Fall 2020

Ensemble Learning Methods for Educational Data Mining Applications

Joshua Ryan Beemer
Claremont Graduate University

Ensemble Learning Methods for Educational Data Mining Applications

By
Joshua Beemer

Claremont Graduate University and San Diego State University
2020

APPROVAL OF THE DISSERTATION COMMITTEE

This dissertation has been duly read, reviewed, and critiqued by the Committee listed below, which hereby approves the manuscript of

Joshua Beemer

as fulfilling the scope and quality requirements for meriting the degree of

Doctor of Philosophy

Richard Levine, Chair
Department of Mathematics and Statistics, San Diego State University

---

John Angus
Institute of Mathematical Sciences, Claremont Graduate University

---

Barbara Bailey
Department of Mathematics and Statistics, San Diego State University

---

Juanjuan Fan
Department of Mathematics and Statistics, San Diego State University

---

Claudia Rangel-Escareño
Institute of Mathematical Sciences, Claremont Graduate University

---

Approval Date

ABSTRACT OF THE DISSERTATION

Ensemble Learning Methods for Educational Data Mining Applications
by
Joshua Beemer
Doctor of Philosophy in Computational Science - Statistics
Claremont Graduate University and San Diego State University, 2020

Student success efficacy studies are aimed at assessing instructional practices and learning environments by evaluating the success of and characterizing student subgroups that may benefit from such modalities. We develop an ensemble learning approach to perform these analytics tasks with specific focus on estimating individualized treatment effects (ITE). ITE are a measure from the personalized medicine literature that can, for each student, quantify the impact of the intervention strategy on student performance, even though the given student either did or did not experience this intervention (i.e., is either in the treatment group or in the control group). We illustrate our learning analytics methods in the study of a supplemental instruction component for a large enrollment introductory statistics course recognized as a curriculum bottleneck at San Diego State University. As part of this application, we show how the ensemble estimate of the ITE may be used to assess the pedagogical reform (supplemental instruction), advise students into supplemental instruction at the beginning of the course, and quantify the impact of the supplemental instruction component on at-risk subgroups.

Higher Education researchers and Institutional Research practitioners struggle with the analysis of observational study data and estimation of treatment

effects. Propensity score matching has widely been accepted to counteract inherent selection bias in these studies. We present an ensemble learner for propensity score estimation, and consider the use of inverse probability of treatment weighting (IPTW), variance stabilization weighting, and weight truncation to improve treatment effect estimation over propensity score matching.

We run a simulation study to validate the treatment effect and propensity score estimation performance of the ensemble learner compared to logistic regression and random forest within the matching and weighting techniques. The results show that the use of the ensemble learner and variance stabilization with truncation result in the lowest mean squared error for treatment effect estimation. We contribute a new package in the statistical software environment R, `matchED`, that will provide educational researchers with a tool to help analyze student success study data and present actionable results. A tutorial guides the user through the use of each function and it's parameters. A student success intervention is evaluated using the `matchED` package, and we are able to show that the intervention does help reduce an inherent equity gap between students in the intervention and their peers.

DEDICATION

Dedicated to my family and friends who consistently pushed me until the end.

It is difficult to make predictions, especially about the future.

– Niels Bohr

ACKNOWLEDGMENTS

I would like to thank Dr. Richard Levine for guiding me through my master and doctorate programs, and offering me endless opportunities through the years. To my committee Dr. John Angus, Dr. Barbara Bailey, Dr. Juanjuan Fan, and Dr. Claudia Rangel-Escareño thank you for being amazing mentors and supporters of my work. Last but not least to my family and friends for keeping me as close to sane as humanly possible, I love you all.

TABLE OF CONTENTS

PAGE

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

Student success is a top priority in higher education, with universities pushing for increased graduation rates, stronger retention rates, smaller achievement gaps, and overall better academic performance from students (Pelletier, 2019). In order to improve these metrics universities rely on student success programs to target at-risk students and provide timely interventions for them to succeed. Adequately evaluating student success programs is crucial for meeting the expectations of university stakeholders. Universities must continuously look for ways to improve the impact of student success programs and interventions on student learning and success (Rincones-Gómez, 2009).

As a response to the pressure by university stakeholders to uphold student success and evaluate student success interventions, universities look to institutional researchers to find new ways of applying learning analytics and educational data mining methods to educational data (Huebner, 2013). "Learning analytics is the measurement, collection, analysis, and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environments in which it occurs"(1st International Conference on Learning Analytics and Knowledge, 2011). Learning analytics and educational data mining research support data-informed approaches and decisions within the educational setting (Prado et al., 2017). Hora et al. (2017) present that higher education

institutions hold a strong emphasis on making decisions based on evidence, and the use of data to inform these decisions is vital. Researchers have access to a myriad of educational and student data through student information databases, learning management systems, and course assessments. Researchers thus have the opportunity to perform statistical analyses of student success innovations (Spoon et. al., 2016).

He et al. (2018) summarizes that in recent learning analytics literature institutional researchers primarily use logistic regression as their modeling approach to predict student performance metrics. Kučak et al. (2018) looks at the use of machine learning in the educational setting and states a major benefit of machine learning is its ability to utilize educational data and predict student performance accurately. We shift the narrative further by introducing novel ensemble learning methods for individualized treatment effects, and propensity score matching and weighting techniques. Institutional researchers can use these learning analytics techniques to evaluate student success interventions and student performance in higher education.

In Chapter 2 introduce of an ensemble learner for student success studies. We look at a study at San Diego State University with the goal to evaluate student success in a bottleneck course, a course with high withdrawal or failure rate where students get "stuck". In the study we compare students who participated in the supplemental instruction intervention to those that did not. To analyze this intervention, develop an early warning system to identify at-risk students, and

study the impact on at-risk subgroups we look at using an ensemble learner to estimate individualized treatment effects.

Due to the nature of the university setting, most student success studies must be analyzed as observational studies, with random treatment assignment on students being unethical. This becomes a limiting factor for institutional researchers when trying to draw inference. We find that the ensemble learner in combination with matching or weighting methods offers the researcher the ability to control for confounding variables and selection bias, normally handled by randomization, and draw inference from estimated treatment effects.

The simulation study in Chapter 3 will evaluate and validate the ensemble learner compared to logistic regression and random forest for propensity score estimation. A comparison of propensity score matching, inverse probability of treatment weighting, variance stabilization of weights, and truncation of weights is used to better estimate treatment effects. Data was generated to represent possible real world variable types, and "subjects" probability of treatment was simulated with five models varying in complexity, from main effects only to a model with three-way interactions and quadratic terms. Through these comparisons we show that the ensemble learner best estimates propensity scores compared to logistic regression and random forest, and methods to truncate and stabilize the weights provide the best estimate of treatment effects.

In Chapter 4 we contribute a new R-package (R Core Team, 2020), `matchED` that offers institutional researchers an accessible way to estimate propensity scores and treatment effects for their educational data mining applications. We explain

the functions and parameters within the R-package and offer a walk-through tutorial that users can reference as a guide to the package. To show the package in use we present a current student success intervention study.

In Chapter 5 we conclude with a discussion of results from our research, and offer challenges that we encountered along the way. We offer recommendations to further our research and future direction we are considering to take our analysis.

CHAPTER 2

Ensemble Learning for Estimating Individualized Treatment Effects in Student Success Studies

2.1 Introduction

In striving to improve graduation rates and reduce achievement gaps, Universities have experimented with a suite of instructional practices and learning environments (for example, see the 2015 issue 2 of *Peer Review* from the Association of American Colleges & Universities). Broadly speaking, these strategies foster student success and engagement through common and collaborative intellectual experiences, student research and internships, and study abroad experiences (Kuh, 2008) as well as supplemental instruction and instructional technologies (for example see Dawson et al., 2014; Henrie et al., 2015). An analytics goal is identifying at-risk students that will benefit from one or more of these intervention strategies and, early in their college careers, advise these students accordingly. On the flip side, we also must evaluate each instructional practice and each learning environment on at-risk subgroups for purposes of strategic planning, resource allocation, and program development.

We propose an ensemble learning approach to estimate individualized treatment effects (ITE) to characterize at-risk students and assess student success

and retention under intervention strategies. ITE were introduced in the personalized medicine literature (Dorresteijn, 2011) to quantify the difference in an outcome of interest between treatment and control for any subject, whether they experience only the treatment or only the control modality. In our setting, ITE allow us to predict the performance difference between experiencing an intervention strategy or not for each student. We may use these predictions to

- evaluate the success of an intervention;
- characterize student subgroups that may benefit from an intervention;
- evaluate the impact of an intervention on at-risk subgroups;
- quantify the impact of an intervention on individual students; and
- provide an early warning system to advise students into an intervention.

An ensemble learning approach provides a natural analytics environment within which to leverage the wealth of data on students from student information system databases and learning management systems to estimate ITE and student success measures. The student success measures may include categorical outcomes, such as non-repeatable grade in a course (e.g., C or better), graduation success, and retention, or continuous outcomes such as course grade (e.g., on a four-point GPA scale), final exam score, and time to graduation. In a learning analytics setting, a set of base learners are trained and then used to predict the student success outcome of interest for each student. A meta-learner combines these base learner predictions for each student. Moon et al. (2007) proves that in the case of classification, an ensemble average (meta-learner) over a suite of classifiers (base learners) will improve accuracy over a single classifier (single base learner). The

case is not as clear cut in a regression context, though Moreira et al. (2007) presents a number of approaches to create an ensemble with improved prediction accuracy.

To our knowledge, the educational literature contains only a few student success studies that take advantage of ensemble learning. Pardos et al. (2011) considers ensemble methods to combine latent student knowledge predictions from multiple models of data within a tutoring system. Kotsiantis et al. (2010) considers the use of ensemble methods to predict student success in distance learning using three different techniques (WINNOW, naive Bayes, and 1-nearest neighbour). Cortez and Silva (2008) compares several ensemble techniques to assess student performance under binary, multi-class factor, and continuous responses. Though no ensemble learning approach is applied, Jayaprakash (2014) evaluates a set of base learners (logistic, SVM, decision trees, and naive Bayes) for predicting academic risk. To our knowledge, the education literature contains only two applications of estimating treatment effects, in the context of digital learning environments without random assignment. Beck and Mostow (2008) apply learning curve analysis using nonlinear regression to estimate individual student learning in acquiring reading skills. Pardos, Dailey, and Heffernan (2011) apply Bayesian knowledge tracing to study the effectiveness of tutorial help in a math tutoring system.

In Section 2.2, we detail our ensemble learning approach and computation of ITE. In Section 2.3, we step through the applications of ITE in student success studies. For purposes of illustration, we evaluate the success of a supplemental instruction course introduced in a San Diego State University (SDSU) large enrollment introductory statistics course. We stress that the ensemble learning

approach we propose is modular. In our application we fix the set of base learners we consider. However, as a function of application ease, computational cost/complexity, or prediction performance, any base learner may be used as part of the ensemble. Analogously, we introduce stacked generalization (Alpaydin, 2010, Chapter 17) to construct the meta-learner. Again, this component of the ensemble learner is modular, allowing flexibility in choice of meta-learner for combining the base learner predictions. In Section 2.4, we provide a concluding summary, limitations of our proposed approach, and recommendations for future research.

2.2 Analytics Methods

In this section we detail the ensemble learning approach for student success study analytics applications. We then detail predicting individualized treatment effects (ITE) with the ensemble learner.

2.2.1 Ensemble Learning

Ensemble learning entails combining predictions from a set of base learners. Intuitively, the ensemble balances base learners that over fit and under fit the data, with an aim of improving overall prediction accuracy. An ensemble learner will see the greatest gain in predictive performance when combining diverse predictions, that is, base learner predictions that are not highly correlated. A basic ensemble learner is a weighted sum of the predictions from each base learner, weights by minimizing an objective criterion such as mean squared error, likelihood, or entropy (Alpaydin, 2010, Chapter 17). We focus on stacked generalization (Wolpert, 1992) to combine the base learners. The particular form we use is a variation of stacked

regression introduced to the statistics literature by Brieman (1996) and LeBlanc and Tibshirani (1996).

Algorithm 1 presents pseudocode for our proposed ensemble learner. The algorithm requires three data subsets created within nested cross-validation loops. Suppose we have n students in our data set. Let the size of the validation set be denoted by K_E. The first cross-validation loop (steps 1-3) randomly divides the data into n/K_E subsets of students of size K_E. For example, in leave-one-out cross-validation, $K_E = 1$; in ten-fold cross-validation, $K_E = n/10$; etc. In each cycle of the first cross-validation loop, we put aside the K_E students for that subset as a *validation set*. The remaining $n_T = (n - K_E)$ students we call the *ensemble training set*.

We perform K-fold cross-validation on the ensemble training set. That is, we randomly partition the ensemble training set into n_E/K subsets of K students each (step 5). In each cycle of this cross-validation loop (step 6), we put aside the K students for that subset as a *testing set*. We then train each base learner chosen on the *training set* of $n_E - K$ students left. Again, this training may be performed using leave-one-out cross validation by setting $K = 1$. The trained base learners are then used to predict the outcome of interest for each of the K students in the testing set. At the end of this loop (steps 5-6), we have a prediction for each of the n_E students in the ensemble training set from each base learner.

The meta-learner entails a regression (step 7) of the true outcome on the predictions from each base learner for the n_E students in the ensemble training set. The regression coefficients represent the weights for combining the base learners

Algorithm 1 Ensemble Learner: Stacked Generalization

1. `Randomly` partition the data into subsets of size K_E.
2. `Fix` cross-valdiation counter $cv = 1$. (Note that $cv \in \{1, \ldots, n/K_E\}$.)
3. `Label` the subset of K_E students in data partition cv as the validation set and the remaining $n_E = (n - K_E)$ as the ensemble training set.
4. `Choose` L base learners for constructing the ensemble learner.
5. `Randomly partition` the ensemble training set into subsets of size K.
6. For each partition,
   - `Label` the subset of K students as the test set and the remaining $n_E - K$ students as the training set;
   - `Fit` each of the L base learners to the training set;
   - `Obtain` a prediction for each student in the test set from each fitted base learner.

   end-loop over each K-fold cross validation partition.
7. `Regress` true outcome on the predictions from each base learner: L predictions for each student as inputs into the regression model on n_E students.
8. `Fit` each of the L base learners to the ensemble training set.
9. `Obtain` a prediction for each student in the validation set from each fitted base learner from step 8.
10. `Combine` the predictions from step 9 using the regression coefficient estimates from step 7 as weights in the linear combination.
11. `Increment` cv by one.
12. `Repeat` steps 3-10 until $cv > n/K_E$.

into an ensemble prediction. Breiman (1996) notes that the base learner predictions may be highly correlated leading to challenges if linear regression (via ordinary least squares, OLS) is used as the meta-learner. Ridge regression (James et al., 2013, Chapter 6), a common approach in the presence of multi-collinearity, is suggested. As an extension, Reid and Grudic (2009) proposes regularization which also allows for lasso or elastic net (James et al., 2013, Chapter 6) regression techniques. These latter methods provide an alternative method of shrinkage estimation that may select a weight of zero (sparse model) for a base learner. In our application, we find ridge regression sufficient for estimating ITE. However, regularization provides options for stacked generalization to avoid overfitting and improve predictive performance.

The so-called validation set contains students left out of the process for constructing the meta-learner. We thus may use the meta-learner to make predictions for each of the students in the validation set at the conclusion of the outer cross-validation loop (over cv). First, each base learner is trained on the ensemble training set (set 8). A set of predictions is then made for each student in the validation via each base learner (step 9). We thus will have L predictions for each of the n/K_E students in the validation set. These L predictions are combined using the meta-learner (step 10). We thus come out of the outer cross-validation loop with predictions for each student in the data set, predictions made in groups of n/K_E.

