

AN ABSTRACT OF THE DISSERTATION OF

Njesa Totty for the degree of Doctor of Philosophy in Statistics presented on July 26, 2023.

Title: Statistics in Education: Contributions to Teaching and Data Analysis

Abstract approved: _____

Claudio Fuentes

James Molyneux

In this dissertation we present a compilation of the research conducted during the author's doctoral program. In the first part, we discuss a case study regarding the impact of scholarships on student success at Oregon State University (OSU). Specifically, we look at the graduation and retention rates and aim to determine how the amount of financial aid provided to the students impacts these metrics, especially those who belong to vulnerable student groups.

In the case study, we analyze data from first-time full-time (FTFT) freshmen that enrolled at OSU between 2011 and 2013. Using statistical models we first quantify and characterize the relationship between the amount of financial aid received by the students and the corresponding retention and graduation rates. As expected, the results show that the probabilities of retention and graduation increase as the amount of gift aid increases.

We find that financial aid seems to have a greater impact on first year students. Furthermore, we are able to characterize how these probabilities change when comparing students from different demographic groups. We find that these changes are more noticeable when looking at students in groups determined by Pell-eligibility, first-generation student status and financial need, even after accounting for metrics of student performance.

We also discuss the problem of developing accurate models to predict the probability of retention and graduation based on the amount of financial aid offered to students and other relevant information. Such predictive models can be potentially used to guide policies and determine thresholds for scholarship amounts required to achieve the desired levels of graduation and retention rates at the university. Moreover, these models can be used to close achievement gaps for students from traditionally under-privileged backgrounds.

We discuss the technical problem of binary classification with an imbalanced response variable and overlap in the feature space. These data difficulties present a challenge to the development of good predictive models for classification. The development of solutions to this problem is an area of active research in statistical and machine learning. In order to contribute a solution to this problem we first use simulations to characterize the impacts of imbalance and overlap in a variety of scenarios. The results of the simulation study are used in the creation of our novel algorithm for correcting the technical problem. Upon revisiting the predictive component of our practical problem on student success we found evidence of improved performance in certain cases where our algorithm was applied.

The second part of the dissertation concerns the development and expansion of pedagogical practices for teaching statistical methods in higher education. Specifically, we discuss simple bootstrap methods that are often taught in introductory statistics courses. Bootstrapping and other resampling methods are progressively appearing in the textbooks and curricula of courses that introduce undergraduate students to statistical methods.

Some simple bootstrap-based inferential methods have more relaxed assumptions than their traditional counterparts possibly making it difficult to communicate their importance to students. Students and instructors of introductory statistics courses who are made aware of differences in the performance of these methods will better understand the importance of these assumptions. We detail some of the assumptions that the simple bootstrap relies on when used for uncertainty quantification and hypothesis testing.

We emphasize the importance of these assumptions by using simulations to investigate the performance of these methods when they are or are not met. We also discuss software options for introducing undergraduate students to these bootstrap methods including the newly developed R package `bootEd`.

The individual parts of this dissertation fall under the unifying theme of *statistics in education*. The results of our case study and our novel algorithm contribute to the use of statistics in the education sector. Meanwhile our pedagogical research on the bootstrap contributes to the teaching of statistics in the education sector. The ideas presented in this dissertation can, however, be extended to improve the teaching of subjects other than statistics and the analysis of data generated outside of educational settings. This research could also motivate future efforts to increase the functionality of institutions of education, which are quite foundational to a progressive and ethical society.

©Copyright by Njesa Totty
July 26, 2023
All Rights Reserved

Statistics in Education: Contributions to Teaching and Data Analysis

by

Njesa Totty

A DISSERTATION

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Presented July 26, 2023
Commencement June 2024

Doctor of Philosophy dissertation of Njesa Totty presented on July 26, 2023.

APPROVED:

Co-Major Professor, representing Statistics

Co-Major Professor, representing Statistics

Chair of the Department of Statistics

Dean of the Graduate School

I understand that my dissertation will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my dissertation to any reader upon request.

Njesa Totty, Author

ACKNOWLEDGEMENTS

I would like to thank my advisors Claudio Fuentes and James Molyneux for their endless support and encouragement throughout my entire graduate program. It has been a long road to get here and I am so thankful for their patience and grace, especially for allowing me to make and learn from my mistakes along the way. They have contributed immensely to the successful completion of coursework and research as well as an academic job search. Their support has led to much of my growth and success throughout the program and for that I thank them.

I would like to thank Charlotte Wickham and Katie McLaughlin for their contributions to my graduate program as professors and committee members. I received very impactful advice and feedback from Katie while taking the consulting practicum course that helped to improve my presentation skills and my professionalism overall. This advice is still positively impacting me to this day. Additionally, taking data visualization and PCS with Charlotte has had a huge impact on my R programming skills. Working as a TA with Charlotte also provided many great examples of how to teach statistics and R programming very well. These experiences have been and continue to be very valuable to the start of my teaching-focused career in academia.

I would like to sincerely thank my parents and siblings for their prayers, encouragement, and financial and emotional support throughout my entire graduate program. Additionally, my family at Kings Circle Church and Chi Alpha have provided an immeasurable amount of mental, emotional, and spiritual support throughout this program. I am certain that I would not have made it to this point without them in my life. It has been an invaluable experience to have such a great support system. I thank God for bringing these and other wonderful people into my life throughout the program and for helping me to fulfill my purpose in life.

TABLE OF CONTENTS

	<u>Page</u>
I Assessing the Effect of Scholarships on Student Success	1
1 Introduction	2
2 Modeling Student Success: A Case Study at Oregon State University (OSU)	9
2.1 OSU Data Summaries	12
2.1.1 Evaluating Student Success and Demographics	14
2.1.2 Understanding Student Success with Considerations to Financial Need and Demographics	16
2.1.3 Understanding Whether Scholarships Effect Student Success	18
2.1.4 Examining Student Success and Racial Group	23
2.1.5 Trends and Patterns within Demographic Groups	29
2.2 Modeling Approaches for Inference	34
2.2.1 A Brief Introduction to Statistical Learning	34
2.2.2 Accuracy Versus Interpretability	36
2.2.3 Regression Methods for Inference	37
2.3 Statistical Analysis Towards Inference	42
2.3.1 Cohort-by-Cohort Models	43
2.3.2 The Effect of Aid by Demographics	44
2.3.3 Accounting for Academic Performance	47
2.3.4 Further Exploring the Impact of Gift Aid by Race	53
2.4 Moving Towards Predictive Models	56
3 Dealing with Imbalanced Data and the SMOTE Approach	58
3.1 Extending the Logistic Regression Model	60
3.2 Overlap and Imbalance	67

TABLE OF CONTENTS (Continued)

	<u>Page</u>
3.2.1 Overlap Metrics	67
3.2.2 Random Undersampling and Editing Methods	68
3.2.3 Random Oversampling (ROS) and Its Derivatives	69
3.2.4 The Synthetic Minority Oversampling TEchnique (SMOTE)	70
3.3 Statistical Learning Methods for Prediction	72
3.3.1 Random Forests	72
3.3.2 Neural Networks	77
3.4 Statistical Analysis Towards Predictions	80
3.4.1 Single Decision Trees	80
3.4.2 Random Forests + SMOTE	89
3.4.3 Neural Networks + SMOTE	97
3.5 The Need to Further Study and Improve SMOTE	101
4 The Strategic SMOTE (S-SMOTE) Algorithm: A Simulation-Based Approach	102
4.1 Preliminary Details of S-SMOTE	105
4.2 Outline of Simulation Study	110
4.3 Results of SMOTE Simulation Study	117
4.4 Studying Hyperparameters of Strategic SMOTE (S-SMOTE)	143
4.5 Statement of S-SMOTE Algorithm	152
5 Example Applications of S-SMOTE to Real Data	156
5.1 Revisiting OSU Data	157
5.2 Application to Benchmark Imbalanced Datasets	166
6 Conclusions	176

TABLE OF CONTENTS (Continued)

	<u>Page</u>
II A Simulation-Based Approach to Teaching the Bootstrap	180
7 Introduction	181
7.1 Benefits of Teaching Statistical Computing and the Bootstrap	182
8 General Assumptions for Simple Applications of the Bootstrap	186
8.1 Interval Estimation	187
8.1.1 The Basic Interval (The Base Case)	188
8.1.2 The Percentile Interval (The Symmetric Case)	189
8.1.3 The Studentized Interval (The Studentized Case)	193
8.2 Bootstrap-Based Hypothesis Tests	195
8.2.1 Studentized Pivots	197
8.2.2 Locational Pivots	198
8.3 Summary	199
9 Simulation-Based Performance Evaluations of the Bootstrap	201
9.1 Simulation Results	203
9.2 Discussion	218
9.3 <code>bootEd</code> : An R Package for Teaching the Bootstrap	221
10 Conclusions	225
Appendices	242
A Supplementary Visualizations of SMOTE Simulation Results	243
B More Detailed Supplementary Visualizations of SMOTE Simulation Results	247

TABLE OF CONTENTS (Continued)

	<u>Page</u>
C Supplementary Visualizations of Simulation Results Studying Hyperparameters of S-SMOTE	347

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
1.1	Screenshot from the Undergraduate Student Success Initiative (USSI) summary page.	3
2.1	Proportion of students, of those in more severe need, who were retained or graduated. Proportions are broken down by demographics and whether their gift aid was larger than the median amount for all students in need across all cohorts. The medians were 7,095 and 15,197 dollars for first-year and total four-year gift aid, respectively. For some demographics, such as first-generation status, differences in success rates are more severe for students who also received less than the typical amount of aid.	22
2.2	Retention rates for students with more severe need. These are broken down by racial group and whether total first-year gift aid was greater than the overall median for all students with more severe need (7,095 dollars). Lower rates observed for student receiving less aid, and decrease varies across racial groups.	26
2.3	Graduation rates for students with more severe need. These are broken down by racial group and whether total gift aid over the first 4 years was greater than the overall median for all students with more severe need (18,003 dollars). Even lower rates than observed with retention are observed for student receiving less aid, and this decrease greatly varies across racial groups, more than in retention rates.	28
2.4	Scatterplot of first year retention vs total gift aid. The blue points indicate the students that were retained, $y = 1$, and were not retained, $y = 0$, after the first year for the corresponding amount of total gift aid.	38
2.5	Jitter plot of first year retention vs. total gift aid bracket. The points in each bracket represent the students that were and were not retained after their first year and received the corresponding range of total gift aid.	39
2.6	Estimated probabilities for first year retention in terms of total first-year gift aid. The dashed lines depict a first-year retention probability of 0.85 on the y -axis and the average total gift aid for year 1 on the x -axis.	42
2.7	Predicted probability curves for first-, second-, and third-year retention across each cohort. Regardless of cohort, first-year retention is lower across all aid categories. If the average award amount for the first year (about 6,200 dollars for those receiving aid) were also awarded in subsequent years, higher probabilities of retention would result.	43

LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
<p>2.8 Plots for the estimated probabilities of first year retention in terms of gift aid, accounting for the effect of the demographics of interest. The dashed lines indicate a probability of 85% in the y-axis, and the average amount of gift aid in the x-axis. We observe that Pell-eligibility, first generation and financial need seem to have a greater effect in the estimated probabilities.</p>	47
<p>2.9 Plots for the estimated probabilities of second year retention in terms of gift aid, accounting for the effect of the demographics of interest. The dashed lines indicate a probability of 85% in the y-axis, and the average amount of gift aid in the x-axis. We observe that Pell-eligibility, first generation and financial need seem to have a greater effect in the estimated probabilities.</p>	48
<p>2.10 Plots for the estimated probabilities of third year retention in terms of gift aid, accounting for the effect of the demographics of interest. The dashed lines indicate a probability of 85% in the y-axis, and the average amount of gift aid in the x-axis. We observe that Pell-eligibility seem to have the greater effect in the estimated probabilities, and first generation and financial need seem to have a moderate effect. Confidence bands are wider when the standard error of a prediction was larger than normal. Predicted probabilities close to 1 produced upper bounds above 1, which were capped to 1. This was the case with the model for financial need.</p>	49
<p>2.11 Plots for the estimated probabilities of graduation in terms of gift aid, accounting for the effect of the demographics of interest. The dashed lines indicate a probability of 85% in the y-axis, and the mean total amount of gift aid in the x-axis. We observe that Pell-eligibility, first generation and financial need seem to have a greater effect in the estimated probabilities, and student of color seem to have a moderate effect.</p>	50
<p>2.12 First year retention probabilities versus gift aid. The left- and right-hand panels give the estimated probabilities as aid increases, after accounting for composite SAT scores and high school GPA, respectively. In both plots the red line corresponds to the group with lower academic performance in the respective metric, and the blue line corresponds to the group with higher performance. Only students receiving aid were included here.</p>	51
<p>2.13 Fitted probability curves from logistic regression models for total first-year gift aid and first-year retention. Left-most panel are curves from model that accounts only for race. Right-most two panels are results for model that accounts for race and need. Rug gives range of total first-year gift aid variable for each population in the model corresponding to the facet.</p>	54

LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
<p>2.14 Six-year graduation predictions for all students by race, and then less- or more-severely in need students by race. Additive effects are present for both race and need, noted by the change in starting points between the second and third panel and within each of these panels as well. Rug gives range of total gift aid over first four years for each population in the model corresponding to the facet.</p>	56
<p>3.1 Total aid in thousands of dollars given to students who were retained (blue) and who deserted (brown). In each year, many values for aid are similar regardless of a students retention status. This issue is called overlap, and it presents a challenge when trying to classify observations.</p>	60
<p>3.2 ROC Curve and tile plot of specificity and sensitivity from larger logistic regression model for retention. The ROC curve provides TPR (y) and FPR (x) for varying thresholds. The colors in the tile plot are mapped to the value of the sensitivity and specificity for various thresholds (p) on the x-axis. The threshold is the value of the predicted probability over which we classify an observation as a success or retained. These plots provide evidence that using a lower threshold will produce a larger decrease in sensitivity than the increase in sensitivity.</p>	65
<p>3.3 Reliability diagram for test set predictions using fuller logistic regression model. The panel labels give the relative frequency of retention taken over all points with predicted probabilities of retention falling inside the interval on the x-axis.</p>	66
<p>3.4 Visual of the SMOTE. Minority class points are black circles while majority class points are the open triangles. For the two minority class examples of interest (in red), synthetic points are randomly generated along the solid lines joining them to their 5 neighbors.</p>	71
<p>3.5 Two candidate decision trees yielding the same classification error. However, Tree 2 has greater purity.</p>	74
<p>3.6 Visual representation of a single-layer feed-forward neural network for data with $p = 4$ features, $K = 5$ hidden units, and a single output layer. The input layer (yellow) is made up of the features, each of which is provided to the hidden units in the single hidden layer (violet). The output of the hidden units are the activations. The output layer (red) provides a linear combination of these. Credit: <i>An Introduction to Statistical Learning (ISLR)</i> (James et al. 2013).</p>	78

LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
<p>3.7 Average performance metric for each combination of <code>cp</code> and <code>minsplits</code> based on 10-fold cross validation. Accuracy, ROC (the area under the curve), sensitivity, and specificity are provided. The right-top indicates less complex models while the left-bottom indicates more complex models. The direction towards which each metric increases does not agree across all metrics, indicating that a tradeoff is inevitable.</p>	82
<p>3.8 Decision tree for first-year retention. Amongst all variables considered, high school GPA was deemed to be the most influential factor. First-year gift aid was most often used to separate class labels. Salmon colored nodes correspond to predicting retention and brown nodes desertion. The two decimals in each node give the proportion of observations in that node that deserted and returned, respectively. The percentage in each node gives the percent of observations falling into the node out of all students in the training set. There are some discrepancies due to rounding.</p>	83
<p>3.9 Calibration plot for test set predictions from single tree trained using cross-validated hyperparameters. Predicted probabilities of retention were binned from 0 to 1 by 0.05. Purple points give the average predicted probability (x) and retention rate (y) by bin. The line $y = x$ is also provided in black. Labels give the number of observations and bounds of the interval. Intervals without observations are omitted.</p>	85
<p>3.10 Overall and class-specific performance metrics (y) resulting from oversampling the minority class a certain percentage (x). Specificity increases but sensitivity and overall accuracy decrease, indicating an disadvantageous trade-off between class-specific accuracies. Trees were pruned using 10-fold cross-validation to select the complexity parameter and the process was repeated 250 times.</p>	87
<p>3.11 Cross-validated metrics from random forest models fit to unbalanced training data (top row of panels) and after applying SMOTE (bottom row of panels). A total of 5 neighbors were used in SMOTE to generate 4 synthetic examples from each minority example. Models were trained via 5-fold cross validation to find the optimal values for <code>mtry</code>, number of trees (y) and node size (x). SMOTE results may be inflated due to the additional synthetic examples. Note that the color gradient is not the same across all metrics since they did not all have the same range.</p>	91

LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
<p>3.12 OOB and test data error rates by tree. The dotted lines correspond to corrected errors calculated using only non-synthetic data. These were calculated to guard against the reporting of inflated errors alone. The large increase in error for the negative class on test data in comparison to OOB data provides evidence of overfitting.</p>	92
<p>3.13 Calibration plot for random forest model fit to training data with SMOTE applied. Results were obtained using test data predictions. The model over-predicts often, especially when the relative frequency of retention was actually low.</p>	94
<p>3.14 ROC plot for random forest model fit to training data with SMOTE applied. Results were obtained using test data predictions. The threshold giving the highest sum sensitivity plus specificity is plotted. Results indicate that a higher threshold could lead to better predictive results overall.</p>	94
<p>3.15 Variable importance plots for random forest model fit to training data with SMOTE applied. Results were obtained using oversampled data so values are inflated. Left- and right-hand panels correspond to the importance of the variable to node purity (separating the classes) and overall predictions. . . .</p>	96
<p>3.16 Tracking performance as neural networks are trained. Hyperparameters were first selected using holdout validation. Then those which maximized the AUC were selected and the final models were trained. The performance by epoch is given for each metric. Models were trained for 100 epochs. The model trained on SMOTE oversampled data exhibited more signs of overfitting than that trained on unbalanced data. This can be seen by comparing the differences between training set performance and validation set performance.</p>	100
<p>4.1 Example jittered plot of relative dominance and number of minority neighbors used for oversampling. This is plotted for each point in the minority class. Red numbers indicate priority of points when oversampling. Points in region 1 have the least dominance and density, therefore they are used for oversampling the most. Horizontal and vertical lines are the medians of the respective axes. Quadrants are defined using values less than or equal to the median number of neighbors and greater than or equal to the median relative dominance. . .</p>	108
<p>4.2 Median AUC and balanced accuracy, taken over the median performance for each oversampling method. X-axis provides overlap and imbalance amounts. Y-axis indicates which data difficulty was present. The Base Case is $N = 750$, 100 variables, and no categorical or missing data. The median was calculated out of 23 medians which were themselves calculated out of 500 values.</p>	118

LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
<p>4.3 Median specificity and sensitivity given by light gray point. Median was calculated over 500 repetitions for a given sampling method. Each training set had 750 rows and 100 variables. Of those 25% or 85% of variables were categorical and there was no missing data. The purple point gives the median for each group of points over all sampling methods. Dark gray point mark the results for unbalanced data.</p>	121
<p>4.4 Median specificity and sensitivity given by light gray point. Median was calculated over 500 repetitions for a given sampling method. Each training set had 750 rows and 100 quantitative variables. 30% or 70% of cells in the dataset belonging to the minority class were missing at random. The purple point gives the median for each group of points over all sampling methods. Dark gray point mark the results for unbalanced data.</p>	122
<p>4.5 Distribution of balanced accuracy and Kappa coefficient based on oversampling method. Data were aggregated over all simulations to make an initial determination about whether the oversampling method used has an impact.</p>	124
<p>4.6 Distribution of balanced accuracy from training data where SMOTE or S-SMOTE (Strategic) was applied. Results were aggregated over the distribution of w and the amount of overlap and imbalance. Each column label of the facet indicates the change from the base case of $N = 750$, 100 variables, and no categorical or missing data.</p>	127
<p>4.7 Difference in median Kappa between S-SMOTE and population oversampling. The median was calculated for a given data scenario, model, amount of overlap and imbalance, and oversampling method specific to the distribution of w. Positive differences indicate superior performance of S-SMOTE, however when this occurs it occurs for most distributions which does not clarify which version of S-SMOTE is superior.</p>	131
<p>4.8 Difference in median NPV between S-SMOTE and SMOTE. The median was calculated for a given data scenario, model, amount of overlap and imbalance, and oversampling method specific to the distribution of w. Positive differences indicate superior performance of S-SMOTE, however when this occurs it occurs for most distributions which does not clarify which version of S-SMOTE is superior.</p>	132

LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
<p>4.9 Distribution of balanced accuracy aggregated over all amounts of overlap and imbalance, models, and whether S-SMOTE or SMOTE was applied. Though there are changes in the distributions across data scenarios, distributions within a given column are quite similar. Indicating that changes in performance due to the distribution of w may be minimal in comparison to those that occur because of the underlying data scenario.</p>	134
<p>4.10 Distribution of specificity aggregated over all amounts of overlap and imbalance, models, and whether S-SMOTE or SMOTE was applied. Though there are changes in the distributions across data scenarios, distributions within a given column are quite similar. Indicating that changes in performance due to the distribution of w may be minimal in comparison to those that occur because of the underlying data scenario.</p>	135
<p>4.11 PDF and CDF of Four-Parameter Beta distribution with parameters $\alpha = 0.5, \beta = 0.5, \min = -1, \max = 1$.</p>	136
<p>4.12 Number of times (x-axis) that difference in majority and minority class overlap both decreased after oversampling. Color and y-axis provide details of the simulation scenario in which these occurred. 500 repetitions were performed for each data scenario, amount of overlap and imbalance, and oversampling method. When 25% of data were categorical many applications of oversampling resulted in the same or less amounts of overlap in the majority and minority class.</p>	138
<p>4.13 For each oversampling method the amount of overlap introduced into the data was calculated as the difference in the class overlap metrics before and after oversampling. The median differences are calculated over all 500 repetitions for a given oversampling method and data scenario. These are the same regardless of the model applied. Here we visualize their relationship as well as the difference in the AUC of test set predictions for models trained with unbalanced and oversampled data. These values did depend on the model applied. We did not observe any further relationship between the changes in overlap and the predictive performance on the test data.</p>	139

LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
<p>4.14 Ranks of median accuracy given on x-axis for each oversampling method on the y-axis. The y-axis labels were removed for the sake of space. Colors are mapped to the general category of oversampling method applied. Medians were calculated with respect to each data scenario, amount of overlap and imbalance, model, and oversampling method. Ranks were then calculated with respect to the oversampling methods. When superior performance of S-SMOTE or SMOTE occurs (e.g. 25% categorical and random forests) it occurs for most distributions of w. These results indicate that superior performance may be due to the general category of oversampling method rather than changes in the distribution of w.</p>	141
<p>4.15 Distribution of balanced accuracy as the maximum number of nearest neighbors k_{max} considered by S-SMOTE varies. Facet labels correspond to the sequence of dominance thresholds checked (top facet label) and the amount of overlap and percent minority examples in the simulated data (bottom facet label). Data were simulated with 60 columns, 30% categorical variables, and 1000 examples before oversampling. 700 synthetic examples were generated.</p>	146
<p>4.16 Proportion of simulations in which no predictions were made for the minority class out of 50. When 5% of points belonged to the minority class 700 synthetic samples were still generated as in the 15% case. This was done to avoid using very little information on the minority class for a large amount of oversampling.</p>	148
<p>4.17 Distribution of balanced accuracy as the weights used to select minority examples for oversampling ρ varies. The median dominance threshold and median number of neighbors used for oversampling was used to determine whether a point was more or less dominated and crowded, respectively. The median was calculated with respect to all minority class points deemed fit for oversampling. Facet labels correspond to the amount of overlap and percent minority examples in the simulated data and characteristic of the data. Each simulated dataset had 60 columns and 1000 rows before oversampling. 700 synthetic examples were generated.</p>	150
<p>5.1 Calibration plot for OSU retention dataset. Relationship between average predicted probability and relative frequency of retention from test data. Predicted probabilities were binned using sequence from 0 to 1 by 0.05. The average was taken over each bin with respect to each model+method combination. The relative frequency of retention was also calculated in this manner as well. The black line is $y = x$ which serves as a reference of the ideal model.</p>	160
<p>5.2 Accuracy of test set predictions broken down by model and oversampling method, race, and Pell-grant eligibility (an indication of student financial need).</p>	162

LIST OF FIGURES (Continued)

<u>Figure</u>	<u>Page</u>
5.3 Sensitivity of test set predictions broken down by model and oversampling method, race, and Pell-grant eligibility (an indication of student financial need).	163
5.4 Specificity of test set predictions broken down by model and oversampling method, race, and Pell-grant eligibility (an indication of student financial need).	164
5.5 Calibration plot for Pima Indians diabetes dataset. Relationship between average predicted probability and relative frequency of diabetes from test data. Predicted probabilities were binned using sequence from 0 to 1 by 0.05. The average was taken over each bin with respect to each model+method combination. The relative frequency of diabetes was also calculated in this manner. The black line is $y = x$ which serves as a reference of the ideal model.	172
5.6 Calibration plot for Haberman's survival dataset. Relationship between average predicted probability and relative frequency of survival from test data. Predicted probabilities were binned using sequence from 0 to 1 by 0.05. The average was taken over each bin with respect to each model+method combination. The relative frequency of survival was also calculated in this manner. The black line is $y = x$ which serves as a reference of the ideal model.	173
9.1 Shifted sampling distributions. For each population and sample size 10,000 sample statistics were calculated. Each sample statistic was shifted by the corresponding parameter of its population.	204
9.2 Studentized sampling distributions. For each population, sample size, and number of bootstrap samples 10,000 sample statistics were calculated. Each sample statistic was shifted by the corresponding parameter of its population and scaled using its bootstrap estimate of standard error. $B = 99$ and $B = 999$ correspond to dark gray and light gray bars, respectively.	206
9.3 Log of the widths of 10,000 studentized bootstrap, t -, and z -intervals for the population mean when the underlying population was Exponential(1). The widths of the z -interval, which were constant for a given value of N and significance level, are marked by a single dashed line. The studentized bootstrap interval produced very wide intervals, especially when the sample size was very small. This may be a reason for the high coverage proportions we observed. 52 studentized bootstrap intervals had undefined values when $N = 5$ and $B = 99$. These were removed before plotting.	209

LIST OF FIGURES (Continued)

<u>Figure</u>		<u>Page</u>
9.4	Rejection rates of z - and t -tests, and bootstrap hypothesis tests for the mean. The y-axis gives the proportion of tests that resulted in rejection for a given hypothesized value, on the x-axis. The vertical and horizontal long-dashed lines mark the true mean and the significance level 0.05, respectively.	216
9.5	Rejection rates of the z -test for proportions and bootstrap hypothesis tests. The y-axis gives the proportion of tests that resulted in rejection for a given hypothesized proportion on the x-axis. The vertical and horizontal long-dashed lines mark the true population proportion, p , and the desired significance level, $\alpha = 0.05$, respectively. The line types map to values of N : solid for $N = 5$, dashed for $N = 20$, dotted for $N = 50$, dot-dashed for $N = 150$. In some cases the studentized intervals had undefined bounds. These were removed, therefore, some rejection rates were calculated out of fewer than 10,000 intervals and some were exactly zero or one. For $N = 5$ and $B = 999$, none of the studentized intervals were well defined, so no rejection rates are plotted.	217
9.6	Histogram of bootstrap sample statistics. The original sample statistic is marked by a solid line. This plot is returned as part of the output from the <code>percentile</code> function.	222

LIST OF TABLES

<u>Table</u>		<u>Page</u>
2.1	Enrollment numbers for each cohort over their first four years at OSU, and final six-year graduation rates. Numbers in parentheses give the percent of students, out of all initially enrolled, that continued through in subsequent years. The last row shows the totals over the three cohorts.	12
2.2	Retention rates and graduation rates with respect to enrollment in year four per cohort. The overall values shown in the last row are based on the totals included in Table 2.1.	13
2.3	Proportion of students in each demographic groups with respect to the initial enrollment of the corresponding cohorts.	14
2.4	Retention and graduation rates by demographics in each cohort	15
2.5	Number of students in high, medium, low, and no financial need and enrollments per cohort for the first four years. The numbers in parentheses indicate the percentage of students that received some form of gift aid in each group and the numbers within brackets indicate the proportion of students with financial need for the respective year-cohort combination	17
2.6	Percentages of students with high, medium, low, or no financial need during their first year by demographic groups for each cohort. Students with an unknown need status or no budget are not included in the table.	19
2.7	First year retention and graduation rates by demographics for students in the cohort 2011-2012	20
2.8	Percentage of students falling into each racial category and overall, by cohort.	23
2.9	Retention rates of all students, by racial group and cohort, with respect to previous year enrollments.	25
2.10	First-year retention and six-year graduation counts and proportions by racial group. Results indicate that the problem of student success is not the same across all racial groups.	30

LIST OF TABLES (Continued)

Table	Page
2.11 First-year retention and six-year graduation counts and proportions by racial group further broken down by need. Patterns in the original data were not always the same after grouping by race. Data on severity of need status over all four years was not available for any American Indian or Alaska Native students, and there were zero Native Hawaiian or Pacific Islander students with less severe need in their first two years. This missing information or zero counts led to missing proportions which are represented by dashes in the table.	31
2.12 Regression output of simple logistic regression model for first-year retention using total first year gift aid in thousands of dollars. Only data for students receiving more than 0 dollars of aid was used.	41
2.13 Summary of significance of the demographic variables when modeling retention and graduation rates in terms of gift aid. The letter a indicates the additive term was significant at level 0.05. the letter i indicates the interaction term was significant at level 0.05, and None indicates that none of these terms were significant at a significance level of $\alpha = 0.05$	45
2.14 Estimated coefficients for additive (add) and interaction (inter) terms of the corresponding demographic variables when modeling the first year retention and graduation probabilities with and without adjusting for academic performance. The numbers in blue indicate the coefficients that are significant at the level $\alpha = 0.05$.	52
2.15 Coefficients of additive terms for models of retention and graduation on total aid by race for students in need and those not in need. All but one of the coefficients are significantly different from 0 for students in need, while for students not in need only two are. Additionally, many coefficients are more negative or smaller for students in need than those not in need.	57
3.1 Retention and graduation rates over time, for all cohorts. Imbalance observed in first-year retention and six-year graduation.	59
3.2 Cross-validation results from full model for retention with many covariates and interactions. Results across folds are consistent. The model struggles to predict the negative class (desertion) with consistently high specificity but consistently low sensitivity. This shows consistent misclassification for the negative class.	63
3.3 Performance metrics for predictions of test data set using fuller logistic regression model. Balanced accuracy is much lower than overall accuracy, indicating inconsistent performance across the classes.	66

LIST OF TABLES (Continued)

<u>Table</u>	<u>Page</u>
3.4 Performance metrics for predictions of test data set using single decision tree. Balanced accuracy is much lower than overall accuracy, indicating inconsistent performance across the classes.	83
3.5 Performance metrics for test set predictions obtained using random forest models trained using data that were first oversampled with SMOTE. Results from single decision tree and logistic regression applied to original training data are also included for reference. The use of random forests and SMOTE improves all performance metrics.	93
3.6 Performance metrics for test set predictions obtained using neural networks. Networks were trained using unbalanced data and data oversampled with SMOTE. Only the F1 Score is improved by oversampling. Surprisingly, the specificity decreased after applying SMOTE which oversamples the negative class. The increase in sensitivity and corresponding drop in specificity indicates that there is a trade-off in class accuracies after oversampling. The underlying issue is the overlap in the feature space.	99
5.1 Performance metrics from predictions for test OSU dataset. These were obtained using random forests and neural networks. Data were oversampled with S-SMOTE or SMOTE or were left unbalanced. Italicized digits are the maximum for that metric out of all model and method combinations. Performance of neural networks with unbalanced data was better than their performance when SMOTE or S-SMOTE was applied.	158
5.2 Mean of the squared differences between the average predicted probabilities and the relative frequency of binned test set predictions. Predicted probabilities were binned using a sequence from 0 to 1 by 0.05. There were 21 bins for each combination of model and method though some were empty in some cases. The denominator of the mean indicates the number of bins for the respective model and method combination (i). The average was taken over all non-empty bins with respect to each model+method combination. The relative frequency of retention were also calculated in this manner.	161
5.3 Information on benchmark datasets. Top table: Sample size N , label of positive class, number and percentage in positive class, and number and percentage in negative class. Bottom table: Minority class, majority class, and overall overlap metrics. The overlap metric is the proportion of examples with more than three out of six nearest neighbors in the opposite class. This is calculated within classes and for the entire dataset.	167

LIST OF TABLES (Continued)

Table	Page
<p>5.4 Performance of LASSO logistic regression and random forests on Pima Indians and Haberman’s survival datasets. Results after applying SMOTE and S-SMOTE are provided as well as those from unbalanced data. S-SMOTE performs better on the minority class, a positive test for diabetes, at some cost of predictive accuracy on the negative class, negative test for diabetes. Random forests performed better than LASSO logistic regression when S-SMOTE was applied or data were left unbalanced. McNem. p refers to the p-value for the test of equal discordant pairs, or misclassifications, in both classes. For each metric and dataset the maximum out all combinations of model and oversampling method is italicized when it exists.</p>	168
<p>5.5 Mean of the squared differences between the average predicted probabilities and the relative frequency of binned test set predictions. Predicted probabilities were binned using a sequence from 0 to 1 by 0.05. There were 21 bins for each combination of model and method though some were empty in some cases. The denominator of the mean indicates the number of bins for the respective model and method combination (i) The average was taken over all non-empty bins with respect to each model+method combination. The relative frequency of the positive classes (diabetes and survival) were also calculated in this manner.</p>	171
<p>8.1 A summary of our discussion on the basic, percentile, and studentized bootstrap intervals. Here B is the number of bootstrap samples (e.g. 999) and $1-\alpha$ is the desired confidence level. The values $(B+1)(1-\alpha/2)$ and $(B+1)(\alpha/2)$ are assumed to be integers which, when used as subscripts, denote the corresponding order statistics of the distribution. We denote an estimate of the standard error of $\hat{\theta}(X)$, based on the data, as $\hat{S}E(\hat{\theta}(x))$. Also, we denote the studentized distribution of bootstrap sample statistics as $z^* = (\hat{\theta}(x^*) - \hat{\theta}(x)) / \hat{S}E(\hat{\theta}(x^*))$, where $\hat{S}E(\hat{\theta}(x^*))$ is an estimate of the standard error of $\hat{\theta}(x^*)$.</p>	195
<p>8.2 Summary of the bootstrap hypothesis tests discussed. $\hat{S}E(\hat{\theta}(x))$ is an estimate for the standard error of $\hat{\theta}(x)$, while $\hat{S}E(\hat{\theta}(x^*))$ is that for $\hat{\theta}(x^*)$. The achieved significance level (ASL) is an approximate p-value, calculated with respect to the bootstrap distribution of test statistics. The level-α hypothesis test based on the studentized pivot is equivalent to rejecting values of θ_0 which are not contained in the $(1 - \alpha) * 100\%$ studentized bootstrap interval. Similarly, the test based on the locational pivot is equivalent to rejecting values of θ_0 which are not contained in the basic bootstrap interval or, if the symmetry assumption holds, the percentile bootstrap interval.</p>	200

LIST OF TABLES (Continued)

Table	Page
9.1 Coverage proportions (C) of intervals for the population mean. 10,000 samples of the specified size (N) were taken from each population. Then, with each sample, B bootstrap samples were used to construct the bootstrap intervals while the traditional intervals were constructed with their usual formulas. The proportion of intervals which contained the true population mean, out of ten thousand, was calculated. These values should be near 0.95 since the significance level was 0.05. The coverage proportion for the studentized interval is out of 9982 intervals because it contained undefined bounds in 52 cases.	207
9.2 Coverage proportions (C) of bootstrap intervals and the z -interval (Wald interval) for the population proportion. Samples of size (N) were taken from each Bernoulli(p) population and B bootstrap samples were used to construct the bootstrap intervals. The z -interval was constructed using its usual formula. Values in parentheses represent the proportion of basic intervals which contained invalid values, the proportion of percentile intervals which contained equal bounds, the number of studentized intervals which contained undefined bounds, and the proportion of z -intervals which contained both invalid values and equal bounds.	211
9.3 Significance levels (α) of the z - and t -tests for the mean. For each population and sample size (N), 10,000 samples were taken. The z - and t -test were used to test $H_0 : \mu = 1$ (which is true for both populations). The proportion of tests which rejected was recorded.	214
9.4 Significance levels (α) of the z -test for proportions. For each Bernoulli(p) population and sample size (N), 10,000 samples were taken. The z -test for proportions was used to test $H_0 : p = p_0$, where p_0 was the true population proportion. The proportion of tests which rejected was recorded.	214

LIST OF APPENDIX FIGURES

<u>Figure</u>	<u>Page</u>
A.1 Median sensitivity and specificity out of all oversampling methods. Median performance metrics was calculated over all 500 results for each data scenario, amount of imbalance and overlap, model, and oversampling method. Then the median was calculated again over the 23 oversampling methods. X-axis provides overlap and imbalance amounts. Y-axis indicates which data difficulty was present. The Base Case is $N = 750$, 100 variables, and no categorical or missing data. The median was calculated out of 23 medians which were themselves calculated out of 500 values.	244
A.2 Median negative predictive value (NPV) and positive predictive value (PPV) out of all oversampling methods. Median performance metrics was calculated over all 500 results for each data scenario, amount of imbalance and overlap, model, and oversampling method. Then the median was calculated again over the 23 oversampling methods. X-axis provides overlap and imbalance amounts. Y-axis indicates which data difficulty was present. The Base Case is $N = 750$, 100 variables, and no categorical or missing data. The median was calculated out of 23 medians which were themselves calculated out of 500 values.	245
A.3 Median F1 score and Kappa coefficient out of all oversampling methods. Median performance metrics was calculated over all 500 results for each data scenario, amount of imbalance and overlap, model, and oversampling method. Then the median was calculated again over the 23 oversampling methods. X-axis provides overlap and imbalance amounts. Y-axis indicates which data difficulty was present. The Base Case is $N = 750$, 100 variables, and no categorical or missing data. The median was calculated out of 23 medians which were themselves calculated out of 500 values.	246
B.1 Median Performance Metrics: Gray points are median specificity taken over 500 predictive performance results. The characteristics of each training set are given in the title at the top of each plot. Each page corresponds to a different set of data characteristics. The balancing methods with the three largest averages are given at the top of each panel as text. The rank was calculated with respect to the metric, model, and data difficulty scenario. The purple point gives the median for each group of points and it was also calculated with respect to these elements. Dark gray point mark the results for unbalanced data. Viewers of these and other very large and detailed plots in the Appendices may need to use the zoom function of their PDF viewer for a better inspection of the results.	247

LIST OF APPENDIX FIGURES (Continued)

<u>Figure</u>		<u>Page</u>
B.2	Distribution of Performance Metrics Solely By Oversampling Method: Distribution of balanced accuracy and Kappa coefficient based on oversampling method. Data were aggregated over all simulations to make an initial determination about whether the oversampling method used has an impact. . . .	278
B.3	Distribution of Performance Metrics For SMOTE and S-SMOTE By Data Case and Model: Distribution of performance metrics by oversampling method (S-SMOTE and SMOTE). Data were aggregated over all distributions for w and overlap/imbalance amounts in order to make initial comparisons between the two methods.	288
B.4	Differences In Performance Metrics By Oversampling Method: Comparisons were made between S-SMOTE, SMOTE and Population oversampling. Differences were calculated with respect to data case, model, amount of overlap and imbalance, and which distribution was used for w	298
B.5	Distribution For Performance Metrics By Distribution for w and Data Scenario: Distributions of each performance metrics are given aggregated over all characteristics except the data scenario and distribution used for w . These were used to determine if there were any changes in performance due simply to the distribution used for w and what data scenarios these occurred in. . .	326
B.6	Ranks of Median Performance Metrics By Oversampling Method, Model, and Data Scenario: The median performance metrics were calculated with respect to each data scenario, amount of imbalance and overlap, model applied, and oversampling method used. These were then ranked with respect to oversampling method (ranks of 1 to 23 possible). The rank is plotted on the x-axis for each oversampling method. Facets correspond to the model fit and data scenario.	336

LIST OF APPENDIX FIGURES (Continued)

Figure

Page

C.1 Distribution of Performance Metrics For Various Parameters of S-SMOTE: The first set of plots (4 facets per page) provides the distribution of performance metrics (y-axis) as the maximum number of nearest neighbors considered ranges from one to 20. We denoted this parameter as k_{max} when describing S-SMOTE. This is faceted by the sequence of thresholds tried in S-SMOTE and the amount of overlap and percent minority examples in the simulated dataset. The second set of plots (3 facets per page) provide the distribution of performance metrics (y-axis) as the sampling weights used to select minority class examples for oversampling change. These weights determine how often we oversample minority points after they have been separated using the median dominance threshold met and number of neighbors deemed fit for oversampling. The third value corresponds to the quadrant that least dominated and crowded minority examples fall into followed by the second value, fourth value, and the first value. 50 repetitions were performed for each simulation with a different dataset each time. Datasets were simulated in the same manner that was used for the larger simulations. When the overlap amount was 0.9 and 5% of points belonged to the minority class many values for the NPV were missing, leading to more variable results (see Figure 4.16). 347

Part I

Assessing the Effect of Scholarships on Student Success

Chapter 1: Introduction

Understanding how to help students succeed has always been an important goal in education. Research that contributes to this goal comes from both the educational perspective (e.g. Tinto 2006; Kahu and Nelson 2018) and the data-driven perspective (e.g. Kabra and Bichkar 2011; D’Amico and Dika 2013). In recent years, Oregon State University (OSU) has made a number of new efforts to increase student success.

For example, the Undergraduate Student Success Initiative (USSI) at OSU has been tracking and collecting relevant data that may be useful for designing strategic plans, decision making, and the evaluation of current student success strategies. Fig. 1.1 shows a screenshot of one of the many summaries available for these data on the USSI webpage. Graphics such as these allow us to observe current trends in retention and graduation rates using data aggregated at the university level or separated by colleges or demographics of interest.

Though these visuals may be useful for finding general patterns in the data, they typically are not sufficient for properly detecting and quantifying the presence and magnitude of any underlying trends. Therefore, they offer limited insight on the true structures of the data which are needed to answer more relevant questions. One such question is: *What is the role that financial aid plays on student success?* While this question is straightforward, the answer is not trivial as it must consider several aspects. For instance, although it is important to determine the impact that different amounts of financial aid will have on student success, it is also important to determine how such effects will vary across different demographics in particular for those corresponding to vulnerable groups. In other words, it is important to answer the more specific question: *How much financial aid makes an impact and for whom?*

SSI0100 Metrics - Student Success Initiative Summary View



Figure 1.1: Screenshot from the Undergraduate Student Success Initiative (USSI) summary page.

Further support for answering such questions comes from the OSU Board of Trustees 2020 student success goals for the OSU community ¹. These include:

- Raising the *six-year graduation* rate from 63.1 to 70.0 percent for all students
- Raising the *first-year retention* rate for all students from 83.8 to 90.0 percent
- Closing the achievement gap

With these goals in mind, the College of Science (COS) at OSU initiated a project to study the impact that gift aid has on retention and graduation rates of students. Gift aid is defined

¹<https://leadership.oregonstate.edu/provost/undergraduate-student-success-initiative>

as merit- or need-based aid that students do not have to repay. In collaboration with the Financial Aid Office and the Office of Institutional Reporting and Analytics (IRA) at OSU, data were collected on three cohorts of students, entering in the 2011/12, 2012/13, and 2013/14 academic years. These data include information on first-year retention, six-year graduation, gift aid of various types, tuition status, high school GPA, and more. Broadly, the available data can be categorized as financial, academic, and demographical data on students.

Student populations that are of key interest to the administrative bodies involved are *first-time full-time (FTFT) freshmen* and *junior transfer* students. First-time full-time freshmen include those students whose number of credit hours completed, 45 or less, places them in the freshmen category and they have enrolled at the institution with a full-time academic load, 12 or more credit hours, for the first time ever. Junior transfer students are those students who transfer from another institution with at least 90 credits completed.

Freshmen students with little or no prior college experience are considered a vulnerable student population (Ameri et al. 2016). For this reason, and due to the interests of the COS, our research focuses on this student population. The final data set contains about 200 variables for over 9,000 FTFT freshmen students. Specifically, the response variables that we are most interested in modeling are defined as:

- *first-year retention* - The re-enrollment of a student after their first year at the university (binary)
- *six-year graduation* - A student graduating within six years of first enrolling at OSU as a freshmen (binary)

The respective covariates whose impacts we are most interested in are *total first year gift aid*, defined as:

$$\begin{aligned} \mathbf{x}_{\text{FYgift}} = & \text{Amount received from Pell Grants (1st year)} + \\ & \text{Federal Supplemental Educational Opportunity Grant (1st year)} + \\ & \text{State External Gift Aid (1st year)} + \text{Funded Gift Aid (1st year)} + \\ & \text{Unfunded Gift Aid (1st year)}. \end{aligned}$$

and *total gift aid over first four years*, defined as:

$$\begin{aligned} \mathbf{x}_{\text{Tgift}} = & \text{Amount received from Pell Grants (1st year)} + \\ & \text{Federal Supplemental Educational Opportunity Grant (1st year)} + \\ & \text{State External Gift Aid (1st year)} + \text{Funded Gift Aid (1st year)} + \\ & \text{Unfunded Gift Aid (1st year)} + \\ & \text{Amount received from Pell Grants (2nd year)} + \dots + \text{Unfunded Gift Aid (2nd year)} \\ & \text{Amount received from Pell Grants (3rd year)} + \dots + \text{Unfunded Gift Aid (3rd year)} \\ & \text{Amount received from Pell Grants (4th year)} + \dots + \text{Unfunded Gift Aid (4th year)} \end{aligned}$$

We denote the probabilities that we are most interested in modeling as:

$$p_{\text{ret}}(\mathbf{X}) = P(\text{first-year retention} \mid \mathbf{X}) \quad \text{and} \quad p_{\text{grad}}(\mathbf{X}) = P(\text{six-year graduation} \mid \mathbf{X}),$$

where \mathbf{X} is a matrix containing all feature vectors, $\mathbf{x}_1, \dots, \mathbf{x}_p$. In addition to first-year retention, we also look into second- and third-year retention, defined as whether the student enrolled in the university for a third and fourth year, respectively. Other covariates of primary interest for our research include *binary gender*: male or female, *students of color*:

students that are not White or International, *Pell eligibility*: students that are Pell eligible, *residency*: Oregon resident or non-resident, *first generation*: students whose parents did not attend college, and *financial need*: none, low, medium, high.

The goals of our research are two-fold: (1) to assess the effect of $\mathbf{x}_{\text{FYgift}}$ and $\mathbf{x}_{\text{Tgift}}$ on $p_{\text{ret}}(\mathbf{X})$ and $p_{\text{grad}}(\mathbf{X})$, respectively, in the presence of these other covariates, and to (2) develop models for predicting $p_{\text{ret}}(\mathbf{X})$. These two goals are the inferential and predictive components of our case study, respectively. Due to the two-fold nature of our goals, our research includes detailed data summaries and descriptive statistics, as well as a flexible class of models that allows us to characterize the relationships between variables of interest, and obtain estimates of the probabilities of retention and graduation for students in different demographic groups, based on the received amount of financial aid.

While working on the predictive component of our case study a few technical issues were brought to light. Desertion is not a common event and any institution would hope for this to be true of their student populations. However, this does present a challenge to predicting when students will desert because of the small amount of data available on students with this outcome. This issue is termed *imbalance* and we define it more thoroughly in Chapter 3.

The response variables that we are interested in are skewed in the sense that desertion and failure to graduate in six years are far less commonly observed than their positive converses. A classification model that predicts the most frequently observed outcome every time regardless of the covariate vector would arbitrarily achieve a high accuracy. This could mislead us to assume that our models have strong predictive capabilities if we do not assess the class-specific accuracies. Upon doing so we find that the minority class accuracy is poor but the majority class accuracy is high. These two classes the minority and majority classes correspond to students who deserted and returned, respectively.

In addition to the issue of imbalance we also found that many of the covariates in our dataset had little to no variation between students who deserted or did not graduate and that were retained or graduated. This is known as *overlap* in the feature space and it presents a challenge to predicting student outcomes. This may be due to the population of students that we are studying. The students in our dataset are made up of first-time full-time (FTFT) freshmen in the early stages of their studies. These students are likely to have similar attributes since they are coming out of a K-12 system that has similar standards and goals across the country.

Classification models aim to find patterns in the feature space that are indicative of an outcome. When these patterns are similar regardless of the outcome of an observation, incorrect classifications are more likely to occur. Though our outcome of interest is the positive class, retained, the practical goal of our research is to identify students that have a low chance of success and strategically award aid in a manner that will increase their chances of succeeding. Therefore, we must carefully watch out for both false positives, predicting a student will be retained when they actually deserted, and false negatives, predicting a student will desert when they actually returned.

Imbalanced learning is a well-developed and still-growing area of research (e.g. Krawczyk 2016; A. Fernández et al. 2018). The most popular technique for correcting imbalance to date is the Synthetic Minority Oversampling TEchnique (SMOTE) introduced by Chawla et al. (2002). In order to tackle the predictive component of our practical research objective we must also assess solutions to these two data difficulties. Though research concerning the issue of overlap independently exists (e.g. Xiong et al. 2010; Oh 2011), solutions that deal with both imbalance and overlap have become more common (e.g. Denil and Trappenberg 2010; Borsos et al. 2018; Z. Li et al. 2021).

After reviewing the current solutions to the issue of imbalance and overlap, we deemed it necessary to develop a novel method that tackles these two issues in the presence of other data

difficulties, such as missing or mixed-type data. Our research includes a simulation study that connects various oversampling methods and predictive performance as data characteristics change. This simulation study assisted us in characterizing the problem of imbalance and overlap. Our insights and findings are implemented in a novel algorithm called The Strategic SMOTE (S-SMOTE). We discuss the characteristics of this algorithm and return to the OSU dataset to find that we are able to achieve better predictive performance in certain cases.

In Chapter 2 we present our inferential findings from the case study at OSU. In Chapter 3 we begin a preliminary discussion on the predictive component of the research and formally define and explain the issues of imbalance and overlap. In Chapter 4 we discuss the simulation study and the creation of S-SMOTE. Finally in Chapter 5, we provide example applications of S-SMOTE and make comparisons between it and SMOTE. In Chapter 6 we provide concluding remarks and summarize our findings.

Chapter 2: Modeling Student Success: A Case Study at Oregon State University (OSU)

In our review of the literature, we found that solutions to the issue of student success and, more generally, discussions about student success, come from at least two perspectives. Research from the education-focused perspective often looks at the problem from the viewpoint of educators and psychologists and aims to produce fresh educational and psychological insight on how to improve student success in higher education. (e.g. Perna and Thomas 2008; Núñez-Peña et al. 2013; Hwang et al. 2021).

Such research has produced a variety of literature, ranging from the seminal textbook *Leaving College: Rethinking the Causes and Cures of Student Attrition* (Tinto 1987), which specifically focuses on student retention, to more recent literature by Kahu and Nelson (2018) which discusses the intricate and interrelated components that influence student success. Though no single definition of student success dominates the literature, most definitions pertain to the continued and successful progress of students in a program, a university, or higher education in general, over a certain amount of time (Mullin 2012).

Research from the data-focused perspective often views data as a key informant of student success and aims to find patterns in the data that shed light on how to improve student success metrics (e.g. Jones-White et al. 2010; Kabra and Bichkar 2011; Natek and Zwilling 2014). We believe that the literature from this research perspective will best inform our research and, therefore, we will not review further literature from the former perspective, but we refer the curious reader to the references that we have already given as a starting point.

We found that most of the literature that discusses student success from a data-focused standpoint centers around the development of models that are commonly used in the field of statistics and machine learning. Examples include the articles referenced in the previous paragraph and those that we will discuss next. Based on our review, and that of Zeineddine et al. (2021), the most popular models applied in the literature include logistic regression, decision trees, random forests, and neural networks. Often, a combination of these and other methods are used and their results are compared. We discuss some of the literature that led to these conclusions next.

Using linear and logistic regression models D'Amico and Dika (2013) discovered factors that influence the first-year GPAs and second-year retention statuses of first-generation college students and non-first-generation college students, separately. In general, a *first-generation* college student is defined as a student whose parents never attended college (e.g. Terenzini et al. 1996; Ishitani 2006). Logistic regression, decision trees, and neural networks were all used by Raju and Schumacker (2015), where first-semester and high-school GPA were found to be some of the most important factors influencing retention leading to graduation in FTFT freshmen. Logistic regression and machine learning techniques were also applied by C. H. Yu et al. (2010) and Chatterjee et al. (2018) as they studied the impact of a variety of factors on the retention and graduation of FTFT freshmen.

Other insightful, but less commonly used, approaches to understanding student success include those undertaken by Goldrick-Rab et al. (2012), Miller and Lesik (2014), Bettinger (2015), Ameri et al. (2016), and Zeineddine et al. (2021). An experimental design approach was used by Goldrick-Rab et al. (2012) to assess the effect of a need-based college grant on first-year retention. They found that students who were randomly assigned the grant were more likely to return, earned more credits, and had slightly better grades. This study establishes a causal relationship between receiving need-based aid and first-year retention.

Survival analysis models were developed by Miller and Lesik (2014) who found that partic-

ipation in a first-year seminar impacted the first-year retention of students, but its efficacy dropped after their second-year. A survival analysis framework was also used by Ameri et al. (2016) who developed a time-dependent survival model which was able to estimate the semester in which a student would drop out with over 70% accuracy regardless of the semester.

Automatic machine learning (AutoML) was used by Zeineddine et al. (2021) to predict the first-semester GPA of first-time students, using data known at the time of admission. AutoML searches for the most optimal combination of algorithms and hyper-parameters that produce the best predictions (Tuggener et al. 2019). The winning model was an ensemble classifier, that aggregated predictions from neural networks, logistic regression, decision trees, and other machine learning algorithms to predict the GPA of a student.

After reviewing the literature, we identified a need for research into the specific impacts of total gift aid. Much of the literature implements such monetary information in some form, but we found that the effects of total aid are not carefully and directly studied in much of the literature. Rather it is often lumped in with many other covariates possibly masking its significance or changes in its impacts over time. Therefore, we aim to develop simple models which can be used to make inference on the impact of total gift aid more specifically. This will produce useful and fresh insight into the problem of student success from a data-driven perspective.

We begin by looking at data summaries and descriptive statistics before moving on to modeling. These summaries are helpful for condensing relevant information and revealing important patterns and features that may be related to some of our questions of interest. The results of this exploratory analysis will help to guide us when we get to modeling.

2.1 OSU Data Summaries

First we assess enrollment numbers over time for each cohort that we have data on. Table 2.1 gives enrollment numbers for the initial year and four subsequent years for each cohort. Values in parentheses indicate the percentage of students that enrolled in the subsequent year. The last column gives the number of students that graduated within six years for each cohort.

Cohort	Enroll. yr. 1	Enroll. yr. 2	Enroll. yr. 3	Enroll. yr. 4	Grad. 6 yr.
2011-2012	3154 (100%)	2648 (84.0%)	2415 (76.6%)	2245 (71.2%)	2090 (66.3%)
2012-2013	3101 (100%)	2627 (84.7%)	2406 (77.6%)	2271 (73.2%)	2100 (67.7%)
2013-2014	3328 (100%)	2798 (84.1%)	2559 (76.9%)	2359 (70.9%)	2258 (67.8%)
Total	9583 (100%)	8073 (84.2%)	7380 (77.0%)	6875 (71.7%)	6448 (67.3%)

Table 2.1: Enrollment numbers for each cohort over their first four years at OSU, and final six-year graduation rates. Numbers in parentheses give the percent of students, out of all initially enrolled, that continued through in subsequent years. The last row shows the totals over the three cohorts.

For example, 3154 students entered in the 2011-2012 academic year, from whom 2648 (about 84%) continued on to their second year. In the third year, only 2415 students (76.6% of those initially enrolled) continued their education, and so on. In the last cell, we have that 2090 out of the 3154 students initially enrolled for that cohort achieved graduation within six years at OSU. The very last row in the table provides the total values for enrollment and graduation obtained by adding the numbers over the three cohorts.

Although Table 2.1 depicts a clear and consistent decline in the enrollment numbers throughout the years (enrollment drops to about 84% on year 2, 77% on year 3 and 71% on year 4), these values should not be directly interpreted as “retention rates.” In order to obtain the actual retention rates, we need to adjust the values so they account for the enrollment in the previous year and not the starting year of the corresponding cohort. By doing so, contrary to the enrollment downward trend depicted in Table 2.1, the retention rates can

either increase or decrease depending on the number of students that are actually retained from one year to the next. The actual retention rates are shown in Table 2.2, where the first column “1st yr. ret.” displays the proportion of students that were retained by the university going from the first year into the second year in each cohort, the second column “2nd yr. ret.” displays the proportion of students that were retained by the university going from the second year into the third year in each cohort, and so on. For instance, looking at cohort AY2011-2012, we find that the *first year retention*, that is, the proportion of student in year one that continued their education on year two is 84.0%, or equivalently, 2648/3154. Similarly, the *second year retention* for the same cohort was 91.2%, or simply 2415/2648. The last column in this table shows the graduation rates with respect to the enrollment in year four for the corresponding cohorts.

Cohort	1st yr. ret.	2nd yr. ret.	3rd yr. ret	Grad wrt EY4
2011-2012	84.0%	91.2%	93.0%	93.1%
2012-2013	84.7%	91.6%	94.4%	92.5%
2013-2014	84.1%	91.5%	92.2%	95.7%
Overall	84.2%	91.4%	93.2%	93.8%

Table 2.2: Retention rates and graduation rates with respect to enrollment in year four per cohort. The overall values shown in the last row are based on the totals included in Table 2.1.

There are a few things that is important to note in these tables. In Table 2.2 we observe that in all three cohorts the first year retention rates are noticeably lower than the retention rates in years two and three. This suggests students that complete their first year of college and enroll into their second year are more likely to remain at school and continue their education, than they were when starting their first year of college. Or in other words, that we will observe a larger proportion of students dropping out from college in their first year, than in year two or three. In Table 2.1 total enrollment decreased about 16% in year two, 7% in year three and 5% in year four. This decrease shows that while the proportion of students who drop out from college declines year-after-year, there is still a drop of about 4 to 5% between the enrollment in the fourth year and the number of students that achieve

graduation within six years. These numbers suggest that even after four years of college there is still a good chance that an important proportion of students will not complete their education. Finally, the values and trends across the cohorts in both tables are remarkably similar suggesting that these patterns are fairly consistent with little to none cohort-to-cohort variation.

2.1.1 Evaluating Student Success and Demographics

In terms of demographics, Table 2.3 shows the proportions of students in each group with respect to the initial enrollment for each cohort. Information about gender, race, and first-generation status were self-reported by the students when applying. Categorization of this information is done following the USSI guidelines. In the table, we observe the breakups of these groups across cohorts are very similar. For gender we have a fairly balanced split of student in all three cohorts with male students showing a slightly higher proportion. We also observe that a clear majority of the students in each cohort are in-state White students and that only 30% or less correspond to Pell eligible or first-generation students in each cohort.

Cohort	Gender		Stnt. of col.		Pell elig.		State Res.		First-gen.	
	Female	Male	No	Yes	No	Yes	No	Yes	No	Yes
2011-2012	48.2%	51.8%	71.8%	28.2%	67.8%	32.2%	24.8%	75.2%	81.3%	18.7%
2012-2013	47.5%	52.5%	70.7%	29.3%	68.7%	31.3%	23.3%	76.7%	75.1%	24.9%
2013-2014	48.3%	51.7%	70.0%	30.0%	70.9%	29.1%	24.2%	75.8%	77.4%	22.6%

Table 2.3: Proportion of students in each demographic groups with respect to the initial enrollment of the corresponding cohorts.

The retention and graduation rates for each one of these groups are shown in Table 2.4. Looking at the retention rates, we observe small differences, less than 2%, for gender, student of color and state of residency, in contrast with the larger differences between 4 to 6%, observed for the groups determined by Pell eligibility and first-generation status. These differences are fairly consistent across the years and between cohorts. Notice than even though the retention rates for all categories seem to improve over time, Pell eligible and first gener-

ation students exhibit consistently lower retention rates than their respective counterparts for all years. Looking at graduation rates, we observe important differences in all categories. But again, the larger gaps can be found for Pell eligible and first-generation students, where the observed graduation rates go as low as 56.1% for Pell eligible students and 58.6% for first-generation students. Once again we observe that the overall patterns are pretty similar across the different cohorts suggesting a small cohort-to-cohort variation.

	Gender		Stnt. of col.		Pell elig.		State Res.		First gen.	
	Female	Male	No	Yes	No	Yes	No	Yes	No	Yes
2011-2012										
1st year ret.	85.3%	82.8%	83.3%	85.6%	86.4%	78.8%	84.7%	83.7%	84.8%	80.5%
2nd year ret.	91.1%	91.3%	91.5%	90.4%	92.4%	88.5%	91.1%	91.2%	91.9%	88.0%
3rd year ret.	93.0%	92.9%	92.8%	93.3%	93.4%	91.8%	95.7%	92.1%	92.8%	93.5%
Grad. 6 yr.	70.1%	62.7%	66.4%	65.8%	71.1%	56.1%	70.0%	65.2%	68.0%	58.7%
2012-2013										
1st year ret.	84.8%	84.6%	84.5%	85.1%	86.2%	81.4%	83.7%	85.0%	86.7%	78.6%
2nd year ret.	93.0%	90.3%	91.5%	91.9%	93.8%	86.5%	92.1%	91.4%	92.5%	88.4%
3rd year ret.	94.1%	94.7%	94.3%	94.5%	94.8%	93.3%	94.2%	94.4%	93.9%	96.1%
Grad. 6 yr.	71.4%	64.3%	68.4%	66.1%	71.7%	59.1%	66.1%	68.2%	70.2%	60.3%
2013-2014										
1st year ret.	85.1%	83.1%	84.4%	83.4%	86.4%	78.3%	82.3%	84.7%	85.8%	78.1%
2nd year ret.	91.7%	91.2%	92.5%	88.9%	92.1%	89.7%	89.9%	91.9%	91.9%	88.9%
3rd year ret.	92.5%	91.9%	91.7%	93.4%	93.4%	88.9%	93.1%	91.9%	92.8%	89.8%
Grad. 6 yr.	71.7%	64.2%	68.8%	65.7%	72.3%	57.0%	66.4%	68.3%	70.5%	58.6%

Table 2.4: Retention and graduation rates by demographics in each cohort

So far, the biggest takeaways from these summaries are first, that Pell eligible and first-generation students are the most vulnerable populations with respect to student success, and second, that first-year retention and six-year graduation rates need to be improved. However, we have not yet considered the financial component of the problem. Next, we will focus on financial need as a potentially important factor for determining student success.

2.1.2 Understanding Student Success with Considerations to Financial Need and Demographics

The Office of Financial Aid and Scholarships considers four distinct categories for the severity of a student’s financial need. If their expected family contribution (EFC) is greater than or equal to their budget estimate, then their financial need level is “No Need”. If their EFC is less than 33% of their budget estimate, then they are “High Need”. If their EFC is 34-66% of their budget estimate, then they are “Medium Need”, and if their EFC is 67-99% of their budget estimate, then they are “Low Need”. If a student’s Free Application for Federal Student Aid (FAFSA) or budget is missing, then they are marked as “Unknown Need” or “No Budget”, respectively.

In comparison, a student’s Pell-eligibility status is a binary yes or no indicator of need determined using a student’s Cost of Attendance (COA) and their EFC. COA is a more general estimate of how much a student will need in order to attend college. Both financial need and Pell-eligibility capture information on costs not covered by EFC, therefore, we expect these variables to be dependent. However given the motivations behind this research and the many administrative bodies interested in its results we will include both financial need and Pell-eligibility in our analyses whenever independence is not an issue. This will allow both metrics of student need to be evaluated.

Table 2.5 summarizes the actual number of students in each financial need category for the first four years of college by cohort. The last column contains the total enrollment for each year/cohort combination as a reference and the percentages in parentheses indicate the proportion of students that received gift aid in the corresponding category. The numbers within brackets in the last column indicate the proportion of students with financial need for the respective year-cohort combination.

Year 1					
Cohort	High (% aid)	Medium (% aid)	Low (% aid)	None (% aid)	Enroll. [% need]
2011-2012	1189 (95.5%)	449 (64.5%)	257 (59.5%)	687 (60.5%)	3154 [60.1%]
2012-2013	1186 (95.4%)	431 (60.1%)	281 (66.2%)	713 (60.31%)	3101 [61.2%]
2013-2014	1194 (95.8%)	441 (52.8%)	335 (53.7%)	868 (46.2%)	3328 [59.2%]
Year 2					
Cohort	High (% aid)	Medium (% aid)	Low (% aid)	None (% aid)	Enroll. [% need]
2011-2012	1138 (78.4%)	347 (53.0%)	237 (51.1%)	438 (49.5%)	2648 [65.0%]
2012-2013	1064 (82.0%)	378 (53.4%)	242 (52.9%)	497 (43.2%)	2627 [64.1%]
2013-2014	1040 (78.2%)	369 (45.0%)	239 (46.0%)	546 (37.7%)	2798 [58.9%]
Year 3					
Cohort	High (% aid)	Medium (% aid)	Low (% aid)	None (% aid)	Enroll. [% need]
2011-2012	952 (84.9%)	321 (53.9%)	185 (47.6%)	382 (52.1%)	2415 [60.3%]
2012-2013	905 (82.5%)	309 (52.4%)	217 (52.1%)	369 (46.6%)	2406 [59.5%]
2013-2014	888 (83.7%)	315 (53.3%)	211 (45.5%)	424 (42.2%)	2559 [55.3%]
Year 4					
Cohort	High (% aid)	Medium (% aid)	Low (% aid)	None (% aid)	Enroll. [% need]
2011-2012	834 (87.1%)	265 (56.6%)	166 (54.2%)	321 (52.0%)	2245 [56.3%]
2012-2013	797 (86.3%)	308 (61.4%)	160 (51.3%)	332 (47.0%)	2271 [55.7%]
2013-2014	762 (83.3%)	302 (49.0%)	172 (48.3%)	355 (38.9%)	2359 [52.4%]

Table 2.5: Number of students in high, medium, low, and no financial need and enrollments per cohort for the first four years. The numbers in parentheses indicate the percentage of students that received some form of gift aid in each group and the numbers within brackets indicate the proportion of students with financial need for the respective year-cohort combination

For example, 1189 students had high financial need in their first year for the cohort 2011-2012. Of these students, only 1,136, about 95.5%, received gift aid. Furthermore, only 51 of the students withdrew from the university: a withdrawal rate of less than 5%. For students in the 2011-2012 cohort with medium financial need, 64.5% received gift aid and about 22% withdrew. Thus, retention seems to be impacted by both the financial status of students and whether or not students receive gift aid. However, the relationship based on the numbers alone is unclear. This motivates the need to examine the relationships using more advanced modeling. We note that about 60% of the student have some level of financial need during their first year, and for the most part, the proportion of students with financial need fluctuates between 55-65%. We also observe that a large proportion of students with high financial need receive gift aid across the years, with this proportion being as high as 95% on year 1. However, the decrease in the number of students in each category across

years suggests that financial need might be a relevant factor for student retention.

With respect to demographics, Table 2.6 shows the percentages of students in each demographic group, by financial need status during Year 1, for each cohort. Regardless of need, we observe some differences between all demographic groups. The most obvious differences are observed for students with high financial need, however. For these students, we find that the most vulnerable are those who are females, students of color, Pell eligible, in-state and first-generation students. Among these, the two groups that concentrate over 50% of their populations in this category are Pell-eligible and first-generation students. Note that these differences are consistent across the cohorts.

This table also shows that there is a strong relationship between Pell-eligibility and financial need, as we indicated earlier. This can be seen by noting that there is decrease in the proportion in each financial need category as we move from high to low need for non-Pell-eligible students but we see an increase for Pell-eligible students. However, for students who are *not* Pell-eligible, financial need gives a more intricate description of their financial need.

2.1.3 Understanding Whether Scholarships Effect Student Success

Determining the effect that gift aid has on student success is difficult. We have seen in previous tables that a majority of students have some level of financial need, and out of the students with a high level of financial need, a vast majority receives some form of financial aid. This situation leaves little room in the data to make meaningful comparisons when controlling for several factors including potential confounding variables. To address this issue and facilitate any comparisons, we aggregate financial need information as follows: for each year we combine the levels *high need* and *medium need* into a level called *more severe need*, and the levels *low need* and *no-need* into *less severe need*. Extending the notion of financial need over the course of several years is more difficult. We define *more severe* if a

Cohort 2011-2012										
Need year 1	Gender		Stnt. of col.		Pell elig.		State Res.		First gen.	
	Female	Male	No	Yes	No	Yes	No	Yes	No	Yes
High	39.6%	35.9%	33.6%	48.1%	9.1%	97.7%	28.1%	40.9%	31.5%	64.5%
Medium	14.4%	14.1%	14.5%	13.5%	21.0%	0.0%	15.0%	14.0%	15.0%	10.7%
Low	8.1%	8.2%	9.0%	6.1%	12.0%	0.0%	8.2%	8.1%	9.0%	4.6%
None	22.5%	21.1%	25.0%	13.6%	32.2%	0.0%	20.9%	22.1%	24.3%	10.7%

Cohort 2012-2013										
Need year 1	Gender		Stnt. of col.		Pell elig.		State Res.		First gen.	
	Female	Male	No	Yes	No	Yes	No	Yes	No	Yes
High	39.7%	36.9%	33.5%	49.6%	11.1%	97.6%	25.9%	42.0%	30.0%	63.0%
Medium	14.3%	13.5%	14.6%	12.2%	20.2%	0.0%	15.4%	13.5%	14.4%	12.5%
Low	10.2%	8.0%	10.6%	5.2%	13.2%	0.0%	10.8%	8.5%	10.0%	6.4%
None	22.7%	23.2%	26.0%	15.6%	33.5%	0.0%	24.1%	22.7%	27.6%	9.2%

Cohort 2013-2014										
Need year 1	Gender		Stnt. of col.		Pell elig.		State Res.		First gen.	
	Female	Male	No	Yes	No	Yes	No	Yes	No	Yes
High	38.1%	33.8%	31.4%	46.3%	9.5%	99.9%	27.0%	38.7%	26.9%	66.5%
Medium	13.6%	12.9%	14.1%	11.2%	18.7%	0.0%	13.8%	13.1%	13.5%	12.2%
Low	10.0%	10.2%	10.9%	8.1%	14.2%	0.0%	11.2%	9.7%	11.5%	5.2%
None	27.1%	25.1%	29.9%	17.2%	36.8%	0.0%	25.4%	26.3%	30.3%	11.7%

Table 2.6: Percentages of students with high, medium, low, or no financial need during their first year by demographic groups for each cohort. Students with an unknown need status or no budget are not included in the table.

given student has high or medium need for 2 or more years, and as *less severe* if a given student is in the low need or no-need category for 3 or more years. Note that by using these definitions it is possible for some students to have an undetermined status for their financial need. For instance, if a given student is identified as medium need for one year, low need for another year, and for the other two years there are no records, then the financial need of that student is not flagged as more-severe nor less-severe, and their status remains unknown. While this aggregation is arbitrary, it facilitates the comparisons and allow us to extract relevant information that distinctly separates students that face high or moderate levels of financial need during their college experience from those that do not. Based on these definitions, the two panels of Table 2.7 summarize information for first year retention and graduation rates in each cohort. On the left panel, we find the success rates based on the corresponding levels of financial need. We observe consistently higher proportion of retention and graduation among those students with less-severe financial need than those in the more

severe category. These differences are substantially larger when comparing graduation rates and remain largely unchanged across cohorts.

Cohort	Fin. need		Cohort	Gift aid	
2011-2012	less-sev	more-sev	2011-2012	below med.	above med.
1st year ret.	87.0%	80.3%	1st year ret.	74.8%	82.3%
Grad. 6 yr.	86.1%	65.6%	Grad. 6 yr.	47.5%	73.4%
2012-2013	less-sev	more-sev	2012-2013	below med.	above med.
1st year ret.	86.7%	82.4%	1st year ret.	76.8%	84.3%
Grad. 6 yr.	87.8%	67.8%	Grad. 6 yr.	45.7%	76.7%
2013-2014	less-sev	more-sev	2013-2014	below med.	above med.
1st year ret.	87.1%	80.9%	1st year ret.	72.8%	84.0%
Grad. 6 yr.	84.7%	66.7%	Grad. 6 yr.	48.8%	72.5%

Table 2.7: First year retention and graduation rates by demographics for students in the cohort 2011-2012

On the right panel, we look only at students with more-severe financial need and compare the students retention and graduation rates based on the amount of gift aid received by them. On the left side we have the success rates for the 50% of students that receive less than the median amount of gift aid, and on the right side the observed rates for the students that receive more than the median amount. The median gift aid for the cohorts AY2011-2012, AY2012-2013, and AY2013-2014 are 2,100, 3,500 and 3,000 dollars, respectively. Note that these medians were calculated without filtering out students who received \$0 first-year aid. We observe in all cases fairly large gaps for the success rates between those students that receive less than the median gift aid and those that receive an amount exceeding the median value. These results provide a deeper insight into those presented in Table 2.2, and suggest that the chances of success for students with financial need can be largely improved based on the amount of scholarship aid that is given to them. What is unclear based on the table is how student demographics might be confounding these results.

In order to assess how the amount of financial aid affects success rates in different demographics, we can use Figure 2.1. Retention and graduation rates are further broken down by whether the awarded gift aid is more or less than the median amount for all students,

and demographics. The median amount of total first year gift aid for students in need was 7,095 dollars. The median amount of total gift aid over the first four years for students in need was 15,197 dollars. These values were again calculated including students who received \$0 for first-year or total four-year gift aid. Demographics of interest are binary gender, Pell-eligibility, whether the student is non-white and non-international, termed a student of color, Oregon residency, and first-generation status. In general, first-year retention is lower for students whose gift aid is less than the median. However, for most demographical characteristics, there are further disparities in first-year retention other than those created by aid.

In the first plot of the first row of Figure 2.1, we see that the bars are approximately the same height for males and females regardless of aid. This indicates that there are no further discrepancies in retention based on gender. However, looking at the first plot in the *second* row of Figure 2.1, the difference in graduation rates between gender categories is larger for students whose gift aid was less than the median. Males who received less than the median gift aid have an even lower retention rate. The largest combined effect of aid and demographic can be seen when considering Pell-eligibility. For students receiving more than the median amount of gift aid, those who were Pell-eligible had a retention rate 12.1% lower than those who were not. Meanwhile, for students receiving less than the median amount of gift aid, Pell-eligible students had a retention rate 41.9% less than those who were not. In summary, the differences in student success between demographic groups are less severe for students receiving more aid. This provides empirical evidence that increasing gift aid could eliminate demographical disparities in student success.

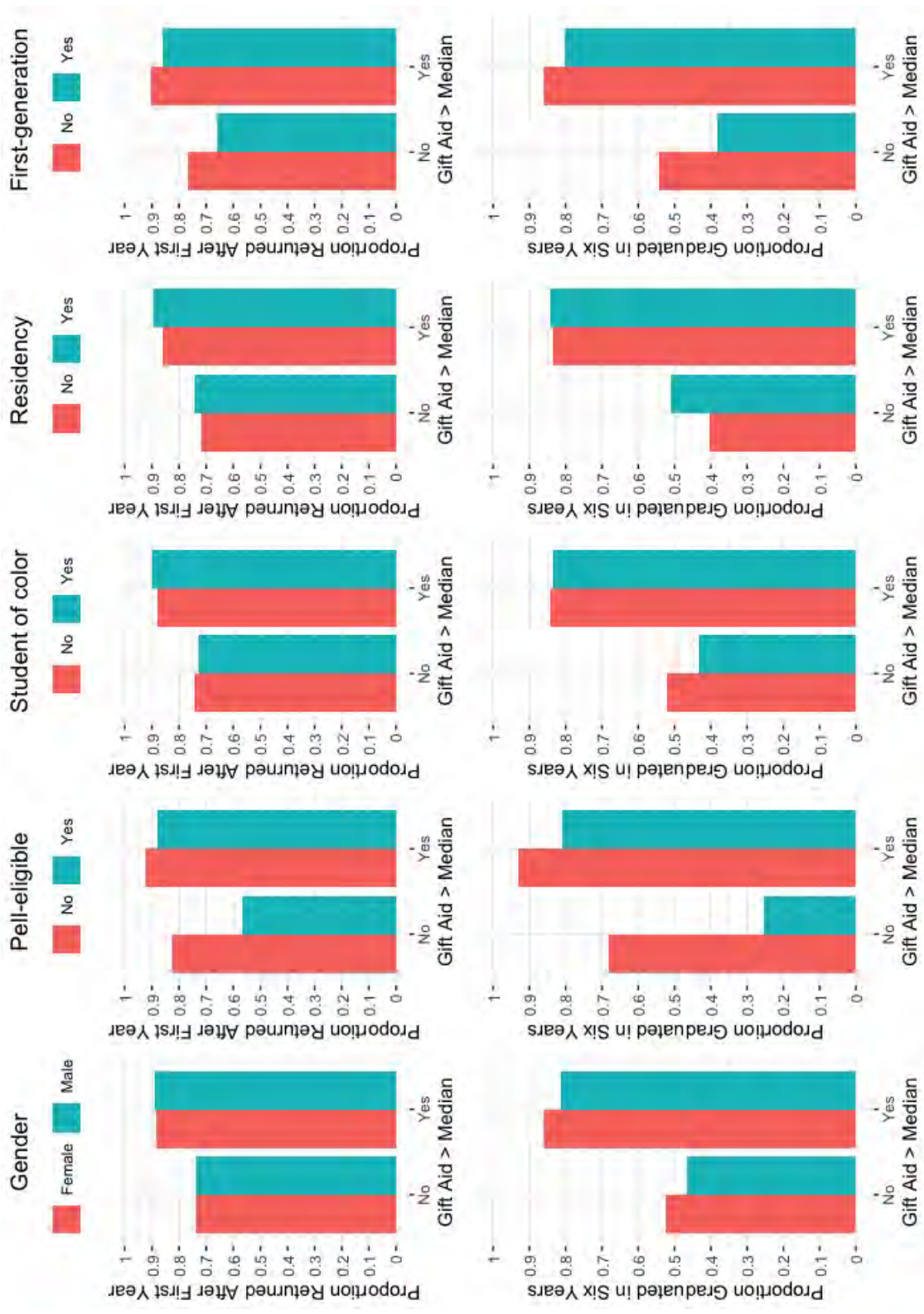


Figure 2.1: Proportion of students, of those in more severe need, who were retained or graduated. Proportions are broken down by demographics and whether their gift aid was larger than the median amount for all students in need across all cohorts. The medians were 7,095 and 15,197 dollars for first-year and total four-year gift aid, respectively. For some demographics, such as first-generation status, differences in success rates are more severe for students who also received less than the typical amount of aid.

2.1.4 Examining Student Success and Racial Group

The variable student of color broadly categorizes students as non-white and non-international, or not, which conforms to the government-set reporting standards used by administrative bodies at OSU. However, combining students across many racial categories could mask important results for students in smaller racial groups. Therefore, in this section we explore similar patterns as before in our data but using the variable race description rather than student of color. This variable gives a students self-selected race from nine categories: American-Indian or Alaska Native, Asian, Black or African-American, Hispanic, Multiple Races, Native Hawaiian or Pacific Islander, White, International Student, and Unknown.

Table 2.8 gives the percentage of students in each racial group by cohort. The majority of students are White, and the next largest groups are students who are Asian, Hispanic, or Multiple races. The other racial groups - American Indian or Alaska Native, Black or African American, Native Hawaiian or Pacific Islander, International, or Unknown - are quite underrepresented in our dataset. We provide Table 2.8 first so that this can be taken into account before generalizing the results and conclusions made in this section, and the rest of the report, to students of all races.

Cohort	Cohort 2011-2012	Cohort 2012-2013	Cohort 2013-2014	Overall
American Indian or Alaska Native	2.63%	2.48%	2.58%	2.57%
Asian	9.35%	8.64%	8.32%	8.77%
Black or African American	0.86%	1.16%	1.05%	1.02%
Hispanic	8.34%	8.77%	9.38%	8.84%
Multiple	6.88%	7.93%	8.38%	7.74%
Native Hawaiian or Pacific Islander	0.16%	0.32%	0.27%	0.25%
International	0.82%	0.52%	0.57%	0.64%
Unknown	0.44%	0.48%	0.54%	0.49%
White	70.51%	69.69%	68.90%	69.69%

Table 2.8: Percentage of students falling into each racial category and overall, by cohort.

Table 2.9 provides the number and percentage of students who were retained in consecutive years by racial group and cohort. For example, 217 students of multiple races enrolled in the 2011-2012 academic year. Of these 217 students, 181 (83.4%) returned for their second year. Of those 181 students, 163 (90.1%) returned for their second year, and so on. We see that the greatest drop in subsequent enrollment for students in all racial groups happens after their first year. However, non-trivial drops in enrollment are seen beyond the first year for students whose racial groups are Unknown, Black or African American, Native Hawaiian or Pacific Islander, American Indian or Alaska Native, or International. Upon examining students by racial groups we now find that additional improvements may be needed in second-year retention and onward for students in underrepresented racial groups.

	Enrolled 1st yr.	Enroll 2nd yr.	Enroll 3rd yr.	Enroll 4th yr.	Grad. 6 yr.
Cohort 2011-2012					
Am. Ind. or AK Nat.	83 (100%)	73 (88%)	64 (87.7%)	60 (93.8%)	55 (91.7%)
Asian	295 (100%)	264 (89.5%)	241 (91.3%)	228 (94.6%)	219 (96.1%)
Black or Af. Am.	27 (100%)	21 (77.8%)	20 (95.2%)	18 (90%)	13 (72.2%)
Hispanic	263 (100%)	218 (82.9%)	196 (89.9%)	181 (92.3%)	161 (89%)
Multiple	217 (100%)	181 (83.4%)	163 (90.1%)	151 (92.6%)	133 (88.1%)
Nat. HI or Pac. Isl.	5 (100%)	5 (100%)	5 (100%)	5 (100%)	5 (100%)
International	26 (100%)	19 (73.1%)	15 (78.9%)	11 (73.3%)	11 (100%)
Unknown	14 (100%)	13 (92.9%)	11 (84.6%)	10 (90.9%)	8 (80%)
White	2224 (100%)	1854 (83.4%)	1700 (91.7%)	1581 (93%)	1485 (93.9%)
Cohort 2012-2013					
Am. Ind. or AK Nat.	77 (100%)	70 (90.9%)	67 (95.7%)	64 (95.5%)	55 (85.9%)
Asian	268 (100%)	246 (91.8%)	237 (96.3%)	220 (92.8%)	208 (94.5%)
Black or Af. Am.	36 (100%)	30 (83.3%)	24 (80%)	23 (95.8%)	19 (82.6%)
Hispanic	272 (100%)	221 (81.2%)	203 (91.9%)	192 (94.6%)	169 (88%)
Multiple	246 (100%)	199 (80.9%)	175 (87.9%)	167 (95.4%)	146 (87.4%)
Nat. HI or Pac. Isl.	10 (100%)	8 (80%)	5 (62.5%)	5 (100%)	4 (80.0%)
International	16 (100%)	14 (87.5%)	12 (85.7%)	11 (91.7%)	10 (90.9%)
Unknown	15 (100%)	13 (86.7%)	10 (76.9%)	9 (90%)	9 (100%)
White	2161 (100%)	1826 (84.5%)	1673 (91.6%)	1579 (94.4%)	1480 (93.7%)
Cohort 2013-2014					
Am. Ind. or AK Nat.	86 (100%)	78 (90.7%)	75 (96.2%)	68 (90.7%)	67 (98.5%)
Asian	277 (100%)	244 (88.1%)	229 (93.9%)	211 (92.1%)	208 (98.6%)
Black or Af. Am.	35 (100%)	29 (82.9%)	22 (75.9%)	21 (95.5%)	19 (90.5%)
Hispanic	312 (100%)	240 (76.9%)	209 (87.1%)	202 (96.7%)	186 (92.1%)
Multiple	279 (100%)	234 (83.9%)	199 (85%)	183 (92%)	171 (93.4%)
Nat. HI or Pac. Isl.	9 (100%)	7 (77.8%)	6 (85.7%)	6 (100%)	5 (83.3%)
International	19 (100%)	14 (73.7%)	12 (85.7%)	9 (75%)	7 (77.8%)
Unknown	18 (100%)	17 (94.4%)	15 (88.2%)	14 (93.3%)	11 (78.6%)
White	2293 (100%)	1935 (84.4%)	1792 (92.6%)	1645 (91.8%)	1584 (96.3%)

Table 2.9: Retention rates of all students, by racial group and cohort, with respect to previous year enrollments.

