Using Anomaly Scores from Unlabeled Data to Improve Supervised Risk Estimation[*]

Patrick Vossler[†] **Jacob Goldin**[‡] **Daniel E. Ho**[§]

Abstract

Firms and governments increasingly rely on supervised learning algorithms to detect fraud and allocate enforcement resources, but a lack of labeled data can hamper the performance of such algorithms. We propose a method for augmenting supervised learning models with information from unlabeled data through dependency-based anomaly scores. We demonstrate the effectiveness of this method (anomaly-based unlabeled residual augmentation or AURA) through a case study of the audit selection challenge faced by tax authorities like the Internal Revenue Service (IRS). We apply AURA to deidentified tax returns claiming business expenses and evaluate performance using randomly selected audits. We find AURA increases audit-detected underreporting relative to a supervised model baseline by an average of 8% per audit across categories of business expenses and 3.48% per audit for business expense categories considered by the IRS to exhibit the highest risk for misreporting.

1 Introduction

Semi-supervised learning (SSL) algorithms have been used in a variety of different settings to address a common data challenge: limited labeled data and massive amounts of unlabeled data. In the public sector, this challenge is particularly acute as shrinking operating budgets decrease the resources available for collecting labeled data necessary for meeting operating responsibilities. For example, many administrative enforcement agencies in the U.S. face the challenge of allocating scarce resources for auditing regulatory noncompliance (Giles, 2022; Wang et al., 2022; Madaio et al., 2016; Gernand, 2014; Bauguess, 2017; Hino et al., 2018; Johnson et al., 2023).

One way to address the audit allocation problem is to use predictions from machine learning (ML) models trained on the outcomes of previous audits to find cases to prioritize. However, ML models often require large amounts of data to produce accurate predictions, hampering their adoption in regulatory settings where resources are limited and audits are resource intensive. In this paper, we focus on leveraging the information in large amounts of unlabeled data to improve the performance of supervised models trained on limited amounts of labeled data.

We propose a model augmentation method for regression problems, *Anomaly-based Unlabeled Residual Augmentation* (AURA), that augments supervised learning models with information from unlabeled data through dependency-based anomaly scores. Unlike traditional proximity-based anomaly scores, which use proximity to other observations (Chandola et al., 2009) to measure anomalousness, dependency-based anomaly scores provide a concise signal of noncompliance by measuring the anomalousness of an observation based on the relationships between features. In risk estimation settings such as tax auditing, relationships between line items on a return may provide a stronger signal of fraud than the magnitude of observed values. This is because, while tax returns with large incomes exist in a sparse neighborhood relative to most tax returns,

[*] The findings, interpretations, and conclusions expressed in this paper are entirely those of the authors and do not necessarily reflect the views or the official positions of the U.S. Department of Treasury or the Internal Revenue Service. All results have been reviewed to ensure that no confidential information is disclosed. We thank Andrew Johns, Brandon Anderson, John Guyton, and Anne Herlache for helpful comments. We are grateful for financial support from Arnold Ventures.

[†] Stanford University

[‡] University of Chicago

[§] Stanford University

that fact alone does not imply tax noncompliance. Instead, the relationships between features provide more information about the anomalousness of an observation relative to the population.

We demonstrate the effectiveness of our proposed method through a case study motivated by the audit selection problem faced by the Internal Revenue Service (IRS) and evaluated using real (deidentified) tax returns. The problem we consider is this: taxpayers can deduct certain expenses on their tax return to reduce their income tax liability, and the government chooses a subset of those tax returns to audit, based on some predicted measure of noncompliance typically generated by a statistical model. Training these statistical models requires data with ground truth information about audit outcomes. But due to shrinking IRS budgets, the quantity of randomly selected research audits and the labels they generate has declined dramatically over the last decade. This problem is particularly acute for complex tax returns that are less frequently audited; in such cases, the IRS has considerably less data to use to train risk assessment models. In its 2023 strategic plan, for example, the IRS reported that the 2019 audit rate for partnerships was 0.05% (Internal Revenue Service, 2023).

Focusing on business expense audits of randomly selected returns, and using the features and data available to IRS, we study whether AURA increases the amount of detected tax underreporting relative to baseline methods for selecting which returns to audit. We compare AURA to a supervised model trained only on labeled data and find that the risk predictions of AURA lead to an 8% average increase in per audit adjustments when considering all business expense line items and a 3.48% average increase in per audit adjustments when considering the most commonly misreported business expenses.

To explore the mechanisms behind AURA's improvement in performance, we conduct several analyses. First, we test whether the improvement in performance comes from leveraging the unlabeled data or from a more efficient representation of the features. To do this, we compare AURA to a model augmented with dependency-based anomaly scores trained only on the labeled data. Relative to this baseline, we find that AURA leads to greater adjustments for a majority of business expense line items, suggesting that AURA leverages information from unlabeled data.

Second, we compare the performance of AURA for a range of unaudited sample sizes to better understand the amount of unlabeled data AURA needs to outperform baseline methods. We find that the average difference in adjustment increases with the size of the unlabeled data set. Third, we vary the amount of labeled training data available during training. Here we find that the signal provided by the unlabeled data is particularly valuable in settings with limited labeled data and that the benefit of the anomaly scores decreases as the amount of labeled data increases.

Finally, we show that the type of anomaly score method is important by comparing supervised models augmented with dependency-based anomaly scores and supervised models augmented with classical proximity-based anomaly scores. We find that the dependency-based anomaly scores outperform proximity-based anomaly scores in the tax audit setting, where information about the relationships between features is particularly important for identifying noncompliance.

2 Related Work

Our work combines ideas from both the semi-supervised learning (SSL) literature and the anomaly detection literature. AURA extends these two areas by using anomaly scores learned from unlabeled data to augment supervised models, enabling improved performance in risk estimation settings with limited labeled data but abundant unlabeled data.

Semi-supervised learning methods aim to improve model performance by leveraging both labeled and unlabeled data. There are two primary approaches in SSL: transductive and inductive. Transductive semi-supervised learning treats unlabeled examples as the test data that need to be predicted. On the other hand, inductive semi-supervised learning involves using labeled data and unlabeled data to learn a model that performs well on unseen test examples. We focus on the inductive paradigm, as it best matches the risk-assessment setting where we want a function that generalizes well beyond the observed data.

Most classical SSL methods have focused on classification problems, with the exception of regression co-training methods (Zhou and Li, 2005; Abdel Hady et al., 2009). One drawback with these co-training methods is that they rely on clustering methods such as k-NN regression and do not scale well beyond low-dimensional data. See van Engelen and Hoos (2020) for a recent review of these classical methods and their effectiveness. More recent work has shown promising results in using SSL methods to improve the performance of deep neural networks (DNN). Numerous methods have been proposed to incorporate unlabeled data to improve DNN performance on image recognition and other vision tasks (Yang et al., 2021). These methods typically focus on classification problems and rely on a cluster assumption that is not necessarily applicable to regression problems (Chapelle and Zien, 2005).

Another line of SSL work has focused on improving kernel-based and metric-based regression methods by leveraging unlabeled data (Wasserman and Lafferty, 2007; Brouard et al., 2011; Niyogi, 2013; Xu et al., 2022) These methods assume that data points close together in the feature space should have similar output values and that the data lies on a lower-dimensional manifold. Similarly, other deep learning SSL methods for regression tasks propose adding an additional term to the loss function that is a function of the unlabeled data (Jean et al., 2018; Olmschenk et al., 2018; He et al., 2022).

AURA differs from the above SSL approaches which directly include unlabeled observations in the learning objective. Instead, AURA distills information from the unlabeled data into anomaly scores that capture how anomalous each data point is relative to typical patterns in the data. This provides the supervised model with a concise signal of the "anomalousness" of each point. Incorporating anomaly scores is particularly well-suited for risk estimation, where anomalous or unusual data points often indicate higher risk. However, the type of anomaly score is important, as different definitions of anomalies make different assumptions about the data.