With nested cross-validation loops, Algorithm 1 appears computationally costly for large data sets. However, the outer cross-validation loop (steps 1-3;

validation set) is easily performed in parallel on say a cluster computer. The inner cross-validation loop (steps 5-6; ensemble training set) may also be performed in parallel upon identification of the validation set.

2.2.2 Individualized Treatment Effects

In student success studies, we wish to quantify the difference in outcome under an intervention and under a control regime (typically no intervention). Of course the student will typically either experience the intervention or not. A crossover type design or randomized controlled experiment is typically not an option for studying, for example, instructional practices and learning environments. We can apply the ensemble learning algorithm predictions of Section 2.2.1 to compute an individualized treatment effect for each student. Algorithm 2 presents the pseudocode.

Algorithm 2 Individualized Treatment Effects (ITE)

1. `Separate` data into treatment group and control group
2. `Train` the ensemble learner of Algorithm 1 on the treatment group
3. `Train` the ensemble learner of Algorithm 1 on the control group
4. `Obtain` a "under treatment" prediction for control group subjects using the treatment group trained learner from step 2
5. `Obtain` a "no-treatment" prediction for treatment group subjects using the control group trained learner from step 3
6. `Compute` ITE for the control group as the difference of the predicted outcome from step 4 and the observed outcome
7. `Compute` ITE for the treatment group as the difference of the observed outcome and the predicted outcome from step 5

In a given study we will have a set of students that receive the "treatment" (experience the intervention strategy) and a set of students that receive the "control" (do not experience the intervention strategy). We may train an ensemble learner on each group separately using Algorithm 1. We then predict the "no-treatment" outcome for the treatment group students using the ensemble learner trained on the control group. The individualized treatment effect for these treatment group students is the difference of the observed outcome under treatment and the predicted outcome under control (step 6). Analogously, we predict the "under treatment" outcome for the control group students using the ensemble learner trained on the treatment group. The individualized treatment effect for these control group students is the difference of the predicted outcome under treatment and the observed outcome under control (step 7). Overall, the ensemble learner is serving as a best guess of the outcome for the treatment (control) group students if they had experienced the control (treatment). Note that the ITE here are formulated as (outcome under treatment) minus (outcome under control). Thus the treatment group ITE are (observed-predicted) and the control group ITE are (predicted-observed).

2.3 Application: Impact of Supplemental Instruction Section on Student Success in Introductory Statistics

The California State University (CSU) Chancellor's Office instituted the "Promising Practices for Course Redesign" program aimed at improving student success in bottleneck, typically large enrollment courses. Introductory Statistics

was identified by CSU as one such bottleneck course, affecting STEM, business, and quantitatively-oriented non-STEM majors. Of particular concern are repeatable grades (at CSU these are grades of C- or worse as well as a withdrawal, W) which in turn potentially lead to lower (STEM) retention/persistence rates, decreased graduation rates, and increased time to graduation.

The San Diego State University (SDSU) Introductory Statistics course under consideration here (STAT 119) enrolls on the order of 1200 students per semester with DFW rate around 30%. DFW denotes grades of D (1.0 on a four-point grade scale), F (failing grade; 0.0 on a four-point grade scale), and withdrawal from the course. To combat this high DFW rate, one arm of our course redesign project introduced supplemental instruction sections to the course. Each section enrolls 20-30 students and meets twice per week for one hour each. The sections are lead by Statistics graduate student teaching assistants (TA) trained prior to the semester for developing an active problem solving environment in the classroom (Savery, 2006). Rather than students watching TAs solve problems, the sessions entail students working through problems related to the topics of the week. The TAs circulate around the room answering questions and facilitate group/class discussions of common conceptual difficulties. Due to caps in general elective units for major programs, this supplemental instruction section is selected voluntarily by STAT 119 students for one additional credit unit.

The supplemental instruction section differs from the UMKC model of Supplemental Instruction (SI; often capitalized to note this particular implementation) originally proposed in 1973 (Martin and Arendale, 1993). In

particular, though students in our study volunteer into the supplemental instruction section, they must enroll in a one-unit course STAT 119A. Furthermore, STAT 119 students who perform below a 70% on an algebra assessment the first week of classes are strongly encouraged to enroll in the supplemental instruction section. Typical SI implementations use "near-peers", namely students who recently took and succeeded in, above a chosen grade threshold, the given course. The STAT 119A instructors are statistics graduate students. That said, the STAT 119A instructors are trained using the SI peer-assisted learning model, facilitate topic content and study skill discussions much like traditional SI sessions, and are regularly evaluated by a course coordinator (akin to an SI supervisor). See Dawson et al. (2014) for discussion of deviations from the traditional SI model in practice.

We consider a Fall semester offering of supplemental instruction in STAT 119 enrolling 17% of students in the course. We consider three student success outcomes: final exam score (on a scale of 0 to 300), final grade in the course (on a four-point GPA scale), and non-repeatable grade indicator (binary response of whether a student received a grade of 'C' or better). Table 2.1 presents descriptive summaries of the STAT 119 and STAT 119A students over a number of key variables in this study. Variables that are not self-explanatory: admission basis identifies a student as a first-time freshman or transfer student; first-generation college identifies a student as being the first in the immediate family to attend college; quiz 0 is an algebra assessment made at the beginning of the semester; and the AP indicators present whether a student took AP Calculus and AP Statistics in high school. STAT 119 is offered in a standard lecture format and in a hybrid

modality, where the two class meetings each week are divided into one live lecture and one synchronous, but archived, online lecture. Table 2.1 thus reports the percentage of students enrolled in the hybrid offering and average number of online units for each group. The complete set of inputs for our model are presented later in Tables 2.3 and 2.4.

We use this study data to illustrate ensemble learning for performing predictive and learning analytics in student success studies as follows:

- Does the supplemental instruction section work? Quantify the impact of the supplemental instruction section on course success.
- On whom does the supplemental instruction section work? Identify characteristics of students benefitting from the supplemental instruction section.
- By how much does the supplemental instruction section work? Quantify the impact the supplemental instruction section has on student success for individuals and for at-risk subgroups.

We note that though our illustration is specifically for this supplemental instruction section, the analytics work up may be used generally to study any intervention strategy or pedagogical innovation for evaluating outcomes in student success studies.

In the remainder of this section, we first evaluate the ensemble learner for predicting student success. We then step through a series of analyses afforded by the ensemble learner prediction of individualized treatment effects within a student success study. All analyses were performed in the open source statistical software package *R* (R Core Team, 2020) environment. The ensemble learner of Algorithm 1 is performed using leave-one-out cross-validation for the validation set ($K_E = 1$)

and ten-fold cross-validation for the ensemble training-testing ($K = 10$). The base learners used are linear regression (or logistic regression depending on the outcome), lasso regression, classification and regression trees (CART), bagging, boosting, random forest, naive Bayes, linear discriminant analysis, support vector machines, and k-nearest neighbors. We refer the reader to James et al. (2013) for details on these methods. Ridge regression is used to combine the base learners (step 7 of the Algorithm 1).

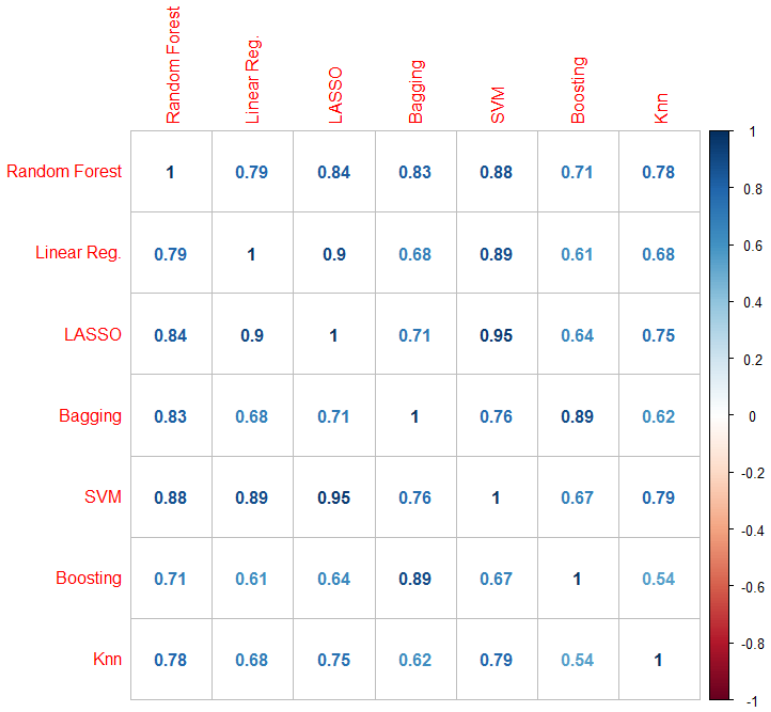

Figure 2.1. Correlation matrix plot for individual learners.

2.3.1 Ensemble Learning Performance Evaluation

Table 2.2 presents the root mean squared error (RMSE) for predicting final exam score (out of 300) and course grade (four-point scale) by the ensemble learner and the individual learners that make up the ensemble. As mentioned in Section 2.1, high correlations are a cause for concern as ensemble learners present the greatest gains when combining individual learners that show diversity in predictions. As Table 2.2 shows, despite the correlated predictions presented in Figure 2.1, the ensemble learner out-performs any single learner for both outcomes.

Table 2.1. Summary statistics on a number of key variables for students enrolled in the STAT 119: Introductory Statistics course as a whole, the subset of STAT 119 students who enrolled in the STAT 119A supplemental instruction course, and the subset of STAT 119 students who did not enroll in the STAT 119A supplemental instruction course. Categorical variables are summarized as percentages for the category names (e.g., "Gender (female)" shows 48% of the STAT 119 students are female). Continuous variables are summarized through the average for that group with standard deviation in parentheses.

	STAT 119 (n=1032)	Enrolled in STAT 119A (n=169)	Not Enrolled in STAT 119 A (n=863)
Gender (female)	48%	64%	45%
EOP	14%	22%	13%
Live in Dorm	52%	32%	56%
Age	19.5 (2.3)	19.7 (2.3)	19.4 (2.3)
Low Income	33%	40%	32%
Level (Freshman, Soph, Junior, Senior)	74%, 12%, 9%, 5%	63%, 20%, 12%, 5%	77%, 10%, 8%, 5%
Admission Basis	92% FTF	92% FTF	92% FTF
First-Gen College	18%	22%	17%
Online Units	1.6 (3.5)	2.5 (4.0)	1.4 (3.4)
Hybrid Class	78%	79%	77%
SAT Math	553 (81)	519 (82)	561 (79)
SAT Verbal	509 (99)	496 (97)	512 (99)
HS GPA	3.47 (0.46)	3.44 (0.45)	3.48 (0.46)
Took Calculus? (AP)	38% (23%)	36% (17%)	38% (24%)
Took Statistics? (AP)	32% (14%)	30% (9%)	32% (15%)
Quiz 0: Score	0.75 (0.24)	0.72 (0.23)	0.75 (0.24)
Quiz 0: Time (min.)	27.9 (11.3)	28.4 (11.4)	27.8 (11.3)
HW 1: Score	0.95 (0.17)	0.96 (0.15)	0.94 (0.17)
HW 1: Time (min.)	80.9 (56.7)	80.1 (51.1)	81.2 (57.8)
Final Exam	0.67 (0.24)	0.71 (0.19)	0.66 (0.25)
% Pass Course	74%	83%	72%

Table 2.2. Ensemble learning performance with respect to final exam score (out of 300), course grade, and non-repeatable grade ('C' or better grade). Root mean squared error (RMSE) is the measure of performance for the former two outputs, accuracy and area under the ROC curve (ROC) for the latter output. The ROC curves appear in Figure 2.2.

	Final Exam Score		Course Grade		'C' or Better Grade	
Method	RMSE	Method	RMSE	Method	Accuracy	AUC
Ensemble	45.3	Ensemble	0.887	Ensemble	80.52%	0.82
Random Forest	45.5	Random Forest	0.893	LASSO	79.94%	0.80
SVM	45.7	LASSO	0.899	SVM	79.07%	0.77
Boosting	45.9	Linear	0.910	LDA	79.09%	0.79
LASSO	46.0	SVM	0.920	Random Forest	78.78%	0.79
Linear Reg.	46.6	Bagging	0.927	Boosting	78.49%	0.71
K-NN	46.7	Boosting	0.928	K-NN	78.49%	0.76
Bagging	46.7	K-NN	0.934	Bagging	77.91%	0.78
				Naive Bayes	77.03%	0.77

Table 2.2 also compares the accuracy of the ensemble learner and individual learners in predicting a non-repeatable grade in the course ('C' or better binary response). The classification ensemble was found by averaging the predicted probabilities from each learner and obtaining an 'optimal' threshold of 0.77 using the `OptimalCutpoints` R package (Lopez-Raton, et al., 2014) for predicting a binary response. Under this threshold, the ensemble learner has the highest classification success. Figure 2.2 presents ROC curves for the ensemble learner and the individual learners. Table 2.2 presents, in the last column, the area under the curve (AUC) for each of these ROC curves. With respect to this ROC comparison, the ensemble learner out-performs the individual learners.

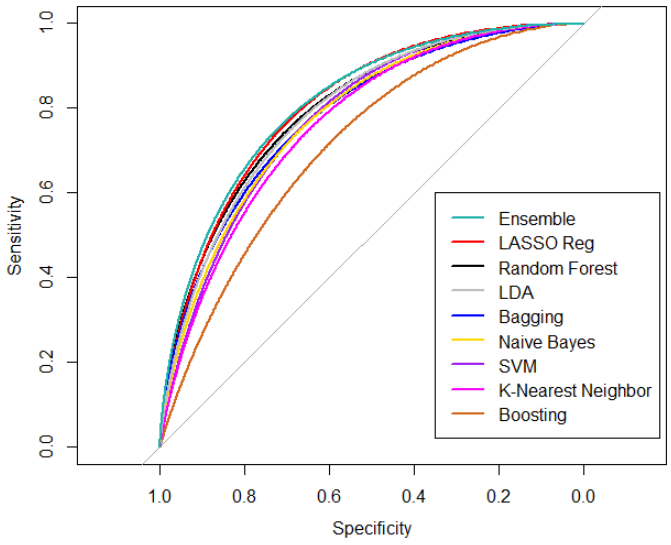

Figure 2.2. ROC curves comparing the ensemble learner with each of the individual learners from Table 2.2.

2.3.2 Success of the Intervention

In our study, the average individualized treatment effect for final exam score was 9.3 with a standard error of 1.48. The average individualized treatment effect for final course grade was 0.45 with a standard error of 0.03. These average ITE are both significantly greater than zero ($p < 0.0001$). Enrolling in a supplemental instruction section not only leads to a moderate improvement in final exam score (on the 300 point scale), but it leads to an increase of almost a half a grade point, on average, in final course grade. (The course is graded on a four-point scale where 4.0=A, 3.0=B, 2.0=C, 1.0=D, and 0.0=F.)

The ITE may be used, say at the beginning of a course, to flag students that may benefit from the supplemental instruction course. We may characterize these at-risk students through demographic and educational markers. To this end, the ITE were split into two subgroups: the top 25% and a comparison group (centered around 0). These subgroups were then analyzed to identify average characteristics of students that benefited the most and were not affected by the recitation course respectively; see Tables 2.3 and 2.4. The inputs on these two tables are self-explanatory for the most part, however a handful require further documentation (see description of Table 2.1 as well):

Math Level:	highest math class completed (algebra, pre-calculus, calculus, ...)
Participation:	score on iClicker questions in Week 2
Quiz 0:	beginning of semester algebra assessment
Calc Level:	applied calculus, calculus 1, calculus 2, calculus 3

Learning Community:	specialized dorm
Compact Scholar:	program partnership with a local school district
First-Gen Some College:	no college degrees in the family
Admission Basis:	first-time freshman or transfer student.

