Identifying features predictive of faculty integrating computation into physics courses

Nicholas T. Young,[1] Grant Allen,[1] John M. Aiken,[1,2]
Rachel Henderson,[1] and Marcos D. Caballero[1,2,3,*]

[1]*Department of Physics and Astronomy, Michigan State University, East Lansing, Michigan 48824, USA*
[2]*Center for Computing in Science Education & Department of Physics, University of Oslo, N-0316 Oslo, Norway*
[3]*CREATE for STEM Institute, Michigan State University, East Lansing, Michigan 48824, USA*

(Received 18 October 2018; published 20 February 2019)

Computation is a central aspect of 21st century physics practice; it is used to model complicated systems, to simulate impossible experiments, and to analyze mountains of data. Physics departments and their faculty are increasingly recognizing the importance of teaching computation to their students. We recently completed a national survey of faculty in physics departments to understand the state of computational instruction and the factors that underlie that instruction. The data collected from the faculty responding to the survey included a variety of scales, binary questions, and numerical responses. We then used random forest, a supervised learning technique, to explore the factors that are most predictive of whether a faculty member decides to include computation in their physics courses. We find that experience using computation with students in their research, or lack thereof and various personal beliefs to be most predictive of a faculty member having experience teaching computation. Interestingly, we find demographic and departmental factors to be less useful factors in our model. The results of this study inform future efforts to promote greater integration of computation into the physics curriculum as well as comment on the current state of computational instruction across the United States.

DOI: 10.1103/PhysRevPhysEducRes.15.010114

I. INTRODUCTION

Computation is a central practice of modern scientific research that has enabled numerous experimental and theoretical discoveries in physics. While this practice is part and parcel to the work of modern physicists, it is not often represented in physics curriculum [1–3]. This is despite the current push from various professional and governmental organizations for the integration of computation into a variety of fields and contexts, including physics [4–6]. Integrating computation into physics courses represents a shift in the curriculum and thus requires faculty to develop, adopt, or adapt materials appropriate for their courses and students. To support faculty and further integration efforts, we need to understand why faculty choose to integrate computation into their courses or why they choose not to do so. In this paper, we address this issue by determining which factors are predictive of a physics faculty member having experience teaching computation to undergraduate students.

*Corresponding author.
caballero@pa.msu.edu

As computation is up and coming as an instructional tool and strategy in physics, there is little literature on the experiences that faculty have when integrating computation into their courses. While we expect there to be challenges unique to integrating computation into the physics classroom, we also expect faculty to encounter similar difficulties as they would for implementing other instructional efforts including research-based instructional strategies (RBIS) [7,8]. For example, faculty may be concerned about having to make time to teach students basic programming principles in addition to the physics content they are already required to cover or having to create instructional materials that utilize computation. Work on faculty change has found that these concerns may be alleviated by supporting and encouraging faculty as they implement RBIS [8,9]. The Partnership for Integration of Computation into Undergraduate Physics (PICUP) is one such group currently working to support faculty as they integrate computation into their courses [10]. However, to alleviate such concerns, we first must understand the nature of those concerns and how they might impact whether a faculty member uses computation in their classroom and why some faculty have decided to include computation in their classrooms. In this paper, we use a machine learning technique called random forest to address the latter, that is to determine the factors that are predictive of whether a faculty member has experience teaching computation or not.

This paper is organized as follows: Sec. II provides an overview of the findings from institutional change literature in physics. We then provide an overview of the random forest methodology and its implementation in Sec. III with details appearing in the Supplemental Material [11]. In Sec. IV, we describe our findings followed by the work done to validate those findings (Sec. IV) and the resulting limitations of those findings (Sec. VI). We conclude with a discussion of our findings (Sec. VII) and their broader implications (Sec. VIII).

II. BACKGROUND

To understand those concerns or factors that might affect a faculty member's decision to teach computation to their students, we can look at which factors are predictive of a faculty member trying RBIS, continuing to use RBIS, and using multiple RBIS. Because we are interested in whether faculty have experience teaching computation, we are interested in whether faculty have reached the implementation stage of Rogers' five stages for adopting an innovation, which include knowing of the innovation, becoming persuaded to adopt the innovation, deciding to adopt the innovation, implementing the innovation, and continuing to use the innovation [12]. We leave questions about why faculty continue or do not continue to use computation to future work. Prior work into adopting new instructional strategies suggests that faculty choose to implement RBIS based on their own decisions [13–17]. Henderson, Dancy, and Niewiadomska-Bugaj extended this line of work by looking at specific factors predictive of adopting a new instructional strategy and found that regularly reading teaching journals, attending talks and workshops related to teaching, attending the New Faculty Workshop, having an interest in using more RBIS, and the type of institution (two-year college, four-year bachelor granting institution, and four-year doctoral granting institution) are predictive of a faculty member trying a RBIS [9]. Alternatively, they found that factors such as class size, research productivity, job type (lecturer, full professor, etc.), departmental encouragement, years of teaching experience, course type (algebra-based or calculus-based), and demographic factors such as gender and highest degree obtained were not predictive of a faculty member trying a RBIS. In addition, perceived implementation challenges such as situational characteristics, including resistance from students, large class sizes, pressure to cover a large amount of content in the course, and student expectations about how class should be structured, and personal reasons, such as the perceived amount of time it would take to implement the change and having a bad experience with trying to implement the change, could prevent a faculty member from trying to implement a RBIS [7,8].

Even though computation is a technique to do physics rather than an instructional strategy like RBIS, its use is informed by physics education research and it is not typically found in traditional lecture-based courses. Just as adopting a RBIS requires adopting new ways, tools, and methods of teaching, integrating computation into a course or curriculum also requires adopting these. Further, many of the implementation challenges for RBIS, such as not enough time to fit in new content, the amount of time needed to implement the change, and student resistance, have been also documented for computation when a department tries to transform their courses to include computation [18–20]. Therefore, we believe that we can treat computation like a research-based instructional strategy at least with respect to adoption and implementation. We can apply an institutional change lens to interpret our results and, thus, we expect the features we find to be predictive of whether faculty have experience teaching computation to also be important predictors of whether faculty have implemented RBIS as found in the literature.

III. METHODOLOGY

The factors found in the literature and the data we use in this study consist of binary, Likert-scale, and open response questions. While these data are not uncommon in physics education research (PER), they are often themselves the only source of the data (i.e., only one form of response) or they are part of some larger data set where some number of the same type of response formats (e.g., multiple choice responses with a single correct answer) are the main source of data. In both cases, PER has accepted methods for analyzing this kind of data to determine key factors that predict the outcomes most strongly. For example, when these data are part of a larger data set, a regression analysis can be performed where the categorical data are treated using binary codes (i.e., "dummy" variables) and are then included in the regression analysis. This is a common technique used in a number of studies in physics education research (PER) [9,21,22]. However, performing such a regression analysis on our data is problematic. As most of our data take some categorical form, our data violate key assumptions in any linear regression model such as normality and equal spreads. [23,24]. As such, the questions posed in our work are rooted in a classification task: what features predict which faculty have experience teaching computation and those that do not? Characterizing our study as a classification task led us to employ a supervised learning method appropriate for the data—the random forest algorithm. Below, we provide a brief overview of the algorithm and our implementation specific to this study (Sec. III B). In the Supplemental Material [11], we provide more thorough background on the random forest algorithm including how the model is validated (Supplemental Material [11], Sec. II), how a subset of important features can be identified (Supplemental Material [11], Sec. III), and how to handle bias in the model (Supplemental Material [11], Sec. IV).

A. The random forest algorithm

A random forest is a supervised machine learning approach that can be used to classify data into categories or model outcomes over some range using regression. The algorithm can also be used to develop a quantitative, relative measure of how important certain factors are in predicting those categories or outcomes [25–27]. As with all supervised machine learning techniques, a random forest is trained on a data set with known classifications. The model is grown using binary decision trees and the results of each decision tree is aggregated into a single result [27,28]. The randomness comes from the fact that only a fraction of the factors are used to construct the decision trees and only a fraction of the data, controlled by the training fraction is used to test the model (see Supplemental Material [11]). Through this training, the algorithm develops a model for the data set. Then, the model is applied to a set of sequestered data, known as the test set, that was not used in the original training. The model is used to predict the classifications for this testing data, which are also known.

