Predictive Multiplicity in Classification

Charles T. Marx [1] **Flavio du Pin Calmon** [2] **Berk Ustun** [2]

Abstract

Prediction problems often admit competing models that perform almost equally well. This effect challenges key assumptions in machine learning when competing models assign conflicting predictions. In this paper, we define *predictive multiplicity* as the ability of a prediction problem to admit competing models with conflicting predictions. We introduce measures to evaluate the severity of predictive multiplicity, and develop integer programming tools to compute these measures exactly for linear classification problems. We apply our tools to measure predictive multiplicity in recidivism prediction problems. Our results show that real-world datasets may admit competing models that assign wildly conflicting predictions, and motivate the need to report predictive multiplicity in model development.

1 Introduction

Many machine learning algorithms are designed to fit the best model from data. Modern methods for empirical risk minimization, for example, fit a model that optimizes a specific objective (e.g., error rate) from a collection of models that obey a specific set of constraints (e.g., linear classifiers with equal FPR between groups). In an ideal scenario where stakeholders agree on such a problem formulation (Passi & Barocas, 2019) and we are given a large dataset of representative examples, we may still face a key ethical issue in building and deploying a model – namely, *there may be more than one best-fitting model*.

In machine learning, *multiplicity* refers to the ability of a prediction problem to admit multiple *competing models* that perform almost equally well. Several works mention that prediction problems can exhibit multiplicity (see e.g., Mountain & Hsiao, 1989; McCullagh & Nelder, 1989), but few discuss its implications. The work of Breiman (2001) is a major exception. In a seminal position paper, Breiman describes how multiplicity challenges the validity of explanations derived from a single predictive model: *if one can fit multiple competing models for a dataset – each of which offers a different explanation of the data-generating process – then how can we tell which explanation is correct?*

Drawing parallels between the discordant explanations of competing models and the discordant testimonies of witnesses in the motion picture "Rashomon," Breiman refers to this dilemma the *Rashomon effect*. In the context of his work, the Rashomon effect is – in fact – an argument against the misuse of explanations. Seeing how prediction problems can exhibit multiplicity, we should not use the explanations of a single model to draw conclusions about the broader data-generating process, at least until we can rule out multiplicity.

Machine learning has changed drastically since Breiman coined the Rashomon effect. Many models are now built for the sole purpose of making predictions (Shmueli et al., 2010; Kleinberg et al., 2015). In applications like lending and recidivism prediction, predictions affect people (Binns et al., 2018), and multiplicity raises new challenges when competing models assign conflicting predictions. Consider the following examples:

Recidivism Prediction: Say that a recidivism prediction problem admits competing models with conflicting predictions. In this case, a person who is predicted to recidivate by one model may be predicted not to recidivate by a competing model that performs equally well. If so, we may want to ignore predictions for this person or even forgo deployment.

Lending: Say we must explain the prediction of a loan approval model to an applicant who is denied a loan (e.g., by producing a counterfactual explanation for the prediction Martens & Provost, 2014). If competing models assign conflicting predictions, then these predictions may lead to contradictory explanations. In this case, evidence of competing models with conflicting predictions would mitigate unwarranted rationalization of the model resulting from *fairwashing* (Aïvodji et al., 2019; Laugel et al., 2019; Slack et al., 2019) or *explanation bias* (Koehler, 1991).

In this work, we define *predictive multiplicity* as the ability

[1]Haverford College [2]Harvard SEAS. Correspondence to: Charles T. Marx <cmarx@haverford.edu>, Berk Ustun <berk@seas.harvard.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Vienna, Austria, PMLR 119, 2020.

of a prediction problem to admit competing models that assign conflicting predictions. Predictive multiplicity affects key tasks in the machine learning life-cycle – from model selection (e.g., how should we choose between competing models and who should have a say in this decision) to downstream tasks in model deployment (e.g., post-hoc explanation). In such settings, presenting stakeholders with information about predictive multiplicity allows them to challenge these decisions.

Our goal is to provide stakeholders with the ability to measure and report predictive multiplicity in that same way that we measure and report test error. To this end, we introduce the following measures to evaluate predictive multiplicity in classification problems:

Ambiguity: How many individuals are assigned conflicting predictions by any competing model?

Discrepancy: What is the maximum number of predictions that could change if we were to switch the model that we deploy with a competing model?

These measures reflect meaningful quantities that warrant stakeholder participation in the model development and deployment (see e.g., Figure 1). For example, ambiguity counts the number of individuals whose predictions are determined by the decision to deploy one model over another. These individuals should have a say in model selection and should be able to contest the predictions assigned to them by a model that is deployed.

The main contributions of this paper are as follows:

1. We introduce formal measures of predictive multiplicity for classification problems: *ambiguity* and *discrepancy*.

2. We develop integer programming tools to compute ambiguity and discrepancy for linear classification problems. Our tools compute these measure *exactly* by solving non-convex empirical risk minimization problems over the set of competing models.

3. We present an empirical study of predictive multiplicity in recidivism prediction. Our results show that real-world datasets can admit competing models with highly conflicting predictions, and illustrate how reporting predictive multiplicity can inform stakeholders in such cases. For example, in the ProPublica COMPAS dataset (Angwin et al., 2016), we find that a competing model that is only 1% less accurate than the most accurate model assigns conflicting predictions to over 17% of individuals, and that the predictions of 44% of individuals are affected by model choice.

Feature Values	# Data Points Where $y_i = \pm 1$		Predictions of Best Linear Classifiers			
(x_1, x_2)	n^+	n^-	$\hat{h}_a$	$\hat{h}_b$	$\hat{h}_c$	$\hat{h}_d$
$(0,0)$	0	25	$-$	$-$	$-$	$+$
$(0,1)$	25	0	$+$	$+$	$-$	$+$
$(1,0)$	25	0	$+$	$-$	$+$	$+$
$(1,1)$	0	25	$+$	$-$	$-$	$-$

Figure 1. Classifiers with conflicting predictions may perform equally well. Here, we show 4 linear models that optimize accuracy on a 2D classification problem with 100 points. The predictions of any 2 models differ on 50 points. Thus, discrepancy is 50%. The predictions of 100 points vary based on model choice. Thus, ambiguity is 100%.