Tables 2.3 and 2.4 suggest that students that may benefit the most from the supplemental instruction course have weaker educational preparation (significantly lower SAT Math, HS GPA, math level, and previous experience with calculus and statistics), and performed worse at the beginning part of the course (lower clicker score, quiz 0 grade, and performance on homework 1). Furthermore, those students are significantly more likely to be EOP, low income, first-generation, commuter, and/or part-time students. These findings could be considered further as early at-risk indicators for success in the course.

2.3.3 Subgroup Analysis

Student success efficacy studies include not only broad-sweeping evaluations of an intervention for students in general, but focus on the impact of the intervention on pre-defined at-risk subgroups. As an example, the STAT 119 class enrolled 32 students from an underrepresented minority (URM) group. For sake of masking, we do not identify the specific URM group. These 32 students displayed ITE significantly greater than zero ($p = 0.007$; the average individualized treatment effect in this group is +24 with a standard deviation of 51). The students scored a 205 out of 300 (68%) on the final exam (standard deviation 44).

Table 2.5 presents characteristics of the 32 students compared to the other 1030 students enrolled in the course. Of note, this URM group contained significantly more EOP, low income, Pell-eligible, transfer, commuter students.

2.3.4 Impact of the Intervention Strategy

The previous two sub-sections considered the impact of the supplemental instruction course on groups of students. The ITE may be used as a form of personalized learning, for each individual student determining if they may benefit from a given intervention strategy. As an illustration, we characterize students who would be predicted to improve their course performance by a letter grade if they had enrolled in the supplemental instruction course. These example students are based on actual students from the STAT 119 class. However, for the sake of confidentiality, we used the STAT 119 students to identify key inputs and then fabricated this group of students for illustration purposes. That said, we also are presenting the summary statistics with a qualifier, rather than exact values, so

there is no chance specific students may be identified. All of these students are envisioned to enroll in the hybrid course section.

- *F→C student, predicted treatment effect of 130.* Female, Freshman, Pell grant, Kinesiology, commuter student with low SAT math and verbal scores. Has experience with online courses, but no previous statistics or calculus courses. Scored above 85% on quiz 0, and handed HW 1 in on time scoring above 95%. Final exam score of 37%.
- *F→C student, predicted treatment effect of 111.* Male, URM, Freshman, Finance, commuter student with HS GPA below 3.0. Did not take Statistics nor Calculus, scored below 70% on quiz 0, but scored 97% on HW 1. Final exam score 54%.
- *C→B student, predicted treatment effect of 51.* Female, International, first generation, EOP, Freshman Management student living on campus, with HS GPA below 3.5 and low SAT math and verbal scores. Took AP Calculus; but scored below 55% on quiz 0 and below 75% on HW 1. Final exam score of 62%.
- *C→B student, predicted treatment effect of 45.* Male, first-generation, Freshman, Undeclared, commuter student with HS GPA below 3.2 and no experience with online courses. Scored above 90% on quiz 0 and 100% on HW 1. Final exam score of 61%.
- *C→B student, predicted treatment effect of 39.* Female, URM, Pell grant, Senior, International Security and Conflict Resolution, commuter student with HS GPA below 3.3. No previous statistics nor calculus courses; scored below 50% on quiz 0 and above 90% on HW 1. Final exam score of 73%.
- *B→A- student, predicted treatment effect of 13.* Male, International, first generation, Pell grant, Sophomore, Marketing, commuter student with low SAT verbal score but high SAT math score. No previous statistics nor calculus courses; scored above 80% on quiz 0 and 100% on HW 1. Final exam score of 78%.
- *B→A student, predicted treatment effect of 22.* Female, URM, first generation, EOP, Pell grant, Freshman International Business student living on campus with low SAT math and verbal scores. No previous statistics nor calculus courses; scored above 70% on quiz 0 but did not submit HW 1. Final exam score of 85%.

- *B→A student, predicted treatment effect of 38.* Female, Pell grant, Freshman, Economics student living on campus with solid SAT math and verbal scores. Had calculus, but no previous statistics course; did not take quiz 0 nor submit HW 1. Final exam score of 90%.

2.4 Discussion

We propose using an ensemble learning approach to make predictions in student success studies of intervention strategies such as instructional practices and learning environments, which we will address further in Chapter 4. In our application evaluating success of a supplemental instruction session in a large enrollment introductory statistics course, we found that the ensemble learner out-performed the base learners in classifying a repeatable grade (C- or worse) and predicting final exam score. In this application, base learner predictions were highly correlated, limiting the predictive performance of the ensemble learner. Applications with more diverse predictions will show markedly better performance by the ensemble learner. We also introduced the concept of individualized treatment effect to evaluate an intervention strategy in student subgroups, identify at-risk students that may benefit from the intervention strategy, and quantify the impact of an intervention to advise individual students into that intervention. As part of the illustration, we presented a set of "example students" that provides further insight on characteristics to be considered when developing early warning systems for student success.

The application of our approach found that students enrolling in the supplemental instruction course performed significantly better than students who did not enroll with respect to final exam score and course grade in a large

enrollment introductory statistics course. These results align with findings in the SI effectiveness review article of Dawson et al. (2014). Of particular note, Dawson et al. (2014) summarize a study by Hodges et al. (2001) where students were either mandated to attend SI sessions (25% of students) or voluntarily attend SI sessions (25% of students). While both groups of SI attendees performed significantly better with respect to DFW-rate than non-attendees, the group mandated to attend SI performed better than the group attending SI voluntarily. Our implementation required students to enroll in, and regularly attend supplemental instruction sessions. A beginning of semester algebra assessment was also used to strongly encourage students. While not a mandate, our model follows more closely to the mandatory attendance of Hodges et al. (2001). Of course none of the SI effectiveness studies provide in-depth subgroup analysis nor individual assessments as we are able to perform through individualized treatment effects.

As a final comment, our proposed application of individualized treatment effects is not limited to course-level analytics problems of the type considered in this paper. ITE may be applied broadly to learning analytics and academic analytics tasks, in the terminology of Long and Siemans (2011). These problems include evaluation of program/department, institutional, and state/national driven intervention strategies for at-risk subgroups. The ITE approach also allows for flexibility in the array of outcomes to assess in these arenas as well, for example program success, (STEM) program retention, time to graduation, graduation rates, and student engagement. As illustrations of learning analytics applications at different scales, we highlight three:

- At a system level, California State University (CSU) recently proposed a "Graduation Success Initiative" (`http://graduate.csuprojects.org/`), setting graduation rate goals for each of its 23 campuses. To justify funding from the state legislature, the CSU will need to assess treatment effects relative to the success of programs aimed at achieving these goals, with respect to the impact of the initiative on at-risk subgroups, and for a cost/benefit analysis.
- At an institution level, major concerns at Universities across the country are STEM persistence and closing the achievement gap (PCAST, 2012). The SDSU Compact Scholar program, briefly mentioned in Section 3, represents a program aimed at improving student engagement and graduation rates to this end in a local school district. Individualized treatment effect estimates are critical to improving program educational practices and evaluating program students relative to graduation success benchmarks and key learning outcomes.
- At a College or Department level, individualized treatment effects are critical for evaluating, for example, new online degree and certificate programs or advising systems and strategies. Again focus is on time to graduation or time to enter major, graduation success, and post-graduation success measures.

Limitations

In our application, the ensemble learner presented the best predictive performance across each outcome measure considered. Furthermore, no single learner came out on top across all of the outputs. See Table 2.2 for performance details. However, the ensemble learner out-performed the best single learner by less than one percentage point in accuracy and 0.02 in ROC AUC for predicting a C or better grade, and less than 0.01 in RMSE for predicting course grade. A user may thus decide to employ a single learner such as LASSO, which performed close to best across the board. From both a computational complexity and interpretable machine learning perspective, LASSO is less computationally expensive (i.e., faster

to fit) and allows for the relationship between inputs and the output to be explained through coefficient estimates.

Table 2.3. ***Final exam outcome:*** **Summaries for students falling in the top 25% in ITE and students falling in a similarly sized comparison group with ITE of zero on average. The p-values are from significance tests between the two groups on each input. The top part of the table considers continuous-valued inputs, presenting the mean value and standard deviation in parentheses for each. The bottom part of the table considers categorical inputs. Except for the multi-category inputs, the features are ordered according to percent difference between the top 25% and comparison groups.**

Continuous inputs	Top 25%	Comp Group	p-value
Homework 1, Days Late	1.60 (7.79)	0.02 (0.19)	0.00
Online Units	2.48 (4.24)	1.26 (3.20)	0.00
Calc Level	0.32 (0.48)	0.61 (0.73)	0.00
Participation Week 2	0.70 (0.45)	0.92 (0.26)	0.00
Quiz 0, Grade	0.65 (0.28)	0.83 (0.16)	0.00
SAT Math	516.83 (75.25)	599.67 (78.85)	0.00
Math Level	4.48 (1.41)	5.16 (1.57)	0.00
SAT Verb	479.83 (99.73)	538.04 (94.94)	0.00
HS GPA	3.3 (0.49)	3.66 (0.35)	0.00
Homework 1, Grade	0.90 (0.24)	0.99 (0.04)	0.00
Term Units Attempted	14.07 (2.16)	14.39 (1.93)	0.09
Homework 1, Time in Minutes	84.13 (54.39)	78.21 (50.14)	0.22
Quiz 0, Time in Minutes	28.54 (13.93)	27.24 (9.00)	0.23
Age	19.99 (2.86)	19.13 (1.74)	0.00
HS Grad Year	2011.48 (2.83)	2012.31 (1.70)	0.00
Final Exam	172.3 (55.44)	244.74 (46.61)	
Treatment Effect	75.72 (33.37)	0.01 (9.18)	

Categorical inputs	Top 25%	Comp Group	p-value
Compact Scholar	12%	3%	0.00
First-Generation College	25%	10%	0.00
EOP Student	21%	9%	0.00
Part-Time Student	51%	23%	0.00
Learning Community	14%	28%	0.00
Live in Dorm	35%	67%	0.00
Stat AP	10%	19%	0.01
Low Income	41%	22%	0.00
First-Generation Some College	44%	24%	0.00
Gender (female)	50%	45%	0.27
Location of Last Math Course			0.00
SDSU	25%	14%	
HS	60%	81%	
TRANS	15%	5%	
Calc AP			0.00
0	83%	65%	
1	16%	28%	
2	1%	7%	
College			0.07
Business	20%	12%	
Sciences	47%	54%	
Liberal Arts	33%	34%	
Admin Basis			0.00
FTF	69%	88%	
LD	3%	5%	
UD	28%	7%	

Table 2.4. ***Course grade outcome:*** **Summaries for students falling in the top 25% in ITE and students falling in a similarly sized comparison group with ITE of zero on average. The p-values are from significance tests between the two groups on each input. The top part of the table considers continuous-valued inputs, presenting the mean value and standard deviation in parentheses for each. The bottom part of the table considers categorical inputs. Except for the multi-category inputs, the features are ordered according to percent difference between the top 25% and comparison groups.**

Continuous inputs	Top 25%	Comp Group	p-value
Homework 1, Days Late	0.85 (6.66)	0.22 (1.93)	0.16
Calc Level	0.38 (0.53)	0.51 (0.68)	0.02
Online Units	1.79 (3.79)	1.51 (3.43)	0.39
Participation Week 2	0.76 (0.42)	0.86 (0.34)	0.00
Math Level	4.62 (1.45)	5.02 (1.57)	0.00
SAT Math	535.88 (80.41)	567.08 (81.06)	0.00
Quiz 0, Grade	0.73 (0.24)	0.77 (0.2)	0.04
Homework 1, Time in Minutes	81.51 (49.52)	85.6 (59.7)	0.41
Homework 1, Grade	0.94 (0.18)	0.97 (0.12)	0.06
SAT Verb	502.79 (104.47)	518.12 (97.71)	0.10
HS GPA	3.43 (0.47)	3.53 (0.48)	0.02
Quiz 0, Time in Minutes	28.94 (11.84)	28.17 (10.14)	0.44
Term Units Att.	14.07 (2.16)	14.39 (1.93)	0.09
Age	19.61 (2.65)	19.23 (1.84)	0.07
HS Grad Year	2011.84 (2.63)	2012.21 (1.78)	0.07
Final Grade	1.71 (1.23)	3.21 (0.90)	
Treatment Effect	1.09 (0.49)	-1.17 (0.37)	

Categorical inputs	Top 25%	Comp Group	p-value
Stat AP	10%	20%	0.01
Learning Community	15%	25%	0.01
Compact Scholar	9%	6%	0.29
Part-Time Student	38%	28%	0.03
EOP Student	17%	13%	0.37
First-Generation College	22%	17%	0.21
First-Generation Some College	39%	31%	0.08
Live in Dorm	48%	59%	0.01
Low Income	35%	29%	0.17
Gender (female)	48%	51%	0.52
Location of Last Math Course			0.00
SDSU	20%	12%	
HS	70%	80%	
TRANS	10%	7%	
Calc AP			0.15
0	77%	73%	
1	21%	21%	
2	2%	5%	
College			0.19
Business	16%	12%	
Sciences	56%	52%	
Liberal Arts	28%	35%	
Admin Basis			0.01
FTF	72%	82%	
LD	4%	5%	
UD	24%	13%	

Table 2.5. Average student characteristics for the 32 students from an underrepresented minority group in STAT 119. Parenthetical values are standard deviations except in the last two rows which report the proportion of students taking AP Calculus and AP Statistics. The admission basis row presents percentage of students admitted as first time freshman (FTF; not transfer students).

	URM group	Remainder of class
Gender (female)	53%	48%
EOP	56%	14%
Live in Dorm	38%	51%
Age	19.1 (0.9)	19.5 (2.3)
Low Income	72%	33%
Level (Freshman, Soph, Junior, Senior)	65%, 28%, 6%, 0%	74%, 12%, 9%, 5%
Admission basis	47% FTF	92% FTF
First-Generation College	19%	18%
Online units	2.25 (2.95)	1.61 (3.48)
Hybrid class	81%	78%
SAT Math	477 (80)	553 (81)
SAT Verbal	481 (90)	509 (99)
HS GPA	3.53 (0.32)	3.47 (0.46)
Took calculus? (AP)	34% (28%)	40% (26%)
Took statistics? (AP)	31% (16%)	33% (14%)

CHAPTER 3

Simulation Study: Ensemble Learning Validation

3.1 Introduction and Motivation

Observational studies are at the heart of data analytics for institutional and student success research. With these studies comes a lack of randomization and therefore biased causal inferences from potential confounding variables. Randomized experiments would normally be our go to solution, but due to ethical reasons are not applicable for institutional research. In order to interpret accurate treatment effects from an observational study we look to matching techniques. Matching pairs a treated subject with a non-treated subject that have the nearest propensity score, the probability that a subject will receive the treatment which are based on the subject's observed covariates. These pairings balance characteristics among subjects and helps to control selection bias (Rosenbaum and Rubin, 1983).