In order to further understand the impact of aid on retention and graduation rates between racial groups, we focus on those students who have more severe need. Recall that we categorized a student as having more severe need in their first year if they had high need or medium need in their first year. Similarly, we categorized a student as having more severe need over their first four years if they had high need or medium need for 2 or more of their first four years. Figure 2.2 provides the first-year retention rates for students with more severe need in their first year. This is broken down by racial group and whether the student received more than the median amount of gift aid in their first year. The median amount of total first year gift aid for students in need was 7,095 dollars including students who received \$0 first-year aid.

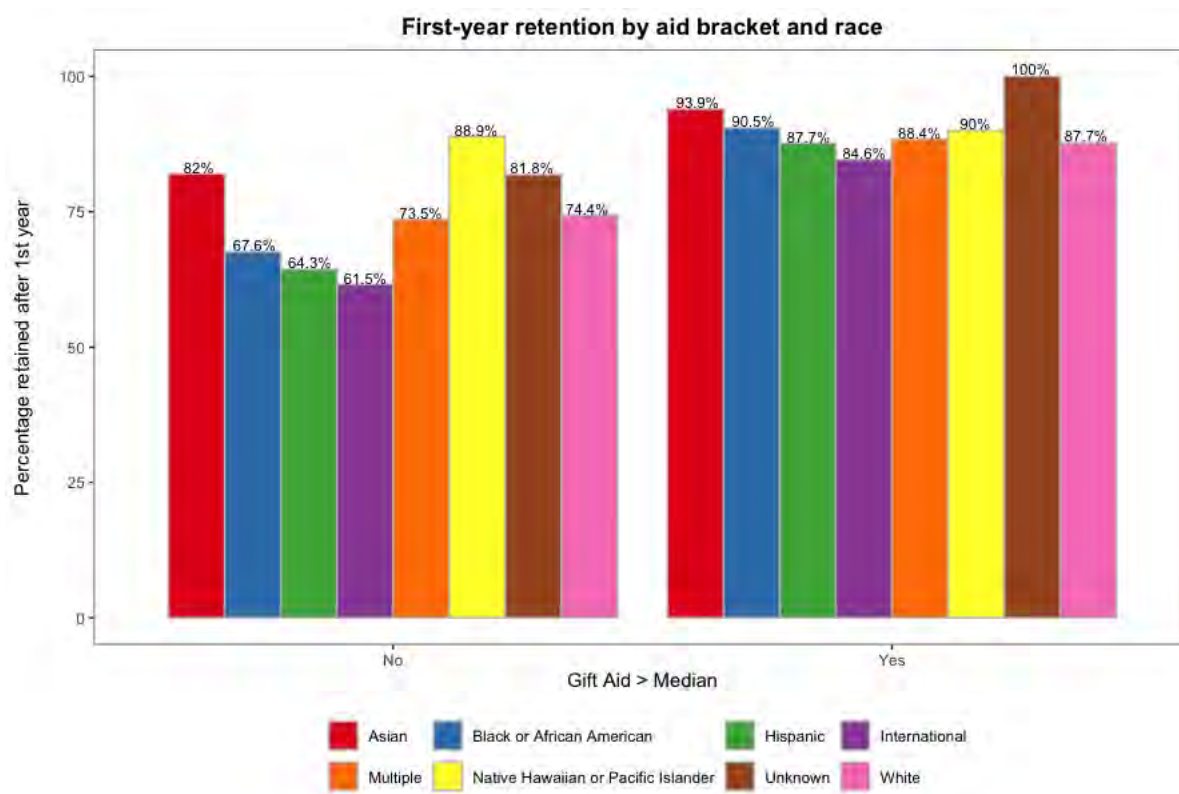


Figure 2.2: Retention rates for students with more severe need. These are broken down by racial group and whether total first-year gift aid was greater than the overall median for all students with more severe need (7,095 dollars). Lower rates observed for student receiving less aid, and decrease varies across racial groups.

Examining Figure 2.2 we see that regardless of racial group first-year retention rates are

lower for students who received less than the median amount of gift aid in their first year. Additionally, there is more variability in retention rates between racial groups for students who received less than the median amount of gift aid. The greatest change in retention rates for students who received more than the median aid amount happens between students whose racial group is Unknown ($N = 9$) and International ($N = 13$). This difference was $100\% - 84.6\% = 15.4\%$. Meanwhile, the greatest change in retention rates for students who received *less* than the median gift aid amount takes place between Native Hawaiian or Pacific Islander ($N = 9$) and International ($N = 26$) students. This difference was $88.9\% - 61.5\% = 27.4\%$. This is an increase of almost 80% in the largest difference in retention rates. Though there were very few students in these groups, if we constrain our comparisons to racial groups with better representation, we still see larger differences for students who received less than the median amount of gift aid. These results indicate that, for students with more severe need, differences in retention rates between racial groups exist, but these differences are further impacted by the amount of gift aid that students receive.

Looking *within* a given racial group, the largest decreases in retention rates between students receiving more or less than the median amount of gift aid were observed for Hispanic, International, and Black or African American students. These differences were $87.7\% - 64.3\% = 23.3\%$, $84.6\% - 61.5\% = 23.1\%$, and $90.5\% - 67.6\% = 22.9\%$, respectively. Meanwhile, Native Hawaiian or Pacific Islander, Asian, and White students saw the smallest change between aid brackets. These differences were 1.1%, 11.9%, and 13.4%, respectively.

Figure 2.3 provides the six-year graduation rates for students with more severe need, broken down by racial group and whether the student received more than the median amount of total gift aid over the first 4 years. The median amount of total gift aid over the first four years for students in need was 15,197 dollars, including \$0 aid amounts. We observe patterns that are similar to, but more severe than, those seen in Figure 2.2.

Specifically, the decrease in graduation rates for students who received less than the median

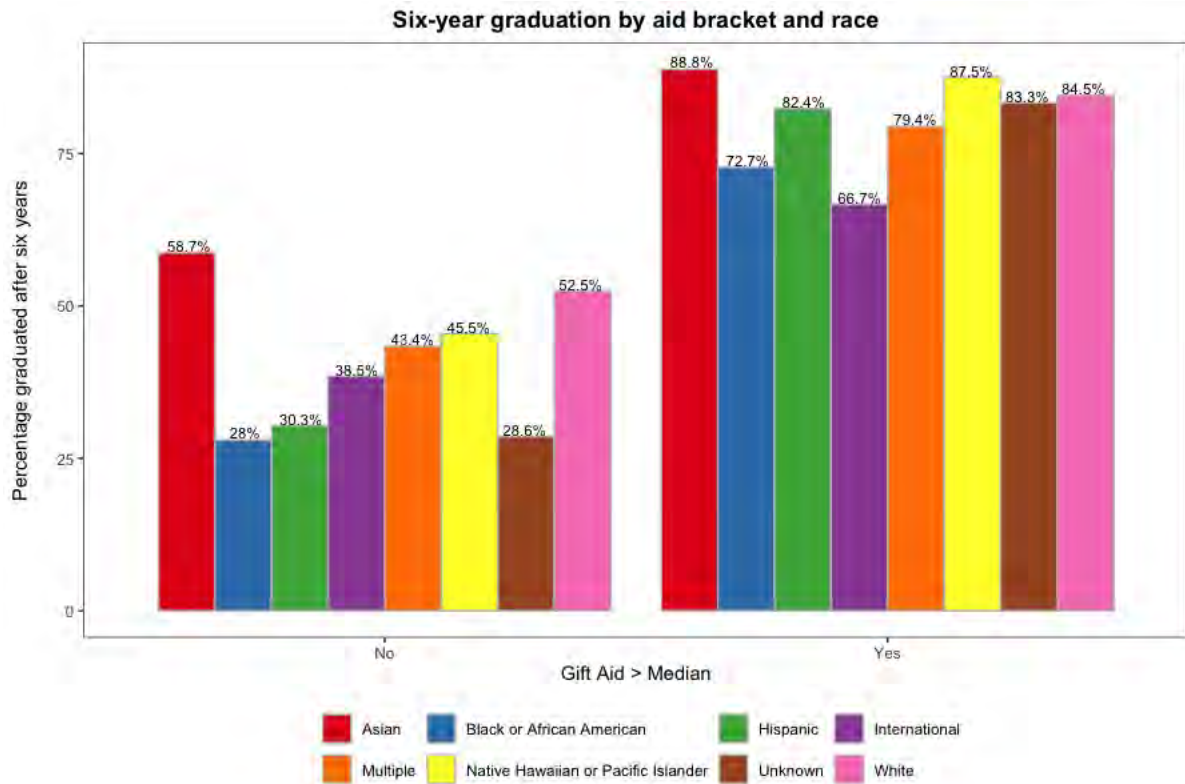


Figure 2.3: Graduation rates for students with more severe need. These are broken down by racial group and whether total gift aid over the first 4 years was greater than the overall median for all students with more severe need (18,003 dollars). Even lower rates than observed with retention are observed for student receiving less aid, and this decrease greatly varies across racial groups, more than in retention rates.

amount of gift aid is much greater regardless of racial group. The largest changes were observed for students whose racial group is Unknown, Hispanic, and Black or African American. These were 54.8%, 52.0%, and 44.7%, respectively. The smallest changes were for students whose racial group is International, Asian, and White. These were 28.2%, 30.1%, and 32.0%, respectively. However, these are still quite large in comparison to the changes in retention rates for a given racial group between those receiving more than the median amount and those that did not. These results indicate that, for students in need and within each racial group, the effect of aid on graduation rates is stronger than that on retention rates. This motivates the need to study metrics beyond the first year, as we will do in some parts of this analysis, in order to fully understand the effect of aid on student success.

2.1.5 Trends and Patterns within Demographic Groups

When analyzing large amounts of data, it is not completely uncommon to observe a pattern in grouped data that disappears or diminishes when data are aggregated. When an observed pattern in grouped data reverses or disappears after aggregating the data, this is referred to as *Simpson's Paradox* (e.g. Alin 2010). It is possible that grouping occurs in our dataset since there are a variety of demographical variables taking a variety of values (e.g. gender, race, need, residency, parent income). Further investigating patterns within some of these groups could help us to make more accurate conclusions based on the aggregated dataset.

For example, Table 2.10 provides first-year retention and six-year graduation rates by racial group. Overall 84.8% of students returned after their first year. However, after grouping the data by race this value ranges from about 74% for International students to about 93% for students of an Unknown race. The overall six-year graduation rate was 68.9% in the aggregate data, but this value ranges from 45.7% for International students to 78.6% for Asian students in the grouped data. These results indicate that the overall retention and

graduation rates do not reflect student success for all student groups with quite the same accuracy.

Race	Deserted (%)	Retained (%)
American Indian or Alaska Native	2 (22.2%)	7 (77.8%)
Asian	56 (8.6%)	594 (91.4%)
Black or African American	16 (19.5%)	66 (80.5%)
Hispanic	132 (18.4%)	585 (81.6%)
Multiple	83 (16.1%)	433 (83.9%)
Native Hawaiian or Pacific Islander	2 (11.8%)	15 (88.2%)
International	12 (26.1%)	34 (73.9%)
Unknown	2 (6.9%)	27 (93.1%)
White	679 (15.4%)	3729 (84.6%)
Overall	984 (15.2%)	5490 (84.8%)
Race	Not Graduated (%)	Graduated (%)
American Indian or Alaska Native	2 (22.2%)	7 (77.8%)
Asian	139 (21.4%)	511 (78.6%)
Black or African American	39 (47.6%)	43 (52.4%)
Hispanic	271 (37.8%)	446 (62.2%)
Multiple	189 (36.6%)	327 (63.4%)
Native Hawaiian or Pacific Islander	7 (41.2%)	10 (58.8%)
International	25 (54.3%)	21 (45.7%)
Unknown	12 (41.4%)	17 (58.6%)
White	1331 (30.2%)	3077 (69.8%)
Overall	2015 (31.1%)	4459 (68.9%)

Table 2.10: First-year retention and six-year graduation counts and proportions by racial group. Results indicate that the problem of student success is not the same across all racial groups.

Table 2.11 gives retention and graduation rates for students by severity of need. This was calculated using the aggregate data and data broken down by racial groups. Data on severity of need status over all four years was not available for any American Indian or Alaska Native students, and there were zero Native Hawaiian or Pacific Islander students with less severe need. This missing information or zero counts led to missing proportions which are represented by dashes in the table. Other groups also had students with missing data on need so there are disparities in the totals when compared to Table 2.10. This table was calculated using data only on students with data on need.

Race	No need		Need	
	Deserted	Retained	Deserted	Retained
American Indian or Alaska Native	-	-	-	-
Asian	9 (6.1%)	139 (93.9%)	46 (10%)	414 (90%)
Black or African American	1 (12.5%)	7 (87.5%)	15 (20.8%)	57 (79.2%)
Hispanic	10 (10.3%)	87 (89.7%)	121 (20.6%)	466 (79.4%)
Multiple	16 (14%)	98 (86%)	66 (17.8%)	305 (82.2%)
Native Hawaiian or Pacific Islander	1 (100%)	0 (0%)	1 (6.2%)	15 (93.8%)
International	1 (9.1%)	10 (90.9%)	11 (32.4%)	23 (67.6%)
Unknown	0 (0%)	12 (100%)	2 (11.8%)	15 (88.2%)
White	136 (9.9%)	1238 (90.1%)	511 (19.4%)	2120 (80.6%)
Overall	174 (9.9%)	1591 (90.1%)	773 (18.5%)	3415 (81.5%)
Race	No need		Need	
	Not Graduated	Graduated	Not Graduated	Graduated
American Indian or Alaska Native	-	-	-	-
Asian	3 (5.5%)	52 (94.5%)	106 (22.8%)	358 (77.2%)
Black or African American	3 (75%)	1 (25%)	27 (42.2%)	37 (57.8%)
Hispanic	6 (12%)	44 (88%)	210 (37.6%)	349 (62.4%)
Multiple	6 (11.5%)	46 (88.5%)	134 (37.3%)	225 (62.7%)
Native Hawaiian or Pacific Islander	-	-	6 (37.5%)	10 (62.5%)
International	0 (0%)	4 (100%)	19 (59.4%)	13 (40.6%)
Unknown	1 (33.3%)	2 (66.7%)	10 (58.8%)	7 (41.2%)
White	59 (9%)	595 (91%)	820 (33.4%)	1636 (66.6%)
Overall	78 (9.5%)	744 (90.5%)	1332 (33.6%)	2635 (66.4%)

Table 2.11: First-year retention and six-year graduation counts and proportions by racial group further broken down by need. Patterns in the original data were not always the same after grouping by race. Data on severity of need status over all four years was not available for any American Indian or Alaska Native students, and there were zero Native Hawaiian or Pacific Islander students with less severe need in their first two years. This missing information or zero counts led to missing proportions which are represented by dashes in the table.

In the overall data we see that the retention rate was higher than the desertion rate regardless of need. We also see that retention rate was higher for students not in need than those in need. 90.1% of student not in need were retained while this was 81.5% for students in need. Similar conclusions apply for graduation rates, with more students graduating than not regardless of need, and students not in need having a higher graduation rate. When we examine the data grouped by race, however, the relationship between need and student success is not so clear. Contrary to the overall pattern, 57.8% of Black or African American students in need graduated while 25% of those *not* in need graduated. Students in need who were International or of an Unknown race also had graduation rates below 50%. Additionally, the drop in retention and graduation rates for students in need is not the same for all racial groups. For example, the difference in retention rates between students in need and not in need is greatest for International students at 23.3%, but this was only 8.6% when the data were aggregated. These differences could be due to the low counts in these racial groups. More data would need to be obtained on students in these racial groups to further examine the relationship between need and student success for them and make comparisons to the overall trend.

Our results also show that there are variables which are more useful to classification for some groups than for others. It is desirable that a variables values can be separated into two or more subsets such that the classes of observations in each of those subsets are mostly the same then this variable. Such variables are useful for the classification task because they can be used to create decision boundaries. For example, if 90% of students in need deserted and 90% of students not in need returned, this variable would be a very clear indicator of student success. In terms of graduation, the behavior of students in need was less clear-cut for students whose race was Black or African American, International or Unknown. This is because the graduation rates for students in these racial groups who were also in need are close to 50%.

Results such as these were continually observed in the data when studying the relationship between student success and two or more demographics. These indicate that there are students whose characteristics are typical of those that succeeded based on the rest of our data, but they did not succeed. These students contribute to anti-separation in the dataset - their covariates are indicative of a certain outcome based on the rest of the data, but they did not achieve this outcome. For example, referring to Table 2.11, we saw that 93.9% of Asian students with less severe need returned after their first year, but 6.1% did not. A classifier is very likely to misclassify these few observations that break the norm. The underlying issue is that we have little data on students with these “success” characteristics who do not actually succeed and we also have little data on students with “failure” characteristics - that is, characteristics indicative of not succeeding based on patterns observed in the rest of the data, who *do* succeed.

Overall, these results indicate that our analysis of these data will be most representative of White students that are ineligible for the Pell grant, Oregon residents, non-first-generation, and less severely in need. Though we aim to find patterns in the data and generalize these to broader populations, caution should be used before generalizing these results to smaller subsets of the population or minority groups within the student population because very little data are available on these groups. There will always be students who do not follow the general patterns that we aim to discover in the data. More work would need to be done to understand factors contributing to or detracting from the success of such students.

The results presented in this section show that first-year retention and six-year graduation can be improved. However, they also show that this problem is more intricate than simply studying retention versus gift aid. The impacts of student demographics and financial need were shown and these have a non-trivial impact on retention and graduation rates. While these results are a good start at exploring the data, formal statistical models will need to be fit in order to further flush out the impacts of aid on student success, and that between

various student groups. In the next section we will discuss our modeling approach to the inferential component of the problem.

2.2 Modeling Approaches for Inference

Our two main goals for this research are to (1) construct models that can adequately describe the relationship between graduation or retention and the amount of gift aid received, while taking other variables into account, and (2) develop predictive models that can be used to determine how the predicted success of a given student changes with gift aid, and how gift aid impacts these predictions. These two goals respectively fall under two pillars of statistics, namely, inference and prediction. We will now discuss our modeling approach to the first component of this problem. We begin with a brief overview of statistical learning, and work our way towards descriptions of the methods that we will use to tackle the first of these two problems.

2.2.1 A Brief Introduction to Statistical Learning

Fundamentally, *statistical learning* is the study of the relationships between predictor variables X_1, \dots, X_p for a population, and one or more response variables Y_1, Y_2, \dots . In the simplest case, we observe the values of one quantitative or categorical response variable, Y , as well as p many predictors X_1, \dots, X_p . We assume there is a usually unknown relationship between these observed values defined as $Y = f(X_1, \dots, X_p) + \epsilon$, where ϵ denotes a random or unobserved error term *independent* of X_1, \dots, X_p .

The overarching goal of statistical learning is, therefore, to obtain an estimate, \hat{f} , of f , given data on X and Y . Doing so results in a model that takes in X as input and outputs our best guess \hat{Y} for Y . However, even if we have a perfect estimate for f in $Y = f(X) + \epsilon$,

the predicted value $\hat{Y} = \hat{f}(X)$ of Y may not equal Y , since Y also depends on ϵ , a term independent of the data. This means that our models have *reducible error*, which comes from better estimating f using X , and *irreducible error*, which cannot not reduced using information from the data, since it is independent of X . Some sources of irreducible error include unmeasured variables or unmeasurable variation in the data.

When we *train* a model, we aim to learn f using the information present in our observed dataset. Often, we use domain knowledge to assume the functional form of f (e.g. a linear equation). Then we use a procedure to estimate the *parameters* of this function in a manner that minimizes the reducible error. These are called *parametric methods*. We may also forgo any assumptions about the shape of f and work with a more general class of functions, with the goal of finding the one that most minimizes the reducible error. These are called *non-parametric methods*. Unfortunately, if care is not taken, these models can be prone to *overfitting*, where the model closely matches the observed data, but does not properly represent the true unobserved relationship between the variables.

Statistical learning problems also fall into a pair of categories: *regression* problems, wherein we measure the magnitude of a *quantitative* response variable, and *classification* problems, wherein we sort a *qualitative* response variable into several discrete classes. Though our problem of predicting student success involves understanding the probability of retention or graduation, when this probability is above a certain threshold, we will classify students as retained/graduated or deserted/did not graduate. Therefore, the underlying problem that we are solving is a classification problem. Next, we discuss how we define error for classification models and issues that can arise when we focus more on models that minimize the reducible error.

A more in-depth introduction to statistical learning can be found in the introductory-level textbook *An Introduction to Statistical Learning* (James et al. 2013), and an advanced treatment of statistical learning can be found in *Elements of Statistical Learning* (Hastie et al.

2009). Note that our discussions will focus on *supervised learning*, where the true value/label of the response variable is available at the time of training, and can be used to evaluate model performance. This is not the case with *unsupervised learning* where the goal is to learn patterns from the unlabeled data.

2.2.2 Accuracy Versus Interpretability

The *accuracy*, *loss*, or *error* of a model gives the user a sense of how well a model is able to correctly predict real observations from the training data or test data (Makridakis 1993). *Training data* refers to the data that a model is trained on, while *test data* is unseen by the model during training and used to assess the accuracy of our predictions after training. In a binary classification problem, let P and N denote the number of examples in the class of interest and its complement, respectively. We call these the *positive class* and *negative class*, respectively. The accuracy is then defined as $Accuracy = \frac{TP+TN}{P+N}$, where TP and TN denote the number of *true positives* and *true negatives*, respectively. The training error of a binary classification model can, therefore, be defined as $Error = \frac{FP+FN}{P+N}$, where FP and FN denote the number of *false positives* and *false negatives*, respectively.

The *interpretability* or *explainability* of a model can be defined as how accessible the decision making process of the model is to its users - that is, how well the model can be understood (Bibal and Fréney 2016). Models that are interpretable provide output that can be used by humans to understand which factors influenced the classifications made by the model. Alternative definitions of interpretability, and its somewhat subjective nature, are further discussed by (e.g. Bibal and Fréney 2016).

When there are a few simple covariates in the data, statistical methods like logistic regression can be easily interpreted and used to understand the effect of \mathbf{x}_{FYgift} and \mathbf{x}_{Tgift} on $p_{ret}(\mathbf{X})$ and $p_{grad}(\mathbf{X})$, respectively. However, they may not take full advantage of larger data sets with

complex features (Bussmann et al. 2020). For example, our dataset includes the variable x_{major} , the primary major of a student, which has over 100 possible levels. The presence of such variables, coupled with the limitations of logistic regression as it pertains to modeling higher-order interactions (Levy and O’Malley 2020), supports the use of more complex algorithms.

It is a common consensus that machine learning algorithms, such as Random Forests (Breiman 2001) and Neural Networks (Rojas 2013) are able to better incorporate information from large and complex data sets, which affords them greater predictability (Bussmann et al. 2020). Each individual decision tree in a random forest models higher-order interactions (Levy and O’Malley 2020), and neural networks overcome the problem of feature selection by finding the feature representation that minimizes the given loss function (Goodfellow et al. 2016). However, this increase in predictability often comes at a cost, namely explainability (Burkart and Huber 2021). Due to the lack of information that these models provide to the user on how their classifications were made, they have been termed “black box models” by some (e.g. Carvalho et al. 2019; London 2019; Bikmukhametov and Jäschke 2020). In comparison to other statistical methods, such as logistic regression, there is more work that the user must do in order to understand the results of these methods. This trade-off between accuracy and explainability is our motivation for using a variety of models to achieve our research goals.

2.2.3 Regression Methods for Inference

Note that our response variables, denoted as Y_{grad} and Y_{ret} for graduation and retention, respectively, are binary. That is, their outcomes can be represented by a 1 if the student successfully graduated within six years or returned after their first year, or a 0 if the student did *not* graduate within six years or return after their first year. It is difficult to find patterns when directly examining binary data. A simple first option is to create a scatterplot, as in

Figure 2.4, where first year retention (y) and gift aid (x) are plotted. The blue points depict whether a particular subject was retained, a y -value equal to 1, or was not retained, a y -value equal to 0, after the first year. It is nearly impossible to observe any patterns by direct visual examination of the data, however, and a better option is needed.

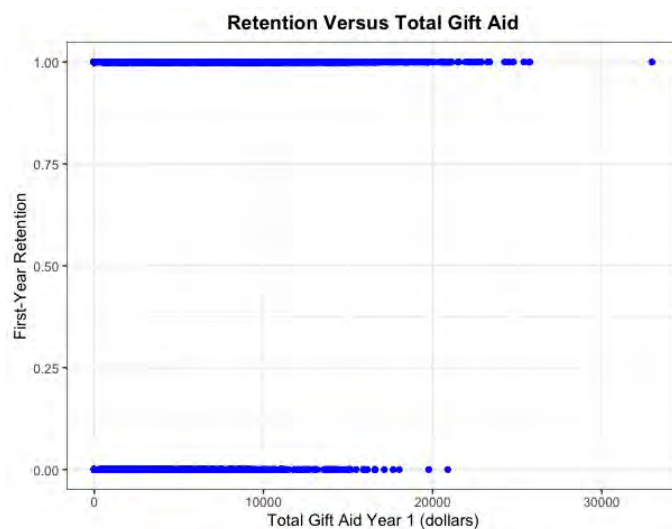


Figure 2.4: Scatterplot of first year retention vs total gift aid. The blue points indicate the students that were retained, $y = 1$, and were not retained, $y = 0$, after the first year for the corresponding amount of total gift aid.

For visualization purposes, we can divide the values of total gift aid into equally sized brackets and represent with points each one of the students that fall into each category, retained or deserted, within each bracket. Figure 2.5 depicts this situation, where the cloud of points next to the “Yes” category represents all the subjects that were retained after year one, for a given range of total gift aid, and the points next to the “No” category represent that information for those who deserted. Only data for students receiving more than 0 dollars of gift aid in their first year is plotted.

Interestingly, the data in Figure 2.5 start revealing a pattern. As the amount of financial aid increases, we observe more students falling in the “Yes” category relative to the “No” category, or equivalently, the observed *odds* of retention increase as the amount of total aid increases. The odds of an outcome (e.g. first-year retention) can be calculated as the

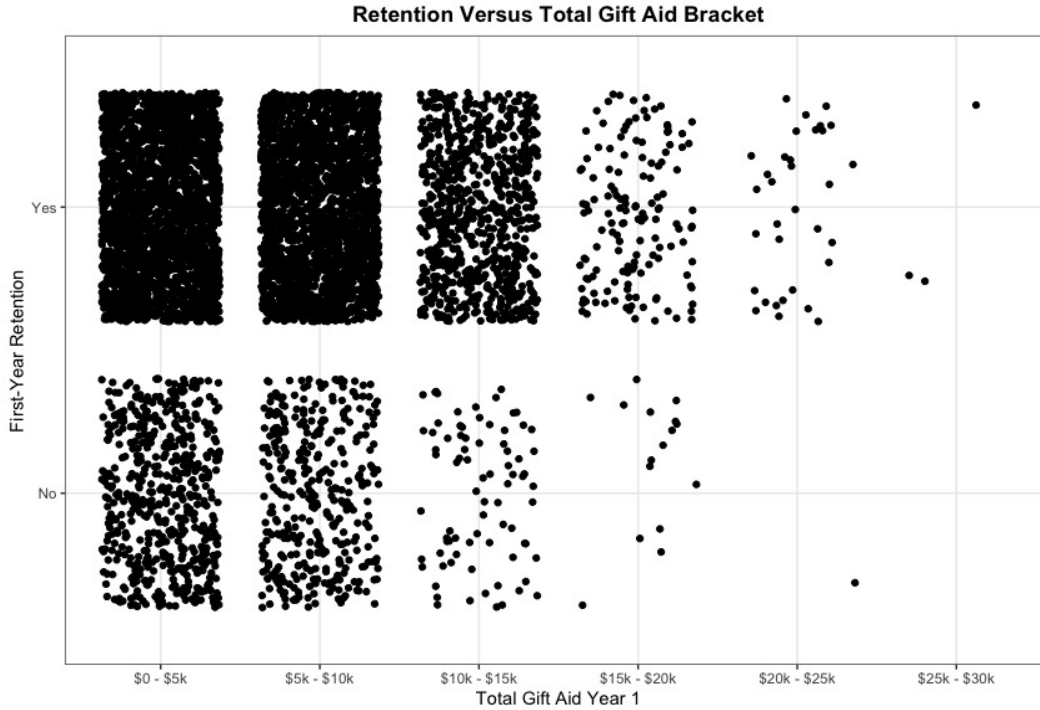


Figure 2.5: Jitter plot of first year retention vs. total gift aid bracket. The points in each bracket represent the students that were and were not retained after their first year and received the corresponding range of total gift aid.

ratio of the number of times that the outcome occurs to the number of times that it does not. Alternatively, we can define the odds of an outcome as the ratio of the probability that the outcome happens to the probability that it does not happen. This is denoted as $odds = p/1 - p$.

The observation that we made about Figure 2.5 can be formalized and further analyzed using *logistic regression* models, which aim to describe and characterize these types of patterns. More specifically, logistic regression models provide a framework to study the relationship between the odds of success of a response of interest and a set of given covariates. We are interested in modeling the probabilities

$$p_{\text{ret}}(\mathbf{X}) = P(\text{first-year retention} \mid \mathbf{X}) \quad \text{and} \quad p_{\text{grad}}(\mathbf{X}) = P(\text{six-year graduation} \mid \mathbf{X}).$$

A more general definition for the probability of success, given a set of variables, \mathbf{X} , is

$$p_Y(\mathbf{X}) = P(Y = 1|\mathbf{X}).$$

Suppose that we have just one variable, X , and that the functional form, f , of the relationship between the log odds of the event “ $Y = 1$ ” is linear in X . Then we have the following relation between the log odds and X ,

$$\ln \left(\frac{p_Y(X)}{1 - p_Y(X)} \right) = \beta_0 + \beta_1 X.$$

Writing this in terms of the odds, we obtain

$$\frac{p_Y(X)}{1 - p_Y(X)} = e^{\beta_0 + \beta_1 X}.$$

Increasing X by 1 increases the log odds of $Y = 1$ by a constant amount, while increasing X by 1 increases the odds of $Y = 1$ by a constant *relative rate*. Lastly, we can solve for $p_Y(X)$ to obtain

$$p_Y(X) = \frac{e^{\beta_0 + \beta_1 X}}{1 + e^{\beta_0 + \beta_1 X}},$$

and we arrive back at our probabilities of interest.

In order to estimate the parameters, β_0, β_1 , we can use *maximum likelihood (ML) estimation*. ML estimation compares all possible models and selects the one for which the observed data has highest probability of occurring. In order to keep our discussion moving forward, we point the reader to Chapters 4-6 of the textbook *Categorical Data Analysis* (Agresti 2012) for more information on ML estimation for generalized and logistic regression models.

To predict the specific outcome, we can define the following rule: predict a success if $p_Y(X) > t$, where $0 < t < 1$, otherwise predict a failure. A typical values for the threshold, t , is 0.50. However, larger or smaller values may be chosen for the threshold if the consequence of incorrectly predicting a success or failure, respectively, is high. If we desire to use p variables to understand the relationship between the log odds of success and X , then we can extend

the functional form of f to

$$\ln \left(\frac{p_Y(\mathbf{X})}{1 - p_Y(\mathbf{X})} \right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p.$$

Calculations for the odds, probability of success, and estimation of the parameters follows closely to those for the simple univariate case.

In order to understand how the probability of first year retention changes for every \$1000 increase in total first year gift aid, we first consider a model of the form

$$\text{logit}(p_Y(X)) = \log \left(\frac{p_y(X)}{1 - p_y(X)} \right) = \beta_0 + \beta_1 X, \quad (2.1)$$

where X is total gift aid for year 1 in thousands of dollars, (β_0, β_1) are constants to be estimated from the data, and p denotes the probability that a student is retained after the first year, for a give value of X . Using statistical software, we fit the model and obtain the regression output in Table 2.12.

	Estimate	Std. Error	z value	p -value
Intercept	1.234	0.062	19.952	< 0.001
Gift Aid Year 1	0.084	0.010	8.779	< 0.001

Table 2.12: Regression output of simple logistic regression model for first-year retention using total first year gift aid in thousands of dollars. Only data for students receiving more than 0 dollars of aid was used.

We can use the estimates in Table 2.12 to directly predict the probabilities of first-year retention for any given amount of total first year gift aid, X . Specifically, the equation for the regression curve is

$$\hat{p} = \frac{e^{1.234+0.084X}}{1 + e^{1.234+0.084X}}. \quad (2.2)$$

Figure 2.6 depicts this regression curve. The probabilities of first year retention (y) are given for the entire range of observed values of total first-year gift aid (x). The vertical dashed line indicates the average amount of total gift aid given to students in their first year and

the horizontal line corresponds to a first-year retention probability of 0.85. According to this model, the probability of first-year retention for students that receive the average amount of total first year gift aid, \approx \$6,233 for aid-receiving students, is about 85.3%, and the amount of total aid required to attain a probability of retention equal to 0.85 is about \$5,953.

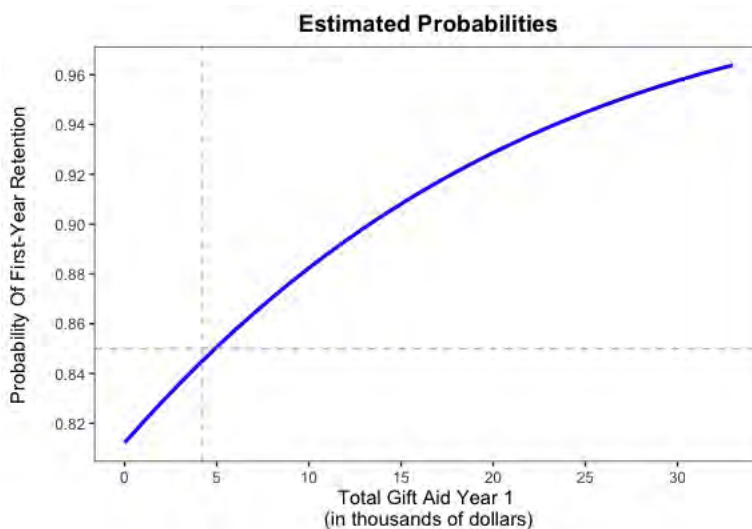


Figure 2.6: Estimated probabilities for first year retention in terms of total first-year gift aid. The dashed lines depict a first-year retention probability of 0.85 on the y -axis and the average total gift aid for year 1 on the x -axis.

These results are somewhat confirmatory of our overall inferential research question. We see that the predicted probability of first-year retention increases as total first-year gift aid increases. However, the relationship between retention and aid is far more complex than this. We will need to include more information from our dataset in our models in order to understand the impact that demographics, academic preparedness, and other information have on this relationship.

2.3 Statistical Analysis Towards Inference

We will use logistic regression models to describe the effect of scholarships on the probabilities of retention and graduation, accounting for different factors. We specifically focus on the

inferential component of the problem, with the aim of fitting and interpreting models that will allow us to understand the effect of gift aid, how this effect changes with demographics, differences between retention and graduation rates, and more. These models will help us to expand our understanding of factors that influence the variability in student success rates.

2.3.1 Cohort-by-Cohort Models

We begin by looking at the differences of this effect when comparing the retention rates for the first, second, and third year, depicted in Figure 2.7. We observe that for almost any fixed amount of aid, the probabilities of retention for first year students are considerably lower than chances of retention of those students in their second or third year. These results confirm the observations made earlier that freshmen students appear to be more vulnerable in terms of retention than upperclassmen.

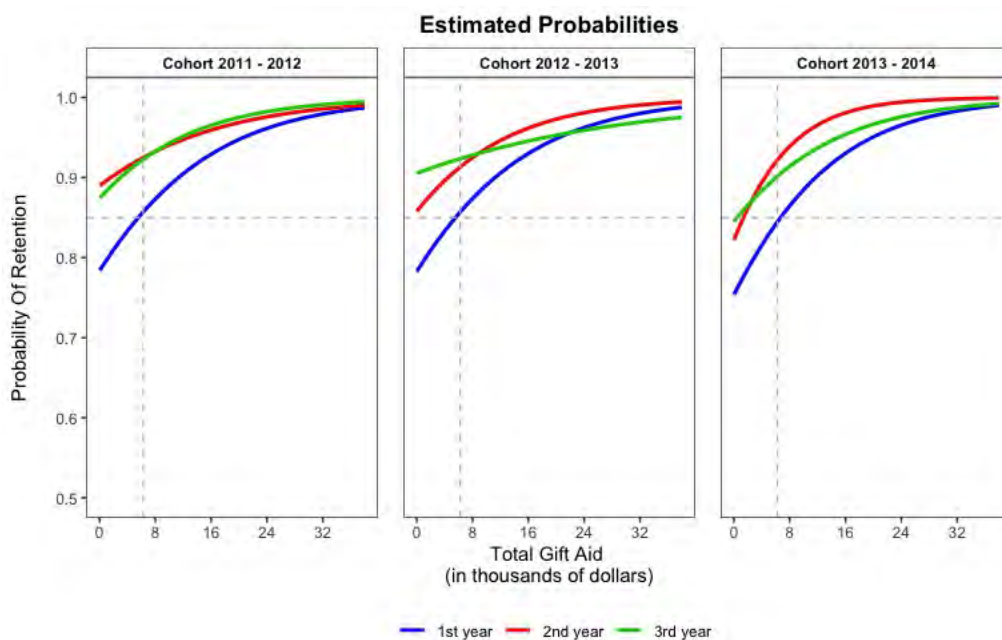


Figure 2.7: Predicted probability curves for first-, second-, and third-year retention across each cohort. Regardless of cohort, first-year retention is lower across all aid categories. If the average award amount for the first year (about 6,200 dollars for those receiving aid) were also awarded in subsequent years, higher probabilities of retention would result.

In fact, a student receiving the average amount of gift aid during their first year, about 6,200 dollars for students receiving aid, still places their chances of success below 85%, compared to nearly 90% for students receiving the same amount in their second or third year. Interestingly, the overall features for the first year retention curve remain almost unaltered when looking at the different cohorts, and the small differences observed for the second and third year retention curves are not statistically significant. Other preliminary analyses show there is no evidence of cohort-to-cohort variation, and therefore we pool the data from these three cohorts in all the results discussed in the rest of this section.

2.3.2 The Effect of Aid by Demographics

In order to look at demographics we considered models that include an *additive* and an *interaction term* to detect group differences. For instance, when looking at gender we considered models of the form

$$\text{logit}(p) = \beta_0 + \beta_1 \text{aidyear1} + \beta_2 \text{gender} + \beta_3 (\text{aidyear1} \times \text{gender}), \quad (2.3)$$

where $\beta_2 \text{gender}$ represents the additive effect and $\beta_3 (\text{aidyear1} \times \text{gender})$ the interaction term. Effectively, these models produce distinctive curves for each group, say

$$\text{logit}(p) = \beta_0 + \beta_1 \text{aidyear1}$$

for female students, and

$$\text{logit}(p) = (\beta_0 + \beta_2) + (\beta_1 + \beta_3) \text{aidyear1},$$

for male students. Therefore, the closer β_2 and β_3 are to zero, the closer these two curves

will be. And if any of these terms deviate from zero, we will notice differences in terms of shifts or curvature between these curves, indicating differences in the scholarship effect due to demographics.

Table 2.13 summarizes the results of fitting the models for first, second and third year retention vs gift aid by demographics. In the table, the letters “a” and “i” indicate the additive and interaction terms were significant, at a significance level of $\alpha = 0.05$, respectively. Observe that while all the variables show evidence of a significant effect in some of the models, only Pell eligibility, first generation and financial need are consistently flagged as significant in all the models. It should be noted, however, that statistical significance implies the data shows evidence that the relationship between the probabilities of graduation/retention in terms of financial aid change when we move from one demographic group to another, but such results might be largely influenced by the number of observations in each group and do not mean necessarily that the magnitude of these changes are of any practical importance. To help visualize these differences Figures 5-8 depict the estimated probabilities of retention

Demographics	Retention year 1	Retention year 2	Retention year 3	Graduation
gender	None	None	i	a
flagpellelig	a i	a i	a i	a i
studentofcolor	None	a	None	a
residency	None	i	None	a i
flagfirstgen	a i	a i	a i	a
financial need	a	a	a i	a i

Table 2.13: Summary of significance of the demographic variables when modeling retention and graduation rates in terms of gift aid. The letter a indicates the additive term was significant at level 0.05. the letter i indicates the interaction term was significant at level 0.05, and None indicates that none of these terms were significant at a significance level of $\alpha = 0.05$.

for years 1-3 and graduation versus gift aid for each of the groups determined by the corresponding demographics. Confidence bands are also included, which were calculated using the Wald interval for maximum-likelihood estimates. In some cases, the standard error of an estimate was larger than normal, resulting in wider confidence bands. In all the figures, the dashed lines indicate a probability of 85% for retention or graduation on the y -axis, and

the average amount of gift aid on the x -axis. We observe that Pell eligibility, first-generation status, and financial need seem to have a greater effect on the estimated probabilities in all years.

For instance, when looking at first-year retention versus gift aid, the predicted probabilities of retention are substantially higher for the students that are not Pell-eligible than for those that are Pell-eligible, in particular, in the range from 0 – 5,000 dollars of gift aid. The average amount of gift aid in year one is about \$4,200 for all students, including those not receiving any aid. For this amount the predicted probability of retention for those that are not Pell-eligible is 89.9%. This is compared to only 64.1% in the Pell-eligible group. Pell-eligible students would need almost twice as much gift aid as those that are not Pell-eligible in order to have a predicted probability of retention around 80%. Not surprisingly, financial need seems to have a substantial effect as well. In particular, for first-year retention and six-year graduation rates.

FIRST YEAR RETENTION VS GIFT AID BY DEMOGRAPHICS

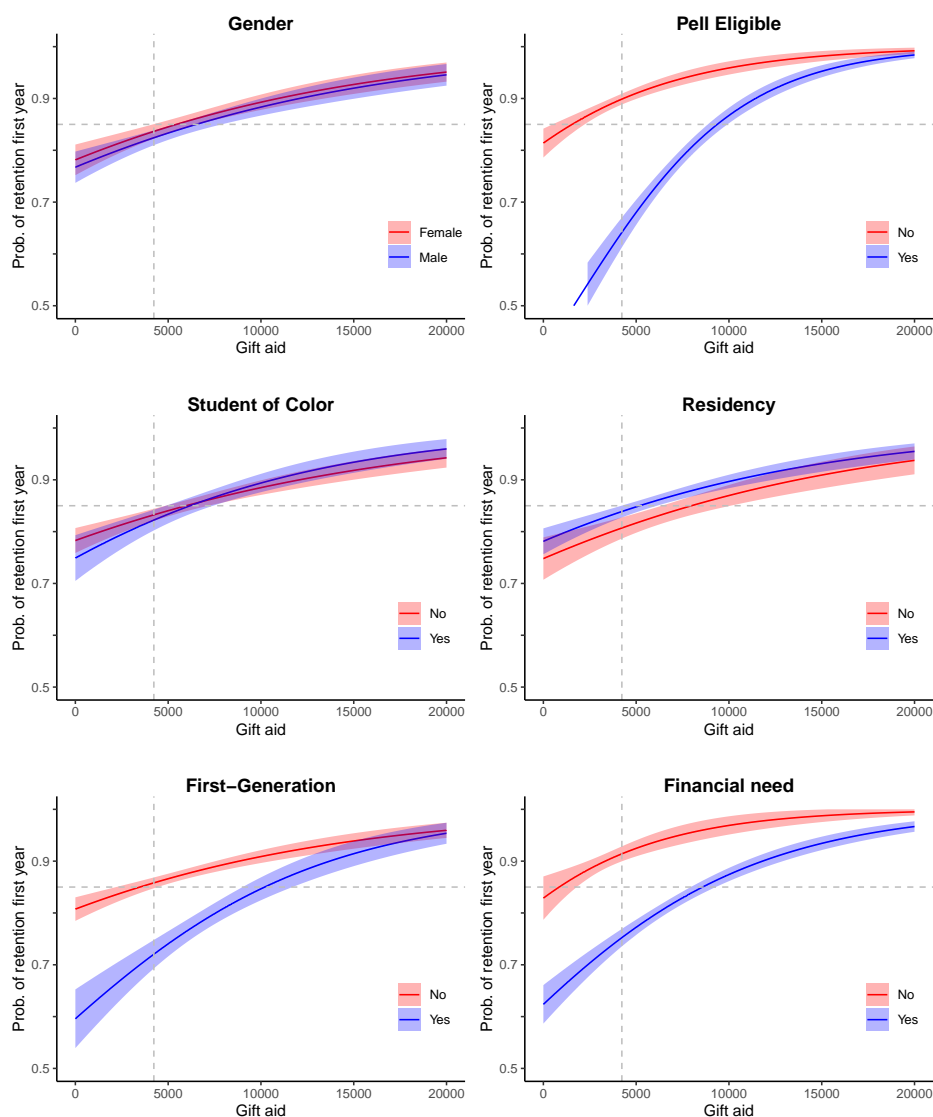


Figure 2.8: Plots for the estimated probabilities of first year retention in terms of gift aid, accounting for the effect of the demographics of interest. The dashed lines indicate a probability of 85% in the y -axis, and the average amount of gift aid in the x -axis. We observe that Pell-eligibility, first generation and financial need seem to have a greater effect in the estimated probabilities.

2.3.3 Accounting for Academic Performance

These results indicate that gaps in student success due to demographics could be closed through the strategic awarding of gift aid. However, any of these models could inadvertently overstate or understate the significance of these results since we have not adjusted for other

SECOND YEAR RETENTION VS GIFT AID BY DEMOGRAPHICS

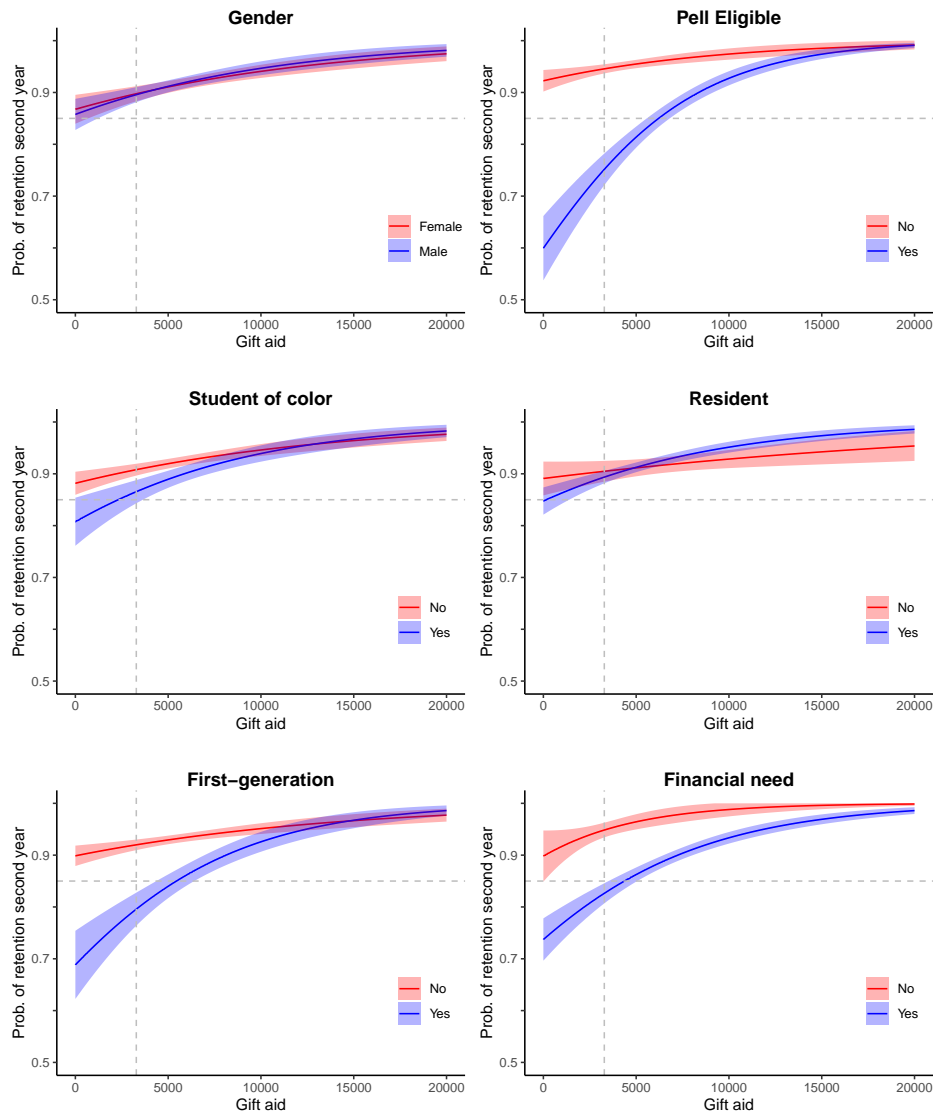


Figure 2.9: Plots for the estimated probabilities of second year retention in terms of gift aid, accounting for the effect of the demographics of interest. The dashed lines indicate a probability of 85% in the y -axis, and the average amount of gift aid in the x -axis. We observe that Pell-eligibility, first generation and financial need seem to have a greater effect in the estimated probabilities.

relevant factors. For example, it would be sensible to incorporate information on academic performance these models as this may have an underlying impact on student success. To be clear, the issue we want to further address is whether these demographic variables continue playing a significant role in student success, after adjusting for academic performance. In this regard, some metrics of academic performance are available, and though most of them

THIRD YEAR RETENTION VS GIFT AID BY DEMOGRAPHICS

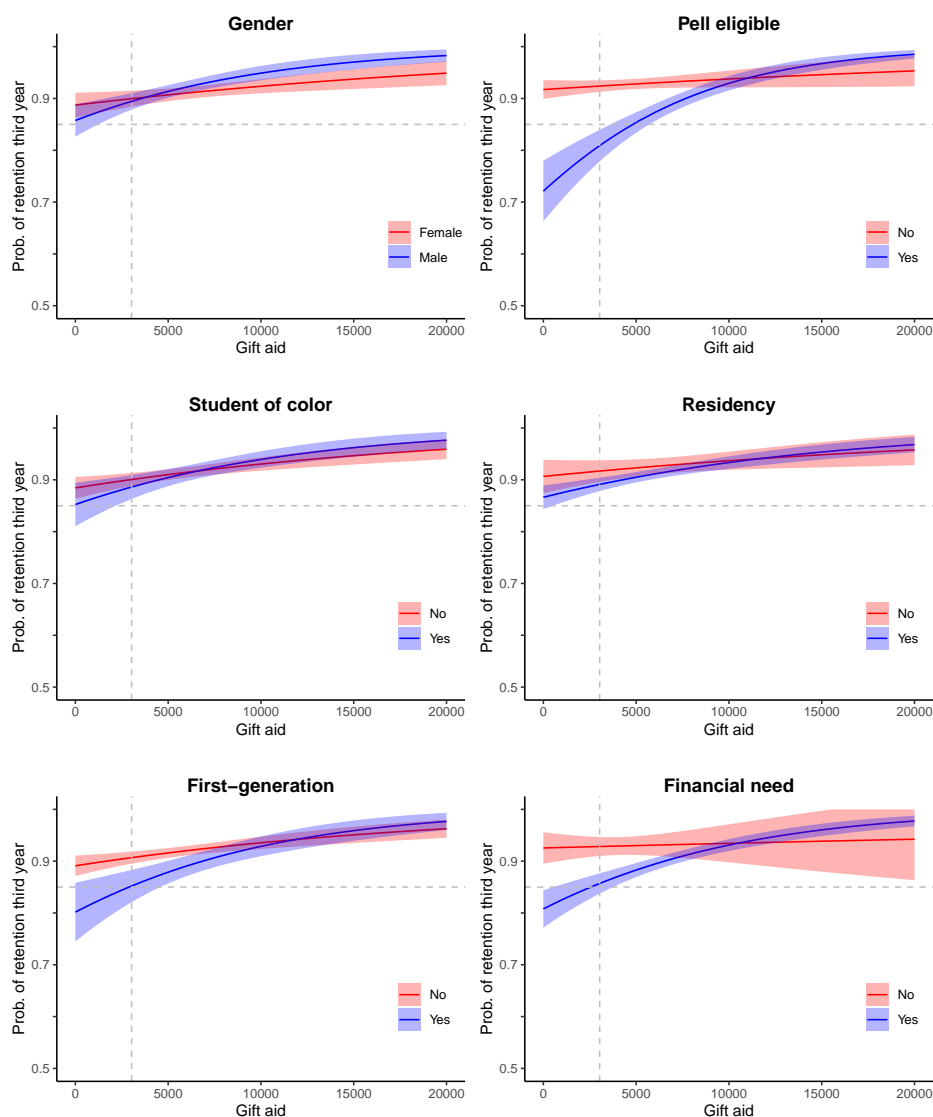


Figure 2.10: Plots for the estimated probabilities of third year retention in terms of gift aid, accounting for the effect of the demographics of interest. The dashed lines indicate a probability of 85% in the y -axis, and the average amount of gift aid in the x -axis. We observe that Pell-eligibility seem to have the greater effect in the estimated probabilities, and first generation and financial need seem to have a moderate effect. Confidence bands are wider when the standard error of a prediction was larger than normal. Predicted probabilities close to 1 produced upper bounds above 1, which were capped to 1. This was the case with the model for financial need.

are correlated, they do seem to play an important role in explaining student success.

For example, Figure 2.12 provides the predicted probabilities of first year-retention for various values of gift aid, after accounting for SAT scores and GPAs. In the left-hand panel we

GRADUATION VS GIFT AID BY DEMOGRAPHICS

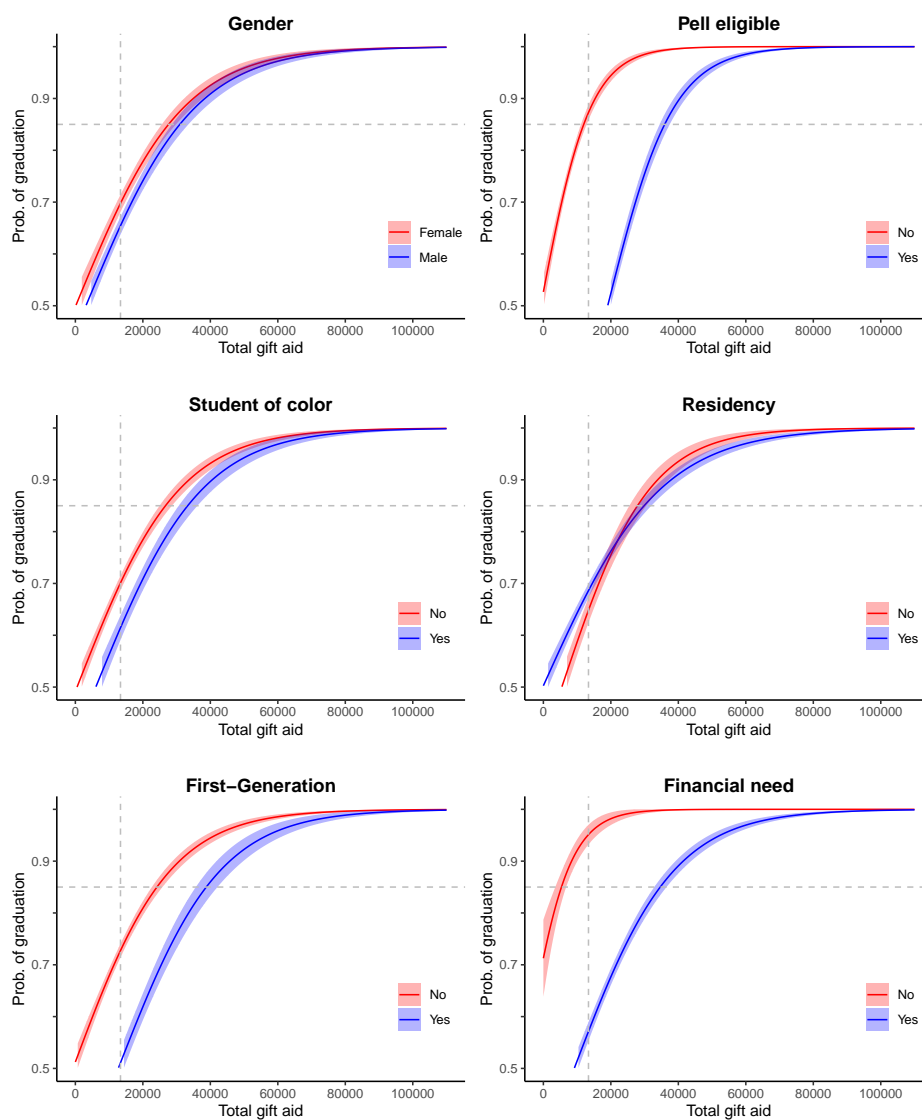


Figure 2.11: Plots for the estimated probabilities of graduation in terms of gift aid, accounting for the effect of the demographics of interest. The dashed lines indicate a probability of 85% in the y -axis, and the mean total amount of gift aid in the x -axis. We observe that Pell-eligibility, first generation and financial need seem to have a greater effect in the estimated probabilities, and student of color seem to have a moderate effect.

provide the predicted probabilities of retention as aid increases, for students whose composite SAT score is less than, or greater than or equal to, a score of 1600. In the right-hand panel we provide the predicted probabilities of retention as aid increases, for students whose high-school GPA at the time of application is less than, or greater than or equal to, a GPA of

3. For reference, the observed median SAT score and high-school GPA of first-year students receiving aid were 1640 and 3.71, respectively.

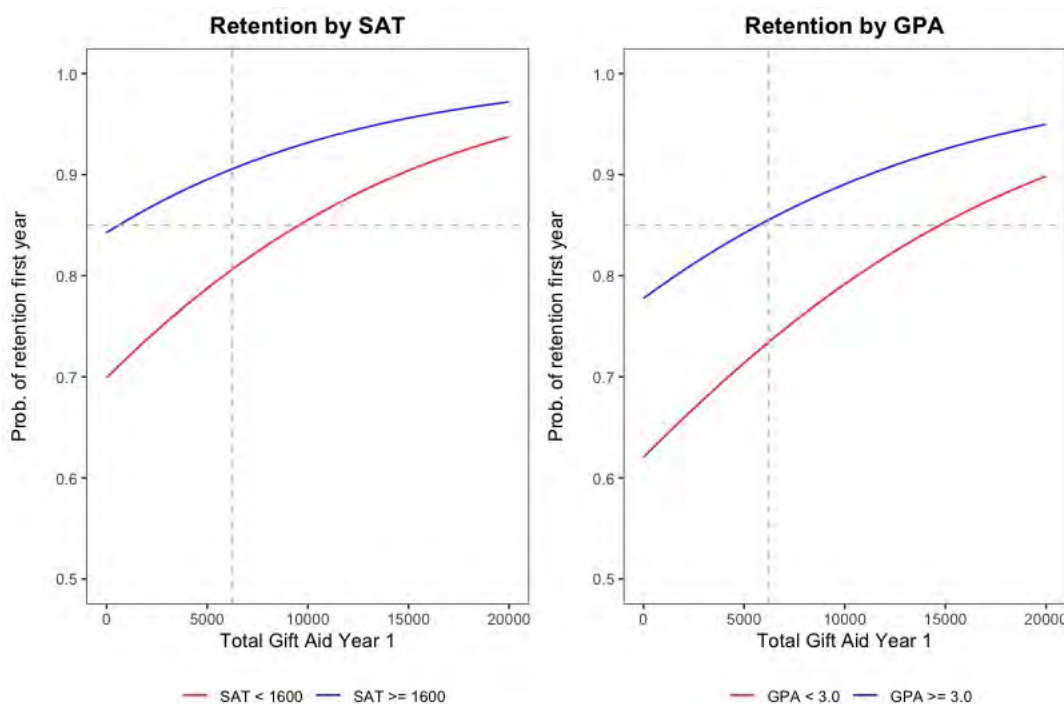


Figure 2.12: First year retention probabilities versus gift aid. The left- and right-hand panels give the estimated probabilities as aid increases, after accounting for composite SAT scores and high school GPA, respectively. In both plots the red line corresponds to the group with lower academic performance in the respective metric, and the blue line corresponds to the group with higher performance. Only students receiving aid were included here.

Differences in the predicted probabilities of the left-hand panel of Figure 2.12 are quite noticeable. Low-performing students receiving the average amount of gift aid have an estimated probability of retention near 80%. In contrast, high-performing students have an estimated probability of retention near 90%. Though this gap in predictions due to SAT scores narrows as gift aid increases, the differences are still fairly substantial at the 10,000 dollar mark, which well exceeds the 75th percentile of gift aid awarded in year one. Similarly, the right panel shows that students entering with a high-school GPA less than 3.0 have lower chances of success which barely narrows as aid increases.

These models that incorporate academic performance provide evidence that not all dis-

crepancies in student success may be effectively dealt with through awarding more gift aid. However, it would be more thorough to consider how academic performance *and* demographics impact student success. Recall that our question was whether the demographic variables continue playing a significant role when explaining the after adjusting for academic performance. Therefore, combining our previous two models into one may give better insight into how these factors impact student success for low-performing and high-performing students in different demographic groups. It is important to include both pieces of information in the models so that we can properly identify significant results for all groups in our population.

Table 2.14 provides the estimated coefficients of the additive and interaction terms for each demographic variable obtained from models describing the first-year retention and six-year graduation rates, before and after adjusting for high school GPA and class rank. The blue color indicates those coefficients that were flagged as statistically significant at $\alpha = 0.05$ level. Aid variables were scaled to thousands of dollars when fitting these models.

Estimated Coefficients (adjusting for academic performance)	Retention year 1 add/inter	Graduation add/inter
gender	0.101/-0.025	-0.096/-0.003
flagpellelig	-1.547/0.048	-2.020/-0.022
studentofcolor	-0.150/-0.039	-0.491/0.008
residency	-0.036/0.029	0.496/-0.020
flagfirstgen	-0.956/0.057	-0.800/0.000
financial need	-1.076/0.039	-1.669/-0.044
Estimated Coefficients (not adjusting for academic performance)	Retention year 1 add/inter	Graduation add/inter
gender	-0.081/-0.001	-0.188/-0.001
flagpellelig	-1.843/0.057	-2.066/-0.033
studentofcolor	-0.189/0.028	-0.353/-0.003
residency	0.186/0.008	0.433/-0.020
flagfirstgen	-1.045/0.045	-0.895/-0.003
financial need	-1.071/-0.043	-1.53/-0.086

Table 2.14: Estimated coefficients for additive (add) and interaction (inter) terms of the corresponding demographic variables when modeling the first year retention and graduation probabilities with and without adjusting for academic performance. The numbers in blue indicate the coefficients that are significant at the level $\alpha = 0.05$.

We observe that the estimated coefficients agree in magnitude, direction, and statistical significance in most cases. When differences exist, they occur mostly in terms with a small size effect. This can be seen in the first-year retention interaction term for Pell-eligible students and the six-year graduation interaction term for financial need. As a result, the overall conclusions remain the same: Pell-eligibility, first-generation status, and financial need status are the demographic variables that are predicted to have the most impacts on retention and graduation rates.

2.3.4 Further Exploring the Impact of Gift Aid by Race

The results of the bivariate logistic regression models of the previous section indicate that Pell-eligibility, first-generation status, and financial need most impact student success. Since the variable student of color broadly categorizes students as non-white and non-international, or not, we could be overlooking important results about the impacts of aid for students of different races. Therefore, in this section we will fit similar models as before, but using race description. Recall that this variable gives a student's self-selected race from nine categories: American-Indian or Alaska Native, Asian, Black or African-American, Hispanic, Multiple Races, Native Hawaiian or Pacific Islander, White, International Student, and Unknown. Due to the very low percentages for their groups seen in Table 2.8, we combined students whose racial categories were American Indian or Alaska Native, Native Hawaiian or Pacific Islander, International, or Unknown into one category termed "Other". The percentage of students in these individual groups was less than 1% each.

Figure 2.13 shows the estimated probability curves for first-year retention as total gift aid in the first year increases, for students in each racial group. In the first panel, we report this information based on a model which accounted for race, but not for severity of need. In the second and third panel, we provide the resulting curves after accounting for both race

and severity of need. It is clear that the model that accounts for racial group alone does not tell the entire story. There is a clear shifting up and down in the predicted probabilities when we include information on whether the student had less or more severe need, respectively. Hispanic and Black or African American students had very similar probabilities after accounting for need, making their probability curves almost indistinguishable.

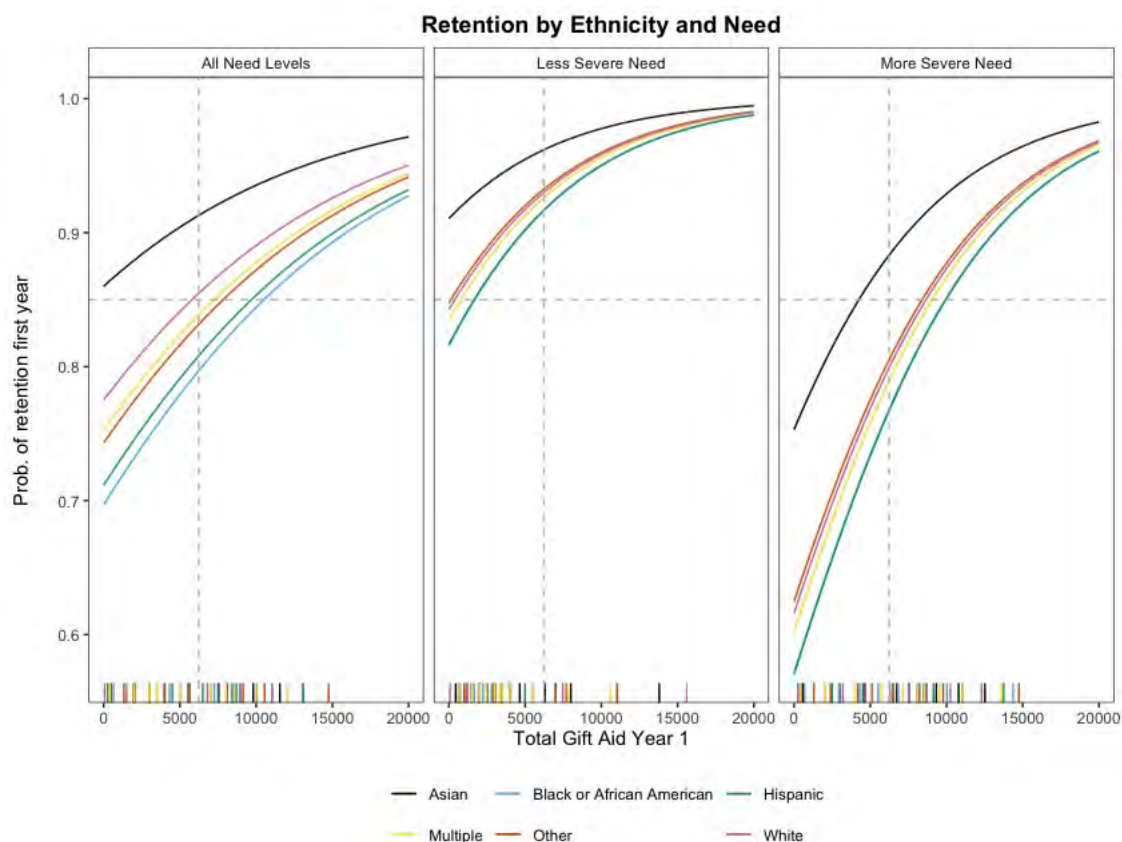


Figure 2.13: Fitted probability curves from logistic regression models for total first-year gift aid and first-year retention. Left-most panel are curves from model that accounts only for race. Right-most two panels are results for model that accounts for race and need. Rug gives range of total first-year gift aid variable for each population in the model corresponding to the facet.

According to Figure 2.13 awarding the average first-year gift aid to students who are in less severe need would result in a predicted probability of success above 0.90 for all races with even higher probability for Asian students. However, awarding this same amount to students who are in more severe need would result in predicted probabilities of retention that are

lower than 0.8 for all students except Asians. It will be around 0.87 for Asian students. These results provide evidence that the effect of race is further impacted by severity of need. The achievement gaps between races are predicted to narrow as more aid is awarded but larger amounts of aid are needed to see this effect when considering only students in need. The effect of race diminishes when we account for need but it is still present, this is more true for students in more severe need when lower amounts of aid are awarded.

Figure 2.14 gives six-year graduation predictions for all students by race, and less- or more-severely in need students by race. Additive effects are present for both race and need, noted by the change in starting points between the second and third panel and within each of these panels as well. In this case however, the gaps in graduation curves are similar regardless of severity of need. This indicates that when considering six-year graduation the impact of severity of need on the probability of graduation is not further related to how much total gift aid is awarded. The impacts of the latter are fairly similar regardless of severity of need.

Table 2.15 gives the coefficients of the additive racial group terms depicted in Figures 2.13 and 2.14. To clarify, there are four separate models of retention or graduation on total aid (in thousands of dollars) and racial group, for students in need and not in need. The coefficients for Asian students are just the intercepts from the models, as all other racial group indicator variables would evaluate to zero. Blue text indicates that the term was statistically significant at the 0.05 significance level. The variables total first-year aid and total aid over the first four-years were significant in all models. These results indicate that a significant additive effect of race on six-year graduation is present for all races when considering students in need. This remains true for all students except those of race Other when considering first-year retention. However, there are fewer significant coefficients when considering students who are not in need. In this case, the only significant additive effects are those for the retention of students in racial groups Asian or Multiple.

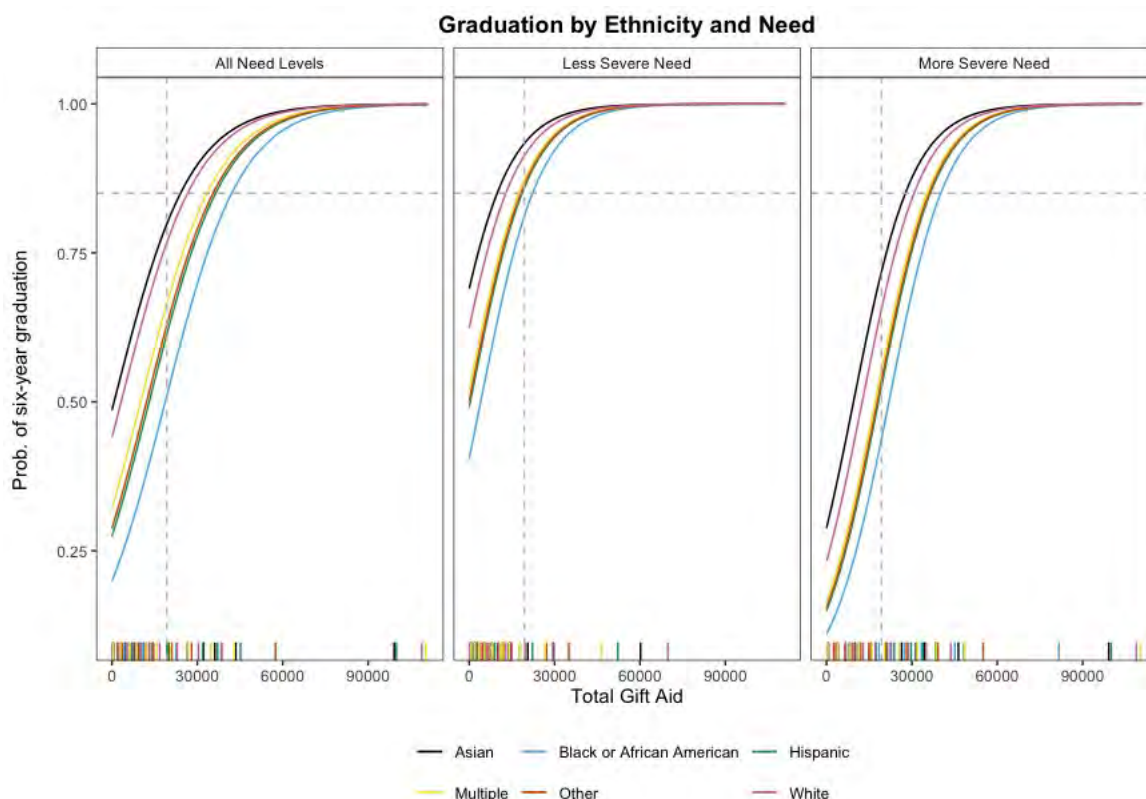


Figure 2.14: Six-year graduation predictions for all students by race, and then less- or more-severely in need students by race. Additive effects are present for both race and need, noted by the change in starting points between the second and third panel and within each of these panels as well. Rug gives range of total gift aid over first four years for each population in the model corresponding to the facet.

2.4 Moving Towards Predictive Models

We have developed models to examine the relationship between gift aid and student success for students from varying demographical backgrounds. The model results indicated that achievement gaps can be closed through the strategic awarding of gift aid for students who are Pell-eligible, first-generation, from minority racial groups, and have more severe financial need. These results remained the same after correcting for the academic preparedness of students as measured using SAT results.

The models that we have fit thus far are useful for making conclusions about the effect of aid

Estimated Coefficients (students in need)	First-Year Retention	Six-Year Graduation
Asian	1.16	-0.54
Black or African American	-0.88	-1.06
Hispanic	-0.87	-0.83
Multiple	-0.65	-0.65
Other	-0.68	-0.91
White	-0.67	-0.29
Estimated Coefficients (students not in need)	First-Year Retention	Six-Year Graduation
Asian	2.06	1.18
Black or African American	-0.39	-2.53
Hispanic	-0.49	-0.63
Multiple	-0.91	-0.79
Other	-0.23	-0.93
White	-0.49	-0.48

Table 2.15: Coefficients of additive terms for models of retention and graduation on total aid by race for students in need and those not in need. All but one of the coefficients are significantly different from 0 for students in need, while for students not in need only two are. Additionally, many coefficients are more negative or smaller for students in need than those not in need.

on student success in the presence of other covariates, such as demographics and academic performance. However, another important component of our analysis pertains to predicting the probability of retention or graduation given a set of covariates. In the next chapter we will shift our focus to this component. We will discuss an underlying issue with our dataset that poses a challenge to predictability. We focus the rest of our analysis on first-year retention because these data are available at the end of each academic year for updating predictive models while those for six-year graduation take longer to collect. We will first tackle the predictive problem using the same logistic regression models already discussed but with more variables included. We will evaluate and discuss the predictive metrics of such a model as a way to motivate our need for more complex models that better tackle the predictive component of the problem. We will then discuss those more complex models and explore some solutions to the technical problem that underlies our analysis.

Chapter 3: Dealing with Imbalanced Data and the SMOTE Approach

Key issues that were addressed by Zeineddine et al. (2021) in their study were that of overlap and imbalance. In general, when certain values of a variable are observed with a higher frequency than others, the variable is called *imbalanced* (He and E. A. Garcia 2009). *Class overlapping* is defined as the presence of examples in areas of the feature space where the decision boundaries of a classifier intersect (Alejo et al. 2013). We found that these issues were ignored in most of the literature on student success models.

For example, Miller and Lesik (2014) notes that the groups which they studied were largely equivalent in terms of ACT scores, class rank, gender, and other factors, but they did not address how this overlap could impact their results. Additionally, the semester drop out rates reported by Ameri et al. (2016) give evidence that these variables are imbalanced, but they did not address the possible impacts of this imbalance on their predictions.

As noted by Zeineddine et al. (2021), in order to better guide student success efforts and help students who are truly in need, student success models must be able to predict negative *and* positive outcomes with high accuracy. Since reductions in classifier performance can be attributed to the combined impact of imbalanced classes and overlapping between classes (Batista et al. 2005), the presence of these two data difficulties is likely to present a challenge to achieving this goal.

Students who do not return after their first year, or do not graduate within six years, form a smaller subset of the entire population. Table 3.1 gives the proportion of students who deserted, stayed, graduated, or did not graduate, for a six year period. Though these negative outcomes are not the events of interest, obtaining accurate predictions for these

students is key to understanding how their probabilities of success can be increased through the strategic awarding of gift aid. Since we have less information on students with these specific characteristics we will need to adapt our statistical procedures to account for this.

	Deserted	Retained	Not Grad	Grad
1st year	15.76%	84.24%	-	-
2nd year	22.99%	77.01%	99.90%	0.10%
3rd year	28.26%	71.74%	97.93%	2.07%
4th year	65.50%	34.50%	64.70%	35.30%
5th year	90.63%	9.37%	39.08%	60.92%
6th year	96.45%	3.55%	32.71%	67.29%

Table 3.1: Retention and graduation rates over time, for all cohorts. Imbalance observed in first-year retention and six-year graduation.

For a binary classification task such as ours, most statistical methods will try to discover values that are not common to both classes. These values will then be used to form decision rules used to determine which class a new observation falls into. Since our models give similar predictions for similar observations, observations that have similar covariate values but different labels, will have the same predictions. This will negatively impact the accuracy of our predictions. An example of this issue is given in Figure 3.1, where total aid in thousands of dollars is plotted for retained and deserted students, over four years. Observe that the range of values for total aid is quite similar between retained and deserted students. There were retained students with higher amounts of total aid in years one through three, however, there is still a considerably amount of overlap.

These results provide evidence that the issue of overlap and imbalance underlies our data analysis. Therefore, we must tackle this issue if we desire to truly develop predictive models that will help all students. In order to further motivate our need for more complex models and a solution for the overlap and imbalance we first attempt to extend the logistic regression models discussed in the last chapter. We show that simply adding more variables is not sufficient for correcting these issues.

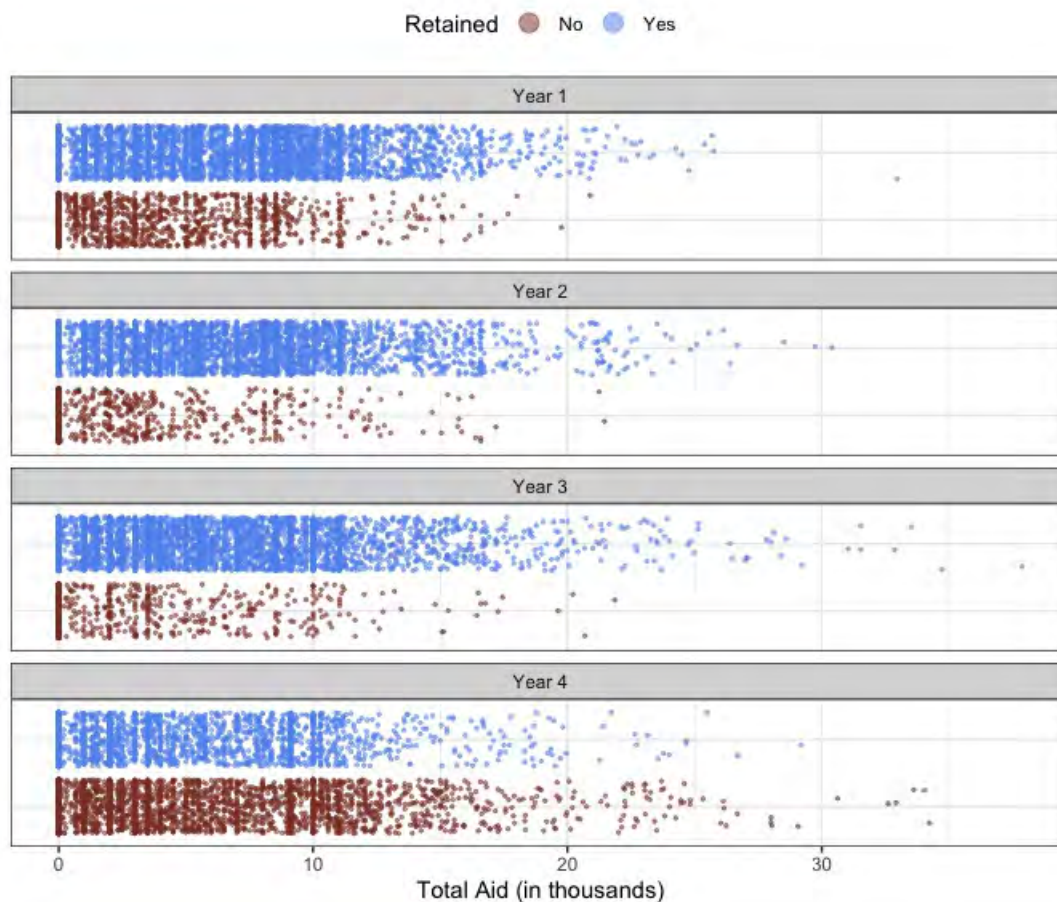


Figure 3.1: Total aid in thousands of dollars given to students who were retained (blue) and who deserted (brown). In each year, many values for aid are similar regardless of a student's retention status. This issue is called overlap, and it presents a challenge when trying to classify observations.

3.1 Extending the Logistic Regression Model

In order to obtain a more accurate reflection of the predictive capabilities of our models, we will train models on 70% of the data, called the training set, and obtain predictions from those models using the other 30% of the data, called the testing set. To preserve the original distribution of the response variable in the training and testing sets, we use stratified sampling to create these subsets. Additionally, we will use *k-fold cross validation* to fit models at times. In order to implement this method we first split the training data into k folds of roughly equal size. For $i = 1, \dots, k$, a model is trained on all but the i -th

fold, and then predictions and their performance metrics are obtained using the i -th fold. This is repeated for all k folds. This method allows us to obtain another estimate of the performance of our models on unseen data while reserving our test data until our very final assessment of predictive capability, after we have fully trained the models. The distribution of the response variable in each fold was consistent with that of the overall dataset, preserving the imbalance.

Denote the number of true positives, true negatives, false positives, and false negatives as TP, TN, FP, and FN, respectively. Let P and N denote the number of positive and negative examples in the dataset. The positive class is retention since we are interested in predicting probabilities of retention. Performance metrics that quantify the quality of a classifier's predictions include those given below.

- Accuracy = $\frac{TP + TN}{P + N}$
- Sensitivity = Recall = True Positive Rate (TPR) = $\frac{TP}{TP + FN}$
- Specificity = True Negative Rate (TNR) = $\frac{TN}{TN + FP}$
- Precision = Positive Predictive Value (PPV) = $\frac{TP}{TP + FP}$
- F-score = $\frac{2 \times PPV \times TPR}{PPV + TPR}$ (geometric mean of precision and recall)
- Balance accuracy = $0.5 \times (TNR + TPR)$

Due to the imbalanced nature of our data it is possible to find models that perform well simply by predicting the positive class always. These models are termed *random classifiers* and can be quite misleading. Usually their poor performance is flagged when they are used to predict unseen data but when training models it is best to protect against selecting these as a final model. In order to quantify the difference in the predictions of our models and

those of a random classifier Cohen’s Kappa coefficient can be used. For a binary classification problem it is

$$\kappa = \frac{2 \times (TP \times TN - FN \times FP)}{(TP + FP) \times (FP + TN) + (TP + FN) \times (FN + TN)}.$$

κ gives a measure of agreement between the truth and predictions in the binary classification case. A value of 1 indicates total agreement between truth and predictions, while a value at or near 0 indicates no agreement other than that due to chance. A negative value could indicate no relationship or non-random differences between the truth and predictions.

The full logistic regression model that we fit includes: high school GPA, class rank, total first-year aid, binary gender, Pell-eligibility, race description, Oregon residency, first-generation status, and severity of need. Interactions were also included between total first-year aid and each of these covariates. The results in Table 3.2 provide the predictive performance results from 10-fold cross-validation. The consistently low specificity and high sensitivity indicates that the model under predicts desertion and over predicts retention. This issue may be due to the much higher number of retained students in our dataset.

