient algorithms (which run quickly and produce sub-optimal solutions in general) and the optimal solution (which may be hard to find, but whose value cannot be improved). To even state this tension, one needs the notion of

[8] Fix any random variable $X = (X_1, X_2)$ taking values in $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2$ and let P denote the joint distribution of (X_1, X_2, Y). As in the main text, define f^* to be the mapping from $\mathcal{X}$ to $\mathcal{Y}$ satisfying $f^*(x) = \arg\min_{y' \in \mathcal{Y}} \mathbb{E}_P[\ell(y', y) \mid x]$ for all $x \in \mathcal{X}$. Now define P_1 to be the marginal distribution of P on $\mathcal{X}_1 \times \mathcal{Y}$ and define f_1^* to be the mapping from $\mathcal{X}_1$ to $\mathcal{Y}$ satisfying $\overline{f}_1^*(x) = \arg\min_{y' \in \mathcal{Y}} \mathbb{E}_{P_1}[\ell(y', y) \mid x_1]$ for all $x_1 \in \mathcal{X}_1$. Then, $\mathbb{E}_{P_1}(f_1^*(X_1), Y) \geq \mathbb{E}_P(f^*(X), Y)$; that is, irreducible error is higher given the smaller set of features.

"the optimal solution" in the first place. This is obvious in context of optimization problems and efficient algorithms for them, but as far as we know, the analog of the "optimal solution" is not in common use in experimental settings. Evaluating the completeness of a model makes it possible to talk about the tension between simple and complex models, as well as related trade-offs, with reference to how close these models come to the best achievable performance.

Restrictive versus Complete Models. In general, the more flexible a model is, the higher its completeness. At the extreme, a model class $\mathcal{F}_\Theta$ that includes all possible mappings from $\mathcal{X}$ to $\mathcal{Y}$ achieves full completeness. But such a model is also vacuous, as it has no falsifiable predictions: For a fixed level of predictive accuracy, we thus prefer models that are more restrictive. Fudenberg et al. (2020) provide an algorithmic measure of a model's "restrictiveness" by evaluating the completeness of the model on a range of synthetic data. Since the best achievable error varies substantially across data sets, low absolute error on these data sets is not enough to conclude that a model is unrestrictive; likewise, high absolute error does not imply that the model imposes substantial restrictions. But a model that is complete on all data is not restrictive. The paper uses completeness together with restrictiveness to evaluate several economic models.

Interpretable Versus Predictive Models. In many applications, researchers may prefer to sacrifice some predictive power and completeness to use a model that is easier to interpret, for example using a model of preferences to predict choice as opposed to a black box. Having a measure for completeness tells us how much we sacrifice in terms of predictive power by requiring the model to be interpretable. In some cases, such as the CPT model in Section 5.1, it turns out that simple and intepretable models achieve completeness comparable to that of black box algorithms, meaning this tradeoff is not present.[9] On the other hand, the Rabin (2000) model achieves only partial completeness—these could be because a better, interpretable model exists, or it could be because human behavior in this domain is fundamentally complex and cannot be capture by a simple model. Having the measure of completeness makes it possible to describe this tradeoff.

[9] Fudenberg and Liang (2019) demonstrates a similar point for the domain of initial play in matrix games.

4 Estimating Completeness from Finite Data

We now discuss how to estimate completeness from finite data. When the feature space X is "small" and there are a large number of observations of Y for each unique $x \in X$, then a natural estimator for the irreducible error is the performance of a lookup table, which simply learns the best prediction of Y for each x. While the assumption that there is a large number of observations per x may seem demanding, it turns out to be satisfied for a potentially large number of experimental data sets, including the ones we subsequently study. We view a substantial part of the contribution of this paper as demonstrating that irreducible error can be approximated quite simply across a diverse range of experimental contexts.

When nonparametric estimation of irreducible error via a lookup table is not feasible, then econometric methods such as splines, sieves, and black box machine learning algorithms (e.g. lasso regression) can potentially be used as substitutes.[10] In those cases, the estimate of the ratio of the model's improvement relative to the improvement achieved by the black box can be seen as an upper bound on the completeness of the model, as in Peysakhovich and Naecker (2017b).

We subsequently describe in detail the estimators we use in this paper, with Section 4.1 describing our estimators for the expected prediction errors in (3), and Section 4.2 describing our estimator for completeness.

4.1 Estimators for Expected Prediction Errors

Our approach applies to an arbitrary set $\mathcal{F}$ of maps from $\mathcal{X}$ to $\mathcal{Y}$. The special cases $\mathcal{F} = \{f_{\text{naive}}\}$, $\mathcal{F} = \mathcal{F}_\Theta$, and $\mathcal{F} = \mathcal{X}^\mathcal{Y}$ (i.e. the unrestricted set of all possible maps from features in $\mathcal{X}$ into outcomes in $\mathcal{Y}$) respectively return the desired prediction errors $\mathcal{E}_P(f_{\text{naive}})$, $\mathcal{E}_P(f^*_\Theta)$, and $\mathcal{E}_P(f^*)$ from (3).

In each case, we select a mapping from $\mathcal{F}$ based on a set of training observations, and evaluate the out-of-sample prediction error of the chosen mapping. Our estimator for the expected prediction error is the tenfold cross-validated out-of-sample error. We describe this procedure in some detail as it may be new for some readers, but it is standard, and familiar readers may skip directly to to Section 5.

[10] These methods may have better finite-sample performance when suitable regularity assumptions apply, but those assumptions may not be directly testable.

1. **Split data into $K = 10$ folds.** All of the available data is randomly split into K equally-sized disjoint subsets $Z_1, \ldots, Z_K$. In each iteration $1 \leq i \leq K$ of the procedure, the subset $Z^i_{\text{test}} \equiv Z_i$ is identified as the test data and the remaining subsets $Z^i_{\text{train}} \equiv \cup_{j \neq i} Z_j$ are used as training data.

2. **Select the mapping from $\mathcal{F}$ that best fits the training data.** For each iteration $i \in \{1, \ldots, K\}$ and mapping f, the in-sample performance of f for predicting the observations in Z^i_{train} is

$$e(f, Z^i_{\text{train}}) = \frac{1}{|Z^i_{\text{train}}|} \sum_{(x,y) \in Z^i_{\text{train}}} \ell(f(x), y)).$$

   This is a sample analog of the expected prediction error in (1). The best-fit model for the i-th training set is $f_i \equiv \arg\min_{f \in \mathcal{F}} e(f, Z^i_{\text{train}})$.[11]

3. **Evaluate how well the chosen mapping performs out of sample.** The selected model $\hat{f}$ is subsequently evaluated on the disjoint set of test observations in Z_{test}. and this model's out-of-sample performance on the i-th test set is

$$\text{CV}_i = e(f_i, Z^i_{\text{test}}). \tag{4}$$

4. **Average over out-of-sample errors.** The average out-of-sample error across the K test sets is

$$\text{CV}(\mathcal{F}, \{Z_i\}_{i=1}^K) = \frac{1}{K} \sum_{i=1}^K \text{CV}_i. \tag{5}$$

4.2 Estimator for Completeness

Define

$$\begin{aligned} \widehat{\mathcal{E}}_{\text{naive}} &\equiv \text{CV}(\{f_{\text{naive}}\}, \{Z_i\}_{i=1}^K) \\ \widehat{\mathcal{E}}_{\Theta} &\equiv \text{CV}(\mathcal{F}_{\Theta}, \{Z_i\}_{i=1}^K) \\ \widehat{\mathcal{E}}_{\text{best}} &\equiv \text{CV}(\mathcal{X}^{\mathcal{Y}}, \{Z_i\}_{i=1}^K). \end{aligned}$$

[11] When there are multiple minimizers, choose between them randomly.

