The Right Tool for The Job? Assessing the Use of Artificial Intelligence for Identifying Administrative Errors

Matthew M. Young
The Maxwell School, Syracuse University
Syracuse, New York, USA
myoung10@syr.edu

Johannes Himmelreich
The Maxwell School, Syracuse University
Syracuse, New York, USA
jrhimmel@syr.edu

Danylo Honcharov
Engineering and Computer Science, Syracuse University
Syracuse, New York, USA
dhonchar@syr.edu

Sucheta Soundarajan
Engineering and Computer Science, Syracuse University
Syracuse, New York, USA
susounda@syr.edu

ABSTRACT

This article explores the extent to which machine learning can be used to detect administrative errors. It concentrates on administrative errors in unemployment insurance (UI) decisions, which give rise to a public values conflict between efficiency and effectiveness. This conflict is first described and then highlighted in the history of the US UI regime. Machine learning may not only mitigate this conflict but it may also help to combat fraud and reduce the backlog of claims associated with economic crises such as the COVID-19 pandemic. The article uses data about improper UI payments throughout the US from 2002 through 2018 to analyze the accuracy of random forests and deep learning models. We find that a random forest model using gradient descent boosting is more accurate, along several measures, than every deep learning model tested. This finding could be explained by the goodness-of-fit between the machine learning method and the available data. Alternatively, deep learning performance could be attenuated by necessary limits to publicly-accessible claims data.

CCS CONCEPTS

• **Applied computing** → **Computing in government**; *IT governance*; • **Computing methodologies** → Supervised learning; Classification and regression trees.

KEYWORDS

AI, Public Administration, Administrative Errors, Unemployment Insurance, Social Policy

ACM Reference Format:
Matthew M. Young, Johannes Himmelreich, Danylo Honcharov, and Sucheta Soundarajan. 2021. The Right Tool for The Job? Assessing the Use of Artificial Intelligence for Identifying Administrative Errors. In *DG.O2021: The 22nd Annual International Conference on Digital Government Research (DG.O'21), June 9–11, 2021, Omaha, NE, USA.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3463677.3463714

DG.O'21, June 9–11, 2021, Omaha, NE, USA

ACM ISBN 978-1-4503-8492-6/21/06…$15.00
https://doi.org/10.1145/3463677.3463714

1 INTRODUCTION

Nobody is perfect; mistakes happen. This paper analyzes whether Artificial Intelligence (AI) can be used to detect mistakes. It focuses on mistakes that State workforce agencies make when they review and process Unemployment Insurance (UI) claims. In deciding claims, mistakes consist of over- or under-paying claimants. Where a claim is denied that should be approved, the mistake is an underpayment of the full dollar amount claimed. Such mistakes are administrative errors, that is, mistakes in organizational decision-making about the allocation of benefits. These administrative errors are documented in data collected by the US Department of Labor (DOL). This paper uses these data to train different Machine Learning (ML) algorithms to predict administrative errors in this setting.

Systemic administrative errors in programs like UI have long been recognized as important [29]. These errors have already been identified as a problem that could be solved - or exacerbated - with technology [8, 13, 19]. Moreover, general theoretical frameworks to guide the implementation AI exist [48]. Some ML classifiers have been used in other studies to detect fraud in Medicare payments [4, 9, 22, 34]. Yet, which concrete AI-technologies can and should be used to address administrative errors has not been answered in full.

This paper is motivated by the question of whether AI *can* and *should* be used to detect administrative errors. It takes the first steps towards answering this research question and improving AI-driven decision-making in government by examining prominent ML-technologies for a specific task. AI plays an increasingly prominent role in the operation of the public sector. AI is used in criminal justice, public health, child-welfare, education, policing, and regulatory enforcement [9, 18]. In each case, AI has the potential to further a public value: *efficiency*. But greater efficiency, such as reducing costs, may come at the expense of at least one other important public value: *effectiveness*, that is, making sure that those who are eligible for services receive them. Administrative errors hence have a public values conflict at their center. We argue that reducing administrative errors helps to overcome this conflict. Both public values can be furthered at the same time.

[1]The authors wish to thank Natalie Gallagher for her research assistance

This is particularly important today. UI fraud has long been a topic of attention and concern. In the US 296,749 cases of fraud were identified in 2019, amounting to $366.8 million [41]. The number of fraudulent claims has increased significantly during the COVID-19 pandemic [40]. At the same time, the pandemic has brought attention to non–fraudulent over- and underpayments of UI benefits and to insufficient timeliness with which UI claims are decided [17].

This paper contributes to the literature on administrative errors and improper payments. It deploys AI-based techniques for identifying improper payments. It does so by using labeled audit data that are likely to be employed in training such systems in practice. The paper also compares the performance of different algorithms and discusses their goodness of fit relative to these data. It begins by sketching a public values conflict of UI (Section 2), describes its historical background in the US and the origin of the data (Sections 3 and 4), and then reports results and discusses studies of different families of ML-algorithms for predicting administrative errors in benefit claims (Sections 5 and 6).

Briefly stated, we find that a random forest classifier using gradient descent boosting (CatBoost) is superior to several different deep learning-based classifiers both for accuracy and explainability. These advantages are to some degree due to the underlying data generative processes, as well as specific features of publicly-accessible US unemployment insurance data.

2 A PUBLIC VALUES CONFLICT

Two public values that UI aims to further are efficiency and efficacy.[2] These values themselves should not be controversial [27, 38]. They motivated the creation of the UI system, inform its legal background, shape how UI is administered, and are reflected in public expectations towards UI. The two values can be defined and analyzed into constituting dimensions as follows.

(1) **Effectiveness**: provide insurance payments to those who are eligible in a convenient and timely manner.
   (a) **Opportunity:** enable individuals who are likely eligible to apply, e.g. offer an application process that is convenient for eligible claimants.
   (b) **Payment:** render goods/services for eligible claims to claimants quickly.
   (c) **Avoid underpayment:** minimize under-payment, i.e. reduce false negative eligibility.
(2) **Efficiency**: reducing unnecessary monetary cost.
   (a) **Cost:** minimize cost of administering insurance claims to eligible claimants.
   (b) **Avoid overpayment:** minimize sum of overpayment amounts, i.e. reduce false positive eligibility.

Efficiency reflects a fiduciary obligation to avoid unnecessary costs. Effectiveness formulates one central aim of UI, namely, to provide insurance payments. Effectiveness is relevant already at the point at which individuals decide whether or not to claim UI. Call this decision the first stage. In the second stage, when workforce agencies make determination decisions about claim eligibility, effectiveness demands that underpayments are avoided and that payments are made quickly.

Efficiency and effectiveness can conflict. For example, efficiency demands to avoid overpayment, whereas effectiveness demands to avoid underpayment. An increase in one value leads to a decrease in the other. On the assumption that eligibility is hard to measure — so that when we predict eligibility the distributions of claims that are in fact eligible and those that are not overlap (see Fig. 1) — avoiding underpayment comes at the expense of increased overpayment. Workforce agencies therefore need a decision-making policy that determines the cutoff point. All claims with an eligibility score below this point are rejected and all claims with an eligibility score higher than this point are accepted (see Fig. 1).

This conflict between efficiency and effectiveness arises even when no explicit eligibility scores are used. Classical statistical hypothesis testing teaches that efforts to reduce the odds of doing the wrong thing (a Type I error) generally increase the odds of not doing the right thing (a Type II error). It is therefore reasonable to assume that efforts to prevent improper overpayments correspondingly make it more likely that improper underpayments will occur.