In order to improve upon the metrics reported in Table 3.2, threshold selection processes could be used to find the threshold that most maximizes the TPR while minimizing the FPR. The threshold is the value of the predicted probability over which we classify an observation as a success. While this is often set at 0.5, a larger or smaller threshold could give better accuracy for both classes overall.

One useful indicator of the predictive performance of a model as the threshold changes is the *receiver operator characteristic (ROC) curve*. The ROC curve maps the False Positive Rate (FPR, or 1-Specificity) to a corresponding True Positive Rate (TPR, or Sensitivity) for various decision thresholds $p \in (0, 1)$. A perfect classifier will result in an ROC curve that passes through the point (0,1) as p decreases. This would indicate that there exists a value

Fold	Accuracy	Kappa	AUC	Sens	Spec	Precision	Recall	F-Score
1	0.849	0.148	0.803	0.974	0.13	0.866	0.974	0.917
2	0.85	0.163	0.769	0.989	0.12	0.855	0.989	0.917
3	0.854	0.163	0.785	0.985	0.125	0.862	0.985	0.92
4	0.854	0.187	0.773	0.977	0.156	0.868	0.977	0.919
5	0.861	0.292	0.754	0.982	0.231	0.87	0.982	0.922
6	0.812	0.015	0.675	0.973	0.037	0.83	0.973	0.896
7	0.856	0.175	0.773	0.975	0.149	0.872	0.975	0.92
8	0.847	0.16	0.781	0.971	0.143	0.865	0.971	0.915
9	0.844	0.243	0.701	0.969	0.212	0.861	0.969	0.912
10	0.853	0.253	0.824	0.988	0.189	0.856	0.988	0.918
Mean	0.848	0.18	0.764	0.978	0.149	0.861	0.978	0.916
SD	0.013	0.075	0.045	0.007	0.054	0.012	0.007	0.008

Table 3.2: Cross-validation results from full model for retention with many covariates and interactions. Results across folds are consistent. The model struggles to predict the negative class (desertion) with consistently high specificity but consistently low sensitivity. This shows consistent misclassification for the negative class.

of the decision threshold p for which all predictions are correct. This is most achievable when the model produces very pure predicted probabilities. However, this rarely happens in practice and a more attainable goal is to achieve an ROC curve above the line $y = x$. A curve lying along this line would indicate that the models predictions are no better than random chance.

Additionally, to quantify the quality of the ROC curve, the *area under the ROC curve (AUC)* can be calculated. This value will range between 0 and 1. A value near 0.5 indicates no better performance than a random classifier, values above 0.5 indicate better than random performance. Values below 0.5 indicate worse performance than a random classifier. Figure 3.2 provides the ROC Curve with the AUC in the title and a tile plot of specificity and sensitivity from the model. The FPR increases non-trivially with the sensitivity across almost all values on the x-axis. This implies that increasing the threshold will always result in a non-trivial decrease in the sensitivity. The tile plot also shows this with more detail. The tile plot shows that the trade-off between sensitivity and specificity does not begin until a threshold of about 0.75 is reached and that it is strong. This indicates that deserted students

are being classified as retained with very high probability.

Given that we can only improve our predictions with a trade-off we are likely dealing with both imbalance and overlap. The predicted probabilities of retention will be large for students who actually deserted when their characteristics in the data are similar to most students who were retained. Visually, this would mean that these points fall into areas of the feature space that are dominated by the majority class. The patterns learned by the model will flag points with these characteristics as belonging to the majority class, but some actually belong to the minority class.

A *reliable* binary classifier is one that predicts a low probability for any class that a point does not truly belong to and a high probability for the class that it does belong to. In our case, we would hope that students who were retained have high predicted probabilities of retention and that those who deserted have low predicted probabilities of retention. We visualize the reliability of our model using Figure 3.3. The predicted probabilities of retention for each point in our test set were binned into the intervals given on the x-axis. These are plotted against the true retention status of the student, on the y-axis. Additionally, the text label above each panel give the relative frequency of retention taken over all points with predicted probabilities of retention falling inside the interval on the x-axis.

A classifier that produces very pure probabilities will result in a reliability diagram with a large density of points in the top-right corner and bottom-left corners. Figure 3.3 reveals that our model predictions for deserted students have high bias *and* high variance. While the bias in predictions for retained students is much lower, there is still quite a bit of variability in the predictions. This figure also shows that threshold selection will not be helpful to the overall quality of the model. Even with a threshold of 0.5, quite a few retained students are misclassified and this would only become worse if we increased the threshold.

More predictive metrics from the logistic regression model are given in Table 3.3 . These were

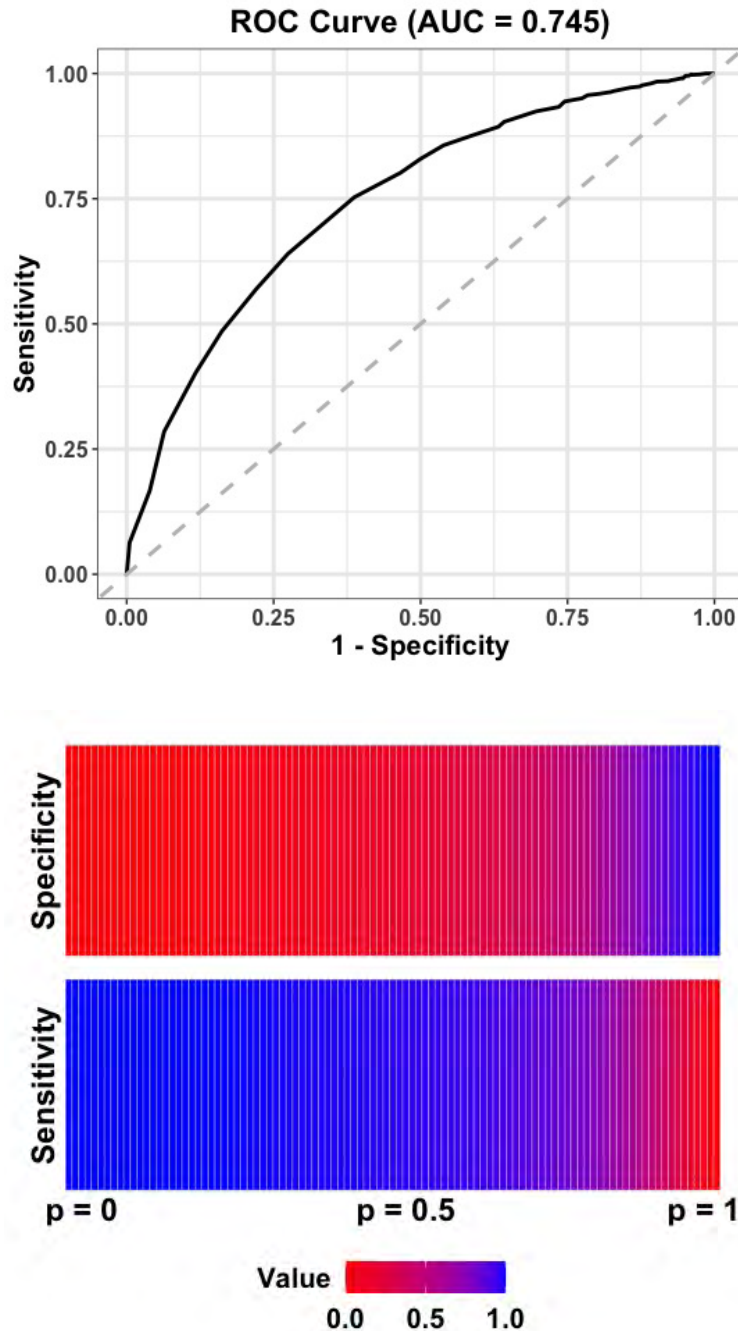


Figure 3.2: ROC Curve and tile plot of specificity and sensitivity from larger logistic regression model for retention. The ROC curve provides TPR (y) and FPR (x) for varying thresholds. The colors in the tile plot are mapped to the value of the sensitivity and specificity for various thresholds (p) on the x-axis. The threshold is the value of the predicted probability over which we classify an observation as a success or retained. These plots provide evidence that using a lower threshold will produce a larger decrease in sensitivity than the increase in specificity.

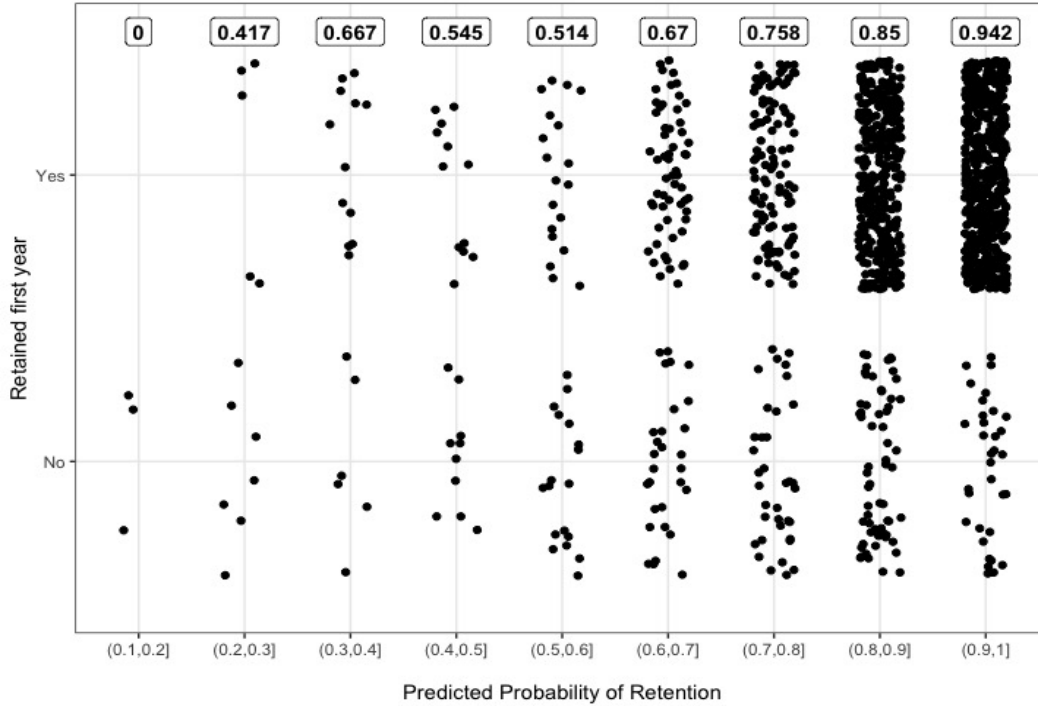


Figure 3.3: Reliability diagram for test set predictions using fuller logistic regression model. The panel labels give the relative frequency of retention taken over all points with predicted probabilities of retention falling inside the interval on the x-axis.

calculated on the test dataset. The underlying issue of imbalance and overlap will impact predictive metrics other than the specificity and sensitivity as well. In order to achieve the predictive goals of our research we should not only use more complex machine learning models but we should also deal with the overlap and imbalance in our dataset before doing so. We will now discuss methods that were developed to address the issues of overlap and imbalance. After this we will discuss machine learning models that are less interpretable but are known to produce better predictions than logistic regression.

Accuracy	Bal. Accuracy	Sensitivity	Specificity	F-Score	AUC	Kappa
0.842	0.551	0.974	0.128	0.913	0.745	0.144

Table 3.3: Performance metrics for predictions of test data set using fuller logistic regression model. Balanced accuracy is much lower than overall accuracy, indicating inconsistent performance across the classes.

3.2 Overlap and Imbalance

The literature on solutions to the imbalance and overlap issue can be placed into three non-exhaustive categories: algorithm-level methods, data-level methods, and combinations of these two (Kotsiantis et al. 2006). Additionally, some have used ensemble methods, which aggregate predictions from multiple models, to deal with overlap and imbalance (e.g. Xiong et al. 2010; Galar et al. 2011; Diez-Pastor et al. 2015). Algorithm-level methods look to adapt how a classifier learns, rather than modifying the data set that it learns on (Alberto Fernández et al. 2018). Such methods include the use of weighted loss functions (e.g. Barandela et al. 2003; Shahee and Ananthakumar 2021), threshold selection (e.g. Johnson and Khoshgoftaar 2019), and cost-sensitive learning (e.g. Domingos 1999; Thai-Nghe et al. 2010).

Though the performance of algorithm-level methods does not depend on the data, their use is harder to generalize since they are specific to certain classifiers (Alberto Fernández et al. 2018). Data-level methods do not have this issue and are, therefore, said to be more versatile (Douzas et al. 2018). The themes that emerge from the literature on data-level methods for overlap and imbalance include feature selection methods, undersampling methods, and oversampling methods. We discuss the literature on these themes in more detail next.

3.2.1 Overlap Metrics

Overlap metrics have been introduced by Oh (2011), Borsos et al. (2018), and Z. Li et al. (2021), which can be used to quantify the amount of overlap present in a feature, a class, or the entire data set. Moreover, these can be used to select features such that the overlap of the resulting data set is minimized. The *R-value* initially introduced by Oh (2011) is equal to the proportion of examples, in either class, who have more than θ nearest neighbors, out

of k , that are of the opposite class. It was found that the R-value of the data set is strongly correlated with the accuracy of classifiers. Improvements that adjust the R-value to account for class imbalance were introduced by Borsos et al. (2018) and Z. Li et al. (2021). These were shown to have successively stronger correlations with the classification performance of a variety of algorithms.

3.2.2 Random Undersampling and Editing Methods

One of the simplest methods for handling imbalance is random undersampling (RUS). In RUS, majority class observations are removed at random until the desired class frequencies are achieved. An obvious concern with RUS is that one risks removing information on the majority class that may be pertinent to making good classifications (He and E. A. Garcia 2009). Many informed undersampling methods have been introduced as solutions to this issue. These include Condensed Nearest Neighbors (CNN) (Hart 1968), Edited Nearest Neighbors (ENN) (Wilson 1972), Tomek Links (TL), (Tomek 1976), One-sided selection (OSS) (Kubat and Matwin 1997), and Near-miss undersampling (NMU) (Mani and I. Zhang 2003). Additionally, an ensemble algorithm, EasyEnsemble, and a sequential algorithm, BalanceCascade, were introduced by Liu et al. (2008), which involve undersampling.

CNN, ENN, TL, OSS, and NMU each aim to undersample majority class examples that can cause confusion for the classifier. These include majority class examples whose neighbors belong to the minority class, who crowd a minority example, or who are near the boundary of the decision region. Removing such examples can correct imbalance and overlap simultaneously. The EasyEnsemble algorithm creates several balanced data sets and trains a classifier on each of them, then combines the output from these learners, creating one ensemble classifier. BalanceCascade creates a sequence of classifiers, each training on a data set from which minority examples that were correctly classified by the last classifier have been removed.

Such algorithms are different from the editing methods previously discussed in that none of the majority class examples are completely ignored.

An upside to editing methods is that the size of the data set, and therefore the computation time, decreases. However, experiments performed by Batista et al. (2004) showed that some undersampling methods provide less accurate results than oversampling, in terms of the area under the ROC curve (AUC). Also, Mani and I. Zhang (2003) note that undersampling can cause a trade-off between precision and recall. Precision is defined as the proportion of correct positives out of all positives *predicted* and recall is defined as the proportion of true positives out of all positive *observed*.

3.2.3 Random Oversampling (ROS) and Its Derivatives

Alternatively, class imbalance can be corrected by simply oversampling minority examples at random until balance is reached. This method is termed random oversampling (ROS) or bootstrap-based oversampling (e.g. He and E. A. Garcia 2009; Yang et al. 2011). However, due to the increased cost of misclassifying points that fall into very specific areas of the feature space, pure oversampling results in decision rules that are too specific, which causes overfitting (He and E. A. Garcia 2009). A variety of more sophisticated oversampling methods have therefore been introduced. These include methods for estimating the distribution of the minority class, uninformed oversampling methods, which do not take characteristics of the feature space into account when introducing new minority examples, and informed oversampling methods, which target areas of the feature space that would benefit the most from having more minority examples. We discuss the most popular oversampling method to date next.

3.2.4 The Synthetic Minority Oversampling TEchnique (SMOTE)

As noted by (S. Garcia et al. 2016), the most popular and influential oversampling method is the Synthetic Minority Oversampling TEchnique (SMOTE) introduced by Chawla et al. (2002). For each minority class example, SMOTE creates synthetic examples by randomly selecting a certain number of points from all those that lie along the line segments formed by the minority example and its k -nearest neighbors from the minority class. The synthetic point generated by SMOTE can be defined as $z = x_0 + w\Delta$, where $w \sim \text{Uniform}(0, 1)$, $\Delta = x_i - x_0$, x_i is the feature vector for the nearest neighbor used, and x_0 is the feature vector for the minority example of interest. More plainly, this method adds back a random proportion of the difference between the point of interest and one of its nearest minority neighbors, and labels this as a new point in the minority class. The k -nearest minority neighbors of a point x_0 are those points x_1, \dots, x_k that belong to the minority class and have the k smallest distances from x_0 . The distance is calculated using a metric of choice (e.g. Euclidean distance). A graphical example of the SMOTE in 2D is given in Figure 3.4.

In order to deal with mixed data SMOTE-NC was proposed by the authors of SMOTE, where NC stands for nominal-continuous. In this adaptation, the standard deviations of all quantitative variables is first calculated. Then, for a given reference point, the median of these standard deviations is added to the distance, calculated using only quantitative variables, between it and another point each time their levels of a given categorical variable differ. We will refer to both of these algorithms more generally as SMOTE when discussing them throughout but our discussions about SMOTE are also applicable to SMOTE-NC. When applying the algorithm we will use the appropriate version for the type of dataset.

Evidently overlap and imbalance presents a challenge to the predictive component of our overall goal. We aim to use data-level methods to overcome this challenge. In order to maximize their usefulness it is important that we combine them with more complex statistical

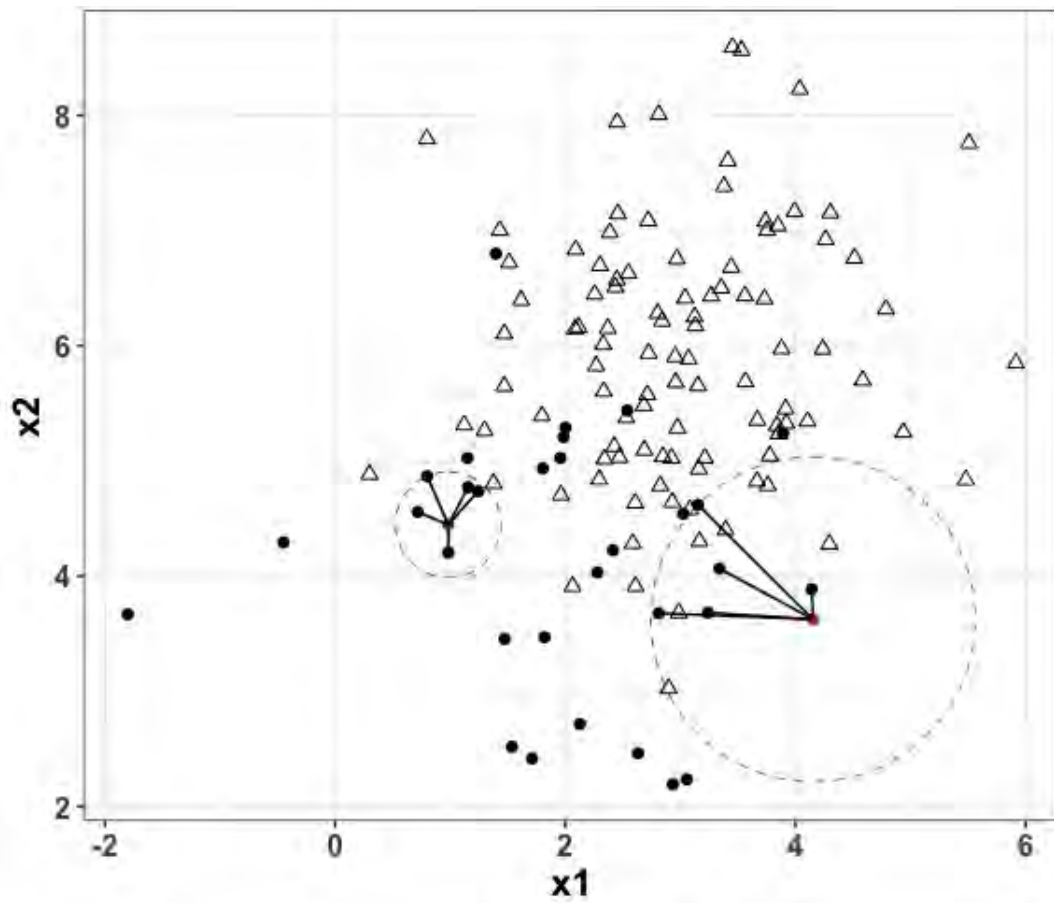


Figure 3.4: Visual of the SMOTE. Minority class points are black circles while majority class points are the open triangles. For the two minority class examples of interest (in red), synthetic points are randomly generated along the solid lines joining them to their 5 neighbors.

learning models with more predictability, though these may be more difficult to interpret. In the next section we will discuss more complex statistical learning models, whose characteristics can also help us to overcome the problem of overlap and imbalance.

3.3 Statistical Learning Methods for Prediction

Due to the two-fold nature of our problem it is necessary to discuss the details of a few statistical learning algorithms that we will use to tackle the predictive component of the problem. Recall our earlier discussion on accuracy versus interpretability, which motivated the need for both simple interpretable models, like logistic regression, and complex, but possibly more accurate, models, like random forests and neural networks. Use of these more complex models can also help us to tackle the overlap and imbalance issue. We will now briefly discuss the underlying details of these two models. Without loss of generality, we discuss them in the context of a binary classification problem. However, it should be noted that random forests and neural networks can be used for a wide variety of classification and regression problems. For a more in-depth discussion of these methods the reader is encouraged to read *An Introduction to Statistical Learning* (James et al. 2013) and *Elements of Statistical Learning* (Hastie et al. 2009).

3.3.1 Random Forests

Random Forests is a tree-based statistical learning algorithm, first introduced by Breiman (2001), for performing regression and classification tasks. Though the original random forests algorithm was introduced over two decades ago, it has continued to grow in popularity through various applications and adaptations (e.g. Hothorn et al. 2006; Lakshminarayanan et al. 2014; Belgiu and Drăguț 2016; Zeini et al. 2023). In order to understand the random

forests algorithm, we must first understand the concept of decision trees.

A basic classification tree partitions a feature space, X , into k regions, R_1, R_2, \dots, R_k with the aim of observations in each region having similar class labels. If an unseen test point, X_0 , falls into region R_t , then the class frequencies for R_t are used to estimate the class probabilities of X_0 . Furthermore, the class occurring most frequently in R_t - that is, the majority vote, is taken to be the predicted class for X_0 .

The following steps can be used to create the regions R_1, R_2, \dots, R_k :

1. Search every value of every predictor of X for the value that, when used for splitting, produces two groups R_1 and R_2 with the greatest improvement in purity - that is, with the most members belonging to the same class in each region as compared to X .
2. Repeat step 1 for each of R_1 and R_2 .
3. Continue splitting each subdivision until some *stopping criteria* is met (e.g. there are few observations in a region)

An obvious metric for quantifying the impurity of a given node in the tree is the *classification error*. This is defined as the proportion of observations in node m that are not in the most common class, or c classes. That is,

$$E_m = 1 - \max_c(\hat{p}_{mc}) \quad \text{where } \hat{p}_{mc} = \text{proportion of observations in node } m \text{ in class } c.$$

Suppose that we have 100 observations of a variable, X , and their classes, Y , labeled as A or B . Consider the two candidate trees given in Figure 3.5, formed by splitting on two different values of X . For Tree 1, the majority vote in R_1 is class B, and the 10 points belonging to class A are misclassified. In R_2 of Tree 1, the majority vote is class A, and 10 points in class B are misclassified. Therefore, the overall classification error of Tree 1 is $20/100 = 0.20$.

Likewise, for Tree 2 the overall classification error is $(20 + 0)/100 = 0.20$.

Though these two trees have the same classification error, R_2 of Tree 2 is a totally pure *terminal node* since all observations belong to the same class. This tree will, therefore, produce more certain predictions, with predicted probabilities closer to 0 or 1. However, this added advantage is not captured by the classification error.

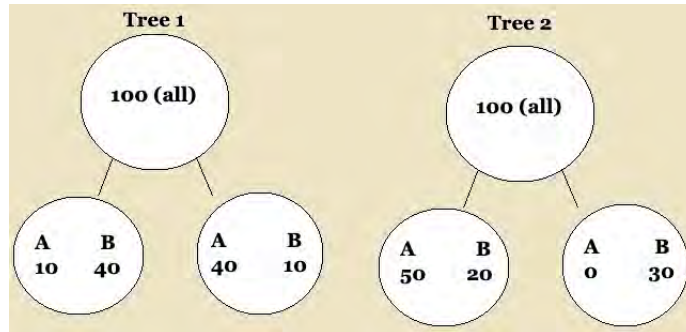


Figure 3.5: Two candidate decision trees yielding the same classification error. However, Tree 2 has greater purity.

An alternative to the classification error is the *Gini Index* (G). This is defined as

$$G_m = \sum_{i=1}^c \hat{p}_{mc}(1 - \hat{p}_{mc}) \quad \text{where } \hat{p}_{mc} = \text{proportion of observations in node } m \text{ in class } c.$$

Suppose we perform an experiment where there are c possible outcomes. Outcome j has probability of occurrence p_j , where $j = 1, \dots, c$. If we reproduce the experiment two independent times, then the probability of observing j twice is $p_j \times p_j$. The probability of observing *any* category *twice* in two independent runs is $\sum_{j=1}^c (p_j \times p_j)$. The probability of observing two *different* outputs is thus

$$1 - \sum_{j=1}^c (p_j \times p_j) = \sum_{j=1}^c p_j - \sum_{j=1}^c (p_j \times p_j) = \sum_{j=1}^c p_j - (p_j \times p_j) = \sum_{j=1}^c p_j(1 - p_j) = G.$$

Therefore, the Gini index measures impurity by quantifying how likely we are to obtain two different outcomes from the same node.

Denote $G_{R_{tL}}$ and $G_{R_{tR}}$ as the left and right *child nodes* created by splitting R_t , respectively. Other than a small node size, a small decrease in G can also be used as a stopping criteria. For a given region, R_t , if no split exists such that $G_{R_t} - G_{R_{tL}} + G_{R_{tR}} < \gamma$, then we may decide not to split that region any further. In practice, a computationally inexpensive way to fit a single decision tree is to grow the tree as large as possible and then work back and remove any splits that did not meet our threshold γ . This is called *cost-complexity tuning* since it decreases the complexity of the tree, increasing its generalizability to test data, and lowering our risk of overfitting. Despite their simplicity, single decision trees are known to have poorer performance than their successors, such as bagging, random forests, and boosting (e.g. Freund 1995; Breiman 1996; Bartlett et al. 1998; Bühlmann and B. Yu 2002; Ali et al. 2012). Therefore, we choose to point readers to Patil et al. (2010) for further discussion on pruning algorithms, and we continue on with a discussion of bagging and random forests, which do not require pruning.

There are three sources of error when estimating f , the relationship between our data and one or more response variables. The irreducible error, ϵ , that we discussed earlier, is independent of the data X . Therefore, it cannot be reduced using information from the data during training. Another source of error, called the *bias*, comes from failing to capture important features of f in our estimate \hat{f} . This is termed *underfitting*, and it *can* be reduced during training. Conversely, another source of error is called the *variance*, which comes from an estimate \hat{f} that captures the unhelpful random noise in the data, not just the important features. This is termed *overfitting*, and it can also be reduced during training.

However, by definition, the bias and variance cannot be reduced simultaneously during training. To reduce the bias, we must increase the complexity of \hat{f} by learning more about its form based on the training data, but this risks overfitting. This is known as the *bias-variance trade-off*. In the context of decision trees, it is known that the results of a single decision tree strongly depend on the data that they are trained on (e.g. Breiman 1996) - that is, they

have high variance from one training set to another.

One solution to this issue, proposed by Breiman (1996), is to fit multiple decision trees to bootstrap samples of the training set, and then, for a given example, predict the majority vote of its classifications from every tree as its class. This method is called *bagging* and it can be used with any underlying model, not just decision trees. Since the prediction is an average of the results of each tree, the variance of the bagged model, from one training set to another, is reduced. Additionally, the probability that an observation does not appear in a bootstrap sample of size n is $(\frac{n-1}{n})^n = (1 - \frac{1}{n})^n \rightarrow e^{-1} \approx 0.37$. Therefore, each observation is left out of the bootstrap samples used to create about 1/3 of the trees. Such observations are termed *out-of-bag* for a given tree, and these can be used to estimate the test error of the bagged model, eliminating the need for an additional validation set. Moreover, these can be used to evaluate the error of the bagged model as more trees are added, in order to determine how many trees are needed to achieve desirable performance without overfitting.

Bagged decision trees are most useful when changes to the training data cause non-trivial changes in the resulting decision tree - that is, when the individual trees vary from one bootstrap sample to another. However, when a few predictors account for most of the variation in the response variable, these predictors are usually selected in all trees in the bagged model, decreasing the variability from one tree to another. When models are correlated in this manner, the bagged models will have high variance, returning us to the original issue.

In order to increase the variability from one tree to another, we can force the model to consider only m randomly selected predictors, out of p , and search these for the best split. This restriction reduces the correlations among trees, decreasing the overall variance of the bagged model. This method is termed *random forests*, and bagging is a special case of it where $m = p$. When there are a large number of strong predictors, selecting a small value for m is best. This will ensure that weaker predictors are not overlooked amongst all of the trees and, therefore, the trees will be more diverse. Similar to the case of bagging, the

out-of-bag error can also be calculated with random forests and used to determine the ideal number of trees.

Due to the complex nature of random forests and bagging, interpretability, an understanding of *why and how* the models make certain predictions, can be lost. Nevertheless, there are at least two interpretable values that can be calculated from random forest models: the mean decrease in the Gini index and permutation-based accuracy. Specifically, these are used to understand how important each predictor is to the certainty and accuracy of our predictions, respectively, if at all. For a given variable, its mean decrease in the Gini index is the total decrease in node impurities from splitting on that variable as each tree is built, averaged over all trees. Its mean decrease in the permutation-based accuracy is obtained by calculating the difference in the prediction accuracy on the out-of-bag portion of the data, before and after permuting the values of that variable, averaged over all trees, and normalized by the standard deviation of these differences.

3.3.2 Neural Networks

While random forests are based on single decision trees, neural networks are inspired by the biological neural networks of the brain. For brevity, we discuss the *feed-forward neural network* in which output from former nodes are only used in latter nodes, and connections between nodes do not form a cycle. We believe that these less complex neural networks may help us tackle the predictive component of our problem. Recall that the technical issue with the predictive component of the problem was binary classification with an imbalanced response variable and overlap in the feature space. Neural networks overcome the problem of feature selection by finding the feature representation that most minimizes a given loss function (Goodfellow et al. 2016), and use of this alternative feature representation could help us overcome the issue of overlap.

The *input layer* of a single-layer feed-forward neural network is made up of p features $X = (X_1, X_2, \dots, X_p)$, each of which are fed into K *hidden units*, or *neurons*, giving a certain output. The output for the k -th hidden unit, called the *activation*, is defined as

$$A_k = h_k(X) = g\left(w_{k0} + \sum_{j=1}^p w_{kj}X_j\right), \quad k = 1, \dots, K,$$

where $g(z)$ is a nonlinear *activation function* that we select before fitting the model. Each A_k is, therefore, a transformation of a linear combination of the original features. These activations are then fed into the *output layer*, resulting in the following linear regression model for the K activations

$$f(X) = \beta_0 + \sum_{k=1}^K \beta_k A_k.$$

A visual representation of the neural network is given in Figure 3.6.

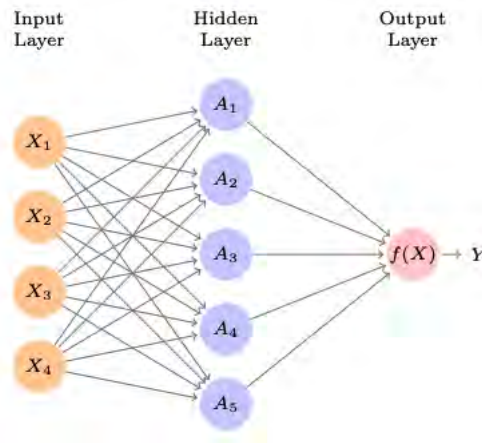


Figure 3.6: Visual representation of a single-layer feed-forward neural network for data with $p = 4$ features, $K = 5$ hidden units, and a single output layer. The input layer (yellow) is made up of the features, each of which is provided to the hidden units in the single hidden layer (violet). The output of the hidden units are the activations. The output layer (red) provides a linear combination of these. Credit: *An Introduction to Statistical Learning (ISLR)* (James et al. 2013).

Of the many options for the activation function, $g(z)$, the most popular include the *sigmoid* activation function, also known as the *logistic function*, and the *rectified linear unit (ReLU)*

activation function (e.g. Nwankpa et al. 2021). These are respectively defined as

$$g(z) = \frac{1}{1 + e^{-z}} \quad \text{and} \quad g(z) = \max(0, z).$$

The sigmoid function can be used to convert values to probabilities. Therefore, when the task is binary classification it can also be used for the output layer. However, its drawbacks include slow convergence and non-zero centered output (e.g. Nwankpa et al. 2021). Meanwhile, the ReLU function is very close to being linear, giving it nice properties for *gradient descent*, an optimization method used to estimate the parameters β_0, \dots, β_K and w_{10}, \dots, w_{Kp} from the data.

Mutli-layer neural networks have additional hidden layers with fully-connected nodes to the next layer, similar to the initial input layer. The algorithms used to train neural networks are quite complicated and this method is not the focus of our research. Therefore, for further discussion on back-propogation, optimization algorithms, and other types of neural networks, such as convolutional and recurrent neural networks, we refer the reader to *An Introduction to Statistical Learning (ISLR)* (James et al. 2013) for preliminary reading and *Deep Learning* (Goodfellow et al. 2016) for deeper reading.

The models that we have discussed in this section will help us to tackle the second of our two main goals for this research. As a reminder, these were to (1) construct models that can adequately describe the relationship between graduation or retention and the amount of gift aid received, while taking other variables into account, and (2) develop predictive models that can be used to determine how the predicted success of a given student changes with gift aid, and how gift aid impacts these predictions. We will now discuss the results of our data analyses towards prediction. We will also discuss the issue of imbalance and overlap and investigate the performance of current solutions on our data when combined with these more complex statistical learning methods.

3.4 Statistical Analysis Towards Predictions

The technical problem of developing good predictive models in the presence of overlap in the feature space and imbalance in the response variable must be tackled directly to achieve our overall research objectives. Therefore in this section we address this issue by using more advanced statistical/machine learning models to predict student success well. We aim to first develop models that have better predictions and then we will study these to recover conclusions that can still contribute to the inferential component of the problem.

3.4.1 Single Decision Trees

Our exploratory analysis of the data showed that there are some observations with feature values that most often match those of observations with the opposite class label. Since classification models aim to partition our feature space into sections that are indicative of class labels, these observations pose a challenge to the classifier. However, we were able to find some of these observations by examining higher order interactions between covariates in our data set (e.g. studying retention rates by need *and* race in Table 2.11). Though logistic regression models are limited in this respect (e.g. Levy and O'Malley 2020), by design, decision trees are able to model these higher order interactions. Each terminal node in a tree is reached by following a sequence of decision rules each involving a single covariate. Therefore, a deep tree that uses a variety of covariates models higher order interactions between covariates. Modeling these higher order interactions could lead to a model that better handles the overlap in our feature space. We include high school GPA, class rank, total first year gift aid, binary gender, Pell-eligibility, race, residency, first-generation status, and severity of first-year need.

In order to fit a decision tree, we must determine the *hyperparameters* of the model. These

are the parameters under which the tree will be created. The primary hyperparameter to tune is the *complexity parameter* (`cp`) which effectively prunes the tree. In the context of binary decision trees, any split that does not improve the Gini index by a factor of `cp` will not be attempted. The larger `cp`, the more conservative our tree growing process and only splits that results in large increases in purity are attempted. Other important constraints include the minimum number of observations that must exist for a node to be further split (`minsplit`), the minimum number of observations in any terminal node (`minbucket`), the maximum depth of any node of the final tree (`maxdepth`). Each of these is related and they control how large the tree is grown.

We used cross-validation to decide on the correct hyperparameters for the decision tree. Data were split into 10 folds. For each combination of `cp` and `minsplit`, a decision tree was fit to 9 of the 10 folds, then predictions were obtained for the left out fold along with predictive metrics. This is repeated until each fold has been left out and the predictive metrics are averaged. Figure 3.7 provides the cross-validated average accuracy, area under the curve, sensitivity, and specificity of trees fit using each combination of `cp` (`x`) and `minsplit` (`y`). Including the standard deviations for each estimate of test error is infeasible given the number of cross-validation iterations. However, the median standard deviation for the average accuracy, AUC, sensitivity, and specificity values reported in Figure 3.7 were 0.0095, 0.0413, 0.0098, and 0.0510, respectively. In each plot, the cells towards the top-right correspond to less complex models while cells towards the bottom-left correspond to more complex models.

The plots in the bottom row of Figure 3.7 indicate disagreement between the models. Complex models with lower values for `cp` produce lower sensitivity but higher specificity. However, the specificity only goes as high as 0.25 and the sensitivity only goes as low as 0.94. Therefore, the models that perform the worst on the positive class still perform very poorly on the negative class, making the trade-off not worth it. This is also reflected in the plot for overall accuracy, in which the highest values near 0.85 most often occur near the center. It

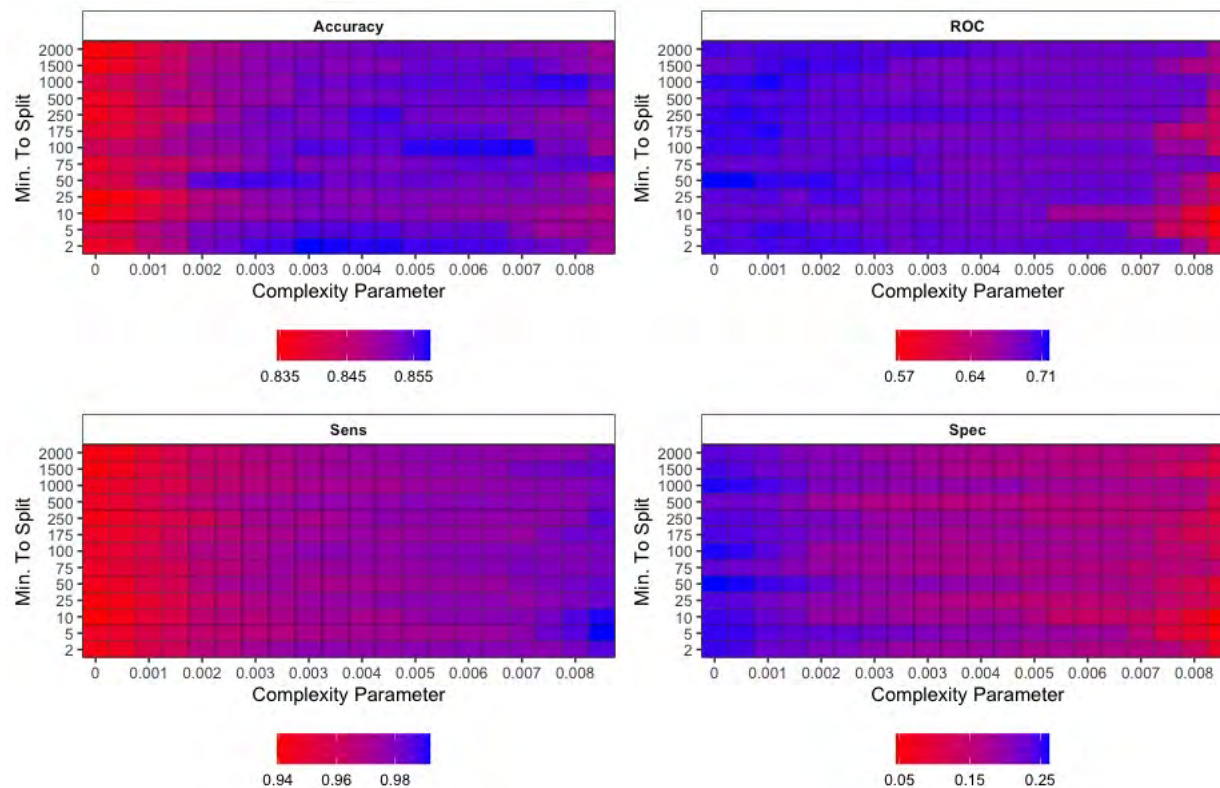


Figure 3.7: Average performance metric for each combination of `cp` and `minsplit` based on 10-fold cross validation. Accuracy, ROC (the area under the curve), sensitivity, and specificity are provided. The right-top indicates less complex models while the left-bottom indicates more complex models. The direction towards which each metric increases does not agree across all metrics, indicating that a tradeoff is inevitable.

should be noted that if we were to predict all observations in the training set as retained rather than using an actual classification model, our accuracy would be 0.848. Therefore, these models are performing similarly to or worse than random guessing using empirical probabilities. Given these disagreements, we elect to use the ROC to select the best set of hyperparameters since it is indicative of performance on both the positive and negative class.

The hyperparameters producing the simplest model whose AUC is within one standard deviation of the largest AUC will be used. Simplicity is first determined by the largest `cp` and then the largest `minsplit`. The final hyperparameters used were therefore `cp = 0.0064` and `minsplit = 25`. Figure 3.8 and Table 3.4 provide the resulting decision tree, and predictive metrics for the test data, respectively.

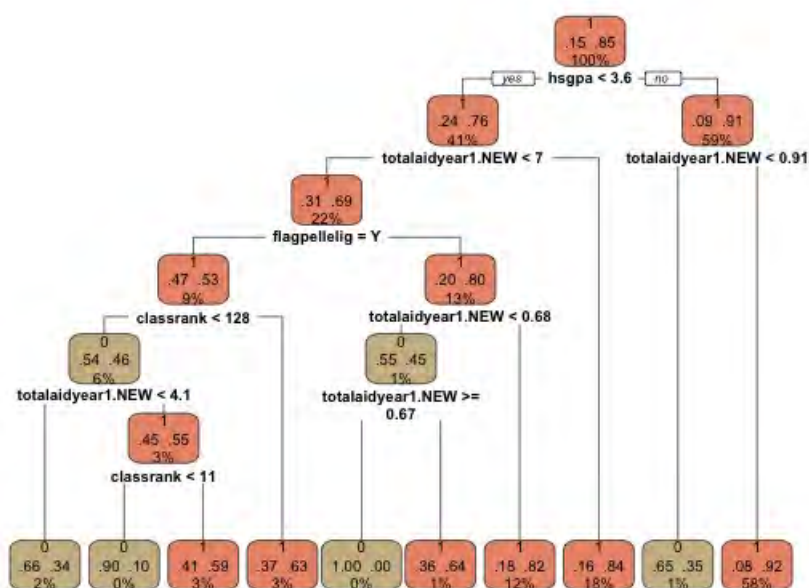


Figure 3.8: Decision tree for first-year retention. Amongst all variables considered, high school GPA was deemed to be the most influential factor. First-year gift aid was most often used to separate class labels. Salmon colored nodes correspond to predicting retention and brown nodes desertion. The two decimals in each node give the proportion of observations in that node that deserted and returned, respectively. The percentage in each node gives the percent of observations falling into the node out of all students in the training set. There are some discrepancies due to rounding.

Accuracy	Bal. Accuracy	Sensitivity	Specificity	F-Score	AUC	Kappa
0.853	0.549	0.985	0.112	0.919	0.696	0.146

Table 3.4: Performance metrics for predictions of test data set using single decision tree. Balanced accuracy is much lower than overall accuracy, indicating inconsistent performance across the classes.

Recall that we included high school GPA, class rank, total first year gift aid in thousands of dollars, binary gender, Pell-eligibility, race, residency, first-generation status, and severity of first-year need in our model. Amongst all variables considered, high school GPA was deemed to be the most useful variable for creating an initial split of the data. The first decision rule tests whether high-school GPA is below 3.6. 41% of students did meet this criteria and 59% did not. After high-school GPA, the most useful predictor of retention was first-year gift aid. This variable was most often used to separate classes throughout the tree as well.

This model indicates that first-year retention is influenced by an intricate relationship between academic preparedness as measured by high-school GPA and class rank, financial need as measured by Pell-eligibility, and the amount of first-year aid awarded. The purest terminal node, occurring on the far right, is allotted to high performing students with a high school GPA above 3.6 and at least 910 dollars in gift aid. 8% of students in this node deserted while 92% returned. Students in this node make-up about 58% of the training set. However, students who were high performing but received less than 910 dollars in aid are predicted to desert. This indicates that the success of students with strong academic backgrounds can still be negatively impacted by receiving low amounts of aid.

If high-school GPA is below 3.6 for a given student but they received more than 7000 dollars of gift aid in their first year, then they are predicted to return after their first year. 18% of all students in the training set fell into this node. Again, this indicates that low academic performance may not immediately results in student failure but that this depends on how much aid they receive. Summarizing the results of the model more generally we conclude that Pell-eligible students with low high school GPAs who also had low amounts of aid and low class ranks were predicted to desert.

While these results certainly contribute to the inferential component of the problem, the predictive component of the problem is not helped nearly as much. This can be noted by examining the purity of the terminal nodes in the trees. Terminal nodes belonging to retained students contain up to 41% students that were *not* retained and those belonging to the deserted class contain up to 35% of students that *were* retained. Over 88% of the data fall into the most pure nodes - that is, those with over 80% of observations belonging to the same class. However, many observations belonging to the minority class (desertion) also fall into these nodes. This issue is also reflected in Table 3.4 where the test set sensitivity was 98.5% while the specificity was only 11.2%. The balanced accuracy (54.9%) was also much lower than the overall accuracy (85.3%) indicating that there was unequal predictive

performance between the two classes.

A calibration plot of predictions for the test set is provided in Figure 3.9. The predicted probabilities of retention were binned into intervals from 0 to 1 by 0.05 with right-side inclusion. The average predicted probability of retention (x) and true retention rate (y) was calculated for observations in each bin. These are plotted along the purple line in Figure 3.9. For reference, the line $y = x$ is also provided in black. Point labels give the number of observations in each bin and the interval for the bin. Intervals that did not contain any observations were omitted.

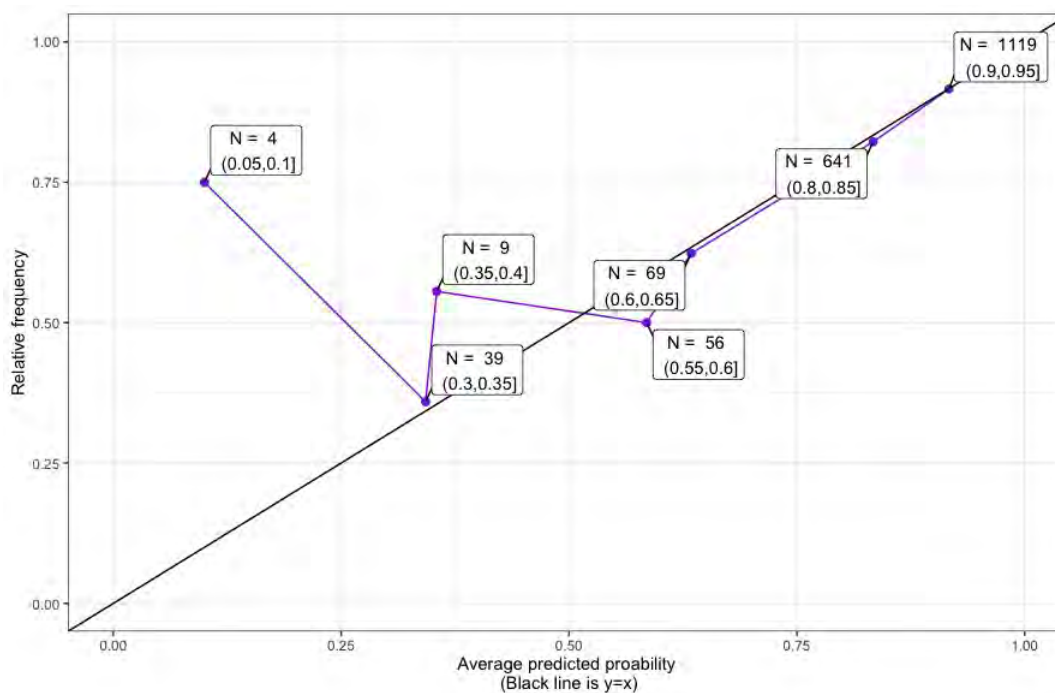


Figure 3.9: Calibration plot for test set predictions from single tree trained using cross-validated hyperparameters. Predicted probabilities of retention were binned from 0 to 1 by 0.05. Purple points give the average predicted probability (x) and retention rate (y) by bin. The line $y = x$ is also provided in black. Labels give the number of observations and bounds of the interval. Intervals without observations are omitted.

A model that is well *calibrated* will result in points that lie very close to or directly on the reference line as we scan the x-axis from left to right. Points falling above the reference line indicate under-prediction because the average predicted probability of retention would be lower than the relative frequency of retention. Similarly, points falling below the reference

line indicate over-prediction. Not surprisingly, the results show that larger predicted probabilities from the model are better calibrated than small predicted probabilities. The relative retention rate is noticeably larger than the average predicted probability for the first and third bins pictured. This indicates that our model is more reliable for predicting student success than failure.

In order to understand how student success can be improved by increasing aid, we must first be able to identify *when* a student is unlikely to succeed - that is, we must be able to predict student failure as well as student success so that students who actually need help can receive it. The lack of information that we have on students that fail creates a barrier to achieving this however, therefore, more complex methods must be used to tackle the predictive component.

Earlier we discussed a simple but naive solution to correcting the imbalance in our response variable called random oversampling (ROS). This method involves selecting a certain percent of minority class examples from the original dataset, by randomly sampling with replacement, and then replicating these rows in the dataset. The number of minority class examples can be increased by any percent with this method. If the number of majority and minority class examples are n_{maj} and n_{min} , respectively, then to achieve complete balance the minority class should be oversampled

$$\left(\frac{n_{maj} - n_{min}}{n_{min}} \right) \times 100\%.$$

Regardless of how one oversamples, the majority class accuracy will likely decrease since the cost of misclassifying the minority class increases. However, ROS increases the cost of misclassifying a *single* minority example in proportion to how many times it is oversampled. In order to decrease misclassification costs, classifiers trained on ROS data must therefore correctly classify points in very small regions of the feature space. This leads to overfit classifiers - that is, classifiers whose decision regions do not generalize well to unseen data.

To determine whether ROS can positively impact our decision tree classifier, we increased the number of minority class examples in our training set by a certain percentage and fit a decision tree. We pruned the tree using 10-fold cross-validation to select the complexity parameter and then calculated class-specific and overall accuracies from predictions on the test data. This was repeated 250 times for each percentage of oversampling. The results are plotted in Figure 3.10.

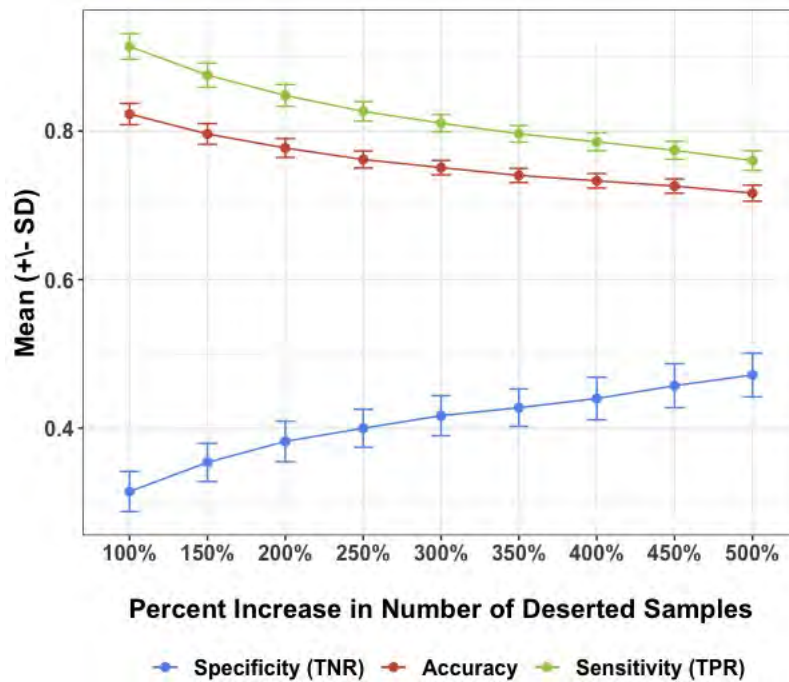


Figure 3.10: Overall and class-specific performance metrics (y) resulting from oversampling the minority class a certain percentage (x). Specificity increases but sensitivity and overall accuracy decrease, indicating an disadvantageous trade-off between class-specific accuracies. Trees were pruned using 10-fold cross-validation to select the complexity parameter and the process was repeated 250 times.

Before oversampling, the accuracy, sensitivity, and specificity were 85.3%, 98.5%, and 11.2%, respectively as reported in Table 3.4. After oversampling the minority class at 100%, the accuracy and sensitivity decrease to 82.5% (± 0.0138) and 91.5% (± 0.0170), respectively, while the specificity increases to 31.7% (± 0.0257). Despite this initial large increase in specificity, oversampling at 200% and beyond does not increase the specificity by any more than 3% each time. Even at 500% oversampling, the specificity only reaches 46.9% (\pm

0.0288), a difference of about 15% compared to unbalanced data. Meanwhile, the sensitivity decreases to 76.2% (± 0.0134) at 500% oversampling, a difference of 22% compared to unbalanced data.

There is also more variability in the performance metrics for the minority class due to the added randomness introduced by ROS. Additionally, trees fit to oversampled data were consistently much larger than the tree fit before oversampling. This happened because the classifier was learning very specific rules so as to not misclassify single points that largely increase the error due to their replication. Larger decision trees are more prone to overfitting as we saw when we applied these to the test data. Moreover, we found that the actual decision regions of the resulting trees were not much different from that of the original tree - that is, the model was not learning new decision rules indicative of desertion. The models simply labeled decision regions as ‘belonging’ to the minority class more often and further partitioned the same initial decision regions to capture our specific dataset. Coupled with the size of the trees, these issues made interpretations difficult so that oversampling also detracted from the inferential component of the problem.

These results show that there is a trade-off between accuracy on the negative class and positive class. Since the losses on the majority class outweigh the gains on the minority class, the overall accuracy decreases. Losing more accuracy on the majority class than is gained on the minority class defeats the predictive purpose of our research, which is to predict student success well for *all* students.

The initial results from our decision tree in Figure 3.8 showed that it was indeed able to model higher-order interactions between covariates. This led to a better understanding of the intricate relationship by which these covariates determine student success. However, due to the imbalance in our dataset, decision trees do not pay careful enough attention to misclassified points in the minority class, and they find patterns that are more indicative of retention than desertion. We would like to use an oversampling method that provides a

more advantageous trade-off than that observed so far. We combine SMOTE and random forests next to overcome the issues with ROS and single decision trees, respectively.

3.4.2 Random Forests + SMOTE

Single decision trees become less biased as they are grown larger but they also became more variable. Bagging is a method where the results of many unpruned decision trees, fit to bootstrap samples of the training data, are averaged. Due to the averaging, the bagged model has less variability than individual decision trees and is therefore less likely to overfit to the training data. Random forests further decreases the variability by searching through $m \leq p$ candidate covariates to determine the best choice for each split. When $m = p$, this amounts to bagging. Random forest effectively decorrelates the individual trees, thereby capturing more of the variability between trees. In the context of our problem, using random forests allows us to grow larger decision trees, thereby finding more specific minority class regions of the feature space, but without as much of a risk of overfitting.

Hyperparameters of the random forest algorithm include the number of candidate predictors to consider at each split point `mtry`, the number of trees to grow `ntree`, the minimum size of terminal nodes `nodesize`, and the maximum number of terminal nodes trees in the forest can have `maxnodes`. Stratified sampling can be used to preserve the distribution of the response variable. ROS, RUS, or a mixture of both can also be applied to each bootstrap sample by using weights. In addition to the issues with ROS that we identified in the last section, when the minority class is oversampled many times there will be some minority class observations who are present in every tree in the forest, therefore these will have no OOB estimate for error. Rather than oversampling, we elect to apply SMOTE before fitting random forests. By way of comparison we also fit random forests to the unbalanced data.

Random forest models can handle much larger amounts of data than we have trained on

thus far. In addition to the covariates used in single decision trees we also include high school graduating class size, whether a student entered college immediately after graduating, number of college credits completed before entering, SAT math score, whether the student received a scholarship in their first year, college of entry, and more. We used a total of 25 covariates and since there were a mixture of categorical and numeric variables we applied SMOTE-NC, which we refer to more generally as SMOTE.

A total of 5 neighbors were used in SMOTE to generate 4 synthetic examples from each minority example. There were 3445 minority examples in the oversampled dataset while the unbalanced dataset had 689. The number of majority examples was 3843 giving an imbalance ratio of 5.578 in the unbalanced set and 1.116 in the oversampled dataset. In order to train the models we used 5-fold cross-validation to select the best values from `ntree` = 50, 500, 1000, `nodesize` = 1, 10, 100, 1000, and `mtry` = 1, 10, 25. Results might have been different if we also trained for the best amount of oversampling and number of neighbors to use for oversampling, however, we choose to focus on training parameters of the random forest model for now.

The results of training are plotted in Figure 3.11. Each cell represents the average AUC, sensitivity, or specificity from fitting a random forest to each of 5 folds of the training data with the corresponding values of `mtry`, `ntree`, and `nodesize`. Due to the presence of synthetic data some of the performance metrics will be inflated. Models fit to oversampled data outperformed those fit to unbalanced data with respect to all metrics except for the sensitivity, which was always greater than or equal to 0.988. Metrics from models fit to oversampled data also had lower variability most of the time. Additionally, models with 500 or 1000 trees, a node size of 1 or 10, and `mtry` = 10 produced larger values for the AUC and specificity in most cases. Sensitivity ranged from 0.988 to 1.000 in all cases. Meanwhile specificity ranged from 0 to 0.755 for unbalanced data and 0.775 to 0.95 for oversampled data.

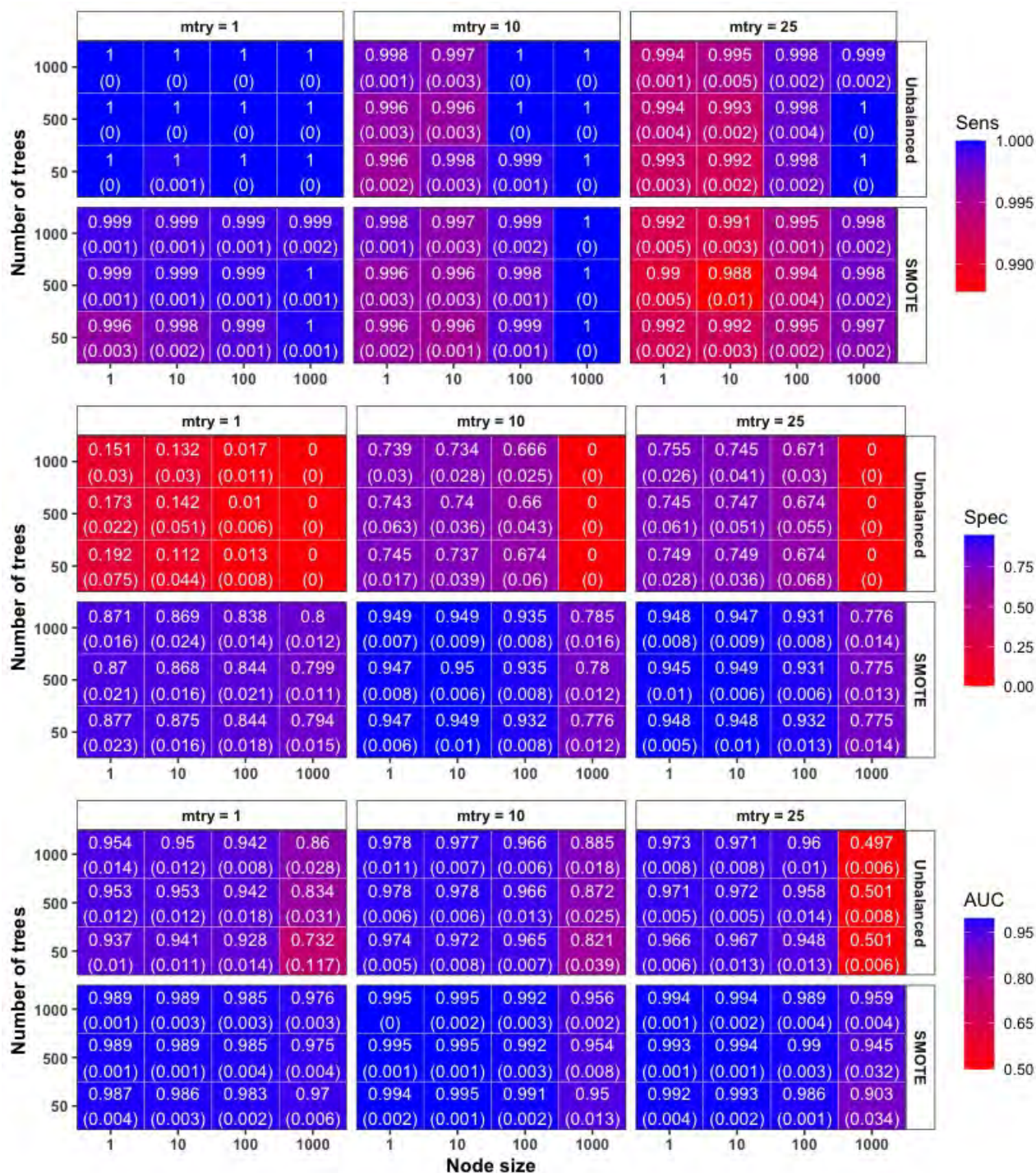


Figure 3.11: Cross-validated metrics from random forest models fit to unbalanced training data (top row of panels) and after applying SMOTE (bottom row of panels). A total of 5 neighbors were used in SMOTE to generate 4 synthetic examples from each minority example. Models were trained via 5-fold cross validation to find the optimal values for $mtry$, number of trees (y) and node size (x). SMOTE results may be inflated due to the additional synthetic examples. Note that the color gradient is not the same across all metrics since they did not all have the same range.

Given these results we move forward with the model fit to oversampled data with `ntree = 500`, `nodesize = 10`, and `mtry = 10`. The by-tree out-of-bag (OOB) error rate for this model is plotted in Figure 3.12. This was calculated for all OOB data, non-synthetic OOB data, and the test data. The final OOB error rates for the minority and majority classes were 0.050 and 0.003, respectively, when including synthetic data points. After dropping synthetic data points the OOB error rate for the minority class increased to 0.253. The error rate on the test data for the positive and negative classes was 0.007 and 0.624, respectively. The large increase in the minority class error rate between non-synthetic training examples and testing examples is an indication of overfitting.

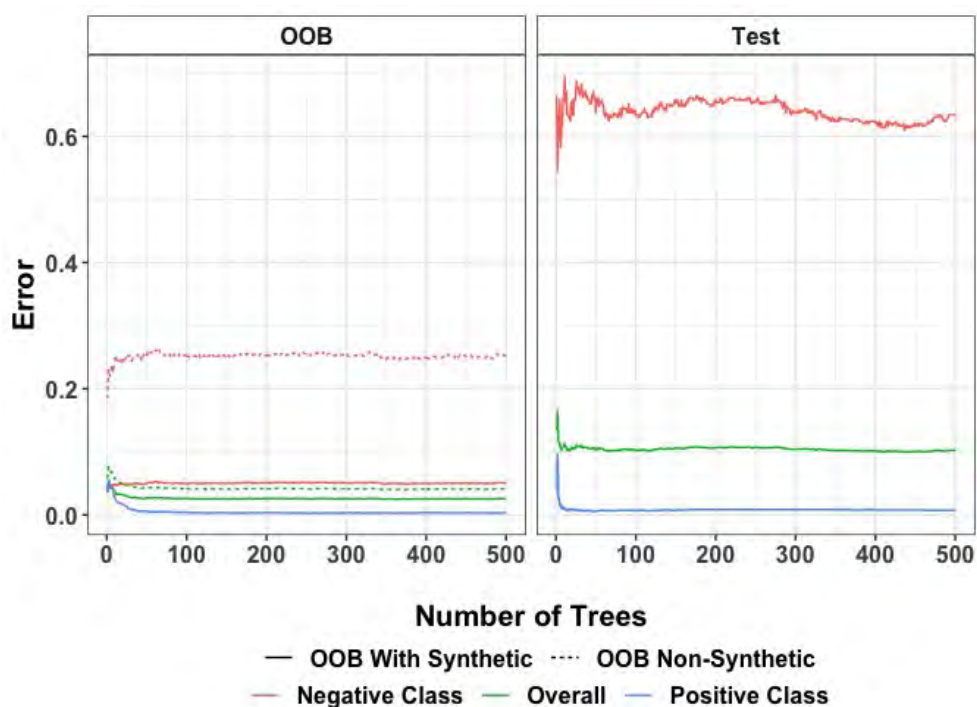


Figure 3.12: OOB and test data error rates by tree. The dotted lines correspond to corrected errors calculated using only non-synthetic data. These were calculated to guard against the reporting of inflated errors alone. The large increase in error for the negative class on test data in comparison to OOB data provides evidence of overfitting.

The use of random forests in combination with SMOTE has improved our predictions in comparison to logistic regression and single decision trees. Table 3.5 provides the predictive metrics from the random forest model, decision trees, and logistic regression. All of the

predictive metrics were higher for predictions obtained using random forests and SMOTE. The largest increases were observed with the specificity, AUC, and Kappa coefficient. About 25% and 26% more minority class examples in the test set were correctly classified than when logistic regression or decision trees was applied without oversampling, respectively. In these cases the overall accuracy only increased by about 6% and 5%, respectively, since there are so few minority examples in the dataset. The balanced accuracy does reflect this improved performance on the minority class however.

Model	Acc	Bal. Acc	Sens	Spec	F-Score	AUC	Kappa
Logistic Regression + No SMOTE	0.842	0.551	0.974	0.128	0.913	0.745	0.144
Decision Tree + No SMOTE	0.853	0.549	0.985	0.112	0.919	0.696	0.146
Random Forest + SMOTE	0.899	0.685	0.993	0.376	0.944	0.960	0.485

Table 3.5: Performance metrics for test set predictions obtained using random forest models trained using data that were first oversampled with SMOTE. Results from single decision tree and logistic regression applied to original training data are also included for reference. The use of random forests and SMOTE improves all performance metrics.

Though this model performs better, its calibration plot shows that it often over-predicts the probability of retention, especially for students that deserted. The predicted probabilities from the model were placed into bins of size 0.05 from 0 to 1 and the average was calculated over each bin. The proportion of students retained in each bin was also calculated. These are plotted on the x- and y-axis of Figure 3.13, respectively. The calibration line consistently falls below the reference line and the difference was worse when the average predicted probability was lower.

There is evidence that the use of a larger threshold than 0.5 could lead to better predictions. This is reflected in Figure 3.14 where the ROC plots shows that a threshold of about 0.86 gives much higher specificity (0.8983) than the default 0.5 whose specificity was 0.3763. Use of this threshold drops the sensitivity from 0.9927 to 0.8913 however, indicating that there is still a trade-off after applying SMOTE. This is because the issue of overlap is not directly tackled by SMOTE, only imbalance.

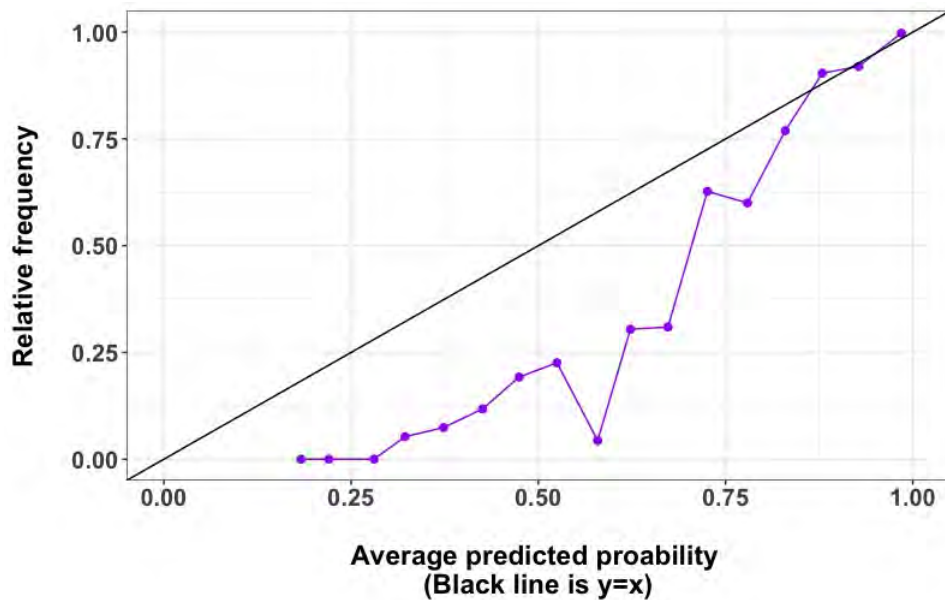


Figure 3.13: Calibration plot for random forest model fit to training data with SMOTE applied. Results were obtained using test data predictions. The model over-predicts often, especially when the relative frequency of retention was actually low.

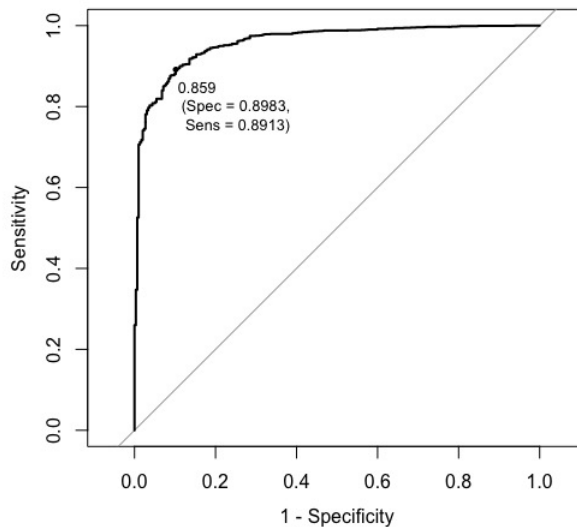


Figure 3.14: ROC plot for random forest model fit to training data with SMOTE applied. Results were obtained using test data predictions. The threshold giving the highest sum sensitivity plus specificity is plotted. Results indicate that a higher threshold could lead to better predictive results overall.

When there is overlap and imbalance is corrected without consideration of the overlap decision rules that should lead to a test point being classified in the majority class will become biased to predict points as belonging in the minority class. This is why we see a decrease in the majority class accuracy. The increase in majority class accuracy is to be expected whenever we oversample, but how much of an increase in specificity we get for this loss will determine if it is worth it. This will also depend on the size of the majority class. When it is very large decreases in majority class accuracy that are small can have a major impact on the overall accuracy.

Though the predictions coming from the use of Random Forest + SMOTE were still negatively impacted by some of the data difficulties in our dataset, we do obtain some useful interpretable output that contributes to the inferential component of the problem. The variable importance plots from this model are given in Figure 3.15. The mean decrease in Gini index is the total decrease in node impurities from splitting on the indicated variable on the y-axis average over all trees. Large amounts indicate that the variable contained split points which were very useful to separating the classes. The mean decrease in accuracy is obtained by permuting the values of the indicated variable before obtaining predictions on the out-of-bag (OOB) data for each tree. This effectively destroys the information in the variable. The difference between this and pre-permutation OOB predictions is calculated and these differences are averaged over all trees and normalized by the standard error of the differences. Large differences indicate that the variable is important to the predictions since loss of its information decreased the accuracy a large amount. These were obtained using data with synthetic examples so they will be slightly inflated in a manner similar to what we saw in Figure 3.12.

In terms of class separability the type of financial aid that a student received for their housing is most useful to separating classes. After this the most useful variables to class separation are the type of tuition that a student was charged, the number of hours they completed

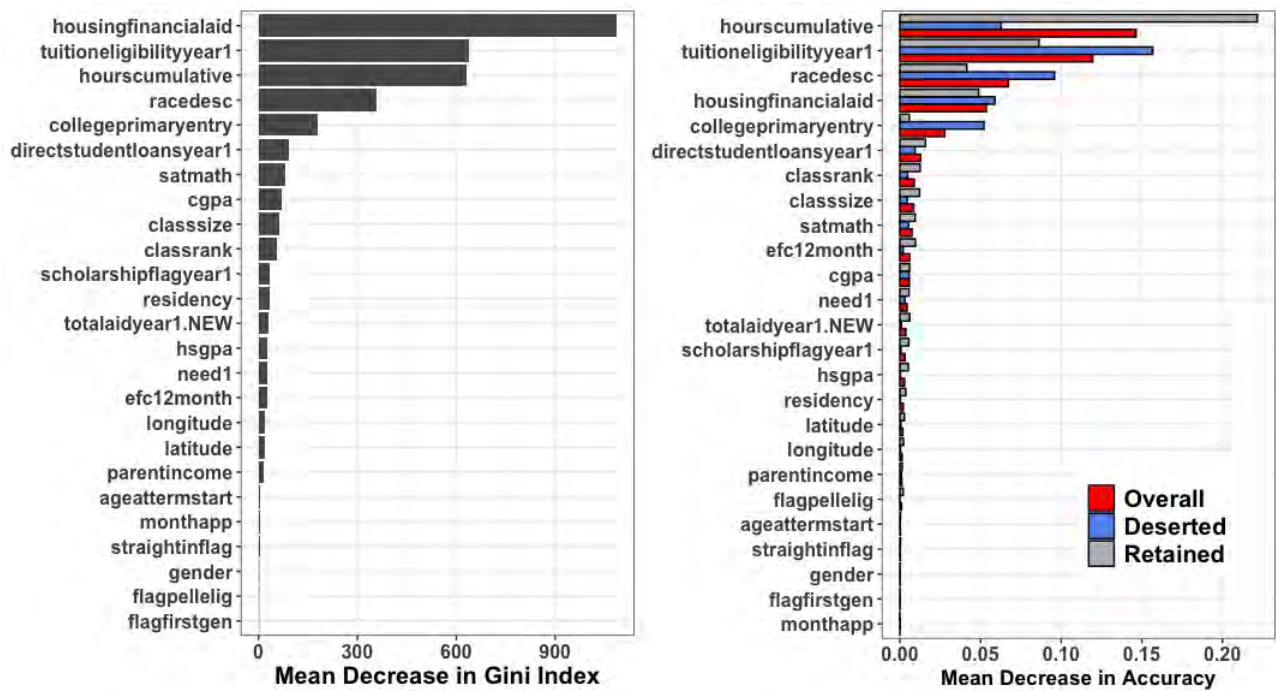


Figure 3.15: Variable importance plots for random forest model fit to training data with SMOTE applied. Results were obtained using oversampled data so values are inflated. Left- and right-hand panels correspond to the importance of the variable to node purity (separating the classes) and overall predictions.

before entering college, their racial group, and the college that they entered. These same five variables are most important in terms of the accuracy of predictions but the order is different. In terms of overall prediction accuracy, the number of hours completed, type of tuition, racial group, type of financial aid for housing, and primary college of a student upon entry are the top five most important variables.

However, the order of importance is not the same when considering the changes in the *class-specific* accuracy. For retained students the type of financial aid awarded for housing is the third-most important factor before racial group. Additionally there are many variables, such as the amount of direct student loans and 12-month expected family contribution (EFC) that are more important for predictions on the positive class than the primary college on entry and these have similar importance measures. Meanwhile the five most important variables to correct predictions of desertion are the type of tuition that a student is charged, their racial group, the number of hours completed, financial aid for housing, and the primarily college they entered into. The changes in the order of importance show that not all variables have the same impact on accuracy for each group. These results also show how using just the decrease in overall accuracy may be misleading because this is large even if the variable is only important for one class. A better metric would be the change in the balanced accuracy.

3.4.3 Neural Networks + SMOTE

We saw that combining random forests with SMOTE gave better predictions with a threshold of 0.5 and that these could be improved with a trade-off if we increased the threshold of about 0.86. By design, neural networks create abstract representations of the feature space which could tackle the issue of overlap and omit this trade-off. To determine if we might be able to obtain better predictions we also applied neural networks to the unbalanced and SMOTE oversampled data. We included as many variables in the dataset as possible with

the only criteria being that they must contain information which would be known after the students first year. Data were split into training, validation, and testing sets, median-mode imputation was applied, and we also one-hot encoded categorical variables to create a strictly numeric matrix.

We selected the combination of hyperparameters from the following that maximized the AUC on the validation set after all epochs were completed: hidden layer dropout rates ($\text{rate1}, \text{rate2} = 0.1, 0.2$), number of units in hidden layers ($\text{units1}, \text{units2} = 43, 21$), and a batch size of 32. The ReLu activation function was applied in the input and hidden units while the sigmoid activation function was applied in the output unit. Binary cross-entropy was used as our loss function with an adadelta optimizer and we trained each model for 100 epochs. The set of hyperparameters that maximized the AUC when data were not oversampled with SMOTE were $\text{rate1} = 0.1, \text{rate2} = 0.1, \text{units1} = 21, \text{units2} = 43$. When they were oversampled with SMOTE the best values were $\text{rate1} = 0.2, \text{rate2} = 0.1, \text{units1} = 43, \text{and} \text{units2} = 21$.

The performance metrics of the final tuned model using predictions on the test data are given in Table 3.6 for the model with unbalanced data and after applying SMOTE. The overall accuracy and balance accuracy do not agree regardless of whether SMOTE was applied or not. This indicates that the accuracy does not reflect the performance of both classes well. After applying SMOTE the accuracy increased to 0.843 from 0.722 while the balanced accuracy decreased to 0.548 from 0.650. Moreover, the specificity decreased after applying SMOTE while the sensitivity increased. These results indicate that there is also a trade-off that occurs when SMOTE is applied and neural networks are fit but in a different way. In comparison to the accuracy, balanced accuracy, sensitivity and other metrics produced by random forests + SMOTE, the neural networks do not perform better. The one area where they do perform better is in regards to the specificity when the unbalanced data are trained. These results may indicate that the algorithms used to fit neural networks can overcome the issue of imbalance and overlap to some extent without the need for additional oversampling.

Model	Acc	Bal. Acc	Sens	Spec	F1 Score	AUC	Kappa
Neural Networks + No SMOTE	0.722	0.650	0.753	0.546	0.821	0.696	0.217
Neural Networks + SMOTE	0.843	0.548	0.972	0.125	0.913	0.696	0.137

Table 3.6: Performance metrics for test set predictions obtained using neural networks. Networks were trained using unbalanced data and data oversampled with SMOTE. Only the F1 Score is improved by oversampling. Surprisingly, the specificity decreased after applying SMOTE which oversamples the negative class. The increase in sensitivity and corresponding drop in specificity indicates that there is a trade-off in class accuracies after oversampling. The underlying issue is the overlap in the feature space.

Figure 3.16 provides the validation and training data performance metrics as training was performed over each epoch. Each iteration of training is conducted using a bootstrap sample of the training data, performance is evaluated on the validation data and the parameters of the model are adjusted. Another bootstrap sample is then taken from the training data and the process continues until the number of epochs has been reached or some other stopping criteria that was specified has been met.

The training accuracy of the unbalanced data increased while the validation accuracy decreased, evidencing that the model is overfit to the training data. When SMOTE was applied the training and validation accuracy agreed. The validation set AUC was mostly consistent between unbalanced and SMOTE oversampled data. However the training AUC was further from the validation AUC when SMOTE was applied indicating more severe overfitting. The sensitivity on the validation data consistently tracked that on the training data when SMOTE was applied. However, the validation set sensitivity for the unbalanced data decreased while that for the training set increased. For both unbalanced and SMOTE oversampled data the validation set specificity decreased while that for the training set increased. This disagreement was more severe when SMOTE was applied. These results indicate that the application of SMOTE bettered the training process only in regards to the specificity, and thereby the accuracy. However, the training process of the AUC and specificity was negatively impacted.

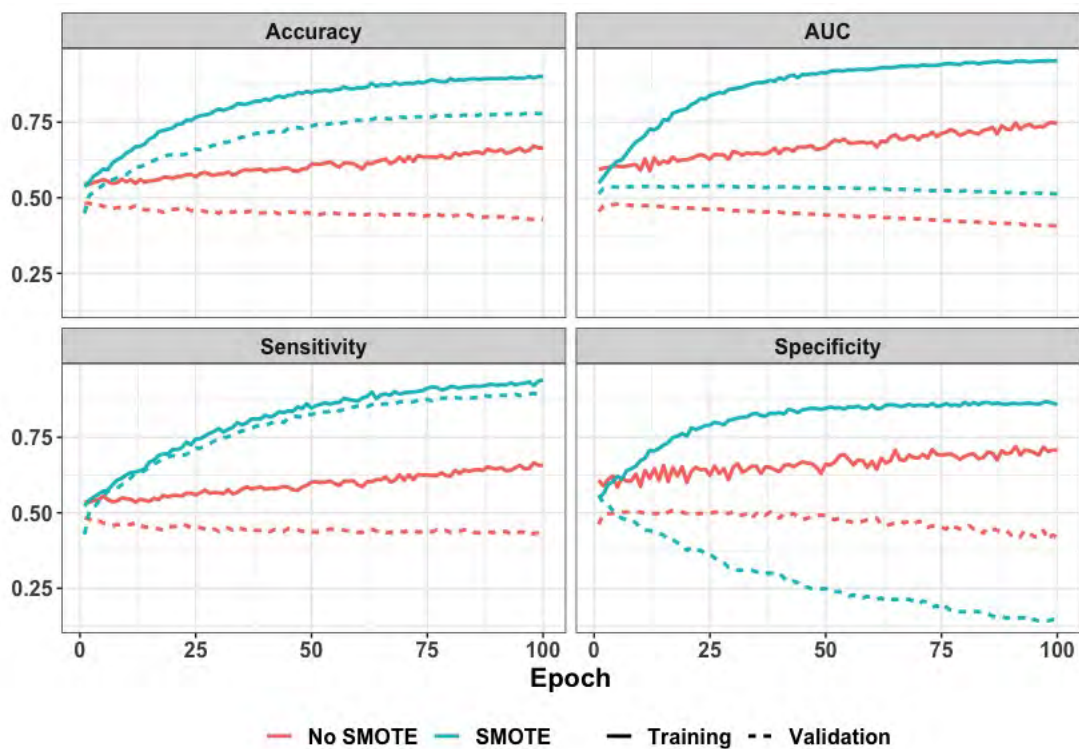


Figure 3.16: Tracking performance as neural networks are trained. Hyperparameters were first selected using holdout validation. Then those which maximized the AUC were selected and the final models were trained. The performance by epoch is given for each metric. Models were trained for 100 epochs. The model trained on SMOTE oversampled data exhibited more signs of overfitting than that trained on unbalanced data. This can be seen by comparing the differences between training set performance and validation set performance.

3.5 The Need to Further Study and Improve SMOTE

The results that we have obtained in this chapter indicate that in order to tackle the predictive component of the problem we must first deal with the issue of imbalance and overlap. We have applied one of the most popular oversampling methods to the data and fit more complex models. However, we still find that there is a trade-off in the class-specific accuracies which inhibits us from obtaining good predictions on the minority class. Moreover, we see that both overlap and imbalance must be tackled in combination in order to truly solve the predictive component of the problem. We now proceed to discuss the shortcomings of SMOTE and our simulation-based investigation into them. We also discuss the novel algorithm that we created using the results of those simulations.