In 2017 San Diego State University ran a pilot study for Supplemental Instruction (SI) as an intervention for at risk students in bottleneck courses on campus. SI, based on the model engineered at the University of Missouri-Kansas City in 1973, allows students to voluntarily enroll in a supplemental peer guided course that meets twice a week outside of class, and follows the main course in subject matter. SI gives students the opportunity to discuss topics, work problems,

and ask questions in a more intimate group setting (Martin and Arendale, 1993; Dawson et al., 2014). It was important to analyze the treatment effects for this intervention to assess improvement of student success in the bottleneck courses. Dawson et al. (2014) summarize that SI participation improves mean exam scores and decreases course failure, but offer the caveat that due to the voluntary nature of SI these findings are based on observational studies and not randomized controlled trials. Guarcello et al. (2017) expands this idea stating that "self-selection may bias commonly practiced analyses towards positive intervention effects" and gives the example of "already higher performing students disproportionately attending SI sessions which could have, in and of themselves, no significant positive effects." To account for volunteer bias, that may influence SI's intervention effect, matching was used to match those who attended SI with those who did not.

After balancing the treatment and control group's covariates through matching we showed "the odds of passing the course (i.e., final course grade of C or better) for students who attended at least one SI Session were 2.2 times higher than those who did not attend any SI Sessions ($n = 299$; p value = 0.006; 95% CI of 1.3–3.8)" and "students who attended two or more SI Sessions were 2.8 times more likely to pass the course than those who did not ($n = 196$; p-value = 0.03; 95% CI of 1.2–6.9)" (Guarcello et al., 2017).

With the success of the SI pilot study and the corresponding paper, Guarcello et al. (2017), we have pushed to further improve how students are matched for institutional research. The aforementioned study used simple propensity scores estimated by logistic regression to match treated and control

subjects. We propose looking beyond logistic regression, to other models, to enhance propensity score estimation, adding propensity score weighting to better estimate the treatment effect, and include truncation and stabilization methods to further improve weighting (Austin, 2015). These methods provide an alternative to logistic regression, when logistic regression struggles with interactions and non-linearities (He et al., 2018), and large number of covariates affect propensity score estimation accuracy (McCaffrey et al., 2005). We will evaluate these refinements through a simulation study. Wrapping these matching techniques in accessible software, a R (R Core Team, 2020) package, for educational researchers and any persons analyzing observational studies in general.

Ensemble learning methods have proven to be a useful tool in analyzing treatment effects within student success studies as shown in Beemer et al. (2017), which is presented in Chapter 2. In the article, an ensemble learner leverages a group of base learners, statistical models, by weighting and combining the predictions from each base learner to obtain an overall better prediction for the true outcome. These ensemble learner predictions allow for treatment effects to be computed, and give the researcher the ability to identify at risk students within the study. With the previous success of ensemble learning methods over individual base learners, we propose to add an ensemble learner as a third model for propensity score estimation. An ensemble learner should provide more accurate propensity score estimation translating to better treatment effect estimation through matching and weighting methods.

3.2 Methods: Ensemble Learner

In Beemer et al. (2017) and Chapter 2 the ensemble learner uses a mix of stacked generalization and stacked regression. Using cross-validation and looping over the data, each observation has its own prediction from each individual learner. Based on the literature of Wolpert (1992), Breiman (1996) and LeBlanc and Tibshirani (1996), the ensemble weights the predictions from the individual learners using a ridge regression. Ridge regression uses a penalty function to minimize coefficients of covariates that are weak predictors of the outcome (James et al. 2013, Chapter 6). In this case, the predictions from the individual learners are the covariates and the coefficients act as the weights, which in linear combination become the ensemble learner.

We present an updated version of the ensemble learner presented in Beemer et al. (2017) and Chapter 2. The ensemble learning method starts with k-fold cross-validation: randomly split the data into subsets of size k, removing one subset and training the base learners on the remaining subsets. The trained base learners are then used to make predictions for the subset that was removed. This "leave one out" method is repeated using the next subset until predictions are made for all observations. The predictions are then stacked, but instead of using a ridge regression to weight the predictions, as described above from Beemer et al. (2017), a random forest approach is used to regress the true outcome against the predictions. We mention in Chapter 2 that the ensemble learner presented the best predictive performance across each outcome measure considered, but only slightly out-performed the base learners. We look to use random forest as our new ensemble learner to improve the ensembles overall performance. In medical research

random forest has proven to be a reliable meta-learner (Wang et al, 2018), and a good alternative to regression methods in educational research (Spoon et al., 2016; He et al., 2018).

Algorithm 3 Ensemble Learner

1. `Randomly` partition data into cross-validation subsets, size k.
2. `Leave one` subset out as test set, remaining subsets are the training set.
3. `Train` base learners on training set.
4. `Predict` test set using trained base learners.
    - Repeat steps 2-4 leaving out a different cross-validation subset as the test set.
    - Continue until all cross-validation subsets have been the test set only once, and have predictions.
5. `Stack` predictions from each base learner in a data frame.
6. `Weight` the stacked predictions from the base learners using a random forest.
7. `Weighted predictions` become the final ensemble learner prediction.

We add the ensemble learner as a third model to compare propensity score estimation for matching and weighting methods. The ensemble will have the ability to weight the predictions from logistic regression, random forest, boosting, bagging, k-nearest neighbor, support vector machines, neural networks, and naive Bayes. A weighted combination of these base learner predictions should provide a more accurate propensity score estimation than any one base learner.

3.3 Methods: Matching

3.3.1 Propensity Score Matching (PSM)

Propensity scores, e_i, are the probability of a subject's assignment to a treatment, while taking into account the subjects characteristic variables,

$$e_i = P(Z = 1|X).$$

Here Z is a binary indicator of whether a subject is in the treatment group (Z=1) or not (Z=0) and X represents all observed inputs. Propensity scores rely on two primary assumptions. First, treatment assignment is independent of the outcome conditional on the observed covariates,

$$(Y(1), Y(0)) \amalg Z|X$$

where Y(1) and Y(0) are the possible subject outcomes for the treatment and control groups. This assumption is known as the "no unmeasured cofounders" assumption, meaning that all variables that affect the outcome and treatment assignment have been measured. Second, every subject has a probability of receiving the treatment greater than zero,

$$0 < P(Z = 1|X) < 1.$$

Together the two assumptions establish if the treatment assignment is strongly ignorable. Conditioning on the propensity score supports obtaining unbiased average treatment effect estimates (Rosenbaum and Rubin, 1983). Propensity scores can be used to estimate treatment effects for observational studies, and

propensity scores act as a balancing score. As a balancing score, treated and control subjects with similar propensity scores will have similar characteristic variables (Rosenbaum and Rubin, 1983). The propensity score method entails the following steps. First estimate the propensity score for each subject via a predictive model such as logistic regression or random forest. Second, starting with a random treated subject, match that subject to the nearest neighbor propensity score from the control group, and remove that control subject from the pool of future matches. We continue this matching until all treated subjects are matched with a control subject. The final matched set will have an equal number of treated to control subjects, with a goal of having a balanced distribution of covariates for both treated and control groups.

Logistic regression is typically used to estimate propensity scores where treatment status is regressed on a set of observed covariates (Austin, 2005). Logistic regression is a strong tool for statistical analysis, however as McCaffrey, et al. (2005) points out, large numbers of covariates tend to hurt its ability to accurately estimate propensity scores, as a result of multi-collinearity. Non-linearities and interaction terms can also increase the number of covariates and can add to over-fitting if iterative model building and variable selection are not performed, further affecting propensity score estimation. Due to many demographic variables in institutional research, this can be a common hurdle in analysis. In order to account for large amounts of covariates, interactions, and non-linear terms we look to the use of random forests. Random forests have shown

great success in educational research and for analyzing student success from pedagogical interventions (Spoon et al., 2016; He et al., 2018).

Random forest selects subsets of training data to grow individual trees by using bootstrap aggregation. Then at each split, in a tree, a small subset of covariates are chosen randomly, from which a single covariate is selected to create the decision rule at that split. Each split within a tree is binary and leads to another split or terminal node. The individual trees continue to grow until the terminal nodes are homogeneous, a preset tree depth is met, or a minimum number of observations is reached in a terminal node. This process is repeated to grow many trees, which becomes the forest. From this forest, the response variable is predicted as a majority vote or average, for classification or regression respectively, of the predictions from all the trees (Brieman, 2001). Unlike logistic regression, random forest is unaffected by interactions and non-linearities as a result of the binary splits, and by taking bootstrapped samples for each tree, over-fitting is reduced (Lee, 2010).

3.4 Methods: Weighting

Austin and Stuart (2015) offers inverse probability of treatment weighting, variance stabilization, and truncation of weights as ways to better estimate treatment effects in observational studies. These methods give an educational researcher alternatives to propensity score matching, while still accounting for observed covariates in the study. This section will walk-through the three propensity score weighting methods.

3.4.1 Inverse Probability of Treatment Weighting

Inverse probability of treatment weighting (IPTW) adjusts the under-represented and over-represented subjects within the control and treatment groups by assigning weights,

$$w_i = \frac{Z_i}{e_i} + \frac{1 - Z_i}{1 - e_i}$$

where Z_i and e_i are the treatment indicator and propensity score ($i = 1, \ldots, n$, for n of observations), respectively. IPTW gives higher weights to those in the treatment group with low propensity scores and those in the control group with high propensity scores, giving a more accurate estimation of the treatment effect (Rosenbaum, 1987). We propose to use IPTW in combination with regression models, as referenced in Austin (2015), to improve estimation of causal treatment effects compared to propensity score matching.

3.4.2 Variance Stabilization of Weights

When the propensity scores are small in the treatment group and close to one in the control group, IPTW will assign a large weight to those subjects. Thus a small group of subjects carry a large proportion of the propensity score weight leading to potentially poor treatment effect estimation. To account for the increase in variability, we stabilize the weights by multiplying the treatment indicator, Z_i by the marginal probability of treatment, $Pr(Z = 1)$, and control, $Pr(Z = 0)$, in the overall sample (Austin, 2015). The adjusted weight is

$$w_i = \frac{Z_i Pr(Z = 1)}{e_i} + \frac{(1 - Z_i)Pr(Z = 0)}{1 - e_i}.$$

3.4.3 Truncation of Weights

Lee, Lessler, and Stuart (2011) proposes trimming or truncating the weights assigned by IPTW, again to account for large weights being assigned to small or close to one propensity scores within treatment and control groups, respectively. The truncation is done by designating a minimum and maximum threshold, and if weights exceed the threshold they are set to that threshold (Austin, 2015). We chose a threshold of the 10th and 90th percentiles to truncate the propensity score weights, after evaluating minimizing bias at different thresholds.

3.4.4 Methods Summary

The simulation study, described in the following section, will assess the different options that will be available to practitioners in the `MatchED` R-package. Logistic regression, random forest, and an ensemble learner will be available for propensity score matching, inverse probability of treatment weighting, variance stabilization of weights, and truncation or trimming of weights. This assessment will guide the input and parameter settings relevant to users.

3.5 Simulation Study and Design

We use a simulation study to evaluate propensity score matching and weighting, looking beyond logistic regression and random forest to an ensemble learner, and adding truncation and variance stabilization methods to improve weighting, all to better estimate treatment effects. In this section we explain how the data is generated and the measures of interest we look at to evaluate treatment

effect estimation. We present results from the study, and offer recommendations on model choice for treatment effect estimation.

3.5.1 Data Generation

Data generation follows previous work by Hallett (2014), Setoguchi et al. (2008), and Su (2006). Eight covariates (x_1 to x_8) were simulated to represent possible variable types in real world studies. These covariates were generated as follows:

- x_1 is generated from a Bernoulli distribution with probability of success p=0.5.
- x_2 is a nominal variable with 5 levels (A, B, C, D, E), with probabilities (10%, 20%, 30%, 20%, 20%).
- x_3, x_4 are generated from a discrete uniform distribution from 0 to 1 with intervals of 0.2, making them ordinal variables with 5 levels (0.2, 0.4, 0.6, 0.8, 1.0), with equal probabilities.
- x_5 to x_8 are generated to be continuous variables from a discrete uniform distribution from 0 to 1 with intervals of 0.02.

A binary treatment status (Z) is generated from a Bernoulli distribution with parameter p, which is the true propensity score. The true propensity score was estimated using a logistic regression,

$$P(Z = 1|X) = \frac{1}{1 + e^{-\beta f(x)}},$$

where the function $f(x)$ is based on models found in Setoguchi et al. (2008). Setoguchi et al. presents seven models that use covariates in varying scale of additivity and linearity, integrating different interaction terms, to determine probability of treatment selection. For this simulation study four models from Setoguchi et al. and a fifth model containing three-way interactions and

non-linearity terms were adopted to estimate the true propensity score. The five treatment selection models are as follows:

- Model A: Additivity and linearity (main effects only)

$$P(Z=1|X) = (1+exp[-(\beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_3+\beta_4X_4+\beta_5X_5+ \beta_6X_6+\beta_7X_7+\beta_8X_8)])^{-1}$$

- Model B: Moderate non-linearity (three quadratic terms)

$$P(Z=1|X) = (1+exp[-(\beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_3+\beta_4X_4+\beta_5X_5+ \beta_6X_6+\beta_7X_7+\beta_8X_8+\beta_9X_3^2+\beta_{10}X_5^2+\beta_7X_7^2)])^{-1}$$

- Model C: Mild non-additivity (four two-way interaction terms)

$$P(Z=1|X) = (1+exp[-(\beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_3+\beta_4X_4+\beta_5X_5+ \beta_6X_6+\beta_7X_7+\beta_8X_8+\beta_9X_3X_4+\beta_{10}X_4X_5+ \beta_{11}X_5X_6+\beta_{12}X_6X_7)])^{-1}$$

- Model D: Moderate non-additivity and non-linearity (ten two-way interaction terms and three quadratic terms)

$$P(Z=1|X) = (1+exp[-(\beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_3+\beta_4X_4+\beta_5X_5+ \beta_6X_6+\beta_7X_7+\beta_8X_8+\beta_9X_3^2+\beta_{10}X_5^2+\beta_7X_7^2+ \beta_{12}X_3X_4+\beta_{13}X_4X_5+\beta_{14}X_5X_6+\beta_{15}X_6X_7+ \beta_{16}X_7X_8+\beta_{17}X_3X_8+\beta_{18}X_5X_7+\beta_{19}X_4X_8+ \beta_{20}X_3X_5+\beta_{21}X_6X_8)])^{-1}$$

- Model E: Severe non-additivity and non-linearity (six two-way interaction terms, four three-way interaction terms and one quadratic term, one cubic polynomial and one square root term)

$$P(Z=1|X) = (1+exp[-(\beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_3+\beta_4X_4+\beta_5X_5+ \beta_6X_6+\beta_7X_7+\beta_8X_8+\beta_9X_4^2+\beta_{10}X_6^2+ \beta_{11}\sqrt{X_8}+\beta_{12}X_3X_4+\beta_{13}X_4X_5+\beta_{14}X_5X_6+ \beta_{15}X_6X_7+\beta_{16}X_7X_8+\beta_{17}X_3X_8+\beta_{18}X_3X_5X_7+ \beta_{19}X_4X_6X_8+\beta_{20}X_3X_4X_5+\beta_{21}X_6X_7X_8)])^{-1}$$

Model coefficients were corrected to keep treatment selection around 25% for all five models. This is the average percent of treatment selection found in the Supplemental Instruction study with which we are motivating our methods and software.