This conflict can play out along multiple causal pathways. For example, fraud might be reduced by requiring claimants to prove having qualified dependents in triplicate instead of a single source. This requirement may cause some recipients to become ineligible even though they are, in fact, eligible. Underpayment errors would result on the margin for the appropriate dependent allowance, or in full if the recipient's claim is placed on administrative hold pending determination or if a claimant elects to not to complete their application or appeal in light of the additional documentation required.

It should be noted that different errors are associated with vastly different practical results. Failing to detect an underpayment is different from failing to detect an overpayment, both from ethical and economic perspectives. From the claimants perspective, failing to detect an underpayment is usually worse than failing to detect an overpayment. The practical consequences of not receiving claims because an agency mistakenly determined them to be ineligible are severe [17, 19].

Efficiency and effectiveness also conflict in their respective dimensions of cost (efficiency) and opportunity (effectiveness). This is because making the claim process easy and convenient is expensive. For example, computer voice assistants or chat bots are cheaper than a human customer service representative but chat bots are currently unlikely to answer concerns that claimants have satisfactorily. The question of whether to replace an expensive but effective call center with more efficient software poses another trade-off between effectiveness and efficiency.

This conflict between conflict between efficiency and effectiveness is relevant for two reasons. First, the conflict can inform an analysis of the legal and policy history. We argue in the next section that social insurance programs in the US tended to focus on efficiency. Second, the conflict motivates our investigation into the use of AI to identify administrative errors, which we undertake in the subsequent sections. Avoiding under- and over-payment is immediately relevant to automatic classification (because these

[2]We concentrate on these two because they are immediately relevant to classification errors. Avoiding under- and over-payment is avoiding false negatives and positives respectively.

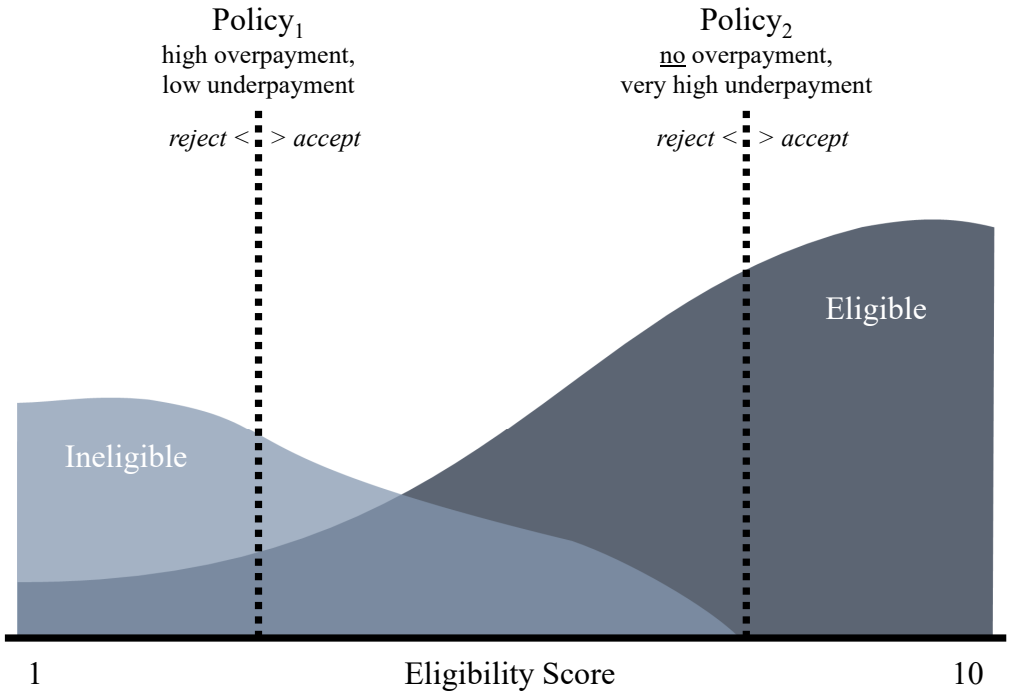


Figure 1: The conflict between avoiding over- and under-payments is resolved differently by different decision policies.

two values related to avoiding false negatives and positives respectively). AI could be a way of partially overcoming the conflict between efficiency and effectiveness.

Finally, one further public value will be relevant for our discussion: explainability. *Explainability*, for the purposes here, means that a workforce agency can provide reasons for each decision that they reached. When AI is used for decision-making, this requires that this use of AI is both scrutable (in some sense) as well as intuitive. ML models often fail on both counts [44]. Explainability is an important value for intrinsic as well as for instrumental reasons. Intrinsically, the idea that a government agency can explain its decisions, even when they are made by AI, is rooted in democratic theory [5]. Instrumentally, explainability, firstly, could help prevent future mistakes — assuming that explainability allows that causes of administrative errors are recognized, understood, and rectified faster[49]. Secondly, explainability is instrumentally valuable insofar as it allows to document reasons for which decisions were reached. In this way, explainability instrumentally promotes the public value of procedural due process or rule of law [27, 38].

3 BACKGROUND

US social insurance programs have been the subject of policy and administration studies since their inception in the 1930s and extension in the 1960s. In 1947, Simon noted that, with the exception of military logistics, most of what was then known about the science of administration in public organizations was owed to research on programs such as the Aid to Families with Dependent Children (AFDC, now Temporary Assistance to Needy Families, or TANF), Social Security, or Food Stamps (now Supplemental Nutrition Assistance Program, or SNAP) [45].

US social insurance programs are interesting for several reasons. Their size and scope set them apart from other administrative divisions of government. Their public-facing nature gave rise to a new type of street-level bureaucrat [32], as well as a new source of administrative burden [10]. Social insurance programs are moreover highly political, both because they are large and because they redistribute money across individuals and over time.

The political nature of these programs and policies is reflected in their administrative complexity. With few exceptions, US social insurance programs are targeted towards specific sub-populations that are socially constructed as deserving of material aid [43]. Additional complexity arises from other policy design choices. These include delegating implementation to state or county governments; limitations on the benefit eligibility by length of time; restrictions on goods or services purchased using benefits; and limits to other sources of income while receiving benefits. Complexity is increased further still by the fact that these policies are mutable, and have been altered by legislation multiple times over the years.

From an administrative perspective, these design choices have the net effect that determining whether a claimant is eligible for benefits, and if so at what level, often requires information from multiple parties — claimants, other government bureaus or agencies, and employers. All this complexity gives rise to the risk that decisions are made in error. When such errors occur frequently enough in programs that are as large as welfare, Medicare, or UI, the result is a significant misallocation of public money.

3.1 Institutional History: Focus on Overpayments

Throughout its history, legislation on social insurance concentrated predominantly on avoiding overpayments. The US Federal government recognized the increasing financial and political toll of administrative errors in social insurance programs in the 1970s. This led to a "war on fraud and error." Quality control (QC) audit processes were adapted from manufacturing operations and introduced to social policy [7, 36]. Broadly speaking, in QC both federal and state agencies audit samples of programmatic claims data. Federal agencies would then use these audits as input for determining fault tolerance thresholds, which then served as a benchmark for individual state agencies [31]. Over time, the emphasis of QC morphed from identifying error sources and establishing corrective measures to a performance management regime under the Financial Integrity Act of 1982 (FIA). In addition to additional reporting requirements, the FIA introduced sanctions, in the form of withholding of federal funding, for States that failed to keep their error rates below the established tolerance level [47]. This change reflected both the politicization of QC as an instrument of control, and the broader trend in emphasizing efficiency over effectiveness in public policy and administration [7, 29, 33].