Chapter 4: The Strategic SMOTE (S-SMOTE) Algorithm: A Simulation-Based Approach

Though SMOTE was shown to be an improvement over ROS, the authors of the technique note that when minority examples are crowded by majority class examples, the resulting decision rules will likely be biased towards the minority class. This creation of ‘noisy examples’, that is, examples in overlapping areas, is a known shortcoming of SMOTE. This effectively creates a trade-off between majority class and minority class accuracy. Additionally, since SMOTE does not consider whether a minority example is crowded by other minority examples, already-dense clusters of the minority class may see a larger increase in density than sparse clusters of the minority class. The presence of such within-class imbalance, known as the ‘small disjuncts’ problem, can cause poor classifier performance (e.g. Prati et al. 2004; Denil and Trappenberg 2010; Stefanowski 2016). In Figure 3.4, a small disjunct would be created if we blindly oversampled the minority class cluster in the left circle. Similarly, the amount of overlap in that toy data set could increase if we oversampling using points in the top-left quadrant of the circle on the right side.

Improvements upon the original SMOTE usually involve rectifying these two issues - where to generate synthetic examples, so as to not create more overlap, and how much to oversample, so as to not further inflate already dense spaces, or ignore sparse areas. Though most adaptations of SMOTE usually apply a cluster-based approach, there are some that do not use clustering to inform the oversampling process.

The most popular adaptation to SMOTE that does not rely on clustering is Borderline-SMOTE1 and Borderline-SMOTE2 (Han et al. 2005). The former algorithm only oversam-

ples examples from the minority class that are on the borderline of the areas of the feature space belonging to each class. An example is called a *borderline example* if more than half of its k -nearest neighbors belong to the majority class. Borderline-SMOTE1 only uses nearest neighbors from the minority class while Borderline-SMOTE2 additionally uses neighbors from the majority class. Though improvements were shown for the minority class, the authors did not discuss how their algorithm impacts classifications for the majority class and the issue of small-disjuncts was not addressed. The Adaptive Synthetic Sampling Approach (ADASYN) introduced by He, Bai, et al. (2008) generates more synthetic observations for harder to learn examples, where an example is considered harder to learn when many of its k nearest neighbors belong to the majority class. Though this deals with the small disjuncts issue, this approach is likely to increase the presence of noisy examples (Barua et al. 2012).

As a cluster-based adaptation of SMOTE, k -means SMOTE (Douzas et al. 2018), focuses on removing between- and within-class imbalance by only applying SMOTE within clusters that are dominated by the minority class and by generating more synthetic examples in sparse minority class dominated clusters. This algorithm addresses the issues of overlap, imbalance, and small-disjuncts, however, the authors note that its effectiveness depends on finding an appropriate number of clusters to use, if any can be found. Cluster-SMOTE (Cieslak et al. 2006) uses k -means to cluster the minority class and then performs SMOTE in a cluster-based manner. However, the issues of overlap and small-disjuncts are not rectified by this because the minority class dominance of the cluster is not factored into the oversampling process. Cluster-based oversampling was also introduced by Jo and Japkowicz (2004), which aims to balance the number of examples within majority class clusters and minority class clusters, while balancing the classes overall. Since this method only oversamples, it is possible that it could create overly specific decision regions, as does ROS. Other solutions include those introduced by Bunkhumpornpat et al. (2009), Sáez et al. (2015), Zhu et al. (2017), and Rivera (2017).

A few works have also been recently introduced which focus on how to generate synthetic examples along the line segments which SMOTE creates between a point and its nearest neighbor of the minority class. In a novel and comprehensive analysis of SMOTE, Elreedy and Atiya (2019) showed that, in general, the distribution of synthetic data generated by SMOTE will be more ‘contracted’ than the distribution of the original minority examples, when points are generated along these line segments according to a $\text{Uniform}(0, w^*)$ distribution. They showed that as w^* increases, this compactness decreases - that is, as these line segments are allowed to intersect the neighbors of interest, and extend beyond them, the contraction becomes less severe. However, as this line is extended, it may encroach into an area of overlap or majority class dominance. Their work does not directly address solutions to the issues of overlap, imbalance, or small disjuncts.

Some recent extensions of SMOTE use different probability distributions to select points along the line segments joining an example and its neighbors (e.g. Lee et al. 2017; Kamalov and Denisov 2020; Tarawneh et al. 2020; Bernardo and Della Valle 2021). The use of the Gamma distribution is discussed by Kamalov and Denisov (2020) who aimed to place synthetic examples in proximity to, and in the direction of, nearby minority points. However, they did not discuss how their method impacts overlap or the small disjuncts issue. The use of both a Uniform and Gaussian distribution, in subsequent steps, is discussed by Lee et al. (2017), however, they also did not discuss the issues of overlap or small disjuncts.

To the best of our understanding there has not been a thorough simulation study conducted to understand how the distribution of w^* , the interpolation and extrapolation factor in SMOTE, impacts the predictive performance of models fit to data oversampled with SMOTE. This factor is used to determine where points fall along the line segments joining a minority class example and its nearest neighbor. The location of synthetic points in the feature space determines whether SMOTE will introduce further overlap and small disjuncts.

We have identified a need for a simulation study that not only studies this element of the

SMOTE algorithm but also does so with consideration to overlap, imbalance, and other data characteristics (e.g. missing data). Such a study can be used to characterize the performance of SMOTE in a variety of situations. The results can also be used to create a novel SMOTE-inspired algorithm that more informedly generates synthetic data in scenarios where multiple data difficulties are present.

Most of the research conducted regarding the performance of SMOTE and possible improvements was not conducted under the premise that negative and positive predictions are equally important. Therefore, there is also a need for a simulation study that sheds insight on the performance of SMOTE in this scenario. In this chapter, we will discuss the results of a simulation study that tackles each of these needs. We will also present our novel SMOTE algorithm called Strategic SMOTE (S-SMOTE) whose creation was informed by the results of our simulation study. Lastly, we provide an example application of S-SMOTE.

The technical problem of binary classification with an imbalanced response variable and overlap in the feature space is a difficult one to overcome. We do not aim to provide S-SMOTE as an outright solution to the problem. Rather, we present an algorithm that incorporates the useful insights and novel findings that we have discovered so far as we forge towards a solution. Our goal is to use simulations to (1) characterize the challenges that this problem creates for predictive modeling in a variety of data scenarios, (2) determine whether SMOTE overcomes or fails to overcome these challenges, and (3) find scenarios in which S-SMOTE also overcomes or better overcomes these challenges, if any.

4.1 Preliminary Details of S-SMOTE

In order to apply S-SMOTE we first require a training dataset. This dataset can have any number of rows and columns but the size of the minority class in the dataset will directly impact the computation time of the algorithm. In addition to the unbalanced dataset, the

response variable and number of synthetic examples desired must be specified. Though the user could also specify a distribution to use for interpolation/extrapolation, the results of our simulation study in the next section will be used to select this more informedly. In order to explain the details of the technique and provide examples we use a Uniform(0,1) distribution for now which is the same as SMOTE. The parameters of S-SMOTE include the maximum number of nearest minority neighbors k_{max} that can be used for oversampling a given minority example and the dominance $\eta = (\eta_1, \dots, \eta_l)$ that each minority class example is tested for to approve its use for oversampling. η is a decreasing vector. Gower's distance (Gower 1971) is used in order to incorporate differences in categorical variables when determining the nearest neighbors of a point.

The oversampling technique begins by using $p_{min} = 0\%$ of minority examples for oversampling and $\eta_{now} = \eta_1$. While $p_{min} < 75\%$ and we have not used all values in η , for each minority example and for $k = 1, \dots, k_{max}$, of all points that are as close as the k -th minority neighbor, the proportion that belong to the minority class is calculated. We denote this vector of proportions as prop_{min_i} which is the dominance at each minority neighbor of minority point i , where $i = 1, \dots, n_{min}$ and n_{min} is the number of minority examples. For each point i , we then determine how many neighbors we can use before this threshold first drops below η_{now} and use that many neighbors for oversampling.

If less than 75% of minority points had at least one neighborhood meeting η_{now} , we decrease the dominance to the next largest value for η and find points which have neighborhoods meeting that dominance. This process continues as we successively find the most strongly dominated minority neighborhoods of the feature space until we have enough minority points, at least 75%, for oversampling. If no minority points were deemed fit for oversampling after exhausting all thresholds, then no points were deemed fit for oversampling. In this case the dominance threshold may be too strict for the amount of overlap in the feature space and it could be lowered.

We then calculate the relative dominance of the neighborhoods of the minority class points deemed fit for oversampling. We denote this vector as rprop_{min} for the i -th point. By calculating the relative dominance we are able to determine which minority class neighborhoods are least dominated relative to those that met at least η_l dominance. The algorithm returns the percent of minority examples used for oversampling. If too few examples are used relative to how much oversampling is desired then too little information is being used to generate synthetic examples which could negatively impact the predictability of the resulting models. Additionally, the five number summaries for the number of neighbors used and the relative final dominance of each point are returned. Note that if $\max_i\{\text{prop}_{min_i}\} = 1$ then the relative dominance is the same as the original dominance for all points.

Figure 4.1 gives a plot of the relative dominance and number of neighbors deemed fit for oversampling for a simulated dataset. We applied S-SMOTE to a simulated dataset of 1000 observations with 60 variables, 27 of which were categorical, an overlap amount of 0.70, 85% majority examples, and no missing data. We provide more details on how we simulate data with these characteristics in the next section. In order to see the points clearly we jittered them. We used $k_{max} = 15$ and the dominance thresholds checked were the decreasing sequence from 0.6 to 0.20 by -0.02. 91.3% of minority examples were deemed fit for oversampling (137/150), the minimum number of neighbors used was 1 and the maximum was 10, and the final relative dominance thresholds ranged from 0.583 to 1.

In Figure 4.1 the red values in parentheses indicate the preference with which points should be oversampled. The horizontal and vertical lines indicate the median number of neighbors used and relative dominance threshold, respectively. Quadrants are defined using values less than or equal to the median number of neighbors and greater than or equal to the median relative dominance. Points in quadrant 3 have the least dominance and density, therefore they are used for oversampling the most, after which points in quadrant 2 are used for oversampling.

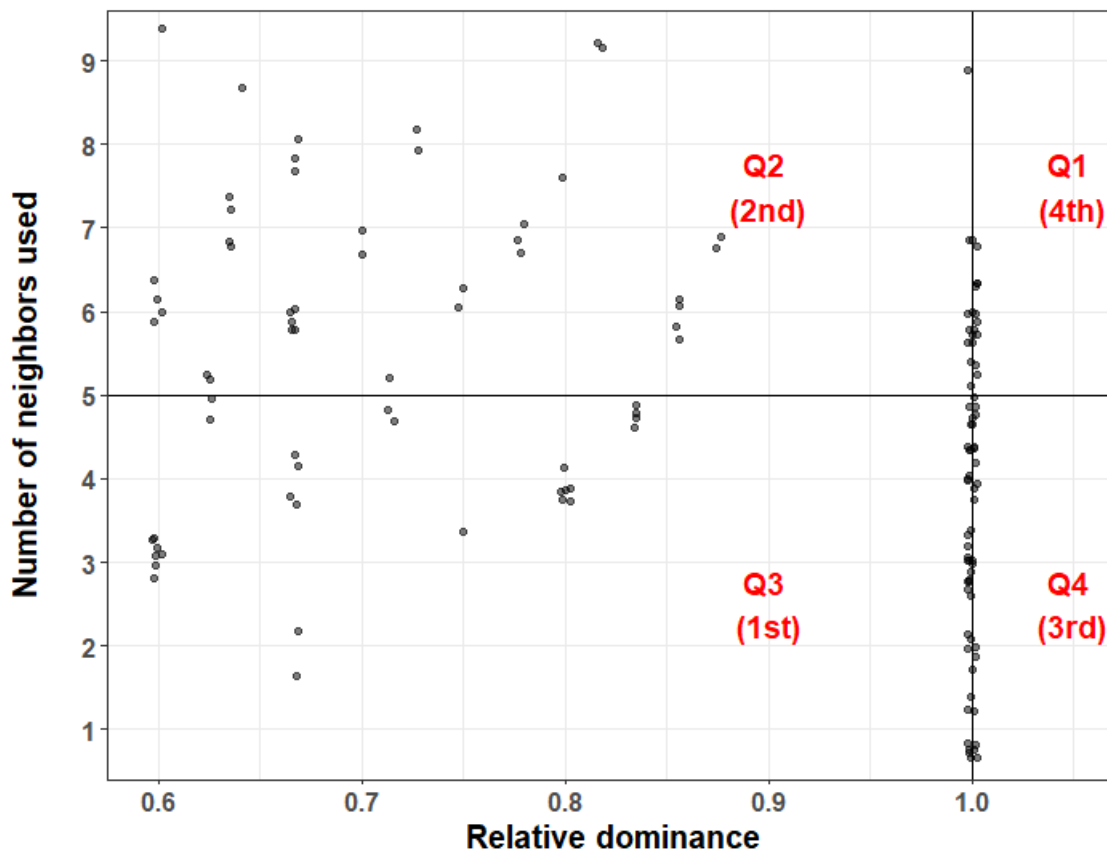


Figure 4.1: Example jittered plot of relative dominance and number of minority neighbors used for oversampling. This is plotted for each point in the minority class. Red numbers indicate priority of points when oversampling. Points in region 1 have the least dominance and density, therefore they are used for oversampling the most. Horizontal and vertical lines are the medians of the respective axes. Quadrants are defined using values less than or equal to the median number of neighbors and greater than or equal to the median relative dominance.

There is clear variability in the number of neighbors used for oversampling and the relative dominance of each point. This indicates that not all points deemed fit for oversampling have equal need for oversampling. This is where our algorithm tackles an issue that the original SMOTE algorithm does not. In order to oversample with respect to this information we select a vector of decreasing weights $\rho = (\rho_1, \rho_2, \rho_3, \rho_4)$ (e.g. $(0.1, 0.3, 0.5, 0.1)$) which we use for oversampling. These values correspond to the probability with which we select a point in quadrants 1 through 4 for oversampling, respectively. Using the example vector of weights, when selecting a minority point to oversample we will select from quadrant 1 with probability

0.1, for quadrant 2 with probability 0.3 and so on. These weights are also hyperparameters of S-SMOTE.

In order to perform the actual oversampling we randomly select a minority neighbor deemed fit for oversampling and one of its neighbors within the neighborhood that met the strongest dominance criteria possible. The data values of the synthetic point generated can be defined as $z = x_0 + w\Delta$, where $w \sim F$, $\Delta = x_i - x_0$, x_i is the feature vector for the nearest neighbor used, and x_0 is the feature vector for the minority example of interest. Similar to SMOTE-NC, when there are categorical variables the level of each categorical variable that was most commonly observed amongst its neighbors is given to the synthetic point. Similar to SMOTE, this method adds back in a random proportion of the difference between the point of interest and one of its nearest neighbors, and labels this as a new point in the minority class. The key differences between S-SMOTE and SMOTE/SMOTE-NC are that (1) we determine which points to oversample and which of their neighbors to use for oversampling by successively checking for the strongest possible neighborhood dominance, (2) we use the results of our simulation to guide the selection of the distribution for w which determines how interpolation and extrapolation are performed, and (3) we use Gower's distance to determine how similar points are which incorporates differences in categorical variables.

Recall that SMOTE-NC uses the median of the standard deviations of the quantitative variables to adjust the Euclidean distance (calculated using only the quantitative variables) when the levels of a categorical variable are different. Similar to SMOTE, missing data are perpetuated throughout the dataset. If a neighbor has missing values for a given variable, then the synthetic point generated with it will also have missing values for that variable. Additionally if all neighbors of a reference point are missing data for a given categorical variable, the synthetic point will have missing values for that categorical variable, otherwise the majority vote after dropping NAs is used. In order to avoid this, missing data should be handled separately before oversampling. Our simulations include the evaluation of the

predictive performance of our algorithm in the presence of missing data.

4.2 Outline of Simulation Study

We will now provide the details of our simulation study. Understanding these details is necessary to understand the results of the simulations and the performance of these algorithms. The purpose of the simulation study is to evaluate the performance of S-SMOTE and SMOTE under a variety of data difficulties as w changes. The results will allow us to characterize the problem of binary classification with overlap in the feature space in the presence of other data difficulties. These results will also bring us closer to the final version of S-SMOTE as we will use them to select a distribution for w .

In order to evaluate the performance of these algorithms under a variety of scenarios we need to generate simulated data with varying characteristics. After setting the sample size (`sampsize`) and number of variables (`ndim`), we set the proportion of majority class observations (`propmaj`, ranging from 0 to 1), and the proportion of categorical variables (`perccat`, ranging from 0 to 1). Each quantitative column in the simulated dataset contains random deviates sampled from a statistical distribution (e.g. Normal, Poisson, Gamma). The distribution is the same for the majority and minority class, therefore the dataset contains overlap at the onset. We refer to the value used to adapt overlap as `ovlap` and it ranges between 0 to 1.

We remove overlap from the dataset by shifting the majority class distribution of each quantitative variable away from that of the minority class by an amount determined by `ovlap`. Each categorical variable in the simulated dataset is created by randomly sampling from a vector of a certain size containing factor levels. The probabilities with which elements in this vector are selected is determined by randomly sampling a vector of weights from a Dirichlet distribution. In order to remove overlap in the categorical variables we find the

largest frequency for the minority class and share the majority class frequency of this level amongst the other levels. This effectively causes the most common level of the minority class to appear the least frequently in the majority class.

In order to introduce missing data we use two schemes (`typemiss`). Data can be missing completely at random (`MCAR`), missing at random because the observation is in the minority class (`MAR`), or not missing at all (`none`). The percent of missingness (`amountmiss`) is a value ranging between 0 and 1 giving the proportion of cells in the entire dataset, or only the minority subset if `typemiss = "MAR"`, that are set to `NA`.

The values that we considered for each of these characteristic is given below:

- `ovlap`: 0.2 and 0.8
- `propmaj`: 0.55, 0.85, and 0.95
- `sampsize`: 750 and 1750
- `ndim`: 20, 100, and 150
- `perc_cat`: 0, 0.25, and 0.85
- `typemiss`: ("MCAR"), ("MAR"), and ("none")
- `amountmiss`: 0, 0.3, and 0.7

Given that we desire to investigate many data characteristics, we held some characteristics constant while studying others. Each group of simulated datasets holds four combinations of `ovlap` and `propmaj` constant while the other data characteristics change. For example, the first two sets of simulations that we performed used training data with the characteristics given below.

```
ovlap propmaj sampsize ndim perc_cat typemiss amountmiss
```


0.2	0.55	750	100	0	none	0
0.8	0.55	750	100	0	none	0
0.2	0.85	750	100	0	none	0
0.8	0.85	750	100	0	none	0

ovlap	propmaj	sampsize	ndim	perc_cat	typemiss	amountmiss
0.2	0.55	1750	100	0	none	0
0.8	0.55	1750	100	0	none	0
0.2	0.85	1750	100	0	none	0
0.8	0.85	1750	100	0	none	0

We studied a total of 10 set values for the data difficulties in combination with imbalance and overlap. This gave a total of 40 individual scenarios that we studied. For each scenario we simulated both training and testing datasets. We then applied different oversampling techniques to compare the predictive performance of models trained on oversampled data with those trained on the unbalanced dataset.

There were four options that we considered when balancing datasets:

- *Population oversampling (Pop)*: Balancing the dataset by sampling minority class points from the population distributions that the data were originally simulated from.
- *Random oversampling (ROS)*: Balancing the dataset by taking bootstrap samples from the minority class and replicating each selected rows in the dataset.
- *Synthetic Minority Oversampling TEchnique (SMOTE)*: Using interpolation to randomly generate synthetic minority class examples as explained earlier. The interpolation is performed by randomly adding back in some percentage w of the difference between the point of interest and the selected neighbor.

- *Strategic SMOTE (S-SMOTE)*: Using interpolation and extrapolation to randomly generate synthetic minority class examples as explained earlier. The most minority class dominated areas are first oversampled and then the dominance is decreased incrementally until enough examples have been deemed fit for use. The interpolation/extrapolation is performed by randomly adding back in some percentage w of the difference between the point of interest and the selected neighbor.
- *No oversampling (None)*: In this case we do not oversample the dataset at all and it remains unbalanced.

For both SMOTE and S-SMOTE, w was randomly generated from the distributions below:

Four-Parameter Beta Distribution	Gamma Distribution	Uniform Distribution
FPB(0.5, 0.5, -1, 1)	$G(shape = 0.5, scale = 1.0)$	$U(-3, 4)$
FPB(1, 3, -3, 4)	$G(shape = 1, scale = 2)$	$U(0, 1)$
FPB(1, 3, 0, 1)		
FPB(2, 2, 0, 1)		
FPB(2, 5, 0, 1)		
FPB(5, 1, -2, 3)		

There were, therefore, a total of 23 different oversampling methods applied to each of the 40 types of simulated datasets. This gives a total of 920 individual simulation scenarios. For each of these 920 combinations of oversampling method and dataset type, we performed 500 simulations and fit models to the resulting training datasets. The models considered were LASSO logistic regression, random forests and neural networks. Neural networks were computationally expensive to fit in a loop. Due to computation times we decreased the simulation size to 200 when training some neural networks. This entire process was repeated for every model under consideration. In order to keep results consistent the same datasets were used for all 23 oversampling methods. That is, for a given set of data characteristics, we applied all 23 oversampling methods to each of the same 500 datasets and fit models to

these same datasets. We did this in order to evaluate changes in performance due only to the changes in oversampling method and model used, not the underlying dataset that was oversampled itself.

The regularization parameter of LASSO logistic regression models was the value in $2^x, x \in (-15, -14, \dots, 14, 15)$ that minimized the 10-fold cross-validation AUC. The variable selection properties of LASSO logistic regression could come at the cost of predictive capability. We include this model amongst random forests and neural networks in order to further investigate this. Random forest models were fit using $\min(\text{samsize}/4, 250)$ trees, \sqrt{p} variables tried, and a minimum node size of $\text{samsize} * 0.05$. Neural networks had two hidden layers with number of nodes equal to 45% and 20% of the number of columns in the training data, respectively. The *ReLU* activation function was applied to each hidden layer and the Sigmoid activation function was applied to the output layer, giving probabilities between 0 and 1. Binary Cross-Entropy was used for the loss function and the Adadelta optimizer was used. Networks were trained for 100 epochs on batches of size 32. A threshold of 0.5 was used for all class predictions for all models.

We would like to make a note about computational issues that we faced. We did not tune random forest and neural network models as we did in the case of LASSO logistic regression due to their lengthier training times. We fit a total of 460,000 models since there were 23 oversampling methods applied to 40 individual data scenarios with 500 repetitions. When available, the defaults suggested by the makers of `randomForest` were used.

However such defaults do not exist for neural networks since they have many hyperparameters taking many values. Incompatibilities between the R interface to `keras` and `caret` prevented us from performing a grid search for the best hyperparameters. Additionally, the tuning functions provided by `tfruns`, a package with training tools for `tensorflow` requires the use of a different script for each tuning iteration which is not feasible given how many models we desire to fit. We also attempted to use the R interface of `kerastuneR` but we had

unresolvable issues installing its Python module with the R interface and the Python installation was not recognized. The time taken to complete our simulations for neural networks (without tuning) was approximately 1.5 months using the R interface to `keras`. A

Future work will involve performing these simulations in a faster programming language (e.g. Python) so that all models can be trained well. We will still include these models in our discussion about the results of our simulations. However, the challenges that we faced should be taken into account as we present our simulation results and make comparisons between models.

In order to evaluate the predictive performance of models we calculated a variety of values from the test set predictions. Let the number of true positives = TP, true negatives = TN, false positives = FP, false negatives = FN, observed positives = P, and observed negatives = N. For every model that we fit, we obtained predictions for the test set and calculated the values below.

- Accuracy (Acc) = $\frac{TP+TN}{TP+TN+FP+FN}$
- Sensitivity (Sens) = True positive rate (TPR) = $\frac{TP}{TP+FN}$
- Specificity (Spec) = True negative rate (TNR) = $\frac{TN}{TN+FP}$
- Balanced accuracy (Bal Acc) = $\frac{TPR+TNR}{2}$
- Negative predictive value (NPV) = $\frac{TN}{TN+FN}$
- Positive predictive value (PPV) = $\frac{TP}{TP+FP}$
- F1 score = $\frac{2TP}{2TP+FP+FN}$, the harmonic mean of Sens and PPV
- Area under the receiver operating characteristic (ROC) curve (AUC): Area under the plotted curve of TPR and FPR (or 1 - Specificity) for various probability thresholds.

- Kappa coefficient: $(\kappa) = \frac{2 \times (TP \times TN - FN \times FP)}{(TP + FP) \times (FP + TN) + (TP + FN) \times (FN + TN)}$. This gives a measure of the agreement between the truth and predictions. A value of 1 indicates total agreement between truth and predictions, while a value at or near 0 indicates no agreement other than that due to chance. A negative value could indicate no relationship or non-random differences between the truth and predictions.

One effect of the many computational issues that we had with neural networks is that the balanced accuracy, sensitivity, specificity, PPV, and NPV produced by them could be under- or over-estimated. We observed many values near 55/45 and 85/15 for the sensitivity/specificity and many values near 50 for the balanced accuracy, even in the best case data scenarios. As we discuss the results of our simulation study we will point out these issues.

In order to study how these oversampling methods impact the within-class imbalance for the minority class, and how this further impacts the predictive performance of models, we performed k -means clustering on the minority class, after oversampling the training set, and calculated the values below. We first attempted all cluster sizes from two to the maximum amount that would allow for at least 20 points in each cluster. The average silhouette widths (taken over all clusters) were calculated and we selected the number of clusters that maximized this. This cluster allocation was then used to calculate the minimum, Q1, median, Q3, mean, maximum, and IQR of the resulting cluster sizes and the number of clusters.

In order to understand how other characteristics of the training data are related to the performance of models that they are trained on, we also calculated values indicative of the minority class distribution after oversampling using the methods that we outlined earlier. The first value was the relative difference in the Frobenius norms of the correlation matrix of the quantitative variables in the minority portion of the training data, between the unbalanced and balanced datasets. This gives us an idea of how the amount of variability in the feature space changes between the unbalanced dataset and the oversampled dataset, for the minority class. The second value is the proportion of points in the minority or majority

class that have more than two out of six nearest neighbors from the opposite class. These are called the minority class and majority class overlap metrics, respectively. We aim to understand how these training data characteristics are related to performance metrics as we study the results.

4.3 Results of SMOTE Simulation Study

Our first step towards understanding the simulation results is to evaluate the median predictive performance of LASSO logistic regression, neural networks, and random forests applied to data which have been oversampled using each of the oversampling methods discussed earlier (including no oversampling). For every oversampling method, type of dataset (type as determined by amount of imbalance, overlap, missingness, and other characteristics), and model, we have 500 values for each performance metric. We calculated the median of these 500 values, giving us one median for each oversampling method, type of dataset, model, and performance metric. We then calculated the median and standard deviation (SD) of these medians over all oversampling methods, giving us one overall median and SD of these for each type of dataset, model, and performance metric.

Figure 4.2 provides the numeric value for this overall median for the AUC and balanced accuracy, as well as the standard deviation in parentheses. Plot for other metrics can be found in the Appendices. These are plotted using a gradient of yellow (0) to green (1) for the AUC, Balanced Accuracy, F1 Score, Kappa Coefficient, NPV, PPV, TPR, and TNR. Each point on the y-axis gives the change that we made to the Base Case. The Base Case is datasets with $N = 750$, 100 variable, and no categorical or missing data. In the case of 85% categorical variables and 70% data MAR, only 5% of simulated data belonged to the minority class, not 15%. We looked at more severe imbalance in these cases than in the others but we use the same x-axis label for organizational purposes.



Figure 4.2: Median AUC and balanced accuracy, taken over the median performance for each oversampling method. X-axis provides overlap and imbalance amounts. Y-axis indicates which data difficulty was present. The Base Case is $N = 750$, 100 variables, and no categorical or missing data. The median was calculated out of 23 medians which were themselves calculated out of 500 values.

It is evident that there is a drop in performance whenever overlap increases. For most metrics, data cases, and models, the color goes back and forth from dark green to yellow-green as we scan across the x-axis. However in the case of neural networks this pattern does not occur for AUC, balanced accuracy, NPV, PPV, and Kappa. Neural networks had higher sensitivity when the percent minority was 15% but they have lower sensitivity when it was 45%. Random Forests had high sensitivity when the percent minority was 15% and continued to have high sensitivity when it was 45%. Although it decreased when the overlap was 0.8, it remained high in the harder cases when data were MCAR and MAR.

The sensitivity of neural networks increased when the percent in the minority class decreased but the specificity decreased when the percent in the minority class decreased. Random forests performed very poorly in terms of AUC when 70% of minority data were missing at random and the overlap was 0.8 and only 5% of the data belonged to the minority class. Logistic regression had the best overall performance in terms of specificity followed by random forests. However, in most cases all models struggled to achieve high sensitivity. Especially when the overlap was 0.8 and 15% of rows belonged to the minority class. In this case it was always lower than 0.50 and random forests struggled the most. The specificity was especially poor for random forests when 70% of the minority class data were MAR and only 5% of all rows belonged to the minority class

Evidently the presence of categorical and missing data has a huge impact on the predictive performance of these models. There is evidence that overlap may have an even greater negative impact in these cases than imbalance. When both are present there is an even sharper decrease in performance. Our results also provide evidence that after specificity overlap and imbalance impact the Kappa coefficient and then the NPV the most negatively.

To gain further understanding about the performance of these models when there are categorical or missing data we visualize the individual median performance for each oversampling method rather than the overall median. We plot these in Figures 4.3 and 4.4, respectively.

We used a darker gray for the median corresponding to unbalanced data. Then we also plotted the medians of these in purple. Sometimes this overlaps with the darker gray point. These plots are faceted by metric and model combination. Each space on the x-axis notes the overlap-imbalance combination used to simulate that dataset. These plots give us a finer view of the results discussed in the previous set of plots because we can now see the medians relative to each oversampling method, and unbalanced data, as well as the overall median. Additionally, we can see how variable the medians actually are across oversampling methods. Similar plots for all performance metrics and data scenarios are available in the Appendices.

We assessed all of the plots in the Appendices including those here and found that there were many non-trivial differences in the performance. We saw more variability in the medians when it came to the NPV, especially when 15% of the data belonged to the minority class. When $N=20$ though the variability was low as in the general case. The PPV was not quite variable in comparison to other metrics but was most variable with respect to itself when neural networks were fit and 85% of data were categorical with 5% of rows belonging to the minority class. This was also true when data were MCAR or MAR.

The Kappa coefficient was also noticeably more variable than other metrics. Especially when there was also imbalance. The balanced accuracy exhibited more variability when $\text{overlap}=0.2$ and 15% of data belonged to the minority class. When 30% of minority data were MAR the AUC from random forests were quite variable when there was also imbalance. Specificity was also quite variable in a manner similar to the NPV. In other cases, the medians were fairly similar and there was little variability due to the oversampling method.

These results indicate that when there is categorical or missing data *as well as imbalance and overlap* the decision about which oversampling method and distribution for w to use is more important. This is evidenced by the increased variability in performance metrics when we enter these scenarios than when we look at the base case for example.

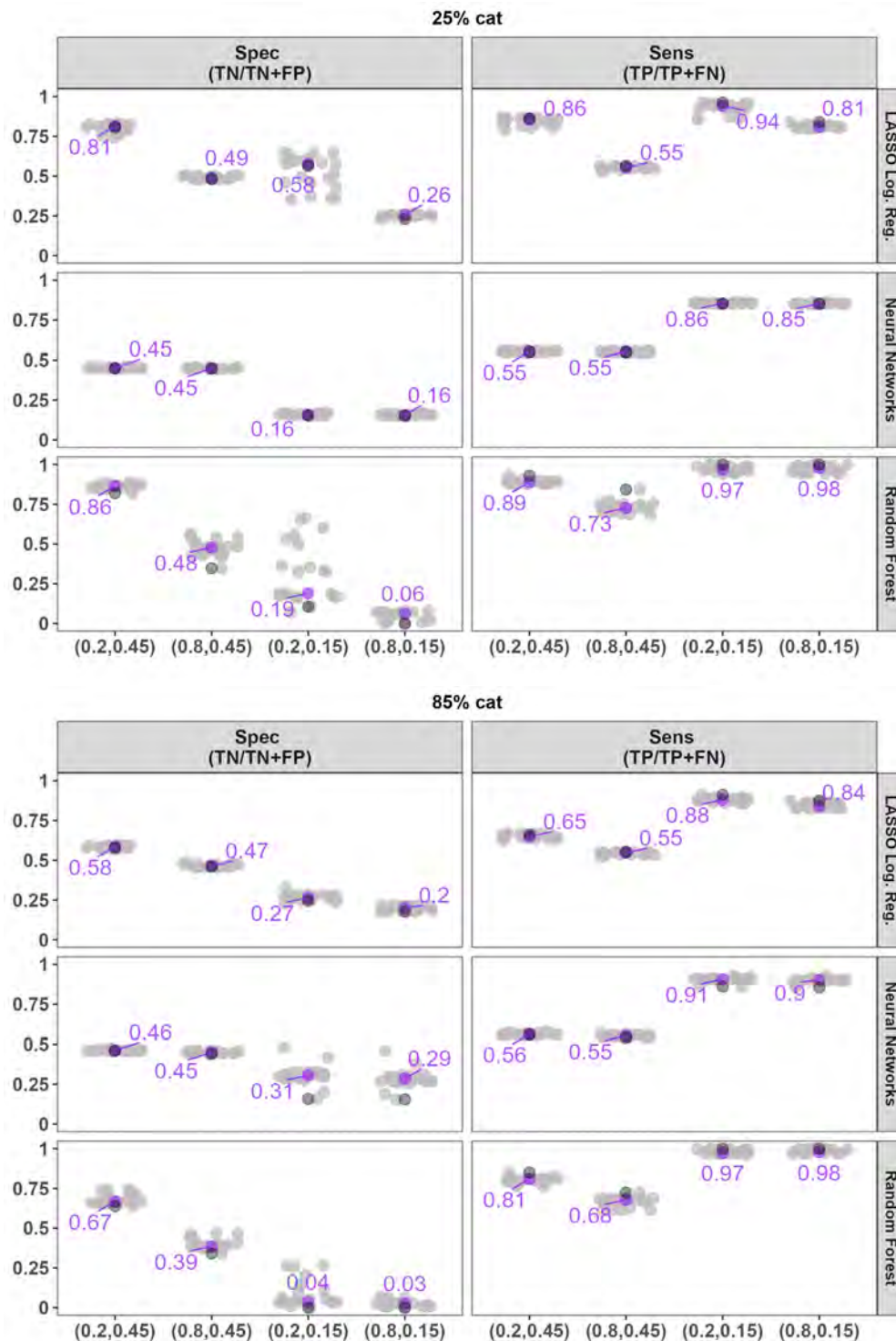


Figure 4.3: Median specificity and sensitivity given by light gray point. Median was calculated over 500 repetitions for a given sampling method. Each training set had 750 rows and 100 variables. Of those 25% or 85% of variables were categorical and there was no missing data. The purple point gives the median for each group of points over all sampling methods. Dark gray point mark the results for unbalanced data.

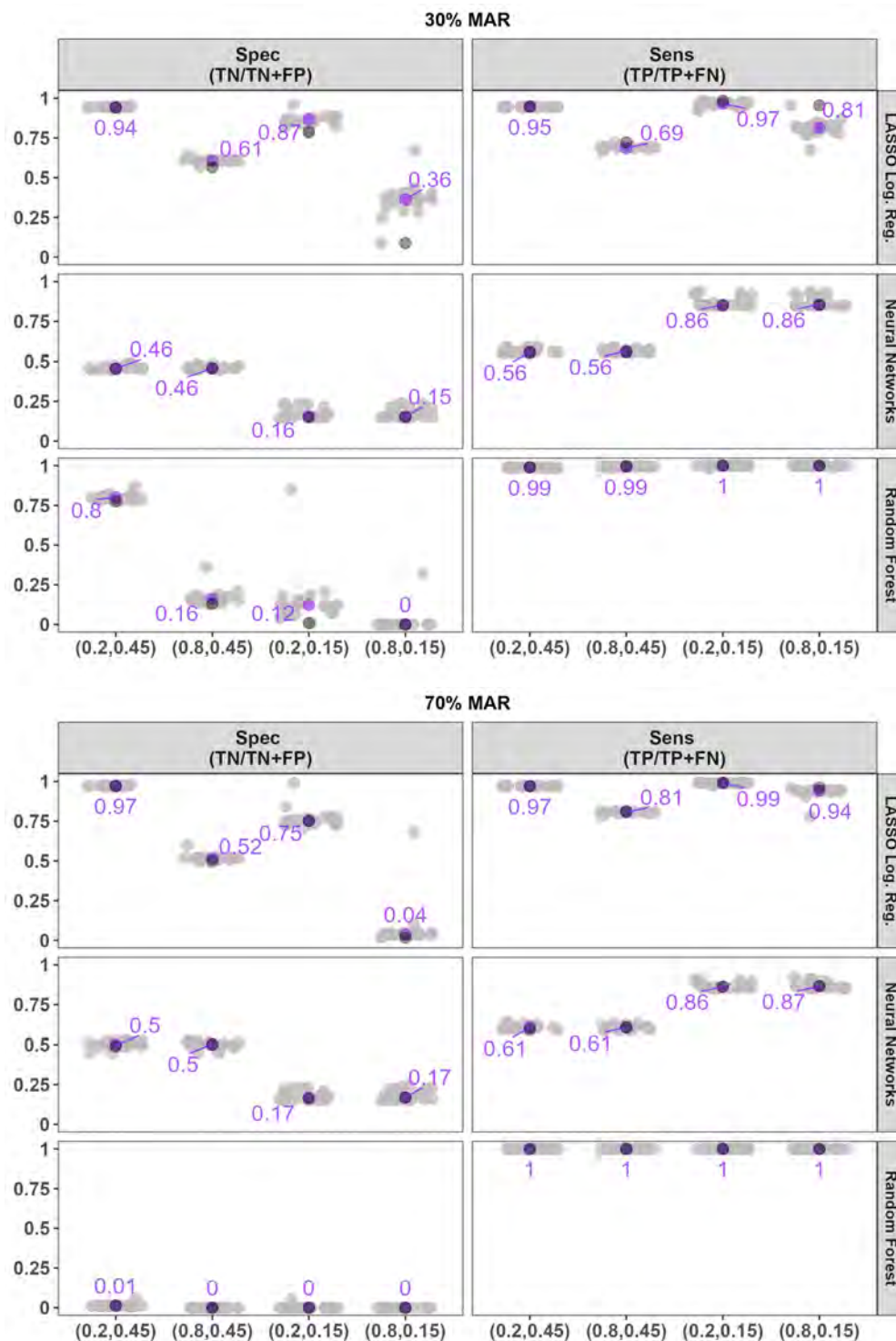


Figure 4.4: Median specificity and sensitivity given by light gray point. Median was calculated over 500 repetitions for a given sampling method. Each training set had 750 rows and 100 quantitative variables. 30% or 70% of cells in the dataset belonging to the minority class were missing at random. The purple point gives the median for each group of points over all sampling methods. Dark gray point mark the results for unbalanced data.

We desire to know whether the variability that we observed in the median performance metrics is simply due to the oversampling method regardless of the distribution used for w . If so, we might expect to see differences simply due to the oversampling method. To that effect, we plotted the distributions of each performance metric for each oversampling method. We aggregated over the distribution of w for SMOTE and S-SMOTE and we aggregated over the amount of imbalance and overlap, the data scenario, and the model used for all methods. The median and standard deviation of each performance metric taken with respect to oversampling method is given at the top of each facet in the plots. Distributions for the balanced accuracy and Kappa coefficient are given in Figure 4.5. We will discuss the full results however which can be found in the Appendices.

Models trained on unbalanced datasets produced predictions with the highest median accuracy, sensitivity, NPV, and F1 score. These also had the lowest balanced accuracy, specificity, AUC, and Kappa coefficient. Models trained on datasets that were balanced by oversampling directly from the minority class population produced predictions with the lowest median accuracy, sensitivity, and F1 score but the highest median balanced accuracy, specificity, AUC, PPV, and kappa coefficient. It had the second-highest NPV. The second-lowest median accuracy and F1 score were produced by ROS but it also produced the second-highest balanced accuracy and AUC. Furthermore, ROS produced predictions with the third lowest sensitivity, specificity, NPV, and Kappa.

SMOTE produced predictions with the second-highest accuracy and sensitivity after no oversampling but it also produced the second lowest balanced accuracy and specificity. S-SMOTE had third highest accuracy and balanced accuracy. Though S-SMOTE the second-lowest sensitivity it had the second highest specificity. SMOTE and S-SMOTE were essentially tied for third place in terms of the median AUC. All methods except for population oversampling were tied for second place in terms of the PPV but S-SMOTE had the lowest standard deviation. SMOTE and S-SMOTE had the lowest and second-lowest NPVs, respectively.

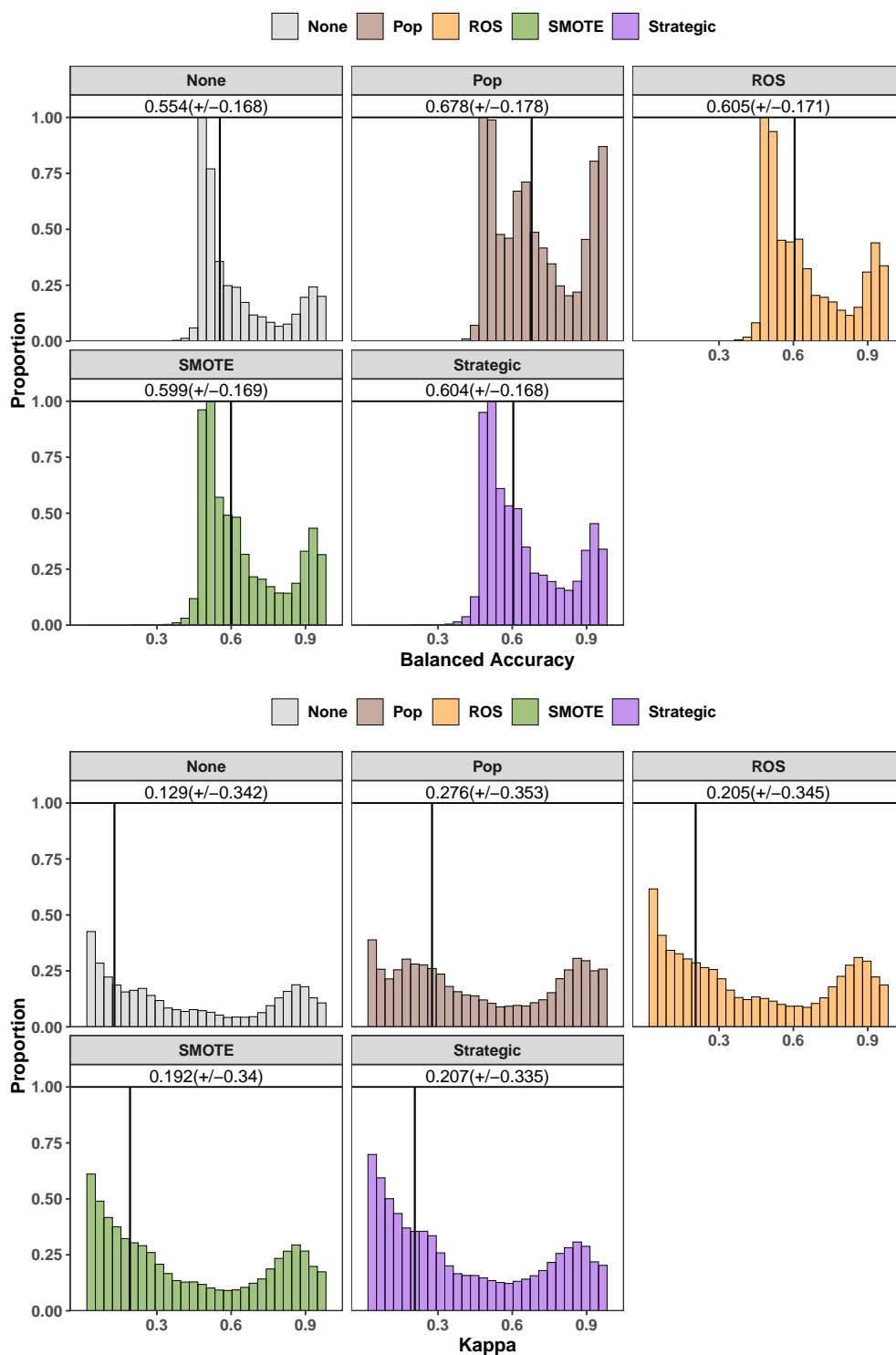


Figure 4.5: Distribution of balanced accuracy and Kappa coefficient based on oversampling method. Data were aggregated over all simulations to make an initial determination about whether the oversampling method used has an impact.

SMOTE had the second highest F1 score while Strategic smote had the third-highest one. S-SMOTE produced the second highest median Kappa coefficient while SMOTE had the second lowest one. This was a difference of 0.015 for values on a scale of 0 to 1.

It is not surprising that unbalanced data produces the highest median accuracy but the lowest balanced accuracy since accuracy is a biased measure of performance in the presence of imbalance. The same can be said of the high median sensitivity which can be made arbitrarily high when the data are imbalanced by simply predicting the positive class always. This is exposed by the low Kappa coefficient of 0.129 on a scale of 0 to 1 which indicates that the predictions and truth had little to no agreement other than that due to chance. Obtaining more minority examples directly from the population produced superior results with respect to all metrics except the accuracy, sensitivity, and F1 score. Oversampling directly from the population balances the dataset in the best possible way but there may be a small cost to pay for accuracy on the majority class. This cost is likely smaller than all other methods though since the rest of the performance metrics were still higher than those for all other methods.

Though ROS is a competitor in terms of balanced accuracy and AUC its performance with respect to the other predictive performance metrics falls short of the other oversampling methods often. This is to be expected given that ROS produces overly specific decision regions. Since SMOTE produced higher median accuracies than balanced accuracies this indicates that after oversampling with it the accuracy still may not be an accurate predictor of model performance. On the other hand, S-SMOTE produced the third highest accuracy and balanced accuracy indicating that its accuracies may better reflect performance on both classes. S-SMOTE produced the second highest specificity and second lowest sensitivity indicating that there is a tradeoff between these two. The same is true for SMOTE but it produced higher sensitivity than specificity.

There is clearly quite a bit of variability in performance metrics due to the oversampling

method applied. It is evident that population-balanced data are the best option in many cases, but this cannot be applied in reality. These will be used as a sort of baseline for understanding our simulation results. It is also evident that working with unbalanced data produces poor results but it is not always poorer than the use of oversampled data. However, these results do not take into account the model applied or any characteristics of the data. An oversampling method may prove superior in aggregate but it may be weaker when considering special cases. This brings us to our next set of results.

We also created plots of the distribution of each metric for SMOTE and S-SMOTE aggregated over the distribution of w and overlap and imbalance amount. We did facet based on the data scenario and the model used though. The median and SD taken over the distribution for w and amount of overlap and imbalance is given at the top of each panel in the plot. The text is bolded whenever S-SMOTE performed better than SMOTE. The x-axis indicates the metric being plotted. All of these plots are given in the Appendices in their full size. We provide a smaller version in Figure 4.6 for reference but our discussion pertains to the full results.

In terms of accuracy S-SMOTE performs better than or as good as SMOTE in 12/18 of the cases where there was categorical or missing data. There were 18 of these scenarios: 6 different data scenarios to which 3 models were applied. This was especially true when random forests or neural networks were applied. In other cases the accuracies produced from data oversampled with SMOTE were larger than those produced by S-SMOTE. The median balanced accuracy for S-SMOTE was higher than that for SMOTE in 9/10 data scenarios when neural networks were applied and 8/10 data scenarios when random forests were applied. However, it was only higher than SMOTE in 2/10 cases when LASSO logistic regression was applied. These were (1) when 85% of data were categorical and (2) when 70% of data were MAR. Recall that this was also when only 5% of observations belonged to the minority class.

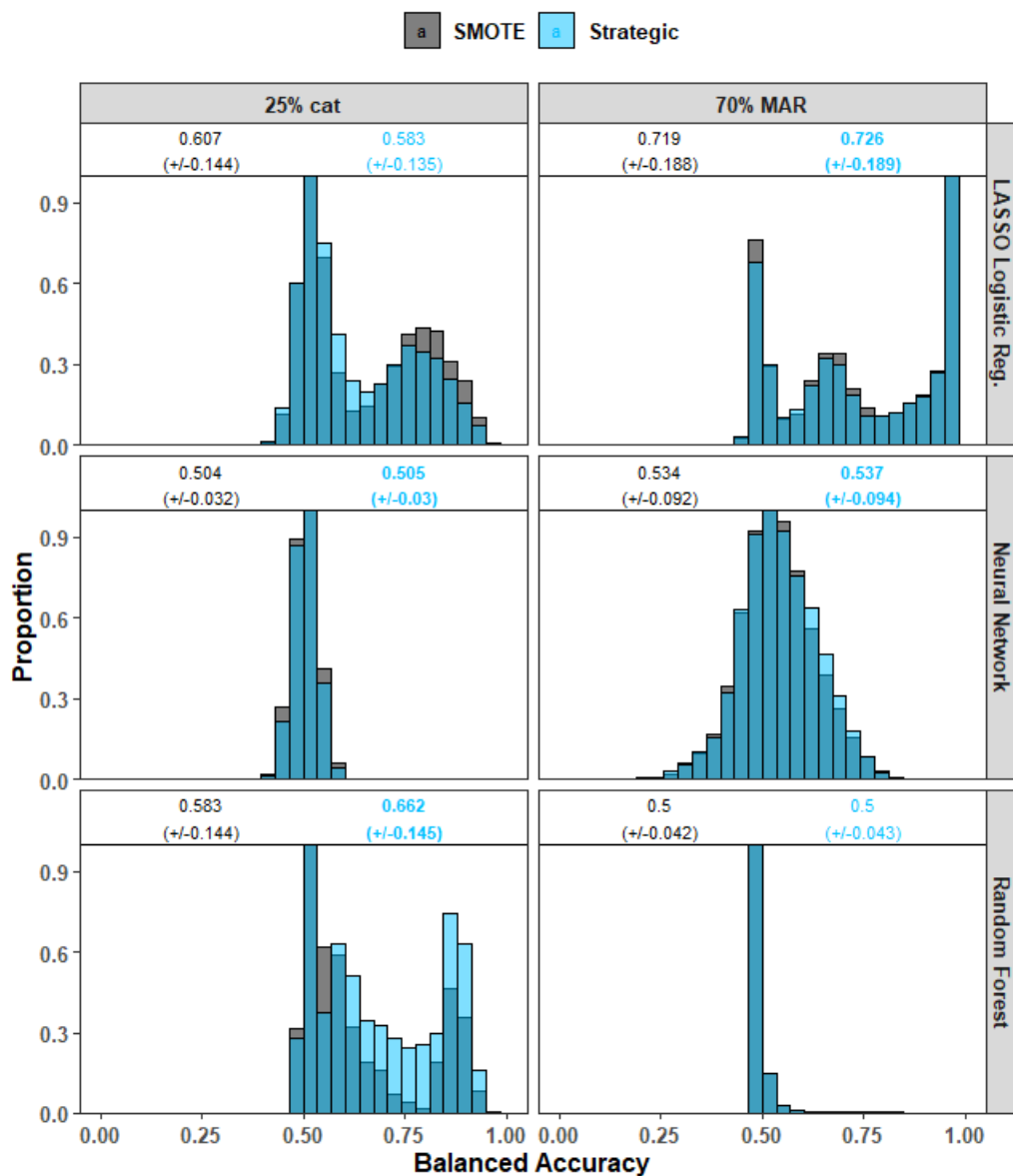


Figure 4.6: Distribution of balanced accuracy from training data where SMOTE or S-SMOTE (Strategic) was applied. Results were aggregated over the distribution of w and the amount of overlap and imbalance. Each column label of the facet indicates the change from the base case of $N = 750$, 100 variables, and no categorical or missing data.

S-SMOTE lagged behind SMOTE quite a bit when considering the median sensitivity except where neural networks were applied. In that case, it was superior to or as good as SMOTE in 8/10 cases. Otherwise, SMOTE was superior to S-SMOTE in terms of sensitivity. In terms of specificity though, S-SMOTE outperformed SMOTE in many cases, especially when random forests and neural networks were applied. There were some cases in which both the sensitivity and specificity produced by S-SMOTE was higher than or as good as that produced by SMOTE. This occurred in all cases when neural networks were applied except the base, 85% categorical, and 70% MCAR cases. This occurred in no cases where LASSO logistic regression was applied, and in the 25% categorical and all missing data cases when random forests was applied. In terms of the AUC, when logistic regression was applied we again saw that S-SMOTE outperformed SMOTE only when 85% of data were categorical and only 5% of observations belonged to the minority class. It outperformed SMOTE in all cases when neural networks were applied and mostly when data were missing or 25% of data were categorical and random forests was applied.

The results observed for the PPV from LASSO logistic regression and neural networks were similar to those observed for the specificity. There were a lot more cases where the PPV was very similar for S-SMOTE and SMOTE than the sensitivity however. When random forests was applied the PPVs produced by S-SMOTE were equal to or trivially larger than the median PPV for SMOTE. Most differences and the largest ones again occurred when data were categorical or missing. Though SMOTE performed equally to or superior to S-SMOTE in terms of NPV, in the case of 25% and 85% categorical variables, the median percentage of negative predictions that were truly negative was 18% and 22% higher, respectively, when S-SMOTE was applied than when SMOTE was applied. In terms of F1 score, SMOTE outperformed S-SMOTE in most if the cases where data were not categorical or missing. The Kappa coefficient was higher for S-SMOTE when neural networks were applied in most cases and for random forest when data were categorical or missing.

These results concerning specificity and sensitivity occur again because of the trade-off that S-SMOTE is making by paying more attention to the minority class. However, since there were some cases where both of these metrics were superior to SMOTE when S-SMOTE was applied this indicates that S-SMOTE does not always make a trade-off but rather it may actually produce better predictions on both classes at times. These results show that both the oversampling method *and* model applied impact predictive performance. Therefore, these should not necessarily be selected independent of each other. In almost every case where logistic regression was used SMOTE outperformed S-SMOTE, except when there were many categorical variables and 5% of rows were of the minority class. However, when random forests or neural networks were applied, S-SMOTE often outperformed SMOTE. Especially when there was categorical or missing data.

Lastly, these results show that in the presence of categorical variables or missing data, S-SMOTE often outperforms SMOTE when neural networks or random forests are applied. In almost every case, when 85% of data were categorical and logistic regression was applied, S-SMOTE also outperformed SMOTE. This leads us to believe that the case of categorical and missing data should be furthered study. Though these results show that there are definite differences in performance based on whether SMOTE or S-SMOTE is applied, it is not clear how the amount of imbalance and overlap impacts these results. It also is not clear how the selection of the distribution for w impacts these results. We have only studied the individual data cases by model type and we aggregated over imbalance and overlap amounts and the distribution of w . In order to incorporate information on these other characteristics of our simulations even more detailed plot are needed. We discuss these next.

We begin with the median performance metric taken with respect to the oversampling method, including SMOTE and S-SMOTE with each of their distributions for w , data scenario, amount of overlap and imbalance, and the model applied. Then we calculate the difference between the median performance metrics from predictions made on data balanced

using SMOTE or S-SMOTE and those balanced using Population data. We also calculate the difference for data balanced with S-SMOTE and SMOTE. This gives us a set of differences for each performance metric. With this information we aim to discover the specific scenarios in which S-SMOTE, SMOTE, or Population oversampling are superior to one another.

We made comparisons between SMOTE and S-SMOTE and population oversampling since we saw that population oversampling performed better than all other oversampling methods with respect to many metrics. A snippet of the results is given in Figures 4.7 and 4.8 while the full results are in the Appendices. At times the difference was very small or values were missing. For example, an NA for the NPV because there were no negative predictions. This created some facets that seemed to have no data plotted since a common x-axis scale was used.

The median accuracies produced by S-SMOTE and SMOTE were larger than those produced by population oversampling when neural networks were applied and there was any type and amount of missing data. This was always true for S-SMOTE and was true for all but one distribution of w for SMOTE. It was true for all combinations of imbalance and overlap, but the difference was largest when 15% of examples, or 5% in the case of 70% MCAR, belonged to the minority class. This was also true when lasso logistic regression was applied and the combination of overlap and % minority was (0.8, 0.15) which is the worst case scenario. In that case it was true for all distributions of w . When $w \sim FPB(5, 1, -2, 3)$ S-SMOTE showed greater improvements over population oversampling than SMOTE.

The median AUCs produced by SMOTE and S-SMOTE were mostly smaller than those produced by population oversampling. When 30% or 70% of data were MAR and neural networks were applied there were small improvements. In these cases, S-SMOTE with distributions $FPB(5, 1, -2, 3)$, $FPB(1, 3, -3, 4)$ and $G(1, 2)$ showed slightly more improved performance over population oversampling than SMOTE also applied with these distributions for w . Note that when there was any type and amount of missing data we obtained some

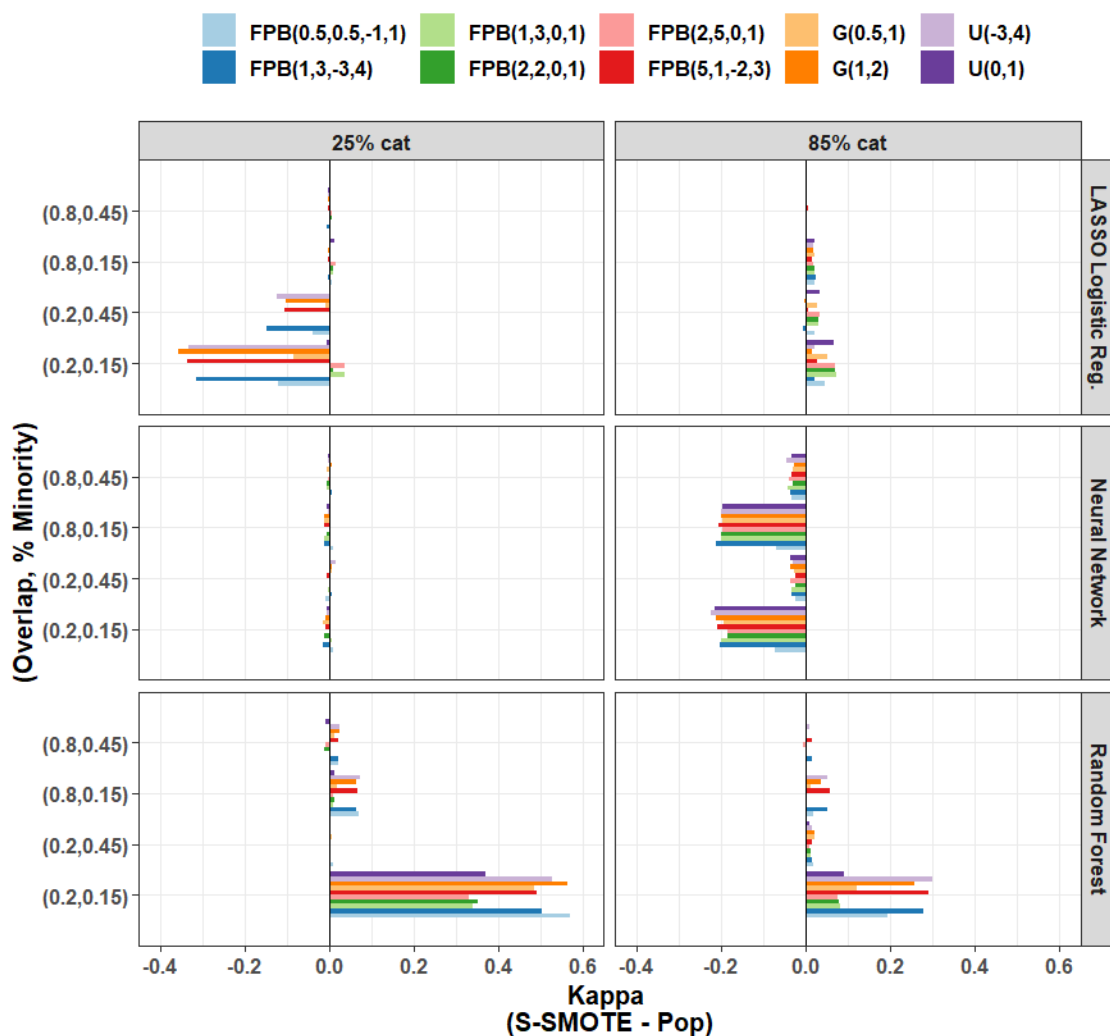


Figure 4.7: Difference in median Kappa between S-SMOTE and population oversampling. The median was calculated for a given data scenario, model, amount of overlap and imbalance, and oversampling method specific to the distribution of w . Positive differences indicate superior performance of S-SMOTE, however when this occurs it occurs for most distributions which does not clarify which version of S-SMOTE is superior.

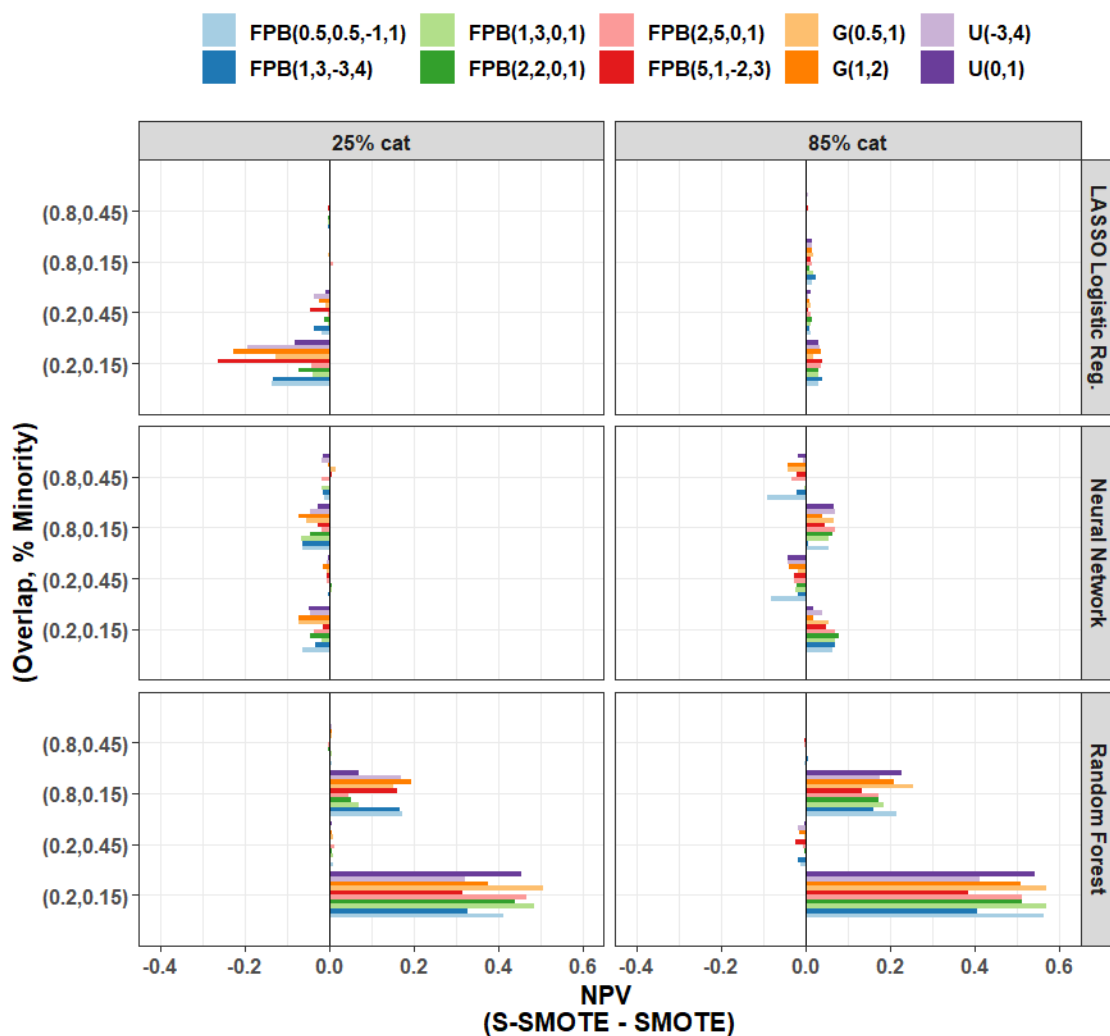


Figure 4.8: Difference in median NPV between S-SMOTE and SMOTE. The median was calculated for a given data scenario, model, amount of overlap and imbalance, and over-sampling method specific to the distribution of w . Positive differences indicate superior performance of S-SMOTE, however when this occurs it occurs for most distributions which does not clarify which version of S-SMOTE is superior.

NAs for the sensitivity, NPV, and balanced accuracy resulting from applying SMOTE and S-SMOTE with neural networks and random forests. We also consistently saw very similar performance metrics in these cases while analyzing the results of these simulations.

Due to computational issues we were not able to obtain a median balanced accuracy or sensitivity when neural networks were applied to data oversampled with SMOTE and the $FPB(5, 1, -2, 3)$ distribution or when population oversampling was applied and there was any kind of missing data of any amount. Also, the NPV from random forests were missing in three data scenarios where there was missing data. After investigating this issue we found that it occurred whenever the minority class was never predicted.

With respect to the NPV and Specificity S-SMOTE outperformed SMOTE and population oversampling when there were 25% or 85% categorical variables and random forests was used on data with 15% minority examples and overlap 0.2 or 0.8. This also held true against population oversampling when 30% or 70% of data were MCAR and the overlap and percent minority were 0.8 and 0.15, respectively. Though there were a non-trivial number of cases where population oversampling or SMOTE performed better than S-SMOTE, we noticed that when S-SMOTE *was* superior the percent in the minority class was often 0.15 (or 0.05). This indicates that S-SMOTE may have superior performance in the presence of more extreme imbalance as well as when there is missing and categorical data.

We also visualized each performance metric by data case and distribution for w aggregated over amount of overlap and imbalance, model, and whether SMOTE or S-SMOTE was applied. Figures 4.9 and 4.10 give snippets of the full results which are available in the Appendices. The majority of distributions had multiple modes indicating differences in performance due to the imbalance and overlap, model, and whether SMOTE or S-SMOTE was applied. However, these modes were often located in similar places on the x-axis across distributions for w and for the same data scenario. Upon further inspecting the medians of these aggregated distributions for the balanced accuracy, we found that the $\text{Gamma}(1,2)$

distribution was superior to others for the Base, $N = 1750$, 20 vars, and 150 vars cases.

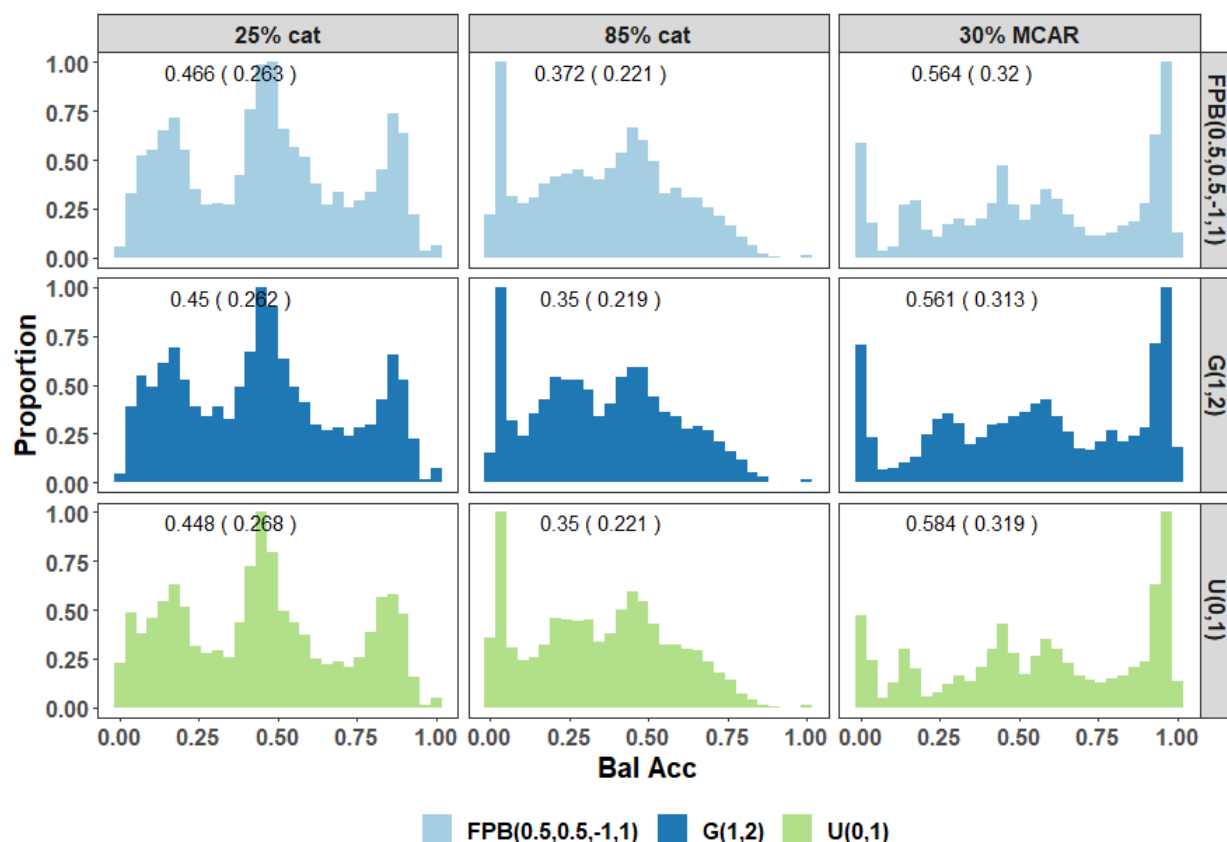


Figure 4.9: Distribution of balanced accuracy aggregated over all amounts of overlap and imbalance, models, and whether S-SMOTE or SMOTE was applied. Though there are changes in the distributions across data scenarios, distributions within a given column are quite similar. Indicating that changes in performance due to the distribution of w may be minimal in comparison to those that occur because of the underlying data scenario.

After that the $FPB(0.5, 0.5, -1, 1)$ distribution had superior performance when there was categorical data. When data were missing the differences in balanced accuracy due to the distribution of w were even smaller. When data were MCAR no one method was found to be superior with respect to the balanced accuracy. When 30% and 70% of minority data were MAR, the Gamma(1,2) and Uniform(-3,4) distributions were superior.

When $w \sim FPB(0.5, 0.5, -1, 1)$ the median specificity was superior in all data cases except those for which data were missing. The largest difference in the medians was for the base case with a 9.3% increase in the median number of true positive identified. When there

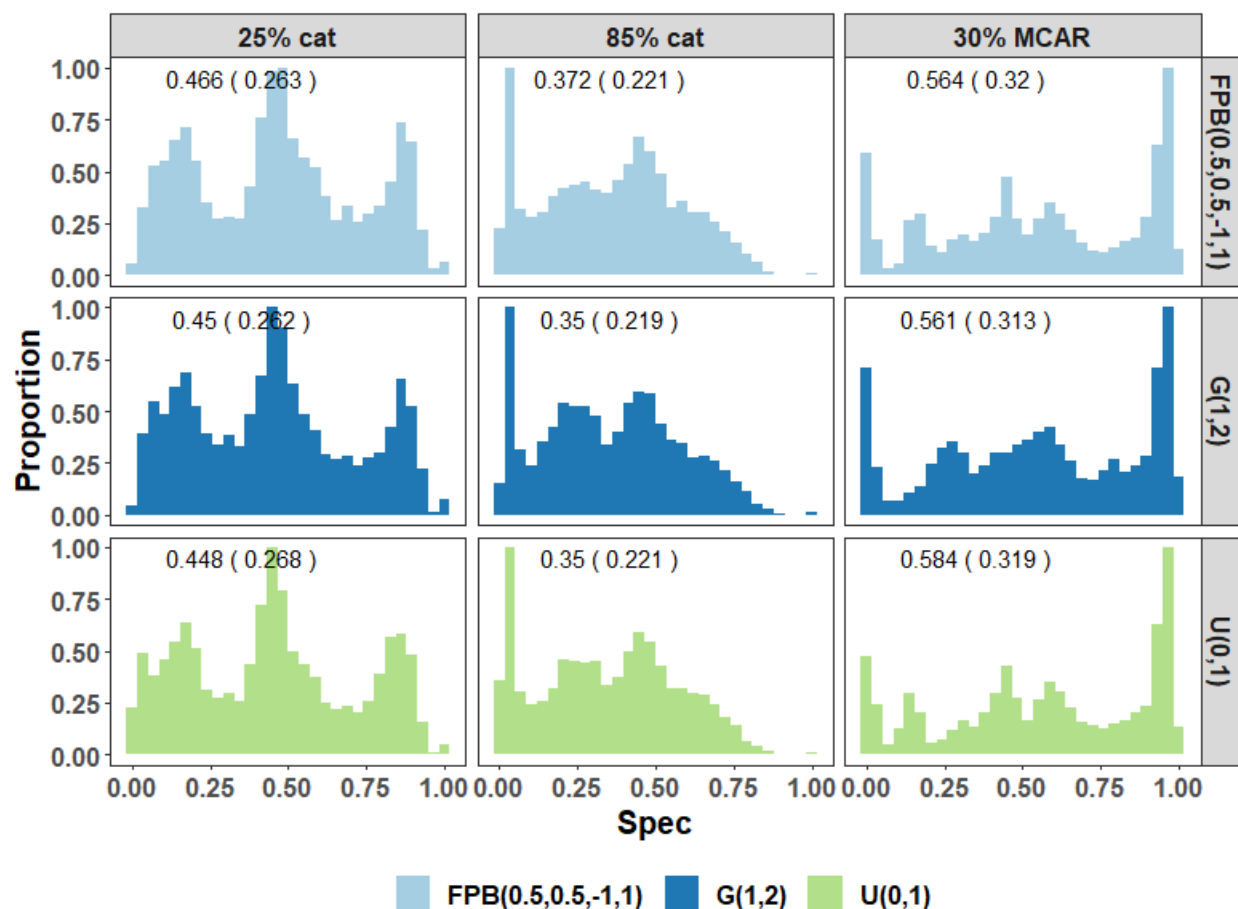


Figure 4.10: Distribution of specificity aggregated over all amounts of overlap and imbalance, models, and whether S-SMOTE or SMOTE was applied. Though there are changes in the distributions across data scenarios, distributions within a given column are quite similar. Indicating that changes in performance due to the distribution of w may be minimal in comparison to those that occur because of the underlying data scenario.

were 20 variables and 25% or 85% categorical variables, the differences were 3.8%, 2.6% and 2.2%, respectively. Though use of this distribution did not also produce the highest median sensitivities its results were competitive with those of the other distributions and there was no noticeable trade-off.

Using a minimum of -1 and a maximum of 1 with this distribution effectively performs an equal amount of interpolation and extrapolation. A key characteristic of this distribution is that it samples further away from a reference point more often than nearby (see Figure 4.11). Values between -0.5 and 0.5 are sampled about 33% of the time while values less than -0.5 or greater than 0.5 are sampled about 67% of the time. This could lead to lesser amounts of overlap introduced when this distribution is used for oversampling.

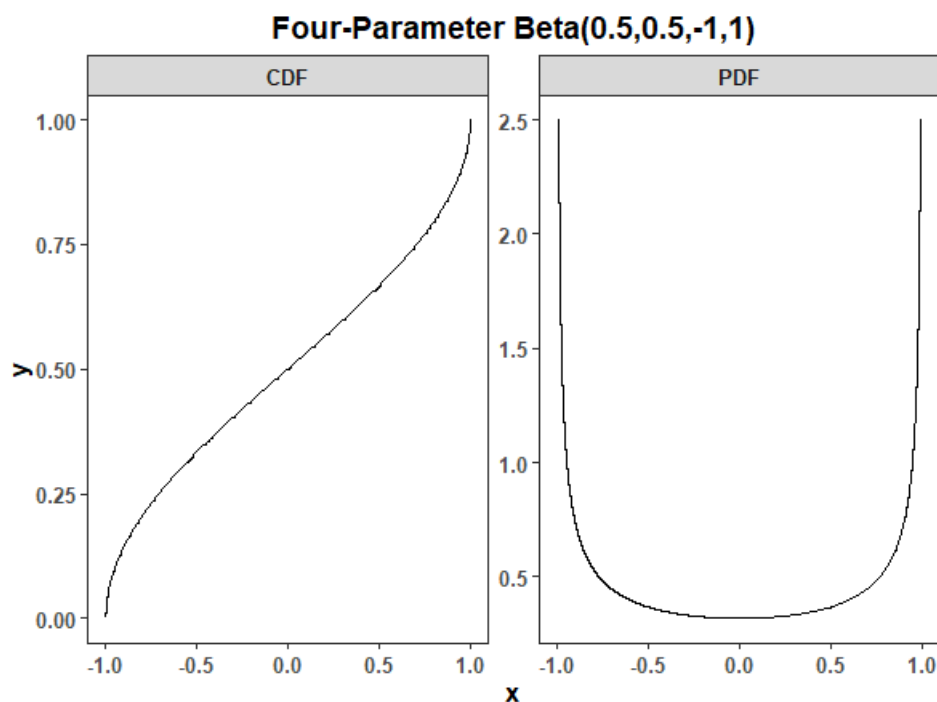


Figure 4.11: PDF and CDF of Four-Parameter Beta distribution with parameters $\alpha = 0.5$, $\beta = 0.5$, $\min = -1$, $\max = 1$.

The main purpose behind our study of the distribution of w is to determine if there is a distribution that does not introduce as much overlap into the dataset. Such a distribution can be expected to generate better predictions. To that effect, we visualized the difference

in the amount of overlap in the unbalanced training set and the oversampled training set for each method of oversampling. We again broke these down by the characteristics of each simulated data set (e.g. imbalance, overlap, missingness). Since we oversampled before fitting the model these results are the same regardless of the model applied.

We found that there was only one simulation scenario in which the median difference in overlap metrics for both classes decreased after oversampling. This was when overlap was 0.8, 45% of observations belonged to the minority class and 25% of data were categorical. This was observed for about 6 different oversampling methods, however, and there was no one distribution or method that produced this characteristic more frequently than another. Upon examining the individual differences we observed the counts in Figure 4.12 for the number of times that the difference was negative in both cases, meaning that the overlap decreased for both classes after oversampling. This occurred in only about 1% of cases out of 460,000 simulation scenarios and 6 out of 920 scenarios (0.652%) after taking the median over each set of 500 simulations. Most often this occurred when 25% of data were categorical.

To determine whether changes in the overlap between unbalanced and oversampled data impact the predictive performance we visualized their relationships with respect to the value of each performance metric. We calculated the difference in performance metrics of random forests between the test set predictions obtained using a model trained on unbalanced data and that for predictions obtained using a model trained on oversampled data. A small snippet of the results is given in Figure 4.13 but we do not include the full results in the Appendices for the sake of space and since they are redundant.

We did not observe improvements in the performance metrics from random forest for points near the origin. These points are of interest because median differences near the origin would indicate less change in the amount of overlap for each class meaning that the oversampling method tampered with the overlap a minimal amount. This was not necessarily linked to better performance though because the performance metrics were not higher in cases where

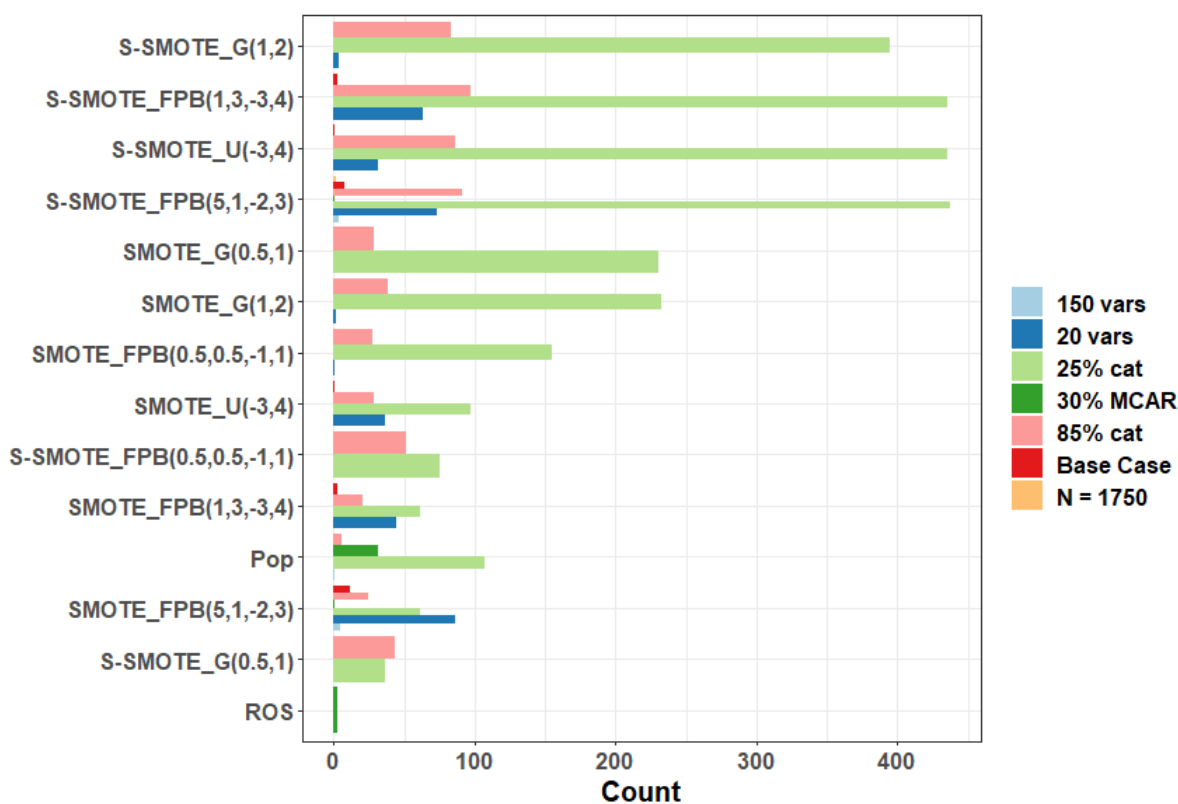


Figure 4.12: Number of times (x-axis) that difference in majority and minority class overlap both decreased after oversampling. Color and y-axis provide details of the simulation scenario in which these occurred. 500 repetitions were performed for each data scenario, amount of overlap and imbalance, and oversampling method. When 25% of data were categorical many applications of oversampling resulted in the same or less amounts of overlap in the majority and minority class.