In the main text, we refer to these estimates simply as prediction errors, understanding that they are finite-data estimates. In place of the theoretical completeness measure described in (3), we compute the empirical ratio

$$\frac{\widehat{\mathcal{E}}_{\text{naive}} - \widehat{\mathcal{E}}_{\Theta}}{\widehat{\mathcal{E}}_{\text{naive}} - \widehat{\mathcal{E}}_{\text{best}}} \tag{6}$$

from our data. The tables we report in the subsequent applications in Sections 5.1-5.3 are structured as follows:

	Error	Completeness
Naive Benchmark	$\widehat{\mathcal{E}}_{\text{naive}}$	0%
Economic Model	$\widehat{\mathcal{E}}_{\Theta}$	$100 \times \left(\widehat{\mathcal{E}}_{\text{naive}} - \widehat{\mathcal{E}}_{\Theta}\right) / \left(\widehat{\mathcal{E}}_{\text{naive}} - \widehat{\mathcal{E}}_{\text{best}}\right)$ %
Irreducible Error	$\widehat{\mathcal{E}}_{\text{best}}$	100%

Theoretical Guarantees. The empirical quantities $\widehat{\mathcal{E}}_{\text{naive}}$, $\widehat{\mathcal{E}}_{\Theta}$, and $\widehat{\mathcal{E}}_{\text{best}}$ are consistent estimators for $\mathcal{E}_P(f_{\text{naive}})$, $\mathcal{E}_P(f^*_{\Theta})$, and $\mathcal{E}_P(f^*)$, respectively (Hastie et al., 2009), and the empirical estimate of completeness in (6) is a consistent estimator for (3).

These estimates are good approximations for the theoretical quantities when the number of observations is sufficiently large. In particular, for $\widehat{\mathcal{E}}_{\text{best}}$ to be a good approximation of the irreducible noise $\mathcal{E}_P(f^*)$, the analyst must have access to a sufficiently large number of observations for each distinct $x \in \mathcal{X}$. This can be a demanding criterion. To evaluate whether we have "enough" data in our applications, we consider two tests in Appendices A.1 and A.2. First, we compare the performance of the lookup table with a machine learning algorithm that is better suited to smaller data sets (bagged decision trees). The out-of-sample performances are comparable, but the lookup table has a lower error for all of our applications (see Appendix A.1). Second, we investigate whether the out-of-sample performance of the lookup table has converged by evaluating its performance on subsamples of our data. The prediction errors using just 70% of the data are very close to those using all of our data. These analyses suggest that our estimate for irreducible error is a reasonable approximation in each of our applications. Additionally, we report for each model class a heuristic

(but popular) estimate of the standard error, which is

$$\sqrt{\frac{1}{K}\sum_{k=1}^{K}(\text{CV}_k - \overline{CV})^2}$$

with CV_i as defined in (4) and $\overline{CV} = \frac{1}{K}\sum CV_k$ denoting the average cross-validated error across the folds. For all of the data sets we analyze, the standard errors are small relative to the magnitudes of the prediction errors.

In general, the condition that the data includes many observations per feature is easier to satisfy in experimental settings, where the experimentalist has control over the structure of the data and can choose to acquire a large number of observations for each of a fixed set of feature values. For example, in the data sets that we consider, there is an average of 179 observations per unique x for estimation of a mean (Section 5.1), 50 observations per unique x for estimation of a mode among three outcomes (Section 5.2), and 164 observations per unique x for estimation of a mean (Section 5.3).

5 Three Applications

5.1 Domain #1: Assigning Certain Equivalents to Lotteries

Background and Data. An important question in economics is how individuals evaluate risk. In addition to Expected Utility models (von Neumann and Morgenstern, 1944; Savage, 1954; Samuelson, 1952), one of the most influential models of decision-making under risk is Cumulative Prospect Theory (Tversky and Kahneman, 1992). This model provides a flexible family of risk preferences that accommodates various behavioral anomalies, including reference-dependent preferences and nonlinear probability weighting.

A standard experimental paradigm for eliciting risk preferences, and thus for evaluating these models, is to ask subjects to report certainty equivalents for lotteries—i.e. the lowest certain payment that the individual would prefer over the lottery. We consider a data set from Bruhin et al. (2010), which includes 8906 certainty equivalents elicited from 179 subjects, all of whom were students at the University of Zurich or

the Swiss Federal Institute of Technology Zurich. Subjects reported certainty equivalents for the same 50 two-outcome lotteries, half over positive outcomes (e.g. gains) and half over negative outcomes (e.g. losses).

Prediction Task and Models. In this data set, the outcomes are the reported certainty equivalents for a given lottery, and the features are the lottery's two possible monetary prizes $\overline{z}$ and $\underline{z}$, and the probability p of the first prize. A prediction rule is any function that maps the tuple $(\overline{z}, \underline{z}, p)$ into a prediction for the certainty equivalent, i.e. a function $f : \mathbb{R} \times \mathbb{R} \times [0, 1] \to \mathbb{R}$. We use mean-squared error as the loss-function: In a test set of n observations $\{(\overline{z}_i, \underline{z}_i, p_i; y_i)\}_{i=1}^n$—where $(\overline{z}_i, \underline{z}_i, p_i)$ is the lottery shown in observation i and y_i is the reported certainty equivalent—the mean-squared error of f is

$$\frac{1}{n} \sum_{i=1}^{n} \left(f(\overline{z}_i, \underline{z}_i, p_i) - y_i \right)^2 .$$

We evaluate a prediction rule based on *Cumulative Prospect Theory* (CPT)[12], which predicts

$$v^{-1} \left(w(p)v(\overline{z}) + (1 - w(p))v(\underline{z}) \right)$$

for each lottery, where w is a probability weighting function and v is a value function. We follow Bruhin et al. (2010) in our choice of functional forms:

$$v(z) = \begin{cases} z^\alpha & \text{if } z > 0 \\ -(-z^\beta) & \text{if } z \leq 0 \end{cases} \qquad w(p) = \frac{\delta p^\gamma}{\delta p^\gamma + (1-p)^\gamma}. \tag{7}$$

This model has four free parameters: $\alpha, \beta, \delta, \gamma \in \mathbb{R}_+$.

Finally, as a naive benchmark, we predict the *expected value* of the lottery, which is $p\overline{z} + (1 - p)\underline{z}$. As we report in Appendix C.1 completeness of CPT is very similar with other benchmarks such as risk-averse variants of Expected Utility.

Results. The following table reveals that CPT's out-of-sample predictions improve upon the Expected Value benchmark:[13]

[12] CPT and the original Prospect Theory are equivalent on the 2-outcome lotteries we consider.

[13] The parameter estimates for CPT are $\alpha = 0.8, \beta = 1.2, \delta = 0.9$, and $\gamma = 0.5$.

	Error
Expected Value	103.81
	(4.00)
CPT	68.43
	(4.18)

Table 2: CPT predicts better than EV.

CPT does much better than the expected value benchmark, but falls far short of perfect prediction. It is difficult to interpret the size of CPT's error based on Table 2 alone. It is not surprising that these models do not achieve perfect prediction, as we expect different subjects to report different certainty equivalents for the same lottery, and thus a model that provides the same prediction for each $(\overline{z}, \underline{z}, p)$ input cannot possibly predict every reported certainty equivalent. But besides the intrisic variation in certainty equivalents for any fixed lottery, another potential source of error is the functional form imposed in (7). Could a different (potentially more complex) specification for the value function or probability weighting function lead to large gains in prediction? Relatedly, might there be other features of risk evaluation, yet unmodelled, which lead to even larger improvements in prediction?

To separate these sources of error, we need to understand how CPT's error compares to the irreducible error for this data. We estimate the irreducible error in this problem using a lookup table, where each of the 50 unique lotteries is mapped to the average certainty equivalent for that lottery in the training data. With 179 observations for each of the lotteries, we are able to approximate the mean certainty equivalent for each lottery using the training data, thus (approximately) minimizing the out-of-sample prediction error. We report the estimated irreducible error and its standard error in Table 3.

