PRELIMINARY DRAFT

A Machine Learning Approach to Identifying Postdocs in LEHD Data*

James C. Davis† Holden A. Diethorn‡ Gerald R. Marschke§ Andrew J. Wang¶

April 18, 2023

Click here for the latest version.

Abstract

This paper details the creation of the ACS-LEHD Doctorate Panel—a new linked employer-employee longitudinal dataset of the doctoral workforce enabling researchers to analyze the quarterly labor market outcomes of STEM doctorates and postdocs within the secure environment of a Federal Statistical Research Data Center (FSRDC). To impute the quarterly postdoc employment status of doctorates in matched ACS-LEHD data, we train a machine learning algorithm on the small share of data for which quarterly postdoc employment status is known, yielding an out-of-sample imputation accuracy of over 97%. We include a preliminary analysis of the earnings disparity between postdoc-trained and nonpostdoc-trained biomedical doctorates in the ACS-LEHD Doctorate Panel, finding that postdoc-trained biomedical doctorates tend to earn less than their nonpostdoc-trained counterparts, and that this difference in pay narrows, but does not disappear, when including firm and occupation fixed effects.

*This paper is based upon work supported by the National Science Foundation under Grant No. DGE-1661278. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. This paper uses data provided by the U.S. Census Bureau. Any opinions and conclusions expressed herein are those of the authors and do not necessarily represent the views of the U.S. Census Bureau. The findings and conclusions in this paper are those of the authors and should not be construed to represent any official USDA or U.S. Government determination or policy. All results have been reviewed to ensure that no confidential information is disclosed.

†USDA Economic Research Service. Email: James.Davis2@usda.gov

‡NBER. Email: hdiethorn@nber.org

§SUNY Albany and NBER. Email: gmarschke@albany.edu

¶Stanford University. Email: andrewjwang@stanford.edu

1 Introduction

If in developed economies economic growth and sustained increases in living standards primarily arise from scientific and technological advances, then the state of the STEM workforce is of first-order importance. Are our educational institutions producing enough of the right kinds of STEM workers? Are these institutions passing over some households due to reasons of ethnicity, gender, geography, or income, thus leaving valuable human capital unharnessed? Do federal and state policies designed to grow the STEM workforce "work"? Do they produce better jobs, increase innovation rates, increase the tax base, and stimulate local, regional, and national economies? Are there enough students in the STEM educational pipeline to meet current and future STEM demand? How does the churning of STEM workers in the economy impact innovation and returns to R&D? What is the value to a student and to society of a STEM graduate degree and postdoctoral training, and how does this vary across STEM fields?

To prepare STEM students for the workforce, policymakers, education officials, and the students themselves need to know about the different types of career paths available to STEM workers, where in the economy STEM skills are most valued, and the returns to different levels of STEM training, including both doctoral and postdoctoral training. Policymakers require detailed data of the STEM workforce to effectively allocate educational resources and implement policies to head off STEM labor shortages or gluts.

Currently, the bulk of information concerning the STEM doctoral labor market comes from two longstanding National Science Foundation (NSF) surveys: The Survey of Earned Doctorates (SED), which is an annual census of doctorates graduating from US institutions, and the Survey of Doctorate Recipients (SDR), a longitudinal biennial survey of a representative sample of STEM doctorates who earned their PhD in the US. SDR respondents are sampled from the SED, and so these two sources of data are easily linked to carry out studies of the STEM doctoral workforce. The SED provides detailed demographic data and information about each doctorate's graduate school experience, such as length of study and source of financial support, while the SDR is able to track STEM doctorates as their careers develop and contains self-reported employment information such as annual salary and primary work activity. This data is commonly used to analyze the careers of STEM doctorates, allowing for research into a breadth of topics such as the mobility patterns of new PhDs (Stephan, 2006), gender-differences in academic careers (Fox and Stephan, 2001; Ginther and Kahn, 2009), the returns to postdoctoral training (Kahn and Ginther, 2017; Diethorn and Marschke, 2022), and the impact of immigration policy on the stay rates and employment decisions of foreign-born doctorates (Khosla, 2018; Kahn and MacGarvie, 2019; Diethorn, 2022) and on the salary of native doctorates (Borjas, 2009).

However, many critical research questions related to the doctoral workforce cannot be answered with the SDR, specifically those concerning the demand-side of the STEM labor market. This is because SDR data lacks detailed employer characteristics, including firm identifiers, that are

necessary both to analyze and forecast PhD labor demand. One important factor influencing the demand for STEM PhDs and postdocs in the economy is private business' desire to access the fruits of externally-conducted research, including that taking place at universities and other firms. Studies in both the economics and sociology of innovation literatures suggest that new scientific knowledge is frequently "tacit" and difficult to transmit to the uninitiated via spoken or written communication (Polanyi, 1958, 1966; Kaiser, 2005)—that knowledge is often "wrapped up in a person," in which case the most efficient means of transmitting knowledge across organizational boundaries may be via person-to-person contact facilitated by the churning of workers throughout the economy. Understanding how knowledge spillovers are facilitated across universities and between firms is important as these spillovers likely play a role in both local economic development and national economic growth. A linked employer-employee dataset able to track doctorates across jobs quarter-to-quarter, and to identify the possibly multiple jobs held by a doctorate in each quarter, is ideal for analyzing these spillovers, which can be done by, for example, measuring how the employment history of doctorates, including their previous exposure to high R&D employers (including academic employers), impacts their labor market value and the innovation outcomes of their subsequent employers.

There is also a long-standing literature on the complementarity between technology and skills (e.g., Acemoglu, 1998; Goldin and Katz, 1998; Bresnahan, Brynjolfsson, and Hitt, 2002; Autor, Levy, and Murnane, 2003; Acemoglu and Autor, 2011; Deming and Noray, 2020), and firm panel data with detailed information on firm inputs and outputs is necessary to study the extent to which STEM workers stimulate the creation of new technology, and the ways in which the emergence of new technologies stimulates the demand for newly-trained STEM workers. In contrast to the SDR, a linked employer-employee dataset of the doctoral workforce enables researchers to investigate how technological change interacts with the utilization of STEM PhDs in private business and influences PhD labor demand.

In this paper, we construct a new panel dataset of PhD-holders and postdocs that contains demographic information such as age, sex, and race, as well as quarterly employment information for each individual that includes, for each job, the identity of their employer as well as their earnings. The current paper focuses on the implementation of a machine learning strategy to predict/impute the postdoc status of university employees with PhDs. Since machine learning methods may be unfamiliar to many social scientists, we discuss the machine learning model used in this paper—random forests—in some detail, as well as some standard diagnostics used to assess the performance of machine learning models. Building on Diethorn and Marschke (2022), we also include a preliminary analysis of how earnings between doctorates and postdocs in biomedical science differ by employment sector, and how controlling for the firms at which doctorates work impacts these estimated differences. In future work, we will utilize this dataset to describe the career trajectories of STEM PhDs, formulate and estimate models of STEM PhD and postdoc demand, including evaluating the interplay between firm innovation and the employment of STEM

PhDs and postdocs, and more rigorously explore the earnings differences between postdoc-trained and nonpostdoc-trained doctorates.

2 Merging Datasets of Disparate Size

Existing survey-based and administrative-based big data such as the American Community Survey (ACS) and the linked employer-employee Longitudinal Employer-Household Dynamics (LEHD) database, respectively, are rich sources of demographic and economic information, but, as with most datasets, are lacking in a wide-range of details of interest to researchers.[1] To obtain variables of interest, researchers commonly search for another dataset with such variables and then link across data sources using the "conventional method" illustrated in Figure 1: a new Dataset A is merged to an existing big Dataset B in order to create a new Dataset C comprised of all matched observations between the two datasets; the potential majority of the observations in big Dataset B that go unmatched are dropped as they lack variables that are key to the anticipated analysis.[2] Thus, a valuable part of big data, namely a part of what makes them "big" in the first place (i.e., containing many observations), is lost in the conventional process due to a tradeoff between more variables and more observations.

Imputation methods allow researchers to avoid sample loss when extracting key variables from smaller datasets by "predicting" the values of the key variable for unmatched observations in the bigger dataset. The use of machine learning algorithms (e.g., random forests, neural nets, etc.) to impute data carries strong appeal as these methods are known for their strong out-of-sample predictive performance and enable a transparent assessment of imputation accuracy.[3] The machine learning (imputation) approach to merging datasets of disparate size is shown as the "machine learning method" in Figure 1: First, the new Dataset A is merged with big Dataset B, but, in contrast to the conventional method, we retain the unmatched observations from big Dataset B. Then, for each variable originally unique to Dataset A, the researcher trains a machine learning algorithm on the matched observations, using only those variables found in Dataset B as predictors.

[1] For example, researchers wanting to use the ACS or LEHD to examine the career trajectories of Ph.D. recipients who have completed a postdoc face significant difficulty as there is no postdoc occupation category in the ACS and no occupation categories at all in the LEHD. Researchers could try to infer such information by formulating their own *ad-hoc* rules based on an individuals age and earnings, but determining an effective age and earnings cutoff may be difficult; additionally, there are likely a variety of additional variables that could be useful for imputation, but including additional variables increases the complexity of the problem for a researcher considering manual imputation. This motivates the automated machine learning approach used in this paper to identify postdoc-trained doctorates in linked ACS-LEHD data using a wide variety of predictors.

[2] In Figure 1 we assume, for the sake of simplicity, that Dataset A shares a single unique identifier with Dataset B, that each observation in Dataset A matches to an observation in Dataset B, that $N > n$, and that the j variables in Dataset B are distinct from the k variables in Dataset A (where the "+1" variable in each dataset is the common unique identifier).

[3] Machine learning methods are also useful in that they are able to generate predictions in cases where conventional methods fail, such as when the number of observations in a dataset is greatly exceeded by the number of potential predictors ($n << p$) or when a great number of interaction effects exist among the predictors but is unknown to the researcher *ex ante*.

Lastly, the trained algorithm is used to predict the values of the key variables for the unmatched observations. Of course, the efficacy of this method depends on the extent to which the variables in Dataset B are predictive of the key variables in Dataset A, but this can be transparently assessed using methods discussed in this paper.[4] Another key assumption is that the observations in Dataset A are representative of those contained in Dataset B (i.e., that we do not have a problem of "selective labels" as discussed in Kleinberg et al. (2018) and Mullainathan and Obermeyer (2017)).[5]

In the next section, we detail our construction of a new linked employer-employee longitudinal dataset of the STEM doctoral workforce that tracks the career outcomes of doctorates and can be linked to firm-level datasets maintained in the Census Bureau's Federal Statistical Research Data Centers ("RDCs") such as the Longitudinal Business Database (LBD) and Business R&D and Innovation Survey (BRDIS). Then, we detail the underlying machine learning methods used to impute the postdoc-trained status of doctorates in our new dataset and the methods used to assess imputation accuracy; while none of these methods are new, we hope our discussion is helpful for readers not yet acquainted with machine learning methods. After discussing how these methods perform at predicting postdoc spells among the doctorates in our dataset, we include a preliminary analysis comparing the earnings of postdoc-trained and nonpostdoc-trained biomedical doctorates, analyzing how much of the disparity is explained by inclusion of firm and occupation fixed effects.

3 Data Description and the Data Linkage Process

We utilize three major data sources in the construction of our linked employer-employee dataset of the doctoral workforce: 1) the American Community Survey (ACS), 2) Longitudinal Employer-Household Dynamics (LEHD) data, and 3) UMETRICS. These data are accessible to researchers with Special Sworn Status on approved projects via the Federal Statistical Research Data Centers (FSRDCs) maintained by the US Census Bureau.

The ACS is an annual survey administered by the US Census Bureau to obtain demographic and economic information for a nationally-representative sample of the US population. In each year, the US Census Bureau contacts approximately 3.5 million addresses to participate in the ACS, with the sample changing in each year.[6] The data collected from the ACS includes information on the occupations, educational attainment, and background characteristics (e.g., age, sex, place of

[4]This paper does not introduce new machine learning techniques, but rather explains how existing techniques can be used to impute key variables to avoid sample attrition, which is especially important when merging datasets of disparate size.

[5]This problem can be difficult to avoid. In our application, we only know whether an individual from a UMETRICS university is a postdoc or not if the person is grant-funded (this includes both federal and non-federal grants). Thus, the labels are selected based on grant-funded status, and so using our algorithm to predict on a sample that also contains individuals who are not grant-funded could be problematic if there are significant differences between individuals who are and are not grant-funded. As most postdocs are primarily funded by grants (`https://ncsesdata.nsf.gov/gradpostdoc/2018/html/gss18-dt-tab003-2.html`), we believe that it is unlikely to be a significant source of selective labels.

[6]See `https://www.census.gov/programs-surveys/acs/about/top-questions-about-the-survey.html`.

birth) of survey respondents.[7] We utilize the annual ACS person files for years 2005-2017 as part of constructing our doctoral panel dataset.

For each year of the ACS, we limit the sample to persons who indicate that they hold a doctorate degree, where persons are uniquely identified by nine-digit Protected Identification Keys (PIKs).[8] We then append each yearly ACS doctorate dataset to form an "ACS Doctorate Panel" that spans the years 2005-2017.[9]

LEHD data is maintained by the US Census Bureau and is primarily based on administrative data collected by US States such as Unemployment Insurance (UI) earnings data, as well as the Quarterly Census of Employment and Wages (QCEW). We utilize two LEHD datasets in the course of creating our doctoral panel dataset: 1) The Employment History Files (EHF) and 2) The Employer Characteristics Files (ECF). For both datasets, we utilize all observations between 2001-2015. The EHF contains information on where individuals work each year and the earnings generated from their job(s) in each quarter. As in the ACS, individuals are uniquely identified by their PIK. Firms are identified by state employer identification numbers (SEINs) and establishments within each firm are identified by the SEIN reporting unit (SEINUNIT) so that an establishment is uniquely identified by SEIN-SEINUNIT (Vilhuber, 2018). The raw EHF dataset is structured as a yearly job-level dataset where the unit of observation is an employer-employee combination within the given year (PIK-SEIN-SEINUNIT-year). For each employer-employee combination, we have the quarterly earnings, and so we reshape the EHF into a quarterly job-level dataset so that the unit of observation becomes PIK-SEIN-SEINUNIT-year-quarter.[10]

The ECF contains establishment-level information on US employers, including the establishment's federal Employer Identification Number (EIN), industry (six-digit NAICS code), and measures of the size and age of the firm associated with the establishment. The ECF is a quarterly establishment-level dataset and is thus unique at the SEIN-SEINUNIT-year-quarter level.

To create the "LEHD Panel", we link the EHF and ECF datasets by merging on establishment-year-quarter (SEIN-SEINUNIT-year-quarter). This effectively gives us the job profile for all individuals in LEHD states during 2001-2015 who have positive earnings reported in state UI data.[11] Since we are presently interested in the job profile of individuals who have earned doctorates, we keep only those observations that are associated with PIKs found in the previously-created ACS

[7] After the 2000 Census, the ACS replaced the "long-form" of the decennial census which had previously collected this information.

[8] PIKs are internal Census identifiers randomly generated for each individual in order to protect the privacy of each individual person while also facilitating linkage across Census data platforms (Mulrow et al., 2011).