Most anomaly detection methods define anomalies based on distance or density – if a point is far from others or in a sparse region, it is considered anomalous. Distance-based methods use the distance to a point's k-nearest neighbors to measure anomalousness (Ramaswamy et al., 2000; Angiulli and Pizzuti, 2005). Density-based methods compare the density around a point to the density around its neighbors (Breunig et al., 2000). Clustering-based methods assume normal points cluster together while, anomalies do not belong to clusters or are in small or fringe clusters (Chandola et al., 2009).

In contrast, another class of anomaly detection methods defines anomalies as points that deviate from normal feature dependencies or correlations in the data (Noto et al., 2012; Paulheim and Meusel, 2015; Lu et al., 2020a;b). These methods aim to learn the typical inter-feature relationships present in the majority of the data. Anomalies are then identified as points that do not conform to these usual feature relationships, even if they are not isolated from other points in the feature space. The anomaly score is commonly calculated based on the difference between a feature's observed value and its expected value as predicted by the other features.

AURA incorporates the ideas behind this latter class of dependency-based anomaly detection methods. We hypothesize that unusual feature relationships are a stronger signal of risk than distance-based outliers in risk estimation settings like tax auditing. By augmenting a supervised model with dependency-based anomaly scores learned from a large unlabeled dataset, AURA provides the model with information about anomalous patterns that are more likely to indicate noncompliance. To the best of our knowledge, AURA is the first SSL method to utilize anomaly scores to represent information from unlabeled data.

In summary, AURA bridges semi-supervised learning and anomaly detection, using ideas from both areas to enable learning from limited labeled data and extensive unlabeled data. While SSL methods traditionally directly combine labeled and unlabeled data to minimize a given learning objective, AURA uses anomaly scores, from a model trained on unlabeled data, to flexibly represent information about typical and atypical data patterns in an unsupervised manner. In particular, AURA employs dependency-based anomaly detection to capture unusual feature relationships that are particularly relevant for risk estimation. This novel combination of ideas allows AURA to improve supervised models in the common setting where labels are scarce but unlabeled data is plentiful.

3 Preliminaries

In this section we specify the notation used throughout the rest of the paper as well as formalize the audit selection problem we study in our empirical application in Section 5.

3.1 Notation

We denote vectors with boldface lowercase characters and matrices with boldface uppercase characters. Scalar values and functions are written as plain lowercase characters. Sets of functions are denoted as plain uppercase characters. The notation $\boldsymbol{x}_j$ denotes the value of the jth index of the vector $\boldsymbol{x}$, similarly $\boldsymbol{X}_{ij}$ denotes the value at row i and column j of the matrix $\boldsymbol{X}$. We refer to columns and features interchangeably. We use the notation $\boldsymbol{X}_{-j}$ to represent a matrix containing all columns except the jth column. We write estimates as $\hat{\boldsymbol{x}}$.

Throughout this paper we assume access to a set of n labeled observations and m unlabeled observations. The labeled observations are indexed by $i \in \{1, \ldots, n\}$ and contain d-dimensional features $\boldsymbol{x} \in \mathbb{R}^d$ and scalar responses $y \in \mathbb{R}$ while the set of m unlabeled observations, indexed by $j \in \{n+1, \ldots, n+m\}$, have only features $\boldsymbol{z} \in \mathbb{R}^d$. We use $\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{y}$ to represent the aggregated features and responses such that $\boldsymbol{X} \in \mathbb{R}^{n \times d}$, $\boldsymbol{Z} \in \mathbb{R}^{m \times d}$, and $\boldsymbol{y} \in \mathbb{R}^n$.

3.2 Audit selection problem

For every line item on a tax return, indexed by $j \in \{1, \ldots, d\}$, a taxpayer reports a value $\tilde{\boldsymbol{x}}_j$, which may be different from the true value of that line item $\boldsymbol{x}_j$. Let $\boldsymbol{\delta} \in \mathbb{R}$ be a vector of adjustment amounts such that $\boldsymbol{\delta}_j = \boldsymbol{d}_j(\boldsymbol{x}_j - \tilde{\boldsymbol{x}}_j)$ is the adjustment amount for the j line item and $\boldsymbol{d}_j$ describes the direction of the adjustment, with a value of $+1$ for a line item corresponding to a profit and -1 for a line item corresponding to an expense or deduction.

Audit selection is concerned with selecting a subset of tax returns to audit, given a budget constraint. We use K to represent a fixed percentage of the population that can be audited and let $\boldsymbol{a} \in \{0, 1\}^n$ be a vector denoting the audit status of each taxpayer. Under this fixed budget, various objectives may align with different auditing priorities; however, for this paper, we focus on selecting audits with the goal of maximizing revenue. If the optimal choice of $\boldsymbol{a}$ is known, then, for a given line item j, the total revenue is given by $\sum_{i=1}^{n} a_i \boldsymbol{\Delta}_{ij}$, where $\boldsymbol{\Delta}$ is the matrix containing the n adjustment amount vectors. In practice, our goal is to choose a set of tax returns to audit $\boldsymbol{a}$ to maximize this value. The audit selection problem can be formalized as

$$\max_{\boldsymbol{a}} \sum_{i=1}^{n} \boldsymbol{a}_i \boldsymbol{\Delta}_{ij} \quad \text{s.t.} \quad \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{a}_i < K. \tag{1}$$

We wish to learn a mapping $f \colon \mathbb{R}^d \to \mathbb{R}$ that maps from a d-dimensional feature vector $\boldsymbol{x}$ to a scalar value that represents the risk of noncompliance for line item j that can be compared to the ground truth adjustment value $\boldsymbol{\delta}_j$. Accurate predictions of the risk of noncompliance $\hat{\boldsymbol{\delta}}_j$ are critical for inducing a ranking of the observations that maximizes the objective of (1) and results in a greater amount of audit adjustment revenue.

3.2.1 Evaluation Metrics

We use two primary metrics for comparing models in terms of their ability to select audits with the goal of maximizing revenue subject to a budget constraint.

Trajectory. The trajectory metric provides a way to compare the performance of models that directly takes into account the auditing budget K by first ranking tax returns according to the magnitude of the predicted adjustment $\hat{\boldsymbol{\delta}}_j$ and then selecting the top K returns. More formally, we select the taxpayers to audit according to the formula,

$$\hat{\boldsymbol{a}} = \arg\max_{\boldsymbol{a}} \sum_{i=1}^{n} \boldsymbol{a}_i \hat{\boldsymbol{\Delta}}_{ij} \quad \text{s.t.} \quad \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{a}_i \leq K. \tag{2}$$

Then, we can calculate the trajectory value for a given audit budget K:

$$t_K = \sum_i \hat{\boldsymbol{a}}_i \hat{\boldsymbol{\Delta}}_{ij}.$$

We report trajectory values for different values of K, averaged over the five cross-validation folds with a focus on the case where $K = 1.5\%$ as it is similar to the audit rate during our sample period of returns filed with at least one Schedule C.

Difference of Trajectories. Instead of comparing the methods based on their average trajectory, we can compare the performance of the methods within each fold. To compare two different methods, we take the difference of their trajectories for each audit budget amount K. So, for a comparison model C and a baseline model B, we define the difference of trajectories metric as the difference of their trajectories at a given audit budget K:

$$\gamma_K^{(C,B)} = t_K^C - t_K^B.$$

4 Anomaly-based Unlabeled Residual Augmentation

We introduce *feature internal consistency augmentation* (FICA) for prediction problems with abundant unlabeled data but limited labeled data. For each feature j, we learn a function $g_j : \mathbb{R}^{d-1} \to \mathbb{R}$ by training a dependency-based anomaly score model that uses the unlabeled data $\boldsymbol{z}_{-j}$ to predict the values of $\boldsymbol{z}_j$. Next, we construct a vector of anomaly scores for the labeled data as $\tilde{\boldsymbol{x}} = \boldsymbol{x}_j - g_j(\boldsymbol{x}_{-j})$, where the anomaly score is the residual error from predicting the value of $\boldsymbol{x}_j$ using the remaining features $\boldsymbol{x}_{-j}$. We collect the anomaly score vectors into a matrix $\tilde{\boldsymbol{X}}$ and create an augmented labeled data matrix $\check{\boldsymbol{X}} = (\boldsymbol{X}, \tilde{\boldsymbol{X}}) \in \mathbb{R}^{n \times 2d}$. The augmented labeled data matrix is then used to train a flexible regression model f to predict audit adjustments or other quantities of interest.