	Error	Completeness
Naive Benchmark	103.81	0%
	(4.00)	
CPT	68.43	92%
	(4.18)	
Irreducible Error	65.58	100%
	(3.00)	

Table 3: CPT is nearly complete for prediction of our data.

Table 3 shows that the CPT prediction error is almost as low as the irreducible error—CPT achieves 92% of the feasible reduction in prediction error over the naive baseline.[14] Thus this data suggests that there is no reason to try to construct more predictive theories that use only the features $(\overline{z}, \underline{z}, p)$.[15] To further reduce error, we would need to expand the set of variables on which the model depends. For example, as we discuss in Section 10, we could group subjects using auxiliary data such as their evaluations of other lotteries or response times, or use non-choice data, such as the hypothetical choices in Bernheim et al. (2020).

We note that our completeness measure does not imply that in general CPT is a nearly-complete model for predicting certainty equivalents, since the completeness measure we obtain is determined from a specific data set, so its generalizability depends on the extent to which that data is representative. However, Peysakhovich and Naecker (2017a) find that CPT approximates the performance of regularized regression models for a data set of 3 outcome lotteries, which suggests that our finding is robust to certain 3 outcome lotteries, although the results of Bernheim and Sprenger (2020) show this will not be true for all of them.[16]

[14] In Appendix C.2, we show that completeness is nearly identical for other popular functional form specifications of CPT.

[15] It is hard to know whether the high completeness of CPT (in the specified functional form) comes from its good match to actual behavior or because it is flexible enough to mimic most functions in $\mathcal{X}^{\mathcal{Y}}$. This question is explored in Fudenberg et al. (2020).

[16] The specification of CPT in Peysakhovich and Naecker (2017a) sets $\delta = 1$ and thus has one fewer free parameter, so its model error may be higher.

5.2 Domain #2: Initial Play in Games

Background and Data. In many game theory experiments, equilibrium analysis is a poor predictor of the choices that people make when they encounter a new game. This has led to models of initial play that depart from equilibrium theory, for example the level-k models of Stahl and Wilson (1994) and Nagel (1995), the Poisson Cognitive Hierarchy model (Camerer et al., 2004), and the related models surveyed in Crawford et al. (2013). These models represent improvements over the equilibrium predictions, but we do not know whether these models exhaust the main regularities in initial play.

Prediction Task and Models. We consider prediction of the action chosen by the row player in a given instance of play of a 3×3 normal-form game. The available features are the 18 entries of the payoff matrix, and a prediction rule is any map $f : \mathbb{R}^{18} \to \{a_1, a_2, a_3\}$ from 3×3 payoff matrices to row player actions.

For each prediction rule f and test set of observations $\{(g_i, a_i\}_{i=1}^n$—where g_i is the payoff matrix in observation i, and a_i is the observed row player action—we evaluate error using the *misclassification rate*

$$\frac{1}{n} \sum_{i=1}^{n} \mathbb{1} \left(f(g_i) \neq a_i \right).$$

This is the fraction of observations where the predicted action was not the observed action.

As a naive baseline, we consider guessing uniformly at random for all games, which yields an expected misclassification rate of 2/3. We use this benchmark to evaluate a prediction rule based on the *Poisson Cognitive Hierarchy Model* (PCHM), which supposes that there is a distribution over players of differing levels of sophistication: The *level-0* player randomizes uniformly over his available actions, while the *level-1* player best responds to level-0 play (Stahl and Wilson, 1994, 1995; Nagel, 1995). Camerer et al. (2004) defines the play of level-k players, $k \geq 2$, to be the best response to a perceived distribution

$$p_k(h) = \frac{\pi_\tau(h)}{\sum_{l=0}^{k-1} \pi_\tau(l)} \qquad \forall\ h \in \mathbb{N}_{<k} \tag{8}$$

over (lower) opponent levels, where π_τ is the Poisson distribution with rate parameter τ.[17] We can derive a predicted distribution over actions by supposing that the proportion of level-k players in the population is proportional to $\pi_\tau(k)$. Assuming this is the true distribution of play, the misclassification rate is minimized by predicting the mode of this distribution. We define the PCHM prediction to be that mode.

Comparison Across Games. We compare the performance of the PCHM relative to the best achievable performance on three subsamples of a data set from Fudenberg and Liang (2019).[18] Our full data set consists of 23,137 total observations of initial play from 486 3×3 matrix games, where observations are pooled across all of the subjects and games.[19,20]

The first subsample, *Game Set A*, consists of the 16,660 observations of play from the 359 games with no strictly dominated actions.[21] *Game Set B* consists of the 7,860 observations of play from the 161 games in which the profile that maximizes the sum of the players' payoffs is much larger (at least 20% of the largest row player payoff in the game) than the highest sum of payoffs that can be achieved when the row player chooses a level-k action (for any k).[22] For example, in the game below (which is included in Game Set B), the action profile (a_2, a_2) leads to a payoff sum of 160,

[17] Throughout, we take τ to be a free parameter and estimate it from the training data.

[18] Fudenberg and Liang (2019) studied a related prediction task, namely predicting the modal row player action in a given game. For that prediction task, the best achievable error is always zero. Here we consider prediction of the action played, where the best achievable error depends on the true distribution of play.

[19] This data is an aggregate of three data sets: the first is a meta data set of play in 86 games, collected from six experimental game theory papers by Kevin Leyton-Brown and James Wright, see Wright and Leyton-Brown (2014); the second is a data set of play in 200 games with randomly generated payoffs, which were gathered on MTurk for Fudenberg and Liang (2019); the third is a data set of play in 200 games that were "algorithmically designed" for a certain model (level 1) to perform poorly, again from Fudenberg and Liang (2019).

[20] There was no learning in these experiments—subjects were randomly matched to opponents, were not informed of their partners' play, and did not learn their own payoffs until the end of the session.

[21] Specifically, we consider games where no pure action is strictly dominated by another pure action.

[22] Following Stahl and Wilson (1995) and Nagel (1995), level-0 corresponds to uniform play, and each level-k action is the best response to level-$(k-1)$ play.

but the largest payoff sum using level-k actions is 120. The difference, 40, is more than 20% of the max row player payoff in this game, 100.[23]

	a_1	a_2	a_3
a_1	$40, 40$	$10, 20$	$70, 30$
a_2	$20, 10$	$80, 80$	$0, 100$
a_3	$30, 70$	$100, 0$	$60, 60$

Finally, *Game Set C* consists of the 9,243 observations of play from the 175 games where the level 1 action's expected payoff against uniform play is much higher than the expected payoff of the next best action (specifically, it is larger by at least 1/4 of the max row player payoff in the game).

The analysis we perform for these three subsamples can be conducted for arbitrary sets of games.

Results. Below we report the estimated irreducible error and associated completeness measures for each of the three sets of games.

	Game Set A		Game Set B		Game Set C	
	Error	Completeness	Error	Completeness	Error	Completeness
Naive Benchmark	0.66	0%	0.66	0%	0.66	0%
PCHM	0.49	68%	0.44	68%	0.28	97%
	(0.006)		(0.009)		(0.004)	
Irreducible Error	0.41	100%	0.34	100%	0.27	100%
	(0.005)		(0.006)		(0.005)	

Table 4: Comparison of the completeness of PCHM across the three sets of games.

Our estimate for the irreducible error is derived using a lookup table, where each game is mapped to the action most commonly chosen in that game in the training data. Since we have on average 50 observations per game, the modal action in the training data is a good approximation for the modal action in the test data. High

[23] In this game, action a_3 is level 1, since it yields the highest expected payoff against uniform play, and action a_1 is level 2, since it is a best response against play of a_1. Because (a_1, a_1) is a pure-strategy Nash equilibrium, action a_1 is level-k for all $k \geq 2$.

irreducible error means that there is substantial heterogeneity in play, so predicting the mode still leads to a high rate of incorrect classification. Low irreducible error means that play across subjects is more coordinated on a single action. We find that the estimated irreducible error is largest—and hence, there is the most heterogeneity in play—in Data Set A, which includes only games where there are no strictly dominated actions, and smallest in Data Set C, which includes only games where the level-1 action has by far the highest expected payoff against uniform play.