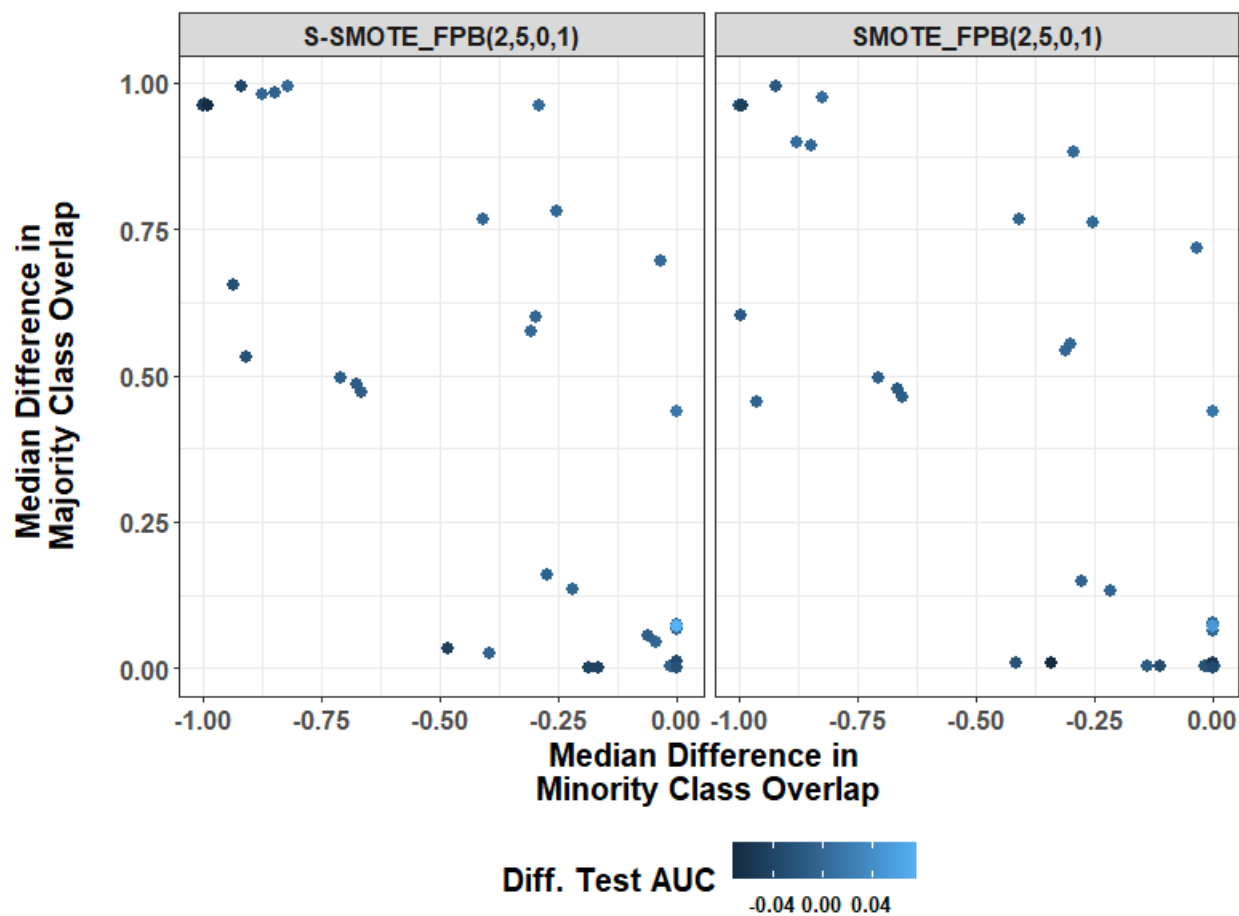


Figure 4.13: For each oversampling method the amount of overlap introduced into the data was calculated as the difference in the class overlap metrics before and after oversampling. The median differences are calculated over all 500 repetitions for a given oversampling method and data scenario. These are the same regardless of the model applied. Here we visualize their relationship as well as the difference in the AUC of test set predictions for models trained with unbalanced and oversampled data. These values did depend on the model applied. We did not observe any further relationship between the changes in overlap and the predictive performance on the test data.

the amount of overlap introduced was smaller. We did however see the same negative trend in all plots meaning that the median difference in the majority class overlap increased as the corresponding value for the minority class decreased.

In order to further confirm whether there were changes in performance due to the type of distribution used for sampling w , we ranked the median performance metrics. For each metric, the medians were calculated with respect to the data scenario, amount of overlap and imbalance, and the oversampling method and model applied. These medians were then ranked with respect to the oversampling methods. A snippet of the results for ranks is given in Figure 4.14. The fuller results are available in the appendices. We discuss the key findings out of all that we observed.

When 70% of minority class data were MAR or MCAR and random forests were applied there was no clear method that performed best in terms of balanced accuracy and all of the methods performed quite similarly. When 30% of minority class data were MAR and random forests were applied, the ranks of balanced accuracy for S-SMOTE were noticeably better than those for SMOTE but there were still many methods with little variability in between them. When data were 25% or 85% categorical and random forests was used, S-SMOTE had noticeably better ranks. In Figure 4.14 we see many of the brown points falling to the left of the green points indicating a higher rank. This was also true for 85% categorical data when LASSO logistic regression was applied. Considering other metrics, the sensitivity had very little variability in oversampling methods when data were missing and random forests were applied. Whenever the ranks for S-SMOTE or SMOTE were better than the other it was the case for multiple distributions for w . Therefore, these results do not quite provide more insight into which distribution is better.

The distribution of w may not always have any impact on how S-SMOTE and SMOTE perform. There are clearly cases (e.g. missing data) where no amount of oversampling is helpful or superior. Though we saw some changes in the performance metrics when

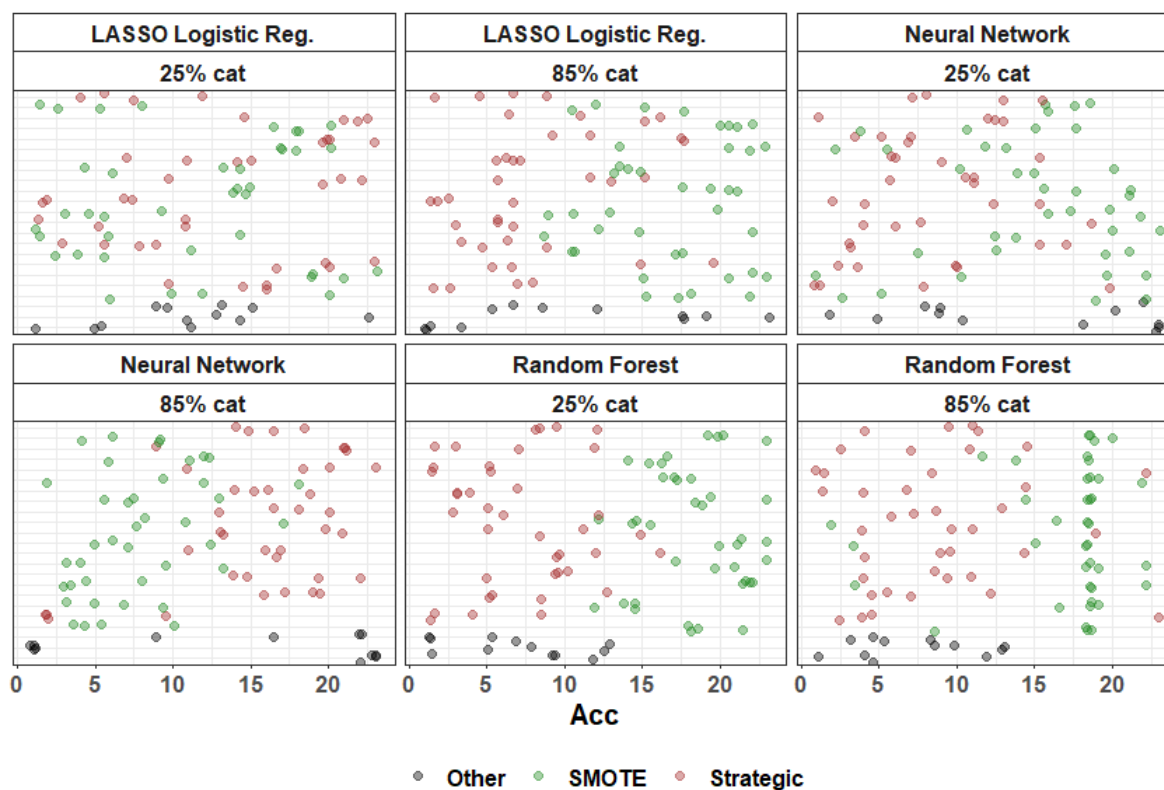


Figure 4.14: Ranks of median accuracy given on x-axis for each oversampling method on the y-axis. The y-axis labels were removed for the sake of space. Colors are mapped to the general category of oversampling method applied. Medians were calculated with respect to each data scenario, amount of overlap and imbalance, model, and oversampling method. Ranks were then calculated with respect to the oversampling methods. When superior performance of S-SMOTE or SMOTE occurs (e.g. 25% categorical and random forests) it occurs for most distributions of w . These results indicate that superior performance may be due to the general category of oversampling method rather than changes in the distribution of w .

comparing population oversampling to S-SMOTE and S-SMOTE to SMOTE, these changes were observed for all distributions of w for a specific data scenario. Changes in the underlying data scenario caused more changes in the predictive performance between methods than changes in the distribution of w . The model applied also caused more changes in performance than the distribution of w .

Though the changes observed in the median performance metrics in the plots that we have discussed so far were often small in size imbalanced classification problems can include datasets with millions of examples and only 1% of rows belonging to the minority class (e.g. credit card fraud data). In cases such as these, a small drop in our ability to predict the majority class can result in a major decrease in overall predictability. Meanwhile a large increase in our ability to predict the minority class result in only a small increase in overall predictability. The impact this has on the metrics that we have discussed depends on which of these is the positive class. Either way, the larger and more imbalanced the dataset, the more of an impact a few incorrect or correct predictions can have on the class-specific and overall accuracy.

Therefore, we will move forward with the oversampling method that proved to be best with respect to minority class performance metrics and had comparable performance with respect to other metrics. Due to its superior performance in terms of the sensitivity in many data scenarios we will move forward with the distribution $FPB(0.5, 0.5, -1, 1)$ in our algorithm. However, future work will involve further studying the impacts of the distribution of w , possibly from a theoretical approach.

Though our results are not quite conclusive, they do provide evidence that there is some type of impact on predictive performance as w changes. These simulations have helped us to characterize the issue of oversampling in a variety of scenarios where overlap and imbalance are present. When random forests were applied in scenarios with the largest amount of imbalance considered and there were categorical variables, S-SMOTE showed

superior performance. It also had comparable or better performance in many scenarios where minority class examples were MAR.

We will now proceed to further investigate the other elements of S-SMOTE. By way of reminder these were the number of neighbors checked for dominance, the sequence of dominance thresholds checked for, the weights with which quadrants 1-4 of Figure 4.1 are sampled, and the percentage of minority class examples that should be approved for oversampling before the search ends.

4.4 Studying Hyperparameters of Strategic SMOTE (S-SMOTE)

The number of neighbors that S-SMOTE will check the dominance of is denoted as k_{max} . The distances between a reference point and k_{max} minority neighbors are used to specify the neighborhoods around the point. These neighborhoods are then checked to determine what proportion of minority class points are as far as the k -th neighbor, $k = 1, \dots, k_{max}$, out of all points that are as far as it. This proportion is called the dominance for the k -th neighborhood.

We denote a set of dominance thresholds that we desire for the neighborhoods to meet as $\eta = (\eta_1, \dots, \eta_l)$. This is a decreasing sequence of values such as 0.60, 0.58, \dots , 0.20 whose length varies. If this dominance is greater than or equal to the current threshold η_1 then the neighborhood is said to be minority class dominated and we move to the next neighborhood made up of all points as close as the next furthest minority neighbor and check the dominance there as well. This process continues until a neighborhood that does not meet the threshold is found. We then move back to the previous neighborhood and use it for oversampling.

The minimum proportion of minority points that we would like to use for oversampling is denoted as p_{min} . This value should be set to at least 0.75 at the onset of the algorithm

to ensure that enough information on the minority class is used for oversampling. After all minority points are checked for a neighborhood with a dominance of at least η_1 the algorithm then checks whether at least p_{min} minority points were deemed fit for oversampling. If not, then the algorithm decreases the threshold to η_2 and searches for areas that meet this new dominance threshold. This process repeats itself until the proportion of points that we have found neighborhoods for reaches p_{min} or we have reached η_l . If no points were deemed fit for oversampling in the end an error is returned and the values in η can be adjusted.

When k_{max} is set to a larger value (e.g. 20 or 30) S-SMOTE is able to find larger neighborhoods that are dominated by the minority class, if they exist. This would lead to the use of more minority neighbors for oversampling to create large areas dominated by the minority class. If the threshold is also high then this can be done without biasing majority class areas of the feature space. However, if the algorithm has already run for a few iterations and the threshold has been lowered then a large value for k_{max} could result in finding large areas of the feature space that are *weakly* dominated by the minority class. Oversampling in these areas could contribute to biasing more areas that truly belong to the majority class.

If S-SMOTE is provided with a large value for η_1 , such as 0.98 or 0.95, the search for the best areas to oversample in will begin with stricter criteria. How long the search takes depends on the difference between η_1 and η_l and the increments between values in η . As the difference between η_l and η_1 increases and the space between values in η decreases the algorithm takes longer to run. Though using small increments for η will take longer doing so may prevent the algorithm from skipping over better dominated areas. For example, checking for a dominance of at least 0.80 and then 0.78 would result in finding areas that are more strongly dominated than if the algorithm checked for dominance thresholds of 0.80 and then 0.76. In the latter case the algorithm may skip over the areas that have dominance thresholds of 0.78.

It is important to select a value for k_{max} that is small enough to avoid biasing majority class regions of the feature space as the threshold decreases. Future work involves studying

the performance of this algorithm with an adaptation that allows k_{max} to decrease with the threshold. This would effectively oversample the largest and most strongly dominated areas first and then smaller more weakly dominated areas. To determine the impact of k_{max} and η on performance we simulated datasets with varying amounts of imbalance and overlap and applied S-SMOTE to these unbalanced datasets with varying k_{max} and η . We then performed 5-fold cross validation to select hyperparameters for a random forests model for each dataset. The hyperparameters of random forest (number of trees, minimum node size, and number of variables tried) that maximized the ROC were used to fit the final random forest model and we obtained predictions on the test set. We performed 50 simulations for each scenario and datasets were simulated in the same manner that was used for the larger simulations discussed earlier.

The results for the balanced accuracy are given in Figure 4.15 while the appendices contains those for other performance metrics which we will also discuss. Specifically, Figure 4.15 provides the distribution of the balanced accuracy (y-axis) for various values of k_{max} (x-axis). The results are faceted by the sequence of thresholds and the amount of overlap and percent minority points in the simulated dataset. When 5% of points belonged to the minority class 700 synthetic samples were still generated as in the 15% case. This was done to avoid using very little information on the minority class for a large amount of oversampling.

The variability in the spread and center of the distributions of each of the performance metrics decreases as k_{max} increases and the distributions become more similar. The most obvious changes come from the amount of overlap and imbalance in the simulated data set to which S-SMOTE was applied. Comparing the distributions of accuracy to those of the balanced accuracy shows that the accuracy is biased whenever the imbalance is high. The balanced accuracy better reflects the true performance of the models as we would expect.

Changes observed in the distribution of most performance metrics as k_{max} varied were similar for a given amount of overlap and imbalance even if η differed. This was true for most

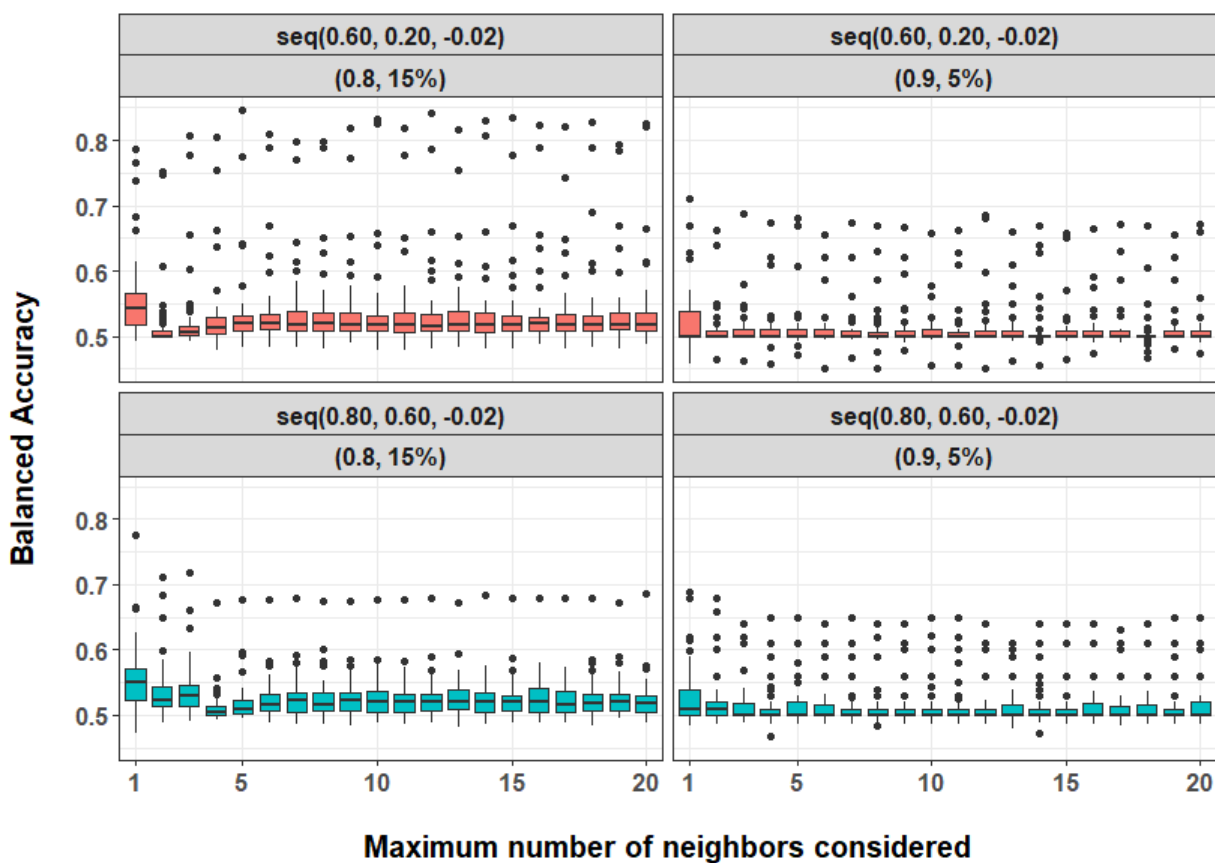


Figure 4.15: Distribution of balanced accuracy as the maximum number of nearest neighbors k_{max} considered by S-SMOTE varies. Facet labels correspond to the sequence of dominance thresholds checked (top facet label) and the amount of overlap and percent minority examples in the simulated data (bottom facet label). Data were simulated with 60 columns, 30% categorical variables, and 1000 examples before oversampling. 700 synthetic examples were generated.

performance metrics except the AUC and the NPV. Using the stricter thresholds of values between 0.80 and 0.60 decreasing by 0.02 produced distributions of the AUC that were shifted slightly higher than those when thresholds between 0.60 and 0.20 decreasing by 0.02 were used. However, when weaker thresholds were used the results were less variable. The distributions for the NPV had more spread, larger centers, and less variability across k_{max} when the stricter set of thresholds was used.

There were some instances in which no predictions were made for the minority class and the NPV was missing. Figure 4.16 provides the proportion of times that this occurred out of each set of 50 simulations. This occurred most often when the percent of points in the minority class was 5%. In addition, the less strict threshold produced slightly more NAs than the stricter threshold. A model that never predicts the minority class when provided with test data has failed to learn decision rules for the minority class that generalize well to unseen data. This can occur when the model over-learns the decision regions of the minority class that are specific to the training set. It can also occur if the feature space is so over-saturated with majority class examples that the model simply does not learn to predict the minority class.

For each minority point that S-SMOTE deems fit for oversampling there is a corresponding number of neighbors that will be used for oversampling. This defines the neighborhood where oversampling is performed and its dominance defines how saturated this area already is by minority class instances. These two characteristics - the dominance and density of the neighborhood where oversampling takes place - are used to determine how often each minority class point is selected for oversampling. Minority class neighborhoods that contain only a few minority points and that are not well dominated, relative to other areas deemed fit for oversampling, should be oversampled first.

The median relative dominance of the neighborhood around each minority point and the number of neighbors used for oversampling it can be used to determine whether a point is

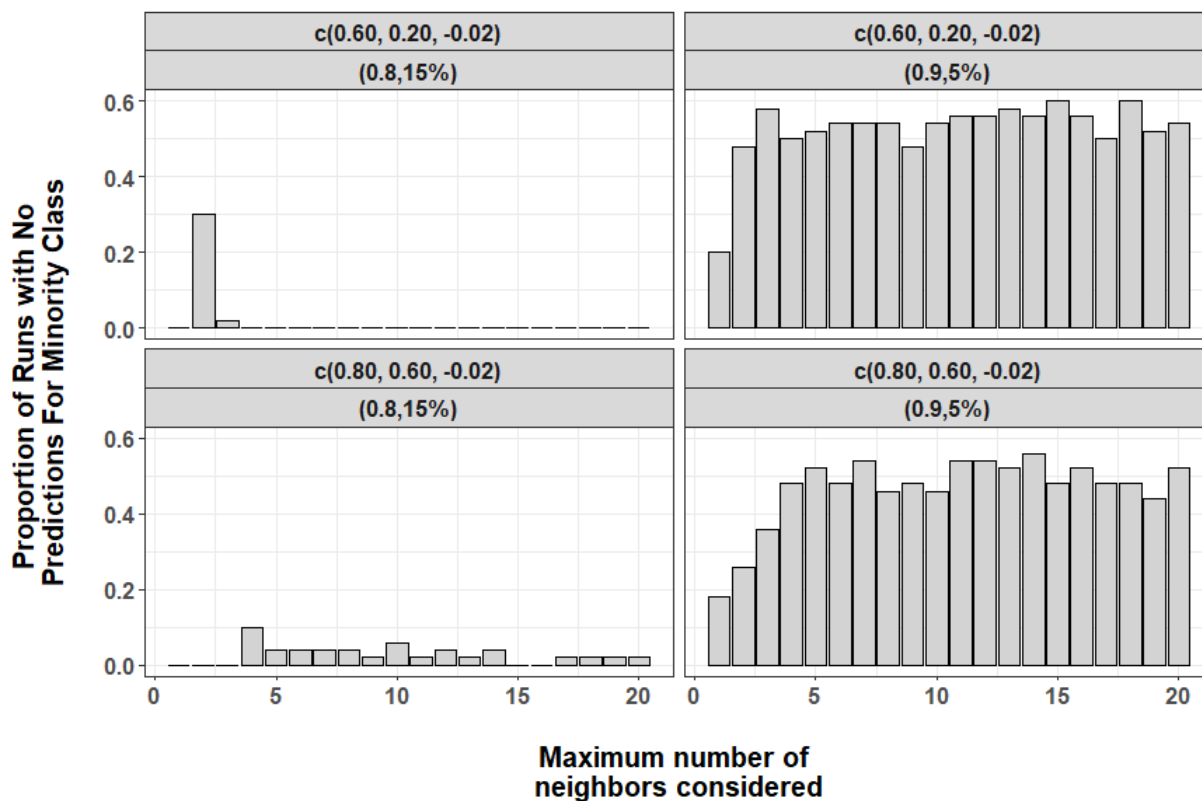


Figure 4.16: Proportion of simulations in which no predictions were made for the minority class out of 50. When 5% of points belonged to the minority class 700 synthetic samples were still generated as in the 15% case. This was done to avoid using very little information on the minority class for a large amount of oversampling.

more or less dominated and crowded relative to other points. Categorizing the points that will be oversampled using these medians was exemplified in Figure 4.1. We denoted the vector of weights used for oversampling points in each quadrant of such a plot as ρ . This effectively oversamples points that are in sparser more weakly dominated areas more often.

The default that we have used for ρ thus far in our simulations has been $\rho = (0.1, 0.3, 0.5, 0.1)$. We used simulations to gain insight on how the values in ρ impact the performance of models fit to data oversampled with S-SMOTE. Figure 4.17 gives the distribution of the balanced accuracy for varying ρ . The results are faceted by the amount of overlap and imbalance and what percent of variables were categorical. Results for other metrics are given in the appendices for brevity but we will discuss these here as well.

The distributions for the balanced accuracy were quite similar for the first scenario (left-most panel) in Figure 4.17 but the distribution of $\rho = (0.1, 0.4, 0.4, 0.1)$ was slightly shifted above the others. The distributions shifted up as ρ_1 decreased and more weight was distributed to other quadrants. When 85% of variables were categorical (middle panel) the distributions were even more similar in spread and center and there were very small changes in the center and spread of the distributions. When only 5% of points belonged to the minority class (right-most panel) there was little to no variability in the balanced accuracy as compared to other data scenarios and the boxplot can hardly be seen. The values ranged from 0.486 to 0.594 for all weight vectors in this case. This lack of variability was observed for all metrics except the NPV and AUC for data with 5% of points belonging to the minority class.

The distributions for the accuracy showed similar patterns to those of the balanced accuracy but they were shifted much higher due to its susceptibility to the imbalance. In the first scenario the distributions for the sensitivity shifted down as ρ_1 decreased but those for the specificity increased as ρ_1 decreased. This indicates that oversampling more or only in neighborhoods that are the least relatively dominated and crowded by the minority class benefits the majority class. Meanwhile sampling in all minority neighborhoods improves the

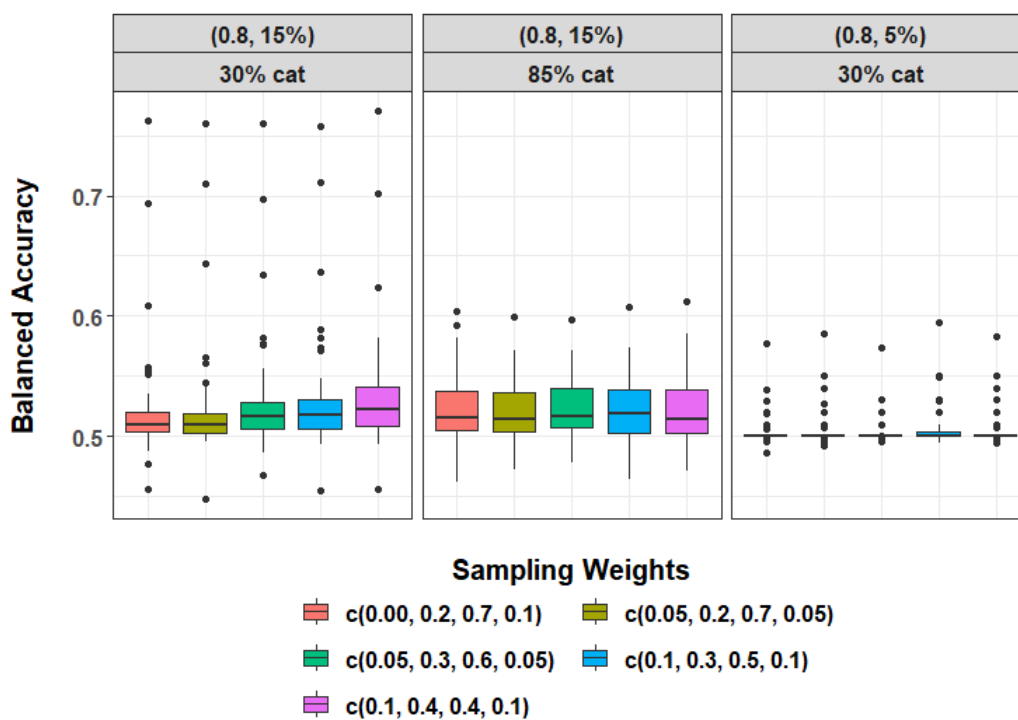


Figure 4.17: Distribution of balanced accuracy as the weights used to select minority examples for oversampling ρ varies. The median dominance threshold and median number of neighbors used for oversampling was used to determine whether a point was more or less dominated and crowded, respectively. The median was calculated with respect to all minority class points deemed fit for oversampling. Facet labels correspond to the amount of overlap and percent minority examples in the simulated data and characteristic of the data. Each simulated dataset had 60 columns and 1000 rows before oversampling. 700 synthetic examples were generated.

specificity though it decreases the sensitivity. The distributions of the AUC, PPV, F1 score, and Kappa were quite similar as ρ differed but these varied across data scenarios. There were again missing values for the NPV when 5% of examples belonged to the minority class.

These results show that the performance of S-SMOTE depends mostly on the difficulty of the data to which it is applied. Our simulations support setting $k_{max} = 15$ since trivial changes in the distribution of performance metrics were observed for $k_{max} \geq 15$ and using a very large k_{max} will increase computing time. If the number of minority examples is less than 16 then $k_{max} = n_{min} - 1$, the number of minority examples minus one. This is the maximum number of neighbors possible. Due to the less variable AUC observed for $\eta = (0.6, 0.58, \dots, 0.20)$ we continue forward with this vector of thresholds. In order to curb the trade-off between specificity and sensitivity that results from oversampling, $\rho = (0.05, 0.3, 0.6, 0.05)$ will be used. This vector had comparable performance for many metrics and its use produced the third highest median sensitivity and third lowest specificity. These medians were fairly similar so selecting this vector for ρ^* could minimize the loss on both the sensitivity and specificity. These simulations were performed with only 50 repetitions in order to provide time for properly training the random forest models. Future work includes performing a larger number of simulations to obtain more accurate results.

The original SMOTE algorithm is a special case of S-SMOTE. Let $\eta = \{0\}$, $k_{max} = k$ (a single value for all minority examples), $p_{min} = 1$, and $\rho = \{1/n_{min}\}$. In this case all k neighbors are used for oversampling and all minority points are oversampled with the same probability until the desired balance is reached. In this case our algorithm is the same as SMOTE except that we use Gower's distance to incorporate information on the categorical variables when calculating distances. If we additionally calculated the Euclidean distance using the quantitative variables and added the median standard deviation of the quantitative variables to the distance for each mismatched categorical variable then our algorithm would be exactly the same as SMOTE-NC.

4.5 Statement of S-SMOTE Algorithm

Algorithm *S-SMOTE*(data, drop, nsynth)

Input: Unbalanced training data with response variable *data*; Column index of response variable in data *drop*; The number of synthetic examples desired which will default to the amount needed for complete balance *nsynth*

Output: The balanced dataset with *nsynth* synthetic examples and a flag indicating which rows are synthetic

1. *minority* = level of the response variable
2. *nmin* = number of minority examples
3. *nmax* = number of majority examples
4. *If the number of synthetic points is not defined then calculate and use the number needed for perfect balance.*
5. if nsynth = NULL then nsynth = nmax - nmin endif
6. *minexs* = row indices of minority examples
7. *isfact* = column indices of nominal variables
8. *distances_minority* = matrix of distances between each minority example and all other examples with *nmin* rows and (*nmin* + *nmax*) columns
9. $k_{max} = \min\{15, nmin - 1\}$ = the maximum number of nearest neighbors of the minority class that we will check the dominance of.
10. *threshestotry* = seq(from = 0.60, to = 0.20, by = -0.02) (*S-SMOTE will iteratively check for minority class areas that contain this proportion of minority class examples as near as the k-th nearest neighbor.*)
11. *run*: keeps count of how many values we have used from *threshestotry*, initialized to 0
12. *donemin*: tracks of the proportion of minority examples that S-SMOTE has deemed fit for oversampling, initialized to 0
13. *nminseq*: minority examples that have yet to be deemed fit for oversampling, initialized

to seq(from = 1, to = $nmin$, by = 1)

14. $rnmstouse$ = a list containing the neighbors deemed fit for use to oversample each minority example, initially has length $nmin$ (*Each vector in the list is initially empty and remains empty if an example is never deemed fit for oversampling by S-SMOTE.*)

15. k_to_use = number of neighbors deemed fit for use when oversampling each minority class example, initially has length $nmin$

16. $prop_last_met_thresh$ = final dominance of neighborhood around each minority example after S-SMOTE has exited, initially has length $nmin$ (*Each element in the vector is initially zero and remains zero if an example is never deemed fit for oversampling by S-SMOTE.*)

17. while $donemin < 75\%$ and $run < \text{length}(threshestotry)$

18. $run = run + 1$ (*Update the number of runs with each iteration.*)

19. $thresh = threshestotry[run]$ (*Decrease threshold after each run.*)

20. for i in $nminseq$ (*For each minority class example.*)

21. $prop_less_k_dist_in_min$ = the proportion of minority class points that are as far as the k -th nearest neighbor, length k_{max}

22. k_fornow = the value k before which $prop_less_k_dist_in_min$ first drops below $thresh$, will be k_{max} if this never occurs

23. $rnmstouse[[i]]$ = save neighbors 1 through k_fornow to use for oversampling minority example i

24. $k_to_use[i] = k_fornow$ how many neighbors will be used to oversample minority example i

25. $prop_last_met_thresh[i] =$ proportion of minority class points that are as far as k_fornow

26. endfor

27. $nminseq =$ minority examples that have yet to be deemed fit for oversampling

28. endwhile

29. *If all of the values in k_to_use are 0 a warning is issued that S-SMOTE was unable to find any well-dominated minority examples and the algorithm exits.*
30. $rel_prop_last_met_thresh = prop_last_met_thresh / \max(prop_last_met_thresh)$ (*Calculate the dominance relative to the largest minority neighborhood dominance found.*)
31. *Oversampling will take place based on characteristics: (1) number of neighbors used, which corresponds to the size of the area that we are oversampling in and (2) the dominance met, which corresponds to the strength of the area. Points in quadrant 3 will be oversampled the most since they are the most weakly dominated relative to other areas and also have lower density. Then points in quadrant 2, quadrant 4, and lastly quadrant 1 will be oversampled, respectively.*
32. $quad1 = \text{minority examples with } k_to_use > \text{median}(k_to_use) \text{ and } rel_prop_last_met_thresh \geq \text{median}(rel_prop_last_met_thresh)$
33. $quad2 = \text{minority examples with } k_to_use > \text{median}(k_to_use) \text{ and } rel_prop_last_met_thresh < \text{median}(rel_prop_last_met_thresh)$
34. $quad3 = \text{minority examples with } k_to_use \leq \text{median}(k_to_use) \text{ and } rel_prop_last_met_thresh < \text{median}(rel_prop_last_met_thresh)$
35. $quad4 = \text{minority examples with } k_to_use \leq \text{median}(k_to_use) \text{ and } rel_prop_last_met_thresh \geq \text{median}(rel_prop_last_met_thresh)$
36. $initialweights = (0.05, 0.3, 0.6, 0.05)$ (*These are used to perform a weighted selection from the approved minority samples for the creation of each synthetic example. A normalized version is used if one of the quadrants has no points in it.*)
37. $exsprobs = \text{probabilities for each quadrant divided by and repeated for the number of points in that quadrant}$
38. $synthelist = \text{list of new synthetic examples with length } nsynth$
40. $synthdist = \text{distribution that multiplier for interpolation or extrapolation will be sampled from. (This is the Four-parameter Beta distribution with shape1 = 0.5, shape 2 = 0.5, min = -1, and max = 1.)}$

39. for i in 1 to length(*synthexlist*) (*For the creation of each synthetic minority example.*)
40. *refpoint* = data vector of a randomly selected point from the approved minority points using *exsprobs* to weight the sampling, its index is denoted as *refind*
41. *neigh* = data vector of a randomly selected neighbor of *refpoint* from their approved neighbors contained in *rnmstouse*
42. w = a value randomly drawn from *synthdist*
43. *synthexlist*[[i]][*-isfact*] = *refpoint*[*-isfact*] + $w^*(\text{neigh}[\text{-isfact}] - \text{refpoint}[\text{-isfact}])$ (*Use interpolation/extrapolation to generate quantitative values for synthetic points.*)
44. *synthexlist*[[i]][*isfact*] = mode(data[rnmstouse[[*refind*]], *isfact*]) (*The mode of each categorical variable taken over all approved neighbors of refpoint is assigned to the synthetic example.*)
45. *synthexlist*[[i]][*drop*] = minority (*Assign minority level of response variable to synthetic example.*)
45. endfor
46. return rbind(data, synthdat) (*Combine synthetic data and original data and return balanced dataset.*)

Chapter 5: Example Applications of S-SMOTE to Real Data

In the previous chapter we used simulated data to evaluate and compare the performance of SMOTE and S-SMOTE. In this chapter we will apply S-SMOTE to real-world datasets in order to evaluate its performance outside of a simulation setting. First we will revisit the dataset from OSU that we discussed earlier. This dataset was used to obtain inferential findings about the relationship between total gift aid and student success after accounting for a variety of demographic and academic factors. We also began working towards predictive modeling with this dataset when we ran into the issue of overlap and imbalance. We will return to the predictive component of that problem in this chapter and apply S-SMOTE.

We will also discuss other real-world examples of imbalance and overlap and evaluate the performance of S-SMOTE in these cases. We will use the Pima Indians Diabetes dataset (Smith et al. 1988) and Haberman’s survival data (Haberman 1976) for this. These were downloaded from Kaggle and the University of California Irvine (UCI) Machine Learning Repository, respectively. These examples have been used as benchmark datasets in many research papers on imbalance and overlap (e.g. Napierała et al. 2010; D. Li et al. 2010; Jeatrakul et al. 2010; Soh and Yusuf 2019; Nivetha et al. 2020; Mqadi et al. 2021). These benchmark examples come from the medical industry while the OSU data come from the education sector. These are just a few examples of the many areas in which one may encounter data with imbalance and overlap.

5.1 Revisting OSU Data

We return to the predictive component of the problem of predicting student success by fitting random forest and neural network models with SMOTE and S-SMOTE applied. We compare their performance to each other and the use of unbalanced data. In order to train random forest models we first split the entire OSU dataset into training (70%) and testing (30%) sets. We used stratified sampling to do this in order to preserve the class distributions in each. We selected 25 of the most pertinent variables to predicting first-year retention from the entire training set. Some variables such as a student's major upon entry had more than 52 levels and could not be used in random forests.

We fixed missing values in these datasets using median-mode imputation with respect to the classes. Then we applied SMOTE and S-SMOTE to the unbalanced training data and moved forward with three training sets - two balanced via S-SMOTE and SMOTE and the unbalanced set. In order to tune the hyperparameters of random forest we selected the combination of values that maximized the 5-fold cross validation AUC for each of these training sets. These were selected out of all combinations of: minimum node size = 1, 10; number of variables to try at each split = 2, 5; and number of trees = 100, 250, 500.

A similar process was used with neural networks, however, we split the entire dataset into training (50%), testing (30%), and validation (20%) sets. Due to the usefulness of neural networks for larger datasets we selected all columns that an administration office would truly know at the end of a student's first year. We oversampled the data, performed median-mode imputation, one-hot encoded categorical variables, and scaled the data before applying neural networks. Means and standard deviations from each of the training sets were used to scale the validation and test sets. Hyperparameters for the neural networks were selected from all combinations of the following values: dropout rate of first and second hidden layer = 0.1, 0.2, respectively; number of hidden units in two hidden layers: 5% and 10% of the

number of columns. All models were trained for 100 epochs with a batch size of 32. The set of values corresponding to the maximum validation data AUC after 100 epochs were used to fit final models.

After fitting the final models we obtained predictions on the corresponding test sets. Performance metrics for these predictions are given in Table 5.1. Italicized text indicates a maximum with respect to that column and type of model. When random forests were applied S-SMOTE had superior performance in comparison to SMOTE with respect to all metrics except for the sensitivity, NPV, and McNemar’s p -value.

Model + Oversampling Method	Accuracy	Bal. Accuracy	Sensitivity	Specificity
Random Forest + SMOTE	0.913	0.725	0.995	0.454
Random Forest + S-SMOTE	<i>0.933</i>	<i>0.909</i>	0.943	<i>0.875</i>
Random Forest + None	0.909	0.710	<i>0.996</i>	0.424
Neural Networks + SMOTE	0.503	0.515	0.497	0.532
Neural Networks + S-SMOTE	0.341	0.513	0.267	<i>0.759</i>
Neural Networks + None	<i>0.683</i>	<i>0.612</i>	<i>0.715</i>	0.508

Model + Oversampling Method	AUC	PPV	NPV	F1 Score	Kappa	McNem. p
Random Forest + SMOTE	0.964	0.911	0.944	0.951	0.571	0.000
Random Forest + S-SMOTE	<i>0.973</i>	<i>0.977</i>	0.733	<i>0.960</i>	<i>0.757</i>	0.000
Random Forest + None	0.965	0.906	0.947	0.949	0.543	0.000
Neural Networks + SMOTE	0.520	0.856	0.159	0.629	0.015	0.000
Neural Networks + S-SMOTE	0.535	0.861	0.156	0.407	0.010	0.000
Neural Networks + None	<i>0.656</i>	<i>0.890</i>	<i>0.242</i>	<i>0.793</i>	<i>0.154</i>	0.000

Table 5.1: Performance metrics from predictions for test OSU dataset. These were obtained using random forests and neural networks. Data were oversampled with S-SMOTE or SMOTE or were left unbalanced. Italicized digits are the maximum for that metric out of all model and method combinations. Performance of neural networks with unbalanced data was better than their performance when SMOTE or S-SMOTE was applied.

The p -value from McNemar’s test is the resulting p -value from a test of the null hypothesis that the number of discordant pairs (truth, prediction) are equally split. These values correspond to cells $b = [1,2]$ and $c = [2,1]$ of a confusion matrix. A small p -value provides evidence that b is significantly different from c and therefore the model is misclassifying one class more than another. The test was always rejected regardless of the model or oversampling method applied. The sensitivity produced by S-SMOTE when random forests were

applied was about 5% lower than that of SMOTE or the unbalanced data. Meanwhile the specificity was over 40% higher than when SMOTE oversampled or unbalanced data were used for training. This indicates that S-SMOTE performs a worthy trade-off in order to increase the minority class accuracy. That being said, S-SMOTE performed poorly when applied with neural networks. With respect to the accuracy, balanced accuracy, NPV, F1 score, and Kappa it lagged behind SMOTE oversampled and unbalanced data. These results agree with those seen in our simulations where performance results depend on both the model applied and the oversampling method used.

We also assessed the calibration of these models and these results are plotted in Figure 5.1. The application of S-SMOTE with random forests produced a slightly wider range of predictions than use of SMOTE oversampled or unbalanced data. Use of S-SMOTE resulted in less under-prediction than there was over-prediction when SMOTE oversampled or unbalanced data were used. S-SMOTE continued to under-predict however, even for higher probabilities, while SMOTE and unbalanced predictions neared the reference line. In the case of neural networks all three oversampling options performed poorly and results from SMOTE or S-SMOTE oversampled data had more variability. All three options under-predicted quite a bit and this decreased as the average predicted probability of retention increased. For neural networks, the relative frequency of retention stayed fairly high as the average predicted probability of retention ranged from 0 to 1. This indicates that it produced poorer performance than random forests.

These results indicate that random forests provide better predictions than neural networks for our data. Additionally, they indicate that S-SMOTE may work best when applied in conjunction with random forests. Table 5.2 gives the mean of the squared differences between the average predicted probabilities and the relative frequency of retention for binned test set predictions. The predicted probabilities were binned using a sequence from 0 to 1 by 0.05. There were 21 bins for each combination of model and method though some were empty in

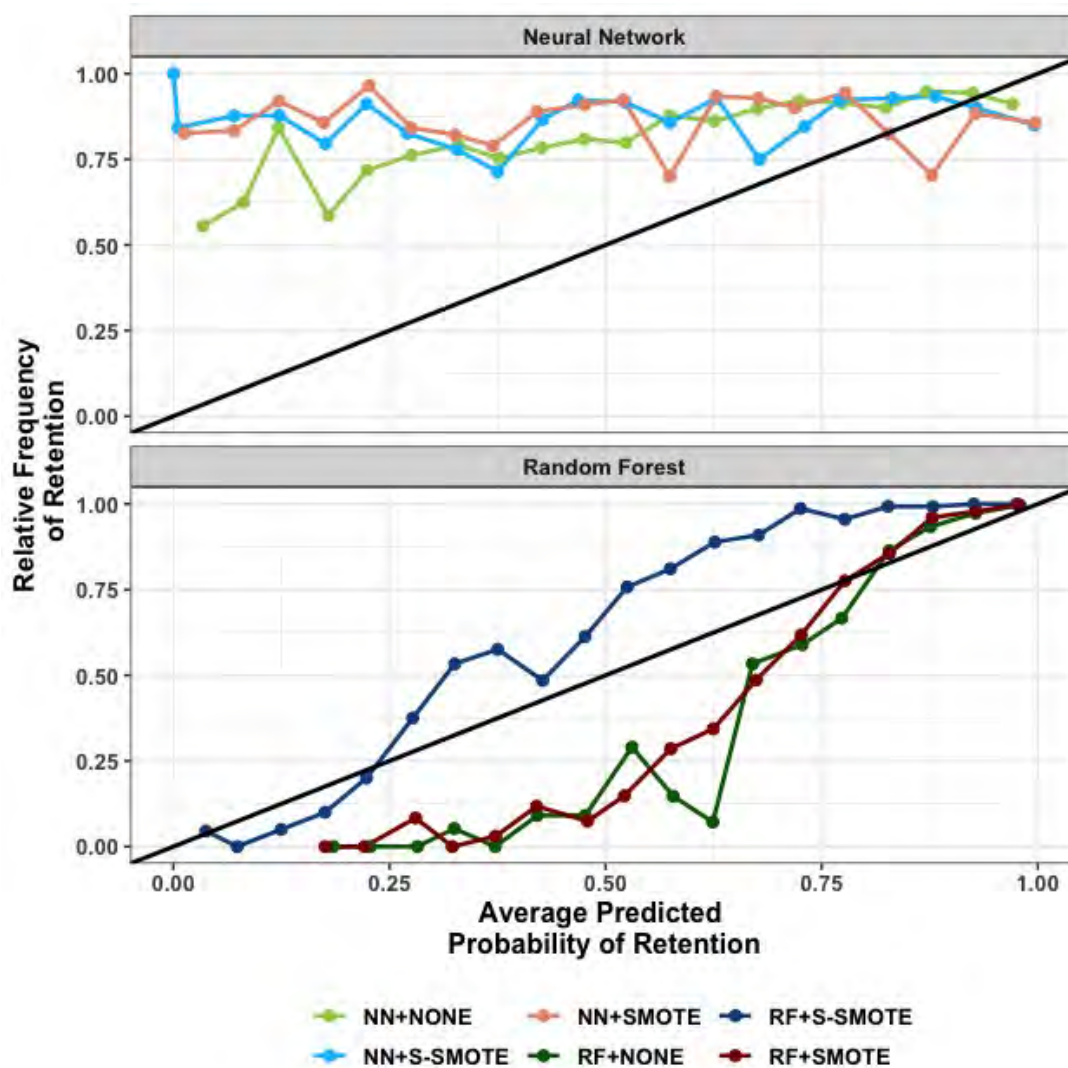


Figure 5.1: Calibration plot for OSU retention dataset. Relationship between average predicted probability and relative frequency of retention from test data. Predicted probabilities were binned using sequence from 0 to 1 by 0.05. The average was taken over each bin with respect to each model+method combination. The relative frequency of retention was also calculated in this manner as well. The black line is $y = x$ which serves as a reference of the ideal model.

some cases. The denominator of the mean indicates the number of bins for the respective model and method combination (i). The average was taken over all non-empty bins with respect to each model+method combination. The relative frequency of retention was also calculated in this manner. The results further confirm that Random Forest + S-SMOTE outperforms other options but that its performance with neural networks was the poorest. When neural networks were applied the unbalanced data produces the best calibration as defined by this average difference.

Model + Method	$\sum_{i=1} (\bar{p}_i - \bar{y}_i)^2 / \text{bin}_i $
Random Forest + SMOTE	0.056
Random Forest + S-SMOTE	0.026
Random Forest + None	0.073
Neural Networks + SMOTE	0.224
Neural Networks + S-SMOTE	0.248
Neural Networks. + None	0.134

Table 5.2: Mean of the squared differences between the average predicted probabilities and the relative frequency of binned test set predictions. Predicted probabilities were binned using a sequence from 0 to 1 by 0.05. There were 21 bins for each combination of model and method though some were empty in some cases. The denominator of the mean indicates the number of bins for the respective model and method combination (i). The average was taken over all non-empty bins with respect to each model+method combination. The relative frequency of retention were also calculated in this manner.

Given that our dataset is also imbalanced in terms of certain covariates (e.g. race, pell-eligibility) it would be sensible to evaluate its performance on these groups. This can help us to understand how well we have achieved the goal of the predictive component of the problem for under-represented students. Figures 5.2 through 5.4 provide the accuracy, sensitivity, and specificity of random forest and neural network models by racial group and Pell-eligibility, respectively. Recall the counts from Table 2.8 and that there were some racial groups that had very low counts. Due to the small number of American Indian or Alaska Native students in our data, there were no students from this racial group in the test set. Additionally, there were other small student groups who had 0% or 100% rates of retention. These data characteristics led to variations in the figures such as missing races or values of 0 or 1 for the performance metrics.

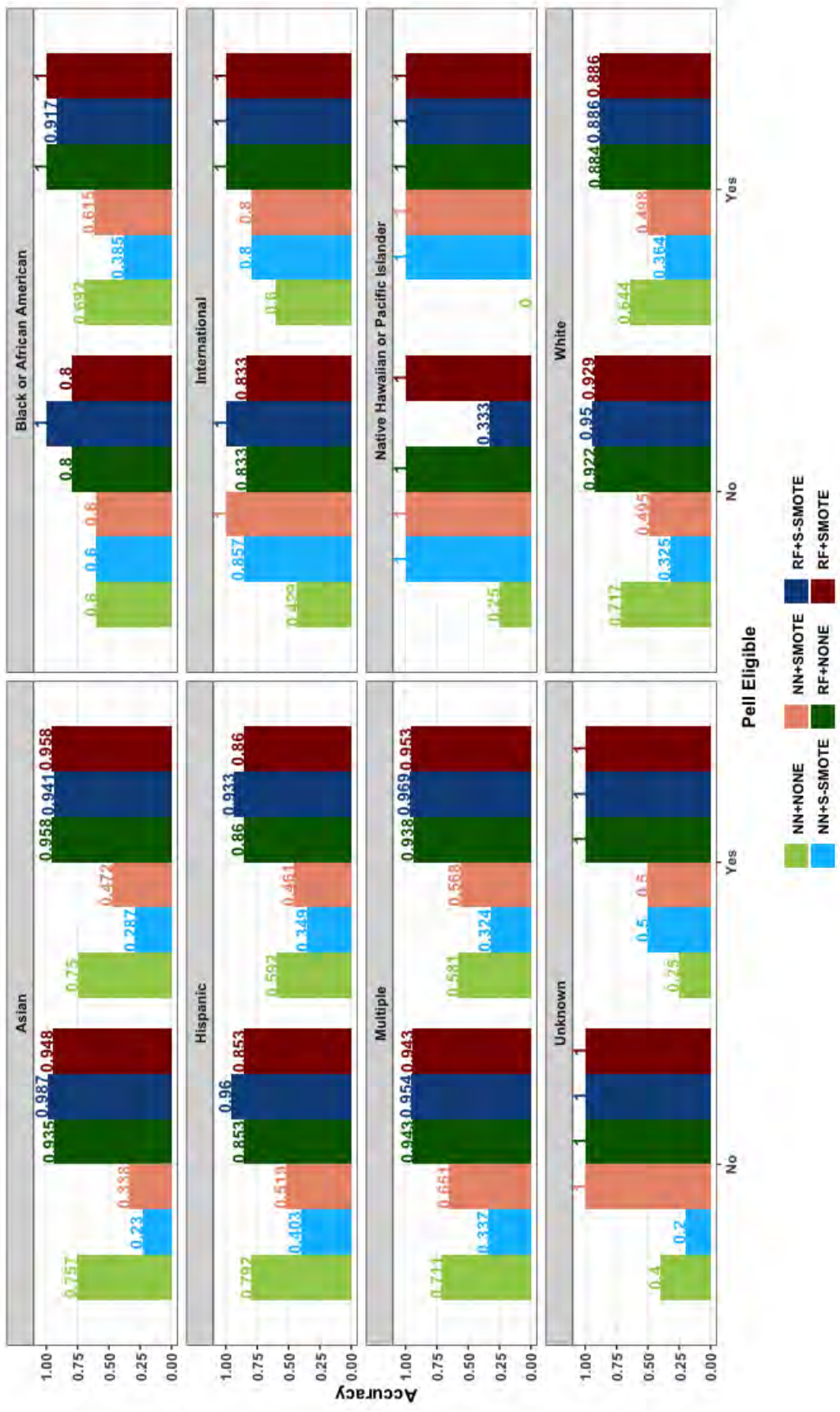


Figure 5.2: Accuracy of test predictions broken down by model and oversampling method, race, and Pell-grant eligibility (an indication of student financial need).

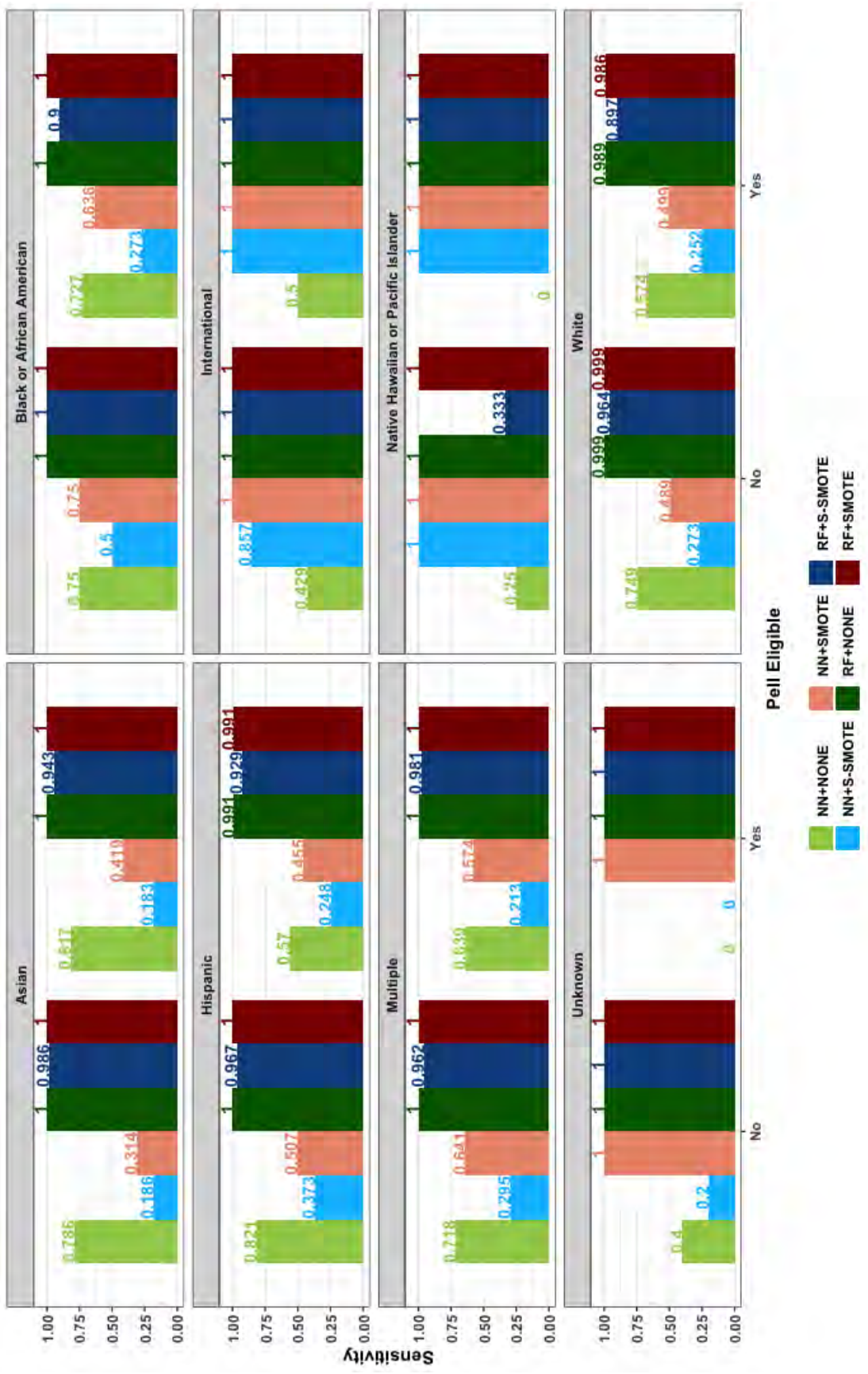


Figure 5.3: Sensitivity of test set predictions broken down by model and oversampling method, race, and Pell-grant eligibility (an indication of student financial need).

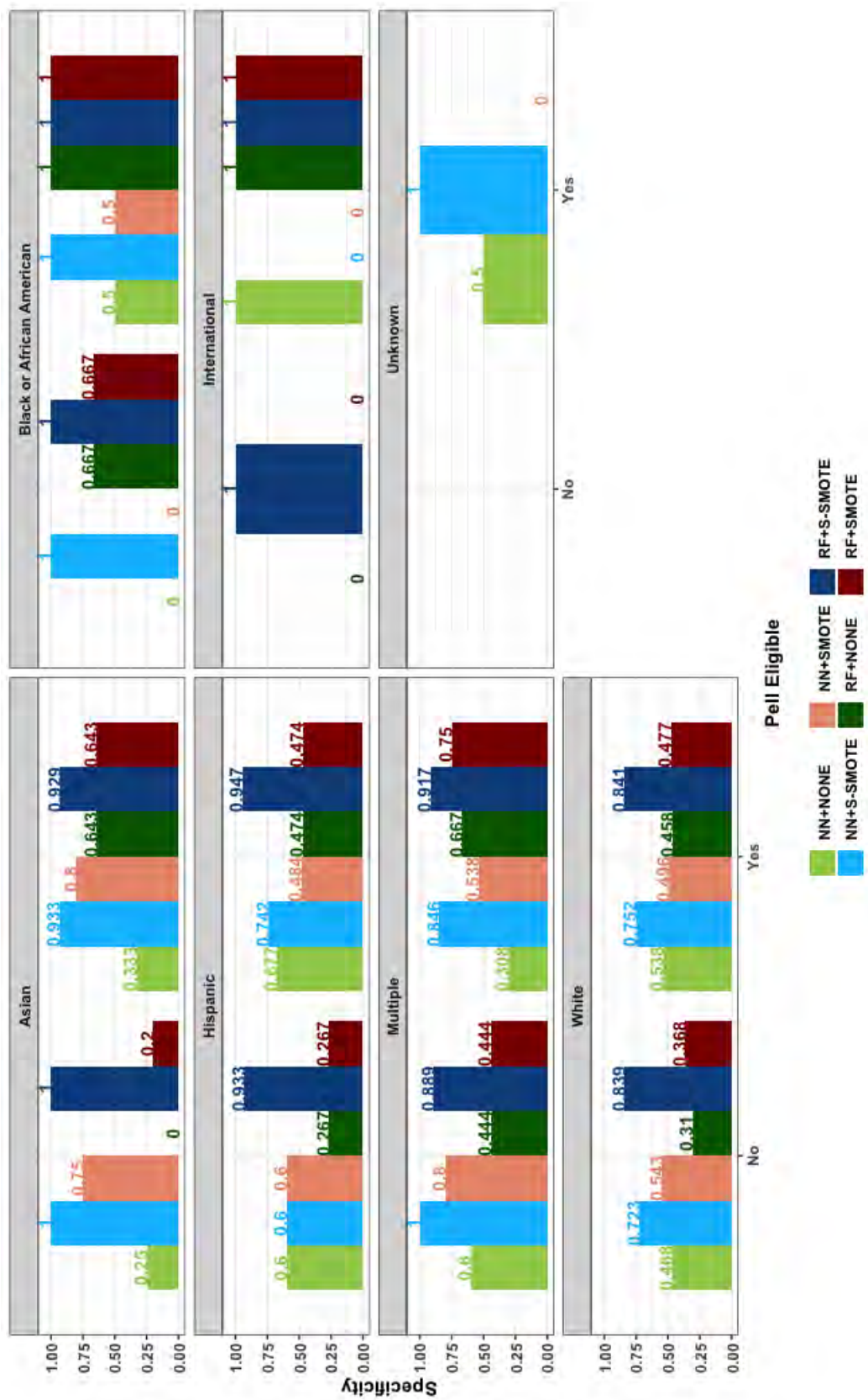


Figure 5.4: Specificity of test predictions broken down by model and oversampling method, race, and Pell-grant eligibility (an indication of student financial need).

For all combinations of Pell-eligibility and racial group, random forests produced higher accuracies than neural networks. This was true in all cases except for Native Hawaiian or Pacific Islander students who were not Pell-eligible. The superior performance of S-SMOTE that was observed when comparing overall accuracies was still mostly present when examining model performance by racial group and Pell-eligibility. However, there were two cases in which random forests trained with S-SMOTE oversampled data had lower accuracy than both SMOTE oversampled and unbalanced data. This occurred for students who were Pell-eligible and also Asian or Black or African-American.

Comparing sensitivity and specificity with respect to student racial group and need reveals that models trained on unbalanced data consistently produced higher sensitivities for all combinations of these covariates. Though neural networks performed more poorly than random forests in aggregate there were some student groups in which the specificity produced by neural networks and S-SMOTE was comparable to that produced by S-SMOTE and random forests and higher than that produced by other methods. This was true for Asian students and Black or African American students regardless of need. This was also observed with regards to the sensitivity but mostly for racial groups with low counts, therefore, the values may be arbitrarily large due to the small sample size.

These results are useful to the discussion of how well our models predict student success for all students. We desire to develop models that can be used to determine how the strategic awarding of gift aid can help students to succeed. Therefore algorithms that are able to predict student success well for all students could lead to more equitable results from this study. Such a characteristic is especially important given the practical component of our research. No model will predict all students perfectly, but this should still be a point of concern in studies such as ours. Next we will discuss the application of S-SMOTE to benchmark imbalanced datasets.

5.2 Application to Benchmark Imbalanced Datasets

The Pima Indians Diabetes dataset provides medical information on a sample of female patients who were at least 21 years old and of Pima Indian heritage. This group of Native Americans living in Mexico and Arizona (e.g. Schulz et al. 2006) were known to have a high rate of diabetes. The data were first used in a 1988 article published by Smith et al. (1988) where the ADAP algorithm was applied to the data. The covariates in the dataset include: the number of times the patient was pregnant, their plasma glucose concentration at 2 hours in an oral glucose tolerance test, diastolic blood pressure (mm Hg), Body Mass Index (BMI, kg/m^2), diabetes pedigree function, the likelihood of diabetes based on family history, age (years), and whether the patient tested positive for diabetes (1) or not (0), the response. A patient is considered to have diabetes if, after taking a liquid containing glucose, their 2-hour blood sugar level is 200 mg/dL or higher.

Haberman's survival dataset contains information on patient survival after breast cancer surgery. The data were introduced by Haberman (1976) in a 1976 article about a study conducted between 1958 and 1970 at the University of Chicago. Covariates in the dataset include the age of the patient at the time of the surgery, the two-digit year in which the surgery happened (years since 1900), and the number of axillary lymph nodes in which cancer was detected. Cancer that has spread from the original tumor site to the axillary lymph nodes is considered to be more advanced. The response variable takes two levels: 1 if the patient survived for five years or longer and 2 if the patient died within five years.

Table 5.3 summarizes relevant information about these three datasets. The Pima Indians diabetes dataset has less imbalance and less overlap in the overall dataset and minority class. The Haberman survival dataset has more imbalance and it has fewer rows. Note that the positive class is the minority class for the Pima Indians dataset but the majority class for Haberman's survival data. In these two cases we would like to predict if a patient has

diabetes and whether a person survived after surgery.

	N	Positive Class	N_{pos}	% Positive	N_{neg}	% Negative
Haberman Cancer	306	Survived	225	73.53	81	26.47
Pima Diabetes	768	Diabetes	268	34.90	500	65.10
	Minority Class Overlap		Majority Class Overlap		All Overlap	
Haberman Cancer	0.70		0.08		0.25	
Pima Diabetes	0.38		0.11		0.20	

Table 5.3: Information on benchmark datasets. Top table: Sample size N, label of positive class, number and percentage in positive class, and number and percentage in negative class. Bottom table: Minority class, majority class, and overall overlap metrics. The overlap metric is the proportion of examples with more than three out of six nearest neighbors in the opposite class. This is calculated within classes and for the entire dataset.

We split these datasets into training and testing sets using a 70/30 split, respectively, stratified by the class of the response variable. Median-mode imputation was used to replace NAs in the training and test datasets with the median and mode for quantitative and nominal variables, respectively. This was also performed with respect to the class of the response variable. Both SMOTE and S-SMOTE were applied to the training dataset and we also performed classification on the unbalanced datasets for reference.

We then performed 5-fold cross-validation to select the best hyperparameters for random forests from the number of trees, minimum node size, and number of variables to try. We also performed 10-fold cross validation to select the best λ for LASSO logistic regression models that maximized the AUC. The variable selection property of LASSO logistic regression may result in a loss of predictive capability, especially since there were only eight columns in the Pima Indians dataset. Due to its popularity for binary classification in practice we include LASSO logistic regression here for completeness. In all cases, the set of hyperparameters that maximized the AUC was selected, the final models were fit, and test predictions were obtained. We performed these analyses for SMOTE and S-SMOTE oversampled data and the unbalanced datasets. The predictive performance results are provided in Table 5.4.

In regards to the Pima Indians diabetes dataset, S-SMOTE outperformed SMOTE with

<i>Pima Indians Diabetes Data</i>						
Model + Oversampling Method	Accuracy	Bal. Accuracy	Sensitivity	Specificity		
Random Forest + SMOTE	0.757		0.740	0.688	0.793	
Random Forest + S-SMOTE	<i>0.791</i>		<i>0.776</i>	<i>0.725</i>	0.827	
Random Forest + None	0.774		0.728	0.575	<i>0.880</i>	
LASSO Logistic Reg. + SMOTE	<i>0.761</i>		<i>0.747</i>	0.700	0.793	
LASSO Logistic Reg. + S-SMOTE	0.757		0.743	0.700	0.787	
LASSO Logistic Reg. + None	0.743		0.687	0.500	<i>0.873</i>	
Model + Oversampling Method	AUC	PPV	NPV	F1 Score	Kappa	McNem. <i>p</i>
Random Forest + SMOTE	0.851	0.640	0.826	0.663	0.473	0.504
Random Forest + S-SMOTE	<i>0.859</i>	0.690	<i>0.849</i>	<i>0.707</i>	<i>0.545</i>	<i>0.665</i>
Random Forest + None	0.844	<i>0.719</i>	0.795	0.639	0.477	0.038
LASSO Logistic Reg. + SMOTE	<i>0.835</i>	0.644	<i>0.832</i>	<i>0.671</i>	<i>0.483</i>	<i>0.418</i>
LASSO Logistic Reg. + S-SMOTE	0.834	0.636	0.831	0.667	0.476	0.350
LASSO Logistic Reg. + None	0.833	<i>0.678</i>	0.766	0.576	0.398	0.009
<i>Haberman's Survival Data</i>						
Model + Oversampling Method	Accuracy	Bal. Accuracy	Sensitivity	Specificity		
Random Forest + SMOTE	0.703		0.585	0.836	0.333	
Random Forest + S-SMOTE	0.714		<i>0.605</i>	0.836	<i>0.375</i>	
Random Forest + None	0.714		0.565	<i>0.881</i>	0.250	
LASSO Logistic Reg. + SMOTE	0.725		0.493	0.985	0.000	
LASSO Logistic Reg. + S-SMOTE	0.736		<i>0.660</i>	0.821	<i>0.500</i>	
LASSO Logistic Reg. + None	0.736		0.500	<i>1.000</i>	0.000	
Model + Oversampling Method	AUC	PPV	NPV	F1 Score	Kappa	McNem. <i>p</i>
Random Forest + SMOTE	<i>0.662</i>	0.778	0.421	0.806	0.181	0.441
Random Forest + S-SMOTE	0.644	<i>0.789</i>	<i>0.450</i>	0.812	<i>0.223</i>	<i>0.556</i>
Random Forest + None	0.624	0.766	0.429	<i>0.819</i>	0.151	0.078
LASSO Logistic Reg. + SMOTE	0.669	0.733	0.000	0.841	-0.022	0.000
LASSO Logistic Reg. + S-SMOTE	0.669	<i>0.821</i>	<i>0.500</i>	0.821	<i>0.321</i>	<i>1.000</i>
LASSO Logistic Reg. + None	0.669	0.736	0/0	<i>0.848</i>	0.000	0.000

Table 5.4: Performance of LASSO logistic regression and random forests on Pima Indians and Haberman's survival datasets. Results after applying SMOTE and S-SMOTE are provided as well as those from unbalanced data. S-SMOTE performs better on the minority class, a positive test for diabetes, at some cost of predictive accuracy on the negative class, negative test for diabetes. Random forests performed better than LASSO logistic regression when S-SMOTE was applied or data were left unbalanced. McNem. *p* refers to the *p*-value for the test of equal discordant pairs, or misclassifications, in both classes. For each metric and dataset the maximum out all combinations of model and oversampling method is italicized when it exists.

respect to all metrics when random forests was applied. Its performance was comparable to SMOTE when LASSO logistic regression was applied for all metrics except the p -value from McNemar's test. In this case, the p -value produced by S-SMOTE was noticeably smaller than that produced by SMOTE. However, it still was not statistically significant at the $\alpha = 0.05$ level and it was much higher than that produced when no oversampling was performed. A key takeaway from the Pima Indians results in Table 5.4 is that S-SMOTE produced better predictions for the minority class at less of a cost to the majority class than SMOTE. The drop in specificity when S-SMOTE and SMOTE were applied was 0.053 and 0.087, respectively. The gain in sensitivity when S-SMOTE and SMOTE were applied was 0.15 and 0.113, respectively. Combining random forests and S-SMOTE also produced the highest p -value from McNemar's test of equal discordant pairs in the confusion matrix. A p -value much larger than 0.05 indicates that its predictions provide no evidence of misclassifying one class more than another. The test only rejected when data were not oversampled at all.

The predictive performance results of Haberman's survival data included zero values for the specificity and NPV when LASSO logistic regression was applied to SMOTE oversampled or unbalanced data. This can occur when there are no correct predictions for the negative class or there are no predictions at all, either correct or incorrect. In the latter case the NPV will be 0/0 which is undefined. When S-SMOTE was applied with LASSO logistic regression the specificity and NPV were the highest at 0.50. However, the sensitivity was lower than that for SMOTE or unbalanced data with LASSO logistic regression. When LASSO logistic regression was applied S-SMOTE performed better than SMOTE and unbalanced data with regards to the balanced accuracy, specificity, PPV, NPV, Kappa coefficient, and McNemar's p -value. This was also true when random forests were applied. Similar to the Pima Indians dataset, these results indicate that S-SMOTE is able to obtain more correct predictions for the minority class but at a cost. In this case, the amount that the sensitivity drops is the same for both SMOTE and S-SMOTE. This was $0.881 - 0.836 = 0.045$. However, the specificity increases from 0.250 to 0.375 with S-SMOTE and from 0.25 to 0.333 with SMOTE.

This indicates that applying S-SMOTE effectively caused more correct classifications on the minority class with the same loss on the majority class.

Given the context of these dataset, consideration should be given to the practical severity of the trade-offs made by oversampling. For the Pima Indians diabetes data, the sensitivity corresponds to the proportion of patients with diabetes that we correctly classified as such. Fewer patients tested positive for diabetes however so the TPR is lower than the TNR since there is less information on positive examples for the model to learn. Increasing the TPR would lead to the correct identification of diabetes more frequently. However, if the TNR correspondingly decreases this could lead to false positives which could lead to the unnecessary implementation of procedures for managing diabetes and the misallocation of resources. In this case however, a false negative is more consequential than a false positive.

Concerning Haberman's survival data, more patients survived five years after surgery than those that did not and this is the positive class. Though a higher TPR can be expected our ability to determine when a patient will not survive correctly is very important. Otherwise a surgery may be performed on a patient who is unlikely to survive afterwards. In this case, the trade-off is well worth it and we would rather incorrectly use an alternative treatment to the surgery than incorrectly perform the surgery. Depending on the efficacy of the alternative treatment one could argue that the cost of a false positive and false negative are equally consequential.

In practice when performing oversampling one should always be prepared for the cost of misclassifying the majority class more often than if oversampling was not performed. Consideration should be given to the implications of this and domain knowledge should be used to assess whether it is worth it. Thus far we have often oversampled in a manner that achieves complete balance 100% of the difference between class counts but one may consider oversampling at a rate of 50% or even 25% of the difference between class counts in order to control the trade-off between the TPR and the TNR.

Figures 5.5 and 5.6 contain the calibration plots for the final models applied to the Pima Indians and Haberman’s survival data test sets. Table 5.5 also contains the mean of the squared differences between the average predicted probabilities and the relative frequency of a positive. These were also obtained using the test set. The calibration plots show the average predicted probability and relative frequency of a positive in the test data. The predicted probabilities were placed into bins using a sequence of values from 0 to 1 by 0.05 with the right end of the interval closed. The average probability of a positive was calculated for each bin with respect to each model and method combination. The relative frequency of a positive was also calculated for these subsets of results. The black line indicates the line $y = x$ which gives the results we would expect from a perfect model.

<i>Pima Indians Diabetes Data</i>	
Model + Method	$\sum_{i=1} (\bar{p}_i - \bar{y}_i)^2 / \text{bin}_i $
Random Forest + SMOTE	0.0148
Random Forest + S-SMOTE	0.0275
Random Forest + None	0.0197
LASSO Logistic Reg. + SMOTE	0.0310
LASSO Logistic Reg. + S-SMOTE	0.0510
LASSO Logistic Reg. + None	0.0374
<i>Haberman’s Survival Data</i>	
Model + Method	$\sum_{i=1} (\bar{p}_i - \bar{y}_i)^2 / \text{bin}_i $
Random Forest + SMOTE	0.0838
Random Forest + S-SMOTE	0.0909
Random Forest + None	0.0990
LASSO Logistic Reg. + SMOTE	0.1118
LASSO Logistic Reg. + S-SMOTE	0.1747
LASSO Logistic Reg. + None	0.2059

Table 5.5: Mean of the squared differences between the average predicted probabilities and the relative frequency of binned test set predictions. Predicted probabilities were binned using a sequence from 0 to 1 by 0.05. There were 21 bins for each combination of model and method though some were empty in some cases. The denominator of the mean indicates the number of bins for the respective model and method combination (i) The average was taken over all non-empty bins with respect to each model+method combination. The relative frequency of the positive classes (diabetes and survival) were also calculated in this manner.

Regarding the calibration of models for the Pima Indians diabetes dataset, the calibration plot shows that there is clear variability in their predictive capabilities. The calibration lines

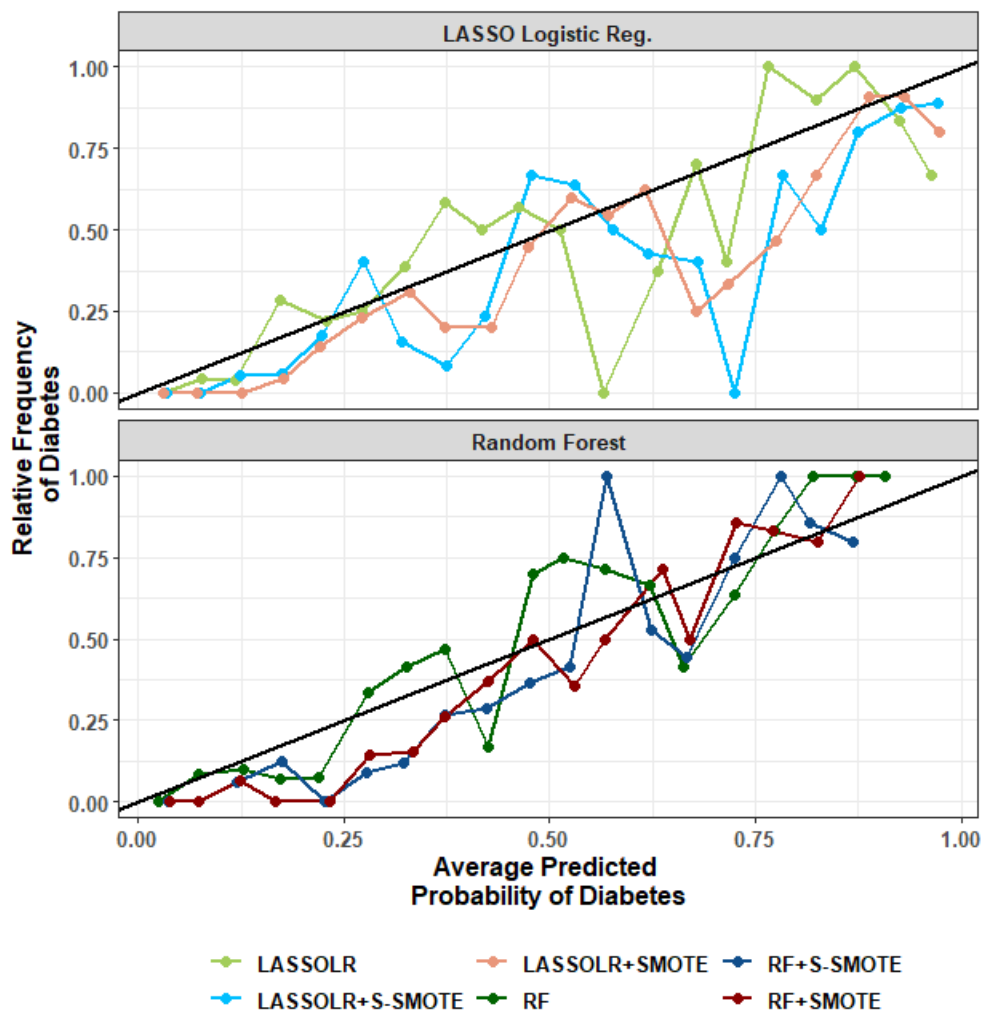


Figure 5.5: Calibration plot for Pima Indians diabetes dataset. Relationship between average predicted probability and relative frequency of diabetes from test data. Predicted probabilities were binned using sequence from 0 to 1 by 0.05. The average was taken over each bin with respect to each model+method combination. The relative frequency of diabetes was also calculated in this manner. The black line is $y = x$ which serves as a reference of the ideal model.

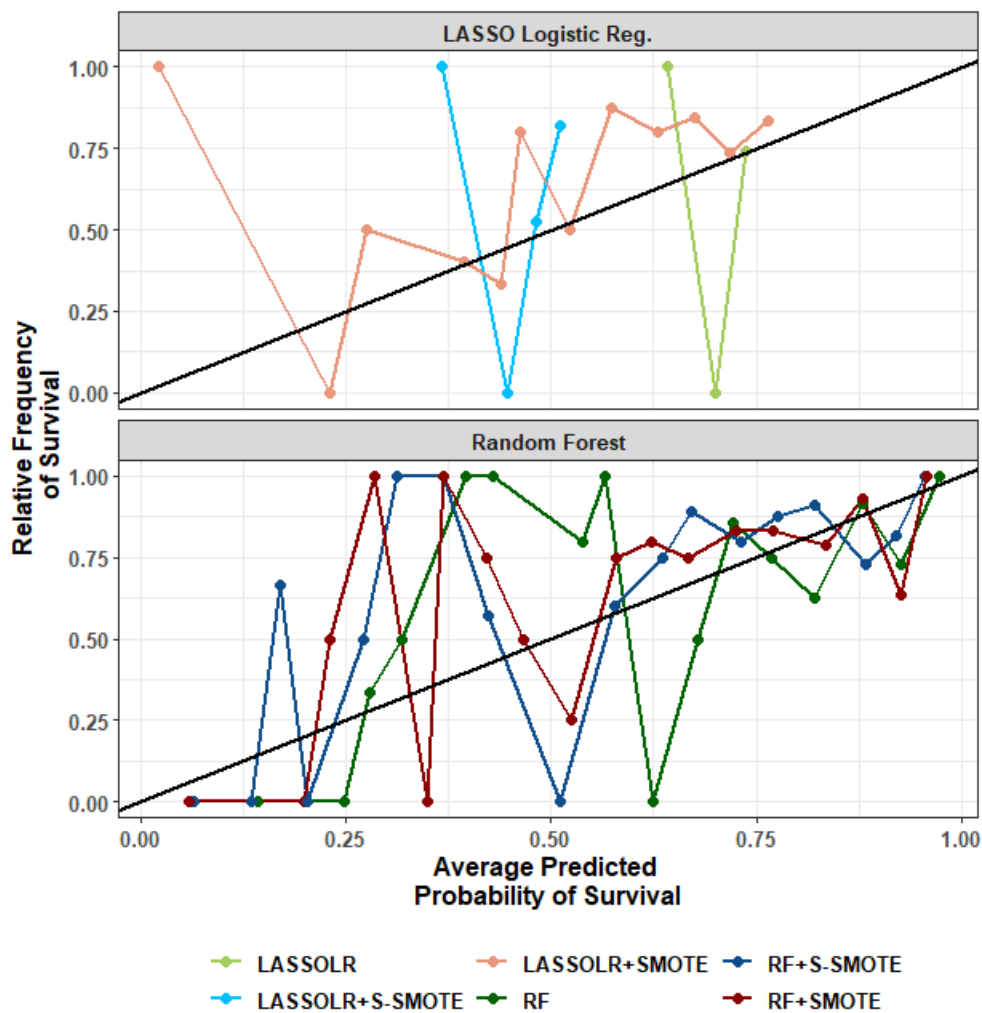


Figure 5.6: Calibration plot for Haberman's survival dataset. Relationship between average predicted probability and relative frequency of survival from test data. Predicted probabilities were binned using sequence from 0 to 1 by 0.05. The average was taken over each bin with respect to each model+method combination. The relative frequency of survival was also calculated in this manner. The black line is $y = x$ which serves as a reference of the ideal model.

for predictions from LASSO logistic regression models extend all the way to 0.973 while those for random forests extend to 0.907. This indicates that logistic regression was able to predict higher probabilities than random forests. These high average predicted probabilities were also quite close to the reference line, especially when the data were oversampled with S-SMOTE.

However, the sum of the squared differences between the average predicted probabilities and the relative frequency were largest when S-SMOTE was applied. This can be noted in Table 5.5. They were smallest when SMOTE was applied and the unbalanced data produced the second highest differences. The differences produced by S-SMOTE were noticeably larger than those produced by SMOTE, especially when logistic regression was applied. This indicates that logistic regression models trained using data oversampled with SMOTE may be better calibrated than those trained with S-SMOTE.

The calibration plot for Haberman's survival data shows that the predicted probabilities from LASSO logistic regression models trained on S-SMOTE oversampled and unbalanced data do not vary much. The probabilities produced by random forests became more accurate with respect to the black reference line as the relative frequency of survival increased. As with the Pima Indians dataset, S-SMOTE produced higher differences between the average predicted probability of survival and the relative frequency of survival. However, this was lower than those produced by the predictions obtained from models trained on unbalanced data.

These real-world examples provide evidence that S-SMOTE is also useful in applications outside of institutional research. Furthermore, they show that there are instances where S-SMOTE has superior performance to the original SMOTE. When applied in combination with random forests, S-SMOTE had very good performance and was often superior to the performance of SMOTE. Regardless of how oversampling is performed a drop in accuracy on the majority class will occur. The goal of S-SMOTE is not to completely eliminate this

trade-off but to achieve a better trade-off than that exhibited by the original SMOTE. Our results, especially those obtained when random forests were applied show that this goal has been achieved.

Chapter 6: Conclusions

In this part of the dissertation we first considered the problem of assessing the effect of scholarships on student success at Oregon State University (OSU). The population represented by the data in our case study was first-time full-time (FTFT) freshmen at OSU. These are students who have enrolled full-time for the first time and have completed few enough credit hours to give them freshmen standing. We defined student success in terms of graduation and retention rates and determined whether and how these rates differ as student demographics and academic backgrounds change.

In order to tackle the inferential component of our research, we first used logistic regression models to discover the relationship between retention and graduation and the amount of gift aid received by students. One of the advantages of this approach is the flexibility and interpretability of these models. This characteristic allows us to directly estimate the probability of retention or graduation for a given amount of financial aid. It also enables us to determine amounts of financial aid that will lead to the preset probabilities of retention or graduation desired by administrative offices.

On the other hand, the flexibility of the models allow us to incorporate relevant demographic information to quantify the changes in the response curves observed for different groups. As expected, the models showed a positive relationship between financial aid and student success. Larger amounts of financial aid paired with improved chances of retention and graduation. In terms of demographics, we found that Pell-eligibility, first-generation status and financial need were important factors that exhibited gaps in achievement, even after accounting for academic performance.

We also developed models that can be combined with more recently collected data for the predictions of first-year retention. Our initial attempts at predictive modeling revealed two difficulties in our dataset that were preventing our models from achieving a high balanced accuracy. These were the issue of imbalance in the response variable and overlap in the distributions of features between classes of the response variables. In order to accomplish the predictive component of our research we, therefore, deemed it necessary to better understand the issue of imbalance and overlap and develop a new solution. Therefore, we performed a simulation study to characterize the performance of the most popular method for handling class imbalance, the Synthetic Minority Oversampling TEchnique (SMOTE), as data characteristics changed. This algorithm uses interpolation to generate synthetic examples in the feature space but it does not handle overlap in the feature space. This effectively creates more decision regions that belong to the minority class while possibly biasing majority class regions, thereby bringing down the accuracy on the majority class.