In order to assess the model, a few measures are employed. First, the accuracy of the model is computed, which is the fraction of the data in the test set that was correctly classified. Second, a receiver operating characteristic curve (ROC curve) of the model is generated [29]. An example is shown in Fig. 2 of the Supplemental Material [11]. The ROC curve plots the true positive rate (proportion of people who have a specific trait such as teaching computation that are correctly classified as having that trait) as a function of the false positive rate (the proportion of people who do not have that specific trait but are incorrectly classified as having the trait). The ROC curve allows one to visualize the trade-off between creating a model that has a high true positive rate but has many false positives and models have few false positives but also fewer true positives. The ROC curve can be represented as a single number called the area under the curve (AUC), where a perfect classifier will have an AUC of 1 while a binary classifier that is randomly classifying data will have an AUC of 0.5. While there is not general agreement on what constitutes different levels of the significance for the AUC measure, the literature suggests that an AUC > 0.7 is a reasonable lower bound for a random forest model [30].

In addition to classifying data, the random forest algorithm is able to empirically determine the relative importance of each factor to the model. While there are many ways to calculate this importance, we chose to use an importance measure based on the AUC because it has been shown to be unbiased when the predicted variable is unbalanced and the data types of the factors are different [31] as is the case with our data. Using this importance measure, the relative importance of a factor to the model is determined how much the AUC changes when the information from that factor is removed from the model. If the factor removed from the model was useful for prediction, the AUC will decrease to a greater extent than if the factor was less useful for making predictions. By removing each factor one at a time, the change in AUC can be determined for each factor and the relative importance of each factor can be determined.

However, computing the relative importance for each factor does not provide any information about whether the factor is actually important to the model. To determine this, some type of selection technique must be used. While many techniques exist, we used recursive backward elimination, which means the less important features were recursively removed from the model until the "best" model is found as measured by the accuracy [32]. Here, best model refers to the model that uses the fewest number of factors to produce a model that is within 1 standard error of the highest possible accuracy.

Random forests are one of a number of different possible machine learning classifiers that can be used on any given data set. Naïve-Bayes methods, support vector classifiers, *k*-nearest neighbors, and gradient tree boosting are all possible classification schemes that could have been used for this study. Olson and collaborators modeled a number of open-source data sets with these and other classifiers in order to offer best practices for using machine learning classification algorithms [33]. In their work, Olson *et al.* found the random forest algorithm to be one of the best algorithms overall—second only to gradient tree boosting. In head-to-head comparisons where parameter tuning was allowed, random forest predictions were as accurate, within error, of gradient tree boosting and support vector machines. In principle, several algorithms could be applied to the same data set and the resulting classifications compared, but that is not the purpose of our work. We selected the random forest algorithm for its documented robustness and its intuitive nature.

B. Implementation

1. Survey

To determine which factors are most important in predicting whether a faculty member has experience teaching computation, we analyzed survey responses from 1246 faculty at 357 unique institutions [3]. The survey focused around five broad topics: attitudes toward computation, experience with computation, computational resources provided by their department, motivations for teaching or not teaching computation, and departmental views of computation. As prior work on adopting research-based instructional strategies has found that learning about new strategies as well as interest in using new RBIS to be significant explanatory variables in determining whether a faculty member will try a new RBIS [9], we expect that faculty's attitudes toward computation will be predictive of them choosing to incorporate it into their classroom. For example, we would expect a faculty member who sees clear

benefits of using computation in the classroom such as allowing for new problems and concepts to be covered in the course or being able to visualize or simulate phenomena to incorporate computation into the classroom. Likewise, we would expect that a faculty member who has experience with computation, either having learned it during their schooling or using computation in research or other nonteaching duties, would be more likely to integrate computation into their classroom than an instructor who has never used computation before and hence would have to teach themselves before including computation in their classrooms. While Henderson, Dancy, and Niewiadomska-Bugaj did not find departmental encouragement or research productivity measures to be useful explanatory variables for determining whether a faculty member tried a RBIS [9], faculty might be motivated to incorporate computation if their department encourages them with incentives for integrating computation into their courses (such as increased resources for the course or as criteria for tenure or promotion) or if they believe using computation in their courses would open new funding opportunities or other research benefits. Finally, demographic or institutional factors may influence a faculty member's willingness to incorporate computation into their course and, therefore, questions regarding these factors were also included on the survey.

The constructed survey items varied in scale of measurement from yes or no binary questions, to Likert scales, and open-ended responses. Given the broad range of questions and the fact that not all questions would be relevant to all survey takers, the survey used binary logic; some survey participants saw different questions based on their response to the first question, "do you have experience teaching computation." Of the 1246 respondents, 751 faculty said they did have experience teaching computation while 495 faculty said they did not have experience teaching computation.

2. Sample

In order to determine important factors for integrating computation into physics courses, we could only use questions that were seen by both faculty who have and do not have experience teaching computation. This left us with 44 questions which were binary, Likert-scale, and open response. Because the 44 questions we selected were of different data types (from binary to near continuous) and our data set was unbalanced, we utilized conditional inference forests via the `cforest` function in the Party package for R [34–37]. As Kim and Loh have found that different proportions of missing values can introduce bias into classification trees, we excluded any faculty member from the sample who did not answer all 44 questions [38]. We address our choice to remove faculty who did not answer all questions from the sample instead of using multiple imputations or other methods to address missing data in Sec. VI. After doing this procedure, we were left with 693 faculty (56% of our sample). In the original sample, 60% of the faculty had indicated that they had experience teaching computation while in our reduced set with only faculty who answered all questions, 62% indicated they had experience teaching computation, suggesting that the data we are using is still representative of the overall sample.

3. Growing the random forest

To run the `cforest` algorithm, we first randomly split the data into a training set and a testing set, where 70% of the data were used in the training set (corresponding to a training fraction of .70), a common value in the literature [39–41]. Next, we set mtry $= \sqrt{N}$ using `cforest_control`, where ntree is the default value in the `cforest` algorithm and mtry is equivalent to n_{in} in Ref. [42]. All other `cforest` function parameters were set to their default values from `cforest_unbiased`, which includes subsampling without replacement and $n_{\text{tree}} = 500$. We then ran the `cforest` algorithm on our data set to grow the forest. To calculate the accuracy and AUC of the model, we used the `caret` and `ROCR` packages [43,44]. To calculate the variable importances we used the `varimpAUC` function from the party package. We then ran the `cforest` algorithm an additional 29 times, for a total of 30 trials, allowing us to use the central limit theorem [45] to define the mean and standard error of the importances. Thirty trials is typically the minimum number of trials to apply the central limit theorem [46] and Shapiro-Wilk tests, a test of normality where the null hypothesis is that the data are normally distributed, show that with 30 trials, the data are normally distributed; therefore, additional trials were deemed not necessary and would have only consumed additional computational resources. Further, QQ plots [47], which are scatter plots that compare the theoretical normal distribution with the actual data and will appear as lines if the data are in fact linear, do not show any nonlinear behavior, suggesting that 30 trials was sufficient. In addition, the data were resplit into training and testing data sets before each trial to minimize the inherent randomness of the `cforest` algorithm and any bias that could result from the training data not being representative of the overall data.

In order to find the meaningful factors, we used the recursive backward elimination technique described in Ref. [32] as we did not want to presuppose a set number of meaningful variables, only had a few negative values such that the resulting distribution would not be useful, and generating null importances for our 44 variables would not be practical (see Supplemental Material [11] for details of these alternative approaches). The recursive backward elimination technique was implemented through a modification of `varSelRF` function in the `varSelRF` package such that the forests grown during the process would be

conditional inference forests rather than random forests and the initial importances would come from the results of the 30 trials rather than being generated within the algorithm (and thus would allow the results to be replicated) [32,48]. We used the default value of 20% of the variables being dropped after each trial. We use the term "meaningful" instead of "significant" to signify that selected factors are the factors found to provide the most information to the model and not found from a test of statistical significance.