1.1 Related Work

Multiplicity. Recent work in machine learning tackles multiplicity from the "Rashomon" perspective. Fisher et al. (2018) and Dong & Rudin (2019) develop methods to measure variable importance over the set of competing models. Semenova & Rudin (2019) present a measure of the relative size of the set of competing models and show that it can characterize settings where simple models perform well. Our work differs from this stream of research in that we study competing models *with conflicting predictions* (see Figure 2). This type of multiplicity reflects irreconcilable differences between subsets of predictions – similar to the impossibility results in the fair machine learning literature (see e.g., Chouldechova, 2017; Kleinberg et al., 2016; Corbett-Davies et al., 2017).

Model Selection. Techniques to resolve multiplicity typically focus on tie-breaking or reconciliation. Classical approaches for model selection choose a single model that is "closest" to a "true model" by breaking ties on the basis of measures such as the AIC, BIC, or K-CV error (see e.g., McAllister, 2007; Ding et al., 2018). More recent approaches to model selection suggest breaking ties on the basis of secondary criteria, such as simplicity (Semenova & Rudin, 2019) or operational cost (Tulabandhula & Rudin, 2013). These approaches, which are designed to improve out-of-sample performance, may fail to achieve their intended goal on problems exhibit predictive multiplicity. In Figure 1, for example, tie-breaking between models would fail to improve out-of-sample performance (if we assume that the training data is drawn according to the true underlying distribution).

Bayesian approaches. Bayesian approaches represent multiplicity by computing a posterior distribution over models. In practice, however, one would use the posterior distribution to construct a single model for deployment (e.g., via a majority vote or a randomization procedure as in PAC Bayesian approaches, McAllester, 1999; Germain et al.,

2016). In theory, the posterior distribution could support an ad hoc analysis of predictive multiplicity – e.g., by counting the number of conflicting predictions over a set of models sampled from the posterior distribution (see e.g., Dusenberry et al., 2020). Note that this analysis may underestimate the degree of predictive multiplicity since the sample is not guaranteed to contain the full set of competing models.

2 Framework

In this section, we introduce measures of predictive multiplicity. We present measures for binary classification problems where we want a model that maximizes out-of-sample accuracy. Our measures generalize to settings where models should optimize other performance metrics (e.g., AUC), predict multiple outcomes, or satisfy additional constraints (e.g., group fairness constraints as in Zafar et al., 2019; Celis et al., 2019; Cotter et al., 2019).

Preliminaries. We start with a dataset of n examples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ where each example consists of a feature vector $\boldsymbol{x}_i = (1, x_{i1}, \ldots, x_{id}) \in \mathbb{R}^{d+1}$ and a label $y_i \in \{\pm 1\}$. Our goal is to fit a classifier $h : \mathbb{R}^{d+1} \to \{\pm 1\}$ that optimizes true risk (i.e., test error) given a hypothesis class $\mathcal{H}$. To this end, we fit a baseline classifier that optimizes empirical risk (i.e., training error):

$$h_0 \in \operatorname*{argmin}_{h \in \mathcal{H}} \hat{R}(h)$$

where $\hat{R}(h) := \frac{1}{n} \sum_{i=1}^n \mathbb{1}[h(\boldsymbol{x}_i) \neq y_i]$. This practice implicitly assumes that we h_0 to generalize, which is a reasonable assumption in our setting since we will be fitting classifiers from a simple hypothesis class (see also empirical results in Table 1).[1]

The Set of Competing Models. We measure predictive multiplicity over the set of classifiers that perform almost as well as the baseline classifier. We refer to this set as the *ϵ-level set* where ϵ is the *error tolerance*.

Definition 1 (ϵ-Level Set) *Given a baseline classifier h_0 and a hypothesis class $\mathcal{H}$, the* ϵ-level set *around h_0 is the set of all classifiers $h \in \mathcal{H}$ with an error rate of at most $\hat{R}(h_0) + \epsilon$ on the training data:*

$$S_\epsilon(h_0) := \{h \in \mathcal{H} : \hat{R}(h) \leq \hat{R}(h_0) + \epsilon\}.$$

[1] Deploying a model that optimizes performance on the training data reflects standard best practice in machine learning – even in settings where we would use a validation dataset (see e.g., Cawley & Talbot, 2010, for a discussion). For example, in a setting where we would tune hyperparameters to control overfitting, we would first find a set of hyperparameters that optimizes an estimate of out-of-sample error (e.g., mean 5-CV error). We would then fit a model that optimizes performance for this set of hyperparameters using all of the training data.

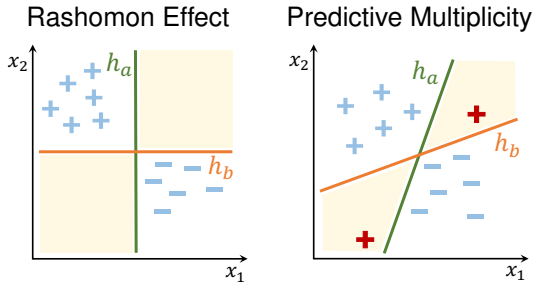

Figure 2. Predictive multiplicity reflects irreconcilable differences between the predictions of competing models. Here, we depict two classification problems where the competing classifiers h_a and h_b optimize accuracy. We highlight points that are assigned conflicted predictions in red and regions of conflict in yellow. On the left, h_a and h_b assign the same predictions on the training data but produce conflicting explanations of the importance of x_1 vs. x_2, as per the Rashomon effect. On the right, h_a and h_b assign conflicting predictions on the training data as per predictive multiplicity.