Table 3.1. Coefficients Used for Each Covariate Model

Model	β_0	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	β_{10}
A	-0.5	-0.5	1.2	-1.0	-0.62	-0.7	-0.4	0.6	0.2		
B	-0.5	-0.5	-1.2	-1.2	-0.72	0.7	0.4	0.6	0.2	0.3	-0.4
C	-0.5	-0.5	-1.2	-1.2	-0.72	0.7	0.4	0.6	0.2	0.3	-0.4
D	-0.5	-0.5	-1.2	-1.2	-0.72	0.7	0.4	0.6	0.2	0.3	-0.4
E	-0.5	-0.5	-1.2	-1.2	-0.72	0.7	0.4	0.6	0.2	0.3	-0.4
Model	β_{11}	β_{12}	β_{13}	β_{14}	β_{15}	β_{16}	β_{17}	β_{18}	β_{19}	β_{20}	β_{21}
A											
B	1.1										
C	1.1	0.46									
D	1.1	-0.2	0.42	-0.8	0.9	-1	0.32	-0.45	0.36	-0.47	0.35
E	1.1	-0.2	0.42	-0.8	0.9	-1	-0.32	-0.45	-0.36	0.47	0.35

Two sets of continuous outcomes were calculated using the covariates and treatment status, as previously detailed, using a simple linear model,

- Model 1:

$$Y = \alpha_{00} + \alpha_0 Z + \alpha_1 X_1 + \alpha_2 X_2 + \alpha_3 X_3 + \alpha_4 X_4 + \alpha_5 X_5 + \alpha_6 X_6 + \alpha_7 X_7 + \alpha_8 X_8 + \varepsilon,$$

  and a more complex model that incorporates non-linear and two-way interaction terms,

- Model 2:

$$Y = \alpha_{00} + \alpha_0 Z + \alpha_1 X_1 + \alpha_2 X_2 + \alpha_3 X_3^2 + \alpha_4 X_4^2 + \alpha_5 lnX_5 + \alpha_6 \sqrt{X_6} + \alpha_7 X_7 X_8 + \alpha_8 X_3 X_7 + \varepsilon,$$

where $\varepsilon \sim N(0,1)$. These two models allow us to evaluate treatment effect estimation for simple and complex data. The true treatment effect α_0 and other

coefficients in Models 1 and 2 are defined in Table 3.2. The coefficients for the outcome models were set to simulate strong and weak covariate effects, while still keeping the treatment effect strongest.

Table 3.2. True Treatment Effect (α_0) and Other Outcome Model Coefficients (α_{00}, $\alpha_1, \ldots, \alpha_8$)

α_{00}	α_0	α_1	α_2	α_3	α_4	α_5	α_6	α_7	α_8
0.5	1.5	0.5	0.3	0.7	0.6	0.1	-1.2	-0.5	-1.0

The simulation study consists of eight covariates and ten models (5 treatment selection models each with 2 outcome models). The simulation was ran twice: each of the 10 models has 100 data sets simulated with first a sample size of $n = 500$ and second with a larger sample size of $n = 1000$.

3.5.2 Measures of Interest

The simulation study focuses on minimizing three measures of interest: standardized absolute mean difference as a percentage (SMD), for matching only, bias, and mean squared error (MSE).

The first step in evaluating a "good match" from propensity score matching is the balance correction for covariates between the treatment and control groups. SMD gives us a percentage difference between the treatment and control group covariates after the groups have been matched. SMD is calculated as

$$SMD = \frac{|\overline{X}_{ctrl} - \overline{X}_{trt}|}{\sqrt{\frac{s^2_{ctrl} + s^2_{trt}}{2}}} * 100$$

where $\overline{X}$ is the sample mean and s^2 is the sample variance. Bias is calculated by,

$$bias = \widehat{\alpha_0} - \alpha_0$$

where $\widehat{\alpha_0}$ is the estimated treatment effect and α_0 is the true treatment effect. The primary performance measure to evaluate different matching and weighting techniques will be MSE because it takes into account bias,

$$MSE = SE^2 + bias^2 \text{ or } \sum_{i=1}^{n} \frac{(\widehat{\alpha_0} - \alpha_0)^2}{N}$$

where n and N are the sample size and number of iterations, respectively, and standard error is the standard deviation of $\widehat{\alpha_0}$.

3.5.3 Simulation Results

Simulation Sample Size: n = 500

Propensity scores were predicted from logistic regression, random forest, and an ensemble learner for the true propensity score models A-E, and from these predictions a mean squared error (MSE) was computed. Table 3.3 shows that the ensemble learner performs better than random forest and logistic regression for every treatment selection model. Ensemble learning propensity score prediction performance, especially for more complex models, support the idea of using an ensemble learner over random forest or logistic regression for propensity matching and weighting techniques.

Table 3.3. Logistic Regression, Random Forest, and Ensemble Learner Propensity Score MSE for Models A-E ($n = 500$)

	A	B	C	D	E
LR	0.041	0.052	0.054	0.048	0.047
RF	0.039	0.048	0.049	0.044	0.044
EL	0.018	0.017	0.017	0.017	0.018

As mentioned above, we use SMD to measure balance between treatment and control group covariates after matching. Tables 3.4 and 3.5 show the SMD before matching, after matching using logistic regression, random forest, and ensemble learning for models A-E. Logistic regression, random forest, and ensemble learning matching reduced the SMD for all five treatment selection models, compared to before matching. Austin et al. (2007) notes that a standardized mean difference greater than 10% suggests an imbalance in a covariate. Balance is achieved by matching with all three propensity score estimation models. Overall, SMD largely decreased from before to after matching with logistic regression, random forest, and the ensemble learner, meaning covariate balance did improve between treatment and control groups.

Table 3.4. Model A, B, and C: Covariate Balance Percentages ($n = 500$)

	Model A				Model B				Model C			
Covariate	Before	LR	RF	EL	Before	LR	RF	EL	Before	LR	RF	EL
x1	21.7	5.4	6.4	6.1	22.7	6.3	9.0	5.8	23.4	5.6	8.5	6.1
x2	22.2	4.9	5.6	6.7	25.0	6.9	7.1	6.8	24.5	4.8	6.4	6.0
x3	25.5	5.2	6.4	6.0	22.7	5.3	6.9	7.0	25.1	6.4	6.4	5.4
x4	16.2	5.6	6.7	5.9	18.4	6.1	5.2	4.8	19.6	5.9	6.4	5.7
x5	18.7	4.7	6.9	5.7	11.0	5.7	5.6	5.3	29.2	6.6	6.1	6.6
x6	12.8	5.4	6.1	5.5	13.6	5.9	5.9	5.8	31.4	6.6	6.6	6.5
x7	17.8	4.5	6.2	5.4	47.1	8.1	9.3	8.0	19.9	5.4	6.3	6.4
x8	7.8	5.0	5.5	4.6	8.1	5.1	5.8	5.3	9.3	5.0	5.5	5.6
mean	17.8	5.1	6.2	5.7	21.1	6.2	6.9	6.1	22.8	5.8	6.5	6.0

Table 3.6 outlines the measures of interest (bias and MSE) for matching, inverse probability of treatment weighting, variance stabilization, and weight truncation for logistic regression, random forest, and ensemble learning propensity

Table 3.5. Model D and E: Covariate Balance Percentages ($n = 500$)

	Model D				Model E			
Covariate	Before	LR	RF	EL	Before	LR	RF	EL
x1	22.7	5.7	8.3	6.6	22.0	5.3	8.8	7.3
x2	26.7	6.0	6.9	7.0	25.5	5.6	7.0	6.0
x3	26.4	5.9	6.9	6.3	38.2	7.6	7.4	6.8
x4	10.0	5.4	5.7	5.9	9.5	5.0	5.5	5.6
x5	11.1	6.1	6.6	5.6	14.9	5.8	5.5	5.0
x6	19.1	6.7	6.4	6.5	17.7	4.7	5.4	5.4
x7	38.0	6.5	9.1	6.7	13.6	5.6	5.8	4.8
x8	9.7	5.6	6.1	6.0	11.8	5.6	5.9	5.7
mean	20.5	6	7	6.3	17.9	5.6	6.4	5.8

score estimation. These results offer comparisons between models, logistic regression, random forest, and ensemble learning, and between matching and weighting.

We look first at the performance, MSE, of propensity score estimation models within matching and weighting techniques. The ensemble learner outperforms, smallest MSE, logistic regression and random forest the majority of the time for matching, IPTW, and variance stabilization, and always outperforms logistic regression and random forest for weighting with truncation and variance stabilization with truncation. Now looking across all matching and weighting techniques, the ensemble learner boasts the lowest MSE for every treatment selection model A-E and for both outcomes. As we implement matching and different weighting techniques, we see in Table 3.6 that the lowest MSE for treatment effect estimation consistently comes when utilizing weighting with truncation or variance stabilization with truncation.

Table 3.6. Treatment Effect Estimation Performance Measures For All Models A-E and Both Outcomes 1 & 2* ($n = 500$)

		Matching			IPTW			Var. Stab.			Weight Trunc.			Var. Stab. Trunc.		
		LR	RF	EL	LR	RF	EL	LR	RF	EL	LR	RF	EL	LR	RF	EL
A	BIAS	-0.016	-0.021	0.005	-0.028	-0.017	-0.004	-0.027	-0.015	0.003	-0.03	-0.025	-0.013	-0.029	-0.022	-0.007
	MSE	0.028	0.035	0.02	0.022	0.027	0.01	0.023	0.027	0.009	0.02	0.021	0.008	0.02	0.026	0.01
B	BIAS	0.012	0.029	0.044	0.02	0.025	-0.025	0.021	0.027	-0.004	0.009	0.015	0.007	0.013	0.026	0.019
	MSE	0.043	0.024	0.027	0.045	0.042	0.051	0.047	0.041	0.042	0.036	0.035	0.023	0.037	0.038	0.02
C	BIAS	-0.049	-0.037	-0.06	-0.041	-0.037	0.002	-0.04	-0.04	0.006	-0.055	-0.052	-0.039	-0.045	-0.048	-0.043
	MSE	0.033	0.057	0.023	0.026	0.026	0.011	0.026	0.028	0.01	0.029	0.03	0.01	0.029	0.028	0.012
D	BIAS	-0.005	0.001	0.028	-0.007	-0.005	-0.02	-0.004	0.001	0.001	-0.005	-0.008	-0.002	-0.009	-0.002	0.01
	MSE	0.016	0.029	0.016	0.045	0.034	0.04	0.047	0.031	0.032	0.028	0.028	0.022	0.028	0.027	0.017
E	BIAS	-0.029	-0.049	-0.02	-0.011	-0.011	-0.016	-0.009	-0.016	-0.033	-0.012	-0.016	-0.015	-0.011	-0.017	-0.031
	MSE	0.029	0.015	0.013	0.01	0.009	0.016	0.01	0.01	0.015	0.01	0.01	0.007	0.01	0.01	0.007
A*	BIAS	0.063	0.049	0.054	0.098	0.075	0.101	0.098	0.069	0.094	0.064	0.068	0.067	0.068	0.065	0.058
	MSE	0.029	0.043	0.019	0.039	0.035	0.032	0.04	0.034	0.03	0.028	0.03	0.017	0.031	0.033	0.016
B*	BIAS	-0.078	-0.093	-0.074	-0.102	-0.095	-0.033	-0.104	-0.092	-0.016	-0.101	-0.095	-0.089	-0.088	-0.085	-0.079
	MSE	0.025	0.019	0.01	0.026	0.018	0.007	0.026	0.015	0.006	0.021	0.018	0.014	0.016	0.014	0.011
C*	BIAS	-0.044	-0.043	-0.044	0.042	0.009	0.031	0.047	0.001	0.019	0.004	0.005	0.021	0.006	-0.003	0.012
	MSE	0.015	0.014	0.019	0.012	0.011	0.01	0.013	0.012	0.011	0.01	0.011	0.006	0.012	0.011	0.007
D*	BIAS	0.007	0.014	0.025	0.021	0.011	-0.033	0.022	0.013	-0.041	0.014	0.014	0.019	0.01	0.016	0.024
	MSE	0.012	0.016	0.008	0.019	0.014	0.04	0.02	0.013	0.042	0.013	0.012	0.008	0.012	0.011	0.007
E*	BIAS	0.055	0.048	0.043	0.034	0.035	0.08	0.033	0.038	0.082	0.034	0.039	0.036	0.038	0.037	0.041
	MSE	0.009	0.015	0.006	0.009	0.011	0.017	0.008	0.01	0.018	0.009	0.01	0.005	0.008	0.009	0.005

Simulation Sample Size: n = 1000

As we increase the sample size from 500 to 1000, logistic regression, random forest, and the ensemble learner predict true propensity scores for Models A-E with smaller MSE compared to the sample size of 500. Table 3.7 shows that the ensemble learner outperforms logistic regression and random forest for all Models A-E for predicting true propensity score as the outcome.

Table 3.7. Logistic Regression, Random Forest, and Ensemble Learner Propensity Score MSE for Models A-E ($n = 1000$)

	A	B	C	D	E
LR	0.039	0.049	0.052	0.045	0.045
RF	0.041	0.049	0.05	0.045	0.047
EL	0.017	0.016	0.015	0.015	0.017

The covariate balance after matching for logistic regression, random forest, and the ensemble learner shows a good balance with all SMD below a 10% difference between treatment and control groups, for all models A-E in Table 3.8 and 3.9. Again we see a large decrease in SMD from before to after matching. The largest decrease in SMD for most individual covariates comes from matching using propensity scores estimated by the ensemble learner.

Table 3.8. Model A, B, and C: Covariate Balance Percentages ($n = 1000$)

	Model A				Model B				Model C			
Covariate	Before	LR	RF	EL	Before	LR	RF	EL	Before	LR	RF	EL
x1	23.5	3.9	5.0	4.7	21.4	4.4	7.2	4.3	21.3	4.3	7.9	4.8
x2	22.4	3.5	4.8	6.3	25.4	3.7	6.7	6.4	25.1	3.8	6.4	5.6
x3	26.3	3.8	6.1	4.9	20.1	4.0	4.6	4.2	24.5	4.9	5.5	4.6
x4	15.2	3.5	4.1	4.6	18.2	4.6	5.4	4.2	20.8	4.5	5.3	4.8
x5	17.2	3.4	6.6	4.0	9.1	3.9	4.2	4.0	28.2	4.8	4.8	4.7
x6	11.1	3.8	5.0	3.9	10.1	3.4	4.5	4.6	32.3	5.3	5.7	5.1
x7	16.9	3.9	5.1	4.0	44.7	6.7	7.2	5.8	21.7	4.8	4.5	5.0
x8	7.9	3.8	4.4	3.8	7.5	3.4	3.9	4.7	7.0	3.8	3.7	3.7
mean	17.5	3.7	5.1	4.5	19.6	4.2	5.5	4.8	22.6	4.5	5.5	4.8

Table 3.9. Model D and E: Covariate Balance Percentages ($n = 1000$)

	Model D				Model E			
Covariate	Before	LR	RF	EL	Before	LR	RF	EL
x1	22.6	3.9	7.6	4.6	22.3	4.6	7.1	5.0
x2	25.2	3.8	7.5	7.5	26.3	4.1	8.5	6.9
x3	25.9	4.4	5.5	4.9	37.7	5.5	7.3	5.5
x4	8.9	3.9	4.1	3.9	7.9	3.5	3.6	4.2
x5	10.4	4.1	4.2	3.7	15.3	4.5	4.6	4.4
x6	17.3	4.7	4.2	4.2	4.9	4.0	4.1	4.0
x7	37.7	5.8	7.1	5.7	12.4	3.9	4.3	4.2
x8	8.2	4.3	4.0	4.0	9.1	4.1	3.6	4.0
mean	19.5	4.4	5.5	4.8	17.0	4.3	5.4	4.8

Table 3.10. Treatment Effect Estimation Performance Measures For All Models A-E and Both Outcomes 1 & 2* ($n = 1000$)