IV. RESULTS

A. Model validation

Across the 30 trials, our model successfully predicted whether a faculty member had experience teaching computation 77.4% $\pm$ 0.5% of the time and had an AUC of 0.838 ± 0.002 (see representative ROC curve in Fig. 1). As 62.2% of the sample had experience with teaching computation, our accuracy is significantly (both in the practical and statistical sense) higher than the noninformation rate, which is the accuracy if the model were to predict every data point as belonging to the majority class, in our case, the faculty with experience teaching computation. From the confusion matrix shown in Table I, we see that the model is better at predicting faculty with experience teaching computation compared to faculty without experience teaching computation. This difference in prediction ability may be caused by the fact that there are approximately 50% more faculty with experience teaching computation than faculty without experience teaching computation; we address this further in Sec. VI. Since our accuracy is significantly higher than the noninformation rate and the AUC is above 0.8, our model can satisfactorily predict whether a faculty member has experience teaching computation.

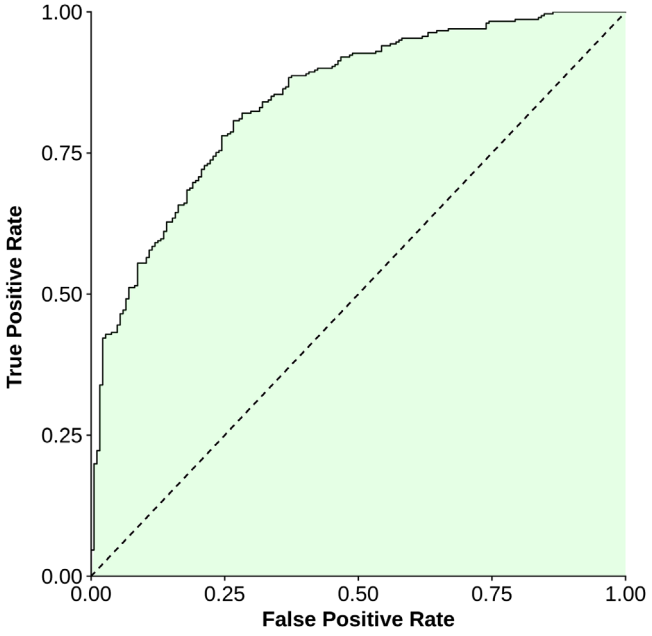

FIG. 1. Representative ROC curve for one of the 30 trials of predicting whether faculty do or do not have experience teaching computation from 44 predictor variables. The AUC for this trial is 0.8250, suggesting a good model, as the AUC is greater than 0.7.

TABLE I. Confusion matrix for a representative trial of the 30 trials. Numbers in bold are correct predictions and add to the accuracy.

		Data says	
Do you have experience teaching computation?		Yes	No
Model	Yes	**56.7%**	16.8%
Predicts	No	5.8%	**20.7%**

B. Feature importance

1. Features that are more important

In addition to generating predictions, our model is able to determine the importance of each variable that was used in predicting whether a faculty has experience teaching computation; these importances are shown in Fig. 2. The variable importances here are computed using AUC-based permutations methods, meaning the importance shown in the plot is the average decrease in the AUC if the variable were permuted and its association with the response variable were broken. For example, if the responses in the variable "Use computation in research with students" were randomly shuffled, the AUC in Fig. 1 would drop from 0.825 to approximately 0.760, a 0.065 decrease, which is that variables' importance as shown in Fig. 2.

We find that the most important features are "I use computation in research with students," "I do not personally use computation," "computation allows me to bring new physics into my classroom," "computation allows me to bring new problems into my classroom," and the highest physics degree offered by the institution. Actionable plans to increase computational instruction, "I use computation in my research," institution type, and tenure status are slightly less important features for predicting whether faculty have experience teaching computation. When we perform the recursive backward elimination technique, we find that the meaningful features are "I use computation in research with students," "I do not personally use computation," "computation allows me to bring new physics into my classroom," "computation allows me to bring new problems into my classroom," and the highest physics degree offered by the institution.

As a check that these five meaningful variables are in fact meaningful, we then reran the `cforest` algorithm 30 times with just these five variables. We obtained an average accuracy of 76.4% $\pm$ 0.4% and an AUC of 0.818 ± 0.002. Recall that the values obtained when using all 44 variables were 77.4% $\pm$ 0.5% and 0.838 ± 0.002, respectively. As the accuracies are nearly the same and the AUC of the five meaningful variables model is still above 0.8, we can

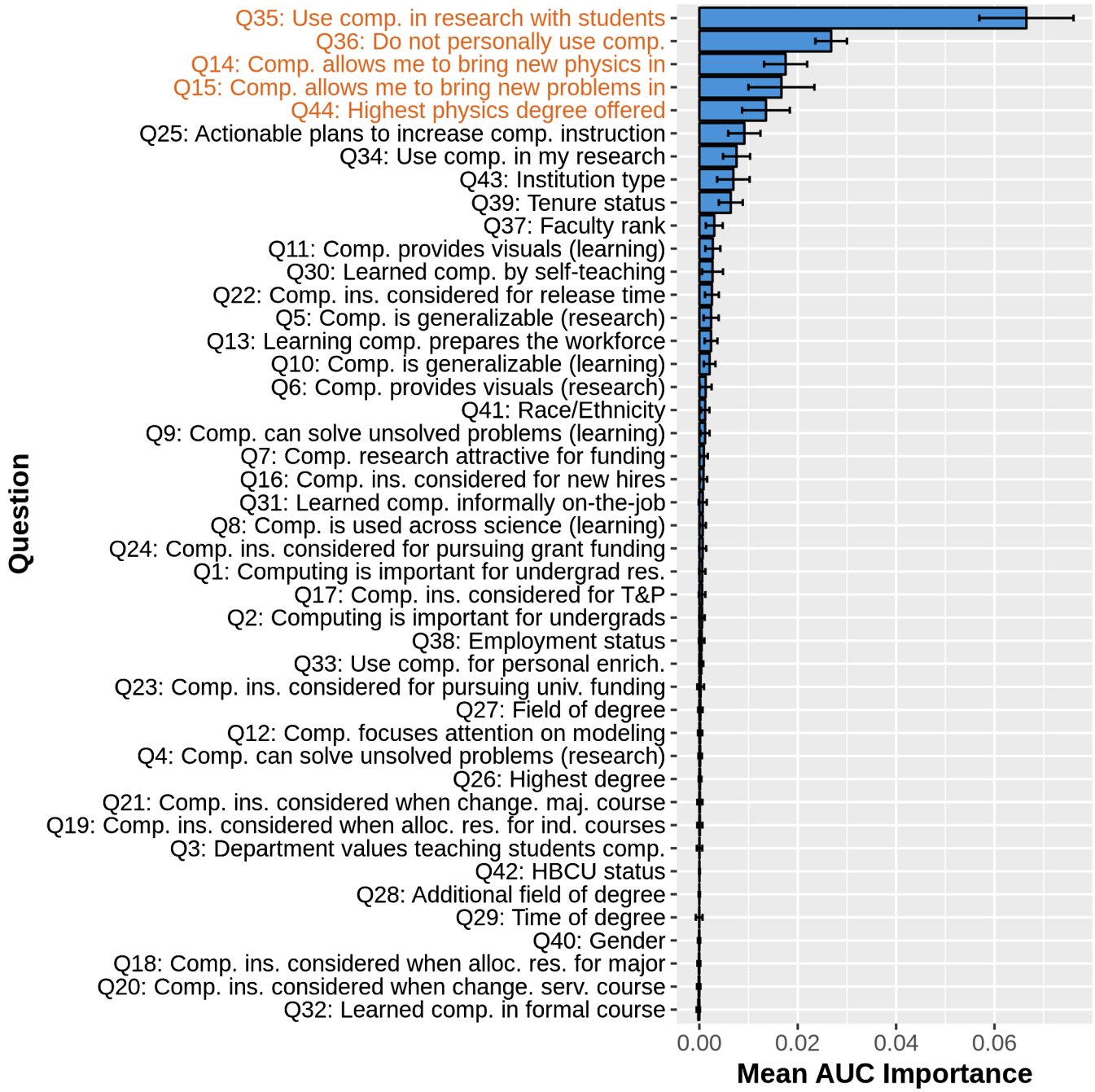

FIG. 2. Variable importances for each of the 44 factors used to predict whether a faculty member has experience teaching computation. The importance is based on the average change in the AUC if the factor is permuted. Thus, factors that change the AUC the most have the largest importances. The first five factors in color are the ones selected from the recursive backward elimination approach. The error bars represent a standard error of the mean AUC importance. Full questions can be found in Table 2 of the Appendix.

further support the claim that these five variables are meaningful.