Predictive multiplicity can arise over an ϵ-level set where $\epsilon = 0$ (see e.g. Figure 1). We measure predictive multiplicity over an ϵ-level set where $\epsilon > 0$. This is because a competing model that attains near-optimal performance on the training data may outperform the optimal model in deployment. In such cases, it would not be defensible to rule out competing models on the basis of small differences in training error. In practice, the value of ϵ should be set so that the ϵ-level set is likely to include a model that attains optimal performance in deployment. This can be achieved by computing confidence intervals for out-of-sample performance (e.g., via bootstrapping or cross-validation) or by using generalization bounds (e.g., by setting ϵ so that with high probability the ϵ-level set contains the model that optimizes true risk).

2.1 Predictive Multiplicity

A prediction problem exhibits *predictive multiplicity* if competing models assign conflicting predictions over the training data.

Definition 2 (Predictive Multiplicity) *Given a baseline classifier h_0 and an error tolerance ϵ, a prediction problem exhibits predictive multiplicity over the ϵ-level set $S_\epsilon(h_0)$ if there exists a model $h \in S_\epsilon(h_0)$ such that $h(\boldsymbol{x}_i) \neq h_0(\boldsymbol{x}_i)$ for some $\boldsymbol{x}_i$ in the training dataset.*

The fact that competing models assign conflicting predictions means that model selection will involve arbitrating irreconcilable predictions.

Measures of Predictive Multiplicity. We measure the severity of predictive multiplicity by counting the number of examples that are assigned conflicting predictions by com-

peting models in the ϵ-level set. In what follows, we present formal definitions of these measures and then discuss their implications.

Definition 3 (Ambiguity) *The ambiguity of a prediction problem over the ϵ-level set $S_\epsilon(h_0)$ is the proportion of points in a training dataset that can be assigned a conflicting prediction by a competing classifier $h \in S_\epsilon(h_0)$:*

$$\alpha_\epsilon(h_0) := \frac{1}{n}\sum_{i=1}^{n} \max_{h \in S_\epsilon(h_0)} \mathbb{1}[h(\boldsymbol{x}_i) \neq h_0(\boldsymbol{x}_i)].$$

Definition 4 (Discrepancy) *The discrepancy of a prediction problem over the ϵ-level set $S_\epsilon(h_0)$ is the maximum proportion of conflicting predictions between the baseline classifier h_0 and a competing classifier $h \in S_\epsilon(h_0)$:*

$$\delta_\epsilon(h_0) := \max_{h \in S_\epsilon(h_0)} \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}[h(\boldsymbol{x}_i) \neq h_0(\boldsymbol{x}_i)].$$

Ambiguity represents the number of predictions that can change over the set of competing models. This reflects the number of individuals whose predictions are determined by model choice, and who should have a say in model selection. Importantly, ambiguity also determines the number of individuals who could contest the prediction assigned to them by the deployed model.

Discrepancy represents the maximum number of predictions that can change if we switch the baseline classifier with a competing classifier. This reflects that in practice, in order to change multiple predictions, the conflicting predictions must all be realized by a single competing model.

We end with a discussion of the relationship between accuracy and predictive multiplicity. In Proposition 1, we bound the number of conflicts between the optimal model and any model in the ϵ-level set. We include a proof in Appendix A.

Proposition 1 (Bound on Discrepancy) *The discrepancy between h_0 and any competing classifier in the ϵ-level set $h \in S_\epsilon(h_0)$ obeys:*

$$\delta_\epsilon \leq 2\hat{R}(h_0) + \epsilon.$$

Proposition 1 demonstrates how the severity of predictive multiplicity depends on the accuracy of a baseline model. As accuracy of the baseline model, there is more "room" for predictive multiplicity. This result motivates why it is important to measure discrepancy and ambiguity with respective to the best possible baseline model.

3 Methodology

In this section, we present integer programming tools to compute ambiguity and discrepancy for linear classification problems.

3.1 Preliminaries

We compute ambiguity and discrepancy by fitting linear classifiers from the ϵ-level set. Our methods can compute these measures given any baseline classifier – so long as it is linear. In our experiments, we compute these measures with respect to a baseline classifier h_0 that maximizes accuracy by solving the MIP in Appendix B. This ensures that multiplicity does not arise due to suboptimality. Thus, the only way to avoid multiplicity is to change the prediction problem (e.g., by changing the dataset, incorporating tie-breaking constraints, or choosing a different hypothesis class).

Path Algorithms. We present *path algorithms* to compute ambiguity and discrepancy over all possible ϵ-level sets, and show how these measures change with respect to ϵ (see e.g., Figure 3). Using path algorithms relaxes the need for practitioners to choose ϵ a priori, and calibrates the choice of ϵ in settings where small changes in ϵ can result in large changes in ambiguity and discrepancy.

Integer Programming. We fit each classifier by solving a discrete *empirical risk minimization* (ERM) problem. We express each problem as a *mixed integer program* (MIP) that we pass to a MIP solver such as CPLEX (ILOG, 2019). MIP solvers find the global optimum of a discrete optimization problem through exhaustive search processes (e.g., branch-and-bound Wolsey, 1998). In our setting, solving a MIP returns: (i) an upper bound on the objective value; (ii) a lower bound on the objective value; and (iii) the coefficients of a linear classifier that achieves the upper bound. When the upper bound (i) matches the lower bound (ii), the solution in (iii) is *certifiably optimal*, and our measures are exact. In settings where the MIP solver cannot find a certifiably optimal solution in a reasonable timeframe, the upper and lower bounds from (i) and (ii) correspond to bounds on ambiguity and discrepancy.