[9] Individuals may appear more than once in the ACS Doctorate Panel if they are randomly surveyed in multiple ACS years. In these cases, we keep the observation for the most recent year that an individual is surveyed in the ACS.

[10] Individuals employed at an establishment but who do not have strictly positive earnings at their employing SEIN in a given quarter will not have earnings reported within that SEIN for that quarter (Vilhuber, 2018). Therefore, we code earnings as zero for any quarter where earnings is missing.

[11] US states can voluntarily opt into or out of the LEHD program, and are also able to decide for which research projects they will make their state's data available. In this paper, "LEHD states" refers to the subset of states that participated in the LEHD program in a given year for which we have data access.

Doctorate Panel. This LEHD Doctorate Panel is then linked to the ACS Doctorate Panel by person (PIK), creating the ACS-LEHD Doctorate Panel, where the unit of observation is person-establishment-year-quarter (PIK-SEIN-SEINUNIT-year-quarter). Figure 2 gives a diagrammatic summary of the process described above used to create the ACS-LEHD Doctorate Panel.

The ACS-LEHD Doctorate Panel has one shortcoming that prevents us from carrying out a comparative analysis of doctorates who have and have not completed a postdoc—the absence of variables indicating whether a doctorate is employed or was every employed as a postdoc. Therefore, we introduce a third data source, UMETRICS, to obtain labels for whether a doctorate is employed as a postdoc in a given quarter for a subset of the ACS-LEHD Doctorate Panel; once we have postdoc labels for a subset of the observations, we can use a machine learning approach to predict the postdoc status of the unlabeled subset of the ACS-LEHD Doctorate Panel. This then allows us to carry out our preliminary analysis where we explore how controlling for the firm where a biomedical doctorate works changes the estimated effect of postdoc training on after-postdoc earnings.

UMETRICS (Universities: MEasuring The impacts of Research on Innovation, Competitiveness, and Science) is a database maintained by the Institute for Research on Innovation & Science (IRIS) at the University of Michigan and is accessible to Special Sworn Status researchers on approved projects in the Census FSRDCs.[12] UMETRICS is based on administrative data obtained from 36 research universities that contain information on both federal and nonfederal grants ("awards") received by each university including which university employees are paid from each award and what vendors receive funds from the awards in exchange for goods and services (Institute for Research on Innovation & Science, 2019). UMETRICS data (2019 release) spans the years 2001-2018, with more universities being added to the sample over time.[13]

We utilize the Employee Transaction File (ETF) of the UMETRICS 2019 data release. The ETF contains university payroll transactions for employees paid on any (1) research-related federal or non-federal grants or (2) non-research-related activities such as work-study programs. Each IRIS-member university is assigned a unique "institutionid" for de-identification purposes, and each university employee paid on a grant or award is assigned an institution-specific "empnumber" so that individuals within UMETRICS can be uniquely identified by institutionid-empnumber.[14] Each observation contains a "unique award number" that identifies an award and its funding source, the empnumber and institutionid of the person being paid from the award's funds, the period start date and end date that represents the beginning and end of the monthly pay period, and the UMETRICS

[12] UMETRICS data can also be accessed via the IRIS virtual data enclave by approved researchers. For more information on UMETRICS data, see `https://iris.isr.umich.edu/` and Lane et al. (2015). See Buffington et al. (2016) and Zolas et al. (2015) for descriptive analyses using UMETRICS data.

[13] See Appendix A of Institute for Research on Innovation & Science (2022) for a list of the universities in UMETRICS data.

[14] While a single individual will only have one empnumber within a single institutionid, if that individual moves to a different UMETRICS university, he/she will be identified by a new institutionid-empnumber. However, UMETRICS data has been matched to Census PIKs, which allow researchers to track the individuals as they move from one university or firm to another.

occupational classification of the employee which is determined by IRIS based on job title information provided in university HR records. The UMETRICS occupational classifications encompass six major groups of workers: Faculty, Post Graduate Researcher (i.e., Postdoc), Graduate Student, Undergraduate, Staff, and Other.[15] Persons in UMETRICS have been matched to Census persons and assigned a PIK, and so we are able to match individuals in UMETRICS to the ACS-LEHD Doctorate Panel. Linking the UMETRICS ETF to the ACS-LEHD Doctorate Panel allows us to obtain the true postdoc status for the subset of observations in the ACS-LEHD Doctorate Panel that matches to UMETRICS.

Figure 3 shows the steps used to merge UMETRICS data to the ACS-LEHD Doctorate Panel. A quick summary of these steps is the following: First, we obtain PIKs for observations in UMETRICS ETF data and convert the data from transactions-level to PIK-year-quarter level. Next, we merge this data to the ACS-LEHD Doctorate Panel which allows us to identify the postdoc status of the UMETRICS subset of the ACS-LEHD Doctorate Panel. Then, we keep only those observations in the overall sample where the employee is working in either "Colleges, Universities, and Professional Schools" (NAICS = 611310) or "General Medical and Surgical Hospitals" (NAICS = 622110) since the vast majority of UMETRICS university employees are classified as working in these industries. This should improve the representativeness of the UMETRICS subsample, which is important since this subsample will be used to train a machine learning model used to predict the postdoc status of the rest of the observations in the ACS-LEHD Doctorate Panel. This leads us to the final prediction sample referred to as the "ACS-LEHD Academic Doctorate Panel with UMETRICS" in Figure 3, which is unique on person-year-quarter (PIK-year-quarter).[16]

Table A.1 displays the variable names and definitions for this dataset. Our goal is to predict, for each individual in the prediction sample, which quarters between 2001-2015 (if any) represent a period of employment as a postdoc. Comparing postdoc observations to nonpostdoc observations, Table 1 shows that doctorates employed as postdocs in a given quarter are younger, are more likely to be foreign-born and Asian, and typically earn less than those doctorates that are employed in nonpostdoc positions. Postdoc observations are also associated with doctorates who have shorter spells at both the employer for whom they work the most quarters and the employer for whom they work the least quarters.[17]

Altogether, the UMETRICS subset of the ACS-LEHD Doctorate Panel contains approximately

[15] Employees classified as Staff are then classified into one of six subcategories so that there is a total of 11 occupational categories in the data altogether.

[16] We are interested in predicting which quarters (if any) of an individual's career are spent as a postdoc, and so a dataset unique on person-year-quarter is sufficient for this purpose. The LEHD-based variables (see Table A.1), such as the job count variables and total earnings variables, are created to incorporate useful information from the more "general" job-level and non-NAICS restricted intermediate datasets used to form the final NAICS-restricted quarterly person-level prediction sample. For simplicity, in the course of describing our machine learning approach and prediction results, we abuse nomenclature by referring to the "ACS-LEHD Academic Doctorate Panel with UMETRICS" simply as the ACS-LEHD Doctorate Panel.

[17] This is in part because postdoc positions are temporary, and so observing a person in a postdoc position makes it likely that we then observe them subsequently with a new employer, whereas nonpostdocs may stay with the same employer throughout all their observations.

18,500 observations representing about 1,900 unique doctorates, whereas the full ACS-LEHD Doctorates contains approximately 2,463,000 observations representing about 98,500 unique doctorates. Since the UMETRICS subsample represents less than 1% of all observations in the ACS-LEHD Doctorate Panel, the conventional method of simply dropping those observations for where postdoc status is unknown would come at great cost. Imputing postdoc status by manually generating rules based on the summary statistics in Table 1 is possible, but it would be difficult to construct such rules using more than just a few predictors. Instead, we seek an approach where the rules used to predict postdoc status are automatically generated based on the data and where the accuracy of the imputation procedure can be reliably assessed. We implement such as approach by utilizing the UMETRICS subset of the ACS-LEHD Doctorate Panel to train a machine learning algorithm where the postdoc status of each doctorate in a given quarter is the target (i.e., the variable to be predicted) and the rest of the variables (or "features") listed in Table A.1 are the predictors.

4 Prediction: Methods and Results

4.1 Random Forests: What They Are and How To Tune Them

We utilize the random forest algorithm originally developed by Breiman (2001) and implemented in the R package randomForest (Liaw and Wiener, 2002) to predict the postdoc status of observations in the ACS-LEHD Doctorate Panel. Random forests are one of the most popular out-of-the box machine learning methods, being utilized in a variety of tasks such as image classification (Bosch, Zisserman, and Muñoz, 2007), gene selection (Díaz-Uriarte and de Andrés, 2006), and land cover classification (Gislason, Benediktsson, and Sveinsson, 2006). Random forests work by "growing" an ensemble of decision trees, obtaining predictions from each of these trees, and then averaging the predictions across these trees to generate a final prediction.[18] In this subsection, we give a summary of classification trees and the random forest algorithm used for classification.

Classification trees are grown by iteratively partitioning a sample of data to group together observations with the same class label (e.g., "postdoc" or "not postdoc") in a process known as recursive binary splitting. Figure 4 shows a fictional classification tree based on two important predictors of postdoc status—age and earnings—along with its equivalent predictor-space representation. Classification trees partition the data at each step by selecting a predictor-cutpoint combination as the basis for the split; for example, in Figure 4 at internal node $N1$, observations are split based on the predictor age and the cutpoint of 35 years, resulting in the two daughter nodes $N2$ and $N3$. Generally, to determine how to split the observations, an optimal cutpoint for each predictor is calculated, and the predictor-cutpoint combination that gives the greatest gain in

[18] See Hastie, Tibshirani, and Friedman (2009) for an extensive and technical treatment of machine learning, and see James et al. (2013) for an introductory treatment with applications using R statistical software. Breiman et al. (1984) is the classic reference for classification and regression trees. As a note on terminology, a decision tree is referred to as a classification tree when the variable to be predicted is a categorical variable, and is referred to as a regression tree in cases where the variable is non-categorical (e.g., continuous variables and count variables).

node purity is chosen to divide the data into two daughter nodes.[19] This process is continued until a stopping criterion, such as the minimum number of observations allowed in a node, is satisfied. Observations in each terminal node (or "leaf") of the decision tree are then predicted as belonging to the class held by the majority of observations in that terminal node. In Figure 4, the terminal nodes are labeled $R1$-$R5$ since the observations grouped into these nodes are the same as those appearing in the identically-named regions in the predictor-space representation of the classification tree.

A strength of classification trees is that they automatically capture interaction effects among predictors without the researcher needing to specify a set of interaction terms *ex ante*.[20] A major weakness of classification trees is that they suffer from high variance: the structure of a given decision tree is highly dependent on the data used to train the model such that a small change to the data may result in a non-negligible change in the tree structure, which can cause a noticeable change in the predictive performance of the model as measured by the model's out-of-sample (or "test") error. To mitigate the weaknesses of unstable (i.e., high-variance) learners such as classification trees, Breiman (1996) introduced an ensemble method known as bagging (Bootstrap AGGregatING).[21] This method works by taking B bootstrap samples from the available training data, fitting a classification tree to each of the bootstrap samples, generating a prediction from each tree for each observation, and then classifying each observation based on a majority vote - that is, the final prediction for each observation is the most commonly predicted class among the B predictions.[22] Figure 5 gives a schematic representation of a bagged tree model.

Random forests improve upon bagged trees by introducing a source of randomness into the tree growing process: at each internal node in each tree, a random subset of the available predictors is first chosen, and then the best split among these randomly chosen predictors is used to split at the node; this contrasts with bagged trees, where the best split among all available predictors is chosen. It may seem odd that random forests typically perform better than bagged trees given that the only difference between these two methods is that random forests restrict the available information considered at each node of each tree. However, the intuition for the performance improvement of random forests over bagged trees stems from the fact that the variance of an average of identically distributed random variables is decreasing in the pairwise correlation of these

[19] Let p_{mk} be the proportion of observations at internal node m that are of class k. Then the Gini index at that node is calculated as $\sum_{k=1}^{K} p_{mk}(1-p_{mk})$ where a smaller value of the Gini index represents a node of greater purity. The predictor-cutpoint combination used to split at an internal node is chosen so that the resulting two daughter nodes give the largest decrease in the Gini index, where the decrease in the Gini index is calculated as follows: first, the Gini index for each daughter node is calculated and weighted by the proportion of parent-node observations falling into that node, and then these measures are subtracted from the value the of the Gini index of the parent node. Recursive binary splitting is referred to as a top-down, greedy approach because at each stage, the data is partitioned to maximize the gain in node purity at that step without considering how a given partition will affect future partitioning of the data and thus ultimate node purity at the terminal nodes—this is done for computational feasibility.

[20] Mullainathan and Spiess (2017) highlight this aspect of decision trees in a regression context.

[21] Ensemble methods are methods that generate predictions by combining the predictions of a set of "base-learners" such as decision trees. Popular ensemble methods include bagging, boosting, averaging, and stacking.

[22] This assumes a default threshold of 50%.

variables. By introducing a source of randomness into the tree-growing process, random forests decorrelate the trees, thus leading to a smaller variance in prediction relative to bagged trees.[23]

Each tree in a random forest is grown on a bootstrapped sample of the original training data which, due to sampling with replacement, contains approximately two-thirds of the original training observations. The approximately one-third of the original training observations that are not used to train a given tree are referred to as the out-of-bag (OOB) observations of that decision tree. It follows that each observation of the original training data will be in the OOB sample of approximately one-third of the decision trees grown in a random forest. The OOB error rate of a random forest is obtained by generating predictions for each original training observation from only those trees for which it is part of the OOB sample, measuring the average error in classification for each observation based on its OOB predictions, and then averaging these error rates across all observations.[24] The OOB error rate is a measure of the predictive performance of a random forest and is commonly used to select the number of decision trees grown in a random forest model: The number of trees is selected to be large enough that the OOB error rate becomes relatively stable—with no risk to overfitting by growing too many trees.

Random forests contain one hyperparameter that the user tunes to obtain the best random forest model: the number of randomly selected variables considered for node splitting at each node in each decision tree, which we refer to as the number of "splitting variables."[25] One way to tune a random forest model is to compare the OOB error rates that are obtained by changing the number of splitting variables and then selecting the hyperparameter value that yields the lowest OOB error rate. However, since the OOB error rate is a measure of the overall classification error rate of the model, it is sensitive to the probability cutoff used for positive prediction. By default, the cutoff is set to 0.5, meaning that, in our application, all observations with a predicted probability of being a postdoc greater than 0.5 would be classified as postdocs.[26] While a seemingly reasonable default, the 0.5 probability threshold may not be optimal as there is no guarantee that this threshold minimizes classification error, and even if it does achieve the minimum classification error, such a property may not be desirable in the presence of class imbalance since prediction will tend to favor the most commonly occurring class, leading, in our case, to a greater prevalence of false negative predictions compared to false positive predictions. A threshold that balances the two types of errors

[23] See Chapter 15 of Hastie, Tibshirani, and Friedman (2009) for technical details.

[24] The OOB error rate and predictions are calculated automatically in the implementation of the random forest algorithm in the R randomForest package.

[25] For classification problems, the recommended default value for the number of splitting variables (m) is equal to the square root of the total number of predictors (p) (Hastie, Tibshirani, and Friedman, 2009). Other hyperparameters that could be adjusted for a random forest is the size or depth of the individual trees making up the random forest and the minimum observations allowed in each terminal node; however, Hastie, Tibshirani, and Friedman (2009) suggest that tuning these parameters do not typically lead to large changes in predictive performance, especially in the case of classification (p. 596). We leave the minimum observations in node hyperparameter set to one and allow trees to be split as many times as needed, which are the default values for classification trees (Hastie, Tibshirani, and Friedman, 2009).