		Matching			IPTW			Var. Stab.			Weight Trunc.			Var. Stab. Trunc.		
		LR	RF	EL	LR	RF	EL	LR	RF	EL	LR	RF	EL	LR	RF	EL
A	BIAS	-0.012	-0.028	-0.033	-0.014	-0.005	-0.043	-0.014	-0.006	-0.048	-0.006	-0.001	-0.001	-0.009	-0.006	-0.005
	MSE	0.017	0.022	0.012	0.013	0.02	0.015	0.013	0.02	0.015	0.014	0.015	0.006	0.014	0.016	0.008
B	BIAS	0.039	0.02	0.017	0.025	0.017	0.038	0.023	0.016	0.027	0.033	0.033	0.037	0.036	0.028	0.032
	MSE	0.01	0.008	0.006	0.013	0.014	0.021	0.013	0.014	0.019	0.01	0.012	0.007	0.011	0.01	0.006
C	BIAS	0.022	0.001	0.02	0.015	0.009	0.046	0.015	0.01	0.056	0.024	0.021	0.016	0.023	0.021	0.024
	MSE	0.028	0.029	0.012	0.011	0.014	0.008	0.012	0.014	0.008	0.015	0.013	0.005	0.014	0.015	0.007
D	BIAS	0.02	0.005	0.018	0.017	-0.001	0.047	0.016	-0.002	0.026	0.01	0.012	0.02	0.012	0.012	0.012
	MSE	0.008	0.012	0.005	0.004	0.008	0.026	0.004	0.008	0.021	0.004	0.004	0.004	0.005	0.003	0.003
E	BIAS	0.021	0.019	0.015	0.03	0.016	0.046	0.031	0.016	0.039	0.032	0.035	0.038	0.034	0.032	0.034
	MSE	0.014	0.02	0.009	0.013	0.012	0.011	0.013	0.011	0.011	0.011	0.011	0.008	0.012	0.012	0.007
A*	BIAS	-0.006	-0.031	-0.033	-0.024	-0.031	0.009	-0.025	-0.032	0.01	-0.03	-0.028	-0.018	-0.024	-0.029	-0.021
	MSE	0.01	0.011	0.008	0.006	0.007	0.004	0.006	0.007	0.004	0.006	0.005	0.004	0.005	0.005	0.003
B*	BIAS	0.026	0.032	0.04	-0.003	0.006	-0.013	-0.003	0.011	-0.002	0.024	0.02	0.02	0.024	0.028	0.035
	MSE	0.037	0.042	0.023	0.033	0.044	0.03	0.033	0.043	0.027	0.029	0.028	0.016	0.029	0.029	0.016
C*	BIAS	0.02	-0.018	0.003	-0.005	-0.033	-0.053	-0.004	-0.033	-0.044	-0.02	-0.021	-0.013	-0.017	-0.015	-0.01
	MSE	0.014	0.016	0.009	0.017	0.022	0.011	0.017	0.021	0.009	0.01	0.011	0.007	0.011	0.011	0.005
D*	BIAS	-0.007	-0.021	-0.016	-0.013	-0.028	-0.017	-0.013	-0.026	-0.019	-0.012	-0.017	-0.025	-0.013	-0.012	-0.021
	MSE	0.023	0.023	0.016	0.012	0.023	0.025	0.012	0.023	0.029	0.018	0.016	0.01	0.019	0.018	0.01
E*	BIAS	0.035	0.015	0.028	0.001	-0.025	-0.016	0.001	-0.022	-0.004	0.002	-0.004	-0.016	0.012	0.003	-0.003
	MSE	0.011	0.009	0.005	0.01	0.015	0.008	0.009	0.015	0.007	0.009	0.01	0.005	0.01	0.01	0.004

From Table 3.10 we can see that using truncation, whether truncating weights or variance stabilization truncation, results in estimated treatment effects with the lowest MSE. For estimation models the ensemble learner out performs logistic regression and random forest for the majority of treatment effect estimation. The ensemble learner outperforms the other two propensity score estimation models for all treatment selections models A-E and for both types outcome models, simple linear and a more complex model. With the larger sample size of n=1000 we do see an overall decrease in MSE for treatment effect estimation across logistic regression, random forest, and the ensemble learner. We see that as the sample size becomes larger the distinction between propensity score estimation models, between matching or weighting, and whether to truncate or to not.

3.5.4 Recommendations From Simulation

The simulation has shown that using the ensemble learner yields the best estimates for propensity scores, with the ensemble learner having the lowest MSE compared to logistic regression and random forest, as shown in both Tables 3.3 and 3.7. The ensemble learner consistently had the lowest MSEs across matching and weighting methods. Comparing matching, inverse probability of treatment weights, variance stabilization weighting, and truncation for both inverse probability of treatment weights and variance stabilization weighting we see that using truncation with weighting produces the best treatment effect estimates with the lowest MSE. With these results from the simulation study we recommend the use of an ensemble learner in conjunction with variance stabilization weighting with truncation.

CHAPTER 4

An Ensemble Learner for Propensity Score Matching and Weighting Techniques

4.1 Motivation

A main obstacle in learning analytics is the development of algorithms and models that come from researchers with learning analytics backgrounds (Gasevic et al., 2014). With this in mind, as data scientists with backgrounds in institutional research and learning analytics, we take on the task of developing analytic methods for institutional researcher. In our development of such analytic methods we take into consideration that our peers in institutional research have varying backgrounds and expertise. We aim to make our methods accessible and able to be implemented by those willing to expand their analytics toolkit.

EDUCAUSE in 2019, as part of their "Student Genome Project", argues that student success initiatives and data-driven decision-making are two of the top priorities to focus on in higher education. We focus our contribution to learning analytics and educational data mining on student success studies, and the evaluation and validation of student success interventions. Student success studies and interventions are a vital part of higher education, especially with the ever changing learning environments that continues to evolve in pandemic era conditions.

In this chapter, we develop a R-package `matchED` to wrap the different matching and weighting techniques, presented in Chapter 3, into a user friendly software geared towards institutional researchers and higher education researchers for use in student success studies. The package will give the researcher the opportunity to try several ways to match or weight their data, and select the underlying model for propensity score estimation. To illustrate the methods and `matchED` R-package, we analyze a real world student success intervention aimed at underprivileged communities. From the results of the simulation study, in Chapter 3, parameter input recommendations to achieve best model and treatment effect estimation performance will be used to analyze the study. We exposit this chapter as a pseudo-manual that explains the functions and parameters available within the package. For ease of use and to further the accessibility of the package, we present a package tutorial. This tutorial provides users the ability to walk-through a student success application of the package to better understand its workings.

4.2 MatchED: Functions and Capabilities

4.2.1 pscorED: Estimating Propensity Scores

The `pscorED` function in the `MatchED` package estimates propensity scores using logistic regression, random forest, or an ensemble learner chosen by the user. A propensity score is the probability of treatment assignment given a subject baseline characteristics (Rosenbaum and Rubin, 1983). A subject's probability of treatment should be greater than 0, and users should include as many observed characteristics as possible to help improve model accuracy.

- `formula`: A formula describing the model to be fit, where the response is a binary treatment indicator and the predictors are pre-treatment baseline characteristics.
- `trt`: A binary treatment indicator of whether a subject was treated or not (should be the response in formula).
- `data`: A data frame that contains the variables in the model called by formula.
- `model`: The model to estimate propensity scores with. "lr" (Logistic Regression), "rf" (Random Forest), "el" (Ensemble Learner) are the current model options.
- `ensemble`: The list of models to use as base learners to build the ensemble learner. Base learners: "lr" (Logistic Regression), "rf" (Random Forest), "bt" (Boosting), "bg" (Bagging), "svm" (Support Vector Machine),"nb" (Naive-Bayes), "knn" (K-Nearest Neighbor), and "nn" (Neural Network) are current base learners available.
- `kfold`: The number of folds for cross-validation. i.e., if `kfold` is 10, 10-fold cross-validation will be used.

The user has many options to modify the ensemble learner by choosing different subsets of base learners, and computational speed can be adjusted by increasing or decreasing the number of k-folds used during cross-validation. The function outputs a vector of propensity scores, size n, where n is the number of subjects in the data provided by the user.

4.2.2 matchED: Propensity Score Matching

The `matchED` function in the `MatchED` package is used to match data using the `MatchIt` package (Ho et al., 2007) and propensity scores found using the function `pscorED` to create a matched set. This pre-processing balances baseline characteristics between the treated and control groups, mimicking a

pseudo-randomization of treatment assignment and allowing for causal inference from further analysis (Rosenbaum and Rubin, 1983).

- `formula`: A formula describing the model to be fit, where the response is a binary treatment indicator and the predictors are pre-treatment baseline characteristics. The formula must be fully typed out even if all variables are to be used from the data; i.e., "$\sim$." will not work.
- `data`: A data frame that contains the variables in the model called by formula.
- `ps`: A vector of propensity scores from the function `pscorED`.

Although the function asks for propensity scores from `pscorED`, users can provide their own propensity scores given the vector they provide is the same length as the data provided and the values within the vector are between 0 and 1. The `matchED` function outputs a data set with each treated subject matched to a control subject by the nearest propensity score. Every treated subject will be matched with a single control subject. `matchED` prints a balance check between the treated and control groups for the baseline characteristics, so the user can verify balance was achieved through propensity score matching.

4.2.3 weightED: Weighting Propensity Scores

The `weightED` function in the `MatchED` package calculates inverse probability of treatment weights, variance stabilization weights, or truncated weights to be utilized for treatment effect estimation.

- `trt`: A binary treatment indicator of whether a subject was treated or not.

- `trt.value`: The value used to represent the treated group within the `trt` variable. For instance "1" indicates treated and "0" indicates not treated in a binary 0/1 treatment variable.
- `ps`: A vector of propensity scores from the function `pscorED`.
- `method`: The type of weights to be calculated. "IPTW" (Inverse Probability of Treatment Weighting), "Var Stab" (Variance Stabilization Weighting) are currently available.
- `truncate`: A logical argument that when TRUE will truncate the weights specified by `method`, using a user specified threshold.
- `threshold`: A two value vector, probabilities between $[0, 1]$, with the lower bound and upper bound for the threshold to be used to truncate.

The function outputs a vector of weighted propensity scores, size n, where n is the number of subjects in the data provided by the user.

4.2.4 treatED: Estimating Treatment Effects Using Propensity Scores

The `treatED` function in the `MatchED` package estimates treatment effects using linear regression or logistic regression and provides the option to use propensity scores as weights to better the estimation.

- `y`: A vector of response data. If a factor, logistic regression will be used for classification, otherwise linear regression will be used.
- `x`: A data frame of predictors.
- `trt`: A binary treatment indicator of whether a subject was treated or not.
- `weighting`: A logical argument, if set to TRUE will perform weighted linear regression using provided weights. When FALSE, the default, normal ordinary least squares linear regression will estimate the treatment effects.
- `weights`: A vector of weights from `weightED` to be used for weighted linear regression

- `level`: The confidence level, in decimal form, to be used for the confidence interval of the treatment effect estimate.

The function outputs a treatment effect used to draw conclusions on the effectiveness of the treatment. Note that if `y` is a factor then the treatment effect will be an odds ratio. The output also includes a p-value and the confidence interval for the treatment effect.

4.3 MatchED Tutorial

With the `MatchED` package's functions and parameters listed and defined above, we will guide the package user through a short tutorial that will show each function in action. To start the practitioner should have data that is ready to be analyzed, meaning it has been collected properly, organized, missing data has been addressed (it should be a complete data set), and "cleaned" to ensure analysis results are valid and accurate. To clarify the data must be appropriately formatted to be used by the functions. The `data` should be c by n, where c is the number of predictors plus the treatment variable, and n is the sample size. Both the `trt` and `y` vectors should be length n and should correspond with the rows in `data`, i.e, the first `y` and `trt` value should be the outcome and treatment indicator for the first subject in `data`. For the tutorial we will use a set of simulated data that an institutional researcher might gather for a student success study.

The first step will be to estimate propensity scores, the probability the subject will receive the treatment; for this the researcher will use `pscorED`. As detailed above in Section 4.2.1, the researcher will need to provide a `formula` that describes the model to be fit, where the response is the binary treatment indicator

and the predictors are the pre-treatment baseline characteristics of the subjects in the study. They then must specify the name of the treatment indicator, `trt`, and a data frame that holds the data for the model, `data`. The data frame for `data` should include as many pre-treatment observed characteristics as possible. The researcher has the freedom of choosing logistic regression, random forest, or the ensemble learner as the `model` to estimate propensity scores, and if they choose the ensemble learner they must further list the base learners in `ensemble`. By default all base learners will be used and might cause an increase in computational time depending on the size of the user's data. The user can specify any k-fold cross validation they would like to use, by default 10-fold cross validation is performed.

In the example code below the `formula` is our binary treatment indicator regressed on all other predictors in our data, `trt` is the indicator "Stud_Succ_Interv", `data` set as "tutorial_data", we have chosen to use the ensemble learner as our `model`, all base learners besides logistic regression will be used for the ensemble learner, and the default 10-fold cross validation is performed.

```
pscores <- pscorED(formula = Stud_Succ_Interv ~ .,
                   trt = tutorial_data$Stud_Succ_Interv,
                   data = tutorial_data,
                   model = "el",
                   ensemble = c("rf","bt","bg","nb","knn"),
                   kfold={10})
```

`pscorED` outputs a vector of propensity scores, in the example code stored in the variable `pscores`, that can now be used by `matchED` and `weightED`, depending on the user's choice.

With the propensity scores calculated we look at the function `matchED` to create a set of matched data. The matching process is described in detail in Section 3.3.1. As explained in the `formula` description for `matchED` the formula must be written out containing the binary treatment indicator regressed on all pre-treatment predictors. In the code below the pre-treatment variables are sex, age, high school GPA, an indicator if a student is first generation, SAT score, an indicator if a student is an under-represented minority, an indicator if a student is a STEM major, and the number of college units accumulated during high school. `data` is again specified to be "tutorial_data", and we insert `pscores`, created above, as the propensity scores, `ps`, to be used to match the data.

```
matched_data <- matchED(Stud_Succ_Interv ~ Sex + Age + HS_GPA +
                                           First_Generation + SAT +
                                           URM + STEM + College_Units,
                        data = tutorial_data,
                        ps = pscores)
```