3.2 Computing Discrepancy

Given a training dataset, a baseline classifier h_0, and a user-specified error tolerance ϵ, we compute the discrepancy over the ϵ-level set around h_0 by solving the following optimization problem.

$$\begin{aligned} \min_{h \in \mathcal{H}} \quad & \sum_{i=1}^{n} \mathbb{1}[h(\boldsymbol{x}_i) = h_0(\boldsymbol{x}_i)] \\ \text{s.t.} \quad & \hat{R}(h) \leq \hat{R}(h_0) + \epsilon \end{aligned} \tag{1}$$

We denote the optimal solution to Equation (1) as g_ϵ. For linear classification problems, we can recover the coefficients of g_ϵ by solving the following MIP formulation, which we refer to as $\mathsf{DiscMIP}(h_0, \epsilon)$.

$$
\begin{aligned}
\min \quad & \sum_{i=0}^{n} a_i \\
\text{s.t.} \quad & M_i a_i \geq \gamma + h_0(\boldsymbol{x}_i) \sum_{j=0}^{d} w_j x_{ij} && i = 1, ..., n && \text{(2a)} \\
& \epsilon \geq \frac{1}{n} \sum_{i=1}^{n} y_i h_0(\boldsymbol{x}_i)(1 - a_i) && && \text{(2b)} \\
& w_j = w_j^+ + w_j^- && j = 0, ..., d && \text{(2c)} \\
& 1 = \sum_{j=0}^{d} (w_j^+ - w_j^-) && && \text{(2d)} \\
& a_i \in \{0, 1\} && i = 1, ..., n \\
& w_j^+ \in [0, 1] && j = 0, ..., d \\
& w_j^- \in [-1, 0] && j = 0, ..., d
\end{aligned}
$$

DiscMIP minimizes the agreement between h and h_0 using indicator variables $a_i = \mathbb{1}[h(\boldsymbol{x}_i) = h_0(\boldsymbol{x}_i)]$. These variables are set via the "Big-M" constraints in (2a). These constraints depend on: (i) a margin parameter $\gamma > 0$, which should be set to a small positive number (e.g., $\gamma = 10^{-4}$); and (ii) the Big-M parameters M_i, which can be set as $M_i = \gamma + \max_i \|\boldsymbol{x}_i\|_\infty$ since we have fixed $\|\boldsymbol{w}\|_1 = 1$ in constraint (2d). Constraint (2b) ensures that any feasible classifier must belong to the ϵ-level set.

Bounds. Solving DiscMIP returns the coefficients of the classifier that maximizes discrepancy with respect to the baseline classifier h_0. If the solution is not certifiably optimal, the upper bound from DiscMIP corresponds to a lower bound on discrepancy. Likewise, the lower bound from DiscMIP correspond to an upper bound on discrepancy.

Path Algorithm. In Algorithm 1, we present a procedure to compute discrepancy for all possible values of ϵ. The procedure solves DiscMIP(h_0, ϵ) for increasing values of $\epsilon \in \mathcal{E}$. At each iteration, the procedure initializes DiscMIP(h_0, ϵ) using the solution from the previous iteration. The process is many times faster than solving DiscMIP(h_0, ϵ) separately for various error tolerances, because the solution from the previous iteration produces upper and lower bounds that reduce the search space of the MIP at the current iteration.

Algorithm 1 Compute Discrepancy for All Values of ϵ

Input
- h_0 — baseline classifier
- $\mathcal{E}$ — values of ϵ for which to compute discrepancy

Initialize

1: **for** $\epsilon \in \mathcal{E}$ **do**
2: Initialize DiscMIP(h_0, ϵ) using previous solution
3: $g_\epsilon \leftarrow$ solution to DiscMIP(h_0, ϵ)
4: $\delta_\epsilon \leftarrow$ number of conflicts between g_ϵ and h_0
5: **end for**

Output: $\{\delta_\epsilon, g_\epsilon\}_{\epsilon \in \mathcal{E}}$ — discrepancy and classifier for each ϵ

3.3 Computing Ambiguity

We will present an algorithm to compute ambiguity for all possible values of ϵ. Our approach will fit a *pathological classifier* g_i for each point i in the training dataset – i.e., the most accurate linear classifier that assigns a conflicting prediction to point i. Given a pathological classifier for each point in the dataset $\{g_i\}_{i=1}^n$, we can compute ambiguity over the ϵ-level set by counting the number of pathological classifiers whose error rate is within ϵ of the baseline model. This corresponds to evaluating the inner terms in our expression for ambiguity (see Definition 3): $\max_{h \in S_\epsilon(h_0)} \mathbb{1}[h(\boldsymbol{x}_i) \neq h_0(\boldsymbol{x}_i)]$.

We fit g_i by solving the following optimization problem:

$$
\begin{aligned}
\min_{h \in \mathcal{H}} \quad & \sum_{i=1}^{n} \mathbb{1}[h(\boldsymbol{x}_i) \neq y_i] \\
\text{s.t.} \quad & h(\boldsymbol{x}_i) \neq h_0(\boldsymbol{x}_i)
\end{aligned}
\qquad (3)
$$

Here, $h(\boldsymbol{x}_i) \neq h_0(\boldsymbol{x}_i)$ forces any feasible classifier h to assign a conflicting prediction to point $\boldsymbol{x}_i$. For linear classification problems, we can recover the coefficients of g_i by solving the following MIP formulation, which we refer to as FlipMIP$(h_0, \boldsymbol{x}_i)$:

$$
\begin{aligned}
\min \quad & \sum_{i=0}^{n} l_i \\
\text{s.t.} \quad & M_i l_i \geq y_i(\gamma - \sum_{j=0}^{d} w_j x_{ij}) && i = 1, ..., n && \text{(4a)} \\
& \gamma \leq -h_0(\boldsymbol{x}_i) \sum_{j=0}^{d} w_j x_j && && \text{(4b)} \\
& w_j = w_j^+ + w_j^- && j = 0, ..., d && \text{(4c)} \\
& 1 = \sum_{j=0}^{d} (w_j^+ - w_j^-) && && \text{(4d)} \\
& l_i \in \{0, 1\} && i = 1, ..., n \\
& w_j^+ \in [0, 1] && j = 0, ..., d \\
& w_j^- \in [-1, 0] && j = 0, ..., d
\end{aligned}
$$

Here, constraints (4a) set the mistake indicators $l_i \leftarrow \mathbb{1}[h(\boldsymbol{x}_i) \neq y_i]$. The parameters in these constraints can be set in the same way as the parameters in the Big-M constraints for DiscMIP. Constraint (4b) enforces the condition that $g_i(\boldsymbol{x}) \neq h(\boldsymbol{x})$.