While the importances are useful for determining which variables are good discriminators between faculty who have experience teaching computation and faculty who do not, the importances by themselves cannot say which group is more likely to have a specific trait. To determine which group is more likely to possess a specific trait, the distributions of responses must be investigated. The distributions of the five meaningful variables are shown in Fig. 3. For example, faculty who have experience teaching computation tend to use computation to provide undergraduate students with research experience while faculty who do not have experience teaching computation tend not to personally use computation. Similarly, faculty who have experience teaching computation tend to agree that

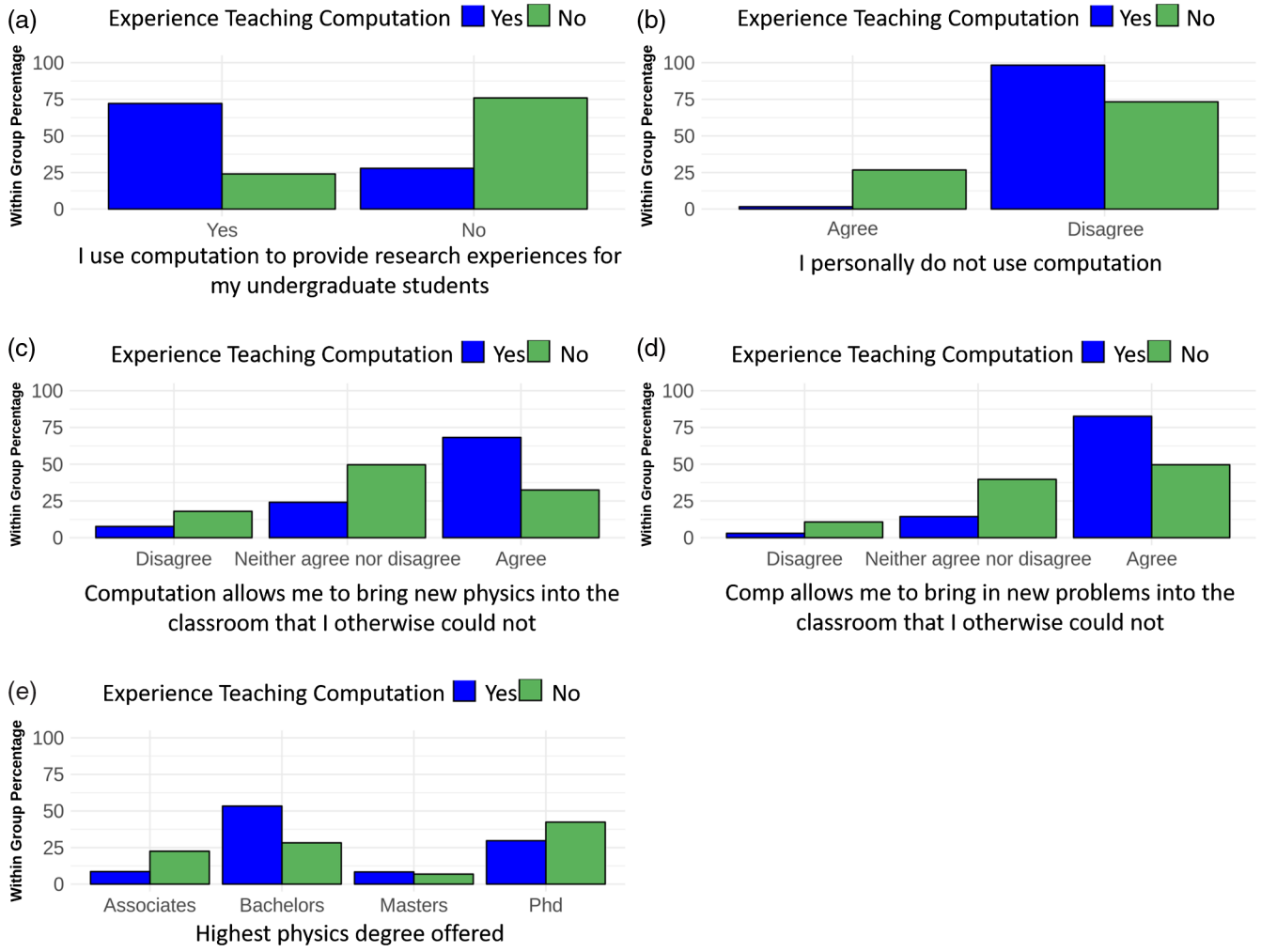

FIG. 3. Distribution of responses based on whether faculty do or do not have experience teaching computation. Here, within group percentage means the percentage of faculty within the group who use computation or the group who does not use computation. Since these five factors are the meaningful factors, we expect that the distribution of responses should be different between faculty with experience teaching computation and those who do not. Plot A shows the distribution of the most important feature while plot E shows the fifth most important feature. All five plots have χ^2 with corrected $p < 0.05$.

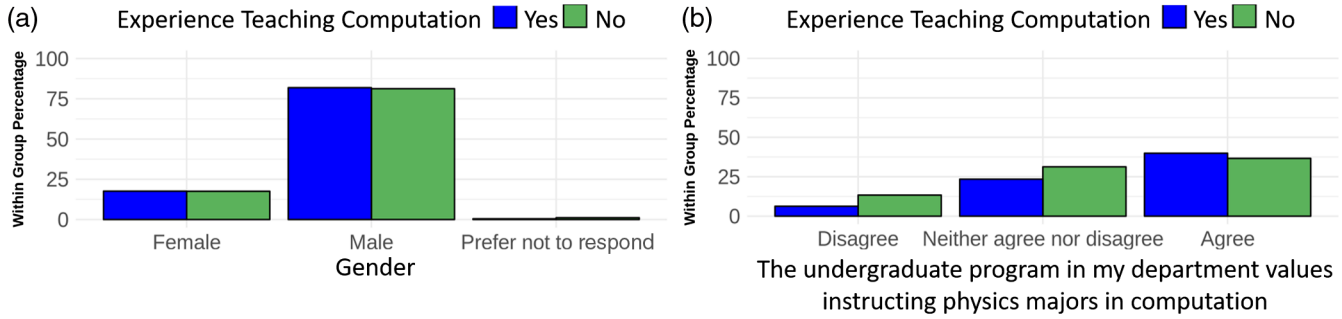

FIG. 4. Distribution of responses based on whether faculty do or do not have experience teaching computation. Here, within group percentage means the percentage of faculty within the group who use computation or the group who does not use computation. Since these two variables are less useful for predictions, we expect that the distribution of responses should not be different between faculty with experience teaching computation and those who do not.

computation allows them to bring new physics and new problems into the classroom that would not be possible without using computation to a greater degree than those who do not have experience teaching computation.

2. Features that are less important

In addition to looking at which features are good discriminators between faculty who do and do not have experience teaching computation, investigating which features are not as good discriminators can be informative. For example, we find that demographic factors such as race or ethnicity, gender, time of degree (a proxy for age), field of degree, and highest degree obtained are among the less important factors. In addition, we find that departmental and institutional factors (statements in Fig. 2 that begin with "computation instruction considered…") are also among the less important factors. The distributions of some of these factors are shown in Fig. 4. Compared to the distributions of features that are more important, the features that are less important seem to appear equally among the faculty who do and do not have experience teaching computation. These differences in distributions provide further support that the five meaningful factors are indeed meaningful.