[26] The probability of being a postdoc is calculated as the proportion of decision trees in a random forest that predict an observation as belonging to the postdoc classification.

may be more desirable, and so tuning a random forest model using a metric that is sensitive to the choice of the cutoff should generally be avoided in the case of class imbalance.[27]

A preferred alternative is to rely on a method that explicitly considers the tradeoff between false positive and false negative errors as the cutoff is altered. One such method is to use the OOB predictions to graph a Receiver's Operating Characteristic (ROC) curve for each value of the hyperparameter and choose the number of splitting variables that maximizes the area under the ROC curve.[28] To understand the reasoning behind this method, it is helpful to first introduce what is referred to as a classification method's confusion matrix, as shown in Table 2. A confusion matrix counts the number of true positive, true negative, false positive, and false negative predictions made by a classifier. For a random forest model, we can obtain a confusion matrix based on how well the model predicts the classes of OOB observations. From there, we can calculate the various error and accuracy measures shown in Table 3.

An ROC curve is simply a plot of the true positive rate versus the false positive rate achieved by a given predictive model across all alternative probability cutoffs. The top panel of Figure 6 shows two fictional examples of ROC plots. The dotted diagonal line in each plot represents the performance expected using random guessing for prediction. The connected red lines touching the border represent the performance of a perfect predictive model since it contains point (0,1) in ROC space, which is associated with a 100% TPR and 0% FPR. The blue and green curves lying above the diagonal are two ROC curves, each associated with a separate hypothetical predictive model such as two random forests with different values for the number of splitting variables. Each point on the ROC curve gives the (FPR, TPR) combination achieved by a particular probability threshold; points farther to the right along a given ROC curve correspond to lower probability cutoffs.[29]

In the upper left-hand panel of Figure 6, we see that the model corresponding to the blue ROC curve strictly dominates the model corresponding to the green ROC curve since the "blue model" achieves a higher true positive rate for any given false positive rate, and thus outperforms the "green model" across all probability thresholds. However, deciding between models based on visual inspection of ROC curves is not always so straightforward. For example, in the right-hand panel of Figure 6, we have ROC curves that intersect and overlap, meaning that which model is "better" depends on the probability threshold under consideration. Without a particular probability threshold in mind *ex ante*, a judicious approach is to select the model that exhibits the greatest "global" skill over all possible probability thresholds, rather than a model that exhibits the greatest "local" skill at a particular probability threshold such as the 0.5 default. This can be done by calculating the area under each ROC curve and then selecting the model that gives the maximum area under the curve (AUC).[30] Since the AUC of a model takes account of a model's

[27] If costs differ between false positive and false negative errors, a threshold minimizing the cost could be selected.

[28] See Lahiri and Yang (2013) and Kuhn and Johnson (2013) for an overview of ROC curve analysis.

[29] Keeping in mind that TPR ≡ 1 - FNR, an ROC curve explicitly shows that lowering the probability cutoff results in a lower incidence of false negative errors at the cost of an increase in false positive errors.

[30] AUC lies in the range [0,1], with a perfect predictive model having AUC = 1 and random guess having AUC = 0.5.

performance across all probability thresholds, it is a more appropriate metric to use when tuning a random forest compared to the OOB error rate of the model which is necessarily dependent upon the choice of a probability cutoff.[31]

After tuning a random forest model by selecting the number of splitting variables that maximizes AUC, one still needs to determine the appropriate probability cutoff to use for prediction. This is particularly important in cases where class imbalance is an issue, as the default cutoff is likely to overpredict the most commonly occurring class. This can be done by selecting the threshold that maximizes some measure of local skill. Two popular cutoff choices are those that either minimize the sum of squared false positive and false negative rates ($FPR^2 + FNR^2$) or maximize the sum of true positive and true negative rates (TPR + TNR). We refer to the cutoff that minimizes the sum of squared false positive and false negative rates as the "top-left" cutoff, as this cutoff identifies the point on the ROC curve closest in Euclidean distance to point (0,1) in ROC space. This cutoff is represented in the bottom panel of Figure 6 as the purple point on the ROC curve. We refer to the cutoff that maximizes the sum of true positive and true negative rates as the "Youden" cutoff since this cutoff maximizes the Youden Index (Youden, 1950): TPR + TNR - 1. The Youden cutoff identifies the point on the ROC curve where the model is most skilled relative to random guessing, that is, where the vertical distance between the ROC curve and the no-skill diagonal is greatest. This cutoff is represented in the bottom panel of Figure 6 as the orange point on the ROC curve. To choose between these cutoffs, one can use the accuracy and error measures in Table 3 and choose the cutoff that is most desirable, in terms of the metrics viewed as most important, for the application at hand.

In our experience, the top-left and Youden cutoffs perform similarly, with both cutoffs increasing the TPR (or "recall") of a predictive model relative to the default cutoff in cases of class imbalance. However, this increase in TPR is at the expense of PPV (or "precision"). In our case, we are interested in both recall (i.e., in predicting true postdocs as being postdocs) and precision (i.e., in having those that we predict to be postdocs to actually be postdocs), and so would like to balance these metrics rather than optimize one at the expense of the other. Therefore, we prefer to select a cutoff that maximizes the F1-score of the prediction model, which is defined in Table 3. The F1-score is the harmonic mean of TPR and PPV, and thus maximizing the F1-score leads to a balance of TPR and PPV by penalizing large deviations of these measures from each other.[32]

[31] An alternative is to focus on the performance of a model across a set of thresholds within a predetermined range, rather than considering performance across all possible thresholds. In this case, one could compare models using the area under the curve between the selected threshold bounds—this is known as the partial area under the ROC curve (pAUC).

[32] One could hypothetically obtain a predictive model with a high PPV and low TPR—for example, imagine a sample with 100 postdocs and 900 nonpostdocs. If only one observation were predicted to be a postdoc, and if this prediction was correct, the model would have a 100% PPV and a 1% TPR. Likewise, one can easily obtain a high TPR and a low PPV by simply classifying all observations as postdocs—in the hypothetical example given here, this would lead to a 100% TPR and 10% PPV. Therefore, it is important to consider both measures when choosing among different cutoffs, rather than one in isolation.

4.2 Model Selection and Assessment of Random Forests: An Application to Predict Postdoc Status in LEHD Data

In this section, we describe our machine learning based strategy for using the UMETRICS subsample of the ACS-LEHD Doctorate Panel to predict postdoc status for the rest of the observations in the ACS-LEHD Doctorate Panel. A quick summary of our method is as follows: First, we split the UMETRICS subset of the ACS-LEHD doctorate data into a training set (50%) and a test set (50%). The training set is used to train/fit competing random forest models, the area under the ROC curves generated from the OOB predictions from each model are used to compare the competing random forest models (i.e., tune the splitting variables hyperparameter), and then the OOB predictions are used to identify alternative probability cutoffs for positive prediction to mitigate possible error rate imbalances caused by class imbalance. To assess the predictive accuracy of our tuned random forest model, we estimate accuracy rates using the test data. Once our model is assessed, we then retrain the model on the entire UMETRICS subsample and use this trained random forest model to predict the postdoc status of the rest of the observations in the ACS-LEHD Doctorate Panel. Table 4 outlines this strategy for model selection, assessment, and prediction. We give the rationale for our method in what follows.[33]

The training (or apparent) error rate of a predictive model is an overly optimistic measure of prediction error; this is because when any model is trained or estimated on a given dataset, it is likely not only to discover signals in that data that are useful for out-of-sample prediction, but also to fit sample-specific noise. To properly assess the predictive power of a model, a portion of the data should be withheld during the training process so that performance of the model on out-of-sample data can be accurately assessed. Therefore, we partition the data into a training set used to train and tune our random forest model (model selection), and save the other 50% of the UMETRICS subsample to be used as the test set used to estimate the out-of-sample performance of our tuned random forest model (model assessment). It is important that no data used in model assessment is used in model selection (i.e., feature selection, hyperparameter tuning, model comparison), and vice versa—Mullainathan and Spiess (2017) refer to this as a *firewall principle*.[34] Therefore, we utilize OOB predictions, rather than test set predictions, as the basis of the performance measures used for model tuning. The test set predictions are only used when estimating the generalization

[33]See Appendix B for an alternative method which enables researchers to compare random forests with other machine learning methods. Specifically, we compare our random forest prediction results with those using another popular tree-based ensemble method known as "boosted trees."

[34]Ambroise and McLachlan (2002) show that using a full dataset for feature selection prior to partitioning the data into a training set and test set leads to an optimistic bias in cross-validation (CV) error estimates. Varma and Simon (2006) show that the CV error rate used to tune a model underestimates the generalization error of the model, although Tibshirani and Tibshirani (2009) provide evidence that this mostly occurs in cases where the number of features greatly exceeds the number of observations ($n << p$). Cawley and Talbot (2010) show that tuning a model's hyperparameters using the full set of data prior to partitioning the data and calculating the test set error will lead to an optimistically-biased estimate of the generalization error. Hastie, Tibshirani, and Friedman (2009) warn that tuning hyperparameters or selecting a model based on minimizing the test set error will cause the test set error to underestimate the generalization error.

error of our tuned random forest model.

After splitting the UMETRICS subsample into a training set and a test set, we train four random forest models, each with a unique value for the number of splitting variables considered at each tree node.[35] To select the number of trees to use in these random forest models, we graph the OOB error rate for each of the models as we add more trees, from 1 tree to 2000 trees. The OOB error rate for each model becomes relatively stable after about 1000 trees such that the model with the number of splitting variables (m) equal to the square root of the number of available predictors (p) tends to perform best on this metric. To err on the side of caution, we choose 2000 trees for our random forest models, which is well past the point where the OOB error rate for each model stabilizes.[36]

As previously noted, the OOB error rate measures the classification error rate of the model, and is thus sensitive to the probability cutoff used for positive prediction. Rather than selecting the random forest model with the lowest OOB error, we aim to select a model based on the global skill of that model over all possible probability thresholds. Therefore, we tune the number of splitting variables by first calculating the area under the ROC curve (AUC) for each random forest model and then selecting the model with the number of splitting variables that maximizes AUC. Table 5 shows that the random forest model with the number of splitting variables equal to the square root of the number of available predictors ($m = \sqrt{p}$) achieves the greatest AUC, and thus represents our tuned random forest model.

Having selected the random forest model with the greatest global skill, we now identify alternative probability thresholds to use for positive prediction of postdoc status. Kuhn and Johnson (2013) suggest considering alternative probability cutoffs for positive prediction to account for class imbalance since class imbalance leads to error rate imbalance—a predictive model seeking to minimize classification error will tend to favor predicting the most commonly occurring class. In our case, since postdoc is the rarer class (see Table 1), we would expect the false negative rate to exceed the false positive rate (or equivalently, the true negative rate to be greater than the true positive rate). We are also interested in balancing the precision (PPV) and recall (TPR) of our model. Therefore, we consider two thresholds that are less sensitive to class imbalance. The first of these

[35] For computational feasibility, Kuhn and Johnson (2013) suggest only tuning over a limited number of values for the number of splitting variables. The first three values are chosen following the exposition in James et al. (2013) whom compare the default value of $m = \sqrt{p}$ with $m = p/2$ and $m = p$ (bagged trees). We also consider $m = p/3$, which corresponds to the suggested default value for random forests in a regression context (Hastie, Tibshirani, and Friedman, 2009) and puts the number of splitting variables roughly halfway between $m = \sqrt{p}$ and $m = p/2$ in our application. In all four cases, we round the number of splitting variables to the integer value closest to these targeted values.

[36] Due to disclosure concerns, we are presently unable to include this figure. See Figure A.1 for an analogous figure based on the "Spambase dataset" from the UCI Machine Learning Repository at `https://archive.ics.uci.edu/ml/datasets/spambase` or easily accessed via the R package "ElemStatLearn." This dataset contains information from 4601 emails including how many times an exclamation mark appears in the email and the longest string of ALL CAPS in the email, as well as whether the email was ultimately classified as spam or not spam. This enables the dataset to be used to predict whether an email is spam based on 57 characteristics of the text. Error rates are not representative of those rates we get using random forests for postdoc prediction, but the shape of the OOB error rates follows the same general pattern, with a steep decline early on before stabilizing.

alternatives is the "F1" cutoff which corresponds to the threshold that maximizes the harmonic mean of TPR and PPV (i.e., the F1-score). The second alternative is the "Youden" cutoff which corresponds to the point on the ROC curve that is the greatest vertical distance away from the no-skill (random) forecast represented by the diagonal line in Figure 6. All cutoffs are derived using the random forest OOB predictions for the training data.

Table 6 gives the values of the two different probability cutoffs that we consider for our model—the Youden cutoff and the F1 cutoff—with the corresponding OOB accuracy rates used to derive the cutoff values.[37] As we can see, the cutoff that maximizes the F1-score is close to the default 0.5 threshold, while the Youden cutoff is significantly lower. As is typically the case, moving from a cutoff near 0.5 to the lower Youden cutoff results in a drop in total accuracy, but a more even distribution of errors in terms of false negatives and false positives. If the ultimate objective is to accurately predict as many true postdocs as possible, then the Youden cutoff appears to be the better choice as it achieves the higher TPR.[38] However, the increase in TPR comes at the expense of lowering the purity of the predicted postdoc sample (PPV); while we increase the percentage of true postdocs that we predict as postdocs, we also increase the percentage of nonpostdocs that we incorrectly predict as being postdocs. This shows up as a decrease in the PPV of the random forest model when moving from the F1 cutoff to the Youden cutoff. This decrease in PPV is comparatively large—the 6 percentage point gain in TPR by moving from the F1 cutoff to the Youden cutoff would come at the cost of a 10 percentage point reduction in PPV.[39] Without information on the relative costs of false positive and false negative errors, it is somewhat a matter of researcher preference as to which cutoff is best. Ultimately, we favor using the F1 cutoff to balance the precision and recall of our model—we are interested in both accurately predicting true postdocs (high TPR) and ensuring that predicted postdocs are indeed postdocs (high PPV). Thus, we select the random forest model with $\sqrt{p}$ splitting variables and a probability cutoff of 0.4825 as our final prediction model.

It is important to note that, while useful for deciding on which probability threshold to select, the accuracy measures in Table 6 are optimistically-biased estimates of the generalization accuracy of the tuned random forest model. This is because the random forest model with $\sqrt{p}$ splitting variables was chosen based on its performance on the OOB observations—the same data used to produce the accuracy measures in Table 6. Therefore, having finished model selection, we use the model's performance on the test set to yield unbiased measures of the generalization accuracy of our selected model, which we report in Table 7.[40] Reassuringly, our model performs strongly on data it had never "seen" at any point in the model selection process.

[37] We also considered the default 0.5 cutoff, the cutoff that minimizes total classification error, and the top-left cutoff. The top-left performs similarly to the Youden cutoff, and the default cutoff and the error-minimizing cutoff are both similar to the F1 cutoff. The results from these cutoffs are not reported due to possible disclosure concerns.

[38] A TPR of 97.11% means that 97.11% of all true postdocs would be predicted as being postdocs by the model.

[39] A PPV of 91.10% means that, of all those observations that we predict as being postdocs, 91.10% are truly postdocs.