Attention to identifying the source of administrative errors in social insurance programs, as well as strategies for reducing them, was reintroduced in the two-page Improper Payments Information Act of 2002. In addition to formally defining these errors as "improper payments," the law requires agency heads to provide information on the causes of any identified improper payments, strategies enacted or planned to reduce those errors, and whether the agency believes it possesses the infrastructure necessary to enact the reductions. These changes were further supplemented by the Improper Payments Elimination and Recovery Act of 2010 and the Improper Payments Elimination and Recovery Improvement Act of 2012, which established procedures for recovering overpayments and required program administrators to verify that their benefit distribution systems included a pre-verification check as an additional safeguard for preventing payments to ineligible recipients.

These 21st century changes to federal performance management of social insurance administration occurred at the same time as new technological approaches for auditing large, complex data for patterns of behavior became viable for commercial and public use. Advances in ML, in conjunction with exponential increases in digital data generation, storage, and computation capacity, made it possible to analyze large-n high-dimensional data in real time and mine it for patterns that would otherwise be imperceptible to human auditors. In the private sector, ML has been widely adopted in efforts to identify financial fraud, both ex post and ex ante [1, 12]. In the public sector, ML approaches are increasingly used for detecting medical insurance fraud and improper payments [4, 20–22, 34]. However, we are unable to find any publicly available research on the use of ML to identify underpayment of entitled funds when auditing payment systems.

3.2 Empirical Context: Unemployment Insurance

Unemployment Insurance was established as part of the Social Security Act of 1935. It is principally funded through a tax on employers. State governments are responsible for financing benefits, and implementation. The federal government provides oversight and covers the program's administration costs. In addition to federal eligibility standards, states are empowered to design their own requirements with respect to eligibility, benefit amounts, length of benefit spells, and disqualification conditions and penalties. Over time these state-level variations in UI policy have become sufficiently complex that the DOL publishes an annual report, The Comparison of State Unemployment Insurance Laws, to document these differences.

Although the definition of "improper payments" in the 2002 Improper Payments Act included underpayments to eligible recipients, the substantive and normative focus of federal performance management of social insurance programs is decidedly oriented towards minimizing overpayments. Overpayments can be classified into two general forms. One form of overpayment occurs when the recipient knowingly and willfully misleads the state with respect to their eligibility information — this is fraud. Reducing administrative errors that result in fraudulent payments is an understandably salient goal for both political and administrative managers. But overpayment errors can also arise due to unintentional errors in eligibility determination, either by the claimant or the administrative agency and its staff. Correcting these errors is likely to be particularly appealing to administrative managers and staff, as they are more likely to be held largely or solely responsible for them. But the third form of improper payment — the underpayment of benefits to eligible recipients — receives considerably less attention by legislators, administrators, and researchers (though see [37] for an example of early recognition of this imbalance in attention).

4 METHODS

In what follows, we describe the data and the machine learning methods used for this project.

4.1 Data

Primary data are drawn from the Department of Labor's Benefits Accuracy Measurement system [16]. This dataset reports results from randomly sampled investigations into UI claims. It provides federally-collected information of improper UI payments, one form of administrative error. The final, analysis-ready dataset contains 785,159 observations, where each observation contains information about one unemployment benefits claim made during the years 2002–2018. Each observation is characterized by 228 features (variables), which contain personal information of the claimant (date of birth, gender, race, etc), information about last employment of the claimant (occupation code, salary, etc), information about interaction between claimant and agency (how claim was filled, was it filled on time or not, etc.), as well as other information.

4.2 Analytic Strategy

Predicting administrative errors is a classification problem. Classification is one of the fundamental tasks in machine learning, and has the goal of predicting the category of a data point. A simple

example of a classification problem is spam detection. An email falls into one of two categories: either it is spam or it is not. Each new email is a data point from which an algorithm predicts whether or not this email is spam. In this paper, we train classifiers using supervised learning. That is, we have a set of *labeled instances* in which both the data point and the classification is given. The labels describe the category of a particular data point. On these labelled instances a *model* is trained that then is used to predict the labels of *unlabeled instances* [2]. In the following, we describe some of the main ML-approaches that we examine in this paper. Although we will report results from testing seven methods (including four deep learning approaches) in total, for limitations of space, we only describe two in detail.

4.2.1 Overview of Machine Learning Classification. Formally, one begins with a *training set* T consisting of n data points $t_1, \ldots, t_n$. Each data point t_i is associated with a feature vector $F_i = (f_{i,1}, \ldots, f_{i,m})$ describing its properties. These features (or variables) may be numerical or categorical. In the email example, relevant features may include the length of the email in words, the time the email was sent, whether the recipient has previously responded to emails from the sender, etc. A categorical feature may be ordinal (i.e., an ordering among the categories exists, like whether the email was sent "High Priority", "Normal", or "Low Priority") or nominal (i.e., no such ordering exists, like whether the sender's email address is from "Gmail.com", "Yahoo.com", etc.). Most classification algorithms require that features be numerical, and so categorical features must be converted to numerical before such data can be used [2].

Additionally, each data point in the training set is associated with a label representing its category (or class). This category is the value that the algorithm will learn to predict, such as "spam" vs. "not spam". These categories are assumed to be known; they may, for example, have been identified by a human individually examining each email.

Next, using the training set, a ML algorithm attempts to train a model, i.e. learn the patterns distinguishing the various categories from one another. The algorithm may learn, for instance, that spam emails tend to come from senders with email addresses from certain domains, and are sent at odd hours of the day. Non-spam emails, on the other hand, come from senders who the recipient has previously replied to, or come from domains matching the domain of the recipient, and so on. Finally, once a model has been trained, it can be applied to unlabeled data in order to perform automatic category prediction.

A model trained to perform classification can reveal other useful information in addition to making predictions on unlabeled data. For example, one can identify which features are most useful for distinguishing between classes (e.g., one might observe that the most useful features for distinguishing spam from non-spam emails are whether the recipient has ever responded to the sender, whether the domain of the sender matches that of the recipient, and the length of the email) [28]. With longitudinal data, one can then explore how classification patterns change over time (for example, perhaps as spammers become more sophisticated, their emails become longer). One can also cluster the data based on feature values, thus identifying the dominant patterns in the dataset [23].

Machine Learning includes many different techniques to train models; more are being developed in on-going research. One very simple algorithm is known as k-Nearest Neighbors (k-NN) [24]. In this algorithm, for each unlabeled data point d, the algorithm identifies the k closest data points from the training set (i.e., those with known labels), and assigns d the category label belonging to a majority or plurality of those known labels. The Naive Bayes approach treats each feature as independent of the others, and then for each feature, through application of Bayes' Theorem, computes the probability that the data point belongs to a particular category given its feature value [30]. By multiplying these probabilities across features, one can derive the probability that the data point belongs to each class. Other popular methods include the Support Vector Machine, which represents the data points in the feature space and then identifies a *boundary* that best separates the classes from one another [6]. By properly mapping the data points to the feature space, this boundary can be efficiently found.

A final important category of algorithms is neural networks and deep learning. Neural networks are a class of algorithms inspired by how brains operate. In these algorithms, mathematical functions called *neurons* are wired together into networks [35]. The analogy to brains consists in the fact that the neurons in these networks transmit signals to other neurons. The output of each neuron is a combination of the signals sent to it as input, and weights indicating the strength of each signal are adjusted as the algorithm proceeds. These linked neurons can be organized into layers, where each layer of neurons receives inputs from the previous layer and forwards its output to the next layer. When a network consists of many layers, we speak of *deep learning*. Each of the layers learns higher and higher-level features from the input data. As layers progress, the input is transformed into high-level features. Most current deep learning methods are based on neural networks.