V. VALIDATING OUR CHOICE OF HYPERPARAMETERS

Random forest and conditional inference forest models have multiple parameters that can be adjusted to control how the forest grows. As these parameters need to be picked before the model is created, they are called hyperparameters and choices for these hyperparameters can affect the quality of the forest grown. For example, if the amount of the data from which the forest is grown (training fraction) is increased, the predictions should improve up to some threshold. Likewise, if the number of trees in the forest is increased, the quality of the predictions should increase up to some threshold. In this section, we assess the stability of our model by varying the training fraction and the number of trees in each forest. If our findings do not vary significantly as the training fraction and number of trees vary, we can be more confident that our results actually are representative of the data and are not artifacts of the model. As we are more concerned with identifying the important factors than the predictive power of the model, we do not perform a grid search to identify the set of hyperparameters that would result in the highest accuracy or area under the curve. Prior work has found that randomly choosing hyperparameters is more efficient and provides comparable results to a typical grid search [49].

A. Effects on accuracy and area under the curve

To check for variation, we selected five training fractions, 0.5 (split the data in half), 0.6 (used when creating a training, validation, and testing set), 0.7 (our original choice), 0.8 (amount used for a 5-fold validation), and 0.9 (amount used for a 10-fold validation) and six forest sizes (50, 100, 250, 500, 750, 1000), where 500 trees was our original choice, informed by practical considerations and Svetnik *et al.*'s finding that error rates stabilize on the order of 10^2 [42]. For each pair of training fraction and forest size, we ran our `cforest` algorithm 30 times, creating 870 new forests (29 new models with 30 trials each). We then averaged our results across forests with the same training fraction and number of trees. The accuracy and AUC for each pair of training fraction and number of trees are shown in Figs. 5 and 6, respectively. A visual inspection suggests that neither the accuracy nor area under the curve vary significantly when the training fraction and the number of trees in the forest are changed. Indeed, the ranges of the accuracy and AUC are 0.03 and 0.02, respectively, which are insignificant from a practical perspective. Thus, while this range represents multiple standard deviations, it is of little practical significance so we can be confident that our model would not significantly improve or become worse by selecting a different set of hyperparameters.

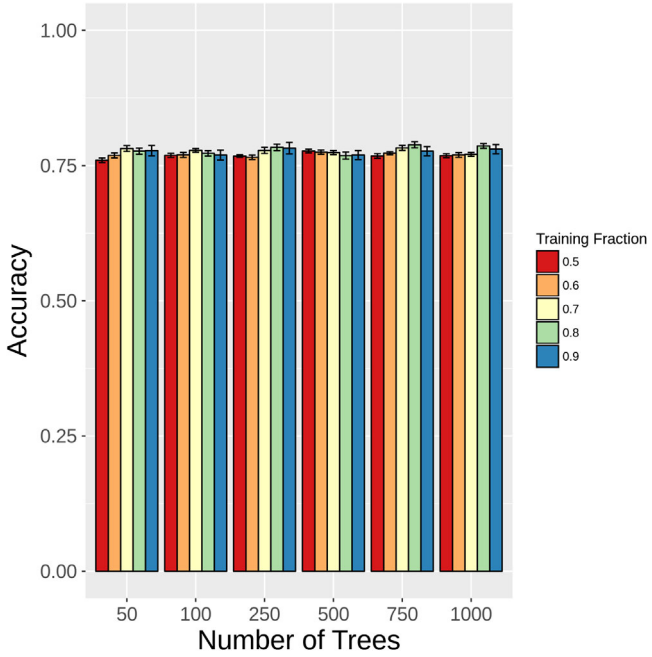

FIG. 5. Average accuracy of the model for various training fractions and number of trees in the forest. Error bars correspond to a standard error.

B. Effects on variable importance

Because there were small variations in the area under the curve for varying training fractions and number of trees, we may expect there to be variation in the variable importances as the variable importances are based on changes in the area under the curve. We expect there to be natural variation

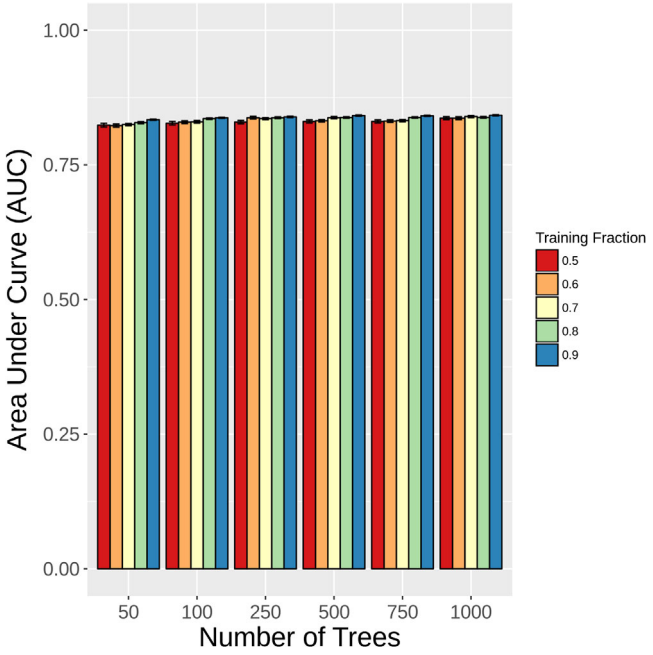

FIG. 6. Average area under the curve of the model for various training fractions and number of trees in the forest. Error bars correspond to a standard error.

in the importances just from using different training sets, so we chose to only focus on the selected meaningful variables. We used the same choices of hyperparameter combinations as in the previous section. Figure 7 shows the fraction of the 29 new models where each variable was found to be meaningful. Variables that are not shown were not found to meaningful in any of the models. We see that "using computation in research with students" and "I do not personally use computation" were selected as meaningful in all the models while "computation allows me to bring new physics into my course" and "computation allows me to bring new problems into my course" were selected as meaningful in over two-thirds of the models. Highest physics degree offered was selected as meaningful in only about a third of the other models, suggesting that the selection of this factor may be influenced by how the model is constructed or should be more accurately described as marginally meaningful. None of the other factors appeared in more than 15% of the models. These results suggest that at least four of the selected meaningful factors are in fact meaningful and not just artifacts of how we constructed our model while highest physics degree offered may be marginally meaningful and influenced by the hyperparameters chosen.

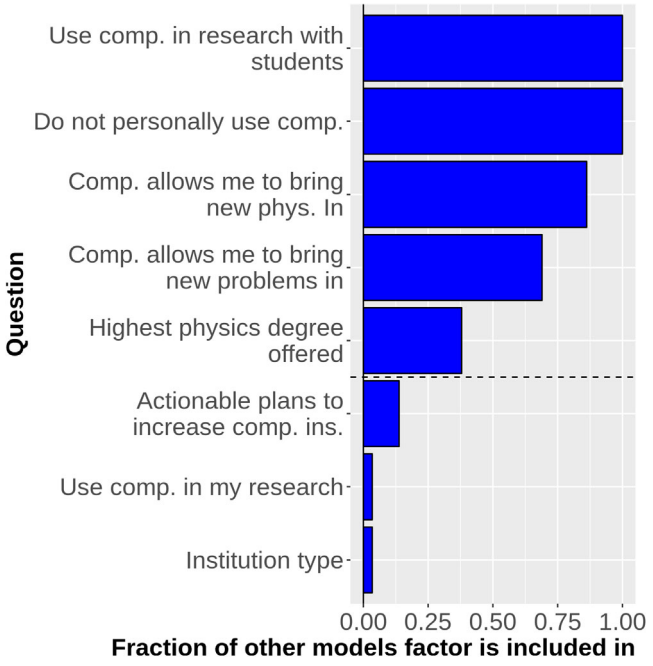

FIG. 7. Fraction of the other 29 models in which the variables are selected as meaningful. Variables above the line were selected as meaningful in the original model.