[40] Recall that these test set observations were not used during any of the model selection steps.

The Table 7 measures of accuracy are unbiased for the tuned random forest model trained on 50% of the UMETRICS subsample, but can be viewed as a conservative estimate of the generalization accuracy expected from a tuned random forest model trained on the full UMETRICS subsample. This is because training on more data typically increases the performance of prediction models. Because of this, we train the tuned random forest model on the entire UMETRICS subsample before generating predictions for the non-UMETRICS subset of the ACS-LEHD Doctorate Panel, and so would like an estimate of the generalization accuracy of this model. Unfortunately, we are not aware of a method to measure this accuracy in an unbiased and computationally-feasible way. Therefore, for simplicity, we estimate the generalization accuracy of the tuned random forest model trained on the full UMETRICS subsample using the OOB accuracy rates generated by this model, noting that this measure of accuracy may be optimistically-biased; we report the results in Table 8. In an informal sense, we can view the accuracy measures in Table 7 and Table 8 as a lower-bound estimate and upper-bound estimate of the generalization accuracy, respectively.

The randomForest package in R has two built-in methods for evaluating the importance of different predictors. The first importance measure is referred to as "mean decrease accuracy" and is described in Breiman (2001). This measure is calculated as follows: first, the OOB accuracy for each tree is recorded.[41] Then, the OOB accuracy for each tree is calculated after randomly permuting the value of each predictor, one predictor at a time; by randomly shuffling a predictor's values in this way, any link between the predictor and postdoc status is effectively broken, and so the OOB accuracy should decrease in proportion to the importance of the variable in prediction. For each predictor, the decrease in OOB accuracy is averaged over all trees and normalized by the standard deviation of these differences. The second measure of predictor importance, called "mean decrease Gini", reports the decrease in the Gini index (a measure of node impurity) from splitting on each predictor, averaged over all trees in the random forest. Figure 7 gives the results for our tuned random forest model trained on the full UMETRICS subsample. As we can see, age and total annual earnings are among the most important predictors, as are the number of quarters that a person spends working for the firm for whom they are employed for the longest duration.

Table 9 compares those observations predicted as postdoc observations to those predicted as nonpostdoc observations in the ACS-LEHD Doctorate Panel. As we can see, the results are similar to those found in Table 1, which reported these statistics for actual postdoc and nonpostdoc observations in the UMETRICS subsample that we use to train our final random forest model: predicted postdoc observations are younger, are more likely to be foreign-born and Asian, earn less than those doctorates that are employed in nonpostdoc positions, and are associated with doctorates who have shorter spells at both the employer for whom they work the most quarters and the employer for whom they work the least quarters.

[41] OOB accuracy = 100% - OOB Error Rate

5 Using the ACS-LEHD Doctorate Panel to Analyze the Effect of Postdoc Training on Future Earnings

Diethorn and Marschke (2022) use linked SDR-SED data to study the impact of postdoc training on the career outcomes of biomedical doctorates, finding that postdoc-trained biomedical doctorates working in for-profit industry earn about 15.8% less than their nonpostdoc-trained counterparts in industry, while those working in academia earn no more or less than their nonpostdoc-trained counterparts. Diethorn and Marschke find no evidence that general ability bias, compensating differentials for tasks performed as part of current employment, seniority, or employer size explains the postdoc salary penalty in industry, instead finding evidence that differences in different types of task-specific human capital explain this earnings disparity. However, the authors note that the employer characteristics available in SDR-SED are quite limited, and that a linked employer-employee dataset of the doctoral workforce would be needed to adequately test whether the industry postdoc salary penalty may in some part be driven by differences in employers. Using the LEHD, Barth et al. (2016) find that the differences in earnings across establishments has increased since the 1970s, and so controlling for employers is likely important when examining earnings-differences across groups of workers.

Here we include a preliminary analysis which augments Diethorn and Marschke (2022) by estimating the following regression specification using the ACS-LEHD Doctorate Panel:

$$\log(earn_{ift}) = \boldsymbol{X_{ift}}\boldsymbol{\beta} + \theta_P POST_{if} + \boldsymbol{\gamma_f} + \boldsymbol{\gamma_t} + \varepsilon_{ift}, \tag{1}$$

where $earn_{ift}$ is the quarterly earnings of doctorate i with a degree in biomedical field f at time t, $POST_{if}$ is an indicator variable for if the individual was ever employed as a postdoc, $\boldsymbol{\gamma_f}$ are field fixed effects, $\boldsymbol{\gamma_t}$ are year-quarter fixed effects, and $\boldsymbol{X_{ift}}$ contains the following limited set of controls: sex, race, foreign-born status, age, age^2, age^3, and age^4. We limit the sample to biomedical doctorates in the ACS-LEHD Doctorate Panel whose earnings are observed between 2001-2015, who were most recently surveyed in the ACS during or after 2009 (when the field of degree variable was first included in the ACS), and who are between the ages of 26 to 60.[42] For each doctorate, we keep only those observations corresponding to quarters after any and all quarters employed as a postdoc since, as in Diethorn and Marschke (2022), we are explicitly interested in how postdoc training impacts *after-postdoc* earnings, as it is common knowledge that postdocs earn less than nonpostdocs during their postdoc employment. The dependent variable $\log(earn_{ift})$ is constructed by adding up the earnings of each individual across all jobs in a given year-quarter, and then taking the natural logarithm of this constructed earnings variable. For each given year-

[42] We classify doctorates as having a PhD in biomedical science if they report any of the following as their primary field of degree in the ACS: Biology, Biochemical Sciences, Botany, Molecular Biology, Ecology, Genetics, Microbiology, Pharmacology, Physiology, Zoology, Epidemiology, Neuroscience, or Miscellaneous Biology. For a full list of available field of degree codes in the ACS, see `https://www2.census.gov/programs-surveys/acs/tech_docs/code_lists/2017_ACS_Code_Lists.pdf?#`.

quarter, we associate workers with the the industry (six-digit 2012 NAICS code) and firm (SEIN) of the job from which they receive the highest earnings. Robust standard errors are clustered at the individual-level (PIK).

Table 10 reports regression results based on equation (1). In column (1), we find that biomedical postdocs appear to face a salary penalty in both academic and nonacademic jobs.[43] Unlike the SDR-SED data used in Diethorn and Marschke (2022), the ACS-LEHD Doctorate Panel includes detailed industry codes, firm identifiers, and finer occupation codes, allowing us to examine how the estimated postdoc salary penalty of biomedical doctorates evolves as we sequentially add fixed effects for industry, firm, and occupation to the control set.[44] Results in Panel A Column (2) show that including industry fixed effects reduces the estimated postdoc salary penalty for the full sample, likely due in part to the fact that postdocs are more likely than nonpostdocs to work in academic jobs which typically pay less than nonacademic jobs (Diethorn and Marschke, 2022). In contrast, Panel C Column (2) shows that the postdoc penalty in nonacademic jobs increases after controlling for industry, suggesting that postdocs finding employment in nonacademic jobs may sort into higher-paying industries than nonpostdoc-trained biomedical doctorates, but that they are still paid less than nonpostdoc-trained doctorates working in the same industry.[45] Column (3) adds firm fixed effects to the specification, which we find reduces all coefficient estimates: the reduction in our full sample estimate is substantial (56% relative to column (1)), although it remains highly statistically significant. Meanwhile,the postdoc salary penalty for academic jobs decreases such that it becomes only marginally significant. The estimated postdoc penalty in nonacademic jobs also decreases, suggesting that postdoc-trained biomedical doctorates may sort into lower-paying firms within a given industry compared to their nonpostdoc-trained counterparts, and that they earn less than nonpostdoc-trained doctorates at the firms where they work. Lastly, including occupation fixed effects in column (4) reduces the postdoc salary penalty for all jobs and for nonacademic jobs, although both remain highly significant.

Altogether, our preliminary results imply that, after controlling for firm and occupation,postdoc-trained biomedical doctorates working in nonacademic jobs are paid about 14.6% less than nonpostdoc-trained biomedical doctorates, while there exists no statistically significant postdoc penalty in academia. While the results in Table 10 suggest that the postdoc penalty on future salary associated with nonacademic jobs is in part explained by the differential sorting of postdoc-trained and nonpostdoc-trained biomedical doctorates across firms and occupations, a sizable postdoc salary penalty in industry still remains to be explained, with the findings in Diethorn and Marschke

[43] We define the academic sector using the NAICS 2012 code 611310 which refers to "Colleges, Universities, and Professional Schools."

[44] Occupation is derived from ACS variable "OCC" and harmonized across ACS years using the crosswalk available at the IPUMS site here: `https://usa.ipums.org/usa/volii/occ_ind.shtml`. Since occupation is measured at a single point in time for each doctorate, there will be measurement error for those whose occupation is different before and/or after the year they were surveyed in the ACS.

[45] The postdoc penalty in academic jobs does not change between columns (1) and (2) since the academic subsample is based on a single industry code.

(2022) suggesting a task-specific human capital based explanation.[46]

While we view these results as useful for exploring the potential sensitivity of the postdoc salary penalty for nonacademic jobs to differential sorting across firms and occupations by postdoc-trained versus nonpostdoc-trained biomedical scientists, we recommend caution in interpreting the magnitudes in Table 10 as representative of the true impact of postdoc training on salary for biomedical doctorates. Beyond the lack of exogenous variationand in postdoc-trained status, there are various shortcomings of our preliminary analysis and some drawbacks associated with the current version of the ACS-LEHD Doctorate Panel. First, we have only included a reduced set of controls in our regression specifications. Second, there is no guarantee that the biomedical doctorates in the ACS-LEHD Doctorate Panel form a representative sample of biomedical doctorates. Additionally, we are constrained in our ability to determine whether certain doctorates in the ACS-LEHD Doctorate Panel have ever obtained postdoc training, which is especially true for the older doctorates in the sample. This is because, for older doctorates, we do not have access to their employment information when they were most likely to have been employed as a postdoc early in their career, and even if we did, UMETRICS data does not cover any years prior to 2001, meaning that we may not be able to reliably predict whether an individual was employed as a postdoc in earlier years. As a result, there are likely many in our sample who were previously employed as postdocs, but for whom we label as never completing a postdoc due to them having completed a postdoc prior to 2001; if it is the case that postdoc-trained biomedical doctorates tend to earn less than their nonpostdoc-trained counterparts, then this source of measurement error would attenuate our estimate of the postdoc penalty.To remedy these issues in the future, we may restrict the sample to younger doctorates to reduce the measurement error in the indicator variable for if a doctorate was ever employed as a postdoc.

We also plan to pursue other sources of data that could improve upon the ACS-LEHD Doctoral Panel. The NSF's Survey of Earned Doctorates has recently become available for RDC researchers, and so we plan to link SED data with LEHD data to construct an SED-LEHD Doctorate Panel. SED data will play a role similar to that of ACS data in the present work, but comes with multiple advantages: First, SED is a census of all doctorates receiving a degree from a US institution, rather than just a sample of doctorates, and so both our sample size and the representativeness of our sample will increase. Second, SED data includes each doctorate's year of graduation (cohort), PhD *alma mater*, and fine field of study, as well as many other background details such as source of funding during PhD studies and postdoc plans. This would allow us to restrict the regression sample based on year of graduation (rather than age), to estimate regressions with PhD cohort fixed effects and PhD university fixed effects, and to measure experience using years since PhD graduation rather than age (age at PhD could then be included as a control variable). Additionally,

[46]More exploration is needed to see if this is a demand-side phenomenon based on the characteristics of the firms where postdocs work, such as R&D intensity, or if this sorting is correlated with other background characteristics such as fine field of study and the PhD university of the doctorate which were controlled for in Diethorn and Marschke (2022).

linking to the subset of SED data that matches with the SDR will allow us to identify postdoc spells in this subset—-we will then carry out our postdoc imputation process on this larger sample, identifying which doctorates present in the SED but not in the SDR are employed as postdocs in each year. Identifying foreign-born members of the ACS-LEHD Doctorate Panel who are *not* in the SED-LEHD Doctorate Panel also opens up the possibility of studying doctorates working in the US but who received their PhD outside the US—an important group given that almost half of all postdocs employed in the US earned their doctorate outside the country (Stephan, 2012).

6 Conclusion

In this paper, we detailed the construction of the ACS-LEHD Doctorate Panel—a linked employer-employee longitudinal dataset of the doctoral workforce that enables researchers to analyze the labor market outcomes of STEM PhD doctorates. This dataset contains demographic information such as age, race, and sex for each individual from the annual ACS files, as well as key quarterly economic information from the LEHD about where each individual works, how much they earn, and how their careers develop over time. By matching a new university-based administrative dataset, UMETRICS, to the ACS-LEHD Doctorate Panel, we were able to implement a machine learning procedure to predict, at a high degree of accuracy, the postdoc status of individuals for whom true postdoc status is unknown. We also compared the prediction performance of our preferred model, random forests, to other predictive models including a linear probability model, logit, and boosted trees, and found that random forests outperformed the standard approaches, as well as achieved slightly better performance than boosted trees.[47] The imputation method used in this paper is sufficiently general to be applied in other research contexts, and we view this method as a way to reliably augment the research capabilities of existing big datasets cheaply and efficiently while avoiding sample loss.

Building on the work of Diethorn and Marschke (2022), we used the ACS-LEHD Doctorate Panel to perform a preliminary analysis of the differences in earnings among postdoc-trained and nonpostdoc-trained biomedical doctorates. Specifically, we focused on how the estimated effect of postdoc training changed as we added industry, firm, and occupation fixed effects, something that was unable to be done in Diethorn and Marschke (2022) due to the lack of employer identifiers in SDR data. We found that the estimated postdoc earnings penalty in nonacademic jobs declined, but was not eliminated, after we added these fixed effects to the control set. In future work, we plan to use this dataset to study a wide-range of pertinent topics including: (1) the returns to education for STEM PhDs and postdocs, and how these differ by demographics, (2) the determinants of STEM labor demand, including an assessment of the complementarity between STEM workers and firm R&D activity, and (3) how the labor mobility of STEM doctorates impacts R&D spillovers, and how the earnings of STEM doctorates depend on measures of their past R&D exposure.

[47] See Appendix B for this comparison.

References

Acemoglu, Daron. 1998. "Why Do New Technologies Complement Skills? Directed Technical Change and Wage Inequality." *The Quarterly Journal of Economics*, 113(4): 1055–1089.

Acemoglu, Daron and David Autor. 2011. "Skills, Tasks, and Technologies: Implications for Employment and Earnings." In *Handbook of Labor Economics*. Vol. 4, Part B, 1043–1171. Elsevier.

Ambroise, Christophe and Geoffrey J. McLachlan. 2002. "Selection Bias in Gene Extraction on the Basis of Microarray Gene-Expression Data." *Proceedings of the National Academy of Sciences*, 99(10): 6562–6566.

Autor, David H., Frank Levy, and Richard J. Murnane. 2003. "The Skill Content of Recent Technological Change: An Empirical Exploration." *Quarterly Journal of Economics*, 118(4): 1279–1333.

Barth, Erling, Alex Bryson, James C. Davis, and Richard Freeman. 2016. "It's Where You Work: Increases in the Dispersion of Earnings across Establishments and Individuals in the United States." *Journal of Labor Economics*, 34(2): S67–S97.