Bounds. If the solution to FlipMIP is not certifiably optimal, then the upper and lower bounds from FlipMIP can be used to bound ambiguity. The upper bounds from FlipMIP will produce lower bounds on ambiguity. The lower bounds from FlipMIP will produce upper bounds on ambiguity.

Path Algorithm. In Algorithm 2, we outline a procedure to efficiently compute ambiguity by initializing each instance of FlipMIP. In line 2, the procedure sets the upper

bound for FlipMIP using the most accurate classifier in POOL that obeys the constraint $h(\boldsymbol{x}_i) \neq h_0(\boldsymbol{x}_i)$. Given a certifiably optimal baseline classifier, we can initialize the lower bound of FlipMIP to $n\hat{R}(h_0)$.

Algorithm 2 Compute Ambiguity for All Values of ϵ

Input
- h_0 — baseline classifier
- $\mathcal{E}$ — values of ϵ for which to compute ambiguity

Initialize
- POOL $\leftarrow \emptyset$ — pool of pathological classifiers

for $i \in \{1, 2, \ldots, n\}$ **do**
  Initialize FlipMIP$(h_0, \boldsymbol{x}_i)$ using best solution in POOL
  $g_i \leftarrow$ solution to FlipMIP$(h_0, \boldsymbol{x}_i)$
  Add g_i to POOL
end for
for $\epsilon \in \mathcal{E}$ **do**
  $\alpha_\epsilon \leftarrow \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}[\hat{R}(g_i) \leq \hat{R}(h_0)]$
end for

Output: $\{\alpha_\epsilon\}_{\epsilon \in \mathcal{E}}$ and $\{g_i\}_{i=1}^{n}$

3.4 Discussion

Our MIP formulations can easily be changed to optimize other objectives (e.g., measures of class-based accuracy) or to obey constraints on model form or model predictions (e.g., constraints on group fairness).

Our use of integer programming is motivated by the fact that popular approaches to reduce computation (e.g., solving a ERM problem with a convex surrogate loss) may return unreliable estimates of predictive multiplicity because they cannot search over a set of competing models defined in terms of a non-convex objective such as the error rate. In contrast, integer programming can search over *all* possible models in an ϵ-level set.

4 Experiments

In this section, we apply our tools to measure predictive multiplicity in recidivism prediction problems. We have three goals: (i) to measure the incidence of predictive multiplicity in real-world classification problems; (ii) to discuss how reporting predictive multiplicity can inform stakeholders; (iii) to show that we can also measure predictive multiplicity using existing tools, albeit imperfectly. We include the software to reproduce our results at https://github.com/charliemarx/pmtools.

4.1 Setup

Datasets. We examine predictive multiplicity for 8 prediction problems that we derive from the following studies of recidivism in the United States:

- `compas` from Angwin et al. 2016;
- `pretrial` from Felony Defendants in Large Urban Counties (US Dept. of Justice, 2014b);
- `recidivism` from Recidivism of Prisoners Released in 1994 (US Dept. of Justice, 2014a).

Dataset	Outcome Variable	n	d	Error of h_0 Train	Error of h_0 Test
`compas_arrest`	rearrest for any crime	5,380	18	32.7%	33.4%
`compas_violent`	rearrest for violent crime	8,768	18	37.7%	37.9%
`pretrial_CA_arrest`	rearrest for any crime	9,926	22	34.1%	34.4%
`pretrial_CA_fta`	failure to appear	8,738	22	36.3%	36.3%
`recidivism_CA_arrest`	rearrest for any offense	114,522	20	34.4%	34.2%
`recidivism_CA_drug`	rearrest for drug-related offense	96,664	20	36.3%	36.2%
`recidivism_NY_arrest`	rearrest for any offense	31,624	20	31.0%	31.8%
`recidivism_NY_drug`	rearrest for drug-related offense	27,526	20	32.5%	33.6%

Table 1. Overview of recidivism prediction datasets. For each dataset, we fit a baseline linear classifier that minimizes training error. As shown, the models generalize as training error is close to test error. This is expected given that we fit models from a simple hypothesis class. Here, n and d denote the number of examples and features in each dataset, respectively. All datasets are publicly available. We include a copy of `compas_violent` and `compas_arrest` with our code. The remaining datasets must be requested from ICPSR due to privacy restrictions.

We process each dataset by dropping instances with missing entries, binarizing features, and oversampling the minority class to equalize the number of instances with positive and negative labels. [2] We present an overview of each dataset in Table 1, and provide additional details in Appendix C.2.

Measurement Protocol. We compute our measures of predictive multiplicity for each dataset as follows. We split each dataset into a *training set* composed of 80% of points and a *test set* composed of 20% of points. We use the training set to fit a *baseline classifier* that minimizes the 0-1 loss directly by solving MIP (6) in Appendix B. We measure ambiguity and discrepancy for *all possible values* of the error tolerance ϵ using Algorithms 1 and 2. We solve each MIP on a 3.33 GHz CPU with 16 GB RAM. We allocate at most 6 hours to fit the baseline model, 6 hours to fit the models to compute discrepancy for all ϵ, and 6 hours to fit the models to compute ambiguity for all ϵ.

Ad Hoc Measurement. We compute ambiguity and discrepancy through an ad hoc approach. We include these results to show that an imperfect analysis of predictive multiplicity can reveal salient information. Here, we produce a pool of competing models using the glmnet package of Friedman et al. (2010). We fit 1,100 linear classifiers using penalized logistic regression. Each model corresponds to an

[2] Oversampling ensures that we can directly minimize the training error without needing to worry about producing trivial classifiers, and allows us to report the error rate as opposed to class-specific error rates. We find that oversampling has a trivial effect on our measures of multiplicity ($< 1\%$).