We augment the labeled data with dependency-based anomaly scores as opposed to proximity-based anomaly scores because the assumptions underlying dependency-based anomaly scores better reflect the structure of the data in risk assessment settings. Proximity-based anomaly score methods assume that observations cluster together based on feature values and that observations with larger than typical values are anomalous. Dependency-based anomaly score methods rely on the assumption that deviations from typical relationships between features are an important signal of the anomalousness of an observation. We hypothesize that the inter-feature relationship of dependency-based anomaly scores methods is a more accurate reflection of risk assessment settings. For example, in the tax noncompliance setting, relationships between each of the line item amounts reported in a return are treated as a more reliable indicator of noncompliance than the actual magnitude of the line item itself. Thus, learning such relationships between features from unlabeled data and distilling the deviation of an observation's values from the expected value of the model trained on the unlabeled data can provide relevant information for identifying returns worth investigating.

5 Application: Reported business expenses

Taxpayers report business income from sole proprietorships and small businesses on Schedule C of their tax returns, like most single-owner LLCs. In prior random audit studies, the IRS has found that the income and expenses claimed on Schedule C tend to be misreported at particularly high rates. In one study covering tax years 2013 to 2015, the IRS found that an estimated 76% of sole proprietors misreported their total expenses, for a net misreported amount of $92 billion on average per year.[1] One factor contributing to these

[1] Misreporting can occur intentionally or unintentionally and may be attributable to the taxpayer or tax preparer.

high rates of Schedule C misreporting is that, unlike for parts of the tax return, the IRS has limited visibility and fewer (non-audit) methods to validate the accuracy of business income and expenses that the taxpayer reports, and taxpayers may be more inclined to misreport income in the absence of third-party information reporting compared to when such reporting is extensive (U.S. Government Accountability Office, 2020).

In this section, we consider the problem of predicting Schedule C expense misreporting by taxpayers. When taxpayers overstate a business expense, it lowers their (net) taxable income and resulting tax liability. There are 23 categories of business expenses listed on Schedule C. Some of the top estimated misreported expenses included car and truck expenses (58%), utilities (57%), travel (56%), and expenses for business use of home (53%).

The analysis in this section serves as proof-of-concept for our proposed methodological approach and illustrates the potential for AURA to significantly improve predictive accuracy in a high-stakes realistic setting.

In what follows, we first describe the tax exam data used in our evaluation, our training procedure, and the evaluation metrics. Then, we show that AURA consistently outperforms a baseline approach across the different expense line items before exploring the mechanisms for our method's improved performance with one of the most consistently misreported expense line items, business use of home.

5.1 Data

For our labeled data set, we use audited tax returns from the IRS's National Research Program (NRP). The NRP is a research program used by the IRS to better understand tax compliance. Each year the NRP forms stratified random samples of the taxpayer population to audit. Most IRS audits are narrow in scope and focus on specific issues with a tax return. However, because NRP audits seek to estimate the correctness of the whole return, they are more exhaustive and examine nearly all line items on a return. NRP samples contain 15,000 tax returns each year, however the size of these samples has been decreasing in recent years and there is pressure to reduce it further (Marr and Murray, 2016; Congressional Budget Office, 2020).

For our unlabeled data set, we take a random sample from the population of unaudited tax returns. Both data sets are limited to tax returns for years 2009 to 2014 that contain at least one Schedule C and did not file other common schedules such as a Schedule E (supplemental income and loss) or Schedule F (farm income). The final unlabeled data set contains 1,343,543 tax returns. In contrast, the final labeled data set from the NRP contains 35,456 tax returns and audit results.

Within each return we consider all expense line item values from the Schedule C that ask taxpayers to report continuous values not derived from other line items on the form. We use the reported values $\tilde{\boldsymbol{x}}_j$ as continuous features and take the adjustment values $\boldsymbol{\delta}_j$ as response variables.

5.2 Training Procedure

Our supervised baseline model is an XGBoost tree trained on the labeled data and the unsupervised baseline model is an isolation forest (Liu et al., 2008) trained on the unlabeled data. We also use XGBoost for the supervised model in AURA and to generate the dependency-based anomaly scores for each of the expense line items from the unlabeled data. To evaluate the performance of our method on the labeled data, we perform a nested five-fold cross-validation scheme. We split the data into five outer folds for the outer cross-validation, holding out each outer fold as a validation set. Then, to mirror the limited amount of labeled data available for less commonly audited line items, we take a baseline subsample ($n = 1,418$) of the remaining four outer folds to use as our training data. We perform a hyperparameter grid search on the baseline subsample of the outer folds with an inner five-fold cross-validation and train a model with the optimal hyperparameters on the subsample. Finally, we use this model to make predictions for the validation set from the outer cross-validation step. For each held-out fold of the outer cross-validation scheme, we repeat the subsampling process with 50 subsamples. The evaluation metrics are then averaged across the subsamples. For further details on the training procedure as well as a diagram of the nested cross-validation procedure, see Appendix A.

5.3 Results across expense line items

While improving prediction of misreporting for expense line items such as car and truck expenses, travel expenses, and wage expenses are particularly important given their high rates of misreporting, we first compare AURA to a supervised baseline for all Schedule C expense line items[2]. We show that AURA provides both a substantial increase in overall adjustments for a fixed audit budget as well as a significant improvement in per audit adjustment amounts.

Figure 1 shows the average difference in adjustment amount for AURA and a supervised model trained on the labeled data for each of the Schedule C expense line items[3]. For all 16 expense line items we observe a positive average difference between AURA and the supervised baseline with a 1.5% audit budget. AURA provides an increase in adjustment amount identified compared to the supervised baseline that is statistically significant for 12 out of the 16 expense line items and for three of the four expense line items with the highest misreporting rates according to the IRS.

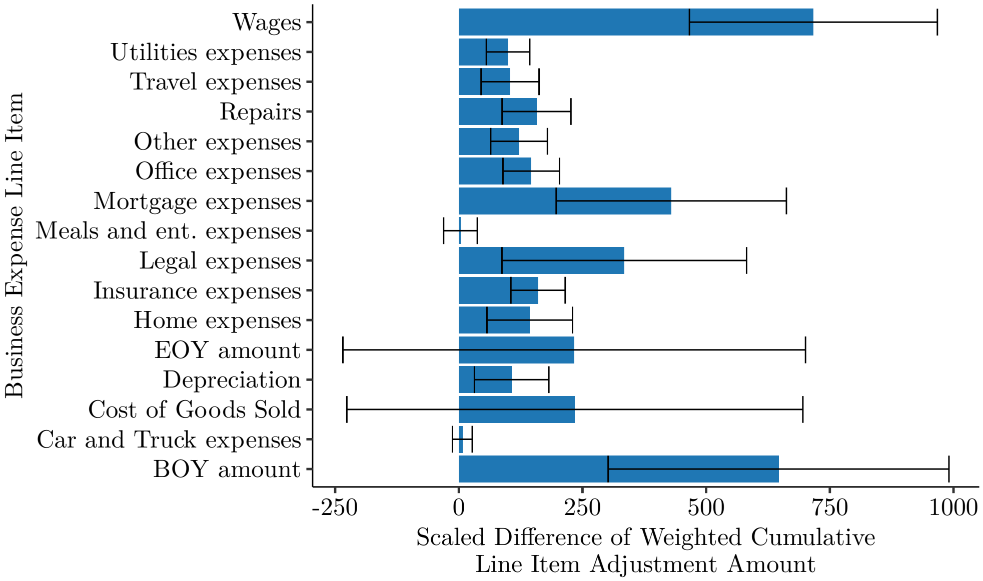

Figure 1: The average difference in adjustment amount between AURA and a supervised model trained only on labeled data, scaled by the average adjustment amount for each expense line item, for an audit budget of 1.5% with 95% confidence intervals. Positive values indicate that AURA outperforms the supervised baseline for the given expense line item, while negative values (not observed here) would indicate that the supervised baseline outperforms AURA.