VI. LIMITATIONS

In this section, we comment on how our model may be limited based on the nature of the data and the amount of missing data.

A. Unbalanced classes

Because 60% of our data are from faculty who have experience teaching computation, the training data set used to grow the random forest model will contain more instances of faculty who have experience teaching computation than faculty who do not have experience teaching computation. As there are more instances of faculty with experience teaching computation to learn from, we expect that the model will be better at correctly classifying faculty with experience teaching computation than faculty without experience teaching computation, which we observed in our data. However, our model appears to be classifying faculty without experience teaching computation almost at random, suggesting that our results are biased. While many methods have been proposed to correct the imbalance, these methods can introduce additional biases of their own. For example, the data can be artificially balanced by bootstrapping the minority class (up-sampling), until the classes are equal. However, the `cforest` algorithm is not unbiased when bootstrapping is used [36]. Alternatively, the data could be down-sampled, where only a random sample of the majority class equal in size to the minority class is used to grow the forest. While this does not introduce bias in the `cforest` algorithm, it would require excluding nearly 20% of the usable data from the model and in our trials, did not improve the accuracy or AUC of the models. Finally, there have

been alternative variations of the random forest algorithm, including balanced random forests, which is based on up-sampling, and weighted random forests, which are based on cost sensitive learning, to handle unbalanced classes. However, these algorithms are based on CART algorithms meaning the original biases conditional inference forests were designed to combat would be reintroduced. Thus, adequately handling imbalanced data without introducing or reintroducing bias appears to be an open problem.

Alternatively, if we are using our model as a truly predictive model, we would not know the true values until after the prediction. In the case that the model does predict that a faculty member does not have experience teaching computation, it would be correct about 80% of the time $[20.7\%/(20.7\% + 5.8\%)]$. Thus, whether the algorithm should be viewed as biased depends on what we view as the given information—the model's prediction or the actual response.

While our predictions may contain some bias, the predictive model was not the focus of this study. Instead, we are concerned with determining which factors are discriminators between faculty with or without experience teaching computation. The factors were determined from values of the AUC importance, which is not affected by class imbalance because the area under the curve weighs the majority and minority class equally [31]. The AUC importance is not involved in the growing of the forest, however, meaning that the predictions can still be biased from unbalanced classes. Thus, we believe that our results are minimally impacted by having more faculty with experience teaching computation in our sample.

B. Missing data

For this paper, we decided to employ complete case analysis, which means we only included faculty who responded to all 44 questions used in the study. This left us with just over half of the original participants, meaning we excluded nearly half of the participants, which could affect our results. As described by Hapfelmeier *et al.*, there do exist methods for random forests with missing data and generating variable importance measures [50]. When we implemented this approach, our accuracy decreased to $72.2\% \pm 0.4\%$ but the AUC increased to 0.889 ± 0.001. In terms of the selected meaningful variables, we found that the five variables selected in our initial model and one additional variable ("I used computation in my research") to be meaningful. However, this set of meaningful variables could also be observed by varying the training fraction and the number of trees in our original model. Therefore, it does not appear that excluding faculty with missing data changes our results and hence we decided to use the complete case analysis because complete case analysis resulted in a better predictive model.

VII. DISCUSSION

To interpret our results, we can compare the meaningful factors to those found in the literature. Of the twenty factors Henderson, Dancy, and Niewiadomska investigated using logistic regression [9], four appear in our study directly: highest degree obtained, gender, type of institution, and type of position, which we called employment status. Additionally, we can relate three of their variables to three of our computation specific variables: we treat their "department encouragement" variable as our "department values teaching students computation" factor, their "interest in using more RBIS" variable as our "actionable plans to increase computation" factor and their "years of teaching experience" variable as our "time of degree" factor as both of these can be viewed as proxies for age. As our survey was designed to cover five broad areas and to limit survey fatigue, not all of the factors from Ref. [9] could be included in the survey. Of these seven factors, we then expect that the factors Henderson *et al.* found to be correlated with trying or not trying RBIS (institution type and interest in using more RBIS) to be among the factors we found to be more predictive of a faculty member having experience teaching computation while their other five variables should be among the factors we found less important. Indeed, we found that type of institution and actionable plans to increase computation were among the more important of our factors while the other five were among the less important factors. We note however that type of institution and actionable plans to increase computation were only meaningful factors in less than 15% of the models we created when varying the number of trees and the training fraction and were not meaningful in our original model.

As teaching and research expectations vary based on the institution type, we may expect some types of institutions to allow their faculty more time to focus on their courses. For example, faculty at institutions that only offered bachelors degrees in physics were more likely than faculty at any other type of institution to have experience teaching computation. This may be due to lower research demands and, hence, more time to devote to developing and preparing their courses. Therefore, faculty at these types of institution may have already overcome one of the implementation challenges, having time to do so. Likewise, those who have already made plans to integrate computation into their courses have overcome the challenge of fitting more material into their courses.

On the other hand, factors such as a faculty members' gender, highest degree obtained, and years of teaching experience do not address any of the implementation challenges so we do not expect these to be important factors. Departmental encouragement and type of position may indirectly relate to an implemental challenge such as

approving changing curriculum to accommodate computation or by creating time to work on implementing computation. However, type of position only refers to full-time, part-time, or course-by-course, not the actual duties of the position, so it is unlikely that this factor provides much information about time for implementing computation beyond the number of hours worked each week. As these two factors are at most indirectly related to challenges with implementing computation or a RBIS, it seems reasonable that they are not important factors for discriminating between faculty who do and do not have experience during computation.

One reason that we may not be finding the same important factors as found in the literature for adopting RBIS is that the faculty who are using computation are likely what Rogers calls the early adopters [12]. The literature on adopting RBIS focuses around more established instructional strategies and hence the early and late majorities. We would expect that the early adopters of a new instructional strategy would be those familiar with the strategy and see a clear benefit to using the strategy instead of continuing to the use the strategies they had previously been using. This is the pattern we observe in our meaningful factors: those who use computation in their research with students tend to use computation while those who do not personally use computation tend not to have experience teaching computation. Likewise, those who believe computation allows new physics and new problems to be incorporated into the curriculum, a clear benefit of using computation, are more likely to have experience teaching computation.

As 60% of the respondents to our survey indicated they have experience teaching computation, our claim that these faculty are early adopters may seem contentious as 60% is a majority of faculty. However, Caballero and Merner note that those who use computation are more likely to respond to the survey than those who do not use computation [3]. Thus, 60% should be thought of as an upper limit on the percentage of faculty using computation in their courses.

Regardless of how we classify these faculty, it is important to note that these results are just a snapshot of the state of computation now. As computation in the classroom becomes adopted by more physics faculty, we expect that these meaningful factors will change and will likely more closely align with factors correlated with trying a RBIS. Currently though, the important factors were focused on what the individual does—using computation in research with students or not using computation personally, or believes—computation adds new physics and problems to the course, and not on institutional or departmental factors, suggesting that integrating computation into a course is a personal choice, which does align with previous findings that faculty adopt new instructional strategies based on their own decisions.

VIII. CONCLUSION AND IMPLICATIONS

In this paper, we created a random forest model to predict whether physics faculty have experience teaching computation. From our model, we find four meaningful factors and one marginally meaningful factor that discriminate between faculty who do and do not have experience teaching computation: using computation in their research with students, not personally using computation, believing computation allows them to bring new physics into their course, believing computation allows them to bring new problems into their course, and the highest physics degree offered at their institution. Since most of the meaningful factors are related to faculty choice and there is lack of institutional or department factors, we conclude that deciding to teach computation is viewed as a choice by physics faculty members.