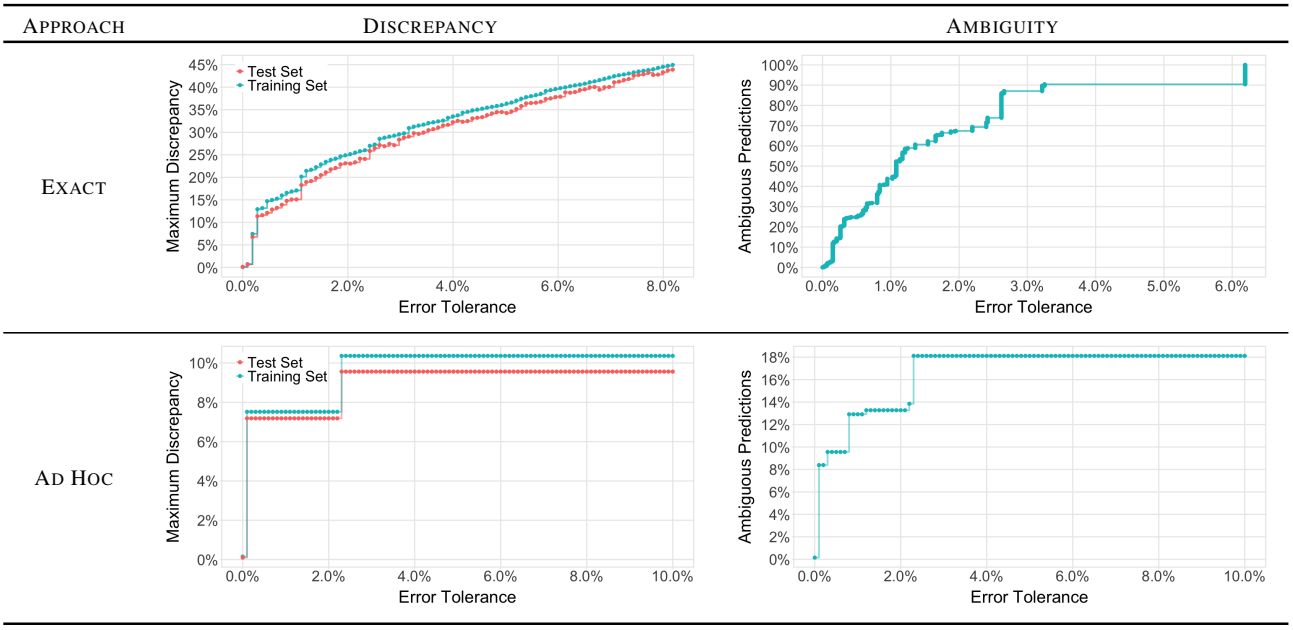

Figure 3. Severity of predictive multiplicity measured using our tools (top) and using an ad hoc approach (bottom) for `compas_arrest`. We plot the values of discrepancy (left) and ambiguity (right) over the ϵ-level set. Here, we find a discrepancy of 17% and ambiguity of 44% over the 1%-level set. This means that one can change 17% of predictions by switching the baseline model with a model that is only 1% less accurate, and that 44% of individuals are assigned conflicting predictions by models in the 1%-level set.

optimizer of the logistic loss with a different degree of ℓ_1 and ℓ_2 regularization. We choose the baseline model as the model that minimizes the 5-fold CV test error.

4.2 Results

In Figure 3, we present plots for the ambiguity and discrepancy for all possible values of the error tolerance ϵ. We compare the exact measures produced using our tools to the measures computed using an ad hoc analysis for `compas_arrest`. We include these plots for other datasets in Appendix C.2. In Table 2, compare competing classifiers for `compas_arrest`. In what follows, we discuss these results. Unless otherwise specified, we discuss the values of ambiguity and discrepancy for an error tolerance of $\epsilon = 1\%$.

On the Incidence of Predictive Multiplicity. Our results in Figure 3 show how predictive multiplicity arises in real-world prediction problems. For the 8 datasets we consider, we find that between 4% and 53% of individuals are assigned conflicting predictions in the 1%-level set. In `compas_arrest`, for example, we observe an ambiguity of 44%. In `compas_violent`, ambiguity over the 1%-level set is 53%, which means that the majority of predictions are affected by which competing model we choose. Considering discrepancy, we see that for `compas_arrest` and `compas_violent` we can find *a single competing model* in the 1%-level set that would assign a conflicting prediction to 17% of individuals.

On the Burden of Multiplicity. Our results show that the incidence of multiplicity can differ significantly between protected groups. In `compas_violent`, for example, the proportion of individuals who are assigned conflicting predictions over the 1% level set is 37.2% for Caucasians but 72.9% of African-Americans. In this case, predictive multiplicity disproportionately affects African-Americans compared to individuals of other ethnic groups. Groups with a larger burden of multiplicity are more vulnerable to model selection, and more likely to be affected by the ignorance of competing models.

On the Implications of Predictive Multiplicity. Our results illustrate how reporting ambiguity and discrepancy can challenge model development and deployment. In `compas_arrest`, for example, our baseline model provably optimizes training error and generalizes. In the absence of an analysis of predictive multiplicity, many practitioners would deploy this model. Our analysis reveals that there exists a competing model that assigns conflicting predictions to 17% of individuals. Thus, these measures support the need for greater scrutiny and support stakeholder involvement in model selection.