Table 1 quantifies the difference in performance between AURA and these baseline methods on a per audit basis by estimating the average per audit dollar difference in audit adjustment amount. We find that AURA leads to an 8% average improvement in per audit adjustment amounts compared to the supervised baseline. For 12 out of the 16 expense line items there is a statistically significant difference in per audit adjustment amount and all 12 have per audit improvements greater than $100.

We repeat these analyses with an unsupervised baseline model trained only on the unlabeled data in Appendix E. We observe an even larger performance gain of AURA relative to a unsupervised baseline. AURA provides statistically significant increases in adjustment amount for all 16 of the expense line items compared to the unsupervised baseline. Furthermore, AURA's predictions lead to an average increase in per audit ad-

[2] Our source of unlabeled data provides information for 16 of the 23 expense line items reported on the Schedule C. We provide a full list of the line items used in our analysis in Appendix A

[3] We provide descriptions of each of the expense line items in Appendix B

justments of 721% when considering all business expenses and an average increase in per audit adjustments of 102% for the top misreported business expenses.

The consistent improvement of AURA compared to baseline methods, both in terms of overall adjustments and on a per audit basis, motivate us to determine what is the predominant source of this improvement.

Business Expense Category	AURA Relative Gain ($)	Supervised Adjustment Amount ($)	Percent Difference	P-value
Beginning of year inventory amount	770	3,600	21.40%	<0.001
Car and truck expenses	59	12,739	0.47%	0.492
Cost of goods sold	2,274	41,473	6.69%	0.322
Depreciation	222	5,288	4.19%	0.005
End of year inventory amount	351	4,648	7.56%	0.329
Home expenses	136	3,049	4.48%	0.001
Insurance expenses	165	1,856	8.87%	<0.001
Legal expenses	251	2,526	9.95%	0.008
Meals and entertainment expenses	4	3,086	0.12%	0.850
Mortgage expenses	165	2,215	7.47%	<0.001
Office expenses	155	2,627	5.91%	<0.001
Other expenses	1,095	12,820	8.54%	<0.001
Repair expenses	230	4,049	5.68%	<0.001
Travel expenses	165	4,424	3.73%	<0.001
Utilities expenses	165	3,136	5.27%	<0.001
Wages	416	1,429	29.10%	<0.001

Table 1: For each expense line item we report the estimated gain of AURA relative to the baseline supervised estimation method, the per audit adjustment amount for the baseline supervised estimation method, the difference between AURA and the supervised baseline in percentage terms, and the p-values from a one-sample t-test. The results are averaged over the results from the 50 subsamples taken in each of the five folds of the cross-validation procedure. See Section 5.2 for additional details.

5.4 Understanding AURA's improved performance

In this section we investigate the mechanisms driving AURA's improved performance relative to the pure supervised and unsupervised baselines. To do so, we focus on one of the line items for which we saw substantial gains from AURA: the business use of home expense line item. In this subsection we perform several experiments to better understand the source of these performance gains.

Sensitivity to size of audit budget. We explore whether AURA's improvement over the supervised baseline is sensitive to the size of the audit budget. Figure 2 compares the performance of AURA and the supervised baseline according to the two metrics described in Section 3.2.1. The plot on the left shows the trajectory metric for the two methods for a range of audit budgets. We see that AURA's predictions dominate the supervised baseline in their ability to choose exams with larger adjustment amounts for a range of audit budgets. The plot on the right shows the difference of the trajectories metric (dark blue line) for the two methods and the 95% confidence interval (blue shaded area) for different audit budgets. The red horizontal line at zero is included for reference, since a positive value means that AURA has a larger average within-fold adjustment value than the supervised model. There is a positive average difference in adjustment amounts across all audit budgets, meaning that at each audit budget, AURA's predictions resulted in a larger average adjustment amount within each cross-validation fold. These results suggest that AURA's improvement over the supervised baseline is stable over a range of audit budgets both smaller and larger than the than typical Schedule C audit budget,

Is AURA using the unlabeled data or providing a better representation of the features? While it appears that AURA's improved performance is consistent over a range of audit budgets, it is not clear whether the improve is driven by inclusion of the unlabeled data or the anomaly score representation of the features.

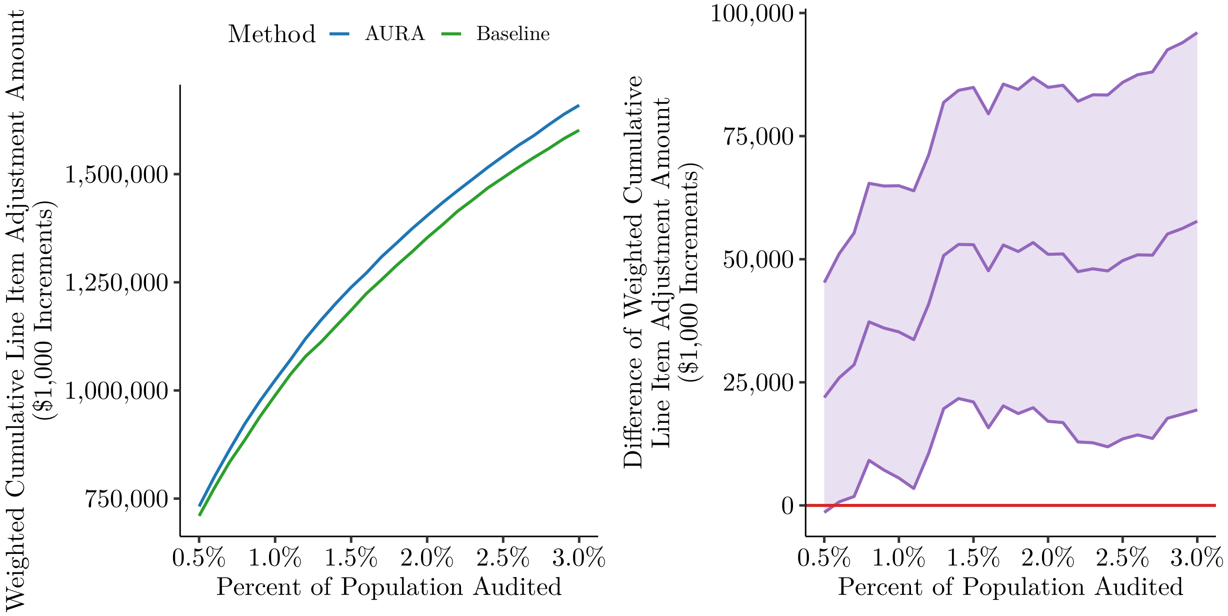

Figure 2: Two plots comparing the performance of AURA and a supervised model trained only the labeled data averaged over five cross-validation folds. The plot on the left shows the amount of cumulative line item adjustment captured as the percentage of the population audited increases. The right plot shows the average difference in the weighted adjustment amount over five cross-validation folds. We report the average within-fold difference (dark blue line) between the two methods and the 95% confidence interval (blue shaded area) for a given budget amount.

We compare AURA to a modified version of AURA in which we first fit a dependency-based anomaly score model using the labeled data for each feature and then augment the labeled data with dependency-based anomaly scores from those anomaly score models. We refer to this modified version of AURA trained only on labeled data as AURA-L. We find evidence that the source of the data used to train the anomaly score model is important. Figure 3 shows that AURA consistently outperforms AURA-L at each of the audit budget amounts, including at the historical audit budget percentage of 1.5%. These results suggest that in the low data regime, the anomaly scores from a model trained on the unlabeled data provide a better signal for audit risk than anomaly scores from a model trained only on the labeled data.