As the meaningful factors were at the individual level instead of the departmental or institutional level, the implications of our study are then that at this moment, efforts to increase computation use should be at the level of individuals rather than at a departmental level. If we do characterize those who use computation as early adopters, then future work should focus on the faculty who will make up the early majority, which need to see evidence that computation adds value to their course before they will adopt it [12]. Broadly, the meaningful factors suggest that faculty who have experience using computation and see value in teaching computation will do so while those who do not use computation in their professional work or do not see how computation can complement their course's current content will not teach computation. These findings are perhaps not surprising as a study outlining a vision for integrating computation into undergraduate physics courses concludes that integrating computation will require "many faculty minds to change" and many "faculty skills to train" [2]. The fact that these factors are still relevant a decade later suggest there is still a long way to go to widespread implementation of computation in undergraduate physics courses.

ACKNOWLEDGMENTS

The authors would like to thank Laura Merner, Norman Chonacky, and Robert Hilborn for their work to develop survey areas. The authors would also like to thank members of PERL@MSU for their helpful comments on drafts of this paper. This work was supported by the National Science Foundation's Division of Undergraduate Education (DUE-1431776 and DUE-1432363) and by Michigan State University.

APPENDIX: QUESTIONS FROM COMPUTATIONAL SURVEY USED IN OUR MODEL

In Table II, we provide the wording of the questions we used in the survey and the shortened versions of those questions that are used throughout the paper.

TABLE II. Full list of survey questions.

Short name	Question statement
Teaching computation	Given this broad definition of computation, do you have any experience teaching computation to undergraduate physics students?
Q1: Computing is important for undergrad research	Rate the degree to which you agree or disagree with the following statements. For the following questions we would like to understand your personal perspective of the role of computation in physics.—I think that computation is important for undergraduate physics research.
Q2: Computing is important for undergrads	Rate the degree to which you agree or disagree with the following statements. For the following questions we would like to understand your personal perspective of the role of computation in physics.—I think that learning computation is important for undergraduate physics majors.
Q3: Department values teaching students computation	Rate the degree to which you agree or disagree with the following statements. For the following questions we would like to understand your personal perspective of the role of computation in physics.—The undergraduate program in my department values instructing undergraduate physics majors in computation.
Q4: Computation can solve unsolved problems (research)	With regard to your personal research, rate the degree to which you agree or disagree with the following statements:—Computation can solve unsolvable (analytical) problems.
Q5: Computation is generalizable (research)	With regard to your personal research, rate the degree to which you agree or disagree with the following statements:—Computation is generalizable to many different kinds of problems.
Q6: Computation provides visuals (research)	With regard to your personal research, rate the degree to which you agree or disagree with the following statements:—Computation affords visualization (graphs, animations) of solutions.
Q7: Computation research attractive for funding	With regard to your personal research, rate the degree to which you agree or disagree with the following statements:—Computational research is attractive to funding agencies.
Q8: Computation is used across science (learning)	Rate the degree to which you agree or disagree with the following aspects of learning computation:—Computation is used in many science and engineering applications.
Q9: Computation can solve unsolved problems (learning)	Rate the degree to which you agree or disagree with the following aspects of learning computation:—Computation can solve unsolvable (analytical) problems.
Q10: Computation is generalizable (learning)	Rate the degree to which you agree or disagree with the following aspects of learning computation:—Computation is generalizable to many different kinds of problems.
Q11: Computation provides visuals (learning)	Rate the degree to which you agree or disagree with the following aspects of learning computation:—Computation affords visualization (graphs, animations) of solutions.
Q12: Computation focuses attention on modeling	Rate the degree to which you agree or disagree with the following aspects of learning computation:—Computation focuses student's attention on modeling the important physics of a problem.
Q13: Learning computation prepares the workforce	Rate the degree to which you agree or disagree with the following aspects of learning computation:—Learning computation prepares students for the modern scientific workforce.
Q14: Computation allows me to bring new physics in	Rate the degree to which you agree or disagree with the following aspects of learning computation:—Computation allows me to bring new physics into the classroom that I otherwise couldn't.

(Table continued)

TABLE II. *(Continued)*

Short name	Question statement
Q15: Computation allows me to bring new problems in	Rate the degree to which you agree or disagree with the following aspects of learning computation:—Computation allows me to bring new problems into the classroom that I otherwise couldn't.
Q16: Computation instruction considered for new hires	What level of consideration does your department give to undergraduate instruction in computation when making decisions regarding:—Hiring new faculty members
Q17: Computation instruction considered for T&P	What level of consideration does your department give to undergraduate instruction in computation when making decisions regarding:—Tenure and promotion decisions
Q18: Computation instruction considered when allocating resources for major	What level of consideration does your department give to undergraduate instruction in computation when making decisions regarding:—Allocating resources for programs or tracks within the undergraduate major
Q19: Computation instruction considered when allocating resources for individual courses	What level of consideration does your department give to undergraduate instruction in computation when making decisions regarding:—Allocating resources for individual undergraduate courses
Q20: Computation instruction considered when changing service course	What level of consideration does your department give to undergraduate instruction in computation when making decisions regarding:—Changing undergraduate service courses
Q21: Computation instruction considered when changing major course	What level of consideration does your department give to undergraduate instruction in computation when making decisions regarding:—Changing courses for undergraduate majors
Q22: Computation instruction considered for release time	What level of consideration does your department give to undergraduate instruction in computation when making decisions regarding:—Releasing time for faculty to develop computation in undergraduate courses
Q23: Computation instruction considered for pursuing university funding	What level of consideration does your department give to undergraduate instruction in computation when making decisions regarding:—Pursuing university funding
Q24: Computation instruction considered for pursuing grant funding	What level of consideration does your department give to undergraduate instruction in computation when making decisions regarding:—Pursuing grant funding
Q25: Actionable plans to increase computation instruction	Do you have concrete and actionable plans to increase your use of computation in your own undergraduate physics teaching in the next year?
Q26: Highest degree	What is your highest degree?
Q27: Field of degree	In what field did you receive your highest degree?
Q28: Additional field of degree	In what other field did you receive your highest degree?
Q29: Time of degree	When did you obtain your highest degree?
Q30: Learned computation by self-teaching	How did you come to learn computation?—Self-taught
Q31: Learned computation informally on-the-job	How did you come to learn computation?—Informal on-the-job
Q32: Learned computation in formal course	How did you come to learn computation?—Formal course(s)
Q33: Use computation for personal enrichment	How do you personally use computation?—Exclusively for personal enrichment or use
Q34: Use computation in my research	How do you personally use computation?—In my research work
Q35: Use computation in research with students	How do you personally use computation?—To provide research experiences for my undergraduate students
Q36: Do not personally use computation	How do you personally use computation?—I do not use computation.
Q37: Faculty rank	What is your current faculty rank?
Q38: Employment status	As of the Spring 2016 term, what was your employment status
Q39: Tenure status	Are you currently a tenured faculty?
Q40: Gender	What is your gender?
Q41: Race or ethnicity	What is your race or ethnicity?
Q42: HBCU status	Is your institution a historically black college or university?
Q43: Institution type	What type of institution do you work at?
Q44: Highest physics degree offered	What is the highest physics degree offered at your institution?

[1] R. Fuller, Numerical computations in US undergraduate physics courses, Comput. Sci. Eng. **8**, 16 (2006).

[2] N. Chonacky and D. Winch, Integrating computation into the undergraduate curriculum: A vision and guidelines for future developments, Am. J. Phys. **76**, 327 (2008).

[3] M. D. Caballero and L. Merner, Prevalence and nature of computational instruction in undergraduate physics programs across the United States, Phys. Rev. Phys. Educ. Res. **14**, 020129 (2018).

[4] NGSS Lead States, *Next Generation Science Standards: For States, By States* (The National Academies Press, Washington, DC, 2013).

[5] S. Olson and D. G. Riordan, *Engage to excel: Producing one million additional college graduates with degrees in science, technology, engineering, and mathematics, report to the President* (Executive Office of the President, Washington, DC, 2012).

[6] E. Behringer and L. Engelhardt, Guest Editorial: AAPT Recommendations for computational physics in the undergraduate physics curriculum, and the Partnership for Integrating Computation into Undergraduate Physics, Am. J. Phys. **85**, 325 (2017).