In the next two sections, we discuss two of the ML algorithms that we use in this paper in depth (due to space, we cannot give a comprehensive description of all methods). Specifically, we describe **CatBoost**, a recent decision-tree method; and after that, we describe **TabNet**, a deep learning method designed for tabular data.

4.2.2 CatBoost: Decision Tree-Based Classification. CatBoost, short for "Category Boosting", belongs to a class of algorithms that make predictions using what is known as *decision trees* [42]. A decision tree is a flowchart-type object in which feature values are used to determine which branch in the flowchart to take, until a prediction is arrived at [26]. The simplest decision tree algorithm constructs such a flowchart by identifying which single feature (and value) is most predictive of category, and builds a branch corresponding to that feature value.

CatBoost uses a variant of this approach. It has been observed that sets of slightly different models collectively make better decisions than single models. A subclass of decision tree algorithms use this observation, but instead of constructing these models independently, builds them sequentially so that each model can improve upon the last. This is called *additive training* or *gradient boost*. Intuitively, the idea is that learning is improved because each new tree can correct errors from past trees. Members of this class include XGBoost (eXtreme Gradient Boosting) [14] and GBM (Gradient Boosting Machine) [39]. Finally, also the CatBoost algorithm used

in this paper [42] uses this method. These algorithms generally show similar behavior on data. We selected CatBoost primarily because it offers a significant speedup advantage when implemented on a GPU (a graphics processing unit, which allows for high levels of parallelism).

Originally, CatBoost was designed to address a flaw in previous gradient boosting algorithms known as *prediction shift*. Essentially, prediction shift is a special type of *target leakage*, which occurs when training data contains information about the target variable (category label) that would not be available to the algorithm for an unlabeled data point. This occurs because by training trees iteratively, the gradients themselves reveal information about the target variables. To address this, CatBoost uses a weighted sampling method known as *ordered boosting*, and samples new training datasets independently.[3] We use the "MultiClass" optimization in the standard CatBoost implementation, which uses a Multiclass Cross-Entropy Loss (log-loss) function given by:

$$MCE = \frac{\sum_{i=1}^{N} w_i log\left(\frac{e^{a_{it_i}}}{\sum_{j=0}^{M-1} e^{a_{ij}}}\right)}{\sum_{i=1}^{N} w_i}$$

where a_ij represents the predicted probability that element i belongs to class j and the w_i values represent weights associated with each element.[4]

4.2.3 TabNet: Deep Learning for Tabular Data. TabNet is a deep learning method, designed by Google, for tabular data [3]. Generally speaking, deep learning methods excel on unstructured data, such as images, but perform badly on structured data, such as tables. One reason for this is that data types in tabular data are often heterogeneous (i.e. they represent fundamentally different things), and the data are often sparse or non-continuous in the feature space. TabNet was developed as a deep learning method to perform better given such data. Because the data available for this project consist of such tabular data, we chose to assess how TabNet models performs in predicting administrative errors.

TabNet inputs raw tabular data, and then uses feature selection to identify the features important for any particular data instance. Through non-linear processing of the selected features, TabNet aims to ultimately provide a 'decision tree-like mapping' to obtain interpretable results. Experiments by its authors show that TabNet can far outperform other deep learning methods — although as we will see, in our experiments, it does not perform particularly well.

4.3 Experimental Setup

First, to prepare the data for use by a ML algorithm, we performed several preprocessing steps:

(1) Features that could act as direct proxies for the target variables were removed. Examples include features like "totaloverpayment" or "underpayment" (which contain information about total overpayment/underpayment in the dollars), or features which contain corrected information obtained after investigation (as this information was unavailable at the time of the original claim audit);
(2) All non-numerical features were converted to categorical features;
(3) For the Logistic Regression and Random Forest algorithms, most of the features were converted to categorical. Some features (such as date of birth) were removed, as categorical representation of this data is very sparse and leads to a very high dimensionality of the processed data.

We then tested the following algorithms:

(1) **Logistic regression** (LR) is one of the most basic algorithms for the classification problems. In the case of multiclass classification, LR is trained for each class separately (i.e., the one-vs-all scheme).
(2) **Random forest** (RF) uses an ensemble of the decision trees trained on different random subsets of the data. RF is very commonly used for tabular data.
(3) **CatBoost classifier** is another, more powerful way of combining decision trees in the one model.
(4) The following deep learning/neural network algorithms:
   (a) **TabNet** is designed for dealing specifically with tabular data. TabNet uses sequential attention and has been shown to perform well on tabular data [3].
   (b) **DeepFM** combines factorization machines for recommendation with neural networks for learning features, and does not require feature engineering. It is not intended specifically for use on tabular data, but can be used in that setting [25].
   (c) **WideDeep** is based on Google's Wide & Deep algorithm, which combines 'wide' linear models with 'deep' neural networks. This algorithm has been used commercially on Google Play [15].
   (d) **DCN** stands for Deep & Cross Network, a type of neural network with feature crossing at each layer, which doesn't require manual feature engineering. It works well for the tabular data. [46]

To split the data into training and test sets, we use the following approaches. In the first scenario, we sample 30% of the dataset uniformly at random (this was because the dataset was too large to handle in its entirety). This sampled data was further split into an 80% training set and a 20% testing set, corresponding to the 16% and 4% of the original data set. The model is trained using data from the training set, and evaluated on the test set.

In the second scenario, dataset predictions were made only for specific year of claims, and models were trained on the data from the previous year or three previous years. This approach is a more realistic representation of actual applications.

4.4 Evaluation Metrics

As mentioned above, we use a dataset from the Department of Labor's Benefits Accuracy Measurement system [16]. Each observation in this dataset is a claim that was randomly selected to be investigated for improper payments. Each observation in the dataset, or each claim, belongs to one of the following classes: "No error", "Overpayment", "Underpayment", or "Wrong issue." "No error" means that these claims were processed correctly and successfully. "Overpayment" and "Underpayment" mean that benefit payments were made that were too high or too low respectively.

[3] There are some other differences between CatBoost and previous methods, but this is the main one.
[4] See https://catboost.ai/docs/concepts/loss-functions-multiclassification.html

Table 1: Count and proportion of improper payment errors by type

Type	Count	Proportion
No Error	629,445	0.807
Overpayment	74,983	0.096
Underpayment	59,080	0.076
Wrong Issue	16,090	0.021
Total	779,598	1

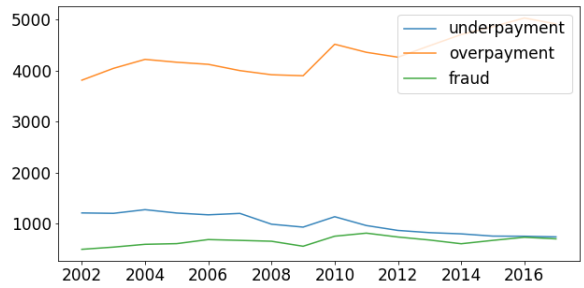


Figure 2: Improper payment error trends over time by type of error

"Wrong issue" means that an error was made in the claim in a way that was unrelated to the level of payment. Some of the samples additionally belong to the "Fraud" class, which indicates that a claim is an unlawful attempt to obtain unemployment benefits. We report the fraud class of claims only in our descriptive statistics. Because fraud is a subset of overpayment, all claims labeled as "Fraud" are treated as "Overpayment" errors in our analysis.