Borjas, George J. 2009. "Immigration in High-Skill Labor Markets: the Impact of Foreign Students on the Earnings of Doctorates." In *Science and Engineering Careers in the United States: An Analysis of Markets and Employment*. 131–161. National Bureau of Economic Research.

Bosch, Anna, Andrew Zisserman, and Xavier Muñoz. 2007. "Image Classification Using Random Forests and Ferns." *IEEE 11th International Conference on Computer Vision*, 1–8.

Breiman, Leo. 1996. "Bagging Predictors." *Machine Learning*, 24: 123–140.

Breiman, Leo. 2001. "Random Forests." *Machine Learning*, 45: 5–32.

Breiman, Leo, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone. 1984. *Classification and Regression Trees*. Chapman & Hall.

Bresnahan, Timothy F., Erik Brynjolfsson, and Lorin M. Hitt. 2002. "Information Technology, Workplace Organization, and the Demand for Skilled Labor: Firm-Level Evidence." *The Quarterly Journal of Economics*, 117(1): 339–376.

Buffington, Catherine, Benjamin Cerf, Christina Jones, and Bruce A. Weinberg. 2016. "STEM Training and Early Career Outcomes of Female and Male Graduate Students: Evidence from UMETRICS Data Linked to the 2010 Census." *American Economic Review: Papers & Proceedings*, 106(5): 333–338.

Cawley, Gavin C. and Nicola L. C. Talbot. 2010. "On Over-Fitting in Model Selection and Subsequent Selection Bias in Performance Evaluation." *Journal of Machine Learning Research*, 11: 2079–2107.

Deming, David J. and Kadeem Noray. 2020. "Earnings Dynamics, Changing Job Skills, and STEM Careers." *Quarterly Journal of Economics*, 135(4): 1965–2005.

Díaz-Uriarte, Ramón and Sara Alvarez de Andrés. 2006. "Gene Selection and Classification of Microarray Data Using Random Forest." *BMC Bioinformatics*, 7(3).

Diethorn, Holden A. 2022. "Green Card Quotas and the Misallocation of Talent: Evidence from the STEM Doctoral Labor Market." Manuscript.

Diethorn, Holden A. and Gerald R. Marschke. 2022. "Task Mismatch and Salary Penalties: Evidence from the Biomedical PhD Labor Market." Manuscript.

Fox, Mary Frank and Paula E. Stephan. 2001. "Career of Young Scientists: Preferences, Prospects, and Realities by Gender and Field." *Social Studies of Science*, 31(1): 109–122.

Freund, Yoav and Robert E. Schapire. 1997. "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting." *Journal of Computer and System Sciences*, 5: 119–139.

Friedman, Jerome H. 2001. "Greedy Function Approximation: A Gradient Boosting Machine." *The Annals of Statistics*, 29(5): 1189–1232.

Ginther, Donna K. and Shulamit Kahn. 2009. "Does Science Promote Women? Evidence from Academia 1973-2001." In *Science and Engineering Careers in the United States: An Analysis of Markets and Unemployment*. 163–194. University of Chicago Press.

Gislason, Pall Oskar, Jon Atli Benediktsson, and Johannes R. Sveinsson. 2006. "Random Forests for Land Cover Classification." *Pattern Recognition Letters*, 27: 294–300.

Goldin, Claudia and Lawrence F. Katz. 1998. "The Origins of Technology-Skill Complementarity." *The Quarterly Journal of Economics*, 113(3): 693–732.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition.* Springer.

Institute for Research on Innovation & Science. 2019. "Summary Documentation for the IRIS UMETRICS 2019 Data Release."

Institute for Research on Innovation & Science. 2022. "Summary Documentation for the IRIS UMETRICS 2022 Data Release."

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2013. *An Introduction to Statistical Learning with Applications in R.* Springer.

Kahn, Shulamit and Donna K. Ginther. 2017. "The Impact of Postdoctoral Training on Early Careers in Biomedicine." *Nature Biotechnology*, 35(1): 90–94.

Kahn, Shulamit and Megan MacGarvie. 2019. "The Impact of Permanent Residency Delays for STEM PhDs: Who Leaves and Why." *Research Policy*, In Press.

Kaiser, David. 2005. *Drawing Theories Apart: The Dispersion of Feynman Diagrams in Postwar Physics.* University of Chicago Press.

Khosla, Pooja. 2018. "Wait Time for Perminent Residency and the Retention of Immigrant Doctoral Recipients in the U.S." *Economic Analysis and Policy*, 57: 33–43.

Kim, Ji-Hyun. 2009. "Estimating Classification Error Rate: Repeated Cross-Validation, Repeated Hold-Out and Bootstrap." *Computational Statistics and Data Analysis*, 53: 3735–3745.

Kleinberg, Jon, Himabindu Lakkaraju, Jure Leskovec, Jens Ludwig, and Sendhil Mullainathan. 2018. "Human Decisions and Machine Predictions." *The Quarterly Journal of Economics*, 133(1): 237–293.

Kuhn, Max. 2008. "Building Predictive Models in R Using the caret Package." *Journal of Statistical Software*, 28(5).

Kuhn, Max and Kjell Johnson. 2013. *Applied Predictive Modeling.* Springer.

Lahiri, Kajal and Liu Yang. 2013. "Forecasting Binary Outcomes." In *Handbook of Economic Forecasting.* Vol. 2, Part B, 1025–1106. Elsevier.

Lane, Julia I., Jason Owen-Smith, Rebecca F. Rosen, and Bruce A. Weinberg. 2015. "New Linked Data on Research Investments: Scientific Workforce, Productivity, and Public Value." *Research Policy*, 44: 1659–1671.

Liaw, Andy and Matthew Wiener. 2002. "Classification and Regression by randomForest." *R News*, 2(3): 18–22.

Mullainathan, Sendhil and Jann Spiess. 2017. "Machine Learning: An Applied Econometric Approach." *Journal of Economic Perspectives*, 31(2): 87–106.

Mullainathan, Sendhil and Ziad Obermeyer. 2017. "Does Machine Learning Automate Moral Hazard and Error?" *American Economic Review: Papers & Proceedings*, 107(5): 476–480.

Mulrow, Edward, Ali Mushtaq, Santanu Pramanik, and Angela Fontes. 2011. "Assessment of the U.S. Census Bureau's Person Identification Validation System." NORC at the University of Chicago.

Polanyi, Michael. 1958. *Personal Knowledge: Towards a Post-Critical Philosophy.* University of Chicago Press.

Polanyi, Michael. 1966. *The Tacit Dimension.* Doubleday.

Ridgeway, Greg. 2007. "Generalized Boosted Models: A Guide to the gbm Package."

Stephan, Paula. 2006. "Wrapping It Up in a Person: The Mobility Patterns of New PhDs." *Innovation Policy and the Economy*, 7: 71–98.

Stephan, Paula. 2012. *How Economics Shapes Science.* Harvard University Press.

Tibshirani, Ryan J. and Robert Tibshirani. 2009. "A Bias Correction for the Minimum Error Rate in Cross-Validation." *The Annals of Applied Statistics*, 3(2): 822–829.

Varma, Sudhir and Richard Simon. 2006. "Bias in Error Estimation when Using Cross-Validation for Model Selection." *BMC Bioinformatics*, 7(91).

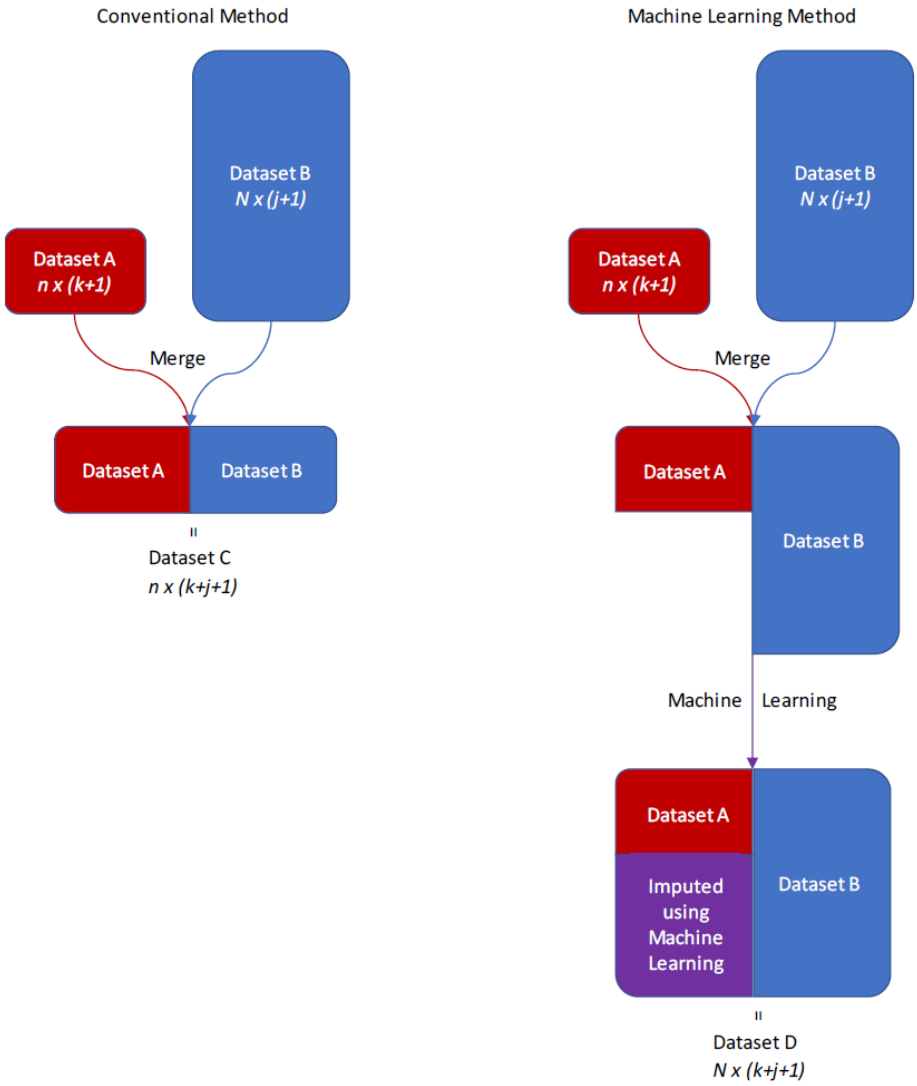
Vilhuber, Lars. 2018. "LEHD Infrastructure S2014 Files in the FSRDC." CES Working Paper No. 18-27.

Youden, W. J. 1950. "Index for Rating Diagnostic Tests." *Cancer*, 3(1): 32–35.

Zolas, Nikolas, Nathan Goldschlag, Ron Jarmin, Paula Stephan, Jason Owen-Smith, Rebecca F. Rosen, Barbara McFadden Allen, Bruce A. Weinberg, and Julia I. Lane. 2015. "Wrapping It Up in a Person: Examining Employment and Earnings Outcomes for Ph.D. Recipients." *Science*, 350(6266): 1367–1371.

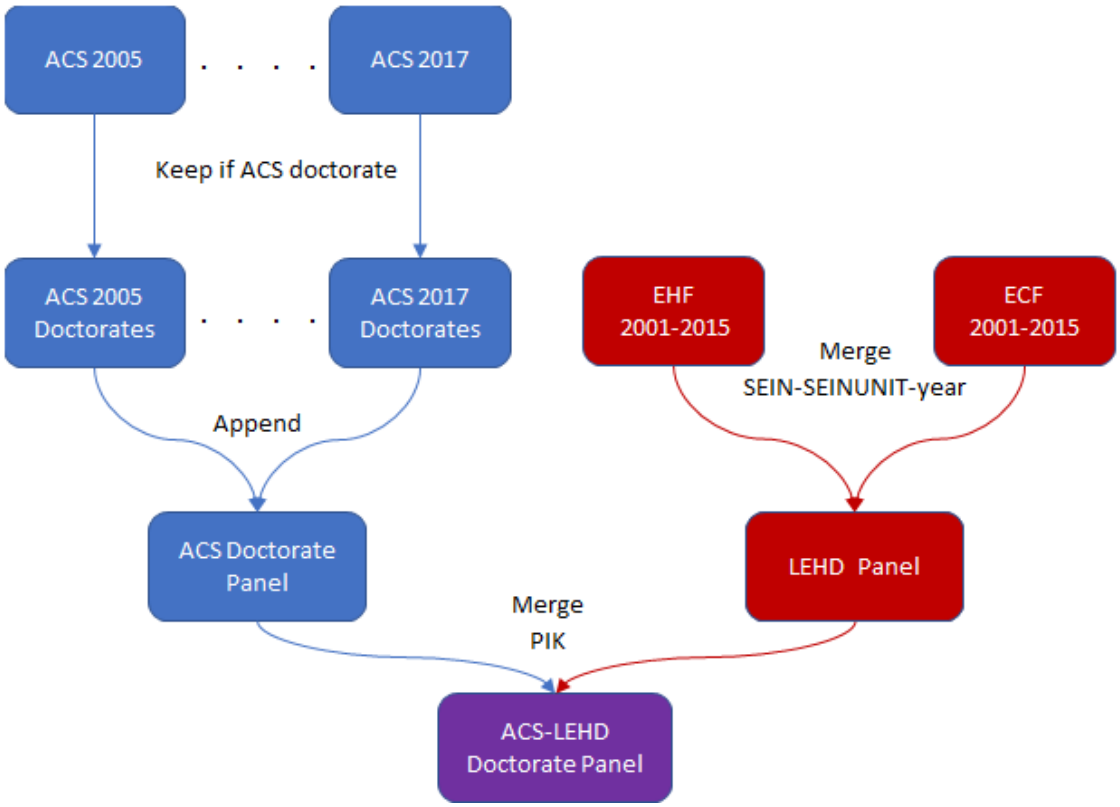
Figures

Figure 1: Conventional and Machine Learning Methods of Merging Datasets of Disparate Size



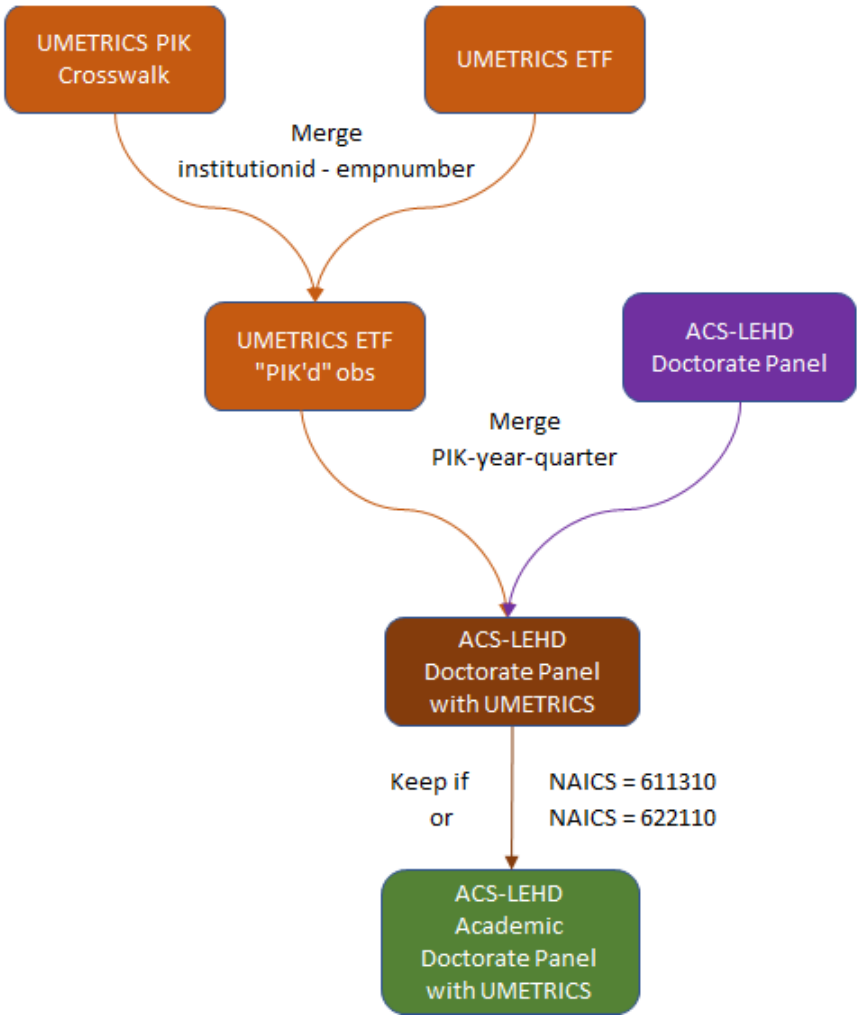
Notes: Figure 1 shows the conventional method of merging two datasets compared to our method using machine learning to impute missing variables. For simplicity, we assume that Dataset A shares a single unique identifier with Dataset B, that each observation in Dataset A matches to an observation in Dataset B, that $N > n$, and that the j variables in Dataset B are distinct from the k variables in Dataset A (the "+1" variable in each dataset is the common unique identifier). The conventional method involves merging the two datasets and only keeping matched observations, whereas the machine learning method merges the two datasets and then uses variables from Dataset B to impute key variables from Dataset A to avoid dropping unmatched observations from Dataset B. The efficacy of the machine learning method depends on the extent to which the variables in Dataset B are predictive of the variables to be imputed from Dataset A and the extent to which the observations in Dataset A are representative of those contained only in Dataset B.