Next we use the estimated irreducible errors as a benchmark to evaluate the completeness of PCHM on the three data sets. Although the PCHM achieves a better absolute prediction error in Game Set A than in Game Set B, its completeness is approximately 68% on both data sets. In contrast, the PCHM achieves 97% of the feasible reduction in prediction error in Game Set C. This means that PCHM captures essentially all of the predictable variation in games where the level 1 action clearly has the largest expected value against uniform play, while there is additional structure beyond the PCHM in Game Sets A and B. We leave to future work the question of what additional properties of the game are important determinants of the completeness of the PCHM.

5.3 Domain #3: Human Generation of Random Sequences

Background and Data. Extensive experimental and empirical evidence suggests that humans misperceive randomness, for example expecting that sequences of coin flips "self-correct" (too many Heads in a row must be followed by a Tails) and are balanced (the number of Heads and Tails are approximately the same) (Bar-Hillel and Wagenaar, 1991; Tversky and Kahneman, 1971). These misperceptions are significant not only for their basic psychological interest, but also for the ways in which misperception of randomness manifests itself in a variety of contexts: for example, investors' judgment of sequences of (random) stock returns (Barberis et al., 1998), professional decision-makers' reluctance to choose the same (correct) option multiple times in succession (Chen et al., 2016), and people's execution of a mixed strategy in a game (Batzilis et al., 2016).

A common experimental framework in this area is to ask human participants to generate fixed-length strings of k (pseudo-)random coin flips, for some small value

of k (e.g. $k = 8$), and then to compare the produced distribution over length-k strings to the output of a Bernoulli process that generates realizations from $\{H, T\}$ independently and uniformly at random (Rapaport and Budescu, 1997; Nickerson and Butler, 2009). Following in this tradition, we use the platform Mechanical Turk to collect a large dataset of human-generated strings designed to simulate the output of a *Bernoulli(0.5) process*, in which each symbol in the string is generated from $\{H, T\}$ independently and uniformly at random. To incentivize effort, we told subjects that payment would be approved only if their (set of) strings could not be identified as human-generated with high confidence.[24,25] After removing subjects who were clearly not attempting to mimic a random process, our final data set consisted of 21,975 strings generated by 167 subjects.[26]

Prediction Task, Performance Metric, and Models. We consider the problem of predicting the probability that the eighth entry in a string is H given its first seven entries. Thus the outcome here is a number in $[0, 1]$—a distribution on $\{H, T\}$—and the feature space is $\{H, T\}^7$ (note that as in the previous examples, we fit a representative-agent model and do not treat the identity of the subject as a feature).

Given a test data set $\{(s_i^1, \ldots, s_i^8)\}_{i=1}^n$ of n binary strings of length-8, we evaluate

[24] In one experiment, 537 subjects each produced 50 binary strings of length eight. In a second experiment, an additional 101 subjects were asked to each generate 25 binary strings of length eight.

[25] Subjects were informed: "To encourage effort in this task, we have developed an algorithm (based on previous Mechanical Turkers) that detects human-generated coin flips from computer-generated coin flips. You are approved for payment only if our computer is not able to identify your flips as human-generated with high confidence."

[26] Our initial data set consists of 29,375 binary strings. We chose to remove all subjects who repeated any string in more than five rounds. This cutoff was selected by looking at how often each subject generated any given string and finding the average "highest frequency" across subjects. This turned out to be 10% of the strings, or five strings. Thus, our selection criteria removes all subjects whose highest frequency was above average. This selection eliminated 167 subjects and 7,400 strings, yielding a final dataset with 471 subjects and 21,975 strings. We check that our main results are not too sensitive to this selection criteria by considering two alternative choices in Appendix D.2—first, keeping only the initial 25 strings generated by all subjects; second, removing the subjects whose strings are "most different" from a Bernoulli process under a χ^2-test. We find very similar results under these alternative criteria.

the error of the prediction rule f using mean-squared error

$$\frac{1}{n}\sum_{i=1}^{n}\left(s_i^8 - f(s_i^1,\ldots,s_i^7)\right)^2$$

where $f(s_i^1,\ldots,s_i^7)$ is the predicted probability that the eighth flip is 'H' given the observed initial seven flips $s_i^1,\ldots,s_i^7$, and s_i^8 is the actual eighth flip.[27] Note that the naive baseline of unconditionally guessing 0.5 guarantees a mean-squared prediction error of 0.25. Moreover, if the strings in the test set were truly generated via a Bernoulli(0.5) process, then no prediction rule could improve in expectation upon the naive error.[28] We expect that behavioral errors in the generation process will make it possible to improve upon the naive baseline, but do not know how much it is possible to improve upon 0.25.

In this task, the natural naive baseline is the rule that unconditionally guesses that the probability the final flip is 'H' is 0.5. We compare this baseline to prediction rules based on Rabin (2002) and Rabin and Vayanos (2010), both of which predict negatively autocorrelated sequences.[29] Our prediction rule based on Rabin (2002) supposes that subjects generate sequences by drawing sequentially without replacement from an urn containing $0.5N$ '1' balls and $0.5N$ '0' balls. The urn is "refreshed" (meaning the composition is returned to its original) every period with independent probability p. This model has two free parameters: $N \in \mathbb{Z}_+$ and $p \in [0,1]$.

Our prediction rule based on Rabin and Vayanos (2010) assumes that the first flip $s_1 \sim$ Bernoulli(0.5) while each subsequent flip s_k is distributed

$$s_k \sim \text{Ber}\left(0.5 - \alpha\sum_{t=0}^{k-2}\delta^t(2\cdot s_{k-t-1}-1)\right),$$

[27] Alternatively we could have defined the outcome to be an individual realization of H or T, so that prediction rules are maps $f:\{H,T\}^7 \to \{H,T\}$, and then evaluated error using the misclassification rate (i.e. the fraction of instances where the predicted outcome was not the realized outcome). We do not take a stand on which method is better, but note that the completeness measure can depend on which approach is used. In Appendix D.1 we show that the completeness measures are very similar using this alternative formulation.

[28] Due to the convexity of the loss function, it is possible to do worse than the naive baseline, for example by predicting 1 unconditionally.

[29] Although both of these frameworks are models of mistaken inference from data, as opposed to human attempts to generate random sequences, they are easily adapted to our setting, as the papers explain.

where the parameter $\delta \in \mathbb{R}_+$ reflects the (decaying) influence of past flips, and the parameter $\alpha \in \mathbb{R}_+$ measures the strength of negative autocorrelation.[30]

Results. Table 5 shows that both prediction rules improve upon the naive baseline. The need for a benchmark for achievable prediction is starkest in this application, as the best improvement is only 0.0008, while the gap between the achieved prediction errors and a perfect zero is large. This is not surprising—since the data is generated by subjects attempting to mimic a fair coin, we naturally expect substantial variation in the eighth flip after conditioning on the initial seven flips.

	Error
Naive Benchmark	0.25
Rabin (2002)	0.2494
	(0.0007)
Rabin and Vayanos (2010)	0.2492
	(0.0007)

Table 5: Both models improve upon naive guessing, but the absolute improvement is small.

For this problem, we can approximate the irreducible error by learning the empirical frequency with which each length-7 string is followed by 'H' in the training data. Although there are 2^7 unique initial sequences, with approximately 21,000 strings in our data set we have (on average) 164 observations per initial sequence.

[30] We make a small modification on the Rabin and Vayanos (2010) model, allowing $\alpha, \delta \in \mathbb{R}_+$ instead of $\alpha, \delta \in [0, 1)$.

	Error	Completeness
Naive Benchmark	0.25	0
Rabin (2002)	0.2494	10%
	(0.0007)	
Rabin & Vayanos (2010)	0.2492	14%
	(0.0007)	
Irreducible Error	0.2441	100%
	(0.0006)	

Table 6: The feasible reduction in prediction error over the naive baseline is small in this problem.