These simulations brought to light issues with SMOTE that exists when there is both imbalance in the response variable and overlap in the feature space. We found that the impact that oversampling has on predictive performance also depends on the model that the oversampled data are fit to. Moreover we found that the distribution used for w during the interpolation of SMOTE leads to less variability in performance as compared to those produced by changes in the underlying characteristics of the dataset.

Using the results of our simulation study we developed a novel algorithm called the Strategic SMOTE (S-SMOTE) that tackles the issue of both overlap and imbalance. S-SMOTE is strategic in that it uses information on the neighborhood of minority points to determine where to oversample and how much to oversample. This effectively leads to the oversampling of minority class regions that are well dominated by the minority class already. Additionally, the amount of oversampling that is performed with a given minority example is determined by how densely populated the example already is. We then revisited the data from our case

study to determine if we could better predict student success using S-SMOTE. The results showed that our algorithm gave superior performance when combined with random forests but more work is needed to refine its performance when combined with neural networks. Specifically we found that S-SMOTE was able to achieve a better trade-off between the true negative rate (TNR) and true positive rate (TPR) after oversampling than that produced by oversampling with SMOTE. The accuracy on the minority class increased while the accuracy on the majority class decreased less than when SMOTE was applied. We saw similar results when we applied S-SMOTE to benchmark imbalanced datasets indicating that our algorithm produces improvements but may need further refinement before being applied with neural networks.

Future work will involve performing the simulation study with a higher number of repetitions and making corrections to the simulations performed with neural networks. We ran into many challenges when fitting neural networks due to the use a package that uses a of a Python back-end. This future work will therefore include fitting these models in a different programming language in order to obtain results more quickly. These improvements will lead to more accurate insight that reflects the true performance of these algorithms and models. We will conduct this work very soon as it is necessary for the completion and publication of our follow-up manuscript.

In the future we also aim to understand how and why S-SMOTE gave better performance when combined with random forests than with neural networks when we revisited the case study data. We face the issues present with neural networks in the simulation study when revisiting the OSU dataset since it was a single-use case. Therefore, it is not clear whether this performance is due to some technical error in our analysis or if the method needs further refinement. These are areas of further research.

Lastly, in the more distant future we desire to study the usefulness of an adaptation to S-SMOTE that may be even more strategic. In this adaptation the number of neighbors

considered during the iterative check for minority class dominated neighborhoods of the feature space would decrease along with the threshold. This may lead to further improvements in predictive performance and even less of a trade-off between the TPR and TNR. The author plans to conduct this follow-up work further into the future as a new faculty member after the current research is in publication.

Part II

A Simulation-Based Approach to Teaching the Bootstrap

Chapter 7: Introduction

Bootstrapping is a computer-based method introduced by Efron (1979) as a technique for estimating the standard deviation of a sample statistic. In its simplest form, the term *bootstrap sampling* refers to the process of randomly sampling with replacement from the original sample. This process is taken to be analogous to sampling from the entire population and, as noted by Efron and Tibshirani (1993), the bootstrap estimate of standard error is always available regardless of the complexity of the original estimator.

Since its introduction, bootstrap methods have gained popularity (see Horowitz 2019; Utzet and Sánchez 2021) and found use in a variety of diverse applications such as linear regression (see Eck 2018; Pelawa Watagoda and Olive 2021) and bootstrap aggregated neural networks (see Khaouane et al. 2017; Osulale and J. Zhang 2018). The growth of statistical computing has also led to the bootstrap appearing more regularly in courses which introduce undergraduate students to statistical methods with examples including courses taught at Stanford University¹, The Pennsylvania State University², Oregon State University³, and Montana State University⁴. Textbooks about, or which feature, the bootstrapping method thus range from the seminal graduate-level text by Efron and Tibshirani (1993) to intro-level texts (e.g. Field et al. 2012; Ismay and Kim 2019; R. H. Lock, P. F. Lock, et al. 2020).

Teaching the bootstrap can equip students with a very powerful tool and lay a solid founda-

¹STAT 191 - Introduction to Applied Statistics at Stanford University (<https://explorecourses.stanford.edu>)

²STAT 200 - Elementary Statistics at The Pennsylvania State University (<https://online.stat.psu.edu/stat200/>)

³STAT 351/352 - Introduction to Statistical Methods I & II (<https://stat.oregonstate.edu/content/yearly-courses>)

⁴STAT 216 - Introduction to Statistics at Montana State University (<https://math.montana.edu/courses/s216/>)

tion for teaching statistical thinking. Therefore the discussion of how to teach the bootstrap well is an important one to have. Pedagogical discussions about whether bootstrap methods should be taught, which bootstrap methods to teach, and how to teach them include Hesterberg (2015b) and Hayden (2019).

In this chapter, we will highlight the assumptions behind simple bootstrap hypothesis tests and confidence intervals. First we discuss the benefits of and current issues pertaining to teaching statistical computing and the bootstrap, as found in the literature on statistics education. Then we discuss the theoretical details of the bootstrap in order to clearly point out the assumptions behind these methods. By assumptions, we mean the suppositions under which the theoretical details of these intervals are derived. Namely, those having to do with pivotal quantities. Our focus is on the studentized, basic, and percentile bootstrap intervals and their corresponding hypothesis tests. We choose this focus because these methods, or methods related to them, are often taught in undergraduate introductory statistics courses.

7.1 Benefits of Teaching Statistical Computing and the Bootstrap

According to the *Guidelines for Assessment in Statistics Education* (GAISE), students in introductory statistics courses should, “Demonstrate an understanding of, and ability to use, basic ideas of statistical inference, both hypothesis tests and interval estimation, in a variety of settings.” (GAISE College Report ASA Revision Committee 2016, p. 8) This implies that students should be able to recognize when a particular statistical method detracts from the quality of their analysis. Furthermore, upon realizing this, they should be able to pull an alternative method from their knowledge-base and apply it appropriately.

Depending on learning goals and student backgrounds, including topics that incorporate statistical computing in a course, such as resampling, randomization, or simulation, can help students achieve these objectives. For example, Wood (2005) notes that through general

simulation methods students are able to “actively and intelligently” apply the methods they are taught to solve problems of current concern. Also, Tintle et al. (2012) found that the use of a randomization-based curriculum led to a higher retention of concepts, after four months, than using the consensus curriculum based on Agresti and Franklin (2007). Simulation methods are also incorporated by Son et al. (2021) in their discussed “practicing connections” approach to building an introductory statistics course. Their approach was found to make students capable of applying previously learned material in new and more sophisticated contexts.

It was noted by R. H. Lock and P. F. Lock (2008) that students’ understanding of confidence intervals and statistical inference rests greatly on their understanding of sampling distributions. They suggested that teaching bootstrapping allows students to make inferences on non-conventional parameters and that their understanding of concepts like sampling distributions can be fortified through the use of simulations and bootstrapping. Indeed, when the form of the standard error of an estimator cannot be derived using statistical theory or it depends on unknown parameters and/or the true distribution of the estimator is completely unknown, the bootstrap can be useful, provided that its own assumptions are met.

Howington (2017) notes that that, though its desirability as a measure of center when the data are skewed is often mentioned, corresponding inferential methods for the median are rarely taught in introductory statistics courses. Suggested methods for teaching confidence intervals on the median included use of the bootstrap. The use of simulation-based inferential methods was also discussed by Gehrke et al. (2021), where the incorporation of methods, such as the bootstrap, in their improved curriculum, helped students to more clearly explain p -values and confidence intervals and to understand the limitations of statistics as it pertains to describing the real world.

The idea that statistical computing should be taught more, in order to better equip students for present-day workforce expectations, undercurrents much of the literature on statistics in

the undergraduate curriculum and, in general, statistics education. Besides the aforementioned literature, various articles in the collection compiled by Horton and Hardin (2015) express this idea. Technically, the more statistical methods a student is introduced to, the better equipped they should be to meet the GAISE guideline discussed earlier and to tackle real-world data challenges. In reality though, as students learn more statistical methods, discerning which one is appropriate to use becomes harder. Especially if students are not clearly taught how to check whether a method is appropriate for their data.

For example, many incorrect or unfounded claims have been made about simple bootstrap intervals, making it hard to know when their use is appropriate. These claims were investigated in greater detail by Hayden (2019) for the percentile bootstrap interval that we define in the next chapter and the bootstrap interval that uses twice the standard deviation of the bootstrap distribution as the margin of error. In their article, the claim that these bootstrap intervals have fewer or no underlying assumptions than their traditional counterparts was debunked and shown to clearly be false.

It was also found that these bootstrap intervals do not actually perform better when normality and large sample size conditions are not met. Their supposed simplicity was said to be the result of a failure to communicate their assumptions as clearly as those of the traditional methods. Introducing these intervals to students, before appropriate scenarios for their use are better established, was discouraged. In Hesterberg (2015b), issues with the percentile and basic bootstrap intervals were also discussed and use of the studentized bootstrap interval (called the bootstrap t interval there) was said to be preferable.

Given the pedagogical and methodological benefits of the bootstrap, students and instructors need to understand the assumptions behind these methods. This can lead to students learning and applying them more carefully. We will now discuss some of the theoretical underpinnings of the basic, studentized, and percentile bootstrap intervals, as well as their corresponding bootstrap hypothesis tests. Specifically, our intention is to show that these

methods rely on assumptions concerning pivotal quantities. For a more rigorous and expansive discussion on the theory behind the bootstrap, we refer readers to Athreya and Lahiri (2006).

Chapter 8: General Assumptions for Simple Applications of the Bootstrap

In order to make an inference on a population parameter, θ , we begin by taking an independent and identically distributed sample of size N , $\mathbf{x} = (x_1, x_2, \dots, x_N)$, from the population of interest. This sample should be taken in such a way that it captures most of the information in the population about θ . We denote an estimate for θ based on the observed data as $\hat{\theta}(\mathbf{x})$. If this is calculated with a bootstrap sample we use $\hat{\theta}(\mathbf{x}^*)$. If it is based on the not yet observed data we use $\hat{\theta}(X)$, where X denotes the unobserved data vector. The estimate, $\hat{\theta}(\mathbf{x})$, should summarize the information about θ that is contained in the observed data. For example, if θ is the population mean, then $\hat{\theta}(\mathbf{x})$ may be the observed sample mean, \bar{x} .

However, we often desire to gather more information than that contained in $\hat{\theta}(X)$ alone. Options for achieving this include confidence intervals and hypothesis testing. Many methods exist for constructing confidence intervals and hypothesis testing, such as z - and t -methods for the mean and jackknife or permutation approaches. When the parameter of interest is one which does not have an established method or the data do not meet the conditions for using traditional methods, alternative methods can be used. These alternative methods will likely have their own assumptions and these should also be checked. If they are reasonable, then the alternative method can be used. One such alternative is the bootstrap, whose details and assumptions we discuss in this section. Specifically, we highlight the dependence of these intervals on pivotal quantities.

The concept of bootstrapping through simple random resampling is as follows: Obtain B samples, $\mathbf{x}_1^*, \mathbf{x}_2^*, \dots, \mathbf{x}_B^*$, each of size N , by resampling from the original sample, \mathbf{x} , with replacement and calculate their corresponding statistics, $\hat{\theta}(\mathbf{x}_1^*), \hat{\theta}(\mathbf{x}_2^*), \dots, \hat{\theta}(\mathbf{x}_B^*)$. These bootstrap statistics make up the bootstrap distribution. Though the underlying concepts of this

bootstrapping method may seem straightforward, there are many details that users should be aware of when applying it for interval estimation and hypothesis testing.

8.1 Interval Estimation

The bootstrap distribution can be used as an estimate of the sampling distribution, which provides a means for quantifying the uncertainty in an estimate. The basic, percentile, and studentized bootstrap intervals each use the bootstrap distribution in this manner but have different underlying assumptions, most of which pertain to the shifted or studentized sampling distribution. The details we discuss next will be helpful for readers who desire to become familiar with these bootstrap intervals as they are presented by Davison and Hinkley (1997) and Efron and Tibshirani (1993). Our explanation is not exhaustive, however, so readers who desire a more in-depth understanding of these methods and their assumptions should consult those texts directly. Those who are already familiar with these methods, can skip to the summary of their form and assumptions given in Table 8.1.

Let $0 < \alpha < 1$ denote the significance level or desired Type I error rate. In order to construct a $(1 - \alpha)100\%$ confidence interval for θ , we may employ a *pivotal quantity* - a quantity whose distribution does not depend on any unknown parameters. When this quantity is a function of the parameter and estimate, the quantiles of its distribution can be used to construct confidence intervals for the parameter.

Denote the $\alpha/2$ and $1 - \alpha/2$ quantiles of the distribution of $\hat{\theta}(X) - \theta$ as $a_{\alpha/2}$ and $a_{1-\alpha/2}$, and suppose that this quantity is pivotal. In general, when we refer to the p quantile of the distribution of $\hat{\theta}(X)$ (or its shifted or scaled versions), we are referring to the value, a , for which $P(\hat{\theta}(X) \leq a) = p$. If $a_{\alpha/2}$ and $a_{1-\alpha/2}$ are known, then

$$1 - \alpha = P(a_{\alpha/2} \leq \hat{\theta}(X) - \theta \leq a_{1-\alpha/2}) = P(\hat{\theta}(X) - a_{1-\alpha/2} \leq \theta \leq \hat{\theta}(X) - a_{\alpha/2})$$

and a $(1 - \alpha)100\%$ equi-tailed interval for θ , provided the expression exists, is

$$\left(\hat{\theta}(\mathbf{x}) - a_{1-\alpha/2}, \hat{\theta}(\mathbf{x}) - a_{\alpha/2} \right). \quad (8.1)$$

If the distribution of $\hat{\theta}(\mathbf{X}) - \theta$ is unknown, the problem becomes one of estimating $a_{1-\alpha/2}$ and $a_{\alpha/2}$. Using the bootstrap distribution, one may estimate these quantiles in a variety of ways. For convenience, we discuss estimation in terms of the p quantile, a_p .

8.1.1 The Basic Interval (The Base Case)

The basic bootstrap interval is obtained by estimating a_p with the $(B + 1)p$ -th smallest value of the distribution of $\hat{\theta}(\mathbf{x}^*) - \hat{\theta}(\mathbf{x})$. For example, if $\alpha = 0.05$ and $B = 999$ then

$$(B + 1)(\alpha/2) = (999 + 1)0.025 = 1000 * 0.025 = 25$$

and similarly, $(B + 1)(1 - \alpha/2) = 975$. Thus, the 25th smallest and 975th smallest values of the distribution of $\hat{\theta}(\mathbf{x}^*) - \hat{\theta}(\mathbf{x})$, denoted as $a_{(25)}^*$ and $a_{(975)}^*$, would be used to estimate $a_{0.025}$ and $a_{0.975}$, respectively. Note that the 25th smallest value is less than the 975th smallest value so the upper bound will be greater than the lower bound since we subtract off a smaller number¹.

Using this estimate, the expression in (8.1) becomes

$$\left(\hat{\theta}(\mathbf{x}) - a_{((B+1)(1-\alpha/2))}^*, \hat{\theta}(\mathbf{x}) - a_{((B+1)(\alpha/2))}^* \right).$$

¹We will assume that $(B + 1)\alpha$ and $(B + 1)(1 - \alpha)$ are integers for our purposes. If they are not, then the procedure outlined by Efron and Tibshirani (1993) can be used, assuming $\alpha \leq 0.5$. Define k as the largest integer that is $\leq (B + 1)\alpha$. Then the α and $1 - \alpha$ quantiles are defined as the k -th largest and $(B + 1 - k)$ -th largest values of the distribution of interest, respectively.

Let r_p and r_p^* be the p quantiles of the distributions of $\hat{\theta}(X)$ and $\hat{\theta}(x^*)$, respectively, then note that

$$a_{((B+1)p)} = r_{((B+1)p)} - \theta \quad \text{and} \quad a_{((B+1)p)}^* = r_{((B+1)p)}^* - \hat{\theta}(x).$$

Therefore, the bounds of the interval can be further simplified to

$$\hat{\theta}(x) - a_{((B+1)(1-\alpha/2))}^* = 2\hat{\theta}(x) - r_{((B+1)(1-\alpha/2))}^* \quad \text{and} \quad \hat{\theta}(x) - a_{((B+1)(\alpha/2))}^* = 2\hat{\theta}(x) - r_{((B+1)(\alpha/2))}^*.$$

The final form of the basic bootstrap interval is then

$$\left(2\hat{\theta}(x) - r_{((B+1)(1-\alpha/2))}^*, 2\hat{\theta}(x) - r_{((B+1)(\alpha/2))}^* \right).$$

If there are any constraints on the value of θ , the bounds of this interval may not meet these constraints. That is, this interval can contain values that are not plausible for the population parameter, such as values below 0 or above 1 in an interval for the population proportion. The accuracy of this interval depends on how well the distribution of $\hat{\theta}(x^*) - \hat{\theta}(x)$ conforms to that of $\hat{\theta}(X) - \theta$. If the latter does not depend on any unknown parameters, then $\hat{\theta}(X) - \theta$ is actually a pivotal quantity and conformity can be expected.

8.1.2 The Percentile Interval (The Symmetric Case)

Suppose that the distribution of $\hat{\theta}(X)$ is asymptotically Normal with mean θ and variance $\text{SE}(\hat{\theta}(X))^2$, where $\text{SE}(\hat{\theta}(X))$ denotes the standard error of $\hat{\theta}(X)$. This provides another option for estimating a_p . Namely, with $\hat{\text{SE}}(\hat{\theta}(x)) * z_p$, where z_p is the p quantile of the standard normal distribution. For example, in the case of the sample mean we may use $\hat{\text{SE}}(\hat{\theta}(x)) = s/\sqrt{n}$ where s denotes the sample standard deviation. The use of standard normal quantiles with an estimated standard error produces what is termed the *standard normal interval* (e.g.

Efron and Tibshirani 1993). This is given as

$$\left(\hat{\theta} - \widehat{\text{SE}}(\hat{\theta}(\mathbf{x})) * z_{1-\alpha/2}, \hat{\theta} - \widehat{\text{SE}}(\hat{\theta}(\mathbf{x})) * z_{\alpha/2} \right).$$

When the assumptions of this interval are not met, it can lead to poor performance. An alternative is the *percentile interval* introduced by Efron and Tibshirani (1993) which is given as

$$(r_{\alpha/2}^*, r_{1-\alpha/2}^*),$$

where we again denote the p quantile of the distribution of $\hat{\theta}(\mathbf{x}^*)$ as r_p^* .

The assumption behind the percentile interval is that there exists some monotone transformation $\hat{\phi} = m(\hat{\theta}(X))$ such that $\hat{\phi} \sim \text{Normal}(\phi, c^2)$ for all population distributions F (including the case $F = \hat{F}$), where $\phi = m(\theta)$, for some standard deviation c . Then, it holds that

$$1 - \alpha = P\left(z_{\alpha/2} \leq \frac{\hat{\phi} - \phi}{c} \leq z_{1-\alpha/2}\right) = P(-\hat{\phi} + z_{\alpha/2} \cdot c \leq -\phi \leq -\hat{\phi} + z_{1-\alpha/2} \cdot c) =$$

$$P(m^{-1}(\hat{\phi} - z_{1-\alpha/2} \cdot c) \leq \theta \leq m^{-1}(\hat{\phi} - z_{\alpha/2} \cdot c)).$$

Since the assumption holds for $F = \hat{F}$ it is also the case that $\hat{\phi}^* \sim \text{Normal}(\hat{\phi}, c^2)$, where $\hat{\phi}^* = m(\hat{\theta}(\mathbf{x}^*))$. Therefore,

$$1 - \alpha = P_*\left(z_{\alpha/2} \leq \frac{\hat{\phi}^* - \hat{\phi}}{c} \leq z_{1-\alpha/2}\right) = P_*(\hat{\phi} + z_{\alpha/2} \cdot c \leq \hat{\phi}^* \leq \hat{\phi} + z_{1-\alpha/2} \cdot c) =$$

$$P_*(m^{-1}(\hat{\phi} - z_{1-\alpha/2} \cdot c) \leq \hat{\theta}(\mathbf{x}^*) \leq m^{-1}(\hat{\phi} - z_{\alpha/2} \cdot c)).$$

Moreover we see that $r_{\alpha/2}^* = m^{-1}(\hat{\phi} - z_{1-\alpha/2} \cdot c)$ and $r_{1-\alpha/2}^* = m^{-1}(\hat{\phi} - z_{\alpha/2} \cdot c)$. Therefore the percentile interval agrees with the standard normal interval applied to the appropriate transformation of θ . That is, the transformation that causes the assumptions of the standard

normal interval to actually hold. However, the appropriate transformation does not need to be known to construct the percentile interval, making it superior to the standard normal interval.

For a finite number of bootstrap replications the two-sided percentile interval is

$$(r_{(B+1)(\alpha/2)}^*, r_{(B+1)(1-\alpha/2)}^*).$$

Similar derivations can also be used to derive the one-sided versions:

$$(-\infty, r_{(B+1)(1-\alpha)}^*) \text{ and } (r_{(B+1)(\alpha)}^*, +\infty).$$

When the assumptions of the standard normal interval are met, the percentile interval will agree with it. When the standard normal interval would be correct for a certain transformation, the percentile interval agrees with the results of the standard normal interval applied under that transformation. There are many cases in which the assumption that such a transformation exists is quite reasonable. Such as when $\hat{\theta}(X)$ is the sample mean, proportion, or a regression coefficient. In these and other cases where a central limit theorem applies the identity transformation suffices.

Since its introduction the percentile interval has been interpreted in a pivotal framework (e.g. Hinkley 1988; Shao and Tu 1995). If the distribution of $\hat{\theta}(X) - \theta$ is symmetric, then $-a_{1-\alpha/2} = a_{\alpha/2}$ and $a_{1-\alpha/2} = -a_{\alpha/2}$. Therefore, we can rewrite (8.1) as

$$\left(\hat{\theta}(x) + a_{\alpha/2}, \hat{\theta}(x) + a_{1-\alpha/2} \right).$$

Upon estimating these quantiles with the appropriate order statistics from the bootstrap

distribution we obtain

$$\left(\hat{\theta}(\mathbf{x}) + a_{((B+1)(\alpha/2))}^*, \hat{\theta}(\mathbf{x}) + a_{((B+1)(1-\alpha/2))}^* \right).$$

Observe that $a_{((B+1)p)}^* = r_{((B+1)p)}^* - \hat{\theta}(\mathbf{x})$, so instead we can write

$$\hat{\theta}(\mathbf{x}) + a_{((B+1)(\alpha/2))}^* = r_{((B+1)(\alpha/2))}^* \quad \text{and} \quad \hat{\theta}(\mathbf{x}) + a_{((B+1)(1-\alpha/2))}^* = r_{((B+1)(1-\alpha/2))}^*.$$

Hence we arrive at the same quantiles of the bootstrap distribution.

The simplicity of the percentile interval provides a pedagogical advantage. Students can easily verify if the method is appropriate by checking the bootstrap distribution for normality. The interval is also transformation-respecting and range-preserving. However, the nonparametric percentile interval has received criticism for its poor performance (e.g. Hinkley 1988; Hesterberg 2015b; Hayden 2019). It has also been noted that the percentile interval uses the “wrong pivot backwards” relative to the basic interval (e.g. Hall 1992). This is discussed by Efron and Tibshirani (1993) who state that neither the percentile nor basic intervals, “work well in general”. However, they note that the percentile interval works better than the basic interval in practice.

A suggested improvement to the percentile interval is the bias-corrected and accelerated percentile interval, which accounts for possible bias in $\hat{\theta}(X)$. Its details are discussed in Chapter 14 of Efron and Tibshirani (1993). These details are more intricate and complex than those of the percentile and basic interval and, depending on the students’ mathematical backgrounds, they may be outside of the scope of an undergraduate introductory statistical methods course.

8.1.3 The Studentized Interval (The Studentized Case)

Under some circumstances, such as when $\hat{\theta}(X) = \bar{X}$, the distribution of $\hat{\theta}(X) - \theta$ is asymptotically Normal with mean 0 and variance $SE(\hat{\theta}(X))^2$, where $SE(\hat{\theta}(X))$ denotes the standard error of $\hat{\theta}(X)$. This provides another option for estimating $a_{1-\alpha/2}$ and $a_{\alpha/2}$. Namely, with $\hat{SE}(\hat{\theta}(x)) * z_{1-\alpha/2}$ and $\hat{SE}(\hat{\theta}(x)) * z_{\alpha/2}$, respectively.

For finite samples, however, this is only an approximation. In the case of the sample mean, a better approximation may be obtained by using the quantiles of a t_{n-1} distribution, which accounts for estimating the standard error. In this case, $a_{1-\alpha/2}$ and $a_{\alpha/2}$ are estimated with $\hat{SE}(\hat{\theta}(x)) * t_{(1-\alpha/2), (n-1)}$ and $\hat{SE}(\hat{\theta}(x)) * t_{\alpha/2, (n-1)}$, respectively.

The studentized bootstrap interval, also known as the bootstrap t -interval, further replaces these t -quantiles with a bootstrap approximation. Though its form is motivated by the t -interval it is useful for inference outside of the mean. Rather than using a z - or t -table, the studentized bootstrap interval uses “bootstrap tables” which are fit for the specific data set observed. This adjusts for skewness in the underlying population and other errors that can arise when $\hat{\theta}(X)$ is not the sample mean.

The values $t_{\alpha/2, (n-1)}$ and $t_{(1-\alpha/2), (n-1)}$ are estimated with the $(B+1)(\alpha/2)$ -th and $(B+1)(1-\alpha/2)$ -th smallest values of the distribution of $z^* = (\hat{\theta}(x^*) - \hat{\theta}(x)) / \hat{SE}(\hat{\theta}(x^*))$, respectively, where $\hat{SE}(\hat{\theta}(x^*))$ is an observed estimate of the standard error of $\hat{\theta}(x^*)$. Substituting these bootstrap estimates leads to an interval whose final form is

$$\left(\hat{\theta}(x) - \hat{SE}(\hat{\theta}(x)) * z_{((B+1)(1-\alpha/2))}^*, \hat{\theta}(x) - \hat{SE}(\hat{\theta}(x)) * z_{((B+1)(\alpha/2))}^* \right).$$

Though the Central Limit Theorem (CLT) gives a formula for the standard error of the mean, there are many statistics which do not have such a formula. The bootstrap may be

used to obtain estimates for the standard errors of $\hat{\theta}(X)$ and $\hat{\theta}(X^*)$. The *plug-in principle* discussed by Efron and Tibshirani (1993) can be used to estimate the standard error of $\hat{\theta}(X)$ with the square root of

$$\hat{\sigma}^2 = \frac{1}{B-1} \sum_{i=1}^B \left(\hat{\theta}(x_i^*) - \bar{\hat{\theta}}(x_{i(\cdot)}^*) \right)^2,$$

where $\bar{\hat{\theta}}(x_{i(\cdot)}^*)$ denotes the mean of the bootstrap sample statistics.

In order to estimate the standard error of $\hat{\theta}(x^*)$ an iterative bootstrap method can be used. In this method one obtains M second-level bootstrap samples from *each* of the B original bootstrap samples. For each of these second-level bootstrap samples, M statistics are then calculated and denoted as $\hat{\theta}(x_{i,j}^*)$ for $i = 1, \dots, B$ and $j = 1, \dots, M$. From these we calculate the bootstrap estimate of standard error for the i^{th} bootstrap sample as the square root of

$$\hat{\sigma}_i^{2*} = \frac{1}{M-1} \sum_{j=1}^M \left(\hat{\theta}(x_{i,j}^*) - \bar{\hat{\theta}}(x_{i(\cdot)}^*) \right)^2,$$

where $\bar{\hat{\theta}}(x_{i(\cdot)}^*)$ now represents the mean of the second-level bootstrap sample statistics.

While Efron and Tibshirani (1993) suggests that $M = 25$ is sufficient for estimating the standard error of a bootstrap estimate, $B = 1000$ is needed for estimating any desired quantiles. A few suggestions for M , ranging from 10 to 50, are also given by Davison and Hinkley (1997) under different scenarios. Depending on computational resources, these bootstrap methods, especially the studentized interval, may be considered computationally expensive. If $B = 999$ and $M = 25$, then over twenty-four thousand resamples must be performed in total.

As with the basic and percentile bootstrap intervals, the accuracy of this interval depends on whether the distribution of $(\hat{\theta}(X) - \theta) / \widehat{SE}(\hat{\theta}(X))$ is indeed pivotal. It is noted by Efron and Tibshirani (1993) that the results of the studentized bootstrap interval can be largely influenced by outliers in the data. They also warn that the studentized bootstrap interval

works best for variance-stabilized parameters and that it is especially applicable to location statistics.

Table 8.1 summarizes the three different bootstrap-based interval estimation methods discussed in this section along with their accompanying assumptions.

Simple bootstrap interval name and form	Underlying assumption(s)
Basic bootstrap interval $\left(2\hat{\theta}(\mathbf{x}) - r_{((B+1)(1-\alpha/2))}^*, 2\hat{\theta}(\mathbf{x}) - r_{((B+1)(\alpha/2))}^*\right)$	This method assumes that the distribution of $\hat{\theta}(X) - \theta$ is approximately pivotal
Percentile bootstrap interval $\left(r_{((B+1)(\alpha/2))}^*, r_{((B+1)(1-\alpha/2))}^*\right)$	There exists some monotone transformation $\hat{\phi} = m(\hat{\theta}(X))$ such that $\hat{\phi} \sim \text{Normal}(\phi, c^2)$ for all population distributions F (including the case $F = \hat{F}$), where $\phi = m(\theta)$, for some standard deviation c .
Studentized bootstrap interval $\left(\hat{\theta}(\mathbf{x}) - \hat{SE}(\hat{\theta}(\mathbf{x})) * z_{((B+1)(1-\alpha/2))}^*, \hat{\theta}(\mathbf{x}) - \hat{SE}(\hat{\theta}(\mathbf{x})) * z_{((B+1)(\alpha/2))}^*\right)$	The distribution of $(\hat{\theta}(X) - \theta) / \hat{SE}(\hat{\theta}(X))$ is approximately pivotal

Table 8.1: A summary of our discussion on the basic, percentile, and studentized bootstrap intervals. Here B is the number of bootstrap samples (e.g. 999) and $1 - \alpha$ is the desired confidence level. The values $(B + 1)(1 - \alpha/2)$ and $(B + 1)(\alpha/2)$ are assumed to be integers which, when used as subscripts, denote the corresponding order statistics of the distribution. We denote an estimate of the standard error of $\hat{\theta}(X)$, based on the data, as $\hat{SE}(\hat{\theta}(\mathbf{x}))$. Also, we denote the studentized distribution of bootstrap sample statistics as $z^* = (\hat{\theta}(\mathbf{x}^*) - \hat{\theta}(\mathbf{x})) / \hat{SE}(\hat{\theta}(\mathbf{x}^*))$, where $\hat{SE}(\hat{\theta}(\mathbf{x}^*))$ is an estimate of the standard error of $\hat{\theta}(\mathbf{x}^*)$.

8.2 Bootstrap-Based Hypothesis Tests

The goal of hypothesis testing is to make an inference about some population parameter of interest, θ , specifically in regards to whether or not there is sufficient evidence to indicate

that the parameter is a value other than one which we hypothesize to be true. Similar to confidence intervals, when the data do not meet the requirements needed to use traditional hypothesis testing methods, such as the z - or t -test, bootstrap hypothesis tests are an alternative so long as their own assumptions are met. Many early manuscripts and textbooks (e.g. Beran 1988; Hinkley 1988; Efron and Tibshirani 1993; Davison and Hinkley 1997) give guidance on bootstrap hypothesis testing and discuss possible approaches. The approach that we outline next is based on the idea of using a pivotal quantity. Readers who desire more details about this approach should reference Chapter 4 of Davison and Hinkley (1997) and Chapter 16 of Efron and Tibshirani (1993). A summary is given in Table 8.2 for readers who are already familiar with these concepts.

In general, to conduct a one-sample level- α bootstrap hypothesis test of $H_0 : \theta = \theta_0$, two components must be obtained: (1) $t(X)$, a test statistic, and (2) \hat{T}_0 , an estimate of, T , the distribution of $t(X)$, under H_0 . Pivotal bootstrap hypothesis tests use test statistics whose distributions do not depend on any unknown parameters, including θ , so that only \hat{T} needs to be estimated, without regards to H_0 .

Using the plug-in principle, B bootstrap test statistics, $t(x^*)$, can be generated from the bootstrap sample data and used to estimate T . The accuracy of this estimate depends on how well the distribution of $t(x^*)$ approximates that of $t(X)$. As was the case when estimating quantiles for confidence intervals in the last subsection, the two will conform well when $t(X)$ is actually pivotal.

For a one-sided lower alternative hypothesis, that is $H_A : \theta < \theta_0$, we can calculate the achieved significance level, an approximate p-value, with

$$ASL = P^* (t(x^*) < t(x)).$$

Here $t(x)$ is the observed test statistic, and we use an asterisk to note that this approxi-

mate probability is calculated using the distribution of the bootstrap test statistics. If the alternative hypothesis is one-sided upper, then

$$ASL = P^*(t(\mathbf{x}^*) > t(\mathbf{x}))$$

and if it is two-sided, then

$$ASL = 2 \times \min \left(P^*(t(\mathbf{x}^*) < t(\mathbf{x})), P^*(t(\mathbf{x}^*) > t(\mathbf{x})) \right).$$

In all cases, we reject H_0 if $ASL < \alpha$, where α is the desired significance level.

8.2.1 Studentized Pivots

Suppose that $t(X) = (\hat{\theta}(X) - \theta) / \hat{SE}(\hat{\theta}(X))$ is a pivotal quantity - its distribution does not depend on θ . Then the bootstrap hypothesis test outlined above may be used. In this case, T is estimated with the distribution of $t(\mathbf{x}^*) = (\hat{\theta}(\mathbf{x}^*) - \hat{\theta}(\mathbf{x})) / \hat{SE}(\hat{\theta}(\mathbf{x}^*))$ and the observed test statistic is $t(\mathbf{x}) = (\hat{\theta}(\mathbf{x}) - \theta_0) / \hat{SE}(\hat{\theta}(\mathbf{x}))$. Depending on the alternative hypothesis, the ASL can be calculated using one of the expressions given earlier.

Note that, if θ_0 is contained in the studentized interval given in Table 8.1, then

$$z_{((B+1)(\alpha/2))}^* < (\hat{\theta}(\mathbf{x}) - \theta_0) / \hat{SE}(\hat{\theta}(\mathbf{x})) < z_{((B+1)(1-\alpha/2))}^*.$$

The quantity in the center is $t(\mathbf{x})$, the observed test statistic based on a studentized pivotal quantity. If we estimate the $\alpha/2$ and $1 - \alpha/2$ quantiles of the distribution of $t(X)$ with the $(B + 1)(\alpha/2)$ -th and $(B + 1)(1 - \alpha/2)$ -th smallest values of the distribution of $t(\mathbf{x}^*)$, then containment of θ_0 in the studentized interval implies that $t(\mathbf{x})$ is in the rejection region of the two-sided level- α bootstrap hypothesis test. Therefore, performing this hypothesis test

is equivalent to rejecting values of θ_0 which are not contained in the studentized interval.

8.2.2 Locational Pivots

If we suppose, instead, that $t(X) = \hat{\theta}(X) - \theta$ is a pivotal quantity, then the bootstrap hypothesis test can again be used in a similar manner. In this case, the observed test statistic is $t(x) = \hat{\theta}(x) - \theta_0$ and the bootstrap test statistics, $t(x^*) = \hat{\theta}(x^*) - \hat{\theta}(x)$, can be used to estimate T . The ASL can be calculated using the statements defined earlier.

If θ_0 is contained in the basic bootstrap interval of Table 8.1 then,

$$r_{(B+1)(\alpha/2)}^* - \hat{\theta}(x) < \hat{\theta}(x) - \theta_0 < r_{(B+1)(1-\alpha/2)}^* - \hat{\theta}(x).$$

If it is contained in the percentile interval, then

$$-(r_{(B+1)(1-\alpha/2)}^* - \hat{\theta}(x)) < \hat{\theta}(x) - \theta_0 < -(r_{(B+1)(\alpha/2)}^* - \hat{\theta}(x)).$$

Again, we see that the quantity in the center of each statement is the observed test statistic, based on a locational pivot. Furthermore, by the symmetry assumption of the percentile interval, the bounds of these statements are the same. If the $(B + 1)(\alpha/2)$ -th and $(B + 1)(1 - \alpha/2)$ -th smallest values of the distribution of $t(x^*)$ are used to estimate the $\alpha/2$ and $1 - \alpha/2$ quantiles of T , then the values in the rejection region of this test are the same as the values contained in the basic bootstrap interval, or the percentile bootstrap interval under the symmetry assumption.

The use of pivotal quantities is not unique to bootstrap hypothesis testing. The z - and t -tests use the same underlying idea, with additional assumptions about the shape of the distribution of the test statistic. When the test statistic is not approximately pivotal, the

performance of these bootstrap hypothesis tests may be negatively impacted. For reference, these bootstrap hypothesis tests are summarized in Table 8.2.

8.3 Summary

The theoretical underpinnings that we have discussed show that the basic and studentized bootstrap intervals, and their corresponding hypothesis tests, rely heavily on the assumption that the distribution of $\hat{\theta}(X)$ can be made approximately pivotal through shifting (by θ) or studentization (shifting by θ and scaling by $S\hat{E}(\hat{\theta}(X))$). The percentile interval assumes that a normalizing transformation of the sampling distribution exists. Whether this is a reasonable assumption depends on the parameter of interest and the underlying population data. In many cases, such as that of the sample mean, the distribution of $\hat{\theta}(X)$ will depend on some scale parameter and the former assumption will be unreasonable. Next we will use simulations to investigate how these bootstrap methods perform when their assumptions are or are not reasonably met.

	Studentized pivot	Locational pivot
Assumption	Distribution of $(\hat{\theta}(X) - \theta)/\hat{SE}(\hat{\theta}(X))$ is approximately pivotal	Distribution of $\hat{\theta}(X) - \theta$ is approximately pivotal
Observed test statistic	$t(\mathbf{x}) = \frac{\hat{\theta}(\mathbf{x}) - \theta_0}{\hat{SE}(\hat{\theta}(\mathbf{x}))}$	$t(\mathbf{x}) = \hat{\theta}(\mathbf{x}) - \theta_0$
Bootstrap distribution of test statistics	$t(\mathbf{x}^*) = \frac{\hat{\theta}(\mathbf{x}^*) - \hat{\theta}(\mathbf{x})}{\hat{SE}(\hat{\theta}(\mathbf{x}^*))}$	$t(\mathbf{x}^*) = \hat{\theta}(\mathbf{x}^*) - \hat{\theta}(\mathbf{x})$
Rejection criteria	<p>Two-sided test rejects if</p> $ASL = 2 \times \min \left(P^*(t(\mathbf{x}^*) < t(\mathbf{x})), P^*(t(\mathbf{x}^*) > t(\mathbf{x})) \right).$ <p>One-sided upper test rejects if</p> $ASL = P^*(t(\mathbf{x}^*) > t(\mathbf{x})) < \alpha$ <p>One-sided lower test rejects if</p> $ASL = P^*(t(\mathbf{x}^*) < t(\mathbf{x})) < \alpha$	

Table 8.2: Summary of the bootstrap hypothesis tests discussed. $\hat{SE}(\hat{\theta}(\mathbf{x}))$ is an estimate for the standard error of $\hat{\theta}(\mathbf{x})$, while $\hat{SE}(\hat{\theta}(\mathbf{x}^*))$ is that for $\hat{\theta}(\mathbf{x}^*)$. The achieved significance level (ASL) is an approximate p -value, calculated with respect to the bootstrap distribution of test statistics. The level- α hypothesis test based on the studentized pivot is equivalent to rejecting values of θ_0 which are not contained in the $(1 - \alpha) * 100\%$ studentized bootstrap interval. Similarly, the test based on the locational pivot is equivalent to rejecting values of θ_0 which are not contained in the basic bootstrap interval or, if the symmetry assumption holds, the percentile bootstrap interval.

Chapter 9: Simulation-Based Performance Evaluations of the Bootstrap

To evaluate the performance of these bootstrap intervals and their corresponding hypothesis tests, we applied their two-sided versions under a variety of simulated scenarios where their assumptions were or were not reasonably met. We discuss the following performance metrics:

Coverage proportion (C): the proportion of two-sided intervals that contained the true parameter value. For a $(1 - \alpha)*100\%$ bootstrap interval, it is desirable to have this equal to $1 - \alpha$.

Significance level (α): the proportion of times that the null hypothesis was rejected, in favor of a two-sided alternative, when it was actually true. In light of the bootstrap hypothesis testing methods discussed, $\alpha = 1 - C$. That is, the proportion of times that $H_0 : \theta = \theta_0$ was rejected in favor of $H_A : \theta \neq \theta_0$, where $\theta_0 = \theta$, the true population parameter, at the α significance level, is equal to the proportion of $(1 - \alpha)*100\%$ two-sided intervals that did *not* contain the true parameter value.

Power (β): the proportion of times the null is rejected, in favor of a two-sided alternative, when it is in fact false. It is usually desirable to have this value increase to 1 as the sample size increases. For more insight, we studied the behavior of β as $|\theta_0 - \theta|$ increased, for a variety of increasing sample sizes. Since the corresponding two-sided bootstrap hypothesis tests reject any values that are not contained in the two-sided interval, this is simply the proportion of two-sided intervals that did not contain each hypothesized value of $\theta_0 \in [\theta - d, \theta + d]$, where d is some constant specifying the absolute distance from the truth.

For simplicity, we call these performance metrics by their theoretical names, however, our results are simulation-based and, therefore, some deviations from what we would expect

based on statistical theory can be expected.

Results pertaining to the proportion of intervals or hypothesis tests which exhibited some behavior (e.g. containment of a true or false parameter value) were calculated out of 10,000 intervals or tests, each constructed using a different random sample taken under the specified simulation constraints. However, in some cases, such as when the sample size was small, there was little to no variability to estimate and this produced studentized bootstrap intervals with undefined bounds ($0/0$ or a value divided by 0). In these cases we only considered intervals that did not contain undefined values when calculating performance metrics, so the performance metric was calculated out of fewer than 10,000 intervals. More information is given on this behavior as we discuss the simulation results and we note how many undefined intervals were observed in the results tables.

All studentized bootstrap intervals were constructed using the iterative second-level method discussed earlier. We elected to use the bootstrap estimate of standard error for the studentized interval in order to gain insight into the performance of the method when a formula for the standard error is not available.

In order to determine if there was any difference in the performance due to the number of bootstrap samples used, bootstrap intervals were constructed using both $B = 99$ and $B = 999$ bootstrap samples. Also, the significance level was kept at $\alpha = 0.05$ throughout. That is, all confidence intervals were constructed with a desired 0.95 coverage probability and hypothesis tests were conducted with a desired Type I error rate of 0.05. For comparison purposes, we included simulation results for traditional z and/or t methods as appropriate for a given problem. These were the one-sample z - and t -tests and intervals for the mean and the one-sample z -test and interval for the proportion (Wald interval). The details of these methods can be found in most any introductory statistics textbook.

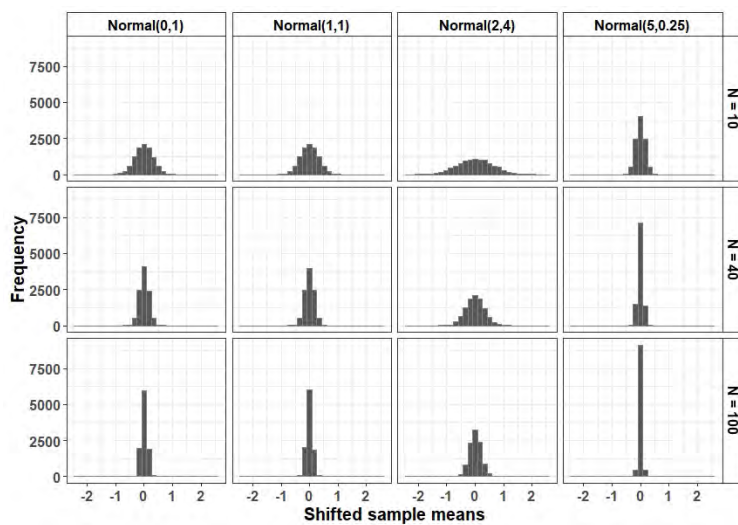
9.1 Simulation Results

We began with the problem of constructing interval estimates for the population mean under different scenarios. In the first scenario, random samples of size $N = 10, 40,$ and 100 were taken from a $\text{Normal}(1, 1)$ population. In the second scenario, random samples of size $N = 5, 10,$ and 20 were taken from an $\text{Exponential}(1)$ population (with rate parameter $\lambda = 1$), which is a right-skewed distribution. Connecting this problem to the notation used in the previous chapter, we have $\hat{\theta}(X) = \bar{X}$, the sample mean, and $\theta = \mu$, the population mean.

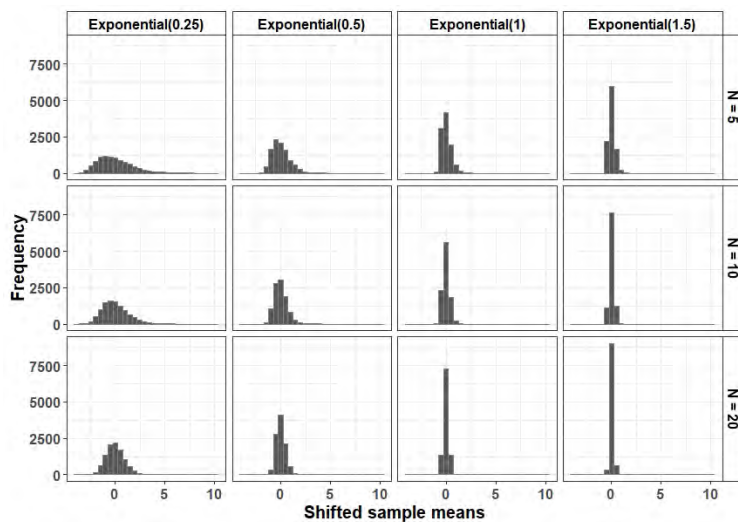
To determine if the assumptions of the basic and percentile bootstrap intervals were met, we calculated $\bar{X} - \mu$ ten thousand times using samples of varying sizes from a variety of $\text{Normal}(\mu, \sigma^2)$ and $\text{Exponential}(\lambda)$ populations. We selected values for λ such that the underlying population distributions would have less right-skew as λ increased. For the Normal population, μ and σ^2 were chosen such that the spread and center of the underlying population slightly varied. Figures 9.1a and 9.1b give the distributions of shifted sample means.

For Normal populations, it was clear that the spread of the distributions of $\bar{X} - \mu$ depended on the population variance. For example, in the first row, third column of Figure 9.1a, the spread of the distribution is greatest, while in the first row, fourth column it is least. This corresponds to changes in the variance of the underlying population. For the Exponential populations, inconsistencies were also observed between distributions as the skew and spread varied with λ . However, as the sample size increased, the distributions became more consistent across populations. For both scenarios, we concluded that the assumptions of the basic and percentile intervals were not met when the sample size was small but became better met as it increased.

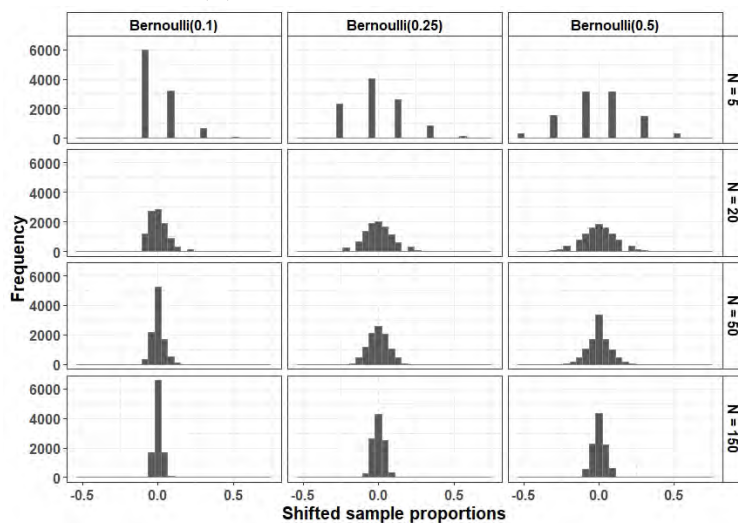
To determine if the assumption of the studentized bootstrap interval was met, we used the same simulations as before, but now we checked whether the distribution of $(\bar{X} - \mu)/\hat{\sigma}$ was



(a) Normal populations



(b) Exponential populations



(c) Bernoulli populations

Figure 9.1: Shifted sampling distributions. For each population and sample size 10,000 sample statistics were calculated. Each sample statistic was shifted by the corresponding parameter of its population.

approximately the same across the different populations. Here $\hat{\sigma}$ is the bootstrap-based plug-in estimate of standard error. We may refer to scaling by this estimate more broadly as “studentization”.

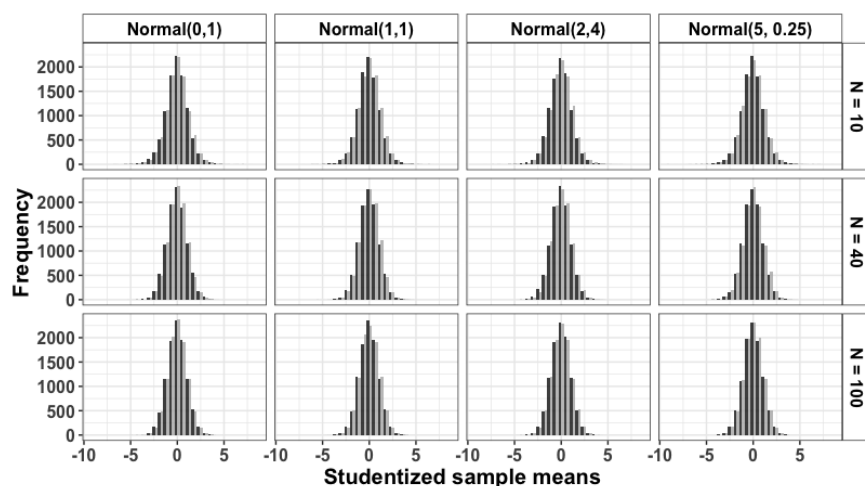
The simulated distributions are given in Figures 9.2a and 9.2b. The dark and light gray shading correspond to the use of $B = 99$ and $B = 999$ bootstrap samples, respectively. The value of B did not have an impact on the resulting distribution, but the sample size did. The first is evidenced by the strong overlap of the dark and light gray bars in the histograms and the second by the differences between histograms within the same column.

For example, comparing the distributions in the first column of Figure 9.2a, the spread in the distributions slightly decreases as N increases. However, making comparisons across the first row of Figure 9.2a, the distributions are approximately the same in shape, spread, and center. For these reasons, we concluded that the assumptions behind the studentized interval were met when the underlying population was Normal or Exponential and the parameter of interest was the mean.

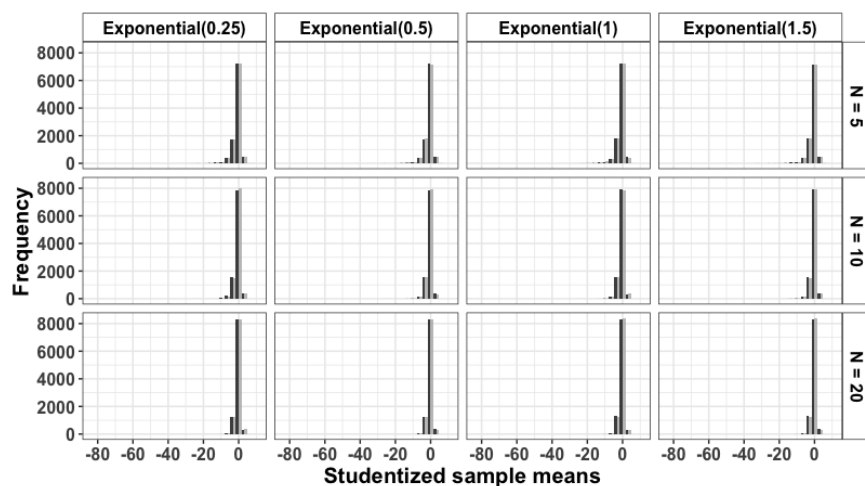
It is known that the t -interval does not perform well when N is small and the data are skewed (see Huang 2017; Meeden 1999). Therefore, the assumptions of the t -interval were reasonably met in the first scenario, where the underlying population was Normal, but less reasonably met in the second scenario, where the population was right-skewed and N was small. The assumptions of the z -interval were met in both scenarios since samples were independent and identically distributed (iid) and the underlying population variance was technically known.

The coverage proportions of the bootstrap intervals and the z - and t -intervals for the mean are given in Table 9.1 for each scenario of interest.

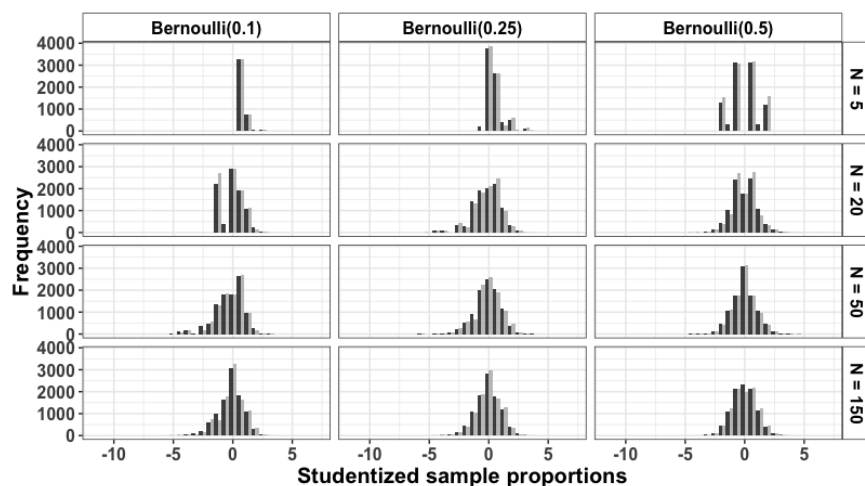
When the underlying population was Normal(1, 1), the coverage proportions of the z - and t -intervals were very close to the nominal 0.95, for most all values of N . Larger discrepancies



(a) Normal populations



(b) Exponential populations



(c) Bernoulli populations

Figure 9.2: Studentized sampling distributions. For each population, sample size, and number of bootstrap samples 10,000 sample statistics were calculated. Each sample statistic was shifted by the corresponding parameter of its population and scaled using its bootstrap estimate of standard error. $B = 99$ and $B = 999$ correspond to dark gray and light gray bars, respectively.

Interval	B	Normal(1, 1)		Exponential(1)	
		N	C	N	C
Basic	99	10	0.905	5	0.773
		40	0.941	10	0.849
		100	0.946	20	0.898
Basic	999	10	0.904	5	0.765
		40	0.941	10	0.842
		100	0.946	20	0.890
Percentile	99	10	0.907	5	0.789
		40	0.940	10	0.862
		100	0.946	20	0.909
Percentile	999	10	0.902	5	0.791
		40	0.940	10	0.863
		100	0.946	20	0.909
Studentized	99	10	0.962	5	0.949 (52)
		40	0.965	10	0.956
		100	0.965	20	0.961
Studentized	999	10	0.956	5	0.935
		40	0.962	10	0.950
		100	0.960	20	0.958
z	-	10	0.950	5	0.958
		40	0.950	10	0.959
		100	0.947	20	0.956
t	-	10	0.948	5	0.878
		40	0.949	10	0.905
		100	0.948	20	0.922

Table 9.1: Coverage proportions (C) of intervals for the population mean. 10,000 samples of the specified size (N) were taken from each population. Then, with each sample, B bootstrap samples were used to construct the bootstrap intervals while the traditional intervals were constructed with their usual formulas. The proportion of intervals which contained the true population mean, out of ten thousand, was calculated. These values should be near 0.95 since the significance level was 0.05. The coverage proportion for the studentized interval is out of 9982 intervals because it contained undefined bounds in 52 cases.

were observed for the bootstrap intervals though. The percentile and basic bootstrap intervals had moderate under-coverage, especially for small N . The lowest coverage observed amongst these two intervals for the Normal(1,1) population was 0.902. Alternatively, the studentized interval had over-coverage with proportions as large as 0.965.

When the population was Exponential(1), the coverage proportions of the t -interval dropped well below 0.95, while those of the z -interval reached above 0.95. Pointed decreases in the coverage proportions of the bootstrap intervals were also observed. The most severe changes were observed for the percentile and basic bootstrap intervals for $N = 5$. In these cases, some coverage proportions dropped by over 10%.

The coverage proportions of the studentized interval were higher than that of the t -interval when small samples were taken from an Exponential(1) population. However, the widths of the studentized bootstrap intervals were significantly larger than those of the t -intervals, especially when N was small. Figure 9.3 gives the distributions of the widths (upper bound - lower bound) of the studentized and t -intervals for each value of N when the underlying population was Exponential(1). These were plotted on the log scale for ease of visibility. The dashed lines in each panel mark the width of the z -interval, which is constant for a fixed N and significance level. The widths of the studentized interval were quite large and varied greatly, especially for $N = 5$. This explains why the coverage proportions were higher than that of the t -interval in this case.

Large widths were observed when the denominator of either $z_{((B+1)(\alpha/2))}^*$ or $z_{((B+1)(1-\alpha/2))}^*$ was near zero and this occurred when there was little variability between the second-level bootstrap sample statistics. In some extreme cases, all of them were the same and the second-level bootstrap estimate of standard error was exactly equal to zero, giving undefined values for $z_{((B+1)(\alpha/2))}^*$ or $z_{((B+1)(1-\alpha/2))}^*$. This behavior was observed in 52 (out of 10,000) intervals for the population mean. These intervals were removed before calculating the coverage proportions in Table 9.1.

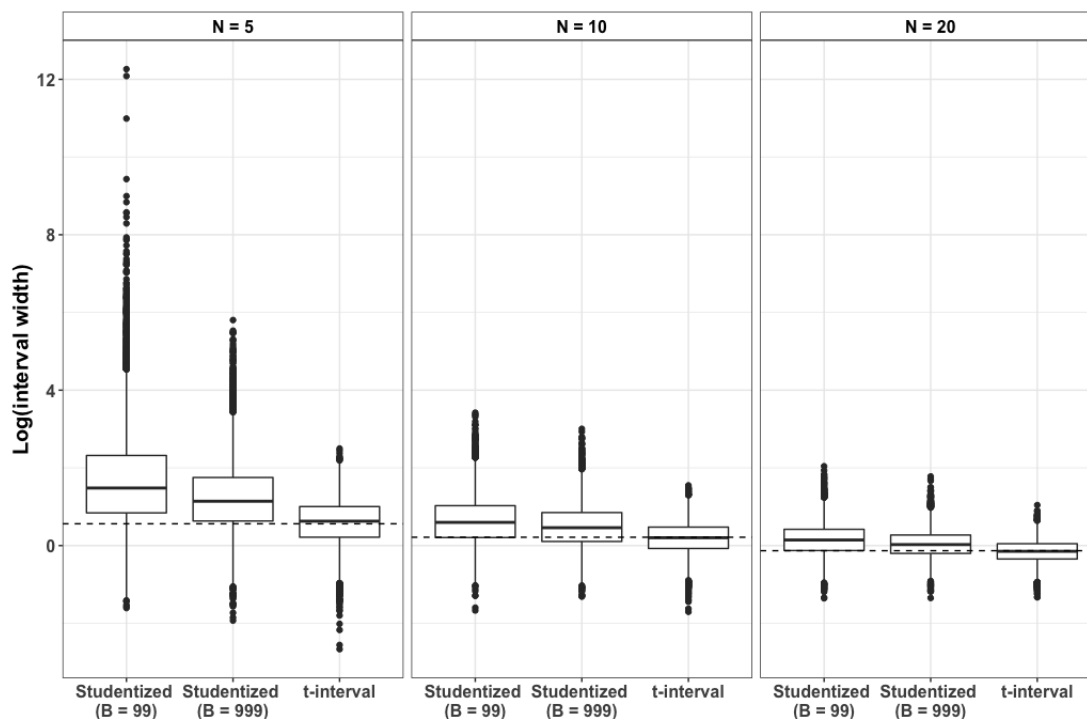


Figure 9.3: Log of the widths of 10,000 studentized bootstrap, t -, and z -intervals for the population mean when the underlying population was Exponential(1). The widths of the z -interval, which were constant for a given value of N and significance level, are marked by a single dashed line. The studentized bootstrap interval produced very wide intervals, especially when the sample size was very small. This may be a reason for the high coverage proportions we observed. 52 studentized bootstrap intervals had undefined values when $N = 5$ and $B = 99$. These were removed before plotting.

In the case of the population proportion, which we discuss next, there was even less variability to estimate since only TRUE or FALSE was sampled. Therefore, the original sample statistic, bootstrap sample statistics, and second-level bootstrap sample statistics were all the same in some cases. This also produced estimates of zero for the first- and second-level bootstrap estimates of standard error and, therefore, undefined bounds for the studentized intervals. We only considered intervals that did not have undefined bounds when calculating performance metrics.

The population proportion, p , is analogous to the mean for binary data. We evaluated the performance of the bootstrap intervals and the z -interval for proportions, also called the Wald interval, under a variety of scenarios. Connecting the notation from the previous chapter

to this problem, $\hat{\theta}(X) = \hat{p}$, the sample proportion, while $\theta = p$, the population proportion. We selected samples of size $N = 5, 20, 50$, and 150 from $\text{Bernoulli}(p)$ populations with $p \in \{0.1, 0.25, 0.5\}$. These values were selected so that the distribution of \hat{p} would vary from right-skewed when N and p were small, to symmetric when N was large and p was 0.5 .

The distributions of shifted and studentized sample proportions for samples from $\text{Bernoulli}(p)$ populations were given back in Figures 9.1c 9.2c, respectively. For $N = 5$ or 20 , the distributions of shifted and studentized sample proportions differed noticeably as p increased. These differences subsided slightly as N increased, though they were still noticeable. When N was small, we obtained some zero estimates for the standard error of \hat{p} , resulting in undefined studentized sample proportions. These were removed before plotting, which likely contributed to inconsistencies between the distributions in Figure 9.2c. Due to these observations, we again concluded that the assumptions of these bootstrap intervals were not well met in this scenario for small N , but, were better met for large sample sizes.

The z -interval for proportions is known to be inappropriate when the sample size is small and p is near zero or one (see Newcombe 1998; Brown et al. 2001). When p is near zero, the distribution of the number of successes, and therefore the proportion of successes, is right-skewed and when p is near one, it will be left-skewed. When the sample size is additionally small, this skewness makes the Normal approximation inappropriate.

The coverage proportions of the bootstrap intervals and the z -interval for the population proportion are given in Table 9.2. The coverage proportions of most intervals were quite far from the desired 0.95 regardless of N , B , or p .

The basic bootstrap interval mostly had under-coverage: for $p = 0.25$ and $p = 0.1$, it never achieved a coverage proportion at or above 0.95 , though it got close with 0.938 . For $N = 50$ or $N = 150$ and $p = 0.5$, the coverage proportions came closer to the desired 0.95 . The percentile interval had a mixture of over- and under-coverage. For $N = 5$, there was only

Interval	B	N	Bernoulli(0.1)	Bernoulli(0.25)	Bernoulli(0.5)
Basic	99	5	0.401 (0.314)	0.747 (0.423)	0.620 (0.414)
		20	0.879 (0.579)	0.897 (0.141)	0.936 (0.005)
		50	0.865 (0.230)	0.929 (0.001)	0.950 (0.005)
		150	0.929 (0.001)	0.938 (0.000)	0.949 (0.000)
Basic	999	5	0.401 (0.326)	0.749 (0.403)	0.620 (0.317)
		20	0.880 (0.611)	0.897 (0.124)	0.956 (0.001)
		50	0.879 (0.222)	0.935 (0.000)	0.948 (0.000)
		150	0.931 (0.000)	0.935 (0.000)	0.947 (0.000)
Percentile	99	5	0.404 (0.590)	0.751 (0.238)	0.924 (0.063)
		20	0.878 (0.119)	0.953 (0.003)	0.959 (0.000)
		50	0.952 (0.005)	0.940 (0.000)	0.957 (0.000)
		150	0.951 (0.000)	0.943 (0.000)	0.959 (0.000)
Percentile	999	5	0.401 (0.590)	0.749 (0.238)	0.937 (0.063)
		20	0.879 (0.119)	0.963 (0.003)	0.958 (0.000)
		50	0.960 (0.005)	0.939 (0.000)	0.960 (0.000)
		150	0.955 (0.000)	0.943 (0.000)	0.958 (0.000)
Studentized	99	5	0 (9999)	1 (9994)	1 (9982)
		20	0.954 (8820)	0.986 (2798)	0.993 (82)
		50	0.981 (3413)	0.981 (15)	0.974 (0)
		150	0.975 (7)	0.967 (0)	0.967 (0)
Studentized	999	5	0 (10000)	0 (10000)	0 (10000)
		20	0.961 (8745)	0.991 (2238)	0.998 (32)
		50	0.985 (2515)	0.983 (3)	0.972 (0)
		150	0.975 (2)	0.961 (0)	0.964 (0)
z	-	5	0.402 (0.410, 0.590)	0.742 (0.759, 0.241)	0.935 (0.935, 0.065)
		20	0.882 (0.744, 0.117)	0.894 (0.216, 0.004)	0.961 (0.002, 0.000)
		50	0.884 (0.244, 0.006)	0.940 (0.000, 0.001)	0.937 (0.000, 0.000)
		150	0.930 (0.000, 0.000)	0.937 (0.000, 0.000)	0.936 (0.000, 0.000)

Table 9.2: Coverage proportions (C) of bootstrap intervals and the z -interval (Wald interval) for the population proportion. Samples of size (N) were taken from each Bernoulli(p) population and B bootstrap samples were used to construct the bootstrap intervals. The z -interval was constructed using its usual formula. Values in parentheses represent the proportion of basic intervals which contained invalid values, the proportion of percentile intervals which contained equal bounds, the number of studentized intervals which contained undefined bounds, and the proportion of z -intervals which contained both invalid values and equal bounds.

under-coverage but results were not consistent for other values of N since there was both over- and under-coverage as p varied.

The studentized interval mostly had over-coverage, with coverage proportions as large as 0.974 when no intervals were undefined. For $N = 5$, there were very few intervals whose bounds were not undefined, if at all, especially when p was also small. The z -interval had under-coverage for the most part: its nearest coverage proportions were 0.940 and 0.961. Its lowest coverage proportions were observed when $p = 0.1$.

Another, possibly more serious, issue that we observed pertained to the behavior of the actual intervals themselves. The basic intervals contained invalid values and the percentile intervals had bounds which were exactly equal. The z -intervals also had invalid values and equal bounds and, as we already noted, the studentized intervals had undefined bounds. The frequency with which these issues were observed is given in parentheses next to the coverage proportions in Table 9.2.

The basic and percentile bootstrap intervals exhibited odd behavior mostly when $p = 0.1$ or N was small. In one case over 61% of basic bootstrap intervals contained invalid values and, in another case, 59% of percentile bootstrap intervals had equal bounds. However, as N and p increased, this behavior was not observed as frequently.

Use of the studentized interval produced many undefined bounds. When the underlying population proportion was near zero, or N was small, some second-level bootstrap estimates of standard error were zero, producing undefined values for the bootstrap z -statistics used to construct the interval, whose divisor is this estimated standard error. If there was also no variability in the original sample, then the z -statistic was $0/0$, which is also undefined. Undefined values were removed before calculating the coverage proportion which is why some coverage proportions were exactly zero or one. The number of intervals which were removed before calculating the coverage proportion is given in parentheses.

The z -interval exhibited behavior similar to that of the basic and percentile bootstrap intervals. This was especially true when N was small or $p = 0.1$. In these cases, both invalid values and intervals with equal bounds were observed. The behavior that we observed with the z -interval, and other issues that arise with its use, are also discussed by Newcombe (1998) and Brown et al. (2001).

As we mentioned earlier, the achieved significance level of the two-sided bootstrap hypothesis tests, α , is equivalent to $1 - C$. That is, since we calculated the proportion of intervals that contained the true parameter value, we also had the proportion of times we would fail to reject this true value if two-sided bootstrap hypothesis tests were performed. Subtracting this from one gave us the proportion of times we rejected this true null value. For brevity, we did not tabulate these since they are just one minus the values given in Tables 9.1 and 9.2. However, note that coverage proportions calculated when many studentized intervals had undefined bounds will less accurately reflect α .

In scenarios where few or no bootstrap intervals were removed, those which had coverage proportions near 0.95 also performed well in terms of significance levels near the desired 0.05. Those that had coverage proportions above or below 0.95 rejected too often or too rarely, respectively. Since many studentized intervals, both for the mean and proportion, had coverage proportions well above 0.95, it was the more conservative method in comparison to the basic and percentile bootstrap intervals.

For direct comparison, we obtained the rejection rates of the one-sample z - and t - tests for the mean as well as the z -test for the proportion. These are given in Tables 9.3 and 9.4. To obtain these, we performed each of these tests 10,000 times under each of the same scenarios used earlier and calculated the proportion of tests which rejected the true hypothesized value.

The rejection rates of the z -test for the mean were near the desired 0.05 in most cases. When $N = 10$ and the underlying population was Exponential(1), it was lowest at 0.041. When

Test	N	Normal(1,1)	N	Exponential(1)
<i>z</i> -test	10	0.048	5	0.041
	40	0.050	10	0.045
	100	0.051	20	0.048
<i>t</i> -test	10	0.051	5	0.120
	40	0.047	10	0.095
	100	0.050	20	0.081

Table 9.3: Significance levels (α) of the *z*- and *t*-tests for the mean. For each population and sample size (N), 10,000 samples were taken. The *z*- and *t*-test were used to test $H_0 : \mu = 1$ (which is true for both populations). The proportion of tests which rejected was recorded.

Test	N	Bernoulli(0.1)	Bernoulli(0.25)	Bernoulli(0.5)
<i>z</i> -test	5	0.083	0.017	0.061
	20	0.045	0.066	0.039
	50	0.031	0.051	0.062
	150	0.064	0.048	0.061

Table 9.4: Significance levels (α) of the *z*-test for proportions. For each Bernoulli(p) population and sample size (N), 10,000 samples were taken. The *z*-test for proportions was used to test $H_0 : p = p_0$, where p_0 was the true population proportion. The proportion of tests which rejected was recorded.

samples came from a Normal(1, 1) population, the *t*-test for the mean produced rejection rates near the desired 0.05. However, a non-trivial increase in its rejection rates was observed when small samples from an Exponential(1) population were used. This concurs with the decrease in coverage proportions that we also observed earlier. The rejection rates of the *z*-test for proportions was consistently far from 0.05 in both directions and there did not seem to be a clear pattern to these rates as N or p decreased.

We also investigated the performance of the *z*, *t*, and bootstrap hypothesis tests in regards to their ability to reject incorrect hypothesized values for the population mean and proportion. This performance metric was defined earlier as the power of these tests.

Figure 9.4 gives the rejection rates of the *z*, *t*, and bootstrap hypothesis tests for the mean under the same simulation constraints used earlier when the parameter of interest was the mean. The studentized intervals that produced invalid results were removed before calcu-

lating these rejection rates so some rates were calculated out of fewer than ten thousand intervals.

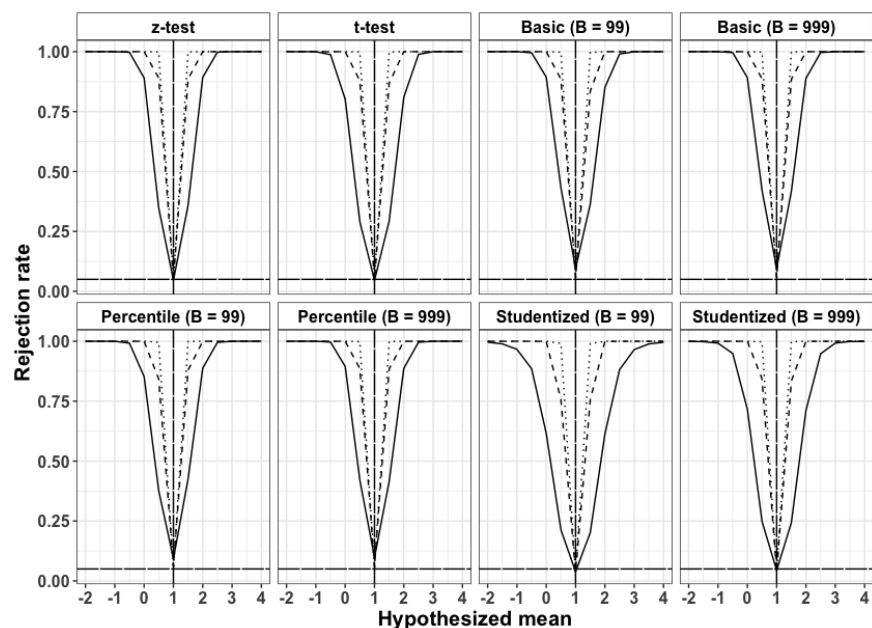
When the underlying population was $\text{Normal}(1, 1)$, the rejection rates of the z - and t -test were similar, though the former rejected incorrect values slightly more often. The studentized interval contained incorrect values, which were not rejected, more often than the basic and percentile intervals. Whether $B = 99$ or $B = 999$ did not seem to have a noticeable impact on the rejection rates. For all methods, the rejection rates improved as N increased.

This was also true when samples came from an $\text{Exponential}(1)$ population but the rejection curves were not nearly as well-behaved. For the t -test and bootstrap hypothesis tests, the rejection rates were far less symmetric about the true mean. Both the distance and direction with which the hypothesized value strayed away from the true mean impacted the results. The rejection rates of the t -test and bootstrap hypothesis tests reached one more quickly as the hypothesized mean moved below the true mean than when it moved above the true mean.

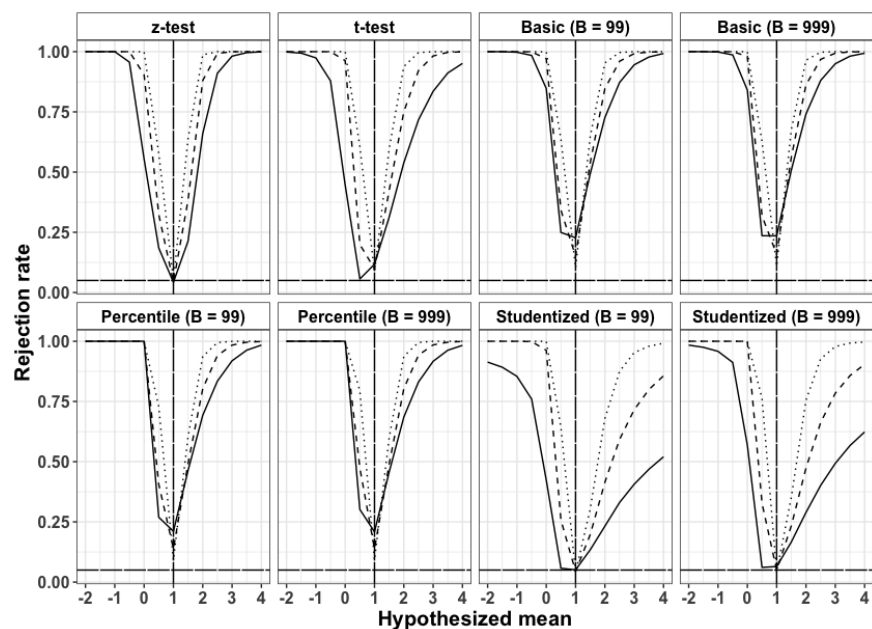
The test based on the studentized interval had even more conservative results than in the $\text{Normal}(1, 1)$ case and, even as the sample size increased, the rejection rates remained lowest of all methods. The distance between the lowest rejection rates and the significance level, marked by a long-dashed line at $y = 0.05$, was quite large for the hypothesis tests based on the basic and percentile bootstrap intervals. This agrees with the low coverage proportions we observed for these intervals earlier in the same population scenario.

The rejection rates of the z -test for proportions and the bootstrap hypothesis tests for the proportion are given in Figure 9.5.

Regardless of the method used or the value of p , the rejection rates went to one very slowly for $N = 5$. However, for $N = 150$, the rejection rates went to one more quickly as p_0 strayed away from p . For $p = 0.1$, larger samples were needed to more quickly achieve



(a) Normal(1,1) population. Line types denote the sample size (N): solid = 10, dashed = 40, dotted = 100.



(b) Exponential(1) population. Line types denote the sample size (N): solid = 5, dashed = 10, dotted = 20.

Figure 9.4: Rejection rates of z - and t -tests, and bootstrap hypothesis tests for the mean. The y-axis gives the proportion of tests that resulted in rejection for a given hypothesized value, on the x-axis. The vertical and horizontal long-dashed lines mark the true mean and the significance level 0.05, respectively.

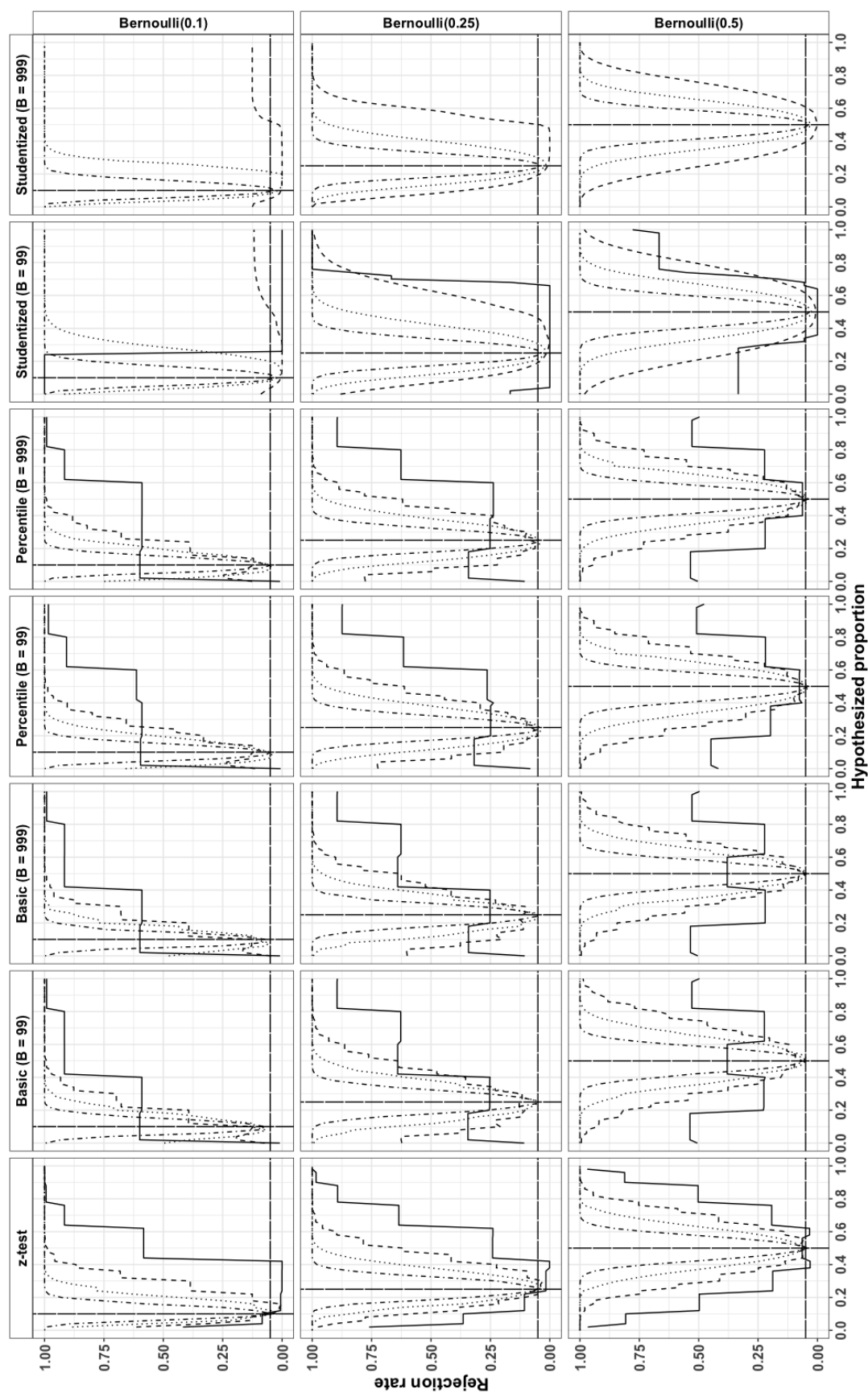


Figure 9.5: Rejection rates of the z -test for proportions and bootstrap hypothesis tests. The y -axis gives the proportion of tests that resulted in rejection for a given hypothesized proportion on the x -axis. The vertical and horizontal long-dashed lines mark the true population proportion, p , and the desired significance level, $\alpha = 0.05$, respectively. The line types map to values of N : solid for $N = 5$, dashed for $N = 20$, dotted for $N = 50$, long-dashed for $N = 150$. In some cases the studentized intervals had undefined bounds. These were removed, therefore, some rejection rates were calculated out of fewer than 10,000 intervals and some were exactly zero or one. For $N = 5$ and $B = 999$, none of the studentized intervals were well defined, so no rejection rates are plotted.

high rejection rates as p_0 strayed from p . The rejection rates of the hypothesis test based on the basic bootstrap interval were slightly less conservative than that based on the percentile bootstrap interval and the z -test. The studentized intervals performance was the most conservative. It had the lowest power since, as p_0 strayed from p , its rejection rates were consistently lower than that of the other methods.

9.2 Discussion

The results that we obtained primarily show that, when their underlying assumptions pertaining to pivotal quantities are not met, there can be non-trivial differences in the performance of the basic, percentile, and studentized bootstrap intervals. We observed decreased coverage proportions, increased Type I error rates, and decreased rejection rates when the null hypothesis was false. In many cases, the frequency with which these were observed increased as N decreased and rarely improved when B increased. Also, there did not appear to be an improvement in performance when these bootstrap methods were used as an alternative to traditional methods whose conditions were broken.

When the parameter of interest was the mean and the underlying population was Normal(1, 1) or Exponential(1), the shifted sampling distributions were inconsistent across populations, especially for small sample sizes. This provided evidence that the assumptions of the percentile and basic bootstrap intervals were less reasonable for small sample sizes. These intervals had lower coverage proportions and higher Type I error rates, when the sample size was small. The assumptions of the studentized interval were reasonable though, and its coverage proportions were better than that of the t -interval when small samples were taken from an Exponential(1) population.

In some cases, though, these high coverage proportions were not simply due to its superiority over the t -interval, but rather due to its larger widths. When the underlying population was

Normal(1, 1), the coverage proportions, Type I error rates, and correct rejection rates of these methods were better than when the population was Exponential(1). This indicates that non-Normality in the underlying population has an impact on the performance of these methods.

Taking this, and the coverage proportions of the z - and t -intervals into account, we concluded that their performance was better than the basic and percentile bootstrap intervals and comparable to, if not better than, the studentized bootstrap interval. Moreover, these bootstrap intervals were not necessarily an improvement over the t -interval in scenarios where it was known to have poor performance - that is, when N was small and the data were skewed.

When the parameter of interest was the population proportion, the assumptions of the basic and percentile bootstrap interval still were less reasonable for small sample sizes. Their coverage proportions and Type I error rates were not at the desired levels, especially for small N and p . In these cases, issues with the studentized interval also became more apparent and its assumptions were not reasonable. Estimating first- and second-level standard errors was an issue when N was small and, in many cases, we obtained undefined estimates. The reliability of the coverage proportions was negatively impacted by this since many intervals had to be disregarded.