Figure 2: Creation of ACS-LEHD Doctorate Panel



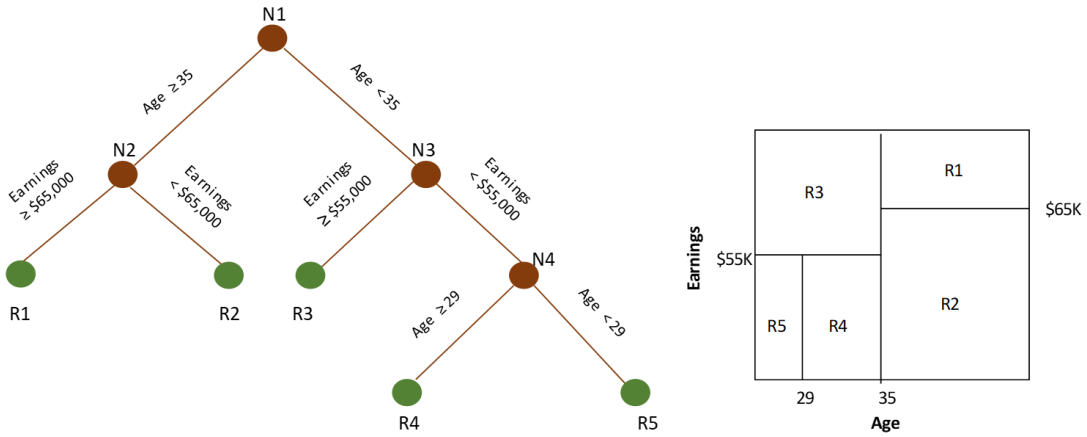
Notes: Figure 2 shows how we construct the ACS-LEHD doctorate panel. For the American Community Survey (ACS) for years 2005-2017 we keep only those observations associated with respondents with a doctorate degree. We then append these yearly ACS doctorate datasets to form an ACS Doctorate Panel. For the LEHD data, we merge the Employment History File (EHF) with the Employer Characteristics File (ECF) for years 2001-2015. We then merge the LEHD panel with the ACS doctorate panel to form the ACS-LEHD Doctorate Panel which is unique on PIK-SEIN-SEINUNIT-year-quarter.

Figure 3: Creation of ACS-LEHD Academic Doctorate Panel with UMETRICS



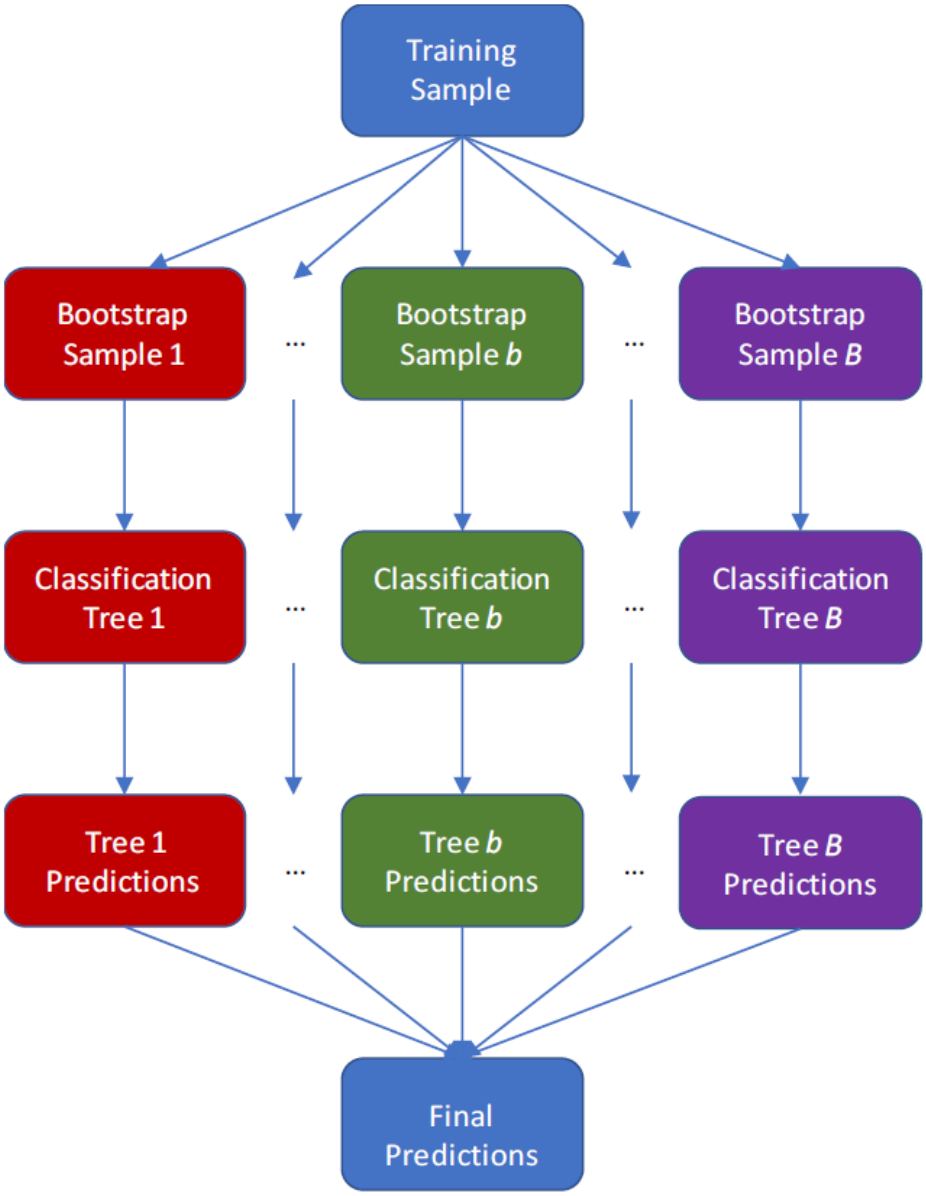
Notes: Figure 3 shows how we construct the ACS-LEHD Academic Doctorate Panel with UMETRICS. For the quarterly UMETRICS Employee Transaction File (ETF), we obtain PIKs from merging with the UMETRICS PIK crosswalk and then keep only those observations associated with UMETRICS employees who are "PIK'd". We convert this transaction-level file to a PIK-year-quarter level file and then merge this dataset with the ACS-LEHD Doctorate Panel to form our ACS-LEHD Panel with UMETRICS. We then create the predictor variables listed in Table A.1. Lastly, we restrict observations to year-quarters where a person is employed in "Colleges, Universities, and Professional schools" (NAICS=611310) or "General Medical and Surgical Hospitals" (NAICS=622110) since these represent academic sectors that provide postdoc positions. We then make the dataset unique on PIK-year-quarter.

Figure 4: A Classification Tree and Its Equivalent Predictor-Space Representation



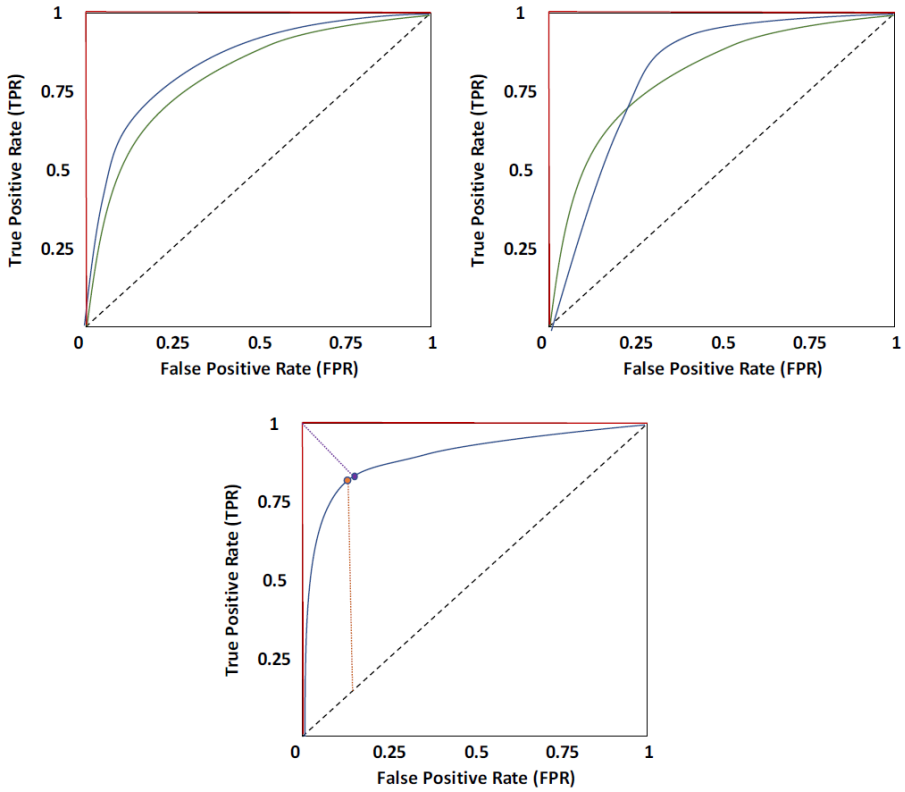
Notes: Figure 4 shows two equivalent representations of a classification tree with two predictors. In the example, employment as a postdoc is predicted using age and current earnings.

Figure 5: Schematic Representation of Bootstrap Aggregated ("Bagged") Classification Trees



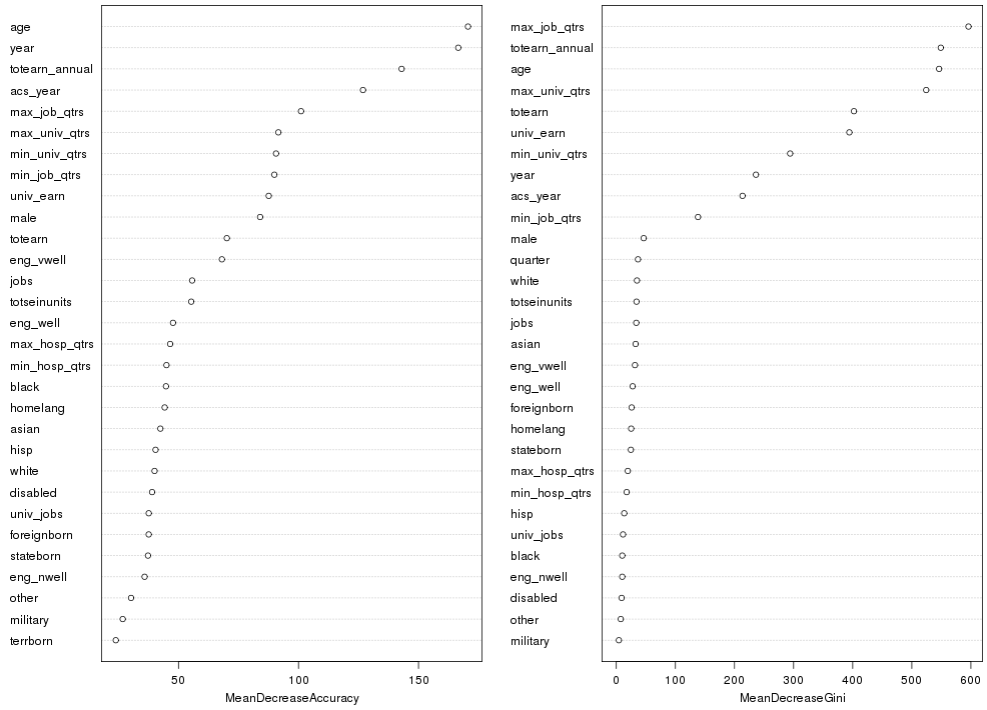
Notes: Figure 5 gives a visual representation of "bagging" classification trees. The process is as follows: a training sample of data is sampled with replacement B times creating B bootstrap samples where each bootstrap sample b is the same size as the original training sample. A classification tree is then fit to each of the B bootstrapped samples. The predicted probability that an out-of-sample observation falls in a given class is generated by the bagged tree model as follows: Each of the B classification trees in the bagged tree model generate a predicted class for the out-of-sample observation. After each tree has generated a prediction, the results are averaged to produce a probability that the observation is of the given class. The researcher then chooses a cutoff probability for prediction of a given class to generate a final prediction.