As shown in Table 1, these classes are highly imbalanced in the dataset. Since administrative errors are an exception, the majority of samples belong to the class "No Error". This makes evaluation of the model somewhat complicated. If only the accuracy of the model is considered, defined as the percentage of correctly classified samples, then a trivial "model" which simply classifies all samples to as belonging to the "No Error" class would achieve a very good accuracy of approximately 80% — without doing any meaningful prediction at all.

To analyze models meaningfully, given that the classes are highly imbalanced, we use several different evaluation metrics. In particular, we use the metrics known as precision, recall, and F-score. *Precision* is defined as the proportion of all positive predictions of the class that are true positives of this class. *Recall*, or probability of detection, is defined as the proportion of all true positives of the class which were positively identified. *F-score* is then defined as the harmonic mean between precision and recall. Intuitively, that means that F-score will be low if precision or recall are low. For the case when precision and recall are both equal to 1 — meaning that the classifier was perfectly accurate — the F-score also is equal to 1. For the multi-class setting, the F-score can be computed in the following two ways:

(1) Recall and precision can be computed over all samples, from all classes. The F-score based on these values is called the ***micro* F-score**.
(2) Recall and precision can be computed separately for each class. In this case, the averaged F-score over all classes called the ***macro* F-score**.

5 RESULTS

We begin with a summary of key descriptive statistics and trends over time. In a second step, we report results of the classification analysis of different ML approaches.

5.1 Descriptive Statistics

In our data, overpayment errors caused by fraud totaled over $25 million, while the total cost of non-fraudulent overpayments was approximately $56 million. The vast majority (87%) of overpayment claims were non-fraudulent. The total amount of underpaid money in our data (i.e., associated with "Underpayment") is approximately $5 million.

The various types of errors have different distributions over time. As can be seen in Figure 2, the overpayment error rate was increasing up until approximately 2016, and then started to decrease. The underpayment error rate has also been slowly decreasing over time. In contrast, the fraud rate has remained roughly constant.

The error types also vary by State, as seen in Figure 3, which aggregate results for all years. These differences may be explainable by differences in State policies. For example, Ohio has a high rate of underpayment errors (one of the highest in the country), but a comparatively low rate of fraud.

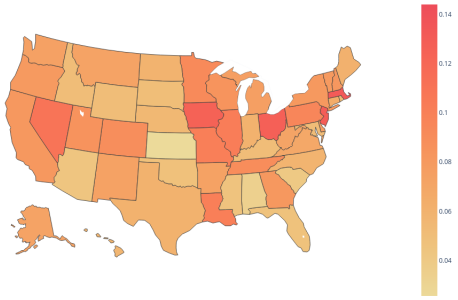
5.2 Classification Analysis

Our goal is to assess whether ML can be used to predict the error-type (if any) of a claim, using the features of that claim, and to use the output from machine learning models to further analyze the data. In these experiments, the class of the claim (e.g., underpayment) was treated as the target variable.

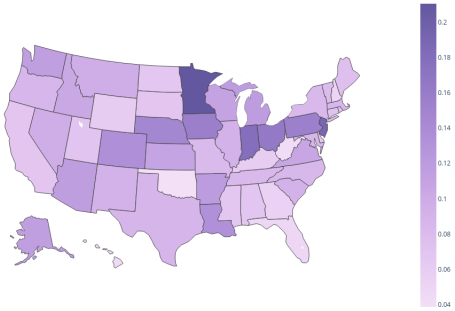
For the first experimental setup (a randomized split of the data, described in more detail in Section 4), results are shown in Table 2. CatBoost demonstrated the best performance. Random Forest performed worse, but not substantially, and Logistic Regression was unable to generalize properly over the data and performed poorly. Tuned Gradient Boosting-based methods, like CatBoost, often outperform RF [11] but in cases of noisy data, as here, performance differences may be small.

For the second experimental setup, the model was trained only on data from the previous one or three years to predict errors in the next year. Results are shown in Figure 4. Because CatBoost performed the best, we present only its results. (Surprisingly, TabNet, although it was designed to perform very well on tabular data, did relatively poorly.) Interestingly, CatBoost's performance improves over time. This could indicate gradual improvement of the administrative procedures, and investigating the causes of this behavior is an avenue for future research.

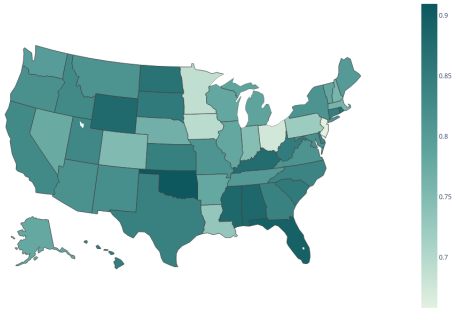
As part of its output, CatBoost is able to provide a ranking of the features based on how important they were to the classification.



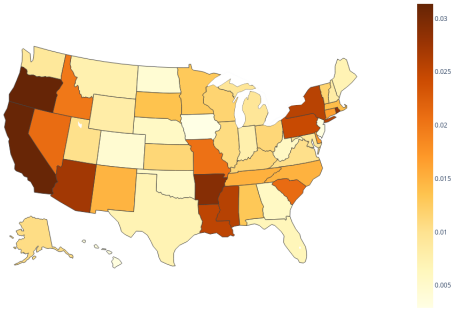
(a) Percentage of claims with an underpayment error.



(b) Percentage of claims with an overpayment error.



(c) Percentage of successful claims.



(d) Percentage of fraudulent claims.

Figure 3: Relative rate of improper payment errors by type of error across states

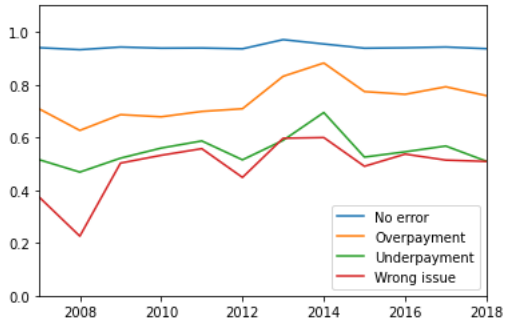
Table 2: F-Scores (micro and macro), Precision, and Recall Values by Classifier Type by Class using data from all time periods

Name	F-score (macro)	F-score (micro)	Precision "no errors"	Precision "over-payment"	Precision "under-payment"	Precision "wrong issue"	Recall "no errors"	Recall "over-payment"	Recall "under-payment"	Recall "wrong issue"
LR	0.229	0.801	0.805	0.211	0.088	0.00	0.994	0.013	0.001	0.00
RF	0.284	0.805	0.814	0.498	0.278	0.400	0.985	0.181	0.023	0.042
CatBoost	0.459	0.842	0.853	0.720	0.655	0.705	0.981	0.409	0.143	0.096
TabNet	0.313	0.808	0.819	0.511	0.348	0.734	0.981	0.177	0.006	0.045
DeepFM	0.289	0.807	0.823	0.459	0.250	0.727	0.970	0.237	0.002	0.008
WideDeep	0.311	0.810	0.825	0.455	0.000	0.704	0.966	0.261	0.000	0.018
DCN	0.337	0.804	0.813	0.565	0.000	0.500	0.992	0.105	0.000	0.004

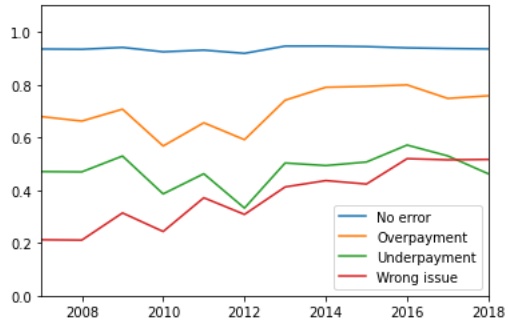
Features which were identified as especially important can be found in Figure 5, with description in Table 3.