Varying the amount of unlabeled data. Given that the evidence that AURA leverages the information from the anomaly score model trained on unlabeled data, we explore the role that the size of the unlabeled data plays in improving the performance of AURA over the supervised baseline. Specifically, we evaluate the performance of AURA with different amounts of unlabeled data used to train the anomaly score model by varying the size of the unlabeled data from 1,000 to 1 million. We repeat the training procedure for each unlabeled data size with 15 different subsamples of the unlabeled data. We report the mean and 95% confidence interval of the difference in the trajectories of AURA and the supervised model with a fixed audit budget of 1.5% in Figure 4. There is a positive average difference in trajectories even for a relatively small unlabeled data sample of 1,000, and the average difference in trajectories increases with the size of the unlabeled data.

Varying the amount of available labeled data. Compared to Schedule C, many other types of returns have less available labeled data to use to train a supervised model. There is a need to improve audit risk estimation for these limited data settings and using unlabeled data is an appealing option. We compare the performance of AURA to the supervised baseline in settings meant to mimic returns with limited labeled data by varying the size of the subsample taken during the cross-validation step. Figure 5 shows the average difference in adjustment amount for different fractions of the base sample size within each fold. The average

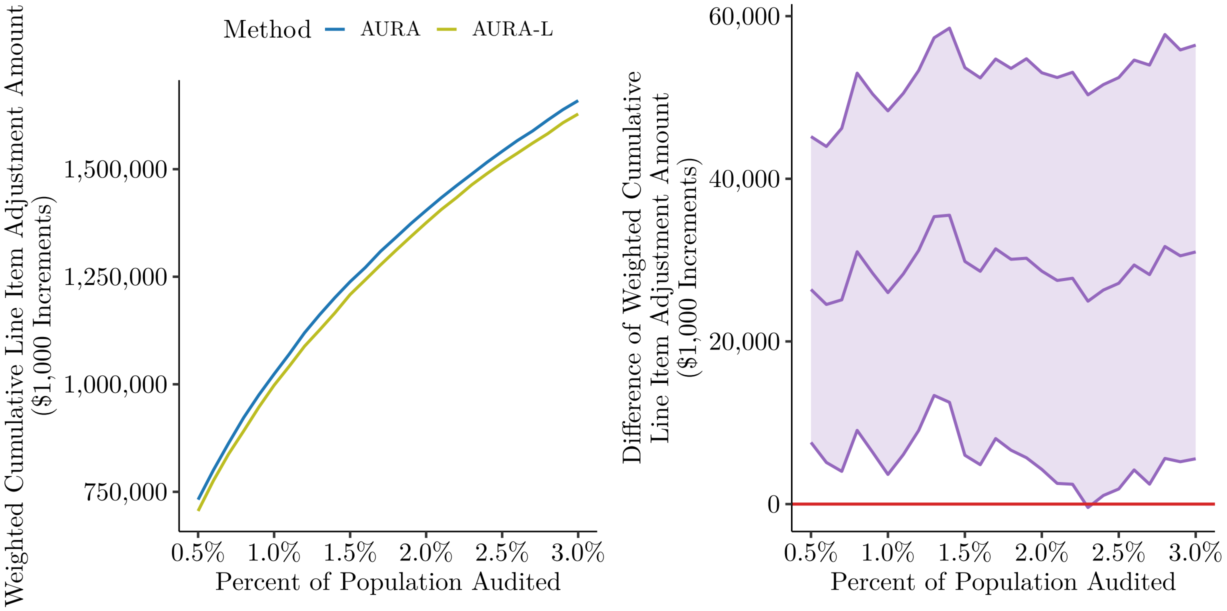

Figure 3: Two plots comparing the performance of AURA and AURA-L over five cross-validation folds. The plot on the left shows the amount of cumulative line item adjustment captured as the percentage of the population audited increases. The right plot shows the average difference in the weighted adjustment amount over five cross-validation folds.

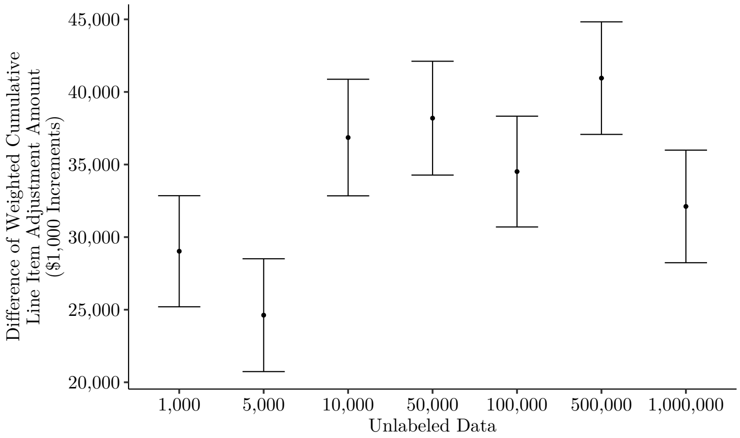

Figure 4: The average difference in adjustment amount captured between a supervised model augmented with dependency anomaly scores trained on different amounts of unlabeled data and a naive supervised model trained on the labeled data for a fixed audit budget of 1.5%. The error bars show the 95% confidence interval of the differences.

difference of trajectories is greatest for the smallest subsamples of the labeled data and that the difference in performance becomes less pronounced as the size of the labeled data available to both models. This suggests that AURA provides a way to make the most of the few audits available by supplementing them with information from the vast amount of unaudited returns.

Comparing types of anomaly scores. We next explore the effect of the type of anomaly score model on AURA's performance. We compare AURA with anomaly scores from a dependency-based model to a variation of AURA (AURA-P) that uses anomaly scores from a proximity-based model. The key differ-

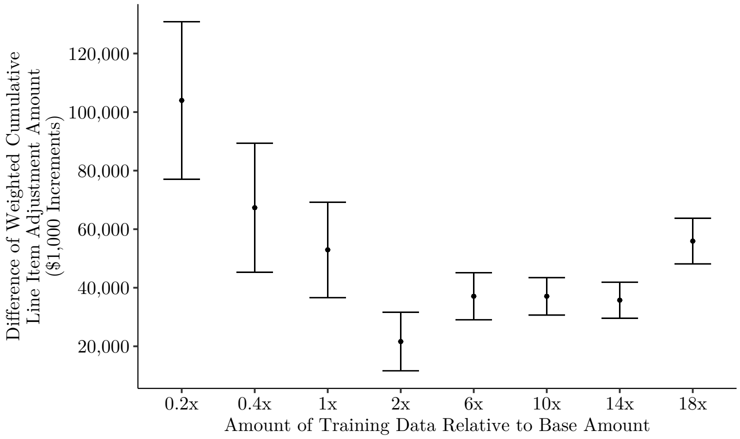

Figure 5: The average difference in adjustment amount captured between a supervised model augmented with dependency anomaly scores trained on the unlabeled data and a naive supervised model trained on the labeled data for a fixed audit budget of 1.5% for different amounts of labeled data. We report the amount of subsampled labeled data relative to the baseline subsample described in Section 5.2.

ence between these two anomaly detection models is that dependency-base anomaly detection models score observations base upon inter-feature relationships while proximity-based anomaly detection models score observations based upon proximity to other observations. We find that dependency-based anomaly scores appear better suited for the tax compliance setting. Figure 6 shows that AURA has a larger cumulative adjustment amount and that the average within-fold difference is positive across the range of audit budgets. In Appendix C.1 we perform this experiment across each of the different Schedule C expense line items and see a similar trend for the majority of expense line items.