We find that irreducible error in this problem is 0.2441, so that naively comparing achieved prediction error against perfect prediction (which would suggest a completeness measure of at most 0.4%) grossly misrepresents the performance of the models. The existing models produce up to 14% of the achievable reduction in prediction error. This suggests that although negative autocorrelation is indeed present in the human-generated strings and explains a sizable part of the deviation from a Bernoulli(0.5) process, there is additional structure that could yet be exploited for prediction.

6 Extensions

6.1 Subject Heterogeneity

So far, we have evaluated the completeness of "representative agent" models that implement a single prediction across all subjects. When we evaluate models that allow for subject heterogeneity, the question of what is the largest achievable reduction in prediction error is still relevant, and the irreducible error for the new expanded feature set can again help us determine the size of potential error reductions. As a simple illustration, we return to our evaluation of risk preferences and demonstrate how to construct a predictive bound for certain models with subject heterogeneity.

The models that we consider extend the Cumulative Prospect Theory model introduced in Section 5.1 by allowing for three groups of subjects. To test the models, we randomly select 71 (out of 171) subjects to be test subjects, and 45 (out of 50) lotteries to be test lotteries. All other data—the 100 training subjects' choices in all lotteries, as well as the test subjects' choices in the 5 training lotteries—are used for training the models.

We first use the training subjects' responses in the training lotteries to develop a clustering algorithm to separate subjects into three groups.[31] This algorithm assigns a group number to new subjects based on the similarity of their choices in the five training lotteries to those of the training subjects in each group. Second, we use each group's training subjects' responses in the test lotteries to estimate the four free parameters for CPT. This yields three versions of CPT, one per group.

Out of sample, we first use the clustering algorithm to assign groups to the test subjects, and then use the associated models to predict each group's certainty equivalents in the test lotteries. We measure accuracy using mean-squared error, as in Section 5.1, and we again report the Expected Value prediction as a naive baseline.

	Prediction Error
Naive Benchmark	104.17
	(12.95)
CPT	57.14
	(7.17)

Table 7: Prediction Errors Achieved by Models with Subject Heterogeneity

What we find from Table 7 is very similar to what we observed in Section 5.1: Both models improve upon the naive baseline, but it is difficult to assess the size of the improvement without an appropriate benchmark.

Our approach here for estimating the irreducible error is to learn the mean response of training subjects in each group for each lottery, and predict those means. With sufficiently many training subjects, this method approximates the best possible

[31] We use a simple algorithm, k-means, which minimizes the Euclidean distance between the vectors of reported certainty equivalents for subjects within the same group.

accuracy for this prediction task. We find that although the CPT error is substantially different from zero, the model is again nearly complete.

	Prediction Error	Completeness
Naive Benchmark	104.17	0%
	(12.95)	
CPT	57.14	96%
	(7.17)	
Irreducible Error	55.45	100%
	(6.26)	

We note that because the same clustering method is used in all of the approaches, the gap between irreducible error and the prediction errors does not shed light on how much predictions could be improved by better ways of grouping the subjects. The development of better clustering techniques is an interesting avenue for future work.[32]

6.2 Comparing Feature Sets

In the main text, we considered a fixed feature set $\mathcal{X}$, and evaluated the completeness of different models for prediction given this feature set. We can alternatively compare irreducible error across different feature sets as a way of contrasting the predictive limits of those features. We illustrate this comparison by revisiting our problem from Section 5.3—predicting human generation of randomness—and considering three feature sets.

The first feature set, $\mathcal{X}_{1:7}$, is our main feature set, which consists of the initial seven flips. Define $\mathcal{X}_{4:7} = \{H, T\} \times \{H, T\} \times \{H, T\}$ as the feature set corresponding to flips 4–7, and $\mathcal{X}_H = \{0, 1, 2 \ldots, 7\}$ as the number of 'H' realizations in the first seven flips. Interpreted as lookup tables, these new feature sets correspond to "compressed"

[32] A comparison of the irreducible error under clustering, 55.45, with the irreducible error from Section 5.1, 65.58, sheds light on the size of predictive gains achieved by the present method for clustering.

lookup tables built on different properties of the initial seven flips, where strings are partitioned based on certain properties. We can estimate irreducible error by predicting the average continuation probability of 'H' among all strings in the same partition element.

Table 8: Comparison of the value of various feature sets.

	Error	Completeness
Naive Benchmark	0.25	0%
Irreducible Error for $\mathcal{X}_{4:7}$	0.2478	36%
	(0.0010)	
Irreducible Error for $\mathcal{X}_H$	0.2464	59%
	(0.0009)	
Irreducible Error for $\mathcal{X}_{1:7}$	0.2441	100%
	(0.0006)	

We find that the feature sets $\mathcal{X}_{4:7}$ and $\mathcal{X}_H$ achieve large fractions of the achievable improvement from using $\mathcal{X}_{1:7}$. For example, using only the number of Heads as a feature, it is possible to achieve 59% of the achievable reduction of the full structure of the initial flips. Using only the most recent three flips achieves 36% of the reduction from using all seven initial flips. On the other hand, the gap between irreducible error for $\mathcal{X}_{4:7}$ and for $\mathcal{X}_{1:7}$ demonstrates that there is predictive content in flips 1–3 beyond what is captured in flips 4–7.

The feature set $\mathcal{X}_{1:7}$ could be expanded to create richer feature sets, and it would be interesting to consider what additional features might significantly improve predictive accuracy, for example "neuroeconomic" data such as the speed with which the strings were entered, or demographic data such as age or education.[33] The exercise in Section 6.1, in which we used subject types (determined based on choices in auxiliary problems), illustrates yet another way to expand the feature set. As

[33] As another example: recent work by Bernheim et al. (2020) test how well a model of Cumulative Prospect Theory that is trained on two-outcome lotteries predicts certainty equivalents for three-outcome lotteries. It finds that these "cross-domain" predictions can be improved using additional non-choice features (e.g. survey responses).

we have shown above, comparing irreducible error across different feature sets is one potentially useful approach for measuring the predictive value of those features.[34]

7 Conclusion

When evaluating the predictive performance of an economic model, it is important to know not just whether the model is predictive, but also how complete its predictive performance is. Thus we should compare the prediction errors achieved by our models against the best achievable error for that problem, namely the irreducible error. What is perhaps striking is that irreducible error can be precisely estimated in certain prediction problems of interest in experimental economics. We demonstrate three domains in which completeness can help us evaluate the performance of existing models. Occasionally, as we found in Section 5.1, a model that has large prediction errors may nevertheless be nearly complete given its feature set.

We conclude with a brief discussion of our completeness measure, its limitations, and possibilities for extension.

Counterfactuals. Economic models are often used to provide counterfactual predictions about the impact of new policies. Of course, if there is no data about such policies, these counterfactual predictions rely on untested intuitions about the robustness of various forces that drive behavior. Suppose for example that the price variation in our data only comes from price changes by firms, and we want to predict the effect of a sales tax. We might conjecture that the price effects are the same as before, but in some cases consumers might be either more or less willing to accept a price increase imposed by the government. With or without an economic theory, any attempt to extrapolate from data in settings without sales taxes to the effects of sales taxes requires an untested hypothesis. And if we do have representative data on the past effect of sales taxes, the prediction problem does not involve a substantive counterfactual.[35]

Experimental Data. Experimental economists have a degree of control over the

[34] Note that the value of individual features will in general depend on what other features are available.

[35] Except in the trivial sense that any extrapolation from past data to future outcomes requires some form of inductive hypothesis.

scope of their data that is not available in field studies. In particular, the experimentalist can choose to acquire a large number of observations for a fixed input space, so that nonparametric estimation of irreducible error for those inputs is feasible. Thus estimating completeness for laboratory data is feasible in many instances, as illustrated in the three applications in this paper. The main tradeoff is between gathering more instances of observations for a given set of feature values, versus ranging over a larger set of feature values. With a sufficiently large budget, both may be possible.