`matchED` will store a data set of matched subjects, in this case in "matched_data", and will automatically output a summary of balance for all data, balance for matched data, the percent balance improvement, and the sample sizes for the control and treated groups. This output allows the user to assess the balance between the treated and control groups before and after matching. The variable means between control and treated groups should have smaller differences after matching (Rosenbaum and Rubin, 1983). This output details, in the "Sample sizes" section, that all 109 treated subjects were matched and 282 control subjects

were not matched, therefore the size of the matched data set will be 109 treated subjects matched to 109 control subjects for a total of 218 subjects.

```
Summary of balance for all data:
                 Means Treated Means Control SD Control Mean Diff eQQ Med eQQ Mean eQQ Max
distance                0.7176        0.0859     0.0422    0.6317   0.634   0.6316   0.642
SexFemale               0.6239        0.6113     0.4881    0.0126   0.000   0.0183   1.000
SexMale                 0.3761        0.3887     0.4881   -0.0126   0.000   0.0092   1.000
Age                    18.5321       18.5038     1.4480    0.0283   0.000   0.2110   2.000
HS_GPA                  3.7089        3.7065     0.3289    0.0024   0.030   0.0332   0.320
First_Generation        0.1927        0.1407     0.3481    0.0520   0.000   0.0550   1.000
SAT                  1217.8899     1208.5166   151.8766    9.3733  10.000  17.7982 110.000
URM0                    0.6147        0.6598     0.4744   -0.0452   0.000   0.0459   1.000
URM1                    0.3853        0.3402     0.4744    0.0452   0.000   0.0459   1.000
STEM0                   0.6055        0.7391     0.4397   -0.1336   0.000   0.1284   1.000
STEM1                   0.3945        0.2609     0.4397    0.1336   0.000   0.1376   1.000
College_Units           9.6128        8.1284     8.1384    1.4845   1.600   1.5083   4.000


Summary of balance for matched data:
                 Means Treated Means Control SD Control Mean Diff eQQ Med eQQ Mean eQQ Max
distance                0.7176        0.1389     0.0341    0.5786   0.586   0.5786    0.60
SexFemale               0.6239        0.5872     0.4946    0.0367   0.000   0.0367    1.00
SexMale                 0.3761        0.4128     0.4946   -0.0367   0.000   0.0367    1.00
Age                    18.5321       18.5413     1.5959   -0.0092   0.000   0.2110    2.00
HS_GPA                  3.7089        3.6829     0.3675    0.0260   0.030   0.0537    0.28
First_Generation        0.1927        0.1468     0.3555    0.0459   0.000   0.0459    1.00
SAT                  1217.8899     1201.8349   142.1278   16.0550  10.000  22.6606  110.00
URM0                    0.6147        0.6606     0.4757   -0.0459   0.000   0.0459    1.00
URM1                    0.3853        0.3394     0.4757    0.0459   0.000   0.0459    1.00
STEM0                   0.6055        0.7431     0.4389   -0.1376   0.000   0.1376    1.00
STEM1                   0.3945        0.2569     0.4389    0.1376   0.000   0.1376    1.00
College_Units           9.6128        7.7394     8.9113    1.8734   2.500   2.2606    6.00

Percent Balance Improvement:
                 Mean Diff. eQQ Med  eQQ Mean  eQQ Max
distance              8.3996   7.571    8.3895   6.5421
SexFemale          -191.2477   0.000 -100.0000   0.0000
SexMale            -191.2477   0.000 -300.0000   0.0000
Age                  67.5519   0.000    0.0000   0.0000
HS_GPA            -1003.9908   0.000  -61.6022  12.5000
First_Generation     11.7780   0.000   16.6667   0.0000
SAT                 -71.2852   0.000  -27.3196   0.0000
URM0                 -1.5584   0.000    0.0000   0.0000
URM1                 -1.5584   0.000    0.0000   0.0000
STEM0                -2.9851   0.000   -7.1429   0.0000
STEM1                -2.9851   0.000    0.0000   0.0000
College_Units       -26.2008 -56.250  -49.8783 -50.0000

Sample sizes:
          Control Treated
All           391     109
Matched       109     109
Unmatched     282       0
Discarded       0       0
```

As a parallel to using `matchED` the researcher might want to calculate propensity score weights using `weightED`. In the code below the `trt.value` is the

value within the vector that defines if a subject is treated, in our case the value is '1'. The user can choose either inverse probability of treatment weighting or variance stabilization weighting with the `method` parameter, and can choose whether the weights should be truncated, using the `truncate` parameter. If `truncate` is set to "TRUE" the user specifies their own `threshold` or uses the default, $(0.1, 0.9)$, corresponding to the $10th$ and $90th$ percentile. The calculation of both types of weights and truncation are described in detail in Sections 3.4-3.4.3.

```
weighting <- weightED(trt = tutorial_data$Stud_Succ_Interv,
                      trt.value='1',
                      ps = pscores,
                      method = "Var Stab",
                      truncate = TRUE,
                      threshold = c(0.1,0.9))
```

We now have a matched set of data, "matched_data", and a set of weights, "weighting", that can be used to estimate a treatment effect. For this example we look at the treatment effect the student success intervention has on students' semester GPA, labeled "Term_GPA". Whether the user decided on propensity score matching or weighting `treatED` is used for both to estimate treatment effects. For both cases the response variable, `y`, will be "Term_GPA". The other parameters differ slightly for propensity score matching and weighting.

We look first at the case where propensity score matching was done. We specify the predictors, `x`, as "matched_data", the `trt` is the binary treatment indicator column in "matched_data", `weighting` will be "FALSE", and we will specify a confidence level, `level`.

We then look at the case where propensity score weighting was done. We specify the predictors, `x`, as "tutorial_data", the `trt` is the binary treatment indicator column in "tutorial_data", `weighting` will be "TRUE", we will then input the weights, `weighting`, and again specify a confidence level, `level`.

```
TE_Matching <- treatED(y = Term_GPA,
                       x = matched_data,
                       trt = matched_data$Stud_Succ_Interv,
                       weighting = FALSE,
                       level = 0.95)

TE_Weighting <- treatED(y = Term_GPA,
                        x = tutorial_data,
                        trt = tutorial_data$Stud_Succ_Interv,
                        weighting = TRUE,
                        weights = weighting,
                        level = 0.95)
```

`treatED` outputs a treatment effect coefficient, p-value for the coefficient, and a confidence interval for the coefficient. If the response variable used in `treatED` is continuous, as is in the example above, then the coefficient can be interpreted as an increase/decrease in the response for those who participated in the student success intervention. On the other hand if the response variable is a binary indicator then the coefficient will be interpreted as an odds ratio for those that participated in the student success intervention. Looking at the example output below the treatment effect coefficient is 0.4 with a p-value of 0.002 and a 95% confidence interval of 0.145, 0.665. We say that those students that participated in the student success intervention had on average a significant increase of 0.4 in their semester GPA compared to those students that did not participate in the intervention, accounting for all other possible factors.

```
  Estimate    Pr(>|t|)      2.5 %      97.5 %
0.404800988 0.002352813 0.144687874 0.664914102
```

`R` by default outputs a large number of decimals, we leave it to the user whether they want to round the output. The `round` function provides an easy way to round all output of `treatED` to the desired decimal place.

4.4 Student Success Case Study

From the California State University fact book, Timothy White, the Chancellor of the California State University system, states "The California State University is key to California's brightest and most hopeful future, opening the door to educational opportunities for all and transforming the lives of students and their families." Keeping this in mind, universities can offer student success interventions such as supplemental instruction discussed in Chapter 2, which are offered while the student is currently enrolled, or even pre-enrollment interventions that aim to help students succeed when the opportunity is not normally available. We look at one such student success intervention that is aimed for students who grow up in underprivileged communities and are given a path that optimally could afford them the opportunity for entrance into higher education.

Table 4.1 presents a comparison of students supported by the student success intervention and their peers not assisted by the intervention. This snapshot of a few student background characteristics shows that those in the intervention have a higher rate of being first generation college students and under-represented minorities. They have lower mean SAT scores, slightly lower mean high school

Table 4.1. Summary of Student Characteristics for Treatment and Control Groups. Mean & (Standard Deviation) Reported for Continuous Variables, and Percentage Reported for Categorical Variables.

	Control	Treated
First Generation	15.8%	21.0%
Under-Represented Minority	32.4%	62.9%
SAT Score	1209.1 (156.3)	1145.3 (85.1)
High School GPA	3.7 (0.3)	3.5 (0.3)
Transfer Units	22.3 (26.3)	10.8 (11.2)

GPA, and tend to earn, and transfer, fewer college level course units than their peers at the university.

Table 4.2 looks at the balance before and after matching, between treated and control groups. Before propensity score matching the standardized mean difference are large for Hispanic students, under-represented minorities, incoming units, first generation students, and SAT score between control and treated. After propensity score matching the standardized mean difference decreases for all covariates. Matching in this study does balance some characteristics of the treated and control groups below the recommended standardized mean difference of 10.

Table 4.2. Standardized Mean Difference Before and After Matching

Covariate	Before	After
Age	62.8	6.7
Gender	10.4	3.2
Eligibility Index	35.9	25.8
SAT Score	50.8	7.3
High School GPA	32.8	18.4
Incoming Units	57.1	8.3
First Generation	13.3	2.6
Under-represented Minority	64.1	7.7
Hispanic	72.2	15.2

To evaluate the success of the intervention we look at the effect the intervention has on students grade point average (GPA) at end of their second semester at the university, specifically the GPA for courses taken on campus. Table 4.3 has the student success intervention coefficient, aka the treatment effect, and the associated p-value and 95% confidence interval. Looking at the results using propensity score variance stabilization weighting with truncation, those students that participated in the student success intervention had on average a non-significant increase, at the 0.05 level, of 0.053 in their end of second semester GPA compared to those students that did not participate in the intervention, accounting for all other possible factors.

Table 4.3. Student Success Treatment Effect, P-value, and 95% Confidence Interval From Matching and Variance Stabilization With Truncation

Method	Treatment Effect	P-value	95% C.I.
Matching	0.063	0.196	-0.033, 0.158
Var. Stab. Trunc.	0.053	0.179	-0.024, 0.130

Normally the results presented in Table 4.3 would be glossed over as non-significant and a result of a failed student success intervention. However, since students provided with the intervention are perceived to be at a disadvantage due to their socioeconomic background, an "on par" result or no significant difference between students in the intervention and those who are not is a success. This results show that by participating in the student success intervention they were able to match their peers in GPA at the end of their second semester.

4.5 Discussion

With the importance of student success studies and relevant learning analytics methods in higher education `matchED` gives institutional researchers a set of machine learning tools to accurately estimate the effect of their student success intervention. `matchED` wraps propensity score matching and weighting techniques into a user friendly software geared towards higher education researchers for use in student success studies. The packages gives the researcher the opportunity to try several ways to match or weight their data, and select the underlying model for propensity score estimation. The package provides a set of default parameters based on our evaluations in the previous Chapters. Nonetheless, the experienced user has the flexibility to tune these parameters based on their specific data on hand. Propensity scores can be estimated by logistic regression, random forest, or an ensemble learner that is trained with user chosen base learners. The user can choose to use propensity score matching or weighting techniques, and choose whether to truncate their weights for treatment effect estimation. We recommend using the ensemble learner to estimate propensity scores and to weight using variance stabilization with truncation to estimate the treatment effect based on results from our simulation studies.

The package manual explains the functions and parameters, and the tutorial guides researchers through a simulated student success study analysis. We present example code and output that offers researchers a look at how `matchED` should be utilized.

Lastly we look at an actual student success study and analyze its effect on students campus grade point average (GPA) at end of their second semester at the university. We present student characteristics between treated and control groups to show the differences in student characteristics. Using `matchED`, specifying the use of propensity score variance stabilization weighting with truncation, we find that students in the intervention have an on par second semester GPA compared to their student peers. We explain that these findings show a successful student success intervention, with the intervention being able to bring participants to the same level of success as their peers at the university.

CHAPTER 5

Conclusion and Discussion

5.1 Results

We develop an early version of our ensemble learner to predict individualized treatment effects to assess student success studies of intervention strategies. The ensemble learner had the best accuracy when predicting a non-repeatable grade in the course, out-performing all individual base learners used to create the ensemble learner. Furthermore, when looking at predicting final exam score and course grade the ensemble learner maintained a lower root mean squared error, out-performing the base learners again. Using the ensemble learner to predict individualized treatment effects we show that enrolling in a supplemental instruction course moderately improves the final exam score and leads to an increase of almost a half a grade point in final course grade, on average. We offer that predicting individualized treatment effects for students enrolling in a course, in future terms, can provide an indicator for students that could be at-risk for failing.

We construct an ensemble learner for estimating propensity scores. We introduce a simulation study to evaluate the ensemble learner's performance when estimating propensity scores and compare propensity score matching and weighting techniques to best estimate treatment effects for student success studies. Our simulation study compared logistic regression, random forest, and the ensemble

learner using treatment selection models that are varying in complexity and with a sample size of 500 and 1000. Tables 3.3 and 3.7 show that the ensemble learner best estimated propensity scores, out-performing both logistic regression and random forest for all treatment selection models. When comparing matching and weighting techniques, propensity score weighting with truncation performed better than matching for treatment effect estimation, the majority of the time. Diving further, weighting using variance stabilization with truncation out-performed other weighting techniques for estimating treatment effects.

Using the results from our simulation study and the validation for our ensemble learner we built an accessible and user friendly R-package, `matchED`. `matchED` gives researchers a set of machine learning tools to accurately estimate propensity scores and treatment effects to efficiently and best analyze their own student success studies. The package has the capability to estimate propensity scores, do propensity score matching, calculate propensity score weighting, and estimate treatment effects. We present a package manual that explains the functions and parameters available to the user and a tutorial that guides users through the use of each function, with example code. We use `matchED` to analyze a current student success study, that provides a higher education opportunity to students in under-privileged communities. We used the ensemble learner to estimate the propensity scores and variance stabilization with truncation to estimate the effect the intervention had on student's campus grade point average (GPA) at end of their second semester. We find that students in the intervention

have "on-par" grade point averages, at the end of their second semester, with their peers outside the intervention.

5.2 Challenges

We have shown that an ensemble learner is a strong predictive modeling tool, out-performing all base learners to estimate ITEs and out-performing logistic regreession and random forest for estimating propensity scores. That said, given a library of base learners, ensemble learning approaches can be made computationally efficient for producing predictions. Furthermore, as mentioned in Section 2.3.1, applications displaying less correlation between the base learner predictions, and perhaps larger sample size given the cross-validation steps required, will realize stronger ensemble predictions. In this sense, our application may provide a level of worst-case scenario for ensemble learning ITE and propensity score estimates in education analytics.

The non-randomized treatment assignment in observational studies may lend to selection bias from imbalance between treatment and control subjects relative to an unobserved confounder. If this important confounder is not collected or excluded from modeling, treatment effects will thus not be sufficiently adjusted. Treatment randomization overcomes this challenge by balancing subjects with respect to all variables/characteristics except the treatment assignment. However randomized controlled trials are often not an option in education studies. Model-based adjustments of confounders, as performed by the base learners in this paper, adjust treatment effects for covariates. Ensemble learning approaches, by combining predictions over a set of single learners, may improve predictive

performance (Poliker, 2006) so that the confounder adjustments are potentially less model-dependent. In a situation where a randomized trial is not an option, no approach can adjust for important, unobserved confounders. This emphasizes the importance of study design in observational studies and pursuit of an approach like ours that is less model-dependent.

In our studies, the inputs to the model consisted of all data available in the SDSU student information database. These variables encapsulate student demographics, educational background, academic (particularly mathematics) preparation, student performance metrics, and SDSU program involvement. Though we believe this set of covariates captures the primary suite of confounders, our study does not include direct measures of student attitudes towards statistics (the course topic under study), social and academic behavior, nor student motivation. Such measures would need be self-reported, that is, collected through standardized survey instruments. These student characteristics may be unobserved confounders that may perhaps bias our idividualized treatment effect or overall treatment effect estimates.

Ensemble learning methods run the risk of trading off interpretability for predictive performance. In many applications, an interpretable machine learning framework is critical to practical use. That said, the flexibility in choice of base learner and meta-learner allows the user to potentially strike a desired balance (see e.g., Otte, 2013).

5.3 Recommendations For Further Study

In our applications, we selected a specific suite of base learners to combine for the ensemble prediction. But of course at that stage of the algorithm the meta-learner needs know only the number of base learners and predictions from each learner. The choice and number of base learners is at the discretion of the user. Choice of meta-learner is also at the discretion of the user. We chose ridge regression for two primary reasons. First, Reid and Grudic (2009) suggest regularized regression in stacked generalization, in fact finding that ridge regression performed best in their experiments. Second, regularized regression provides for an interpretable machine learning framework through an optimal weighting of base learners, with respect to the regression model as a meta-learner.