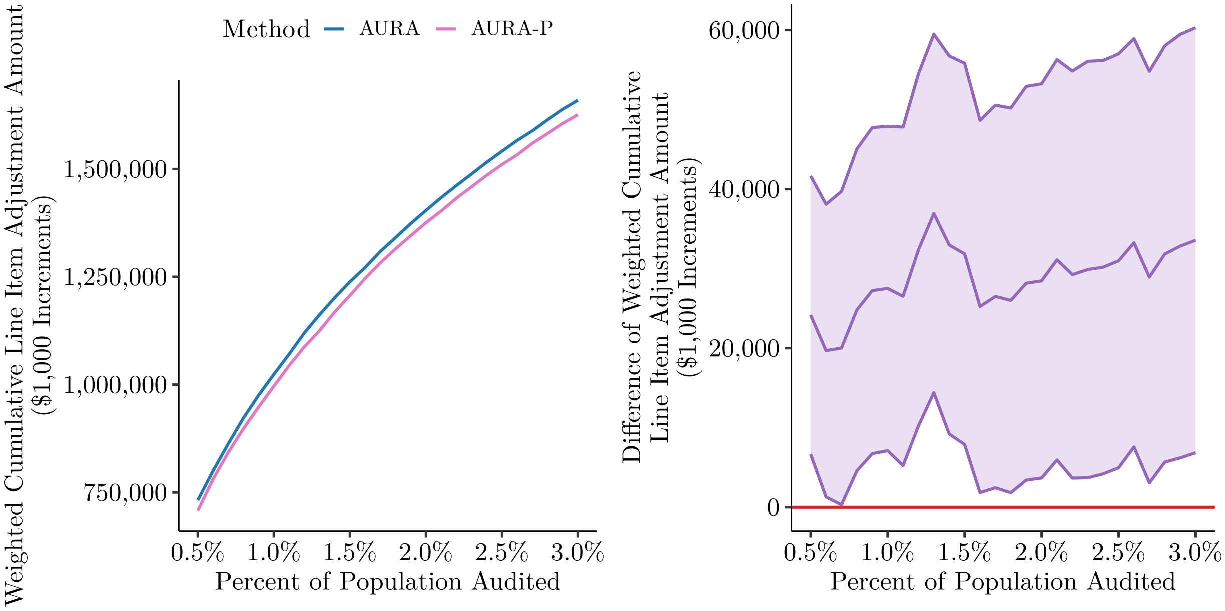

Figure 6: Two plots comparing the performance of AURA and AURA-P over five cross-validation folds. The plot on the left shows the amount of cumulative line item adjustment captured as the percentage of the population audited increases. The right plot shows the average difference in the weighted adjustment amount over five cross-validation folds.

Based on these experiments, we can draw several conclusions about the mechanisms driving the improved performance of AURA compared to the supervised and unsupervised baselines for predicting business expense misreporting on tax returns:

- AURA's improvement over the supervised baseline is consistent across a range of audit budgets, both smaller and larger than the typical Schedule C audit budget. This suggests the gain from AURA is robust to audit budget size.
- The source of data used to train the anomaly score model is important. AURA outperforms its modified counterpart that only uses the limited labeled data. This indicates AURA's gain comes from leveraging the information in the unlabeled data.
- The size of the unlabeled dataset impacts AURA's performance gain. The average difference in adjustments captured by AURA compared to the supervised baseline increases as more unlabeled data is used to train the anomaly model. However, even a relatively small amount of unlabeled data (1,000 returns) provides a benefit.
- AURA's performance gain is most pronounced when labeled data is very limited. As the amount of labeled data used to train both AURA and the supervised model increases, their performance difference decreases. This suggests AURA is particularly valuable in scenarios where audits and labels are scarce.
- The type of anomaly score model matters. Using AURA with dependency-based anomaly scores that capture unusual relationships between features, outperforms using AURA with proximity-based anomaly scores. This indicates that in the tax noncompliance setting, anomalous feature interactions are a stronger signal than proximity-based outliers.

In summary, these experiments provide evidence that AURA's improved performance stems from its ability to extract a signal of anomalousness from a large unlabeled dataset in the form of dependency-based anomaly scores. This augmentation approach is holds over a range of audit budgets, scales with the amount of unlabeled data, and is most impactful when labeled data is scarce - making it a promising method for tax auditing and other risk estimation applications.

6 Conclusion

In this paper, we have introduced Anomaly-based Unlabeled Residual Augmentation (AURA), a novel semi-supervised learning approach for improving risk estimation models in settings with abundant unlabeled data but limited labeled data. AURA augments supervised learning models with dependency-based anomaly scores learned from unlabeled data, providing the model with a concise signal of each observation's anomalousness relative to the broader population.

We demonstrated AURA's effectiveness through a case study of predicting business expense misreporting on tax returns, a critical challenge faced by the IRS. Using real tax data, we showed that AURA leads to an 8% increase on average in audit-detected underreporting on a per-audit basis compared to a supervised model trained only on labeled data.

Through a series of experiments, we provided evidence that AURA's improved performance stems from its ability to extract a signal of anomalousness from the unlabeled data in the form of dependency-based anomaly scores. We found that AURA's performance gain is robust across audit budgets, increases with the amount of unlabeled data, and is most impactful when labeled data is very scarce. Additionally, we showed that dependency-based anomaly scores, which capture unusual feature relationships, are better suited for the tax noncompliance setting than proximity-based anomaly scores.

Our results highlight the potential of AURA and semi-supervised learning approaches more broadly to improve risk estimation in domains with limited labeled data. By leveraging the unlabeled data as a benchmark of typical feature relationships, AURA can help resource-constrained organizations like the IRS make more efficient use of their auditing budgets and better combat noncompliance. More generally, our work underscores the value of developing machine learning methods that can bridge the gap between the vast amounts of unlabeled data available and the practical constraints of labeled data collection.

References

Abdel Hady, M. F., Schwenker, F. and Palm, G. (2009) Semi-supervised learning for regression with co-training by committee. In *Artificial Neural Networks ICANN 2009*, Lecture notes in computer science, 121–130. Berlin, Heidelberg: Springer Berlin Heidelberg. 2

Angiulli, F. and Pizzuti, C. (2005) Outlier mining in large high-dimensional data sets. *IEEE transactions on Knowledge and Data engineering*, **17**, 203–215. 2

Bauguess, S. W. (2017) The role of big data, machine learning, and AI in assessing risks: A regulatory perspective. *SSRN Electron. J.* 1

Breunig, M. M., Kriegel, H.-P., Ng, R. T. and Sander, J. (2000) Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 93–104. 2

Brouard, C., d'Alché Buc, F. and Szafranski, M. (2011) Semi-supervised penalized output kernel regression for link prediction. In *28th International Conference on Machine Learning (ICML 2011)*, 593–600. 2

Chandola, V., Banerjee, A. and Kumar, V. (2009) Anomaly detection: A survey. *ACM computing surveys (CSUR)*, **41**, 1–58. 1, 2

Chapelle, O. and Zien, A. (2005) Semi-supervised classification by low density separation. In *International workshop on artificial intelligence and statistics*, 57–64. PMLR. 2

Congressional Budget Office (2020) Trends in the internal revenue services funding and enforcement. *Tech. rep.* 5.1

van Engelen, J. E. and Hoos, H. H. (2020) A survey on semi-supervised learning. *Mach. Learn.*, **109**, 373–440. 2

Gernand, J. M. (2014) Machine learning classification models for more effective mine safety inspections. In *Volume 14: Emerging Technologies; Engineering Management, Safety, Ethics, Society, and Education; Materials: Genetics to Structures*, V014T08A020. American Society of Mechanical Engineers. 1

Giles, C. (2022) *Next Generation Compliance: Environmental Regulation for the Modern Era.* Oxford University PressNew York. 1

He, R., Tian, Z. and Zuo, M. J. (2022) A semi-supervised GAN method for RUL prediction using failure and suspension histories. *Mech. Syst. Signal Process.*, **168**, 108657. 2

Hino, M., Benami, E. and Brooks, N. (2018) Machine learning for environmental monitoring. *Nat. Sustain.*, **1**, 583–588. 1

Internal Revenue Service (2023) IRS inflation reduction act strategic operating plan. *Tech. rep.* 1