Figure 6: Receiving Operator Characteristic (ROC) Plots



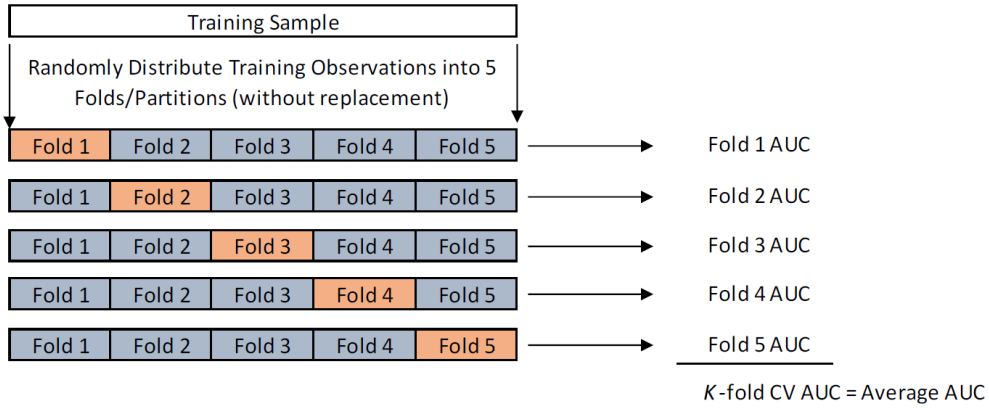
Notes: Figure 6 shows examples of ROC curves that can be used to judge the prediction performance of a classification algorithm. Each point on an ROC curve gives the (FPR, TPR) combination achieved by a particular probability threshold used for prediction; points farther to the right along a given ROC curve correspond to lower probability cutoffs. The dotted diagonal line in each plot represents the performance expected using random guessing for prediction. The connected red lines touching the border represents the performance of a perfect predictive model since it intersects with point (0,1) in ROC space, which is associated with a 100% TPR and 0% FPR. The top-left panel shows two ROC curves, one for each of two different hypothetical prediction models. The model corresponding to the blue ROC curve unambiguously outperforms the model corresponding to the green ROC curve since the "blue model" achieves a higher TPR for every given FPR when compared to the "green model". The top-right panel shows another set of two ROC curves corresponding the hypothetical prediction models. Here, it is uncertain from inspection which model performs best since the ROC curves intersect, with the green model outperforming the blue model for high thresholds but underperforming the blue model for lower thresholds. In this case, calculation of the area under the ROC curve (AUC) for each model is needed to judge which model performs best globally. The bottom panel shows an ROC curve associated with a single hypothetical prediction model. In cases where there is class imbalance, the default cutoff of 0.5 is likely to overpredict the most commonly occurring class, and so the researcher may want to consider alternative cutoffs that trade-off some overall classification accuracy in exchange for more accurate prediction of the least commonly occurring class. Two possible cutoff choices are the "top-left" cutoff that minimizes the sum of squared false positive and false negative rates (FPR2 + FNR2) — represented by the purple point on the ROC curve – or the "Youden" cutoff that maximizes the sum of true positive and true negative rates (TPR + TNR) — represented by the orange point on the ROC curve.

Figure 7: Random Forest Predictor Importance Measures



Notes: Figure 7 shows the importance of predictors using two alternative measures of predictor performance for random forest models. The left panel measures predictor importance using what is in the randomForest package as "mean decrease accuracy" and is described in Breiman (2001). The method works as such: first, the OOB accuracy for each tree is recorded. Then, the OOB accuracy for each tree is calculated after randomly permuting the value of each predictor, one predictor at a time; by randomly shuffling a predictor's values in this way, any link between the predictor and postdoc status is effectively broken, and so the OOB accuracy should decrease in proportion to the importance of the variable in prediction. For each predictor, the decrease in OOB accuracy is averaged over all trees and normalized by the standard deviation of these differences. The right panel measures predictor importance using "mean decrease Gini", which reports the decrease in the Gini index, a measure of node impurity, from splitting on each predictor, averaged over all trees in the random forest.

Figure 8: K-Fold Cross-Validation (CV) Method for Calculating AUC with $K = 5$



Notes: "Fold k AUC" is the AUC calculated by first training a machine learning model on all observations not in the k^{th} fold (i.e. the blue folds), and then using this model to predict the classes of fold k observations (the beige fold); such predictions can be used to graph an ROC curve and calculate the area under the curve.

Tables

Table 1: Summary Statistics by Postdoc Status for the UMETRICS Subset of ACS-LEHD Doctorate Panel

	Postdoc		Not Postdoc	
Predictor	**Mean**	**S.D.**	**Mean**	**S.D.**
Age	33.78	4.620	48.46	11.35
Earnings	10060	4237	27030	22460
Male	0.5915	0.4917	0.6449	0.4786
Foreign-born	0.5451	0.4981	0.3477	0.4763
White	0.5957	0.4908	0.8140	0.3891
Asian	0.3483	0.4765	0.1602	0.3668
Black	0.01841	0.1345	0.01592	0.1252
Hispanic	0.06598	0.4765	0.0307	0.1725
Other	0.03759	0.1902	0.009853	0.09877
Jobs	1.171	0.4209	1.169	0.4330
Max_job_qtrs	20.44	8.857	60.16	28.96
Min_job_qtrs	11.21	8.399	30.12	34.90
	N = 2,600 (450)		N = 16,000 (1,500)	

Notes: This table reports summary statistics for the person-quarter observations in the UMETRICS subset of the ACS-LEHD Academic Doctorate Panel. Unique person counts are given in parentheses. Obervation and person counts are rounded according to Census disclosure requirements. See Table A.1 for definition of predictors.

Table 2: Confusion Matrix

Predicted	**Actual**	
	Not Postdoc	**Postdoc**
Not Postdoc	True Negative (TN)	False Negative (FN)
Postdoc	False Positive (FP)	True Positive (TP)

Notes: Table 2 shows the structure of a confusion matrix which is used to report the number of true negatives (TN), true positives (TP), false negatives (FN) and false positives (FP) produced by a classification prediction model. These counts can then be used to calculate the measures of classification error and accuracy listed and defined in Table 3.

Table 3: Accuracy and Error Measures

Name of Measure	**Definition**
Accuracy	(TP + TN) / (TP + TN + FP + FN)
Misclassification/Error Rate	(FP + FN) / (TP + TN + FP + FN) $\equiv$ 1 - Accuracy
True Positive Rate (TPR)	TP / (TP + FN)
False Positive Rate (FPR)	FP / (TN + FP)
True Negative Rate (TNR)	TN / (TN + FP) $\equiv$ 1 - FPR
False Negative Rate (FNR)	FN / (TP + FN) $\equiv$ 1 - TPR
Positive Predictive Value (PPV)	TP / (TP + FP)
Negative Predictive Value (NPV)	TN / (TN + FN)
F1-score (F1)	2 * (TPR * PPV)/(TPR + PPV)

Notes: Table 3 gives the definition of different measures of classification error and accuracy based on the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) generated by a classification prediction model. Refer to Table 2 for the definition of TP, TN, FP, and FN.

Table 4: Random Forest Model Selection (Steps 1-4), Assessment (Step 5), and Prediction (Step 6)

1. Partition data into a training set and test set (50% - 50% split).
2. For random forest models with different number of splitting variables:
   (a) Train model on the training set.
   (b) Calculate AUC using OOB predictions.
3. Select random forest model that performs best in terms of AUC using OOB predictions.
4. Identify alternative cutoffs/thresholds based on OOB prediction performance.
5. Estimate generalization error using the test set.
6. Retrain selected model on all labeled (UMETRICS) data and use to predict postdoc status for all non-labeled (non-UMETRICS) observations.

Table 5: Random Forest AUC for Different Hyperparameter Values Using Training Set OOB Predictions

	Number of Splitting Variables (m)			
	$\boldsymbol{\sqrt{p}}$	$\boldsymbol{p/3}$	$\boldsymbol{p/2}$	$\boldsymbol{p}$
AUC	0.9949	0.9942	0.9940	0.9934

Notes: Table 5 reports the area under the ROC curve (AUC) for four random forest models, each with a unique value for the number of splitting variables (m) considered at each tree node. The first three values considered are chosen following the exposition in James et al. (2013) whom compare the default value of $m = \sqrt{p}$, where p is the total number of predictors, with $m = p/2$ and $m = p$ (bagged trees). We also consider $m = p/3$, which corresponds to the suggested default value for random forests in a regression context (Hastie, Tibshirani, and Friedman, 2009) and puts the number of splitting variables roughly halfway between $m = \sqrt{p}$ and $m = p/2$ in our application. In all four cases, we round the number of splitting variables to the integer value closest to these targeted values.

Table 6: Random Forest Accuracy by Cutoff Using Training Set OOB Predictions

Cutoff		**Accuracy**						
Type	**Value**	**Total**	**TPR**	**TNR**	**PPV**	**NPV**	**F1**	**AUC**
F1	0.4825	97.46%	91.10%	98.52%	91.10%	98.52%	91.10%	0.9949
Youden	0.3298	96.62%	97.11%	96.53%	82.33%	99.50%	89.11%	

Notes: Table 6 shows the prediction performance of a tuned random forest model for alternative probability thresholds used for prediction of OOB observations. These results can be used to select the cutoff with the error rates most desirable for the research question at hand. The "F1" cutoff which corresponds to the threshold that maximizes the model's F1-score. The "Youden" cutoff maximizes the Youden Index: TPR + TNR - 1.

Table 7: Random Forest Accuracy using Test Set Predictions

Accuracy						
Total	**TPR**	**TNR**	**PPV**	**NPV**	**F1**	**AUC**
97.14%	89.24%	98.42%	90.22%	98.25%	89.73%	0.9929

Notes: Table 7 shows the prediction performance of our tuned random forest model using test set predictions. These results can be used to estimate the generalization error of the prediction model since the test set observations were not used for model selection. However, prediction models often improve performance with greater sample sizes, and since the model here is trained on 50% of the available data (the training set), we may expect better performance when building the model using the full sample.

Table 8: Random Forest Accuracy Using OOB Predictions from Random Forest Trained on Full Data

Accuracy						
Total	**TPR**	**TNR**	**PPV**	**NPV**	**F1**	**AUC**
98.21%	93.90%	98.92%	93.47%	99.00%	93.69%	0.9969

Notes: Table 8 shows the prediction performance of our tuned random forest model fit on the full UMETRICS subsample using OOB predictions. The random forest model assessed here is fit to the full sample of data, rather than 50% of the sample as in Table 6 and Table 7, and so part of the increased performance is likely due to the increase in sample size. However, these results will give an optimistically-biased measure of the generalization error of the prediction model since observations used in model selection (i.e. 50% of the observations which formed the training sample) are also used in generating the measures of prediction performance in this table. Therefore, we view the results in this table as an upper-bound on the prediction performance of this prediction model.

Table 9: Summary Statistics by Postdoc Prediction for the ACS-LEHD Doctorate Panel

	Postdoc		Not Postdoc	
Predictor	**Mean**	**S.D.**	**Mean**	**S.D.**
Age	33.34	5.359	47.81	12.17
Earnings	8790	4686	17860	20880
Male	0.5487	0.4976	0.5941	0.4911
Foreign-born	0.4337	0.4956	0.1937	0.3952
White	0.6442	0.4788	0.8491	0.3580
Asian	0.3020	0.4591	0.09445	0.2925
Black	0.03184	0.1756	0.04474	0.2067
Hispanic	0.05611	0.2301	0.03103	0.1734
Other	0.02193	0.1465	0.01175	0.1077
Jobs	1.188	0.4526	1.361	0.7418
Max_job_qtrs	20.08	7.228	57.73	27.41
Min_job_qtrs	10.36	8.212	24.01	30.71
	N = 49,500 (8,600)		N = 2,413,000 (90,000)	

Notes: This table reports summary statistics for the person-quarter observations in the ACS-LEHD Doctorate Panel. Unique person counts are given in parentheses. Obervation and person counts are rounded according to Census disclosure requirements. See Table A.1 for definition of predictors.

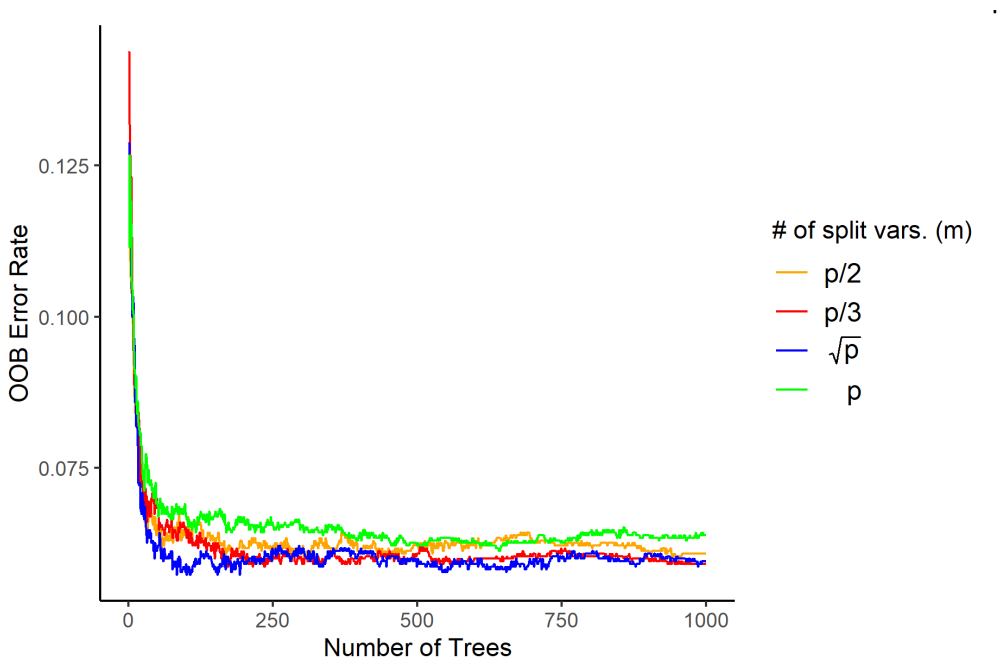
Table 10: Impact of Postdoc Training on After-Postdoc Salary Using the ACS-LEHD Doctorate Panel

Dependent Variable: log(earn)	(1)	(2)	(3)	(4)
Panel A. Full Sample (N = 651,000)				
Postdoc Training	-0.2710***	-0.2075***	-0.1192***	-0.0807***
	(0.0306)	(0.0288)	(0.0246)	(0.0243)
R^2	0.136	0.254	0.604	0.625
Panel B. Academic (N = 100,000)				
Postdoc Training	-0.1097***	-0.1097***	-0.0649*	-0.0224
	(0.0373)	(0.0373)	(0.0354)	(0.0353)
R^2	0.187	0.187	0.306	0.378
Panel C. Nonacademic (N = 551,000)				
Postdoc Training	-0.2294***	-0.2450***	-0.1976***	-0.1455***
	(0.0417)	(0.0388)	(0.0316)	(0.0315)
R^2	0.136	0.261	0.639	0.657
Fixed Effects				
Field	✓	✓	✓	✓
Year-Quarter	✓	✓	✓	✓
Industry (NAICS)		✓		
Firm			✓	✓
Occupation				✓

Notes: This table reports regressions results based on equation (1) where our sample includes all biomedical doctorates in the ACS-LEHD Doctorate Panel between 2001-2015 who were most recently surveyed in the ACS during or after 2009 (when the field of degree variable was first included in the ACS) and who are between the ages of 26 to 60. For each doctorate, we keep only those observations corresponding to quarters after any and all quarters employed as a postdoc (according to predictions from our random forest model). Industry is indicated by six-digit NAICS 2012 code. The academic sector is defined by the NAICS code 611310 which refers to "Colleges, Universities, and Professional Schools." Firm is defined by SEIN. Occupation is derived from ACS variable "OCC" and harmonized across ACS years using the crosswalk available at the IPUMS site here: `https://usa.ipums.org/usa/volii/occ_ind.shtml`. Robust standard errors clustered at individual-level (PIK) are in parentheses. Specifications (1) - (4) include the following controls: age, age^2, age^3, age^4, sex, race (i.e. black, asian, or other), foreign-born status, and year-quarter fixed effects. Dependent variable constructed by summing the earnings of each individual across all their jobs in a given year-quarter, and then taking the natural logarithm of this constructed earnings variable.
* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

A Supplementary Figures and Tables

Figure A.1: OOB Error Rate of Random Forest Models with Different Number of Splitting Variables



Notes: Figure A.1 shows the Out-Of-Bag (OOB) Error Rate of four random forest prediction models, each with a different value given for the random number of splitting variables considered at each node in each classification tree, for a different number of trees. This plot is useful for determining the number of trees that should be used for the random forest models. The researcher will want to select the number of trees based on a level where the OOB error stabilizes for all of the models considered. As we can see, a higher number of trees does not appear to lead to overfitting, so the researcher can be generous in selecting the number of trees, keeping in mind that random forest models with a greater number of trees require greater computational resources. Figure A.1 is based on data from the "Spambase Data Set" that can be found at `https://archive.ics.uci.edu/ml/datasets/spambase` or easily accessed via the R package "ElemStatLearn."