Alternative Measures of Completeness. In some cases, it may be possible to indirectly evaluate irreducible noise. For example, an interesting analogy to our approach to completeness is found in the literature on inheritability. Biologists have discovered a gap between two different methodologies for discovering how much of a particular outcome (say propensity to have a disease) is heritable, dubbed the 'missing heritability problem' (Manolio et al. (2009)). Traditional methods of measuring heritability, such as through carefully controlled twin studies, do not attempt to isolate individual genes. Newer measurement techniques instead allow us to postulate individual genes as the carrier of heritability. Yet for many outcomes, the explanatory power of individual genes has proven far smaller (sometimes by an order of magnitude) than overall measures of heritability suggest. This gap has motivated further theorizing and measurement to isolate where the "missing heritability" may lie. Roughly speaking, the aggregate measures of heritability are in effect being used as an analog of our completeness metric for the specific gene-based theories.

Measuring Portability. One important question is how to compare the transferability of models across domains. Indeed, we may expect that economic models that are outperformed by machine learning models in a given domain have higher transfer performance outside of the domain. In this sense, within-domain completeness may provide an insufficient measure of the "overall completeness" of the model, and we leave development of such notions to future work.

References

Apesteguia, J. and M. Ballester (2020): "Separating Predicted Randomness from Noise," Working Paper.

Bar-Hillel, M. and W. Wagenaar (1991): "The Perception of Randomness," *Advances in Applied Mathematics*, 12, 428–454.

Barberis, N., A. Shleifer, and R. Vishny (1998): "A Model of Investor Sentiment," *Journal of Financial Economics*, 49, 307–343.

Batzilis, D., S. Jaffe, S. Levitt, J. A. List, and J. Picel (2016): "How Facebook Can Deepen our Understanding of Behavior in Strategic Settings: Evidence from a Million Rock-Paper-Scissors Games," Working Paper.

Bernheim, D., C. Exley, J. Naecker, and C. Sprenger (2020): "The Model You Know: Generalizability and Predictive Power of Models of Choice Under Uncertainty," Working Paper.

Bernheim, D. and C. Sprenger (2020): "Direct Tests of Cumulative Prospect Theory," Working Paper.

Bodoh-Creed, A., J. Boenhke, and B. Hickman (2019): "Using Machine Learning to Explain Price Dispersion," Working Paper.

Bourgin, D. D., J. C. Peterson, D. Reichman, T. L. Griffiths, and S. J. Russell (2019): "Cognitive Model Priors for Predicting Human Decisions," *CoRR*, abs/1905.09397.

Bruhin, A., H. Fehr-Duda, and T. Epper (2010): "Risk and Rationality: Uncovering Heterogeneity in Probability Distortion," *Econometrica*, 78, 1375–1412.

Camerer, C. F., T.-H. Ho, and J.-K. Chong (2004): "A cognitive hierarchy model of games," *The Quarterly Journal of Economics*, 119, 861–898.

Camerer, C. F., G. Nave, and A. Smith (2019): "Dynamic unstructured bargaining with private information: theory, experiment, and outcome prediction via machine learning," *Management Science*, 65, 1867–1890.

Chen, D., K. Shue, and T. Moskowitz (2016): "Decision-Making under the Gambler's Fallacy: Evidence from Asylum Judges, Loan Officers, and Baseball Umpires," *Quarterly Journal of Economics*, 131, 1181–1242.

Crawford, V. P., M. A. Costa-Gomes, and N. Iriberri (2013): "Structural models of nonequilibrium strategic thinking: Theory, evidence, and applications," *Journal of Economic Literature*, 51, 5–62.

Diebold, F. and L. Kilian (2001): "Measuring Predictability: Theory and Macroeconomic Applications," *Journal of Applied Econometrics*, 16, 657–669.

Erev, I., A. E. Roth, R. L. Slonim, and G. Barron (2007): "Learning and

equilibrium as useful approximations: Accuracy of prediction on randomly selected constant sum games," *Economic Theory*, 33, 29–51.

Fudenberg, D., W. Gao, and A. Liang (2020): "Quantifying the Restrictiveness of Theories," Working Paper.

Fudenberg, D. and A. Liang (2019): "Predicting and Understanding Initial Play," *American Economic Review*, 109, 4112–41.

Gneiting, T. and A. E. Raftery (2007): "Strictly Proper Scoring Rules, Prediction, and Estimation," *Journal of the American Statistical Association*, 102, 359–378.

Hastie, T., R. Tibshirani, and J. Friedman (2009): *The Elements of Statistical Learning*, Springer.

Karmarkar, U. (1978): "Subjectively weighted utility: A descriptive extension of the expected utility model," *Organizational Behavior & Human Performance*, 21, 67–72.

Kleinberg, J., H. Lakkaraju, J. Leskovec, J. Ludwig, and S. Mullainathan (2018): "Human Decisions and Machine Predictions," *The Quarterly Journal of Economics*, 133, 237–293.

Manolio, T. A., F. S. Collins, N. J. Cox, D. B. Goldstein, L. A. Hindorff, D. J. Hunter, M. I. McCarthy, E. M. Ramos, L. R. Cardon, A. Chakravarti, et al. (2009): "Finding the missing heritability of complex diseases," *Nature*, 461, 747.

Nagel, R. (1995): "Unraveling in Guessing Games: An Experimental Study," *American Economic Review*, 85, 1313–1326.

Nickerson, R. S. and S. F. Butler (2009): "On producing random binary sequences," *The American Journal of Psychology*, 122, 141–151.

Noti, G., E. Levi, Y. Kolumbus, and A. Daniely (2016): "Behavior-Based Machine-Learning: A Hybrid Approach for Predicting Human Decision Making," *CoRR*, abs/1611.10228.

Peysakhovich, A. and J. Naecker (2017a): "Using Methods from Machine Learning to Evaluate Behavioral Models of Choice Under Risk and Ambiguity," *Journal of Economic Behavior and Organization*, 133, 373–384.

——— (2017b): "Using methods from machine learning to evaluate behavioral models of choice under risk and ambiguity," Tech. rep.

Plonsky, O., R. Apel, E. Ert, M. Tennenholtz, D. Bourgin, J. Peterson, D. Reichman, T. Griffiths, S. Russell, E. Carter, J. Cavanagh, and I. Erev (2019): "Predicting human decisions with behavioral theories and machine learning," *CoRR*, abs/1904.06866.

Plonsky, O., I. Erev, T. Hazan, and M. Tennenholtz (2017): "Psychological forest: Predicting human behavior," *AAAI Conference on Artificial Intelligence*.

Rabin, M. (2000): "Risk Aversion and Expected-utility Theory: A Calibration Theorem," *Econometrica*, 68, 1281–1292.

——— (2002): "Inference by Believers in the Law of Small Numbers," *The Quarterly Journal of Economics*, 117, 775–816.

Rabin, M. and D. Vayanos (2010): "The Gambler's and Hot-Hand Fallacies: Theory and Applications," *Review of Economic Studies*, 77, 730–778.

Rapaport, A. and D. Budescu (1997): "Randomization in Individual Choice Behavior," *Psychological Review*, 104, 603.

Samuelson, P. (1952): "Probability, Utility, and the Independence Axiom," *Econometrica*, 20, 670–678.

Savage, L. (1954): *The Foundations of Statistics*, J. Wiley.

Selten, R. (1991): "Properties for a Measure of Predictive Success," *Mathematical Social Sciences*, 21, 153–167.

Stahl, D. O. and P. W. Wilson (1994): "Experimental evidence on players' models of other players," *Journal of Economic Behavior and Organization*, 25, 309–327.

——— (1995): "On players' models of other players: Theory and experimental evidence," *Games and Economic Behavior*, 10, 218–254.

Tversky, A. and D. Kahneman (1971): "The Belief in the Law of Small Numbers," *Psychological Bulletin*, 76, 105.