Jean, N., Xie, S. M. and Ermon, S. (2018) Semi-supervised deep kernel learning: Regression with unlabeled data by minimizing predictive variance. *arXiv [cs.LG].* 2

Johnson, M. S., Levine, D. I. and Toffel, M. W. (2023) Improving regulatory effectiveness through better targeting: Evidence from OSHA. *Am. Econ. J. Appl. Econ.*, **15**, 30–67. 1

Liu, F. T., Ting, K. M. and Zhou, Z.-H. (2008) Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining*, 413–422. IEEE. 5.2, E

Lu, S., Liu, L., Li, J., Le, T. D. and Liu, J. (2020a) Dependency-based anomaly detection: Framework, methods and benchmark. *arXiv preprint arXiv:2011.06716.* 2

— (2020b) Lopad: A local prediction approach to anomaly detection. In *Advances in Knowledge Discovery and Data Mining: 24th Pacific-Asia Conference, PAKDD 2020, Singapore, May 11–14, 2020, Proceedings, Part II 24*, 660–673. Springer. 2

Madaio, M., Chen, S.-T., Haimson, O. L., Zhang, W., Cheng, X., Hinds-Aldrich, M., Chau, D. H. and Dilkina, B. (2016) Firebird: Predicting fire risk and prioritizing fire inspections in atlanta. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM. 1

Marr, C. and Murray, C. (2016) IRS funding cuts compromise taxpayer service and weaken enforcement. *Tech. rep.*, Center on Budget and Policy Priorities. 5.1

Niyogi, P. (2013) Manifold regularization and semi-supervised learning: Some theoretical analyses. *J. Mach. Learn. Res.*, **14**, 1229–1250. 2

Noto, K., Brodley, C. and Slonim, D. (2012) Frac: a feature-modeling approach for semi-supervised and unsupervised anomaly detection. *Data mining and knowledge discovery*, **25**, 109–133. 2

Olmschenk, G., Zhu, Z. and Tang, H. (2018) Generalizing semi-supervised generative adversarial networks to regression using feature contrasting. *arXiv [cs.LG]*. 2

Paulheim, H. and Meusel, R. (2015) A decomposition of the outlier detection problem into a set of supervised learning problems. *Mach. Learn.*, **100**, 509–531. 2

Ramaswamy, S., Rastogi, R. and Shim, K. (2000) Efficient algorithms for mining outliers from large data sets. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 427–438. 2

U.S. Government Accountability Office (2020) Tax administration: Better coordination could improve IRS's use of third-party information reporting to help reduce the tax gap. *Tech. rep.* 5

Wang, X., Bouzembrak, Y., Lansink, A. O. and van der Fels-Klerx, H. J. (2022) Application of machine learning to the monitoring and prediction of food safety: A review. *Compr. Rev. Food Sci. Food Saf.*, **21**, 416–434. 1

Wasserman, L. and Lafferty, J. (2007) Statistical analysis of semi-supervised regression. *Advances in Neural Information Processing Systems*, **20**. 2

Xu, L., Hu, C. and Mei, K. (2022) Semi-supervised regression with manifold: A bayesian deep kernel learning approach. *Neurocomputing*, **497**, 76–85. 2

Yang, X., Song, Z., King, I. and Xu, Z. (2021) A survey on deep semi-supervised learning. *arXiv [cs.LG]*. 2

Zhou, Z.-H. and Li, M. (2005) Semi-supervised regression with co-training. In *Proceedings of the 19th international joint conference on Artificial intelligence*, IJCAI'05, 908–913. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. 2

A Training and evaluation procedure

As described in Section 5.2, the adjustments (i.e., our response variable) contain several high outliers that we want our models to capture well from a revenue-maximizing perspective, as these outliers correspond to large positive adjustments. However, these large adjustment values are fairly sparse, and thus any split of the data may contain different distributions in the upper quantiles; moreover, under our mean-squared error training objective (which is more sensitive to outliers that mean absolute error or other alternatives), the outliers may heavily influence model fit. We want our evaluation procedure to reflect any instability in the models that may arise due to the presence of these outliers, and thus we perform a nested five-fold cross-validation on the 80% training set to train five distinct models, each with validation sets that we can evaluate individually or compare to derive uncertainties in our evaluation metrics. To do so, we first randomly split the training data into five groups. For each of the five groups, we hold out the given group as a validation set and take the remaining four groups as a training set. Then, among the four training groups, we perform five-fold cross-validation to select the hyperparameters with the best mean score on the validation folds, performing a randomized grid search over the following set of parameters.

- Learning rate: [0.001, 0.01, 0.05, 0.1, 0.15, 0.2]
- Number of estimators: [1000, 1500, 2000]
- Maximum tree depth: [3, 4, 5, 6, 7, 8, 9 ,10]
- Subsample ratio of training instances: Draws from $U([0.5, 1])$
- Subsample ratio of columns: Draws from $U([0.4, 1])$

We also provide a visual depiction of the nested cross-validation scheme in Figure 7.

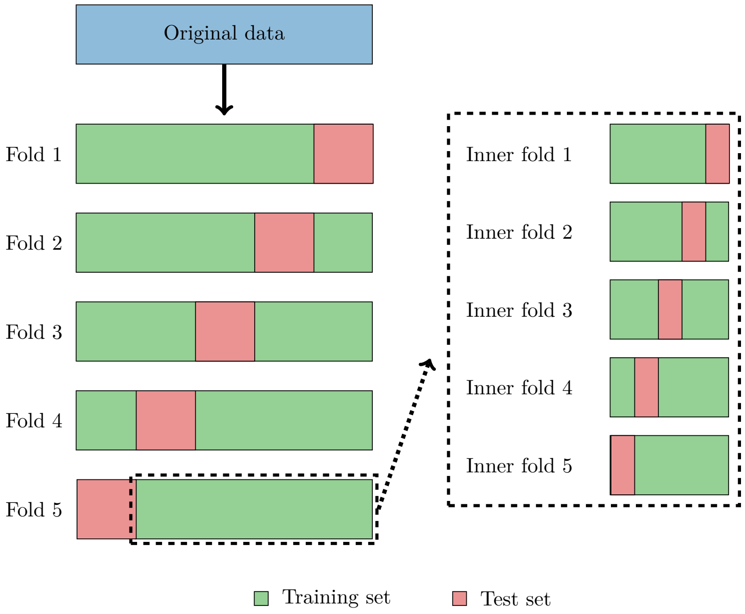

Figure 7: A diagram depicting a nested five-fold cross-validation procedure. The left-most column shows each of the five outer folds created from the full data set as separate rectangles. For each outer fold, we split the training set into five inner folds shown in the right-most column in the dashed box. The five inner folds are used to perform a randomized search of model hyperparameters. The hyperparameters with the highest validation score are then used to train the model on the training set and then make predictions for the held-out test set.

B Schedule C expense line item overview

In this section, we provide descriptions of each of the Schedule C expense line items.

- **BOY amount.** The amount of inventory a sole proprietorship has at the beginning of the year.
- **Car and truck expenses.** The amount of vehicle expenses incurred while running the sole proprietorship using either the actual expenses or a standard mileage rate. These expenses include expenses like fuel, insurance, and other fees.
- **Depreciation.** The amount of depreciation and Section 179 expense deduction. Section 179 of the Internal Revenue Code allows businesses to take an immediate deduction for business expenses related to depreciable assets such as equipment, vehicles, and software.
- **EOY amount.** The amount of inventory a sole proprietorship has at the beginning of the year.
- **Home expenses/Business use of home.** Sole proprietorships who regularly and exclusively use part of their home for work and business-related activities can write off rent, utilities, real estate taxes, repairs, maintenance and other related expenses.
- **Insurance expenses.** The amount of insurance expenses (other than health insurance) accrued while running a sole proprietorship including workers compensation insurance and general liability insurance.
- **Legal expenses.** The amount of short-term professional advice (including lawyers and accountants) related to specific business deals, sales transactions, or yearly taxes.
- **Meals and entertainment expenses.** The amount of expenses for ordinary and necessary meals and entertainment during travel or when meeting clients or business associates.
- **Office expenses.** The amount of office-related expenses from running a sole proprietorship including office supplies, postage, computers, and printers.
- **Repairs and maintenance.** Expenses related to general business place repairs and upkeep such as plumbing repairs, routine servicing for heating or air conditioning, painting, etc.
- **Travel expenses.** Expenses related to business travel, but excludes meals while traveling away from home.
- **Utilities expenses.** Utility payments at an office or business property.
- **Wages.** The amount of salaries and wages (less employment credits) paid to employees or contractors.