Polikar (2009) and Moreira et al. (2012) present meta-learner options as part of their surveys of ensemble learning approaches for classification and for regression respectively. We will not present an exhaustive list of alternatives for the meta-learner here, but mention two promising options we are currently pursuing. Merz and Pazzani (1999) suggests applying principal components regression for overcoming multi-collinearity issues in correlated base learner predictions. Friedman and Popescu (2003) proposes an importance sampling learning ensemble (ISLE) for combining base learner predictions. The models are chosen through a Monte Carlo sampling scheme and the model weights are chosen by a regularized regression scheme. Friedman and Popescu (2008) presents ISLE as a unifying ensemble framework by thinking of the base learners as rules derived from the data. The correct decision analysis for combining these rules will improve prediction accuracy

and, more importantly, aid interpretation. Akdemir et al. (2013) extends this rule ensembles approach by using soft rules (e.g., converting hard binary decision rules from a decision tree into smooth decision functions via logistic regression).

The study in Chapter 2 serves as a first illustration of ensemble learning for estimating individualized treatment effects in student success efficacy studies. Generalizability is a critical component to putting these machine learning approaches into educational data mining practice. Our current work not only considers alternative implementations for predictive performance improvement in our ensemble learning framework, but testing and evaluating the effectiveness of the methods across a suite of educational data sets.

We find the open source statistical software environment *R* (R Core Team, 2020) ideal for our educational data mining tasks. Though we coded our own ensemble learner, we note here that a number of *R* packages exist to perform ensemble learning. The package `Rminer` (Cortez, 2020) presents a suite of 14 classification and 15 regression methods. The package `caret` (Kuhn, 2008) presents a training/tuning environment for a set of 23 machine learning methods in *R*. We may present an ensemble learning wrapper around the output from these *R* packages. The package `subsemble` (LeDell et al., 2015; Sapp et al., 2014) presents a subset ensemble prediction method on a set of up to 30 machine learning methods. Subsemble is a variant of the Super Learner prediction method of van der Laan et al. (2007), which is implemented in the `H2Oensemble` (LeDell, 2020) and `SuperLearner` (Polley et al., 2020) packages.

On the front of student success efficacy studies, course (student) performance, as measured by instructor-created measures of student learning in our study, provides one avenue for evaluating an intervention. Statistical reasoning, student attitudes and beliefs, and student evaluation surveys provide important alternative angles for assessing the effectiveness of an intervention on learning (Gundlach et al., 2015). The Statistics Education field has validated a number of concept inventories and standardized assessment instruments which we plan to incorporate into future studies of reforms in the Statistics classroom.

Our experience and expertise lies within higher education, university systems. However, we may envision analogous student success studies in public school (K-12) or community college districts, of (online) tutoring systems, or for continuing education and adult education programs. In each of these settings, individualized treatment effects allow us to evaluate and refine initiatives/programs, assess impact on (at-risk) subgroups, and quantify program impact relative to resource demands.

BIBLIOGRAPHY

[1] 1st International Conference on Learning Analytics and Knowledge, Banff, Alberta, 2011. https://tekri.athabascau.ca/analytics/.

[2] D. Akdemir, N. Heslot, and J.-L. Jannink, *Soft rule ensembles for supervised learning*, technical report, 2013. arXiv:1205.4476v3.

[3] E. Alpaydin, *Introduction to Machine Learning*, The MIT Press, Cambridge, MA., 2nd ed., 2010.

[4] P. Austin, P. Grootendorst, and G. Anderson, *A comparison of the ability of different propensity score models to balance measured variables between treated and untreated subjects: a monte carlo study*, Statistics in Medicine, 26 (2007), p. 734–753.

[5] P. Austin and E. Stuart, *Moving towards best practice when using inverse probability of treatment weighting (iptw) using the propensity score to estimate causal treatment effects in observational studies*, Stat Med, (2015), p. 34:3661–79.

[6] J. E. Beck and J. Mostow, *How who should practice: Using learning decomposition to evaluate the efficacy of different types of practice for different types of students*, in International Conference on Intelligent Tutoring Systems, Springer, Berlin, Heidelberg, 2008, pp. 353–362.

[7] J. Beemer, K. Spoon, L. He, J. Fan, and R. Levine, *Ensemble learning for estimating individualized treatment effects in student success studies*, International Journal of Artificial Intelligence in Education, 28 (2017). 10.1007/s40593-017-0148-x.

[8] L. Breiman, *Stacked regressions*, Machine Learning, 24 (1996), pp. 49–64.

[9] L. Breiman, *Random forests*, Machine Learning, 45 (2001), pp. 5–32.

[10] P. Coretz, *Data mining classification and regression methods*, (2020). url: https://cran.r-project.org/package=rminer.

[11] P. Cortez and A. Silva, *Using data mining to predict secondary school student performance*, in 5th Future Business Technology Conference, 2008, pp. 5–12.

[12] P. Dawson, J. van der Meer, J. Skalicky, and K. Cowley, *On the effectiveness of supplemental instruction: A systematic review of supplemental instruction and peer-assisted study sessions literature between 2001 and 2010*, Review of Educational Research, 84 (2014), pp. 609–639.

[13] J. A. N. Dorresteijn, F. L. J. Visseren, P. M. Ridker, A. M. J. Wassink, N. P. Paynter, W. W. Steyerberg, Y. van der Graaf, and N. R. Cook, *Estimating treatment effects for individual patients based on the results of randomised clinical trials*, BMJ, (2011), p. 343.

[14] J. H. Friedman and B. E. Popescu, *Importance sampled learning ensembles*, technical report, Department of Statistics, Stanford University, 2003.

[15] J. H. Friedman and B. E. Popescu, *Predictive learning via rule ensembles*, Annals of Applied Statistics, 2 (2008), pp. 916–954.

[16] D. Gasevic, C. Rose, G. Siemens, A. Wolff, and Z. Zdrahal, *Learning analytics and machine learning*, in Proceedings of the Fourth International Conference on Learning Analytics And Knowledge, Association for Computing Machinery, 2014, p. 287–288. DOI: 10.1145/2567574.2567633.

[17] S. Grajek, *Top 10 it issues, 2019: The student genome project*, technical report, EDUCAUSE Review, 2019.

[18] M. Guarcello, R. Levine, J. Beemer, et al., *Balancing student success: Assessing supplemental instruction through coarsened exact matching*, Tech Know Learn, 22 (2017), p. 335–352. doi:10.1007/s10758-017-9317-0.

[19] E. Gundlach, K. A. R. Richards, D. Nelson, and C. Levesque-Bristol, *A comparison of student attitudes, statistical reasoning, performance, and perceptions for web-augmented traditional fully online, and flipped sections of a statistical literacy class*, Journal of Statistics Education, 23 (2015), p. 33.

[20] M. Hallett, J. Fan, X. Su, R. Levine, and M. Nunn, *Random forest and variable importance rankings for correlated survival data, with applications to tooth loss*, Statistical Modelling, (2014), pp. 14:523–547.

[21] L. He, R. A. Levine, J. Fan, and J. Stronach, *Random forest as a predictive analytics alternative to regression in institutional research*, Practical Assessment, Research, and Evaluation, 23 (2018).

[22] C. R. Henrie, L. R. Halverson, and C. R. Graham, *Measuring student*

engagement in technology-mediated learning: A review, Computers & Education, 90 (2015), pp. 36–53.

[23] R. Hodges and W. G. White, *Encouraging high-risk student participation in tutoring and supplemental instruction*, Journal of Developmental Education, 24 (2001), pp. 2–10.

[24] R. A. Huebner, *A survey of educational data-mining research.*, Research in Higher Education Journal, 19 (2013).

[25] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, Springer, New York, 2013.

[26] S. M. Jayaprakash, E. W. Moody, E. J. M. Lauria, J. R. Regan, and J. D. Baron, *Early alert of academically at-risk students: an open source analytics initiative*, Journal of Learning Analytics, 1 (2014), pp. 6–47.

[27] S. Kotsiantis, K. Patriarcheas, and M. Xenos, *A combinational incremental ensemble of classifiers as a technique for predicting students' performance in distance education*, Knowledge-Based Systems, (2010), pp. 529–535.

[28] D. Kucak, V. Juricic, and G. Dambic, *Machine learning in education - a survey of current research trends*, in Proceedings of the 29th DAAAM International Symposium, B. Katalinic, ed., Vienna, Austria, 2018, DAAAM International, pp. 0406–0410. DOI: 10.2507/29th.daaam.proceedings.059.

[29] G. D. Kuh, *High-impact educational practices: What they are, who has access to them, and why they matter*, technical report, Association of American Colleges and Universities, Washington DC, 2008.

[30] M. Kuhn, *Building predictive models in r using the caret package*, Journal of Statistical Software, 28 (2008), pp. 1–26.

[31] M. LeBlanc and R. Tibshirani, *Combining estimates in regression and classification*, Journal of the American Statistical Association, 91 (1996), pp. 1641–1650.

[32] E. LeDell, *H2o ensemble learning*, (2020). url: https://cran.r-project.org/package=h2oensemble.

[33] E. LeDell, S. Sapp, and M. van der Laan, *An ensemble method for combining subset-specific algorithm fits*, (2015). url: https://cran.r-project.org/package=subsemble.

[34] B. Lee, J. Lessler, and E. Stuart, *Weight trimming and propensity score weighting*, PLoS ONE, 6 (2011). https://doi.org/10.1371/journal.pone.0018174.

[35] J. Lee, B.and Lessler and E. Stuart, *Improving propensity score weighting using machine learning*, Statistics in Medicine, 29 (2010), p. 337–346.

[36] P. Long and G. Siemens, *Penetrating the fog: Analytics in learning and education*, EDUCAUSE review, (2011), pp. 31–40.

[37] M. Lopez-Raton, M. X. Rodriguez-Alvarez, C. C. Suarez, and F. G. Sampedro, *Optimalcutpoints: An r package for selecting optimal cutpoints in diagnostic tests*, Journal of Statistical Software, 61 (2014), pp. 1–36.

[38] D. C. Martin and D. R. Arendale, *Supplemental Instruction: Improving First-Year Student Success in High-Risk Courses (2nd Ed)*, Columbia: National Resource for the First Year Experience and Students in Transition, University of South Carolina, 1993.

[39] H. J. P. Matthew T. Hora, Jana Bouwma-Gearhart, *Data driven decision-making in the era of accountability: Fostering faculty data cultures for learning*, The Review of Higher Education, 40 (2017).

[40] D. Mccaffrey, G. Ridgeway, and A. Morral, *Propensity score estimation with boosted regression for evaluating causal effects in observational studies*, Psychological methods, 9 (2005), pp. 403–25.

[41] C. J. Merz and M. J. Pazzani, *A principal components approach to combining regression estimates*, Machine Learning, 36 (1999), pp. 9–32.

[42] H. Moon, H. Ahn, R. Kodell, S. Baek, C. Lin, and J. Chen, *Ensemble methods for classification of patients for personalized medicine with high-dimensional data*, Artificial Intelligence in Medicine, 36 (2007), pp. 197–207.

[43] J. M. Moreira, C. Soares, A. M. Jorge, and J. F. de Sousa, *Ensemble approaches for regression: A survey*, ACM Computing Surveys, 45 (2012), pp. 10:1–10:40.

[44] A. Naimi and L. Balzer, *Stacked generalization: an introduction to super learning*, Eur J Epidemiol, 33 (2018), p. 459–464.

[45] P. C. of Advisors on Science and T. (PCAST), *Engage to excel: Producing one million additional college graduates with degrees in stem*,

technical report, 2012.
https://www.whitehouse.gov/administration/eop/ostp/pcast/docsreports.

[46] C. Otte, *Safe and interpretable machine learning: A methodological review.*, in Computational Intelligence in Intelligent Data Analysis. Studies in Computational Intelligence, M. C. and N. A., eds., vol. 445, Springer, Berlin, Heidelberg, 2013, pp. 111–122.

[47] A. Pardo, O. Poquet, R. Martinez-Maldonado, and S. Dawson, *Provision of data-driven student feedback in la and edm*, Handbook of Learning Analytics, (2017), pp. 163–174.

[48] Z. A. Pardos, M. Dailey, and N. Heffernan, *Learning what works in its from non-traditional randomized controlled trial data*, International Journal of Artificial Intelligence in Education, 21 (2011), pp. 47–63.

[49] Z. A. Pardos, S. M. Gowda, R. S. Baker, and N. T. Heffernan, *The sum is greater than the parts: Ensembling models of student knowledge in educational software*, SIGKDD Explor. Newsl., 13 (2012), p. 37–44.

[50] K. Pelletier, *Student success: 3 big questions*, Technical Report 54, EDUCAUSE Review, 2019.

[51] R. Polikar, *Ensemble based systems in decision making*, IEEE Circuits and Systems Magazine, 6 (2006), pp. 21–45.

[52] E. Polley, E. LeDell, and M. van der Laan, *Super learner prediction*, (2020). url: https://cran.r-project.org/package=SuperLearner.

[53] E. Polley, S. Rose, and M. van der Laan, *Targeted Learning: Causal Inference for Observational and Experimental Data*, Springer, New York, 2011. Chapter 3.

[54] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2020.

[55] J. Rastrollo-Guerrero, J. A. Gomez-Pulido, and A. Dominguez, *Analyzing and predicting students' performance by means of machine learning: A review*, Applied Sciences, 10 (2020).

[56] S. Reid and G. Grudic, *Regularized linear models in stacked generalization*, in Proceedings of the 8th International Workshop on Multiple Classifier Systems, J. A. Benediktsson, J. Kittler, and F. Roli, eds., 2009, pp. 112–121.

[57] R. J. Rincones-Gomez, *Evaluating student success interventions. principles*

and practices of student success, technical report, Lumina Foundation for Education, 2009.

[58] P. Rosenbaum, *Model-based direct adjustment*, Journal of the American Statistical Association, 82 (1987), p. 387–394.

[59] P. Rosenbaum and D. Rubin, *The central role of the propensity score in observational studies for causal effects*, Biometrika, 70 (1983), p. 41–55. https://doi.org/10.1093/biomet/70.1.41.

[60] S. Sapp, M. J. van der Laan, and J. Canny, *Subsemble: An ensemble method for combining subset-specific algorithm fits*, Journal of Applied Statistics, 41 (2014), pp. 1247–1259.

[61] J. S. Savery, *Overview of pbl: Definitions and distinctions*, Interdisciplinary Journal of Problem-based Learning, 1 (2006), pp. 9–20.

[62] B. Schloerke, J. Allaire, and B. Borges, *learnr: Interactive tutorials for r. r package version 0.10.0.*, (2020). url: https://CRAN.R-project.org/package=learnr.

[63] S. Setoguchi, S. Schneeweiss, M. Brookhart, R. Glynn, and E. Cook, *Evaluating uses of data mining techniques in propensity score estimation: a simulation study*, Pharmacoepidemiology and Drug Safety, 17 (2008), p. 546–555.

[64] K. Spoon, J. Beemer, J. Whitmer, J. Fan, J. Frazee, J. Stronach, A. Bohonak, and R. Levine, *Random forests for evaluating pedagogy and informing personalized learning*, Journal of Educational Data Mining, 8 (2016), pp. 20–50. https://doi.org/10.5281/zenodo.3554595.

[65] X. Su, J. Fan, A. Wang, and M. Johnson, *On simulating multivariate failure times*, International Journal of Applied Mathematics & Statistics, 5 (2006), pp. 8–18.

[66] R. Tibshirani, *Regression shrinkage and selection via the lasso*, Journal of the Royal Statistical Society, 58 (1996), pp. 267–288.

[67] M. J. van der Laan, E. C. Polley, and A. E. Hubbard, *Super learner*, Statistical Applications in Genetics and Molecular Biology, 6 (2007).

[68] A. Wise and J. Vytasek, *Learning analytics implementation design*, Handbook of Learning Analytics, (2017).

[69] D. Wolpert, *Stacked generalization*, Neural Networks, 2 (1992), pp. 241–259.