Most of these important features can be grouped into one of these three sets:

(1) Features which describe the individual's previous occupation, including salary;
(2) Features related to time (date of the claim, etc);
(3) Features with information about administrative decisions made prior to the benefit audit.



(a) F1 score for CatBoost model trained on data from 1 previous years and evaluated on the next year.



(b) F1 score for CatBoost model trained on the data from 3 previous years and evaluated on the next year.

Figure 4: Performance of CatBoost when trained on previous years' data

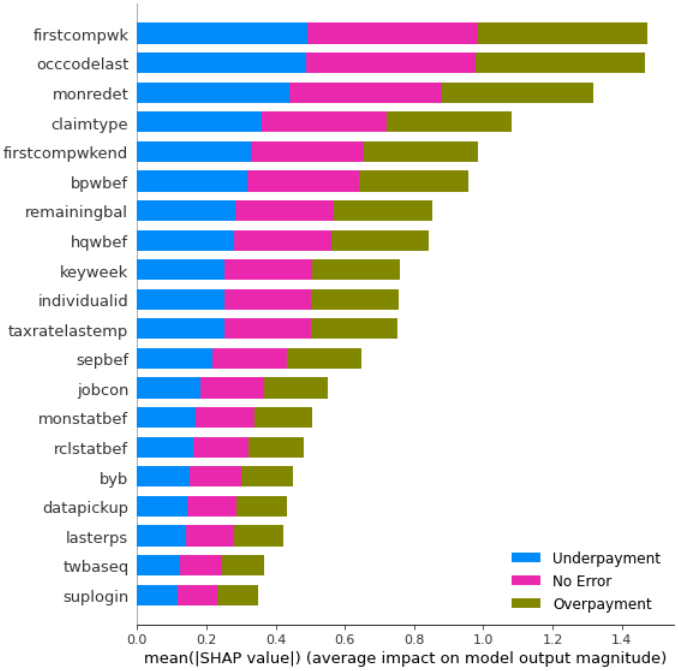


Figure 5: CatBoost feature importance

6 DISCUSSION

In this section we discuss the results of our evaluative assessment of model performance. In particular, we investigate the results of CatBoost, discuss the importance of goodness of fit between data and methods, highlight how model explainability relates to public values, and identify limitations of our analysis.

6.1 Model Performance

The results show that all of the evaluated models, including logistic regression, have reasonably high micro F-scores. This means that the average performance with respect to making both Type I and Type II errors is reasonably strong for all models tested. This, in turn, suggests that ML-based AI may be a good tool for auditing administrative data for errors. However, a closer look reveals crucial performance differences across and within all models that have substantive implications for their efficacy in practice. When we evaluate model performance by individual class, it is clear that the micro F-scores are unduly positively inflated by the precision and recall values for the "No errors" class. This is a problem, because while overall performance is important, the most important task is to correctly identify and classify errors. Model performance deteriorates sharply when each class of error is evaluated separately. This is particularly severe in the case of the WideDeep and DCN models for both precision and recall with respect to underpayment errors: they are completely incapable of classifying this type of error. DeepFM has similar challenges, though its precision score

Table 3: Descriptions of variables that are important for the CatBoost model

Name of the feature	Description of the feature
firstcompwk	Date of first compensable week
occodelast	Occupation code of the last employment
monredet	Monetary redetermination
claimtype	Type of UI claim (new claim, reopened claim, transitional claim)
firstcompwkend	First compensable week ending date
bpwbef	Base period wage before the investigation
remainingbal	Remaining Balance (RB) as of key week ending date
hqwbef	High quarter wages before investigation
keyweek	Week in which claim was filed (beginning date)
individualid	Numeric indicator for each individual UI claim
taxratelastemp	Last tax rate for the individual
sepbef	Reason for separation determination before investigation
jobcon	Number of job contacts listed for key week
monstatbef	Reason for monetary denial before investigation
rclstatbef	Claimants recall status for the determination before investigation
byb	Benefit year beginning
datapickup	Date of picking up data for storage
lasterps	Date of claimants most recent ERP up to and including Key Week
twbaseq	Taxable Wage Base
suplogin	Supervisor identification

is at least equivalent to classifying at random (though this is still decidedly poor performance)[5]. In fact, none of the models' recall values for underpayment or for "Wrong issue" are better than choosing at random. Performance improves somewhat with respect to recall for overpayment errors, but here only CatBoost and WideDeep perform better than choosing at random. Model precision with respect to the three classes of error is stronger — with the exception of logistic regression, which performs uniformly poorly — but still significantly worse than one would expect from their micro F-scores. Once again, WideDeep and DCN are incapable of handling underpayment errors, and DeepFM is only equivalent to random choice.

6.2 CatBoost Performance

It is interesting to note that CatBoost outperformed all other models across all measures, with the singular exception of precision for "Wrong issue" errors, where DeepFM and WideNet scored higher (though this margin is small, and both DL models perform significantly worse than CatBoost for most metrics). CatBoost's dominance is particularly noteworthy for precision on both over- and underpayment errors. This result runs contra to the broader enthusiasm for the power of DL classifiers in popular science and the media. It may therefore come as a surprise to public administrators and managers who are not fully versed in the technical capabilities and limitations of modern AI research that *a random forest-based approach would be the best fit for the task of auditing administrative data*. Viewed in this light, our findings highlight the need for public administrators and managers either looking to adopt AI in their organizations or being sold on the prospect by private software and professional services vendors to become "informed consumers" of these technologies.

Similarly, the fact that CatBoost performed better when trained on only the previous year's data compared to three preceding years may be counter-intuitive. Conventional wisdom suggests that, all else equal, analytic performance should increase as more relevant data are available. One possible case-specific explanation for these results is the volatility of UI claims over time. As its name implies, usage of unemployment insurance is highly correlated to both national and regional labor market conditions; as employment possibilities worsen, the number and variety of UI claims increases (and vice versa). It is possible that using training data that are lagged by more than one year attenuates classifier performance because the fundamental labor market conditions that motivated prior claims no longer apply. Exploring the relationship between historical claims data, labor market conditions, and whether and how they condition data to be more or less useful for training classifiers is one avenue for future research.

6.3 Importance of Goodness of Fit

Our results also demonstrate the importance of assessing the goodness of fit between technology and task. Here, the first assessment is whether AI should be used to audit UI claims in particular, and social insurance claims in general. The evidence paints a contingent picture. In terms of general performance, every ML classifier demonstrated some capacity to correctly predict erroneous claims that exceeded random chance. This suggests that AI has, at a minimum, the potential to be a useful tool for helping auditors screen the millions of claims that are filed annually. However, the performance of all of the classifiers, including CatBoost, is too poor to

[5]Importantly, this comparison is made against a best-case counterfacutal dataset where all 4 classes were perfectly balanced, with the odds of making a correct decision at random being 1/4 = 0.25

recommend their use in unsupervised settings or for their decisions to be given too much weight without substantive follow-up by human auditors. In this way, our results suggest a use for AI as a decision support system (DSS) to help initially filter UI claims in likely need of further investigation.