Large values of ambiguity and discrepancy also lead us to calibrate trust in downstream processes (e.g., evaluating the impact of features Marx et al., 2019). Consider the process of explaining individual predictions. In this case, an ambiguity of 44% of points means one could produce

	Baseline Model	Individual Ambiguity Model	Discrepancy Model
$h(\boldsymbol{x}_p)$	$+1$	-1	-1
Error (Train/Test)	32.7% / 33.4%	32.7% / 33.4%	33.6 / 34.5%
Discrepancy (Train/Test)	0.0% / 0.0%	0.0037% / 0.0%	16.8% / 15.1%
Score Function	+ 0.5 *age≤25* + 0.0 *age_25-45* − 16.4 *age≥46* − 16.3 *female* − 0.2 *n_priors=0* − 0.1 *n_priors≥1* + 16.4 *n_priors≥2* + 16.6 *n_priors≥5* + 0.0 *n_juvenile_misdemeanors=0* − 0.1 *n_juvenile_misdemeanors≥1* + 0.0 *n_juvenile_misdemeanors≥2* − 32.6 *n_juvenile_misdemeanors≥5* + 0.0 *n_juvenile_felonies=0* − 0.2 *n_juvenile_felonies≥1* + 0.3 *n_juvenile_felonies≥2* + 0.0 *n_juvenile_felonies≥5* − 0.2 *charge_degree=M* + 0.0	+ 10.3 *age≤25* + 0.0 *age_25-45* − 9.9 *age≥46* − 9.7 *female* + 0.0 *n_priors=0* + 0.0 *n_priors≥1* + 19.8 *n_priors≥2* + 10.1 *n_priors≥5* + 0.0 *n_juvenile_misdemeanors=0* − 0.1 *n_juvenile_misdemeanors≥1* − 10.1 *n_juvenile_misdemeanors≥2* − 9.5 *n_juvenile_misdemeanors≥5* − 9.9 *n_juvenile_felonies=0* − 10.1 *n_juvenile_felonies≥1* + 0.3 *n_juvenile_felonies≥2* + 0.0 *n_juvenile_felonies≥5* − 0.2 *charge_degree=M* + 0.0	+ 7.7 *age≤25* + 0.0 *age_25-45* − 7.8 *age≥46* − 7.6 *female* − 7.8 *n_priors=0* + 0.0 *n_priors≥1* + 7.4 *n_priors≥2* + 7.8 *n_priors≥5* + 0.0 *n_juvenile_misdemeanors=0* + 0.1 *n_juvenile_misdemeanors≥1* − 0.1 *n_juvenile_misdemeanors≥2* − 15.2 *n_juvenile_misdemeanors≥5* + 7.7 *n_juvenile_felonies=0* + 0.0 *n_juvenile_felonies≥1* + 15.4 *n_juvenile_felonies≥2* + 0.0 *n_juvenile_felonies≥5* − 7.5 *charge_degree=M* − 0.1

Table 2. Competing linear classifiers that assign conflicting prediction to $\boldsymbol{x}_p$ `compas_arrest`. We show the baseline model (left), the competing model fit to measure ambiguity to $\boldsymbol{x}_p$ (middle), and competing model fit to measure discrepancy (right). The baseline model predicts $h(\boldsymbol{x}_p) = +1$ while other models predict $h(\boldsymbol{x}_p) = -1$. As shown, there exists at least two competing models that predict that $\boldsymbol{x}_p$ would not recidivate. In addition, each model exhibits different coefficients and measures of variable importance.

conflicting explanations for individual predictions. While all explanations would reflect how competing models operate, evidence that 44% of explanations would support conflicting predictions would provide a safeguard against unwarranted rationalization.

On Model Selection. When presented with a large set of competing models, a natural solution is to choose among them to optimize secondary objectives. We support this practice in settings where secondary objectives relate to desirable real-world goals (see Section 5 for a discussion). However, tie-breaking does not always yield a unique model. For the `compas_arrest` dataset, for example, we can filter the models in the ϵ-level set to optimize a group fairness criterion (i.e., to minimize the disparity in accuracy between African-Americans and other ethic groups). In this case, we find 102 competing models that are also within 1% optimal in terms of the secondary criterion.

On Ad Hoc Measurement. Our results for the ad hoc approach show how measuring and reporting predictive multiplicity can reveal useful information even without specialized tools. In `compas_arrest`, for example, an ad hoc analysis reveals an ambiguity of 10% and a discrepancy of 7% over the set of competing models. These estimates are far less than those produced using our tools (44% and 17% respectively). This is because the ad hoc approach only considers competing models that can be obtained by varying ℓ_1 and ℓ_2 penalties in penalized logistic regression, rather than all linear classifiers in the 1%-level set. These results show that ad hoc approaches can detect predictive multiplicity, but should not be used to certify the absence of multiplicity.

5 Concluding Remarks

Prediction problems can exhibit predictive multiplicity due to a host of reasons, including feature selection, a misspecified hypothesis class, or the existence of latent groups.

Even as there exist techniques to choose between competing models, we do not advocate a general prescription to resolve predictive multiplicity. Instead, we argue that we should measure and report multiplicity in the same way that we measure and report test error (Saleiro et al., 2018; Reisman et al., 2018). In this way, predictive multiplicity can be resolved on a case-by-case basis, and in a way that allows for input from stakeholders (as per the principles of contestable design; see e.g., Hirsch et al., 2017; Kluttz et al., 2018).

Reporting predictive multiplicity can change how we build and deploy models in human-facing applications. In such settings, presenting stakeholders with meaningful information about predictive multiplicity may lead them to think carefully about which model to deploy, consider assigning favorable predictions to individuals who receive conflicting predictions, or forgo deployment entirely.

Acknowledgements

We thank Sorelle Friedler, Dylan Slack, and Ben Green for helpful discussions, and three anonymous reviewers for constructive feedback.

References

Aïvodji, U., Arai, H., Fortineau, O., Gambs, S., Hara, S., and Tapp, A. Fairwashing: the risk of rationalization. *arXiv preprint arXiv:1901.09749*, 2019.

Angwin, J., Larson, J., Mattu, S., and Kirchner, L. Machine bias. *ProPublica, May*, 23:2016, 2016.

Binns, R., Van Kleek, M., Veale, M., Lyngs, U., Zhao, J., and Shadbolt, N. 'it's reducing a human being to a percentage': Perceptions of justice in algorithmic decisions. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 377. ACM, 2018.