In cases where few or no intervals were thrown out, the coverage proportions were high. However, this was again due to large widths rather than superiority over the other methods. This became apparent when we moved to assess the power of these methods. The studentized interval had lower power than other methods indicating that it contained incorrect values, which were not rejected, more frequently than the other methods. Meanwhile, the z -interval and the basic and percentile bootstrap intervals had comparable power. Again, we found that the bootstrap methods were not necessarily an improvement over the z -interval for proportions when it is known to perform poorly - that is, when N and p were small.

It is worth noting that the behavior of the studentized intervals may have been different if we were to use a formula to obtain estimates of the standard error of the original and bootstrap sample statistics. However, in many cases such a formula does not exist, therefore, we wanted to investigate the performance of the studentized interval when bootstrap estimates of standard error were used.

The performance of the studentized interval using a formula for the standard error of the mean was investigated by Hesterberg (2015b). They found that the studentized interval (called bootstrap t interval there) performed well for small samples from Normal and Exponential populations and outperformed the percentile and basic bootstrap intervals as well as the t -interval. Our results expand on this by analyzing its performance when the data are binary and we included direct comparisons with z - and t -methods.

The results that we obtained further emphasize the falsehood of the claim discussed by Hayden (2019) that “they are more accurate than traditional methods for small samples”. When the sample size was small, the metrics that we observed for the bootstrap intervals were not always an improvement over those of traditional methods. Even when they were, other issues came to the forefront, such as very large widths or undefined bounds. Moreover, the assumptions behind these intervals, which pertain to pivotal quantities, were less reasonable for small sample sizes. Our results show that the performance of these bootstrap methods decreases non-trivially as their assumptions, given in Tables 8.1 and 8.2, become less reasonable. It is important that the bootstrap intervals that we discussed be constructed using quantities that are pivotal after shifting or studentization.

Though the unfavorable results that we obtained were generated under specific settings, they still show that there are situations in which the bootstrap can fail, especially when the sample size is small and they are used for quantities which are not pivotal. Therefore, it is pertinent that the assumptions of these bootstrap methods be discussed when teaching them so that students use more caution when applying them and are aware of changes in their

performance when these assumptions are not met. Though their assumptions may be hard to verify in some cases, students should still be made aware of them and informed of cases where they are known to be unreasonable, such as those we reported and others given in the literature. This can help students to understand that these methods are not a direct solution when the assumptions of traditional methods are not met, but rather another option.

9.3 `bootEd`: An R Package for Teaching the Bootstrap

Section 4 showcased some ways in which simulations can be used to understand the assumptions behind these bootstrap intervals, verify how reasonable they are, and comprehend the repercussions of applying them when they are not reasonable. Performing similar simulations with students in the classroom can help students to better understand the methods taught so that they can reap the many benefits of applying the bootstrap and other statistical computing methods appropriately.

In order to assist with this, the functions that we used for the simulations were compiled into an R package called **`bootEd`**. These functions are straightforward applications of the intervals that we have discussed. We give a minimal example of how to use the package here. More information is given in the package repository at github.com/tottyn/bootEd. The code used to perform our simulations are also given there.

We begin by installing the package from GitHub and loading it:

```
devtools::install_github("tottyn/bootEd")  
library(bootEd)
```

Then, we construct a 95% percentile bootstrap interval for the population median using 999 bootstrap samples:

```
percentile(sample = rnorm(n = 20, mean = 3, sd = 2.5),
           parameter = "median", B = 999, siglevel = 0.05, onlyint = FALSE)
```

The `sample` argument takes the vector of data. In this example, the sample consists of 20 values randomly generated from a Normal(3, 6.25) distribution. The `parameter` argument can take any base R function (e.g. `sd`, `mean`) or any user defined summary function that returns a single value. The arguments `B` and `siglevel` take the number of bootstrap samples and significance level, respectively. When the `onlyint` argument is set to `TRUE` only the bootstrap interval is returned, which can be useful for performing simulations.

The following output and plot are given:

The percentile bootstrap interval for the median is: (2.199231, 3.386698).
 Assumptions: the shifted sampling distribution of the statistic of interest is symmetric and it does not depend on any unknown parameters, such as the underlying population variance.

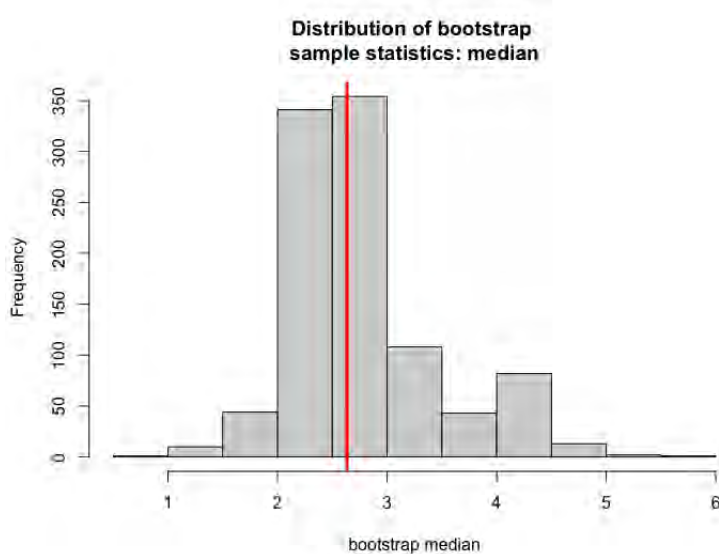


Figure 9.6: Histogram of bootstrap sample statistics. The original sample statistic is marked by a solid line. This plot is returned as part of the output from the `percentile` function.

The output of the function is not only the bootstrap interval, but also information about its assumptions. Verifying assumptions pertaining to the shifted or studentized sampling distribution can be difficult without prior knowledge about the sampling distribution. The bootstrap distribution is our best estimate to the sampling distribution and, though it is not exact, it can at least be used to determine if the assumptions of the method are reasonable.

In this example, the bootstrap distribution is not symmetric. Therefore, either the sampling distribution also is not symmetric or, if it is, then the bootstrap distribution is not an accurate reflection of it. Either way, the assumptions behind the percentile bootstrap interval are not met. Each of the interval construction methods that we discussed have a separate function in the package with similar syntax, output, and plots. These can be used to generate insightful discussion about the plausibility of the assumptions behind these methods, so that students can learn to use them responsibly.

When selecting a tool for statistical computing, teachers should consider the scope of the course in which the tool will be used and the computational backgrounds of the students. Teachers should avoid using tools that will bind students in the "ritualized thinking" that Son et al. (2021) indicates the teaching practices of traditional methods has unfortunately led to. Also, the criteria and goals set forth in the statistics education literature should be brought into consideration.

For example, a goal given in the GAISE is that students, "should be able to interpret and draw conclusions from standard output from statistical software packages." (GAISE College Report ASA Revision Committee 2016, p. 8) Important aspects of a contemporary statistical computing tool, which were discussed in great detail by McNamara (2018), should also be considered. These include accessibility, ease of entry, built-in documentation, and adjustable plot creation.

Other packages that are useful for teaching bootstrapping in introductory statistic courses

include: **boot** (Canty and Ripley (2019)), **wboot** (Weiss (2016)), **simpleboot** (Peng (2019)), **bootstrap** (Tibshirani and Leisch (2017)), **mosaic** (Pruim et al. (2017)), and **resample** (Hesterberg (2015a)). Though the **mosaic** package performs many tasks that do not pertain to bootstrapping, the **do** function is useful for rerunning code multiple times, as is needed for creating a bootstrap distribution. Also, the **resample** package has options for multiple resampling methods including the jackknife and permutation tests as well as capabilities for both one-sample and two-sample problems.

Chapter 10: Conclusions

We have discussed the percentile, basic, and studentized bootstrap intervals and their corresponding hypothesis tests were discussed. We showed that these methods have important underlying assumptions which should be discussed in the classroom. Performance metrics such as the coverage proportion, Type I error rate, and power were obtained under a variety of simulation scenarios. It was shown that the performance of these intervals differs non-trivially when their assumptions pertaining to pivotal quantities are or are not met. Specifically, when the sample size was small, these assumptions were less reasonable.

The performance metrics of their traditional counterparts, z - and t -methods for the mean and the z -interval for proportions (Wald interval), were also obtained under the same simulated scenarios. We found that when the assumptions of traditional methods were not met, these bootstrap intervals were rarely an improvement. Furthermore, their performance was also impacted by a small sample size and non-normality.

When teaching these bootstrap methods, it is pertinent that teachers emphasize that they are not substitutes for traditional methods nor are they solutions for issues that arise from having a small sample size. Their assumptions pertaining to pivotal quantities should be clearly communicated in lectures, course materials, and textbooks so that students leave the classroom with a broader understanding of these methods and how they relate to traditional methods. Teachers should aim to make students well informed about situations where these methods are already known to perform poorly and equip them with the ability to judge whether these methods are best for a given situation.

These methods can also be used as a conceptual stepping stone to teaching more traditional

methods. For example, Hesterberg (2015b) suggests that the distribution of studentized bootstrap sample statistics can effectively be used to evaluate whether CLT-based methods are appropriate for the specific data set. This could be done in addition to checking whether the sample data are Normal or the sample size is above 30. When a formula for the standard error is not available, however, the computational intensity and observed “small- N ” inaccuracies of the second-level bootstrap estimate of standard error should be kept in mind.

Our results pertain to the performance of the basic, percentile, and studentized bootstrap intervals for the population mean or proportion. While we studied Bernoulli(p) populations with $p \in \{0.1, 0.25, 0.5\}$, we did not investigate the performance of these methods for $p \in (0.5, 1]$. The performance of these methods could possibly be different as the population proportion increases past 0.5. We also have not discussed the performance of the “better bootstrap intervals” introduced by Efron and Tibshirani (1993) which are said to be an improvement. There is room for comparison between those bootstrap intervals and the ones that we have discussed here.

Future work could include an assessment of the performance of these methods when order statistics, such as the median, or non-location parameters, such as the correlation and variance, are used. When the data are skewed or there are outliers, we may encourage students to use the median as a measure of center. Statistical methods for the sample median that may be taught in undergraduate introductory statistics courses include the bootstrap intervals we have already discussed and the Sign test. The latter is known to have performance issues, in terms of power and Type I error, when observations are tied (see Coakley and Heise 1996; Fong et al. 2003). A comparison of the performance of the bootstrap intervals and the Sign test in these scenarios could also be investigated.

Also, an effort could be made to assess the performance of these methods in two-sample scenarios and compare their performance to that of permutation or two-sample traditional methods. An assessment of another claim discussed by Hayden (2019) - that these methods

are easier for students to understand could also be undertaken. A qualitative analysis of student understanding and engagement with different forms of instruction and content could be used to accomplish this.

The use of statistical computing in the classroom equips students with a variety of tools to use in many situations. It also increases students' retention of concepts and aids the teacher in explaining complex topics. We aim to benefit both teacher and student by making them aware of the assumptions behind simple bootstrapping methods which pertain to pivotal quantities. We did this so that they can better teach and implement bootstrapping with their introductory statistics students. It is important that these students understand the usefulness and the correct scope of this tool before leaving the classroom so that they are well equipped to handle a variety of situations.

Bibliography

- Agresti, A. (2012). *Categorical Data Analysis*. Vol. 792. John Wiley & Sons.
- Agresti, A. and C. Franklin (2007). *Statistics: the Art and Science of Learning From Data*. 1st edition. Pearson Education Limited.
- Alejo, R., R. M. Valdovinos, V. Garcia, and J. H. Pacheco-Sanchez (2013). “A hybrid method to face class overlap and class imbalance on neural networks and multi-class scenarios”. In: *Pattern Recognition Letters* 34.4, pp. 380–388.
- Ali, J., R. Khan, N. Ahmad, and I. Maqsood (2012). “Random Forests and Decision Trees”. In: *International Journal of Computer Science Issues (IJCSI)* 9.5, p. 272.
- Alin, Aylin (2010). “Simpson’s paradox”. In: *Wiley Interdisciplinary Reviews: Computational Statistics* 2.2, pp. 247–250.
- Ameri, S., M. J. Fard, R. B. Chinnam, and C. K. Reddy (2016). “Survival analysis based framework for early prediction of student dropouts”. In: *Proceedings of the 25th ACM International Conference on Information and Knowledge Management*, pp. 903–912.
- Athreya, K. B. and S. N. Lahiri (2006). *Measure Theory and probability Theory*. Vol. 19. Springer.
- Barandela, R., J. S. Sánchez, V. Garcia, and E. Rangel (2003). “Strategies for learning in class imbalance problems”. In: *Pattern Recognition* 36.3, pp. 849–851.
- Bartlett, P., Y. Freund, W. S. Lee, and R. E. Schapire (1998). “Boosting The Margin: A New Explanation For The Effectiveness Of Voting Methods”. In: *The Annals Of Statistics* 26.5, pp. 1651–1686.
- Barua, S., M. M. Islam, X. Yao, and K. Murase (2012). “MWMOTE—majority weighted minority oversampling technique for imbalanced data set learning”. In: *IEEE Transactions on knowledge and data engineering* 26.2, pp. 405–425.

- Batista, G., R. C. Prati, and M. C. Monard (2004). “A study of the behavior of several methods for balancing machine learning training data”. In: *ACM SIGKDD explorations newsletter* 6.1, pp. 20–29.
- (2005). “Balancing strategies and class overlapping”. In: *International symposium on intelligent data analysis*. Springer, pp. 24–35.
- Belgiu, M. and L. Drăguț (2016). “Random Forest In Remote Sensing: A Review Of Applications and Future Directions”. In: *ISPRS journal of photogrammetry and remote sensing* 114, pp. 24–31.
- Beran, R. (1988). “Prepivoting Test Statistics: A Bootstrap View of Asymptotic Refinements”. In: *Journal of the American Statistical Association* 83.403, pp. 687–697.
- Bernardo, A. and E. Della Valle (2021). “VFC-SMOTE: very fast continuous synthetic minority oversampling for evolving data streams”. In: *Data Mining and Knowledge Discovery* 35.6, pp. 2679–2713.
- Bettinger, E. (2015). “Need-based aid and college persistence: The effects of the Ohio College Opportunity Grant”. In: *Educational Evaluation and Policy Analysis* 37.1_suppl, 102S–119S.
- Bibal, A. and B. Frénay (2016). “Interpretability of machine learning models and representations: an introduction.” In: *ESANN*.
- Bikmukhametov, T. and J. Jäschke (2020). “Combining machine learning and process engineering physics towards enhanced accuracy and explainability of data-driven models”. In: *Computers & Chemical Engineering* 138, p. 106834.
- Borsos, Zalán, Camelia Lemnaru, and Rodica Potolea (2018). “Dealing with overlap and imbalance: a new metric and approach”. In: *Pattern Analysis and Applications* 21.2, pp. 381–395.
- Breiman, L. (1996). “Bagging Predictors”. In: *Machine learning* 24, pp. 123–140.
- (2001). “Random Forests”. In: *Machine Learning* 45, pp. 5–32.

- Brown, Lawrence D, T Tony Cai, and Anirban DasGupta (2001). “Interval estimation for a binomial proportion”. In: *Statistical science* 16.2, pp. 101–133.
- Bühlmann, P. and B. Yu (2002). “Analyzing Bagging”. In: *The Annals Of Statistics* 30.4, pp. 927–961.
- Bunkhumpornpat, C., K. Sinapiromsaran, and C. Lursinsap (2009). “Safe-level-smote: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem”. In: *Pacific-Asia conference on knowledge discovery and data mining*. Springer, pp. 475–482.
- Burkart, N. and M. F. Huber (2021). “A survey on the explainability of supervised machine learning”. In: *Journal of Artificial Intelligence Research* 70, pp. 245–317.
- Bussmann, N., P. Giudici, D. Marinelli, and J. Papenbrock (2020). “Explainable AI in fintech risk management”. In: *Frontiers in Artificial Intelligence* 3, p. 26.
- Canty, A. and B. D. Ripley (2019). *boot: Bootstrap R (S-Plus) Functions*. URL: <https://CRAN.R-project.org/package=boot>.
- Carvalho, D. V., E. M. Pereira, and J. S. Cardoso (2019). “Machine learning interpretability: A survey on methods and metrics”. In: *Electronics* 8.8, p. 832.
- Chatterjee, A., C. Marachi, S. Natekar, C. Rai, and F. Yeung (2018). “Using logistic regression model to identify student characteristics to tailor graduation initiatives”. In: *College Student Journal* 52.3, pp. 352–360.
- Chawla, N. V., K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer (2002). “SMOTE: synthetic minority over-sampling technique”. In: *Journal of artificial intelligence research* 16, pp. 321–357.
- Cieslak, D. A., N. V. Chawla, and A. Striegel (2006). “Combating imbalance in network intrusion datasets.” In: *GrC*. Citeseer, pp. 732–737.
- Coakley, Clint W and Mark A Heise (1996). “Versions of the sign test in the presence of ties”. In: *Biometrics*, pp. 1242–1251.

- D'Amico, M. M. and S. L. Dika (2013). “Using data known at the time of admission to predict first-generation college student success”. In: *Journal of College Student Retention: Research, Theory & Practice* 15.2, pp. 173–192.
- Davison, A. C. and D. V. Hinkley (1997). *Bootstrap Methods and Their Application*. Vol. 1. Cambridge University Press.
- Denil, M. and T. Trappenberg (2010). “Overlap versus imbalance”. In: *Canadian conference on artificial intelligence*. Springer, pp. 220–231.
- Diez-Pastor, J. F., J. J. Rodriguez, C. I. Garcia-Osorio, and L. I. Kuncheva (2015). “Diversity techniques improve the performance of the best imbalance learning ensembles”. In: *Information Sciences* 325, pp. 98–117.
- Domingos, P. (1999). “Metacost: A general method for making classifiers cost-sensitive”. In: *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 155–164.
- Douzas, G., F. Bacao, and F. Last (2018). “Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE”. In: *Information Sciences* 465, pp. 1–20.
- Eck, D. J. (2018). “Bootstrapping for Multivariate Linear Regression Models”. In: *Statistics & Probability Letters* 134, pp. 141–149.
- Efron, B. (1979). “Bootstrap Methods: Another Look at the Jackknife”. In: *The Annals of Statistics* 7.1, pp. 1–26. URL: <https://doi.org/10.1214/aos/1176344552>.
- Efron, B. and R. Tibshirani (1993). “An Introduction to The Bootstrap”. In: *Monographs on Statistics and Applied Probability* 57, pp. 1–436.
- Elreedy, D. and A. F. Atiya (2019). “A comprehensive analysis of synthetic minority oversampling technique (SMOTE) for handling class imbalance”. In: *Information Sciences* 505, pp. 32–64.

- Fernández, A., S. Garcia, F. Herrera, and N. V. Chawla (2018). “SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary”. In: *Journal of artificial intelligence research* 61, pp. 863–905.
- Fernández, Alberto, S. Garcia, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera (2018). “Algorithm-level approaches”. In: *Learning from Imbalanced Data Sets*. Springer, pp. 123–146.
- Field, A., J. Miles, and Z. Field (2012). *Discovering Statistics Using R*. Sage Publications.
- Fong, Daniel Y T, CW Kwan, KF Lam, and Karen S L Lam (2003). “Use of the sign test for the median in the presence of ties”. In: *The American Statistician* 57.4, pp. 237–240.
- Freund, Y. (1995). “Boosting A Weak Learning Algorithm By Majority”. In: *Information and Computation* 121.2, pp. 256–285.
- GAISE College Report ASA Revision Committee (2016). *Guidelines for Assessment and Instruction in Statistics Education College Report 2016*. URL: <http://www.amstat.org/education/gaise>.
- Galar, M., A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera (2011). “A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42.4, pp. 463–484.
- Garcia, S., J. Luengo, and F. Herrera (2016). “Tutorial on practical tips of the most influential data preprocessing algorithms in data mining”. In: *Knowledge-Based Systems* 98, pp. 1–29.
- Gehrke, Matthias, Tanja Kistler, Karsten Lübke, Norman Markgraf, Bianca Krol, and Sebastian Sauer (2021). “Statistics education from a data-centric perspective”. In: *Teaching Statistics* 43, S201–S215.
- Goldrick-Rab, S., D. Harris, R. Kelchen, and J. Benson (2012). “Need-based financial aid and college persistence experimental evidence from Wisconsin”. In: *Available at SSRN 1887826*.

- Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep learning*. MIT press.
- Gower, J. C. (1971). “A General Coefficient of Similarity and Some of Its Properties”. In: *Biometrics*, pp. 857–871.
- Haberman, S. J. (1976). *Generalized Residuals for Log-linear Models*, pp. 104–122.
- Hall, P. (1992). *The Bootstrap and Edgeworth Expansion*. Springer-Verlag New York.
- Han, H., W. Wang, and B. Mao (2005). “Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning”. In: *International conference on intelligent computing*. Springer, pp. 878–887.
- Hart, P. (1968). “The condensed nearest neighbor rule (corresp.)” In: *IEEE transactions on information theory* 14.3, pp. 515–516.
- Hastie, T., R. Tibshirani, and J. H. Friedman (2009). *The Elements Of Statistical Learning: Data Mining, Inference, and Prediction*. Vol. 2. Springer.
- Hayden, R. W. (2019). “Questionable Claims for Simple Versions of the Bootstrap”. In: *Journal of Statistics Education* 27.3, pp. 208–215.
- He, H., Y. Bai, E. A. Garcia, and S. Li (2008). “ADASYN: Adaptive synthetic sampling approach for imbalanced learning”. In: *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*. IEEE, pp. 1322–1328.
- He, H. and E. A. Garcia (2009). “Learning from imbalanced data”. In: *IEEE Transactions on knowledge and data engineering* 21.9, pp. 1263–1284.
- Hesterberg, T. C. (2015a). *resample: Resampling Functions*. R package version 0.4. URL: <https://CRAN.R-project.org/package=resample>.
- (2015b). “What Teachers Should Know About the Bootstrap: Resampling in the Undergraduate Statistics Curriculum”. In: *The American Statistician* 69.4, pp. 371–386.
- Hinkley, D. V. (1988). “Bootstrap Methods”. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 50.3, pp. 321–337.
- Horowitz, J. L. (2019). “Bootstrap Methods in Econometrics”. In: *Annual Review of Economics* 11, pp. 193–224.

- Horton, N. J. and J. S. Hardin (2015). “Teaching the next generation of statistics students to “think with data”: Special issue on statistics and the undergraduate curriculum”. In: *The American Statistician* 69.4, pp. 259–265.
- Hothorn, T., K. Hornik, and A. Zeileis (2006). “Unbiased Recursive Partitioning: A Conditional Inference Framework”. In: *Journal of Computational and Graphical Statistics* 15.3, pp. 651–674.
- Howington, E. B. (2017). “Teach a confidence interval for the median in the first statistics course”. In: *Teaching Statistics* 39.3, pp. 84–87.
- Huang, Hening (2017). “Uncertainty estimation with a small number of measurements, Part I: new insights on the t-interval method and its limitations”. In: *Measurement Science and Technology* 29.1, p. 015004.
- Hwang, G., S. Wang, and C. Lai (2021). “Effects of a social regulation-based online learning framework on students’ learning achievements and behaviors in mathematics”. In: *Computers & Education* 160, p. 104031.
- Ishitani, T. T. (2006). “Studying attrition and degree completion behavior among first-generation college students in the United States”. In: *The Journal of Higher Education* 77.5, pp. 861–885.
- Ismay, C. and A. Y. Kim (2019). *Statistical Inference via Data Science: A Modern Dive into R and the Tidyverse*. CRC Press. URL: <https://moderndive.com>.
- James, G., D. Witten, T. Hastie, and R. Tibshirani (2013). *An Introduction To Statistical Learning*. Vol. 112. Springer.
- Jeatrakul, P., K. W. Wong, and C. C. Fung (2010). “Classification of Imbalanced Data By Combining The Complementary Neural Network and SMOTE Algorithm”. In: *Neural Information Processing. Models and Applications: 17th International Conference, ICONIP 2010, Sydney, Australia, November 22-25, 2010, Proceedings, Part II* 17. Springer, pp. 152–159.

- Jo, T. and N. Japkowicz (2004). “Class imbalances versus small disjuncts”. In: *ACM Sigkdd Explorations Newsletter* 6.1, pp. 40–49.
- Johnson, J. M. and T. M. Khoshgoftaar (2019). “Deep learning and thresholding with class-imbalanced big data”. In: *2019 18th IEEE international conference on machine learning and applications (ICMLA)*. IEEE, pp. 755–762.
- Jones-White, D. R., P. M. Radcliffe, R. L. Huesman, and J. P. Kellogg (2010). “Redefining student success: Applying different multinomial regression techniques for the study of student graduation across institutions of higher education”. In: *Research in Higher Education* 51.2, pp. 154–174.
- Kabra, R.R. and R.S. Bichkar (2011). “Performance prediction of engineering students using decision trees”. In: *International Journal of computer applications* 36.11, pp. 8–12.
- Kahu, E. R. and K. Nelson (2018). “Student engagement in the educational interface: Understanding the mechanisms of student success”. In: *Higher Education Research & Development* 37.1, pp. 58–71.
- Kamalov, F. and D. Denisov (2020). “Gamma distribution-based sampling for imbalanced data”. In: *Knowledge-Based Systems* 207, p. 106368.
- Khaouane, L., Y. Ammi, and S. Hanini (2017). “Modeling the Retention of Organic Compounds By Nanofiltration and Reverse Osmosis Membranes Using Bootstrap Aggregated Neural Networks”. In: *Arabian Journal for Science and Engineering* 42.4, pp. 1443–1453.
- Kotsiantis, S., D. Kanellopoulos, and P. Pintelas (2006). “Handling imbalanced datasets: A review”. In: *GESTS international transactions on computer science and engineering* 30.1, pp. 25–36.
- Krawczyk, Bartosz (2016). “Learning from imbalanced data: open challenges and future directions”. In: *Progress in Artificial Intelligence* 5.4, pp. 221–232.
- Kubat, M. and S. Matwin (1997). “Addressing the curse of imbalanced training sets: one-sided selection”. In: *Icml*. Vol. 97. 1. Citeseer, p. 179.

- Lakshminarayanan, B., D. M. Roy, and Y. W. Teh (2014). “Mondrian forests: Efficient Online Random Forests”. In: *Advances In Neural Information Processing Systems* 27.
- Lee, H., J. Kim, and S. Kim (2017). “Gaussian-based SMOTE algorithm for solving skewed class distributions”. In: *International Journal of Fuzzy Logic and Intelligent Systems* 17.4, pp. 229–234.
- Levy, J. J. and A. J. O’Malley (2020). “Don’t dismiss logistic regression: the case for sensible extraction of interactions in the era of machine learning”. In: *BMC medical research methodology* 20.1, pp. 1–15.
- Li, D., C. Liu, and S. C. Hu (2010). “A Learning Method For The Class Imbalance Problem With Medical Data Sets”. In: *Computers In Biology and Medicine* 40.5, pp. 509–518.
- Li, Zhuang, Jingyan Qin, Xiaotong Zhang, and Yadong Wan (2021). “Addressing Class Overlap under Imbalanced Distribution: An Improved Method and Two Metrics”. In: *Symmetry* 13.9, p. 1649.
- Liu, X., J. Wu, and Z. Zhou (2008). “Exploratory undersampling for class-imbalance learning”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 39.2, pp. 539–550.
- Lock, R. H. and P. F. Lock (2008). “Introducing statistical inference to biology students through bootstrapping and randomization”. In: *Primus* 18.1, pp. 39–48.
- Lock, R. H., P. F. Lock, K. L. Morgan, E. F. Lock, and D. F. Lock (2020). *Statistics: Unlocking the Power of Data*. John Wiley & Sons.
- London, A. J. (2019). “Artificial intelligence and black-box medical decisions: accuracy versus explainability”. In: *Hastings Center Report* 49.1, pp. 15–21.
- Makridakis, S. (1993). “Accuracy measures: theoretical and practical concerns”. In: *International journal of forecasting* 9.4, pp. 527–529.
- Mani, I. and I. Zhang (2003). “kNN approach to unbalanced data distributions: a case study involving information extraction”. In: *Proceedings of workshop on learning from imbalanced datasets*. Vol. 126. ICML, pp. 1–7.

- McNamara, A. (2018). “Key attributes of a modern Statistical Computing Tool”. In: *The American Statistician*.
- Meeden, Glen (1999). “Interval estimators for the population mean for skewed distributions with a small sample size”. In: *Journal of Applied Statistics* 26.1, pp. 81–96.
- Miller, J. W. and S. S. Lesik (2014). “College persistence over time and participation in a first-year seminar”. In: *Journal of College Student Retention: Research, Theory & Practice* 16.3, pp. 373–390.
- Mqadi, N. M., N. Naicker, and T. Adeliyi (2021). “Solving Misclassification of The Credit Card Imbalance Problem Using Near Miss”. In: *Mathematical Problems in Engineering* 2021, pp. 1–16.
- Mullin, C. M. (2012). “Student success: Institutional and individual perspectives”. In: *Community College Review* 40.2, pp. 126–144.
- Napierała, K., J. Stefanowski, and S. Wilk (2010). “Learning From Imbalanced Data In The Presence Of Noisy and Borderline Examples”. In: *Rough Sets and Current Trends in Computing: 7th International Conference, RSCTC 2010, Warsaw, Poland, June 28-30, 2010. Proceedings 7*. Springer, pp. 158–167.
- Natek, S. and M. Zwillling (2014). “Student data mining solution–knowledge management system related to higher education institutions”. In: *Expert systems with applications* 41.14, pp. 6400–6407.
- Newcombe, Robert G (1998). “Two-sided confidence intervals for the single proportion: comparison of seven methods”. In: *Statistics in medicine* 17.8, pp. 857–872.
- Nivetha, S., B. Valarmathi, K. Santhi, and T. Chellatamilan (2020). “Detection of Type 2 Diabetes Using Clustering Methods – Balanced and Imbalanced Pima Indian Extended Dataset”. In: *Proceeding of the International Conference on Computer Networks, Big Data and IoT (ICCBI-2019)*. Springer, pp. 610–619.

- Núñez-Peña, M. I., M. Suárez-Pellicioni, and R. Bono (2013). “Effects of math anxiety on student success in higher education”. In: *International Journal of Educational Research* 58, pp. 36–43.
- Nwankpa, C. E., W. Ijomah, A. Gachagan, and S. Marshall (2021). “Activation Functions: Comparison of Trends in Practice and Research for Deep Learning”. In: *2nd International Conference on Computational Sciences and Technology*.
- Oh, S. (2011). “A new dataset evaluation method based on category overlap”. In: *Computers in Biology and Medicine* 41.2, pp. 115–122.
- Osuolale, F. N. and J. Zhang (2018). “Exergetic Optimisation of Atmospheric and Vacuum Distillation System Based on Bootstrap Aggregated Neural Network Models”. In: *Exergy for A Better Environment and Improved Sustainability 1*. Springer, pp. 1033–1046.
- Patil, D. D., V.M. Wadhai, and J.A. Gokhale (2010). “Evaluation Of Decision Tree Pruning Algorithms For Complexity and Classification Accuracy”. In: *International Journal of Computer Applications* 11.2, pp. 23–30.
- Pelawa Watagoda, L. C. R. and D. J. Olive (2021). “Bootstrapping Multiple Linear Regression After Variable Selection”. In: *Statistical Papers* 62.2, pp. 681–700.
- Peng, R. D. (2019). *simpleboot: Simple Bootstrap Routines*. URL: <https://CRAN.R-project.org/package=simpleboot>.
- Perna, L. W. and S. L. Thomas (2008). “Theoretical Perspectives on Student Success: Understanding the Contributions of the Disciplines.” In: *ASHE higher education report* 34.1, pp. 1–87.
- Prati, R. C., G. Batista, and M. C. Monard (2004). “Learning with class skews and small disjuncts”. In: *Brazilian Symposium on Artificial Intelligence*. Springer, pp. 296–306.
- Pruim, R., D. T. Kaplan, and N. J. Horton (2017). “The mosaic Package: Helping Students to ‘Think with Data’ Using R”. In: *The R Journal* 9.1, pp. 77–102.

- Raju, D. and R. Schumacker (2015). “Exploring student characteristics of retention that lead to graduation in higher education using data mining models”. In: *Journal of College Student Retention: Research, Theory & Practice* 16.4, pp. 563–591.
- Rivera, W. A. (2017). “Noise reduction a priori synthetic over-sampling for class imbalanced data sets”. In: *Information Sciences* 408, pp. 146–161.
- Rojas, R. (2013). *Neural networks: a systematic introduction*. Springer Science & Business Media.
- Sáez, J. A., J. Luengo, J. Stefanowski, and F. Herrera (2015). “SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering”. In: *Information Sciences* 291, pp. 184–203.
- Schulz, L. O., P. H. Bennett, E. Ravussin, J. R. Kidd, K. K. Kidd, J. Esparza, and M. E. Valencia (2006). “Effects Of Traditional and Western Environments On Prevalence Of Type 2 Diabetes In Pima Indians In Mexico and The US”. In: *Diabetes care* 29.8, pp. 1866–1871.
- Shahee, Shaukat Ali and Usha Ananthakumar (2021). “An overlap sensitive neural network for class imbalanced data”. In: *Data Mining and Knowledge Discovery* 35.4, pp. 1654–1687.
- Shao, J. and D. Tu (1995). *The Jackknife and Bootstrap*. Springer New York, NY.
- Smith, J. W., J. E. Everhart, W.C. Dickson, W. C. Knowler, and R. S. Johannes (1988). “Using The ADAP Learning Algorithm To Forecast The Onset Of Diabetes Mellitus”. In: *Proceedings Of The Annual Symposium On Computer Application In Medical Care*. American Medical Informatics Association, p. 261.
- Soh, W. W. and R. M. Yusuf (2019). “Predicting Credit Card Fraud On A Imbalanced Data”. In: *International Journal of Data Science and Advanced Analytics* 1.1, pp. 12–17.
- Son, J. Y., A. B. Blake, L. Fries, and J. W. Stigler (2021). “Modeling First: Applying Learning Science to the Teaching of Introductory Statistics”. In: *Journal of Statistics and Data Science Education* 29.1, pp. 4–21.

- Stefanowski, J. (2016). “Dealing with data difficulty factors while learning from imbalanced data”. In: *Challenges in computational statistics and data mining*. Springer, pp. 333–363.
- Tarawneh, A. S., A. B.A. Hassanat, K. Almohammadi, D. Chetverikov, and C. Bellinger (2020). “SMOTEFUNA: Synthetic minority over-sampling technique based on furthest neighbour algorithm”. In: *IEEE Access* 8, pp. 59069–59082.
- Terenzini, P. T., L. Springer, P. M. Yaeger, E. T. Pascarella, and A. Nora (1996). “First-generation college students: Characteristics, experiences, and cognitive development”. In: *Research in Higher education* 37.1, pp. 1–22.
- Thai-Nghe, N., Z. Gantner, and L. Schmidt-Thieme (2010). “Cost-sensitive learning methods for imbalanced data”. In: *The 2010 International joint conference on neural networks (IJCNN)*. IEEE, pp. 1–8.
- Tibshirani, R. and F. Leisch (2017). *bootstrap: Functions for the Book "An Introduction to the Bootstrap"*. URL: <https://CRAN.R-project.org/package=bootstrap>.
- Tintle, N. L., K. Topliff, J. VanderStoep, V. Holmes, and T. Swanson (2012). “Retention of Statistical Concepts in a Preliminary Randomization-Based Introductory Statistics Curriculum”. In: *Statistics Education Research Journal* 11.1, p. 21.
- Tinto, V. (1987). *Leaving college: Rethinking the causes and cures of student attrition*. ERIC.
- (2006). “Research and practice of student retention: What next?” In: *Journal of college student retention: Research, Theory & Practice* 8.1, pp. 1–19.
- Tomek, I. (1976). “Two modifications of CNN”. In: *IEEE Trans. Systems, Man and Cybernetics* 6, pp. 769–772.
- Tuggener, L., M. Amirian, K. Rombach, S. Lörwald, A. Varlet, C. Westermann, and T. Stadelmann (2019). “Automated machine learning in practice: state of the art and recent results”. In: *2019 6th Swiss Conference on Data Science (SDS)*. IEEE, pp. 31–36.
- Utzet, F. and Á. Sánchez (2021). “Some Applications of the Bootstrap to Survival Analysis”. In.


- Weiss, N. A. (2016). *wBoot: Bootstrap Methods*. URL: <https://CRAN.R-project.org/package=wBoot>.
- Wilson, D. L. (1972). “Asymptotic properties of nearest neighbor rules using edited data”. In: *IEEE Transactions on Systems, Man, and Cybernetics* 3, pp. 408–421.
- Wood, M. (2005). “The Role of Simulation Approaches in Statistics”. In: *Journal of Statistics Education* 13.3.
- Xiong, H., J. Wu, and L. Liu (2010). “Classification with class overlapping: A systematic study”. In: *The 2010 International Conference on E-Business Intelligence*, pp. 491–497.
- Yang, Y. Y., M. Mahfouf, G. Panoutsos, Q. Zhang, and S. Thornton (2011). “Adaptive neural-fuzzy inference system for classification of rail quality data with bootstrapping-based over-sampling”. In: *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*. IEEE, pp. 2205–2212.
- Yu, C. H., S. DiGangi, A. Jannasch-Pennell, and C. Kaprolet (2010). “A data mining approach for identifying predictors of student retention from sophomore to junior year”. In: *Journal of Data Science* 8.2, pp. 307–325.
- Zeineddine, H., U. Braendle, and A. Farah (2021). “Enhancing prediction of student success: Automated machine learning approach”. In: *Computers & Electrical Engineering* 89, p. 106903.
- Zeini, H. A., D. Al-Jeznawi, H. Imran, L. Filipe A. Bernardo, Z. Al-Khafaji, and K. A. Ostrowski (2023). “Random Forest Algorithm For The Strength Prediction of Geopolymer Stabilized Clayey Soil”. In: *Sustainability* 15.2, p. 1408.
- Zhu, Bing, Bart Baesens, and Seppe KLM vanden Broucke (2017). “An empirical comparison of techniques for the class imbalance problem in churn prediction”. In: *Information sciences* 408, pp. 84–99.

APPENDICES

Appendix A: Supplementary Visualizations of SMOTE Simulation Results

	Sens (TP/P)				Spec (TN/N)				
Base Case	0.932 (0.005)	0.627 (0.015)	0.936 (0.05)	0.736 (0.065)	0.917 (0.004)	0.593 (0.017)	0.841 (0.063)	0.416 (0.1)	LASSO Logistic Reg.
N = 1750	0.939 (0.007)	0.641 (0.016)	0.935 (0.053)	0.716 (0.069)	0.931 (0.004)	0.621 (0.017)	0.897 (0.047)	0.502 (0.112)	
150 vars	0.956 (0.003)	0.654 (0.013)	0.964 (0.046)	0.774 (0.06)	0.947 (0.003)	0.611 (0.015)	0.876 (0.061)	0.416 (0.097)	
20 vars	0.746 (0.011)	0.557 (0.039)	0.788 (0.062)	0.664 (0.095)	0.748 (0.01)	0.549 (0.041)	0.69 (0.092)	0.425 (0.112)	
25% cat	0.857 (0.024)	0.554 (0.009)	0.942 (0.033)	0.81 (0.009)	0.813 (0.026)	0.49 (0.009)	0.575 (0.1)	0.257 (0.007)	
85% cat	0.649 (0.014)	0.547 (0.009)	0.88 (0.013)	0.838 (0.016)	0.585 (0.009)	0.466 (0.01)	0.265 (0.02)	0.204 (0.014)	
30% MCAR	0.959 (0.003)	0.651 (0.011)	0.973 (0.033)	0.827 (0.057)	0.944 (0.003)	0.591 (0.013)	0.876 (0.043)	0.327 (0.091)	
70% MCAR	0.99 (0.001)	0.708 (0.013)	0.998 (0.003)	0.973 (0.064)	0.979 (0.001)	0.579 (0.015)	0.796 (0.062)	0.062 (0.129)	
30% MAR	0.947 (0.003)	0.688 (0.009)	0.967 (0.015)	0.815 (0.049)	0.944 (0.003)	0.607 (0.012)	0.867 (0.031)	0.363 (0.098)	
70% MAR	0.971 (0.001)	0.806 (0.005)	0.991 (0.004)	0.943 (0.035)	0.973 (0.002)	0.516 (0.018)	0.752 (0.056)	0.035 (0.134)	
Base Case	0.551 (0.01)	0.552 (0.01)	0.851 (0.027)	0.851 (0.028)	0.45 (0.009)	0.451 (0.009)	0.152 (0.026)	0.152 (0.028)	Neural Network
N = 1750	0.552 (0.022)	0.552 (0.021)	0.855 (0.048)	0.852 (0.048)	0.451 (0.02)	0.451 (0.019)	0.156 (0.047)	0.152 (0.047)	
150 vars	0.553 (0.023)	0.553 (0.021)	0.853 (0.042)	0.852 (0.039)	0.452 (0.02)	0.452 (0.018)	0.155 (0.042)	0.154 (0.039)	
20 vars	0.552 (0.005)	0.55 (0.003)	0.851 (0.008)	0.85 (0.007)	0.45 (0.004)	0.448 (0.003)	0.152 (0.008)	0.151 (0.008)	
25% cat	0.553 (0.003)	0.551 (0.003)	0.855 (0.003)	0.854 (0.003)	0.452 (0.003)	0.45 (0.003)	0.158 (0.005)	0.156 (0.005)	
85% cat	0.563 (0.005)	0.552 (0.006)	0.907 (0.014)	0.901 (0.014)	0.462 (0.005)	0.451 (0.006)	0.307 (0.06)	0.286 (0.056)	
30% MCAR	0.552 (0.008)	0.554 (0.008)	0.851 (0.032)	0.852 (0.031)	0.451 (0.007)	0.452 (0.007)	0.151 (0.031)	0.151 (0.03)	
70% MCAR	0.553 (0.005)	0.55 (0.005)	0.849 (0.019)	0.849 (0.019)	0.451 (0.004)	0.449 (0.005)	0.151 (0.023)	0.151 (0.023)	
30% MAR	0.56 (0.014)	0.563 (0.011)	0.855 (0.035)	0.858 (0.033)	0.458 (0.012)	0.458 (0.01)	0.156 (0.034)	0.153 (0.033)	
70% MAR	0.607 (0.014)	0.607 (0.013)	0.863 (0.022)	0.866 (0.024)	0.499 (0.021)	0.502 (0.019)	0.167 (0.029)	0.169 (0.032)	
Base Case	0.918 (0.01)	0.729 (0.038)	0.984 (0.021)	0.984 (0.068)	0.896 (0.013)	0.54 (0.054)	0.593 (0.164)	0.035 (0.136)	Random Forest
N = 1750	0.904 (0.011)	0.76 (0.035)	0.972 (0.023)	0.983 (0.057)	0.902 (0.013)	0.6 (0.051)	0.635 (0.142)	0.076 (0.14)	
150 vars	0.947 (0.01)	0.758 (0.039)	0.994 (0.014)	0.992 (0.063)	0.929 (0.011)	0.555 (0.058)	0.659 (0.166)	0.018 (0.141)	
20 vars	0.754 (0.016)	0.625 (0.03)	0.895 (0.045)	0.896 (0.076)	0.73 (0.018)	0.493 (0.034)	0.425 (0.126)	0.142 (0.103)	
25% cat	0.889 (0.012)	0.726 (0.034)	0.969 (0.018)	0.978 (0.021)	0.864 (0.015)	0.478 (0.051)	0.19 (0.185)	0.062 (0.028)	
85% cat	0.809 (0.02)	0.678 (0.03)	0.973 (0.012)	0.978 (0.012)	0.668 (0.032)	0.386 (0.034)	0.035 (0.084)	0.027 (0.018)	
30% MCAR	0.993 (0.001)	0.89 (0.021)	1 (0.002)	1 (0.03)	0.97 (0.004)	0.617 (0.043)	0.518 (0.201)	0.009 (0.161)	
70% MCAR	1 (0)	0.964 (0.02)	1 (0)	1 (0.015)	0.997 (0.001)	0.543 (0.066)	0.009 (0.238)	0 (0.173)	
30% MAR	0.988 (0.002)	0.993 (0.003)	1 (0.004)	1 (0.002)	0.8 (0.019)	0.163 (0.047)	0.124 (0.16)	0 (0.066)	
70% MAR	1 (0)	1 (0)	1 (0)	1 (0)	0.012 (0.01)	0 (0)	0 (0.011)	0 (0)	
	(0.2,0.45)	(0.8,0.45)	(0.2,0.15)	(0.8,0.15)	(0.2,0.45)	(0.8,0.45)	(0.2,0.15)	(0.8,0.15)	

(Overlap, % Minority)



0.00 0.25 0.50 0.75 1.00

Figure A.1: Median sensitivity and specificity out of all oversampling methods. Median performance metrics was calculated over all 500 results for each data scenario, amount of imbalance and overlap, model, and oversampling method. Then the median was calculated again over the 23 oversampling methods. X-axis provides overlap and imbalance amounts. Y-axis indicates which data difficulty was present. The Base Case is $N = 750$, 100 variables, and no categorical or missing data. The median was calculated out of 23 medians which were themselves calculated out of 500 values.

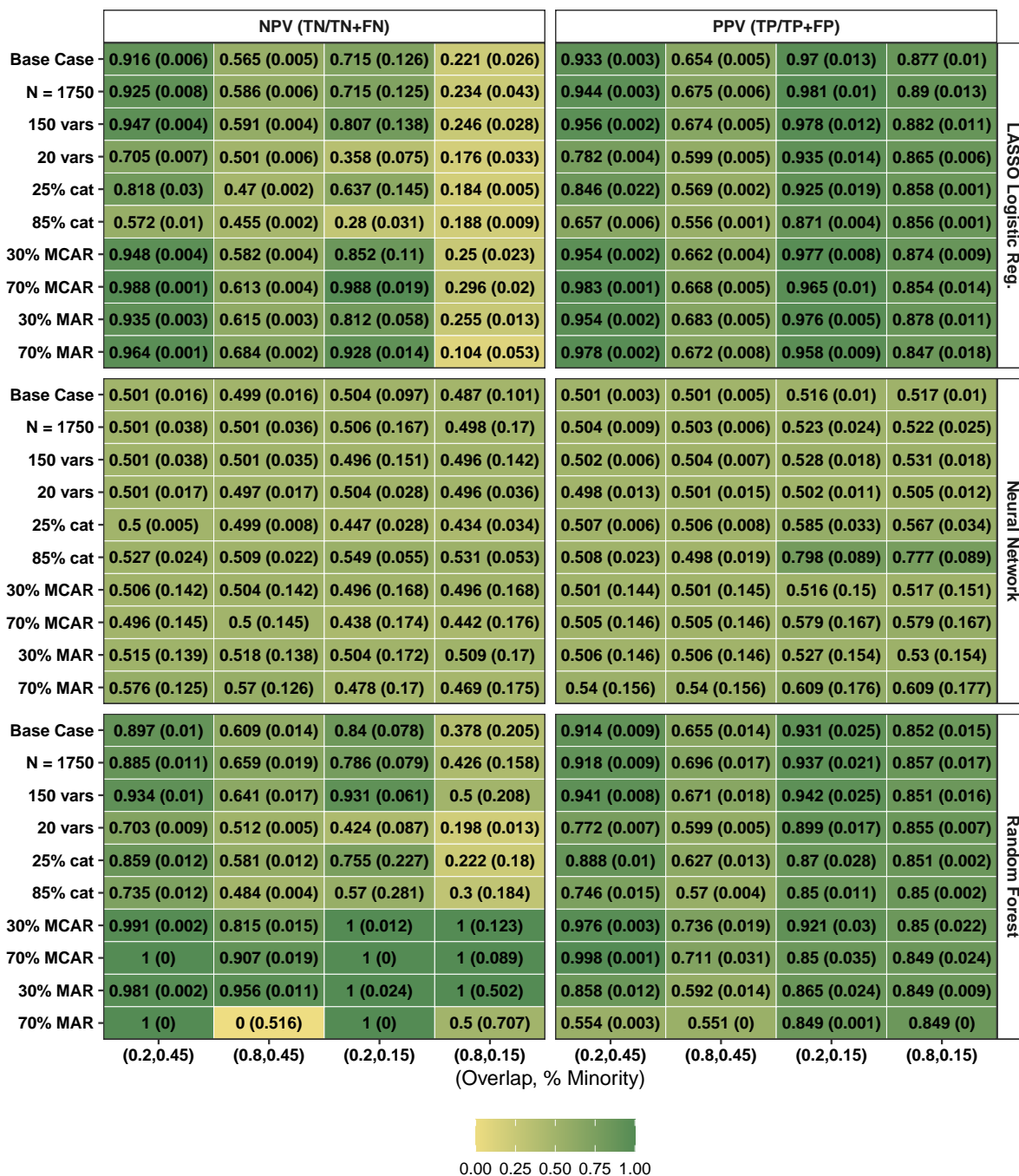
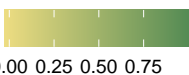


Figure A.2: Median negative predictive value (NPV) and positive predictive value (PPV) out of all oversampling methods. Median performance metrics was calculated over all 500 results for each data scenario, amount of imbalance and overlap, model, and oversampling method. Then the median was calculated again over the 23 oversampling methods. X-axis provides overlap and imbalance amounts. Y-axis indicates which data difficulty was present. The Base Case is $N = 750$, 100 variables, and no categorical or missing data. The median was calculated out of 23 medians which were themselves calculated out of 500 values.

	F1				Kappa				
Base Case	0.932 (0.004)	0.64 (0.006)	0.956 (0.03)	0.801 (0.033)	0.849 (0.008)	0.219 (0.007)	0.731 (0.117)	0.112 (0.028)	LASSO Logistic Reg.
N = 1750	0.941 (0.004)	0.657 (0.007)	0.959 (0.032)	0.795 (0.035)	0.87 (0.009)	0.261 (0.007)	0.76 (0.118)	0.14 (0.035)	
150 vars	0.956 (0.003)	0.663 (0.005)	0.971 (0.028)	0.824 (0.03)	0.903 (0.006)	0.264 (0.007)	0.82 (0.122)	0.143 (0.033)	
20 vars	0.764 (0.004)	0.577 (0.017)	0.854 (0.035)	0.752 (0.053)	0.488 (0.005)	0.099 (0.005)	0.337 (0.055)	0.047 (0.016)	
25% cat	0.846 (0.023)	0.559 (0.004)	0.933 (0.026)	0.833 (0.005)	0.656 (0.051)	0.038 (0.003)	0.532 (0.148)	0.045 (0.006)	
85% cat	0.649 (0.009)	0.55 (0.006)	0.876 (0.008)	0.848 (0.008)	0.226 (0.016)	0.012 (0.003)	0.146 (0.03)	0.044 (0.007)	
30% MCAR	0.956 (0.002)	0.657 (0.004)	0.976 (0.018)	0.849 (0.029)	0.903 (0.005)	0.244 (0.004)	0.833 (0.088)	0.132 (0.023)	
70% MCAR	0.985 (0.001)	0.687 (0.005)	0.98 (0.005)	0.911 (0.029)	0.968 (0.002)	0.278 (0.003)	0.855 (0.04)	0.054 (0.032)	
30% MAR	0.95 (0.002)	0.686 (0.003)	0.971 (0.008)	0.845 (0.022)	0.889 (0.004)	0.296 (0.005)	0.808 (0.04)	0.153 (0.029)	
70% MAR	0.973 (0.001)	0.732 (0.002)	0.973 (0.003)	0.893 (0.01)	0.941 (0.001)	0.331 (0.013)	0.799 (0.026)	-0.027 (0.074)	
Base Case	0.526 (0.006)	0.526 (0.006)	0.645 (0.012)	0.644 (0.013)	0.001 (0.02)	0.003 (0.019)	0.004 (0.053)	0.004 (0.056)	Neural Network
N = 1750	0.527 (0.013)	0.527 (0.012)	0.648 (0.029)	0.648 (0.029)	0.003 (0.041)	0.003 (0.04)	0.01 (0.094)	0.004 (0.094)	
150 vars	0.525 (0.013)	0.527 (0.011)	0.656 (0.022)	0.66 (0.021)	0.004 (0.042)	0.005 (0.039)	0.008 (0.083)	0.007 (0.078)	
20 vars	0.523 (0.006)	0.523 (0.007)	0.63 (0.008)	0.634 (0.01)	0.002 (0.009)	-0.002 (0.006)	0.003 (0.015)	0.001 (0.015)	
25% cat	0.529 (0.005)	0.528 (0.005)	0.695 (0.023)	0.684 (0.025)	0.005 (0.006)	0.001 (0.006)	0.014 (0.008)	0.011 (0.009)	
85% cat	0.533 (0.012)	0.524 (0.012)	0.845 (0.066)	0.835 (0.067)	0.025 (0.009)	0.003 (0.012)	0.245 (0.078)	0.213 (0.073)	
30% MCAR	0.526 (0.152)	0.526 (0.152)	0.642 (0.188)	0.643 (0.188)	0.003 (0.015)	0.004 (0.015)	0 (0.061)	0 (0.06)	
70% MCAR	0.528 (0.152)	0.527 (0.152)	0.689 (0.199)	0.69 (0.199)	0.004 (0.009)	0 (0.01)	0 (0.044)	0 (0.045)	
30% MAR	0.531 (0.154)	0.534 (0.154)	0.654 (0.192)	0.657 (0.191)	0.017 (0.027)	0.017 (0.021)	0.011 (0.068)	0.005 (0.065)	
70% MAR	0.571 (0.166)	0.568 (0.165)	0.717 (0.207)	0.715 (0.208)	0.1 (0.042)	0.105 (0.04)	0.033 (0.055)	0.034 (0.061)	
Base Case	0.913 (0.001)	0.687 (0.01)	0.952 (0.008)	0.915 (0.031)	0.807 (0.004)	0.262 (0.016)	0.637 (0.112)	0.036 (0.047)	Random Forest
N = 1750	0.909 (0.002)	0.719 (0.008)	0.952 (0.007)	0.915 (0.023)	0.799 (0.003)	0.356 (0.02)	0.648 (0.092)	0.09 (0.057)	
150 vars	0.941 (0.001)	0.707 (0.008)	0.961 (0.009)	0.918 (0.026)	0.868 (0.003)	0.304 (0.02)	0.708 (0.121)	0.019 (0.058)	
20 vars	0.759 (0.004)	0.609 (0.012)	0.897 (0.017)	0.874 (0.041)	0.475 (0.003)	0.111 (0.006)	0.323 (0.054)	0.043 (0.018)	
25% cat	0.889 (0.004)	0.672 (0.009)	0.926 (0.018)	0.913 (0.009)	0.753 (0.009)	0.199 (0.02)	0.19 (0.204)	0.018 (0.024)	
85% cat	0.772 (0.004)	0.617 (0.011)	0.919 (0.007)	0.912 (0.005)	0.477 (0.013)	0.05 (0.007)	0.012 (0.105)	0.003 (0.018)	
30% MCAR	0.983 (0.001)	0.801 (0.002)	0.959 (0.016)	0.919 (0.004)	0.962 (0.002)	0.5 (0.019)	0.646 (0.191)	0.012 (0.101)	
70% MCAR	0.999 (0)	0.798 (0.007)	0.919 (0.019)	0.919 (0.002)	0.997 (0.001)	0.469 (0.031)	0.015 (0.248)	0 (0.117)	
30% MAR	0.918 (0.006)	0.742 (0.01)	0.928 (0.011)	0.919 (0.004)	0.8 (0.016)	0.167 (0.048)	0.194 (0.156)	0 (0.086)	
70% MAR	0.713 (0.002)	0.71 (0)	0.919 (0.001)	0.919 (0)	0.013 (0.011)	0 (0)	0 (0.018)	0 (0)	

(0.2,0.45) (0.8,0.45) (0.2,0.15) (0.8,0.15) (0.2,0.45) (0.8,0.45) (0.2,0.15) (0.8,0.15)
(Overlap, % Minority)



0.00 0.25 0.50 0.75

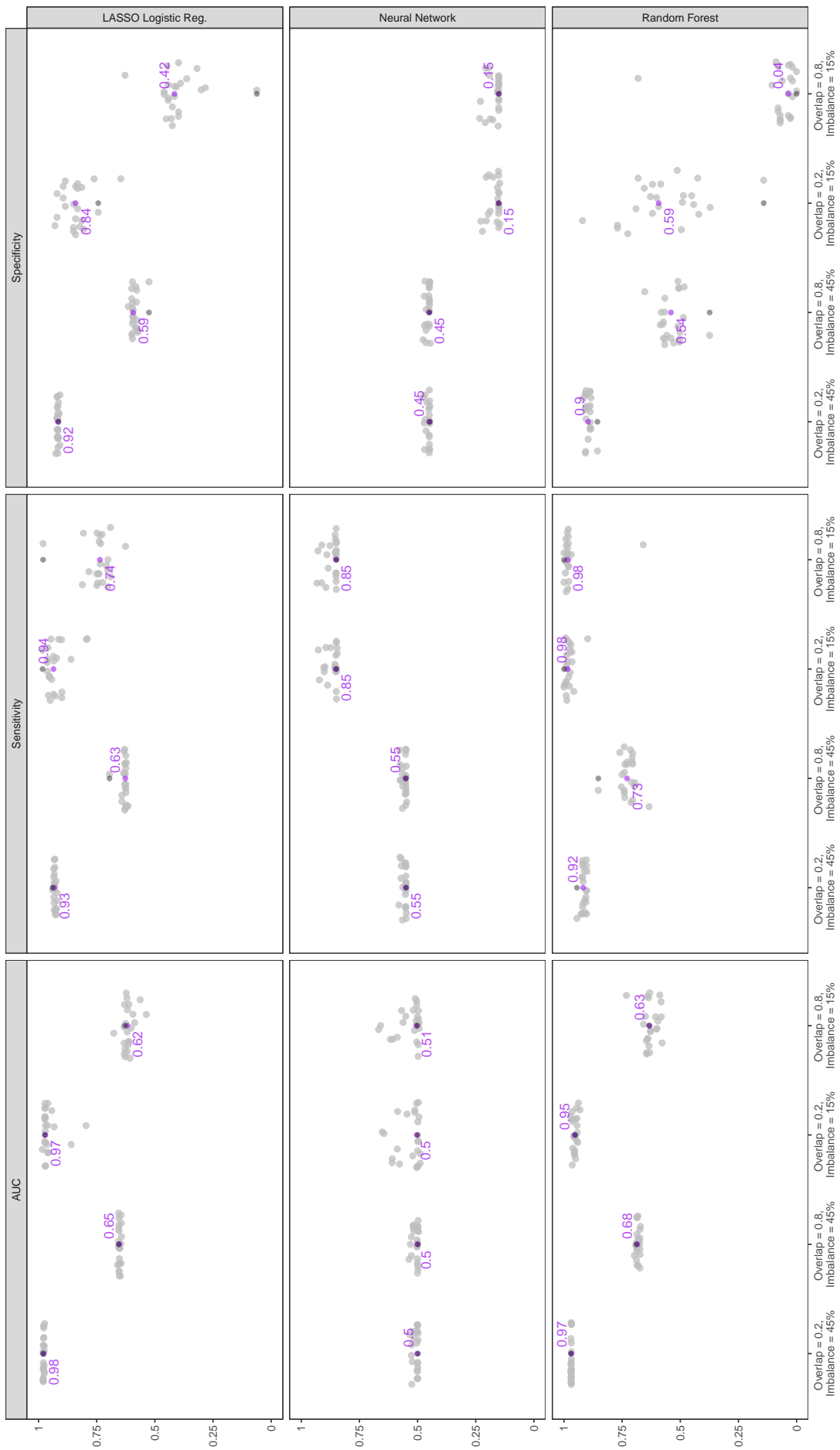
Figure A.3: Median F1 score and Kappa coefficient out of all oversampling methods. Median performance metrics was calculated over all 500 results for each data scenario, amount of imbalance and overlap, model, and oversampling method. Then the median was calculated again over the 23 oversampling methods. X-axis provides overlap and imbalance amounts. Y-axis indicates which data difficulty was present. The Base Case is N = 750, 100 variables, and no categorical or missing data. The median was calculated out of 23 medians which were themselves calculated out of 500 values.

Appendix B: More Detailed Supplementary Visualizations of SMOTE

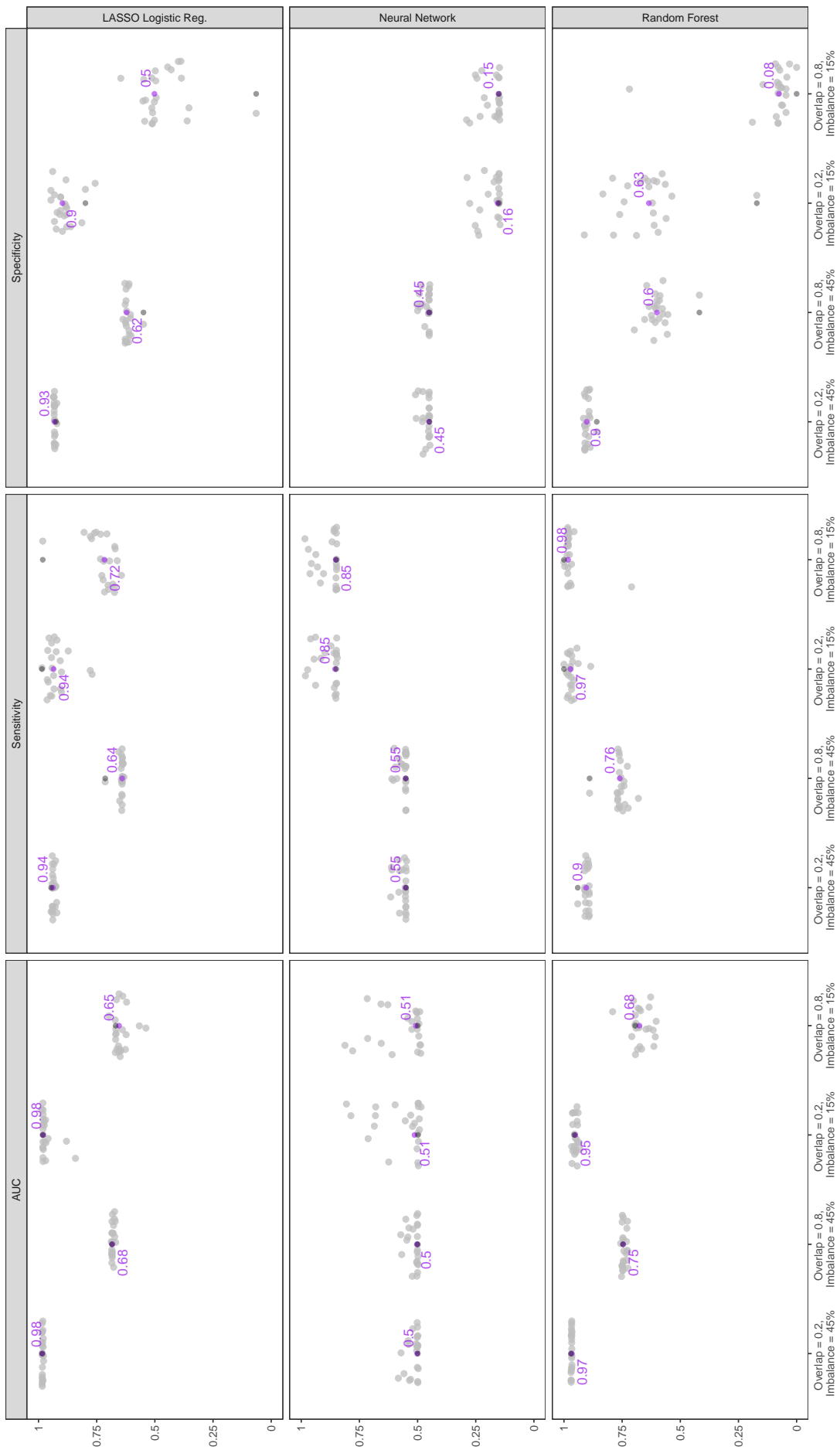
Simulation Results

Figure B.1: Median Performance Metrics: Gray points are median specificity taken over 500 predictive performance results. The characteristics of each training set are given in the title at the top of each plot. Each page corresponds to a different set of data characteristics. The balancing methods with the three largest averages are given at the top of each panel as text. The rank was calculated with respect to the metric, model, and data difficulty scenario. The purple point gives the median for each group of points and it was also calculated with respect to these elements. Dark gray point mark the results for unbalanced data. Viewers of these and other very large and detailed plots in the Appendices may need to use the zoom function of their PDF viewer for a better inspection of the results.

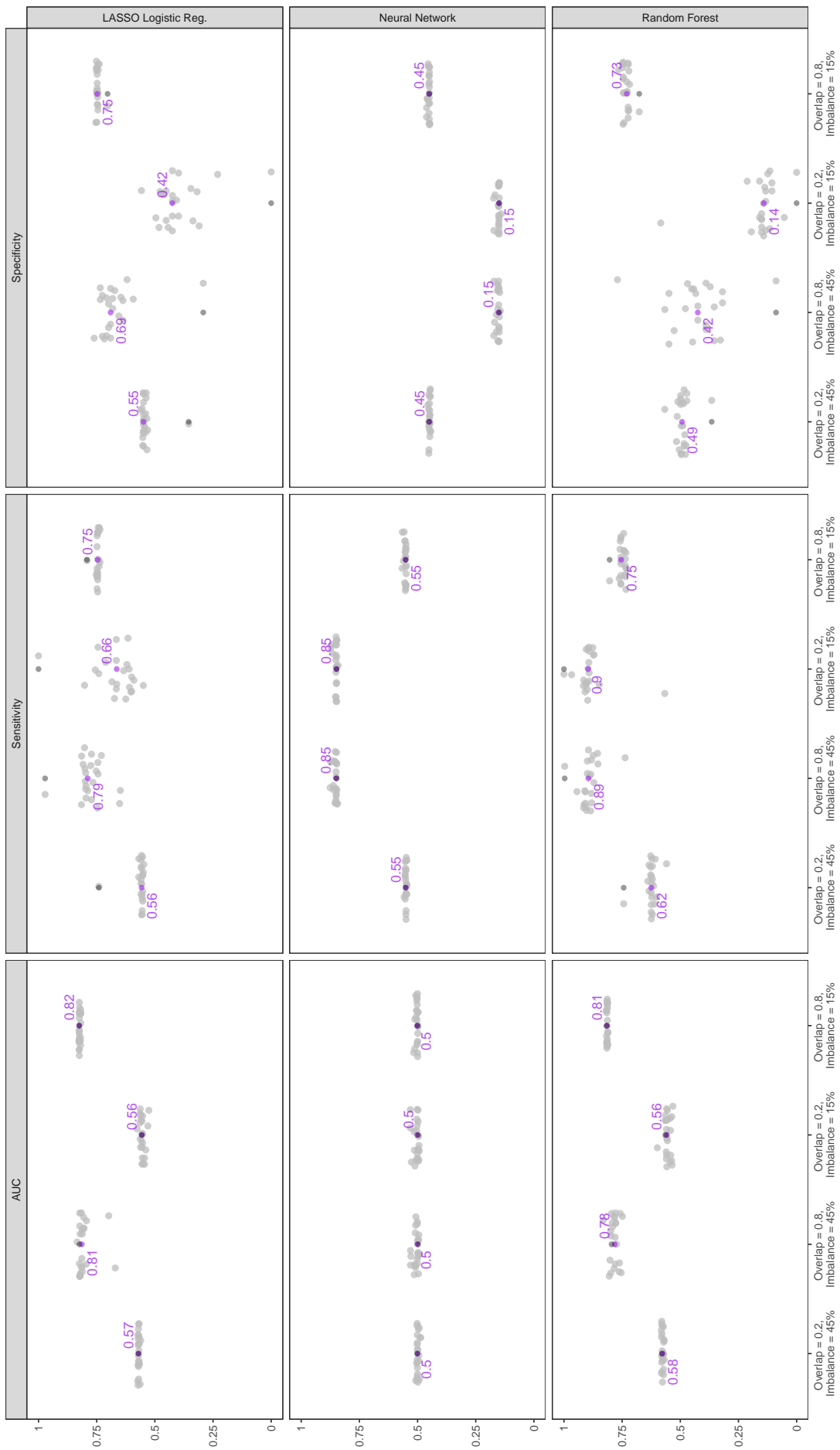
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 100 variables of which 0% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point marks the median for each group of points. The dark gray point marks results on unbalanced data.



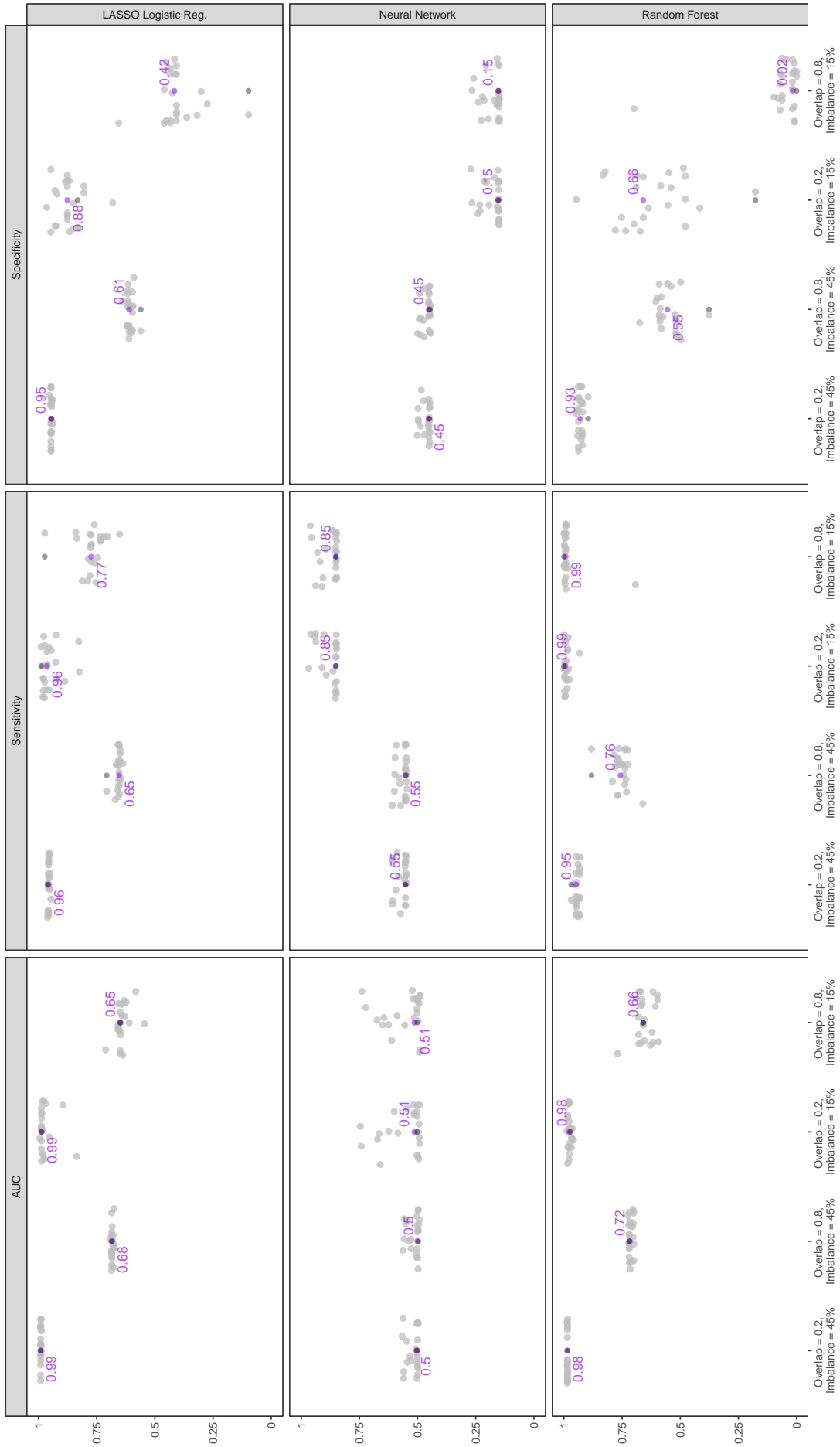
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 1750 samples, 100 variables of which 0% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



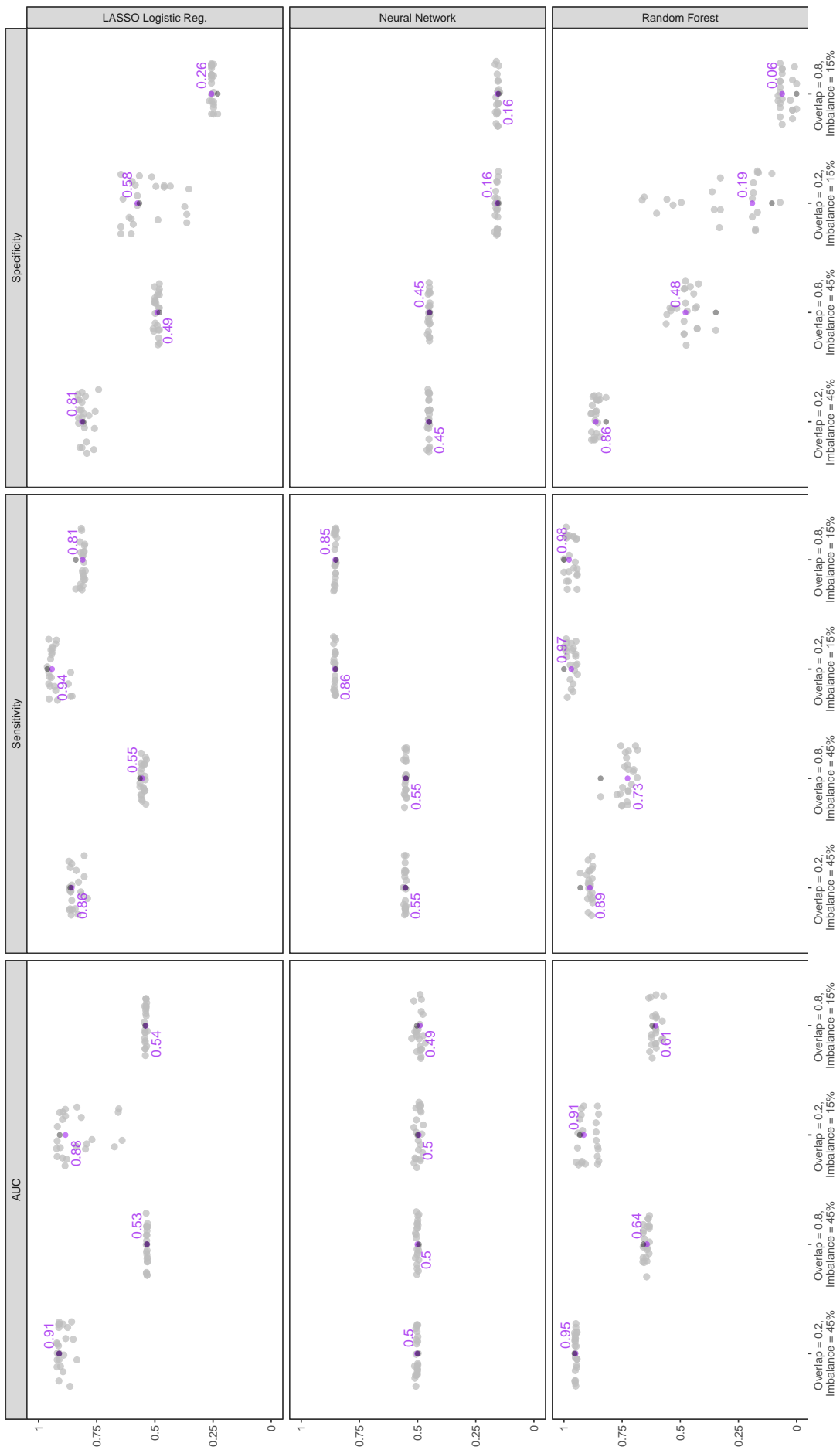
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples. 20 variables of which 0% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



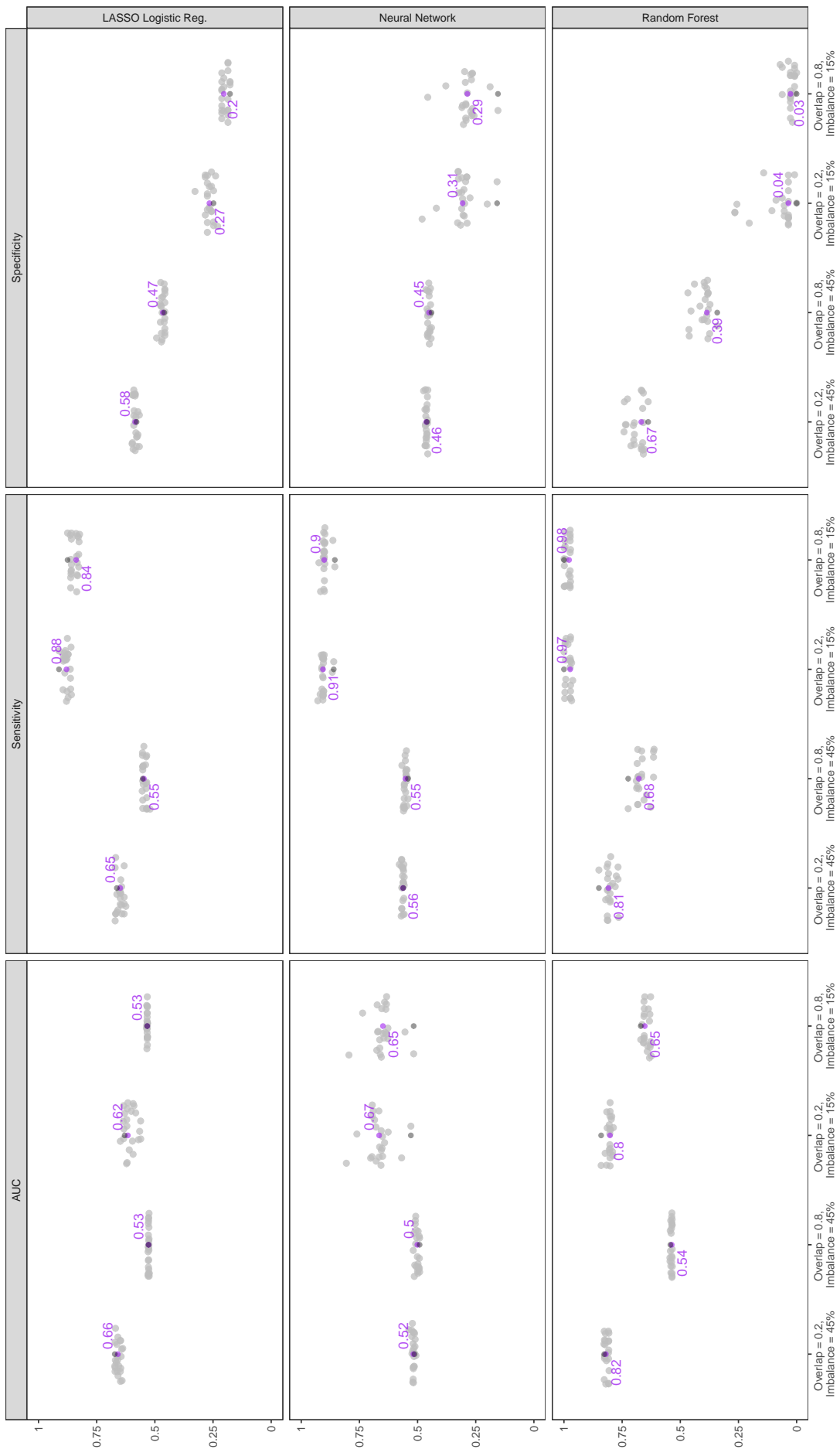
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 150 variables of which 0% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point marks the median for each group of points. The dark gray point marks results on unbalanced data.



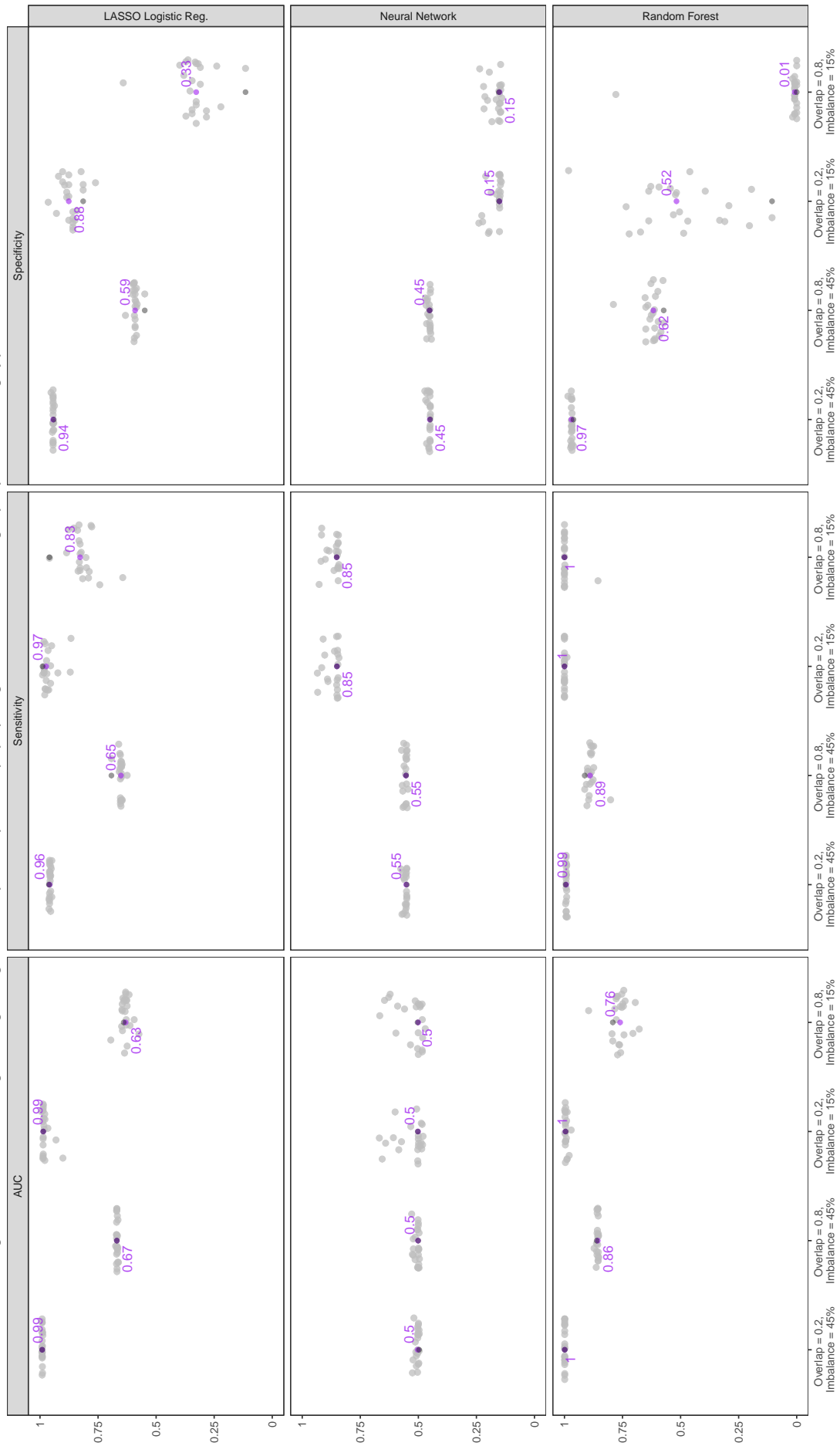
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples. 100 variables of which 25% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



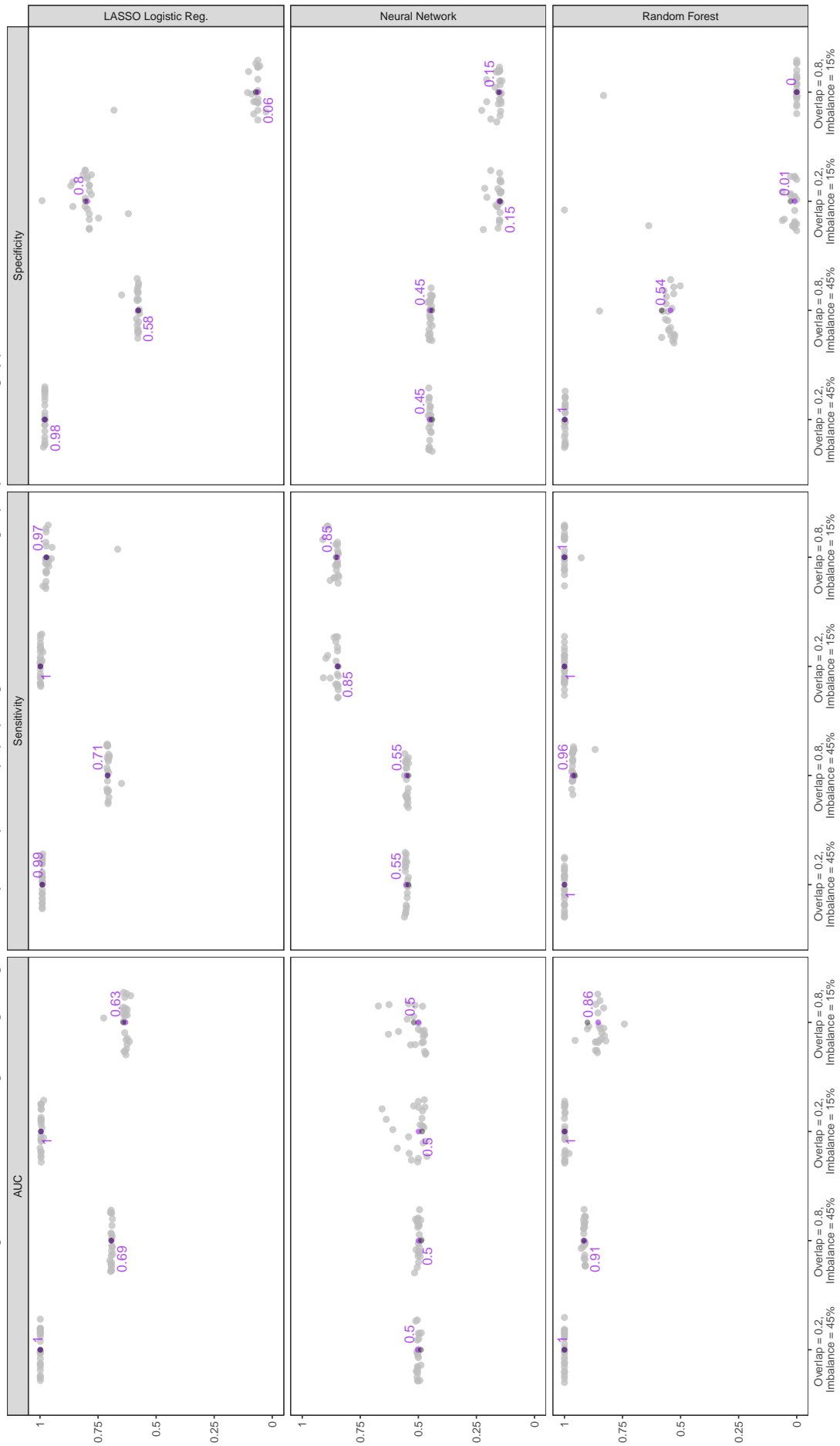
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples. 100 variables of which 85% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point marks the median for each group of points. The dark gray point marks results on unbalanced data.