[7] C. Henderson and M. H. Dancy, Barriers to the use of research-based instructional strategies: The influence of both individual and situational characteristics, Phys. Rev. ST Phys. Educ. Res. **3**, 020102 (2007).

[8] C. Turpen, M. Dancy, and C. Henderson, Perceived affordances and constraints regarding instructors' use of Peer Instruction: Implications for promoting instructional change, Phys. Rev. Phys. Educ. Res. **12**, 010116 (2016).

[9] C. Henderson, M. Dancy, and M. Niewiadomska-Bugaj, Use of research-based instructional strategies in introductory physics: Where do faculty leave the innovation-decision process?, Phys. Rev. ST Phys. Educ. Res. **8**, 020104 (2012).

[10] https://www.compadre.org/PICUP/.

[11] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevPhysEducRes.15.010114 for an overview of the random forest algorithm and decision tree learning, how to validate random forest models, how to select important features, and known biases of the random forest algorithm.

[12] E. M. Rogers, *Diffusion of Innovations*, 4th ed. (Free Press, New York, 1995).

[13] J. Foertsch, S. B. Millar, L. Squire, and R. Gunter, *Persuading Professors: A Study of the Dissemination of Educational Reform in Research Institutions* (University of Wisconsin-Madison, Madison, WI, 1997).

[14] A. Ho, D. Watkins, and M. Kelly, The conceptual change approach to improving teaching and learning: An evaluation of a Hong Kong staff development programme, Higher Educ. **42**, 143 (2001).

[15] J. B. Ellsworth, *Surviving Change: A Survey of Educational Change Models* (Office of Educational Research and Improvement, Washington, DC, 2000).

[16] C. Henderson, The challenges of instructional change under the best of circumstances: A case study of one college physics instructor, Am. J. Phys. **73**, 778 (2005).

[17] M. Prosser and K. Trigwell, *Understanding learning and teaching: The experience in higher education* (SRHE and Open University Press, Buckingham, England, 1999).

[18] J. R. Taylor and B. A. King, Using computational methods to reinvigorate an undergraduate physics curriculum, Comput. Sci. Eng. **8**, 38 (2006).

[19] M. Johnston, Implementing curricular change, Comput. Sci. Eng. **8**, 32 (2006).

[20] R. F. Martin, Undergraduate computational physics education: Uneven history and promising future, Comput. Sci. Eng. **19**, 70 (2017).

[21] Z. Hazari, G. Sonnert, P. M. Sadler, and M.-C. Shanahan, Connecting high school physics experiences, outcome expectations, physics identity, and physics career choice: A gender study, J. Res. Sci. Teach. **47**, 978 (2010).

[22] P. R. Heron, Effect of lecture instruction on student performance on qualitative questions, Phys. Rev. ST Phys. Educ. Res. **11**, 010102 (2015).

[23] J. Fox, *Applied Regression Analysis, Linear Models, and Related Methods* (Sage Publications, Inc, Newbury Park, CA, 1997).

[24] N. R. Draper and H. Smith, *Applied Regression Analysis* (John Wiley & Sons, New York, 2014).

[25] T. K. Ho, *Proceedings of the Third International Conference on Document Analysis and Recognition, 1995* (IEEE, Bellingham, WA, 1995), Vol. 1, pp. 278–282.

[26] T. K. Ho, The random subspace method for constructing decision forests, IEEE Trans. Pattern Analysis Machine Intell. **20**, 832 (1998).

[27] L. Breiman, Random forests, Mach. Learn. **45**, 5 (2001).

[28] L. Rokach and O. Maimon, *Data Mining with Decision Trees: Theory and Applications* (World Scientific, Singapore, 2014).

[29] T. Fawcett, An introduction to roc analysis, Pattern Recogn. Lett. **27**, 861 (2006).

[30] M. B. Arajo, R. G. Pearson, W. Thuiller, and M. Erhard, Validation of species-climate impact models under climate change, Global Change Biol. **11**, 1504 (2005).

[31] S. Janitza, C. Strobl, and A.-L. Boulesteix, An AUC-based permutation variable importance measure for random forests, BMC Bioinf. **14**, 119 (2013).

[32] R. Daz-Uriarte and S. Alvarez de Andrs, Gene selection and classification of microarray data using random forest, BMC Bioinf. **7**, 3 (2006).

[33] R. S. Olson, W. La Cava, Z. Mustahsan, A. Varik, and J. H. Moore, Data-driven advice for applying machine learning to bioinformatics problems, arXiv:1708.05070.

[34] R Core Team, *R: A Language and Environment for Statistical Computing* (R Foundation for Statistical Computing, Vienna, Austria 2017), https://www.R-project.org/.

[35] T. Hothorn, K. Hornik, and A. Zeileis, Unbiased recursive partitioning: A conditional inference framework, J. Comput. Graph. Stat. **15**, 651 (2006).

[36] C. Strobl, A.-L. Boulesteix, A. Zeileis, and T. Hothorn, Bias in random forest variable importance measures: Illustrations, sources and a solution, BMC Bioinf. **8**, 25 (2007).

[37] C. Strobl, A.-L. Boulesteix, T. Kneib, T. Augustin, and A. Zeileis, Conditional variable importance for random forests, BMC Bioinf. **9**, 307 (2008).

[38] H. Kim and W.-Y. Loh, Classification Trees With Unbiased Multiway Splits, J. Am. Stat. Assoc. **96**, 589 (2001).

[39] K. Polat and S. Gne, Detection of ECG Arrhythmia using a differential expert system approach based on principal component analysis and least square support vector machine, Appl. Math. Comput. **186**, 898 (2007).

[40] W. Chen, X. Xie, J. Wang, B. Pradhan, H. Hong, D. T. Bui, Z. Duan, and J. Ma, A comparative study of logistic model tree, random forest, and classification and regression tree models for spatial prediction of landslide susceptibility, Catena **151**, 147 (2017).

[41] D. T. Ahneman, J. G. Estrada, S. Lin, S. D. Dreher, and A. G. Doyle, Predicting reaction performance in CN cross-coupling using machine learning, Science **360**, 186 (2018).

[42] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan, and B. P. Feuston, Random forest: a classification and regression tool for compound classification and qsar modeling, J. Chem. Inf. Comput. Sci. **43**, 1947 (2003).

[43] J. Wing, S. Weston, A. Williams, C. Keefer, A. Engelhardt, T. Cooper, Z. Mayer, B. Kenkel, M. Benesty *et al.* (the R Core Team), Caret: Classification and Regression Training (2017), R package version 6.0-78, URL https://CRAN.R-project.org/package=caret.

[44] T. Sing, O. Sander, N. Beerenwinkel, and T. Lengauer, Rocr: visualizing classifier performance in R, Bioinformatics **21**, 3940 (2005).

[45] C. Heyde, Central limit theorem, in *Wiley StatsRef: Statistics Reference Online*, edited by N. Balakrishnan, T. Colton, B. Everitt, W. Piegorsch, F. Ruggeri, and J. L. Teugels (2014), ISBN 9781118445112, DOI: 10.1002/9781118445112.stat04559.

[46] R. Hogg, E. Tanis, and D. Zimmerman, *Probability and Statistical Inference* (Pearson Higher Educ., Upper Saddle River, NJ, 2015).

[47] M. B. Wilk and R. Gnanadesikan, Probability plotting methods for the analysis of data, Biometrika **55**, 1 (1968).

[48] R. Diaz-Uriarte, Genesrf and varselrf: A web-based tool and R package for gene selection and classification using random forest, BMC Bioinf. **8**, 328 (2007).

[49] J. Bergstra and Y. Bengio, Random Search for Hyper-Parameter Optimization, J. Machine Learn. Res. **13**, 281 (2012); http://www.jmlr.org/papers/v13/bergstra12a.html.

[50] A. Hapfelmeier, T. Hothorn, K. Ulm, and C. Strobl, A new variable importance measure for random forests with missing data, Stat. Comput. **24**, 21 (2014).