Table A.1: Variables in ACS-LEHD Academic Doctoral Panel with UMETRICS

Variable Name	Variable Definition
postdoc[1]	If occupational class = "Post Graduate Researcher", then postdoc = 1; else = 0.
Year	Year between 2002-2014
Quarter	Quarter between 1-4
age[2]	Year - birth year
male[2]	If male, then male = 1; otherwise, male = 0
white[2]	If white, then white = 1; otherwise, white = 0
black[2]	If black, then black = 1; otherwise, black = 0
native[2]	If Native American, then native = 1; otherwise, native = 0
asian[2]	If Asian, then asian = 1; otherwise, asian = 0
hispanic[2]	If Hispanic, then hispanic = 1; otherwise, hispanic = 0
other[2]	If other race, then other = 1; otherwise, other = 0
stateborn[2]	Born in US State
terrborn[2]	Born in US Territory
foreign[2]	Foreign born
homelang[2]	Speaks another language at home
eng[2]	English speaking ability: 1= Very Well, 2 = Well, 3 = Not well, 4 = Not at all.
military[2]	Ever serve in military?
disable[2]	Report a disability?
univ[3]	Employed in a NAICS = 611310 job (Colleges, University, and professional Schools) during quarter
hosp[3]	Employed in a NAICS = 622110 job (General Medical and Surgical Hospitals) during quarter
univ_earn[3]	Quarterly earnings across all NAICS = 611310 jobs
hosp_earn[3]	Quarterly earnings across all NAICS = 622110 jobs
totearn[3]	Quarterly earnings across all jobs
totearn_annual[3]	Annual earnings across all jobs
jobs[3]	Total number of jobs during quarter as counted by number of SEINs
totseinunits[3]	Total number of jobs during quarter as counted by number of SEINUNITs
univ_jobs[3]	Total number of NAICS = 611310 jobs during quarter (SEIN)
hosp_jobs[3]	Total number of NAICS = 622110 jobs during quarter (SEIN)
max_univ_qtrs[3]	maximum # of quarters spent in a single SEIN where NAICS = 611310
min_univ_qtrs[3]	minimum # of quarters spent in a single SEIN where NAICS = 611310
max_hosp_qtrs[3]	maximum # of quarters spent in a single SEIN where NAICS = 622110
min_hosp_qtrs[3]	minimum # of quarters spent in a single SEIN where NAICS = 622110
max_job_qtrs[3]	maximum # of quarters spent in a single SEIN
min_job_qtrs[3]	minimum # of quarters spent in a single SEIN

Notes: Superscripts indicate data sources used to create variable: 1 = UMETRICS, 2 = ACS, 3 = LEHD. The variable "postdoc" is only available for the UMETRICS subset of our ACS-LEHD Doctorate Panel.

B Comparison of Random Forests with Other Predictive Models

Since no single machine learning algorithm dominates all others across all applications (James et al., 2013), it is useful to compare the performance of our random forest model with other prediction models. One popular alternative to random forests is known as boosted trees. Boosting, like bagging, is an ensemble method based on averaging predictions across many simple learners such as classification trees.[48] However, these two approaches differ in several aspects. First, with bagged trees, each tree is grown to be large, while in boosting, typically trees with only a few splits each are grown. A second and more significant difference is that with bagging, each tree is grown independent of the other trees in the ensemble, whereas with boosting, trees are grown sequentially, with each tree's structure depending on the structure of the trees before it. Specifically, each successive tree in a boosted trees model places more weight on correctly predicting observations for which previous trees in the ensemble performed poorly. The predictions of the model are updated as each tree is grown, with more weight being applied to trees that achieve greater accuracy. The rate at which this updating occurs is controlled by a shrinkage parameter. Altogether, boosted trees contain three hyperparameters that the user must tune: the number of trees, the size of each tree ("interaction depth"), and the rate of learning across trees ("shrinkage parameter"). Typically, the choice of a smaller shrinkage parameter will necessitate growing a larger number of trees, and in practice, Hastie, Tibshirani, and Friedman (2009) suggest choosing the size of the trees to be such that the number of terminal nodes is around 6, finding that variation in the size of the trees seldom provides significant improvement.[49] Gradient boosted machines, which are a generalization of boosted trees introduced in Friedman (2001), are implemented in the R package gbm (Ridgeway, 2007).

To compare different types of machine learning models, we adopt the methods of model selection and assessment outlined in Table B.1. While similar to the strategy in Table 4, there are two main differences. First, we partition the UMETRICS subsample into three sets (a training set, validation set, and test set) rather than two sets (a training set and test set).[50] The training set is used to train and tune each individual machine learning model, the validation set is used to compare different machine learning models and to identify alternative cutoffs, and the test set is used to estimate the generalization error of our selected model. The second difference is that we use repeated K-fold cross-validation (CV) to tune the different machine learning models.[51] K-fold CV works as follows: 1) the training set is partitioned into K folds, 2) For each fold k: the model is trained on all folds except fold k, and then predictions are made for fold k and the AUC is calculated, 3) the

[48] Our description of boosting is based on the Adaboost algorithm developed in Freund and Schapire (1997).

[49] A tree with 6 terminal nodes allows for fifth-order interactions between the predictors.

[50] Hastie, Tibshirani, and Friedman (2009) state that it is too difficult to give a general rule for how to partition the data, but that a typical split might be 50%-25%-25%. The validity of our method does not hinge on the choice of data partitions.

[51] While we could again use AUC calculated from ROC curves based on OOB observations instead of K-fold CV to tune our random forest model, we choose the latter so as to follow the general strategy put forth in Table B.1 which can be applied for any machine learning model, including those not considered here, e.g., support vector machines, neural networks, etc.

K AUC calculations are averaged to obtain a single CV estimate of AUC. We give a schematic representation of K-fold CV in Figure 8. The calculated K-fold AUC depends on the partitioning of the original data, and so one way to reduce this source of variance is to repeat K-fold CV multiple times.[52] We use the R package caret (Kuhn, 2008) to perform 10-fold CV repeated 5 times to tune our random forest and boosted trees models. In addition to random forests and boosted trees, we also consider the predictive performance of a linear probability model (LPM) and a logit model where each predictor enters the model additively with no interaction terms.[53]

We tune our random forest model over the same values of the splitting variables that we considered in the previous section. For boosted trees, we tune over combinations of the following parameter values: Number of trees = $\{5000, 8000, 11000\}$, shrinkage rate = $\{0.001, 0.01, 0.1\}$, and interaction depth = $\{1, 2, 3\}$. As before, we find that the random forest model with $\sqrt{p}$ splitting variables performs best amongst the random forest models considered, and the boosted trees model with {number of trees, shrinkage rate, interaction depth} = $\{8000, 0.10, 3\}$ performs best amongst the boosted trees models considered, and so these two models represent our tuned random forest model and boosted trees model, respectively.[54]

Next, we compare the tuned random forest model, tuned boosted trees model, logit model, and linear probability model directly by calculating the AUC of each method when applied to the validation set; we report these values in the last column of Table B.2. While we believe picking the predictive model that achieves the highest AUC is a prudent approach, especially when considering many models, we also show the accuracy rates for alternative cutoffs for each of the four models. As we can see, boosted trees and random forests outperform both the LPM and logit models, likely due to the fact that these tree-based models can automatically capture complex interaction effects in the data without these interactions needing to be prespecified by the researcher. We also see that random forests and boosted trees perform quite similarly, with the tuned random forest model slightly outperforming the tuned boosted trees model in terms of AUC. Because of this, and since boosted trees are more difficult and computationally intensive to tune, we favor the random forest model over boosted trees. Additionally, and choose the F1 cutoff over the Youden cutoff due to the better balance of TPR and PPV associated with the F1 cutoff.

Similar to Table 6, the accuracy measures in Table B.2 are optimistically-biased estimates of

[52] Kim (2009) compares repeated 10-fold CV to other methods of comparative computational requirements and recommends repeated CV for general use.

[53] Part of the value of boosted trees and random forests is that the researcher does not need to specify interaction terms that may be useful for prediction *ex ante*, and so we view these as natural baseline comparisons, while recognizing that the logit and LPM models could be improved at the cost of more effort required by the researcher relative to using an automated machine learning method.

[54] Tuning using 10-fold CV repeated 5 times is computationally intensive, and so we only test a restricted set of hyperparameter combinations. Nevertheless, this still necessitates training 27 boosted trees models. Small trees typically perform well in boosted trees models, and so we only consider trees of size 1-3. The shrinkage parameters are standard values (e.g., see examples in Hastie, Tibshirani, and Friedman (2009) and James et al. (2013)) and the number of trees were selected so as to find an internal solution for the number of trees selected in the tuned model given the shrinkage rates and tree sizes considered; that is, if we would have found that 11000 trees performed better than 8000, we would have then added another value for the number of trees considered (e.g., 14000 trees) to test if it performed better.

the generalization accuracy of each model since selection of the random forest model was chosen based on its validation set performance. To get a measure of the generalization accuracy, we use the tuned random forest model with a cutoff of 0.4680 to predict the postdoc status of the test set observations, and then calculate the accuracy measures for these test set predictions. Table B.3 displays these results.

As in Table 7, we view Table B.3 measures of accuracy as conservative estimates of the generalization accuracy expected of a model trained on the *full* UMETRICS subsample.[55] Therefore, we estimate the accuracy of a random forest model trained on the entire UMETRICS subsample by using the OOB accuracy of such a model, noting that this measure of accuracy is likely optimistically-biased. We informally view the accuracy measures in Table B.3 and Table B.4 as a lower-bound estimate and upper-bound estimate of the generalization accuracy, respectively.

The results in Table B.3 and Table B.4 are similar to those in Table 7 and Table 8, respectively, but there are some differences emanating from the fact that the F1 cutoff obtained when using OOB predictions from the training set (0.4825) is higher than the F1 cutoff obtained when using validation set predictions (0.4680), leading now to a higher TPR and lower PPV relative to the final model obtained in Section 4.2. Another difference is that the test dataset used to produce Table B.3 results was one-half the size of the test set used for Table 7 because, unlike before, we had to reserve some data to form a validation set.

Overall, random forests compare favorably to the other prediction models considered in this section. Random forests performed significantly better than the LPM and logit model and slightly better than boosted trees. Another advantage of random forests over boosted trees is that they are considerably easier and less computationally intensive to tune. We may have been able to obtain a boosted trees model that would have marginally outperformed random forests by tuning over more hyperparameter values, but given the already high performance of the tuned random forest model, the expected return to doing so is small.

[55] These measures are unbiased estimates of a model trained on *50%* of the UMETRICS subsample represented by the training sample.

Table B.1: Machine Learning Model Selection (Steps 1-4), Assessment (Step 5), and Prediction (Step 6)

1. Partition data into a training set, validation set, and test set (50% -25%-25% split).
2. For each machine learning algorithm:
   (a) Train model on the training set using different values of hyperparameters.
   (b) Use repeated CV to estimate AUC for different hyperparameter combinations.
   (c) Output model that performs the best in terms of AUC as measured by repreated CV.
3. Select machine learning model that performs best in terms of AUC using validation set predictions.
4. Identify alternative cutoffs/thresholds based on validation set prediction performance.
5. Estimate generalization error using the test set.
6. Retrain selected model on all labeled (UMETRICS) data and use to predict postdoc status for all non-labeled (non-UMETRICS) observations.

Table B.2: Machine Learning Model Accuracy by Cutoff using Validation Set Predictions

Cutoff		Accuracy						
Type	**Value**	**Total**	**TPR**	**TNR**	**PPV**	**NPV**	**F1**	**AUC**
Model I: Random Forests (RF)								
F1	0.4680	97.31%	90.77%	98.43%	90.77%	98.43%	90.77%	0.9936
Youden	0.3230	96.53%	95.83%	96.65%	82.99%	99.27%	88.95%	
Model II: Boosted Trees (GBM)								
F1	0.3969	96.90%	90.48%	97.99%	88.50%	98.37%	89.48%	0.9862
Youden	0.1198	96.66%	93.15%	97.26%	85.29%	98.81%	89.05%	
Model III: Logit Model								
F1	0.4314	91.45%	71.73%	94.82%	70.26%	95.16%	70.99%	0.9464
Youden	0.1585	86.38%	93.90%	85.09%	51.81%	98.79%	66.77%	
Model IV: Linear Probability Model (LPM)								
F1	0.3464	89.67%	71.58%	92.76%	62.79%	95.03%	66.90%	0.9330
Youden	0.2644	67.35%	98.36%	62.06%	30.67%	99.55%	46.76%	

Notes: Table B.2 shows the prediction performance of four different prediction models for alternative probability thresholds used for prediction of validation set observations. The "F1" cutoff which corresponds to the threshold that maximizes the model's F1-score. The "Youden" cutoff maximizes the Youden Index: TPR + TNR - 1.

Table B.3: Random Forest Accuracy Using Test Set Predictions

Accuracy						
Total	**TPR**	**TNR**	**PPV**	**NPV**	**F1**	**AUC**
96.90%	89.19%	98.10%	87.92%	98.32%	88.55%	0.9920

Notes: Table B.3 shows the prediction performance of our tuned random forest model using test set predictions. These results can be used to estimate the generalization error of the prediction model since the test set observations were not used for model selection. However, prediction models often improve performance with greater sample sizes, and since the model here is trained on 50% of the available data (the training set), we may expect better performance when building the model using the full sample.

Table B.4: Random Forest Accuracy Using OOB Predictions from Random Forest Trained on Full Data

Accuracy						
Total	**TPR**	**TNR**	**PPV**	**NPV**	**F1**	**AUC**
98.25%	94.59%	98.86%	93.16%	99.11%	93.87%	0.9969

Notes: Table B.4 shows the prediction performance of a tuned random forest model fit on the full UMETRICS subsample using OOB predictions. The random forest model assessed here is fit to the full UMETRICS subsample, rather than 50% of the subsample as in Table B.3, and so part of the increased performance is likely due to the increase in sample size. However, these results will give an optimistically-biased measure of the generalization error of the prediction model since observations used in model selection (i.e. 50% of the observations which formed the training sample) are also used in generating the measures of prediction performance in this table. Therefore, we view the results in this table as an upper-bound on the prediction performance of this prediction model.