Conditional on deciding to implement AI in this context, the second assessment of fit is between different classifiers. Here, our analysis suggests that the CatBoost classifier is the dominant choice (again, with the singular exception of precision for "Wrong issue" errors) among those tested. However, in practice it is possible, if not likely, that certain classifiers will outperform others for detecting some types of errors but not others. This suggests that an ensemble approach drawing on the optimal classifier on a case-by-case basis rather than a "one size fits all" uniform solution is likely to produce the best results for administrative organizations.

6.4 Explainability

An important potential advantage of CatBoost is its explainability. In contrast to DL models, which are generally hard to scrutinize and explain intuitively [44], CatBoost essentially learns a flow-chart diagram for classification. In this way, CatBoost is scrutable. For example, a CatBoost can relatively easily be investigated with respect to which features drive classifications, as in Figure 5. As discussed in Section 2, explainability is an important public value related to democratic theory and the rule of law. Also in the case of administrative errors, explainability is a relevant value for intrinsic as well as instrumental reasons.

An explainable model potentially allows to understand why a certain administrative error occurred. This, as such, promotes public values in the sense that it furthers transparency and accountability towards claimants and citizens. Moreover, an explainable model is easily consistent with demands rooted in the value of the rule of law, requiring that decisions are not made arbitrarily. The same goes for the identification of administrative errors. In order to accord with the rule of law, the reasons why an investigation was opened, or the reasons why an improper payment occurred, need to be recorded.

From a policy and managerial perspective, the benefits of explainability also include the potential for proactively addressing the source of administrative errors. Taken to its most extreme for illustration, a classifier that could perfectly predict which claims contained errors but could not explain how it arrived at a given decision would have no practical use for those interested in preventing errors before they occur. Explainability allows agency staff to learn from what the machine learned, and use this knowledge to make programmatic, technical, or other changes to prevent future errors. This capacity is of critical importance to the State workforce agencies responsible for administering UI in the US, because the federal government evaluates state agency performance - and imposes funding sanctions for nonperformance - based on the number of detected payment errors.

6.5 Limitations

Our results require important caveats. The most important of these is that we are limited to the use of publicly accessible data on UI claims. While these data are relatively rich and span a long period of time by the standards of public administration and policy research, they are limited in two fundamental ways. First, they are a (stratified proportional random) sample of the population of UI claims from 2002-2018. This limitation is partially a function of the need to sample large-n data for auditing using traditional, non-ML methods, and also likely due to cost and technical limitations with respect to making the data publicly accessible. Second, they do not include sensitive personal, financial, and employment data that are available to State workforce agencies and the Department of Labor. This latter limitation is necessarily born to protect the privacy and security of claimants. But both nevertheless limit our classifiers' potential performance. Both precision and recall are likely to increase if the models were trained on the full population of claims and the full set of features available to government auditors. Furthermore, this may be a particular handicap to the Deep Learning classifiers, which are particularly well-suited for identifying patterns in high-dimensional, complex data. It may be that DL classifiers outperform CatBoost when trained on "live" data.

With these limitations in mind, our results still contribute to our understanding of AI's potential for identifying administrative errors in social insurance programs. When considered as a whole, our findings highlight the importance of distinguishing between overall precision and recall vs. by-class scores when dealing with unbalanced data, and thus illustrate the challenge of evaluating AI performance for public managers.

7 CONCLUSION

We used the case of unemployment insurance (UI) in the United States to consider the ethical and practical dimensions of using AI to detect administrative errors, operationalized as improper payments of UI benefits. Drawing upon longitudinal data on claims audits and State-level UI policy differences, we trained and evaluated several types of classifiers, including logistic regression, random forest, and deep learning models. Our results show that a random forest classifier using gradient descent boosting (CatBoost) outperformed all others, including multiple popular deep learning models. We then evaluated this classifier's performance when training data were restricted to the previous year or the previous three years, and found that performance was superior for all classes when using only the last year of data.

Our results contribute to the literature on AI applications in the public sector, and also have value for practitioners. Peculiarities of administrative data make it crucial to assess beforehand how well some technology fits a given task. Likewise, public administrators should not assume that well-known technologies, such as deep learning, will necessarily perform best. Furthermore, restricting training data may increase performance. Future research is needed to extend these findings into other administrative and policy domains, to incorporate additional features, and to examine whether and how variations of State-level policies and administration contribute to differential rates of errors across jurisdictions.

REFERENCES

[1] Aderemi O Adewumi and Andronicus A Akinyelu. 2017. A survey of machine-learning and nature-inspired based credit card fraud detection techniques. *International Journal of System Assurance Engineering and Management* 8, 2 (2017), 937–953.

[2] Ethem Alpaydin. 2020. *Introduction to machine learning* (4 ed.). The MIT press, Cambridge, Massachusetts.

[3] Sercan O Arik and Tomas Pfister. 2019. Tabnet: Attentive interpretable tabular learning. *arXiv preprint arXiv:1908.07442* (2019).

[4] Richard A Bauder and Taghi M Khoshgoftaar. 2017. Medicare fraud detection using machine learning methods. In *2017 16th IEEE international conference on machine learning and applications (ICMLA)*. IEEE, 858–865.

[5] Reuben Binns. 2018. Algorithmic Accountability and Public Reason. *Philosophy & Technology* 31 (2018), 543–556. https://doi.org/10.1007/s13347-017-0263-5

[6] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. 1992. A Training Algorithm for Optimal Margin Classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory* (Pittsburgh, Pennsylvania, USA) *(COLT '92)*. Association for Computing Machinery, New York, NY, USA, 144–152. https://doi.org/10.1145/130385.130401

[7] Evelyn Brodkin and Michael Lipsky. 1983. Quality control in AFDC as an administrative strategy. *Social Service Review* 57, 1 (1983), 1–34.

[8] Justin B Bullock, Robert A Greer, and Laurence J O'Toole Jr. 2019. Managing risks in public organizations: A conceptual foundation and research agenda. *Perspectives on Public Management and Governance* 2, 1 (2019), 75–87.

[9] Justin B. Bullock, Matthew M. Young, and Yi Fan Wang. 2020. Artificial intelligence, bureaucratic form, and discretion in public service. *Information Polity* 25, 4 (Jan 2020), 491–506. https://doi.org/10.3233/IP-200223

[10] Barry C Burden, David T Canon, Kenneth R Mayer, and Donald P Moynihan. 2012. The effect of administrative burden on bureaucratic perception of policies: Evidence from election administration. *Public Administration Review* 72, 5 (2012), 741–751.

[11] Rich Caruana and Alexandru Niculescu-Mizil. 2006. An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning*. 161–168.

[12] Raghavendra Chalapathy and Sanjay Chawla. 2019. Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407* (2019).

[13] Robert Charette. 2018. Michigan's MiDAS Unemployment System: Algorithm Alchemy Created Lead, Not Gold - IEEE Spectrum. https://spectrum.ieee.org/riskfactor/computing/software/michigans-midas-unemployment-system-algorithm-alchemy-that-created-lead-not-gold

[14] Tianqi Chen, Tong He, Michael Benesty, Vadim Khotilovich, and Yuan Tang. 2015. Xgboost: extreme gradient boosting. *R package version 0.4-2* 1, 4 (2015), 1–4.

[15] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems* (Boston, MA, USA) *(DLRS 2016)*. Association for Computing Machinery, New York, NY, USA, 7–10. https://doi.org/10.1145/2988450.2988454

[16] Mallory E. Compton and Justin Bullock. 2015. State Unemployment Insurance Claims Errors.