——— (1992): "Advances in Prospect Theory: Cumulative Representation of Uncertainty," *Journal of Risk and Uncertainty*, 5, 297–323.

von Neumann, J. and O. Morgenstern (1944): *Theory of Games and Economic Behavior*, Princeton University Press.

Wright, J. R. and K. Leyton-Brown (2014): "Level-0 meta-models for predicting human behavior in games," *Proceedings of the fifteenth ACM conference on Economics and computation*, 857–874.

Appendix

A How Good is Our Estimate of Irreducible Error?

In the main text, we present an approach for estimating irreducible error, where given training data Z_{train}, we estimate a "lookup table" function $\hat{f}_{LT}$ which satisfies

$$\hat{f}_{LT}(x) \in \arg\min_{y \in \mathcal{Y}} \sum_{(x,y') \in Z_{\text{train}}} \ell(y, y') \qquad \forall x \in \mathcal{X}.$$

Below we investigate whether the data sets we study are large enough for the out-of-sample error of the lookup table to be a good approximation for irreducible error.

In Section A.1, we compare the out-of-sample performance of the lookup table with that of bagged decision trees, an algorithm that works better on smaller quantities of data. We find that in each of our applications, the two prediction errors are similar, and the lookup table weakly outperforms bagged decision trees. In Section A.2, we study the sensitivity of the lookup table's performance to the quantity of data. The predictive accuracies achieved using our full data sets are very close to those achieved using, for example, just 70% of the data. This again suggests that only minimal improvements in predictive accuracy are feasible from further increases in data size.

A.1 Comparison with Scalable Machine Learning Algorithms

One way to evaluate whether the out-of-sample performance of the lookup table approximates the best possible prediction accuracy is to compare it with the performance of other machine learning algorithms. Below we compare the lookup table with a bagged decision tree algorithm (also known as bootstrap-aggregated decision trees). This algorithm creates several bootstrapped data sets from the training data by sampling with replacement, and then trains a **decision tree** on each bootstrapped training set. Decision trees are nonlinear prediction models that recursively partition the feature space and learn a (best) constant prediction for each partition element. The prediction of the bagged decision tree algorithm is an aggregation of the predictions of individual decision trees. When the loss function is mean-squared error, the

decision tree ensemble predicts the average of the predictions of the individual trees. When the loss function is misclassification rate, the decision tree ensemble predicts based on a majority vote across the ensemble of trees.

Table 9 shows that for each prediction problem, the error of the bagged decision tree algorithm is comparable to and slightly worse than that of the lookup table. These results again suggest that our estimate of irreducible error is a reasonable approximation.

	Risk	Games A	Games B	Games C	Sequences
Bagged Decision Trees	65.65	0.45	0.36	0.29	0.2442
	(0.10)	(0.004)	(0.005)	(0.004)	(0.0005)
Lookup Table	65.58	0.41	0.34	0.27	0.2441
	(3.00)	(0.005)	(0.006)	(0.005)	(0.0006)

Table 9: The lookup table outperforms Bagged Decision Trees in each of our prediction problems.

A.2 Performance of the Lookup Table on Smaller Samples

Finally, we report the lookup table's cross-validated performance on random samples of $x\%$ of our data, where $x \in \{10, 20, \ldots, 100\}$. For each x, we repeat the procedure 1000 times, and report the average performance across iterations. We find that performance error flattens out for larger values of x, suggesting that the quantity of data we have is indeed large enough that further increases in the data size will not substantially improve predictive performance.

$x\%$	Risk	Games A	Games B	Games C	Sequences
10%	69.47	0.4191	0.3473	0.2729	0.2592
	(11.13)	(0.012)	(0.018)	(0.0015)	(0.0034)
20%	67.13	0.4183	0.3476	0.2718	0.2504
	(7.95)	(0.0018)	(0.024)	(0.0020)	(0.0018)
30%	66.28	0.4178	0.3472	0.2714	0.2479
	(6.51)	(0.0022)	(0.0029)	(0.0025)	(0.0014)
40%	66.25	0.4169	0.3470	0.2708	0.2464
	(5.65)	(0.0024)	(0.0032)	(0.0028)	(0.0011)
50%	65.68	0.4157	0.3459	0.2703	0.2458
	(4.59)	(0.0025)	(0.0036)	(0.0032)	(0.0010)
60%	65.68	0.4141	0.3449	0.2691	0.2452
	(4.24)	(0.0027)	(0.0040)	(0.0035)	(0.0008)
70%	65.68	0.4131	0.3435	0.2682	0.2448
	(3.95)	(0.0031)	(0.0045)	(0.0037)	(0.0007)
80%	65.68	0.4119	0.3427	0.2677	0.2445
	(3.95)	(0.0034)	(0.0046)	(0.0040)	(0.0007)
90%	65.66	0.4109	0.3416	0.2672	0.2443
	(3.71)	(0.0034)	(0.0047)	(0.0042)	(0.0007)
100%	65.58	0.4100	0.3404	0.2668	0.2441
	(3.00)	(0.0036)	(0.0051)	(0.0045)	(0.0006)

Table 10: Performance of Lookup Table $\hat{f}_{LT}$ using $x\%$ of the data, averaged over 100 iterations for each x

B Completeness and Nonparametric R^2

Consider the special case in which the loss function is mean-squared error $\ell(y', y) = (y - y')^2$, and the naive benchmark is the unconditional mean of the outcome variable, $f_{\text{naive}}(y) = \mathbb{E}_P(y)$, i.e. the unconditional mean of the outcome variable.

Because $\mathcal{E}_P(f_{\text{naive}}) = \text{var}(y)$, the R^2 for the model class is

$$R^2_\Theta = \frac{\mathcal{E}_P(f_{\text{naive}}) - \mathcal{E}_P(f^*_\Theta)}{\mathcal{E}_P(f_{\text{naive}})}$$

Alternatively, defining $f^*(x) = \mathbb{E}_P(y \mid x)$ to be the conditional mean function, we have the nonparametric R^2:

$$R^2_{\text{nonpar}} = \frac{\mathcal{E}_P(f_{\text{naive}}) - \mathcal{E}_P(f^*)}{\mathcal{E}_P(f_{\text{naive}})}.$$

Then, our completeness measure coincides with the ratio $R^2_\Theta / R^2_{\text{nonpar}}$.[36]

C Supplementary Material to Section 5.1

C.1 Alternative Naive Benchmarks

We use the expected value of the lottery as a naive benchmark in the main text. Below, we explore how completeness varies across alternative choices of naive benchmarks from three families of Expected Utility models. Each of these families specifies a value function v over money, and the predicted certainty equivalent is $v^{-1}(p \cdot v(\overline{z}) + (1-p) \cdot v(\underline{z}))$. Completeness is very similar across these different naive benchmarks and lower bounded by 89%. This reinforces our view that CPT is very complete for prediction of this data.

Power Function. The utility function over money is $v(z) = z^\alpha$ for $z \geq 0$ and $v(z) = -(-z)^\alpha$ for $z < 0$.

Constant Absolute Risk Aversion. The utility function over money is $v(z) = -e^{-\rho z}$ for all z.

Constant Relative Risk Aversion. The utility function over money is

$$v(z) = \begin{cases} \frac{z^{1-\gamma}}{1-\gamma} & \text{if } \gamma \neq 1 \\ \ln(z) & \text{if } \gamma = 1 \end{cases}$$

[36] We thank an anonymous referee for making this observation.

for $z \geq 0$, and $-v(-z)$ for $z < 0$.

We sample 1000 choices of α, ρ, and γ from $[0, 1]$ uniformly at random, and report below the corresponding histograms of completeness values.