Breiman, L. Statistical modeling: The two cultures. *Statistical science*, 16(3):199–231, 2001.

Cawley, G. C. and Talbot, N. L. On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11 (Jul):2079–2107, 2010.

Celis, L. E., Huang, L., Keswani, V., and Vishnoi, N. K. Classification with fairness constraints: A meta-algorithm with provable guarantees. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 319–328. ACM, 2019.

Chouldechova, A. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163, 2017.

Corbett-Davies, S., Pierson, E., Feller, A., Goel, S., and Huq, A. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 797–806. ACM, 2017.

Cotter, A., Jiang, H., Wang, S., Narayan, T., You, S., Sridharan, K., and Gupta, M. R. Optimization with non-differentiable constraints with applications to fairness, recall, churn, and other goals. 2019.

Ding, J., Tarokh, V., and Yang, Y. Model selection techniques: An overview. *IEEE Signal Processing Magazine*, 35(6):16–34, 2018.

Dong, J. and Rudin, C. Variable importance clouds: A way to explore variable importance for the set of good models. *arXiv preprint arXiv:1901.03209*, 2019.

Dusenberry, M. W., Tran, D., Choi, E., Kemp, J., Nixon, J., Jerfel, G., Heller, K., and Dai, A. M. Analyzing the role of model uncertainty for electronic health records. In *Proceedings of the ACM Conference on Health, Inference, and Learning*, pp. 204–213, 2020.

Fisher, A., Rudin, C., and Dominici, F. All models are wrong but many are useful: Variable importance for black-box, proprietary, or misspecified prediction models, using model class reliance. *arXiv preprint arXiv:1801.01489*, 2018.

Friedman, J., Hastie, T., and Tibshirani, R. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010.

Germain, P., Bach, F., Lacoste, A., and Lacoste-Julien, S. Pac-bayesian theory meets bayesian inference. In *Advances in Neural Information Processing Systems*, pp. 1884–1892, 2016.

Hirsch, T., Merced, K., Narayanan, S., Imel, Z. E., and Atkins, D. C. Designing contestability: Interaction design, machine learning, and mental health. In *Proceedings of the 2017 Conference on Designing Interactive Systems*, pp. 95–99. ACM, 2017.

ILOG, I. Cplex optimizer 12.8. https://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/, 2019.

Kleinberg, J., Ludwig, J., Mullainathan, S., and Obermeyer, Z. Prediction policy problems. *American Economic Review*, 105(5):491–95, 2015.

Kleinberg, J., Mullainathan, S., and Raghavan, M. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*, 2016.

Kluttz, D., Kohli, N., and Mulligan, D. K. Contestability and professionals: From explanations to engagement with algorithmic systems. *Available at SSRN 3311894*, 2018.

Koehler, D. J. Explanation, imagination, and confidence in judgment. *Psychological bulletin*, 110(3):499, 1991.

Laugel, T., Lesot, M.-J., Marsala, C., Renard, X., and Detyniecki, M. The dangers of post-hoc interpretability: Unjustified counterfactual explanations. *arXiv preprint arXiv:1907.09294*, 2019.

Martens, D. and Provost, F. Explaining data-driven document classifications. *MIS Quarterly*, 38(1):73–100, 2014.

Marx, C., Phillips, R., Friedler, S., Scheidegger, C., and Venkatasubramanian, S. Disentangling influence: Using disentangled representations to audit model predictions. In *Advances in Neural Information Processing Systems*, pp. 4496–4506, 2019.

McAllester, D. A. Some pac-bayesian theorems. *Machine Learning*, 37(3):355–363, 1999.

McAllister, J. W. Model selection and the multiplicity of patterns in empirical data. *Philosophy of Science*, 74(5): 884–894, 2007.

McCullagh, P. and Nelder, J. A. *Generalized Linear Models*, volume 37. CRC Press, 1989.

Mountain, D. and Hsiao, C. A combined structural and flexible functional approach for modeling energy substitution. *Journal of the American Statistical Association*, 84(405): 76–87, 1989.

Passi, S. and Barocas, S. Problem formulation and fairness. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 39–48. ACM, 2019.

Pennsylvania Bulletin. Sentence Risk Assessment Instrument, April 2017.

Reisman, D., Schultz, J., Crawford, K., and Whittaker, M. Algorithmic impact assessments: A practical framework for public agency accountability. *AI Now Institute*, 2018.

Saleiro, P., Kuester, B., Stevens, A., Anisfeld, A., Hinkson, L., London, J., and Ghani, R. Aequitas: A bias and fairness audit toolkit. *arXiv preprint arXiv:1811.05577*, 2018.

Semenova, L. and Rudin, C. A study in rashomon curves and volumes: A new perspective on generalization and model simplicity in machine learning. *arXiv preprint arXiv:1908.01755*, 2019.

Shmueli, G. et al. To explain or to predict? *Statistical science*, 25(3):289–310, 2010.

Slack, D., Hilgard, S., Jia, E., Singh, S., and Lakkaraju, H. How can we fool lime and shap? adversarial attacks on post hoc explanation methods. *arXiv preprint arXiv:1911.02508*, 2019.

Tulabandhula, T. and Rudin, C. Machine learning with operational costs. *The Journal of Machine Learning Research*, 14(1):1989–2028, 2013.

US Dept. of Justice, B. o. J. S. Recidivism of prisoners released in 1994. 2014a. doi: 10.3886/ICPSR03355.v8.

US Dept. of Justice, B. o. J. S. State court processing statistics, 1990-2009: Felony defendants in large urban counties. 2014b. doi: 10.3886/ICPSR02038.v5.

Wolsey, L. A. *Integer Programming*, volume 42. Wiley New York, 1998.

Zafar, M. B., Valera, I., Gomez-Rodriguez, M., and Gummadi, K. P. Fairness constraints: A flexible approach for fair classification. *Journal of Machine Learning Research*, 20(75):1–42, 2019.