[17] Alejandro De La Garza. 2020. States' Automated Systems Are Trapping Citizens in Bureaucratic Nightmares With Their Lives on the Line. https://time.com/5840609/algorithm-unemployment/

[18] David Freeman Engstrom, Daniel E. Ho, Catherine M. Sharkey, and Mariano-Florentino Cuéllar. 2020. Government by Algorithm: Artificial Intelligence in Federal Administrative Agencies. *SSRN Electronic Journal* (2020). https://doi.org/10.2139/ssrn.3551505

[19] Virginia Eubanks. 2018. *Automating Inequality: How High-Tech Tools Profile, Police, and Punish the Poor*. St. Martin's Press, New York, NY.

[20] Boli Fang, Miao Jiang, Pei-yi Cheng, Jerry Shen, and Yi Fang. 2020. Achieving Outcome Fairness in Machine Learning Models for Social Decision Problems. *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence* (2020), 444–450. https://doi.org/10.24963/ijcai.2020/62

[21] Boli Fang, Miao Jiang, and Jerry Shen. 2019. Achieving Fairness in Determining Medicaid Eligibility through Fairgroup Construction. *arXiv preprint arXiv:1906.00128* (2019).

[22] Helmut Farbmacher, Leander Löw, and Martin Spindler. 2020. An explainable attention network for fraud detection in claims management. *Journal of Econometrics* (2020).

[23] Nizar Grira, Michel Crucianu, and Nozha Boujemaa. 2004. Unsupervised and semi-supervised clustering: a brief survey. *A review of machine learning techniques for processing multimedia content* 1 (2004), 9–16.

[24] Gongde Guo, Hui Wang, David Bell, Yaxin Bi, and Kieran Greer. 2003. KNN model-based approach in classification. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, Vol. 2888. Springer, Berlin, Heidelberg, 986–996. https://doi.org/10.1007/978-3-540-39964-3_62

[25] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, Xiuqiang He, and Zhenhua Dong. 2018. Deepfm: An end-to-end wide & deep learning framework for CTR prediction. *arXiv preprint arXiv:1804.04950* (2018).

[26] Nikita Jain and Vishal Srivastava. 2013. Data mining techniques: a survey paper. *IJRET: International Journal of Research in Engineering and Technology* 2, 11 (2013), 116–119.

[27] Torben Beck Jørgensen and Barry Bozeman. 2007. Public Values: An Inventory. *Administration & Society* 39, 3 (May 2007), 354–381. https://doi.org/10.1177/0095399707300703

[28] Samina Khalid, Tehmina Khalil, and Shamila Nasreen. 2014. A survey of feature selection and feature extraction techniques in machine learning. In *2014 Science and Information Conference*. IEEE, 372–378.

[29] Eric R Kingson and Marianne Levin. 1984. Local administrative practice and AFDC error in Maryland. *Journal of Social Service Research* 7, 3 (1984), 41–57.

[30] David D Lewis. 1998. Naive (Bayes) at forty: The independence assumption in information retrieval. In *European conference on machine learning (Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence, Vol. 1398)*. Springer, Berlin, Heidelberg, 4–15.

[31] Elizabeth W Lindsey, Sharman Colosetti, Beth Roach, and John S Wodarski. 1989. Quality control and error reduction in the AFDC program: a review and synthesis of state strategies. *Administration in Social Work* 13, 2 (1989), 29–45.

[32] Michael Lipsky. 1980. *Street-Level Bureaucracy: Dilemmas of the Individual in Public Service*. Russell Sage Foundation, New York.

[33] Michael Lipsky. 1984. Bureaucratic disentitlement in social welfare programs. *Social Service Review* 58, 1 (1984), 3–27.

[34] Juan Liu, Eric Bier, Aaron Wilson, John Alexis Guerra-Gomez, Tomonori Honda, Kumar Sricharan, Leilani Gilpin, and Daniel Davies. 2016. Graph analysis for detecting fraud, waste, and abuse in healthcare data. *AI Magazine* 37, 2 (2016), 33–46.

[35] Weibo Liu, Zidong Wang, Xiaohui Liu, Nianyin Zeng, Yurong Liu, and Fuad E Alsaadi. 2017. A survey of deep neural network architectures and their applications. *Neurocomputing* 234 (2017), 11–26.

[36] Georg E Matt and Thomas D Cook. 1993. The war on fraud and error in the food stamp program: An evaluation of its effects in the Carter and Reagan administrations. *Evaluation review* 17, 1 (1993), 4–26.

[37] John Mendeloff. 1977. Welfare procedures and error rates: An alternative perspective. *Policy Analysis* 3, 3 (1977), 357–374.

[38] Tina Nabatchi. 2018. Public Values Frames in Administration and Governance. *Perspectives on Public Management and Governance* 1, 1 (Feb. 2018), 59–72. https://doi.org/10.1093/ppmgov/gvx009

[39] Alexey Natekin and Alois Knoll. 2013. Gradient boosting machines, a tutorial. *Frontiers in neurorobotics* 7 (2013), 21.

[40] FBI National Press Office. 2020. FBI Sees Spike in Fraudulent Unemployment Insurance Claims Filed Using Stolen Identities. https://www.fbi.gov/news/pressrel/press-releases/fbi-sees-spike-in-fraudulent-unemployment-insurance-claims-filed-using-stolen-identities

[41] John Pallasch. 2020. *Addressing Fraud in the Unemployment Insurance (UI) System and Providing States with Funding to Assist with Efforts to Prevent and Detect Fraud and Identity Theft and Recover Fraud Overpayments in the Pandemic Unemployment Assistance (PUA) and Pandemic Emergency Unemployment Compensation (PEUC) Programs*. Memo. Employment and Training Administration Advisory System, Washington, D.C. 17 pages. https://wdr.doleta.gov/directives/attach/UIPL/UIPL_28-20.pdf

[42] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. 2018. CatBoost: unbiased boosting with categorical features. In *Advances in neural information processing systems*. 6638–6648.

[43] Anne Schneider and Helen Ingram. 1993. Social construction of target populations: Implications for politics and policy. *American political science review* 87, 2 (1993), 334–347.

[44] Andrew D. Selbst and Solon Barocas. 2018. *The Intuitive Appeal of Explainable Machines*. SSRN Scholarly Paper ID 3126971. Social Science Research Network, Rochester, NY. https://papers.ssrn.com/abstract=3126971

[45] Herbert A. Simon. 1997. *Administrative Behavior* (4 ed.). Simon & Schuster, New York.

[46] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. 2017. Deep & cross network for ad click predictions. In *Proceedings of the ADKDD'17*. Association for Computing Machinery, New York, NY, USA, 1–7. https://doi.org/10.1145/3124749.3124754

[47] John Wrafter. 1984. QC - Abbreviation for Failure - it Started as a Good Idea. *Public Welfare* 42, 4 (1984), 14.

[48] Matthew M Young, Justin B Bullock, and Jesse D Lecy. 2019. Artificial discretion as a tool of governance: a framework for understanding the impact of artificial intelligence on public administration. *Perspectives on Public Management and Governance* 2, 4 (2019), 301–313.

[49] Matthew M Young, Johannes Himmelreich, Justin B Bullock, and Kyoung-Cheol Kim. 2021. Artificial Intelligence and Administrative Evil. *Perspectives on Public Management and Governance* (Apr 2021). https://doi.org/10.1093/ppmgov/gvab006