C Experiment results for additional line items

In this section we explore the performance of AURA compared to other methods on other Schedule C expense line items using the same training procedure described in Section 5.2.

C.1 Comparisons to proximity anomaly scores

We perform the same experiment across all line items done in Section 5.3, now comparing AURA to a modified version of AURA using a proximity anomaly score (AURA-P). Figure 8 displays the average difference in adjustment amount between AURA and AURA-P and its 95% confidence interval for each of the Schedule C expense line items. Aside from the large negative value for the “Costs of Goods Sold" expense line item. We see again that the majority of the differences are positive. Once we exclude the “Costs of Goods Sold" and “Other expenses" line items, we again see that the average differences are positive for a majority of the expense line items, indicating that the dependency anomaly scores provide a better measure of anomalousness in the tax compliance setting than proximity anomaly scores.

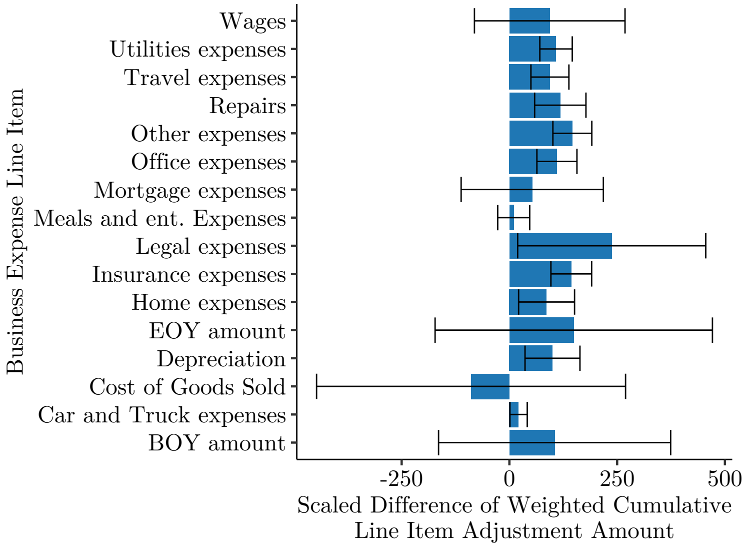

Figure 8: The average difference in adjustment amount between AURA and AURA-P (a modified version of AURA using proximity-based anomaly scores), scaled by the average adjustment amount for each expense line item, for an audit budget of 1.5% with 95% confidence intervals. Positive values indicate that AURA outperforms AURA-P for the given expense line item, while negative values indicate that AURA-P outperforms AURA.

D Statistical testing procedure for comparing line item means

For each of the 16 Schedule C expense line items, we calculate the difference in adjustment amount at a budget of 1.5% from 50 resamples from each of the five folds. We perform a one-way ANOVA to test at least one of the group means differs from zero using the following null and alternative hypotheses

$$H_0 : \mu_1 = \mu_2 = \cdots = \mu_{16} = 0$$
$$H_1 : \mu_i \neq 0 \text{ for some } i \in \{1, \ldots, 16\}.$$

E Comparisons to an unsupervised baseline

In this section we compare the difference in performance between AURA and an unsupervised baseline method. For the unsupervised baseline, we train an isolation forest (Liu et al., 2008) on the unlabeled data and use this model to assign anomaly scores to each of the labeled observations. The observations are then selected for audit based on these anomaly scores. Figure 9 shows the difference in adjustment amount between AURA and this unsupervised baseline model.

In Table E, we quantify the difference in performance between AURA and the unsupervised baseline method on a per audit basis by estimating the average per audit dollar difference in audit adjustment amount. For several expense line items, all of the returns selected for audit based on the predictions of the unsupervised method had zero adjustment, resulting in zero total adjustment captured.

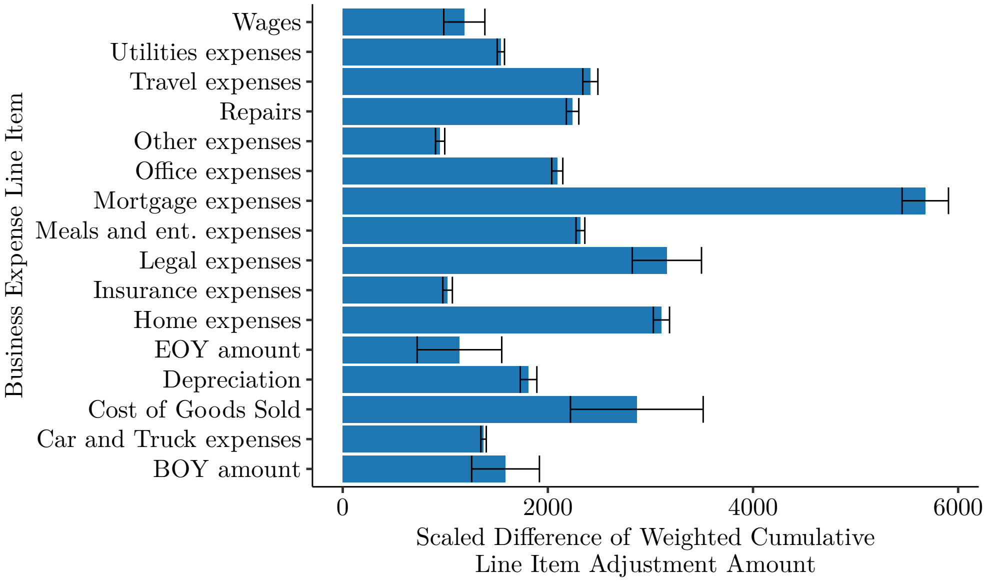

Figure 9: The average difference in adjustment amount between AURA and an unsupervised model baseline, scaled by the average adjustment amount for each expense line item, for an audit budget of 1.5% with 95% confidence intervals. Positive values indicate that AURA outperforms the unsupervised baseline for the given expense line item, while negative values (not observed here) would indicate that the unsupervised baselinme outperforms AURA.

Business Expense Category	AURA Relative Gain ($)	Unsupervised Adjustment Amount ($)	Percent Difference	P-value
Beginning of year inventory amount	4,370	0	—	<0.001
Car and truck expenses	12,522	277	45.19%	<0.001
Cost of goods sold	12,237	32,009	0.38%	<0.001
Depreciation	5,407	102	52.97%	<0.001
End of year inventory amount	5,000	0	—	<0.001
Home expenses	3,159	27	116.05%	<0.001
Insurance expenses	2,020	0	4162.68%	<0.001
Legal expenses	2,727	50	54.15%	<0.001
Meals and entertainment expenses	3,081	10	318.51%	<0.001
Mortgage expenses	2,376	4	573.16%	<0.001
Office expenses	2,727	56	49.11%	<0.001
Other expenses	13,570	345	39.33%	<0.001
Repair expenses	4,277	2	2738.21%	<0.001
Travel expenses	4,539	50	90.57%	<0.001
Utilities expenses	3,281	21	157.35%	<0.001
Wages	1,843	1	1705.77%	<0.001

Table 2: For each expense line item we report the estimated gain of AURA relative to the baseline unsupervised estimation method, the per audit adjustment amount for the baseline unsupervised estimation method, the difference between AURA and the unsupervised baseline in percentage terms, and the p-values from a one-sample t-test. The results are averaged over the results from the 50 subsamples taken in each of the five folds of the cross-validation procedure. See Section 5.2 for additional details.