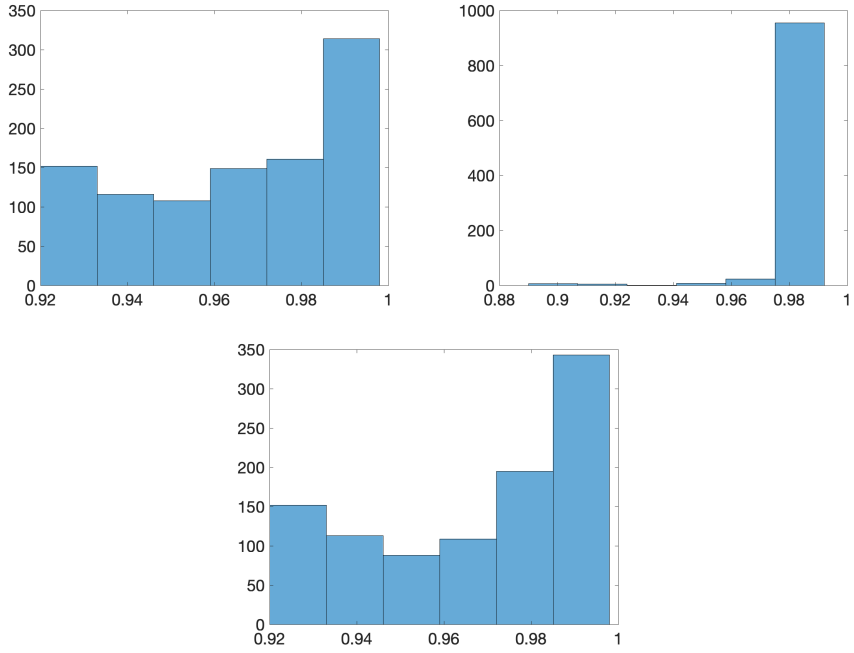


Figure 1: Completeness of CPT relative to different naive benchmarks. Clockwise from top-left: Power, CARA, CRRA.

C.2 Alternative Specifications of CPT

Besides the specification of CPT that we use in the main text, some other common alternatives include the original Tversky and Kahneman (1992) specification, which posits that the weighting function is

$$w(p) = \frac{p}{(p^\gamma + (1-p)^\gamma)^{1/\gamma}}$$

and the Karmarkar (1978) specification, which is equivalent to the one we use in the main text with δ set to 1. We report below the completeness of these alternative specifications of CPT, which are not statistically distinguishable from one another.

	Error	Completeness
Naive Benchmark	103.81	0%
	(4.00)	
CPT (Original)	68.43	92%
	(4.18)	
CPT (Karmarkar)	68.55	92%
	(3.51)	
CPT (Kahneman-Tversky)	68.77	92%
	(6.70)	
Irreducible Error	65.58	100%
	(3.00)	

Table 11: Different specifications of CPT yield comparable levels of completeness.

D Supplementary Material to Section 5.3

D.1 Robustness

Here we check how our results in Section 5.3 change when the outcome space and error function are changed so that prediction functions are maps $f : \{H, T\}^7 \to \{H, T\}$ and the error for predicting the test data set $\{(s_i^1, \ldots, s_i^8\}_{i=1}^n$ is defined to be

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}(s_i^8 \neq f(s_i^1, \ldots, s_i^7)),$$

i.e. the misclassification rate. We use as a naive benchmark the prediction rule that guesses H and T uniformly at random; this is guaranteed an expected misclassification rate of 0.50.

For this problem, we can estimate irreducible error by using a lookup table that learns the modal continuation for each sequence in $\{0, 1\}^7$. We find that the completeness of Rabin (2002) and Rabin (2000) relative to this benchmark are respectively 19% and 9%.

	Error	Completeness
Naive Benchmark	0.50	0
Rabin (2002)	0.45	19%
	(0.003)	
Rabin & Vayanos (2010)	0.475	9%
	(0.01)	
Irreducible Error	0.23	1
	(0.002)	

D.2 Different Cuts of the Data

Initial strings only. We repeat the analysis in Section 5.3 using data from all subjects, but only their first 25 strings. This selection accounts for potential fatigue in generation of the final strings, and leaves a total of 638 subjects and 15,950 strings. Prediction results for our main exercise are shown below using this alternative selection.

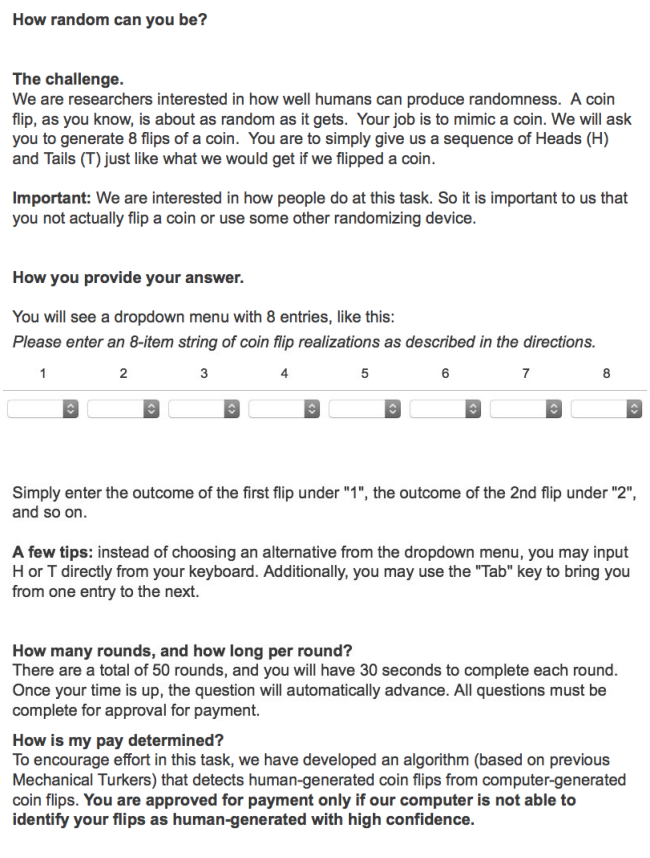
	Error	Completeness
Naive Benchmark	0.25	0
Rabin & Vayanos (2010)	0.2491	5%
	(0.0008)	
Irreducible Error	0.2326	100%
	(0.0030)	

Removing the least random subjects. For each subject, we conduct a Chi-squared test for the null hypothesis that their strings were generated under a Bernoulli process. We order subjects by p-values and remove the 100 subjects with the lowest p-values (subjects whose generated strings were most different from what we would expect under a Bernoulli process). This leaves a total of 538 subjects and 24,550 strings. Prediction results for our main exercise are shown below using this alternative selection.

	Error	Completeness
Naive Benchmark	0.25	0
Rabin & Vayanos (2010)	0.2491 (0.0005)	12%
Irreducible Error	0.2427 (0.0016)	100%

E Experimental Instructions for Section 5.3

Subjects on Mechanical Turk were presented with the following introduction screen:



How random can you be?

The challenge.
We are researchers interested in how well humans can produce randomness. A coin flip, as you know, is about as random as it gets. Your job is to mimic a coin. We will ask you to generate 8 flips of a coin. You are to simply give us a sequence of Heads (H) and Tails (T) just like what we would get if we flipped a coin.

Important: We are interested in how people do at this task. So it is important to us that you not actually flip a coin or use some other randomizing device.

How you provide your answer.

You will see a dropdown menu with 8 entries, like this:

Please enter an 8-item string of coin flip realizations as described in the directions.

1 2 3 4 5 6 7 8

Simply enter the outcome of the first flip under "1", the outcome of the 2nd flip under "2", and so on.

A few tips: instead of choosing an alternative from the dropdown menu, you may input H or T directly from your keyboard. Additionally, you may use the "Tab" key to bring you from one entry to the next.

How many rounds, and how long per round?
There are a total of 50 rounds, and you will have 30 seconds to complete each round. Once your time is up, the question will automatically advance. All questions must be complete for approval for payment.

How is my pay determined?
To encourage effort in this task, we have developed an algorithm (based on previous Mechanical Turkers) that detects human-generated coin flips from computer-generated coin flips. **You are approved for payment only if our computer is not able to identify your flips as human-generated with high confidence.**