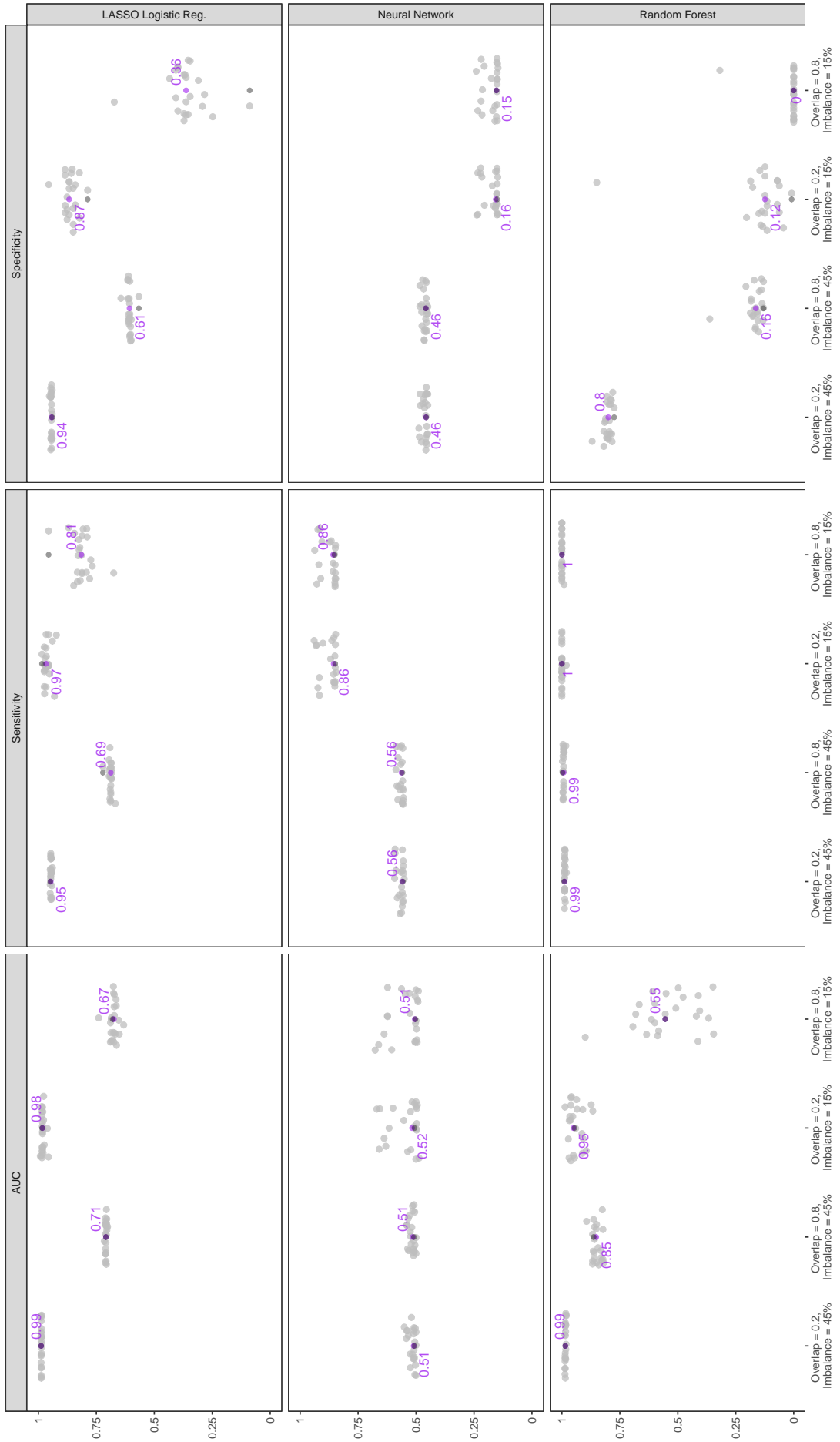
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples. 100 variables of which 0% were categorical, and 30% data missing completely at random. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



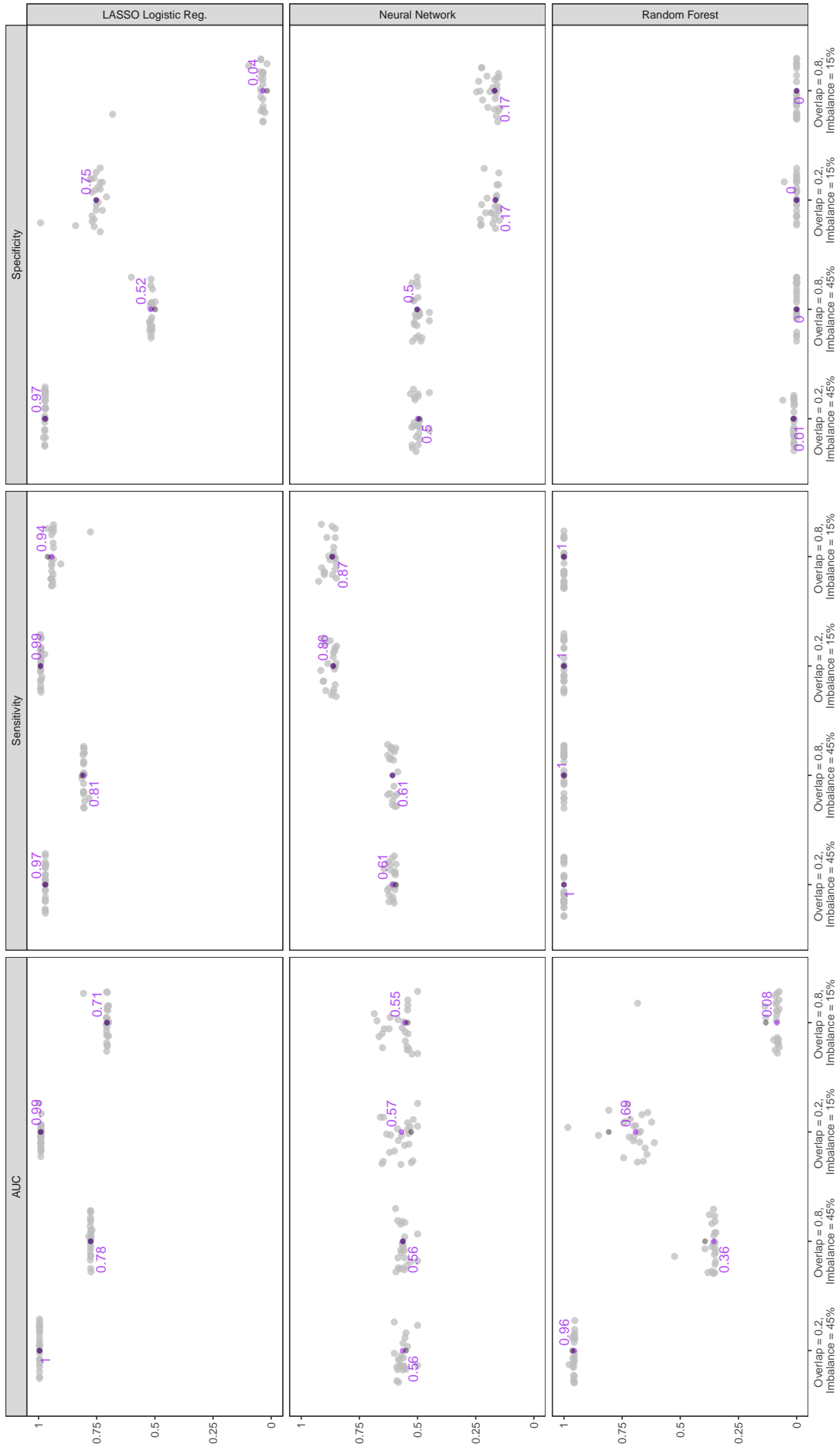
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples. 100 variables of which 0% were categorical, and 70% data missing completely at random. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



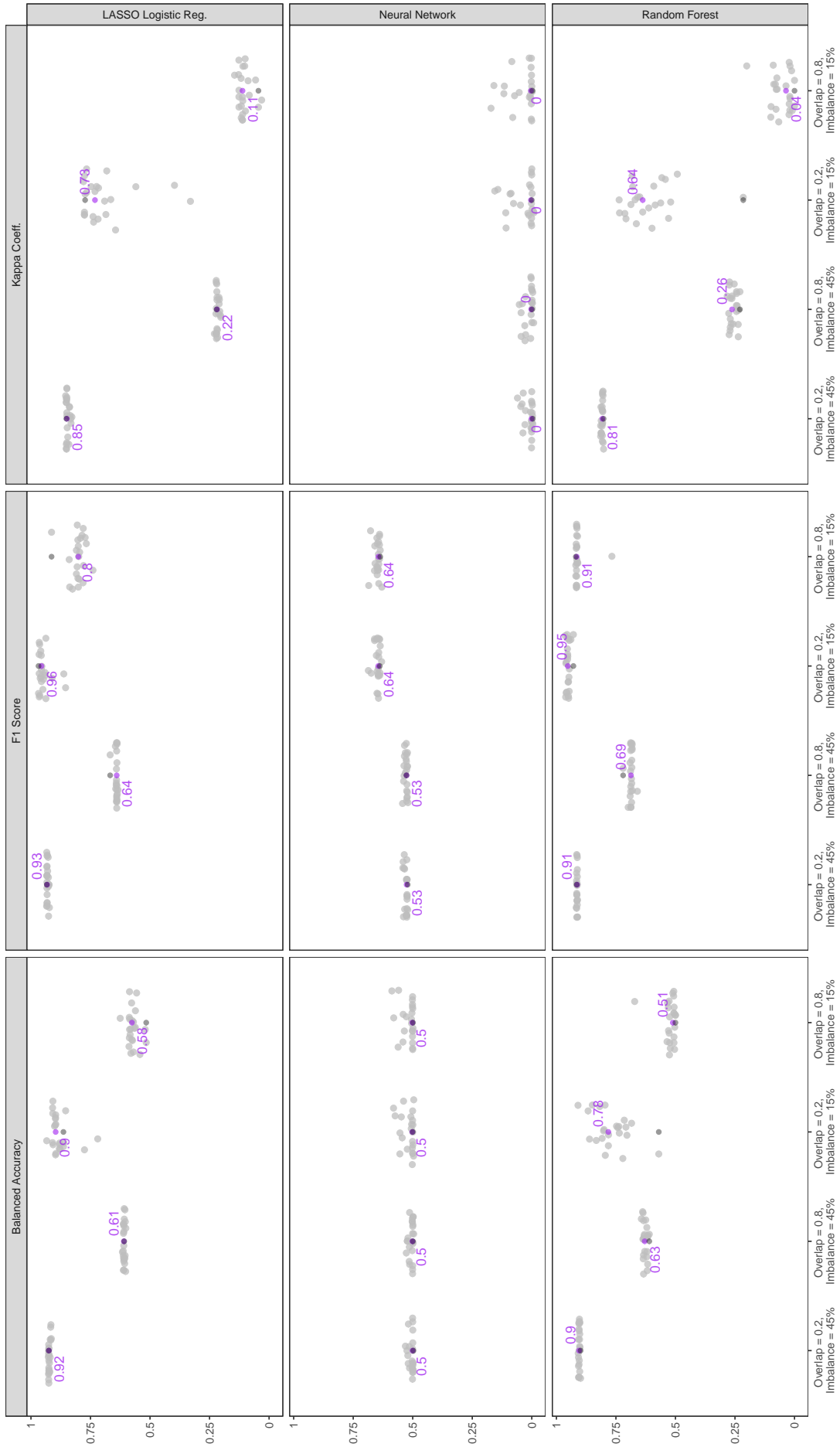
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 100 variables of which 0% were categorical, and 30% minority data missing at random. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



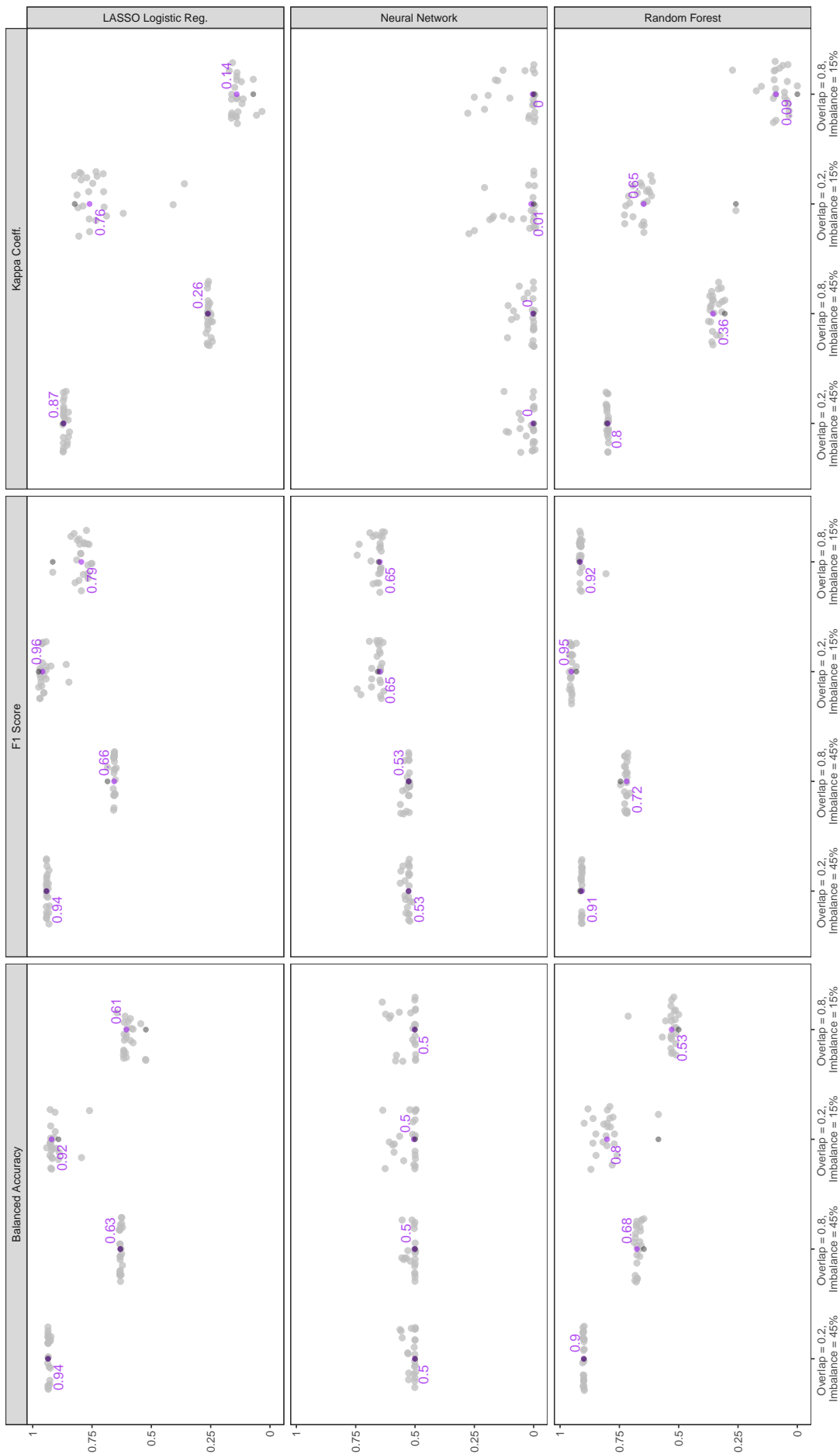
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 100 variables of which 0% were categorical, and 70% minority data missing at random. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



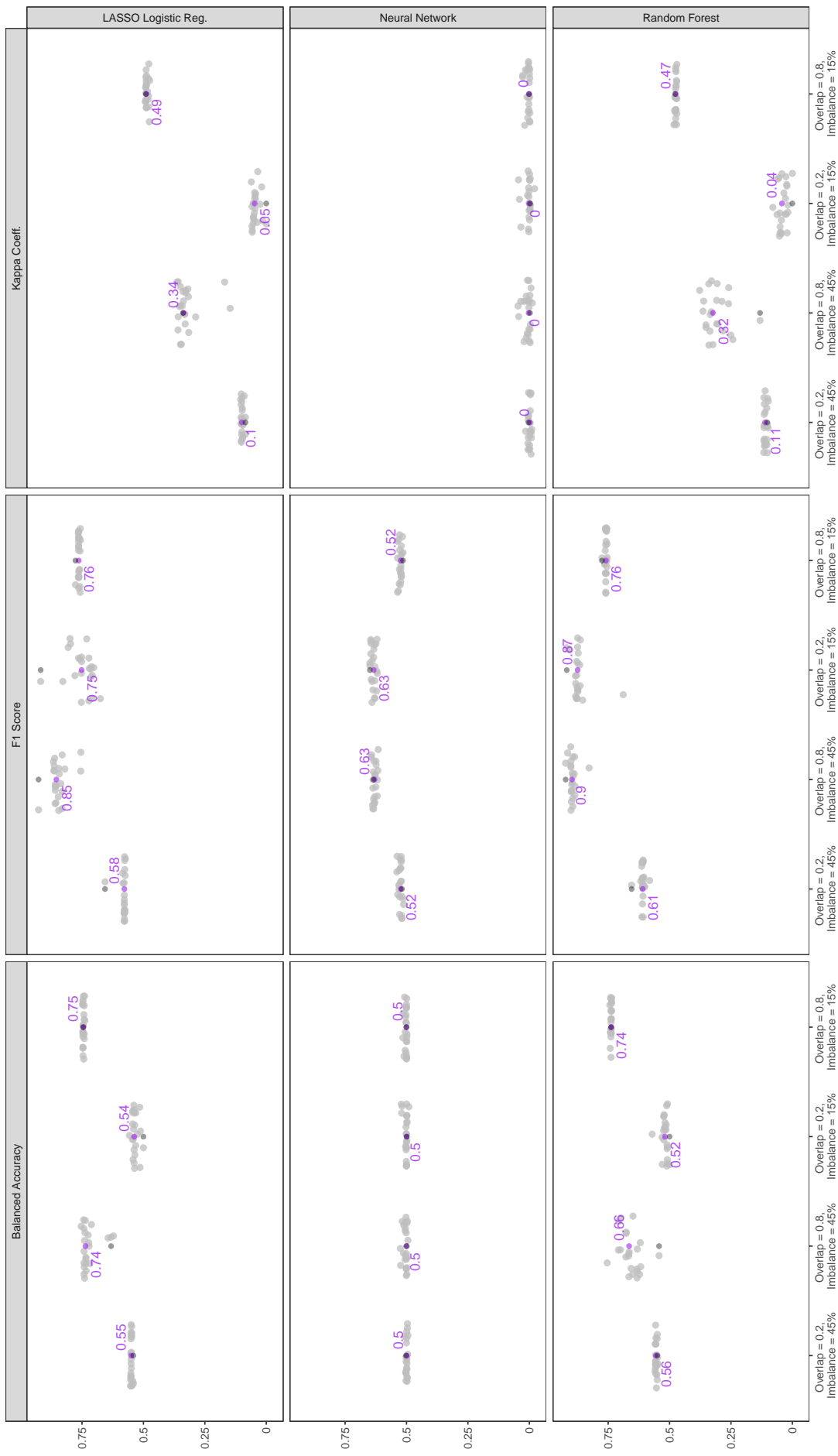
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 100 variables of which 0% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point marks the median for each group of points. The dark gray point marks results on unbalanced data.



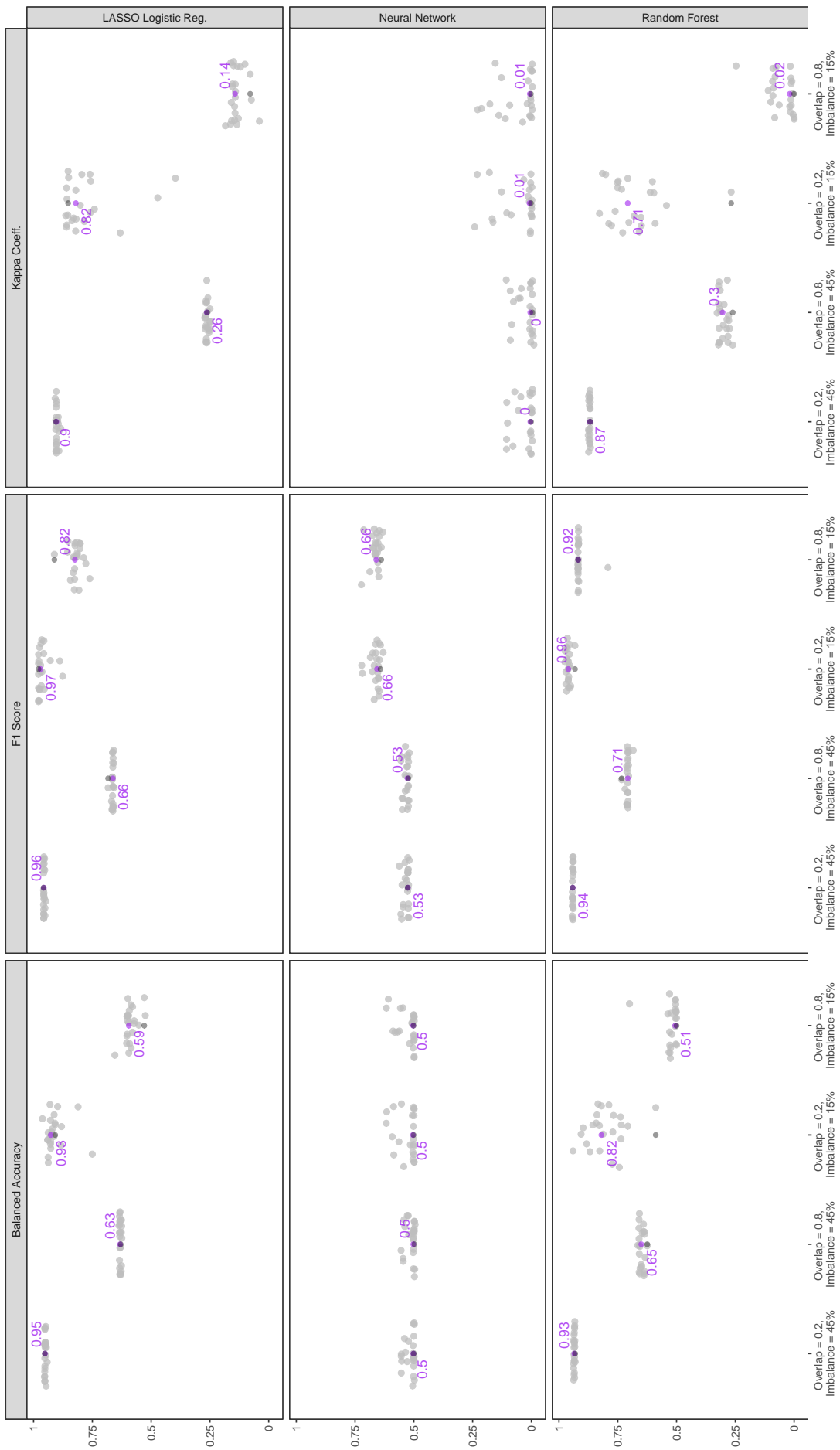
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 1750 samples, 100 variables of which 0% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



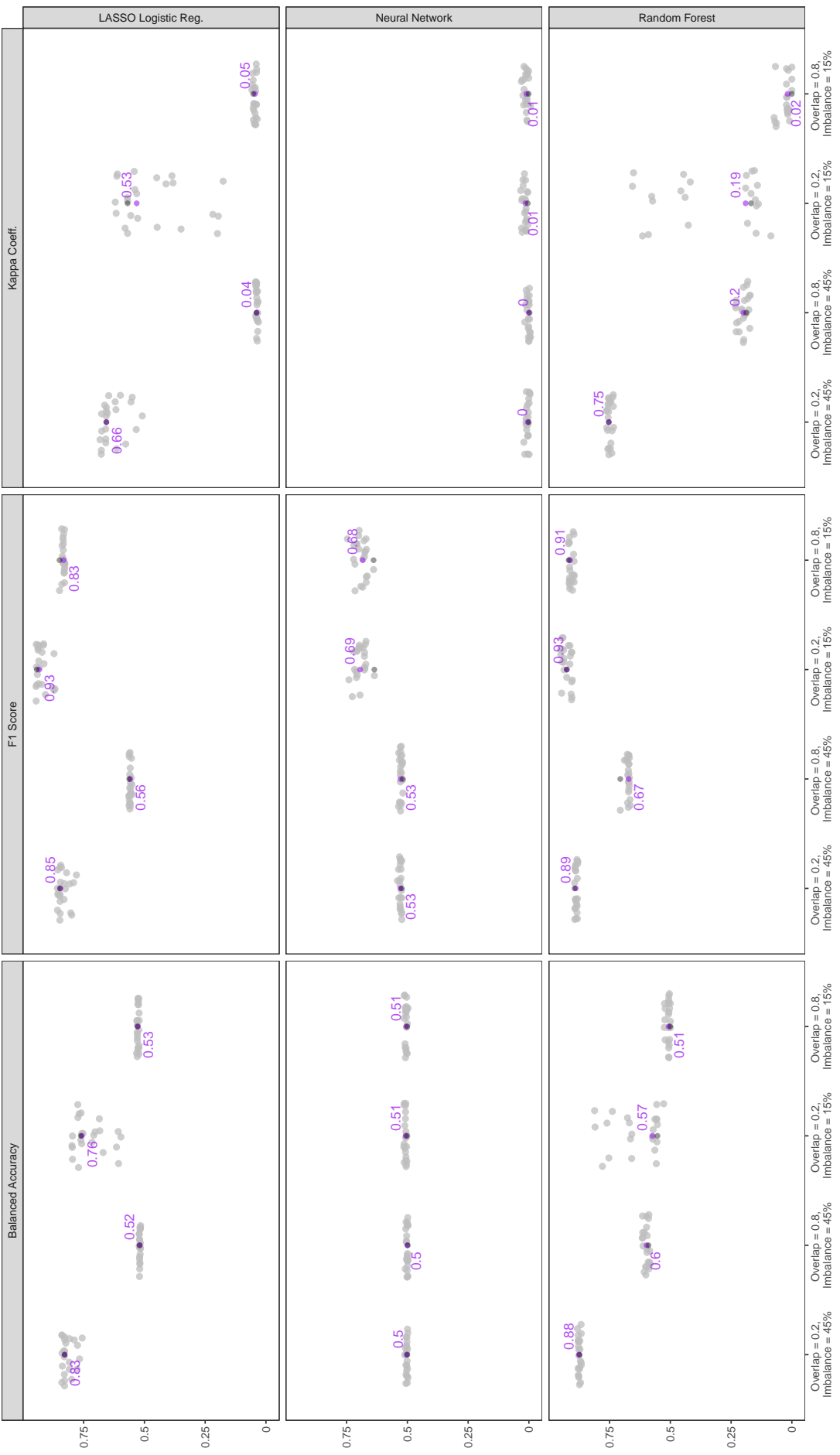
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples. 20 variables of which 0% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



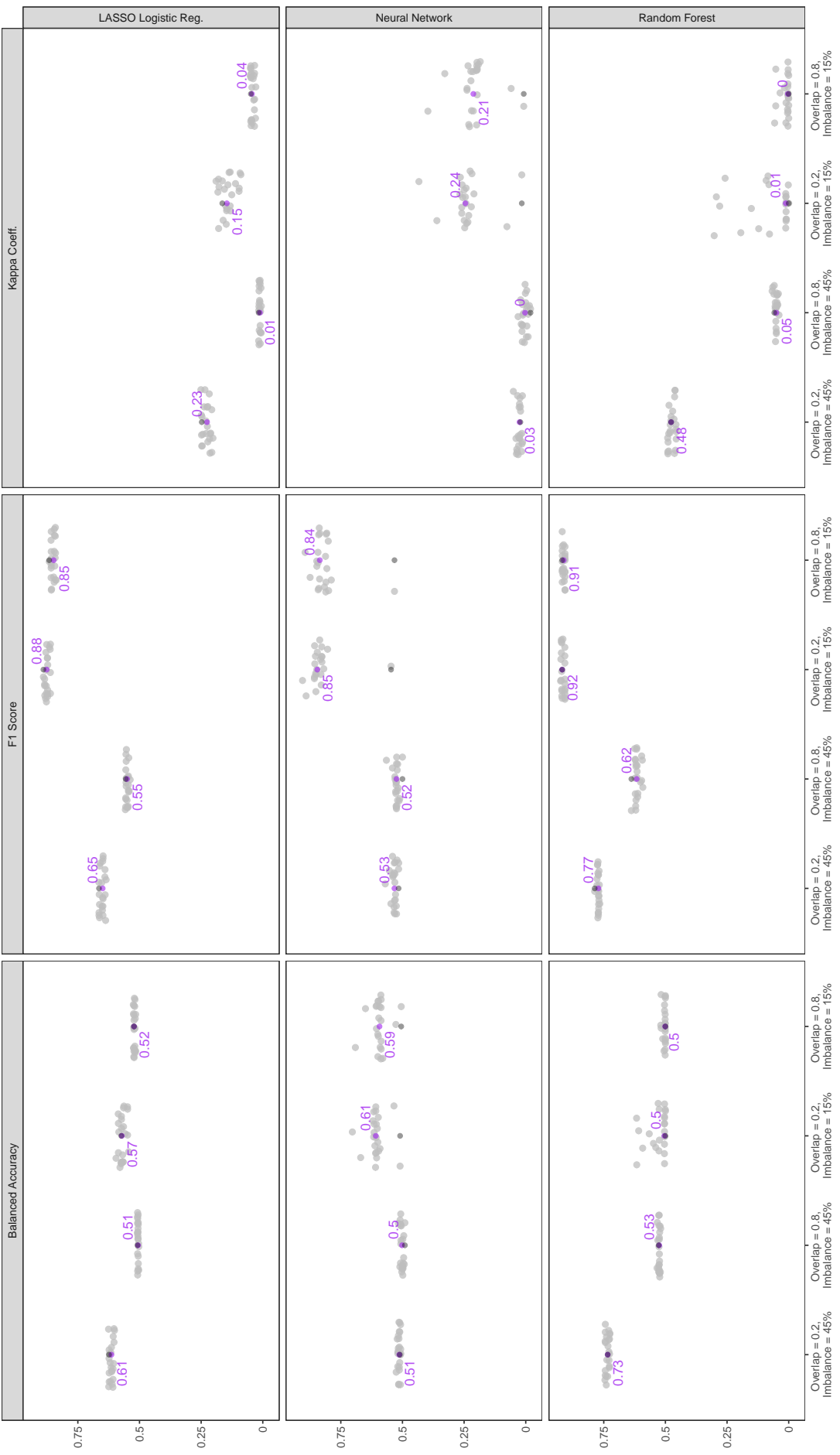
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 150 variables of which 0% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point marks the median for each group of points. The dark gray point marks results on unbalanced data.



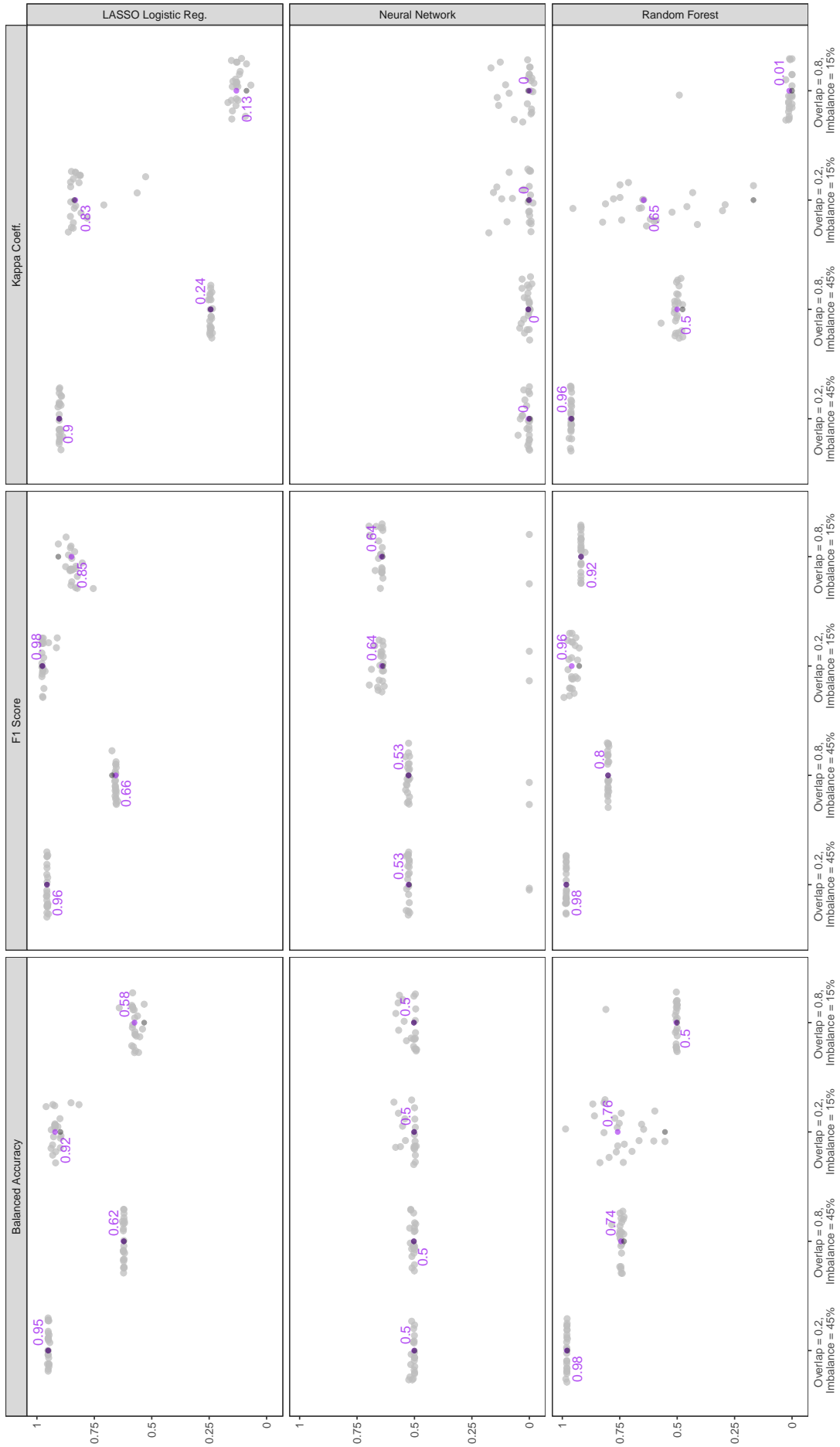
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples. 100 variables of which 25% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



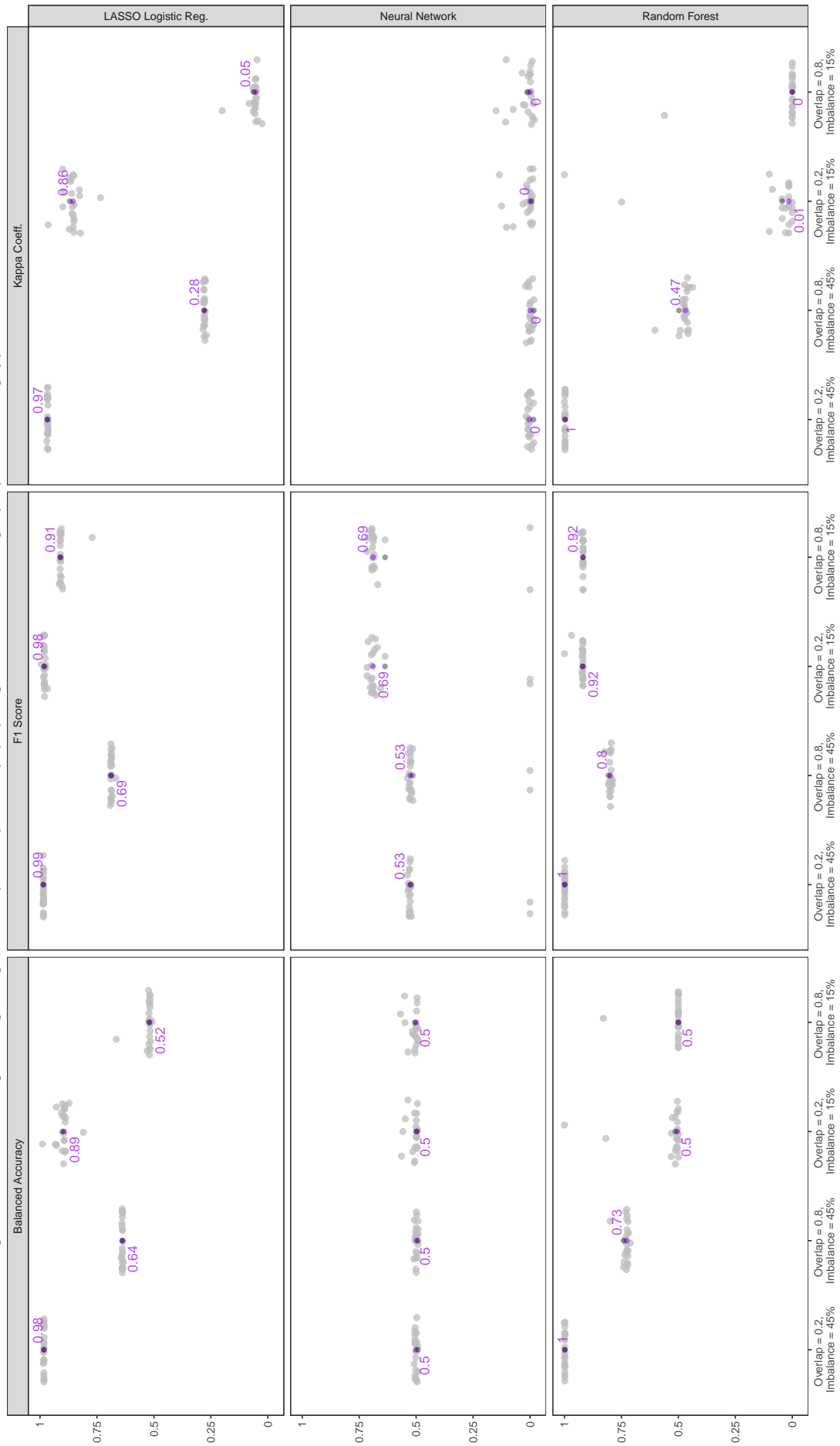
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples. 100 variables of which 85% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



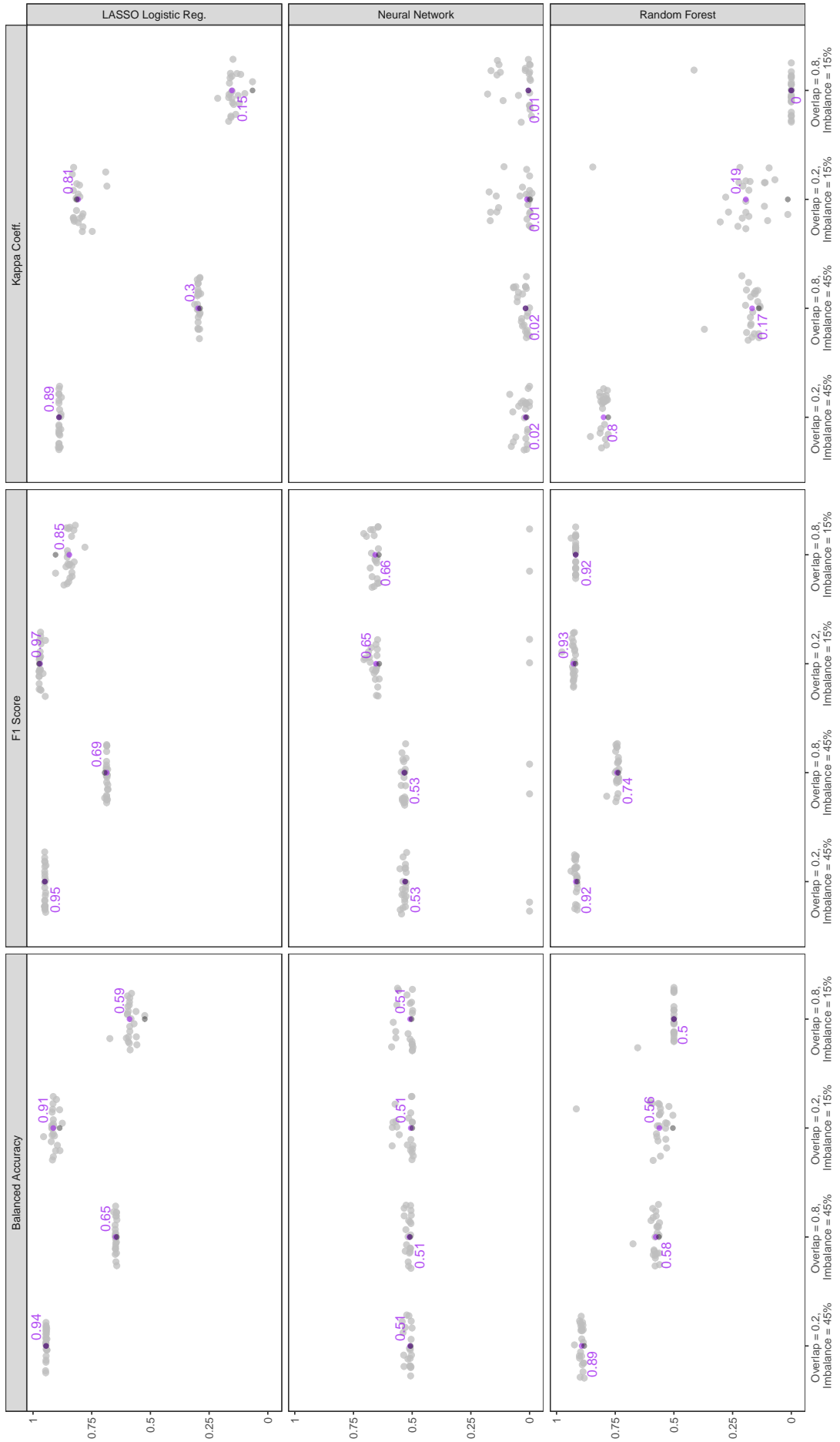
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples. 100 variables of which 0% were categorical, and 30% data missing completely at random. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



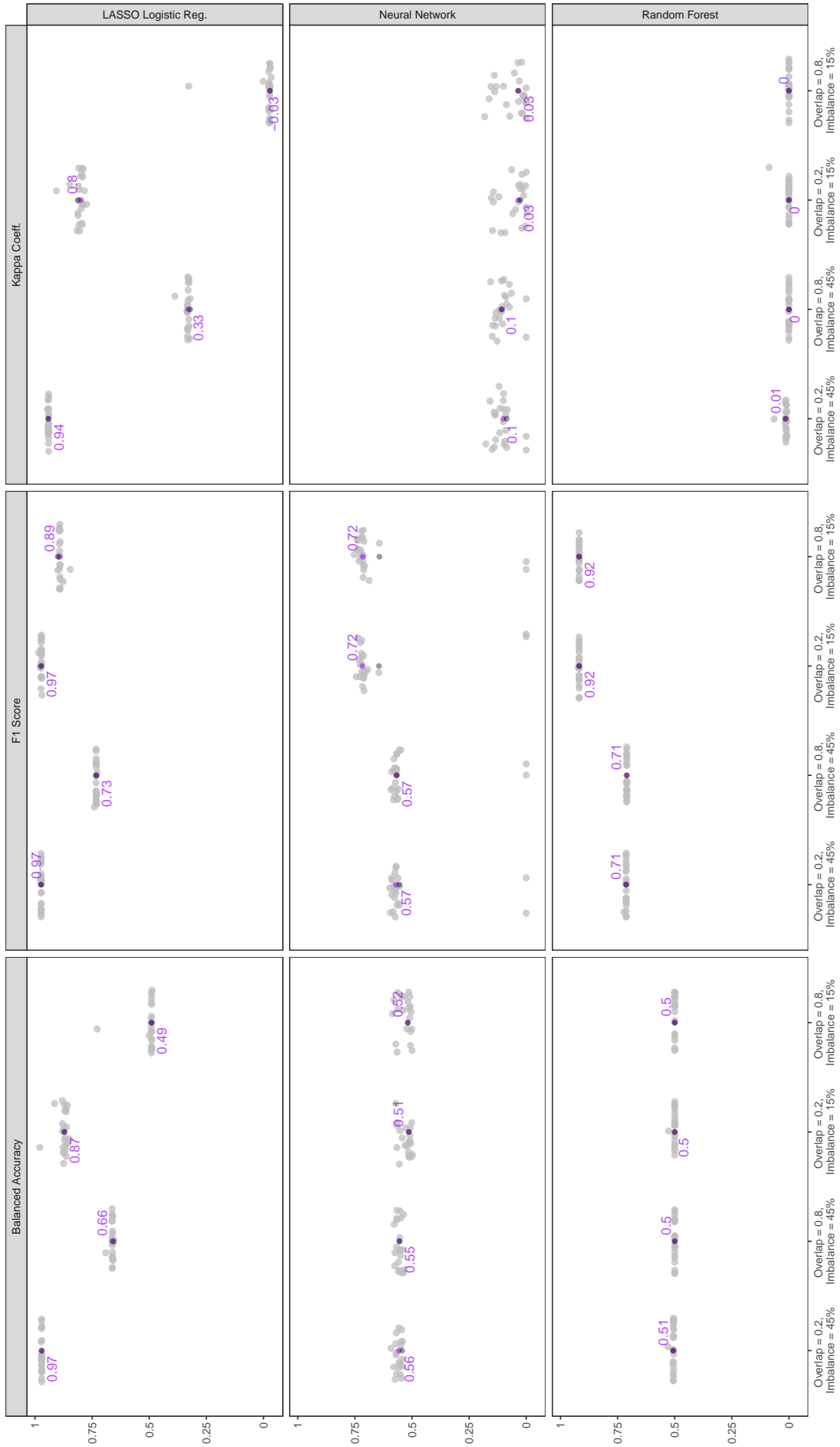
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 100 variables of which 0% were categorical, and 70% data missing completely at random. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



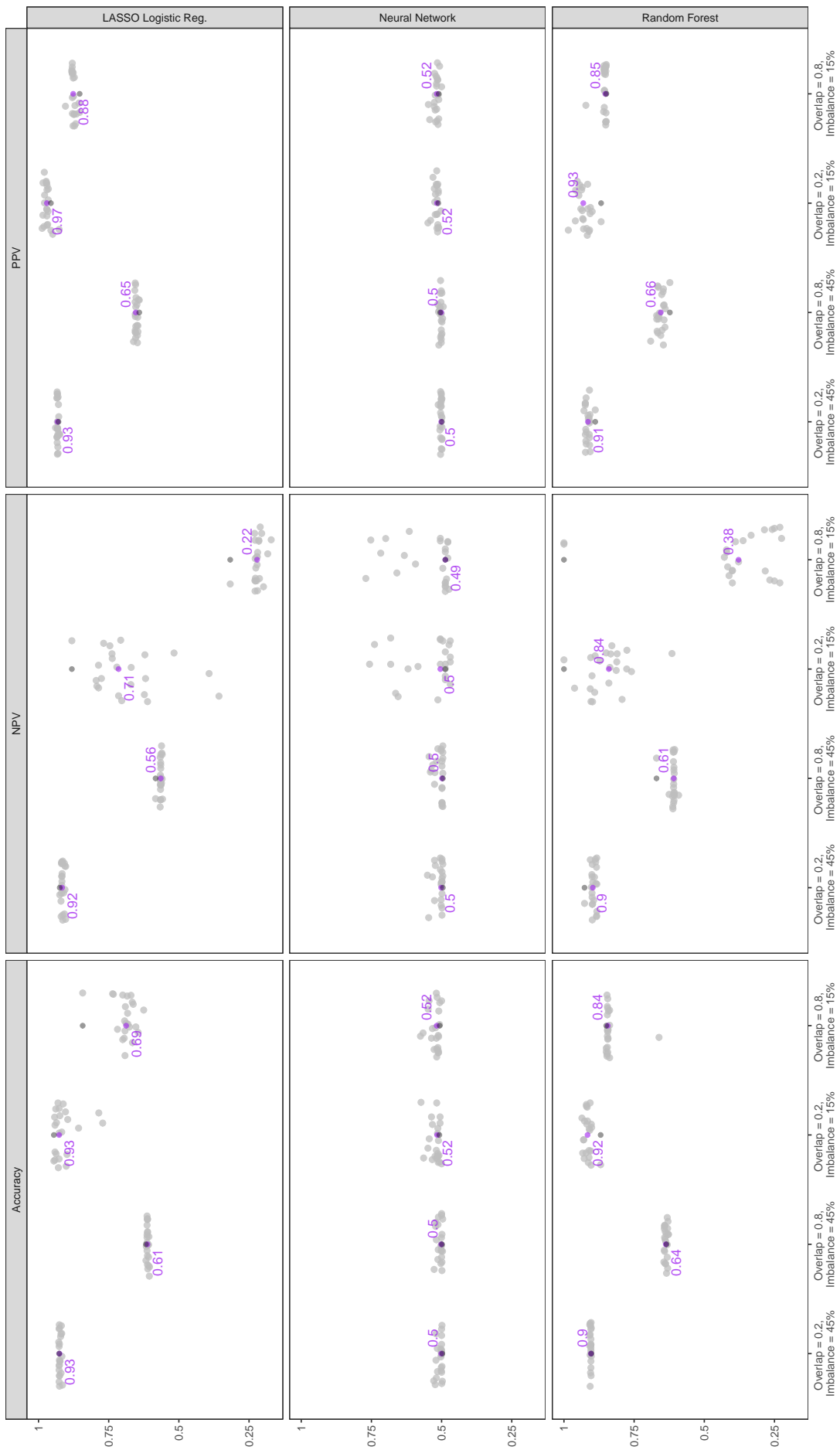
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 100 variables of which 0% were categorical, and 30% minority data missing at random. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



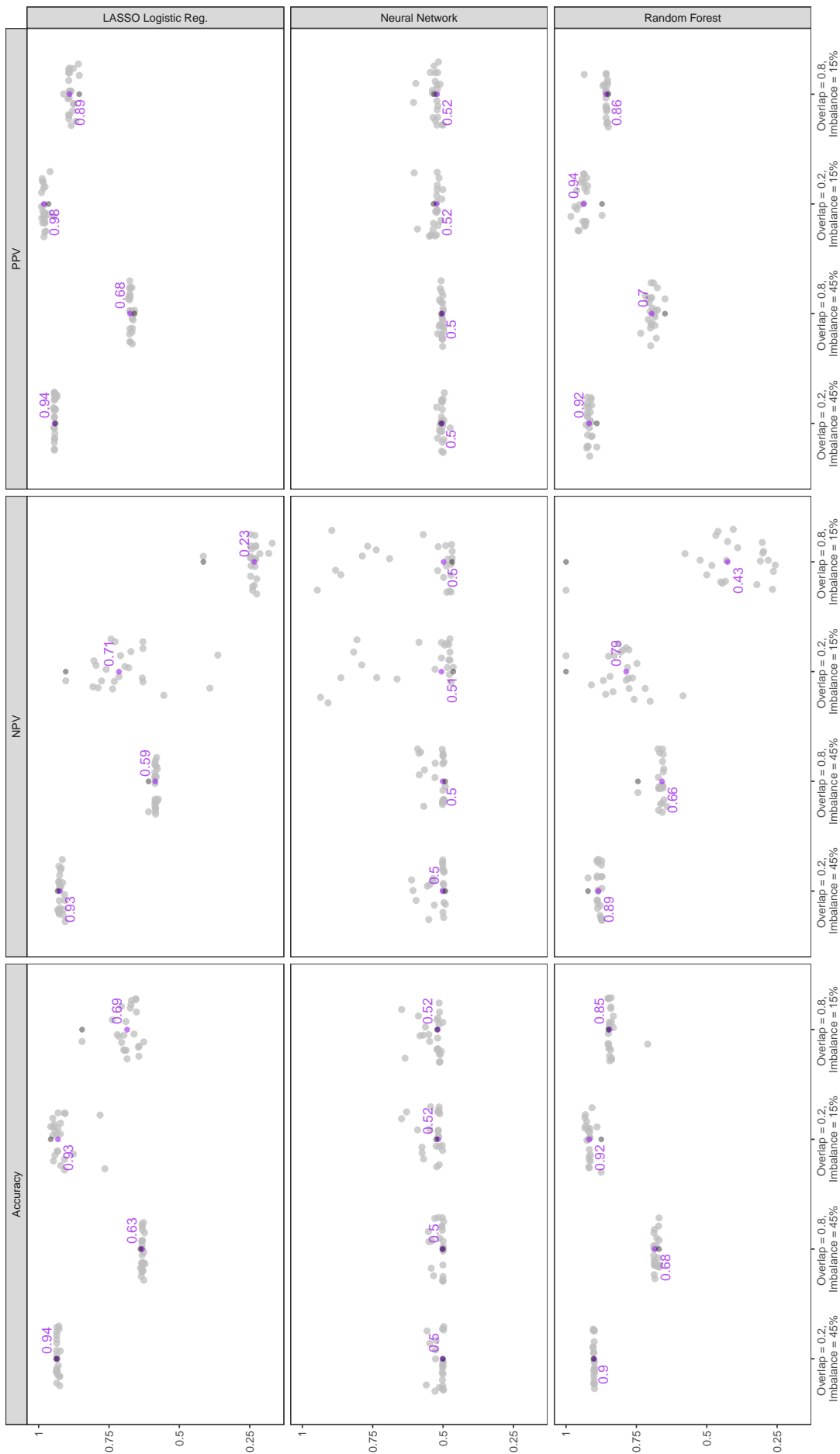
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 100 variables of which 0% were categorical, and 70% minority data missing at random. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



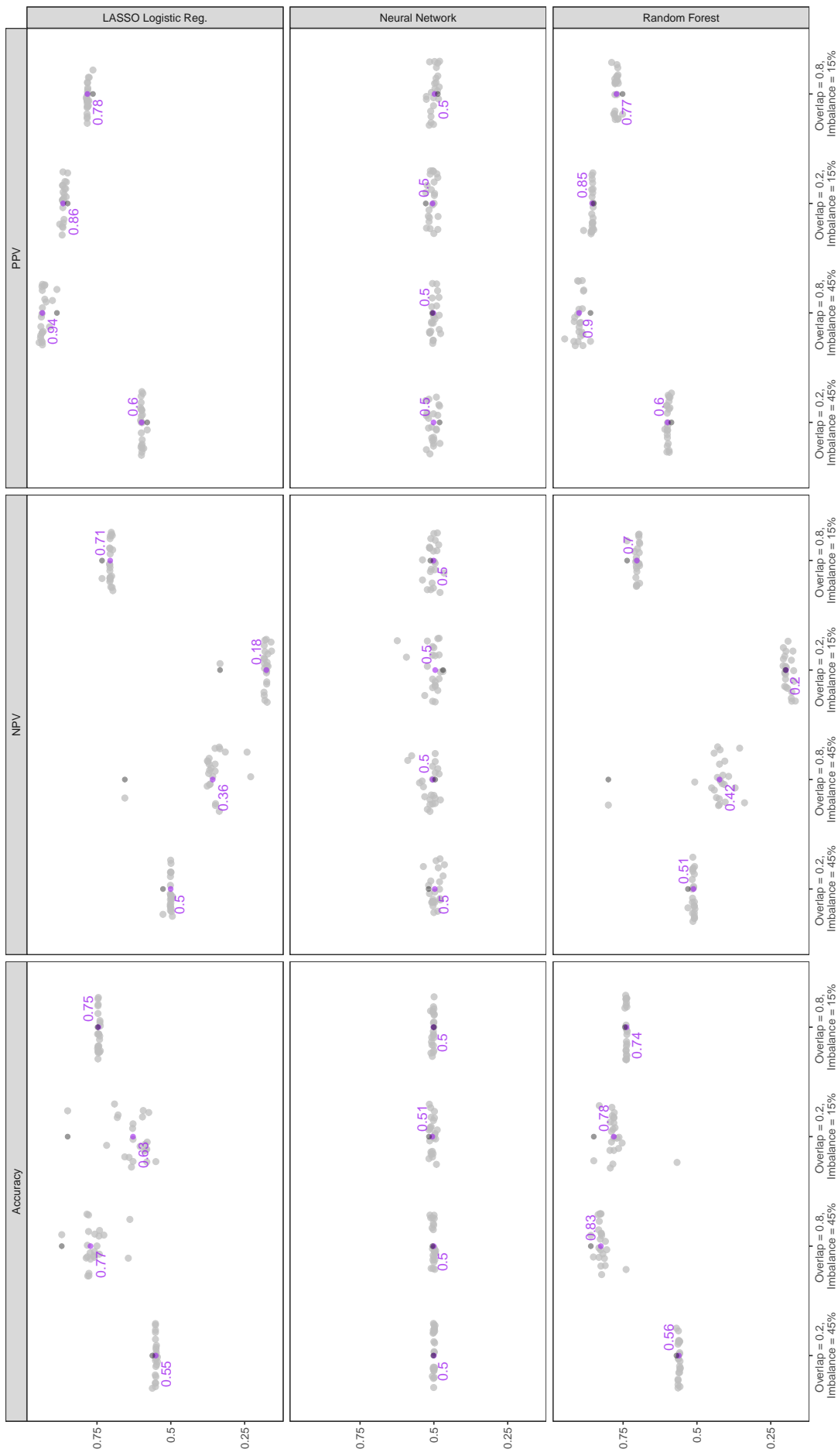
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 100 variables of which 0% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



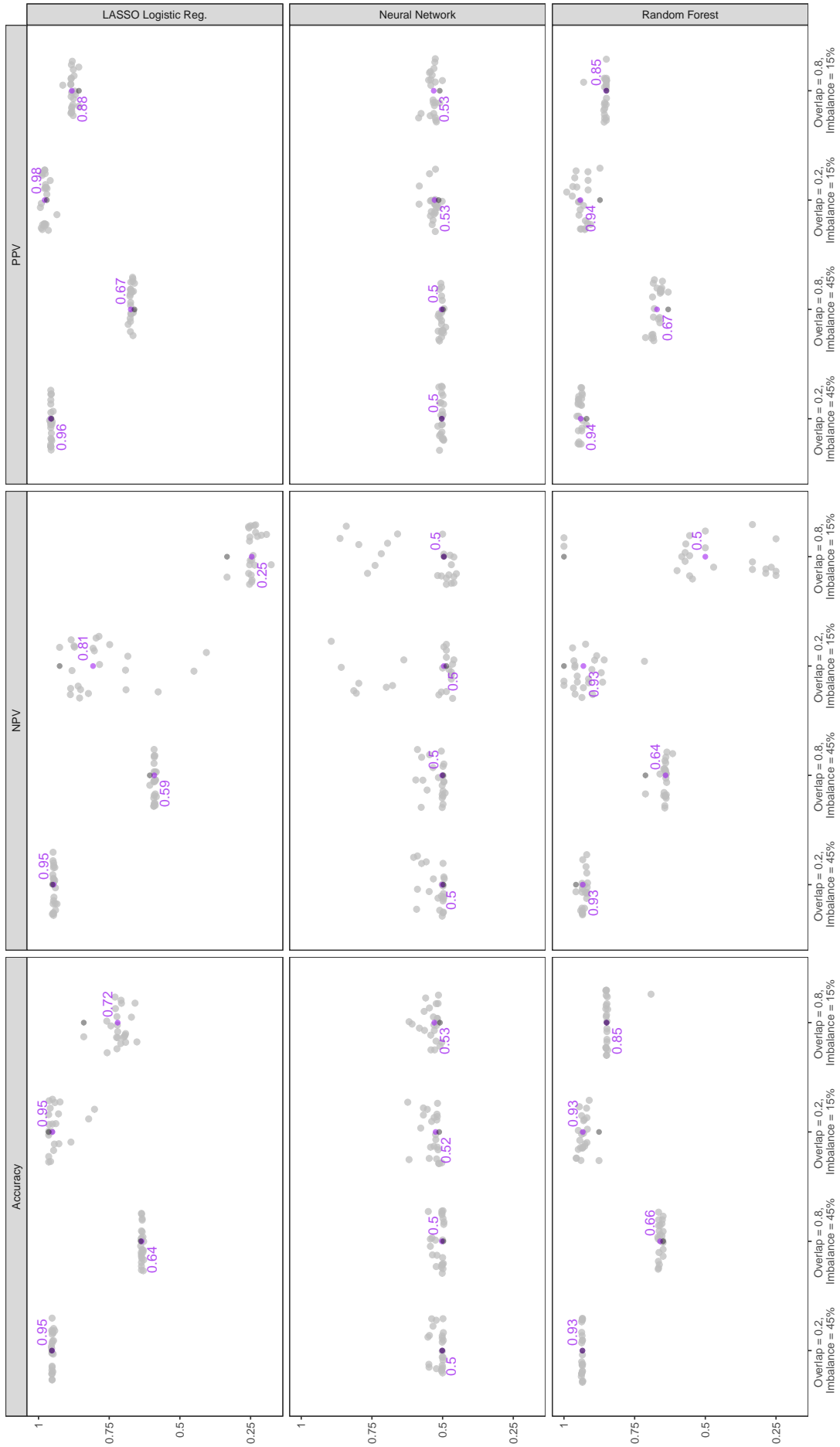
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 1750 samples, 100 variables of which 0% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



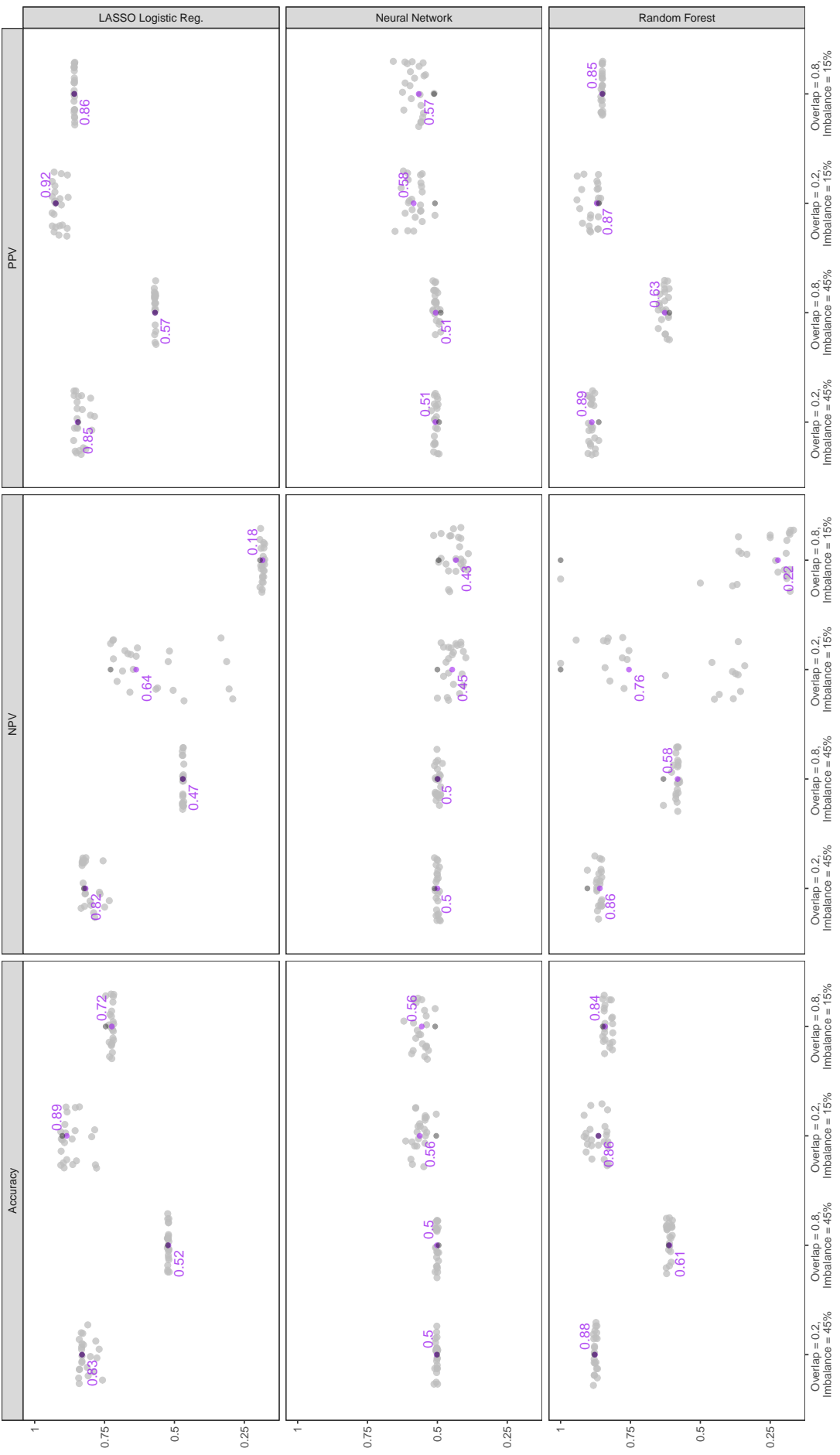
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples. 20 variables of which 0% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



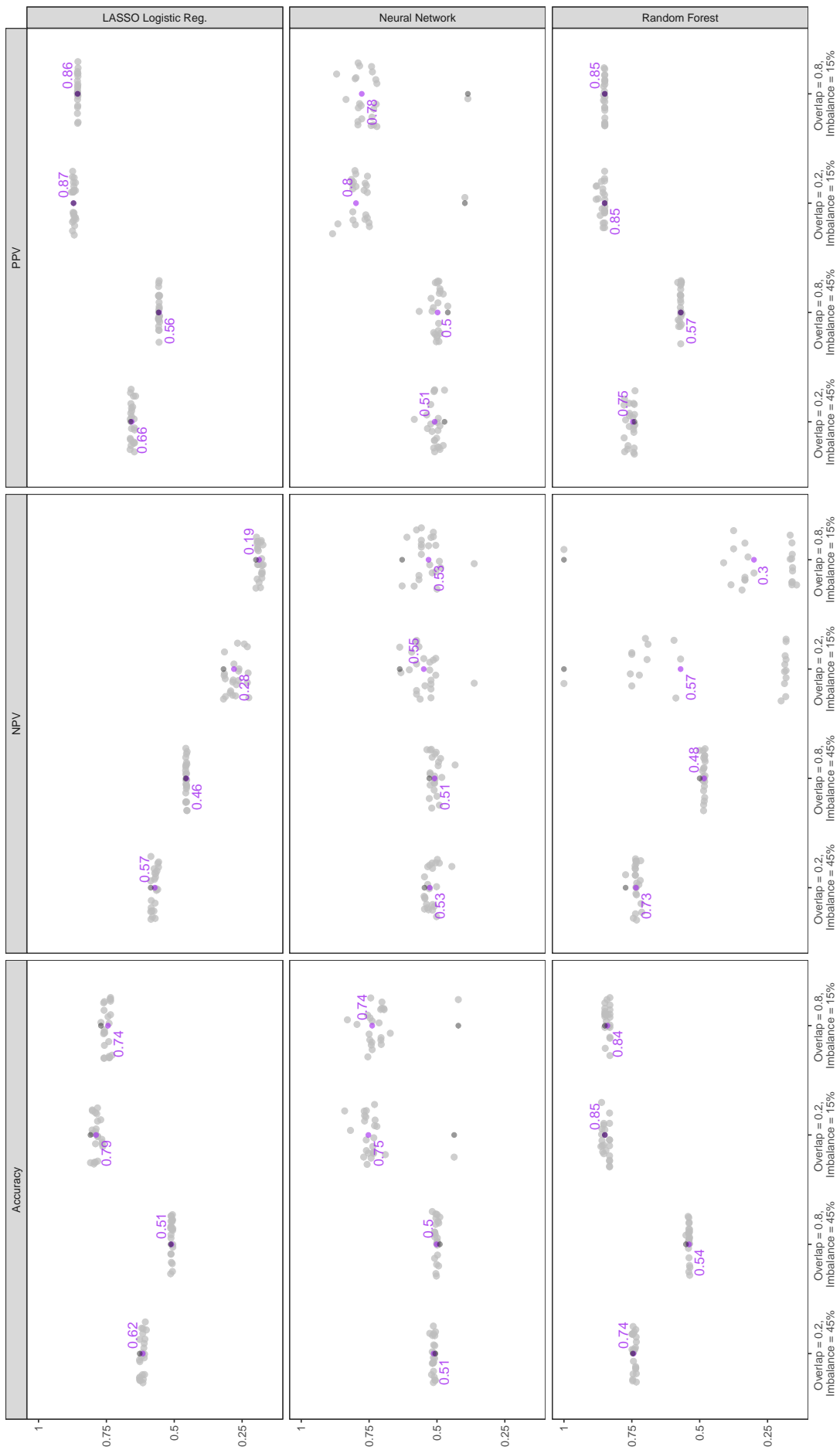
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 150 variables of which 0% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



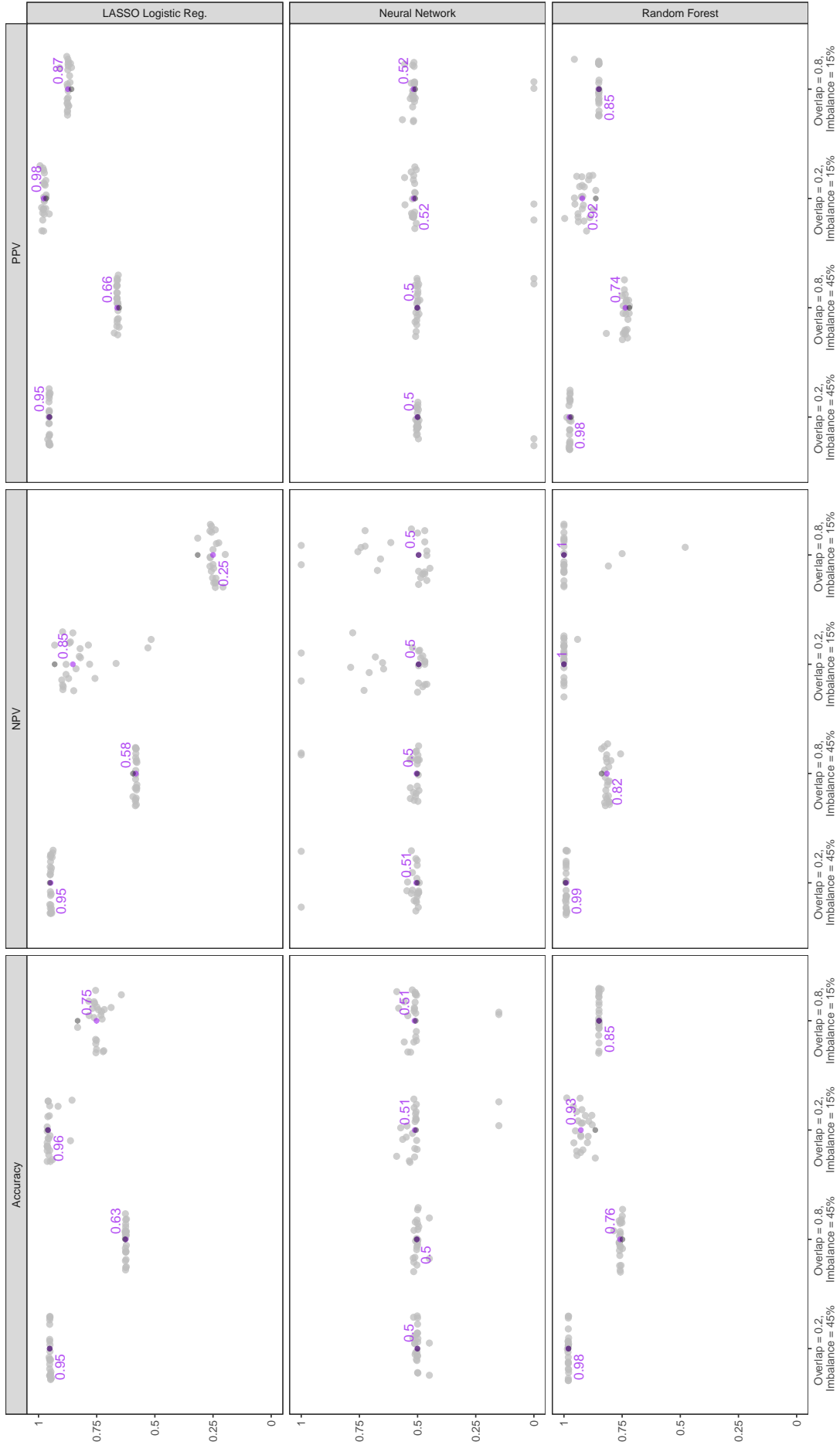
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples. 100 variables of which 25% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



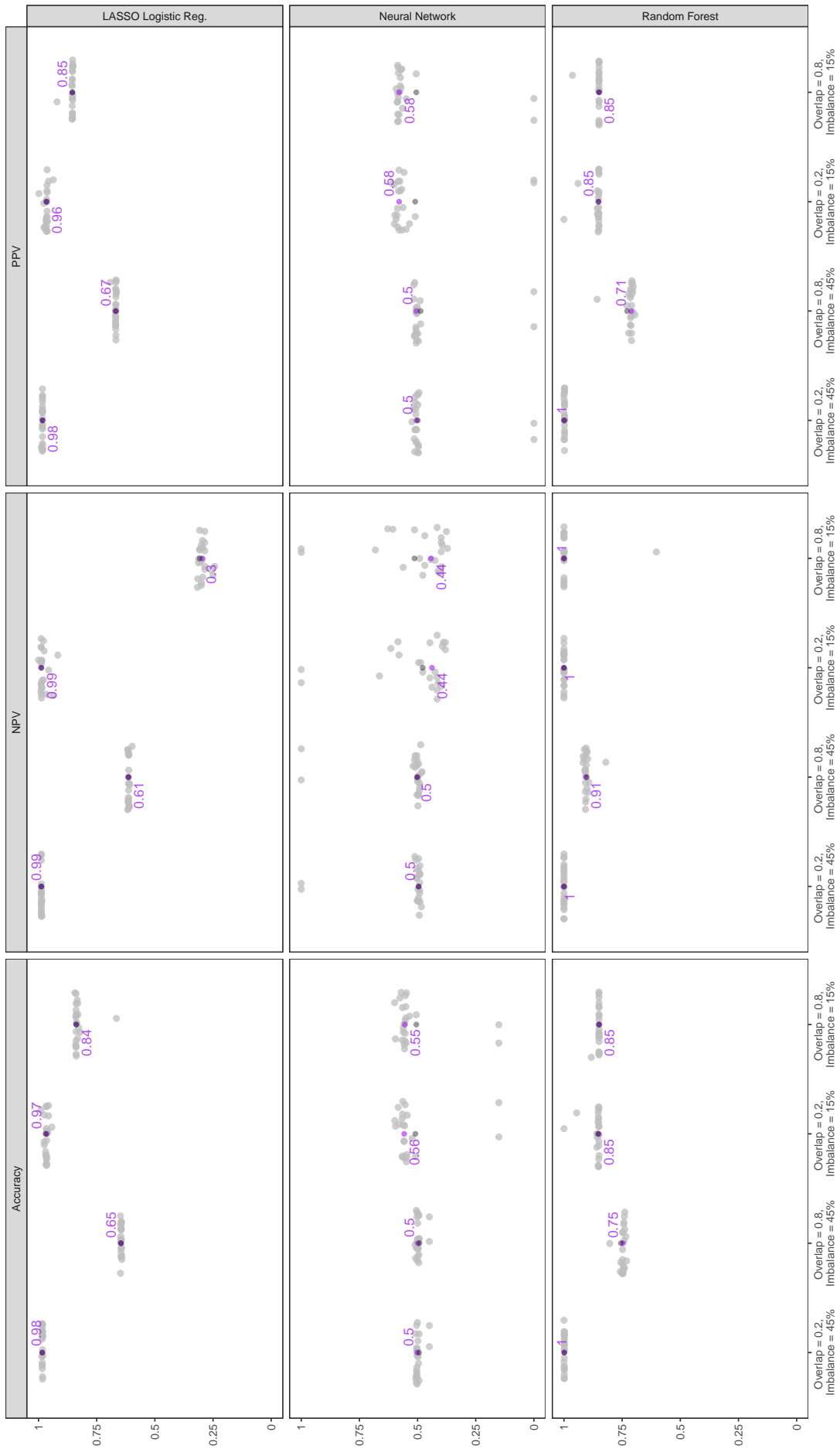
Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples. 100 variables of which 85% were categorical, and no missing data. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



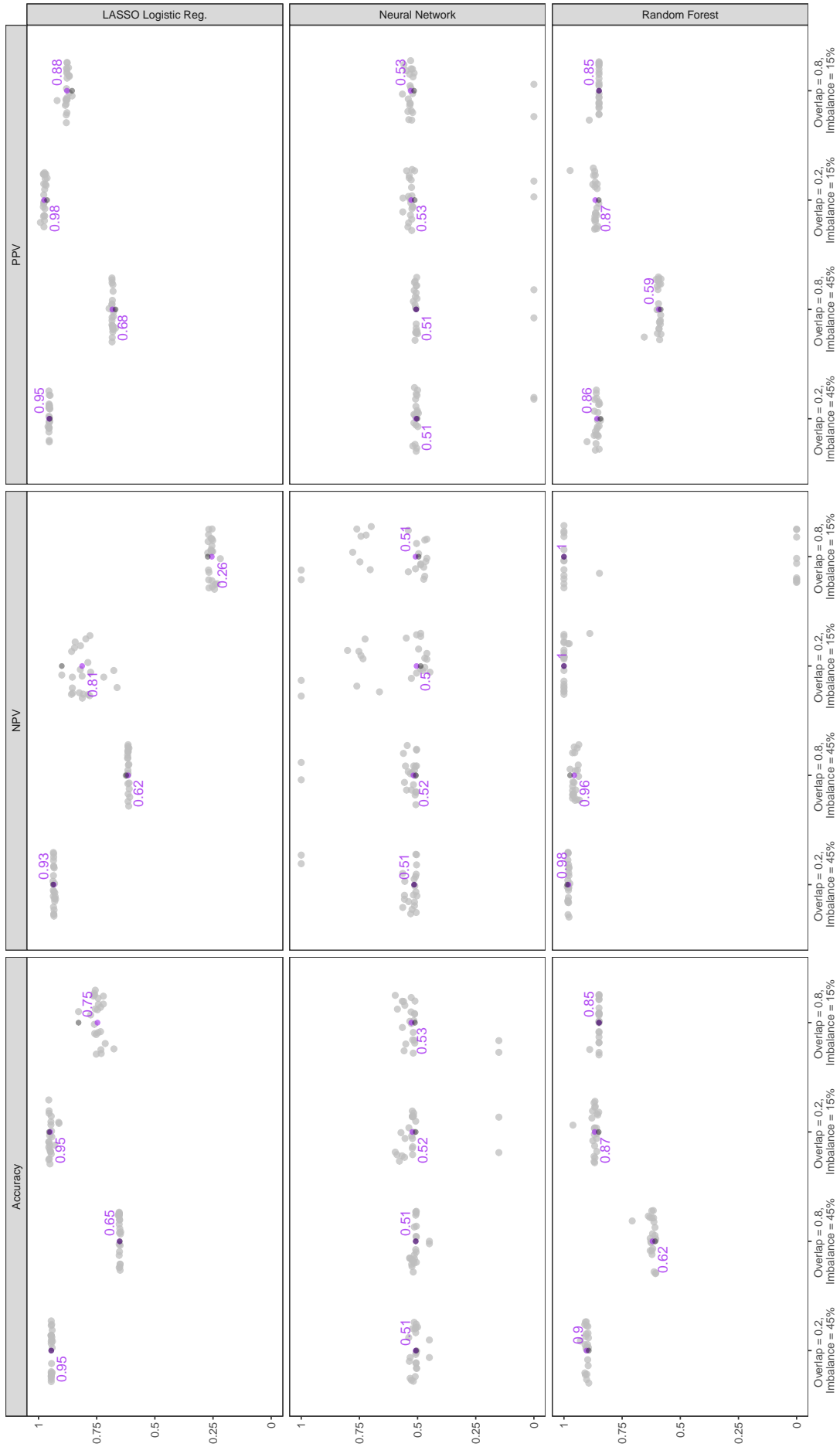
Median performance metric over results from models trained on data balanced with each method. Each training set had 750 samples. 100 variables of which 0% were categorical, and 30% data missing completely at random. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray points marks results on unbalanced data.



Median performance metric over results from models trained on data balanced with each method. Each training set had 750 samples, 100 variables of which 0% were categorical, and 70% data missing completely at random. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 100 variables of which 0% were categorical, and 30% minority data missing at random. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.



Median performance metric taken over results from models trained on data balanced with each method. Each training set had 750 samples, 100 variables of which 0% were categorical, and 70% minority data missing at random. The balancing methods with the three largest averages are given at the top of each panel. The purple point gives the median for each group of points. The dark gray point marks results on unbalanced data.

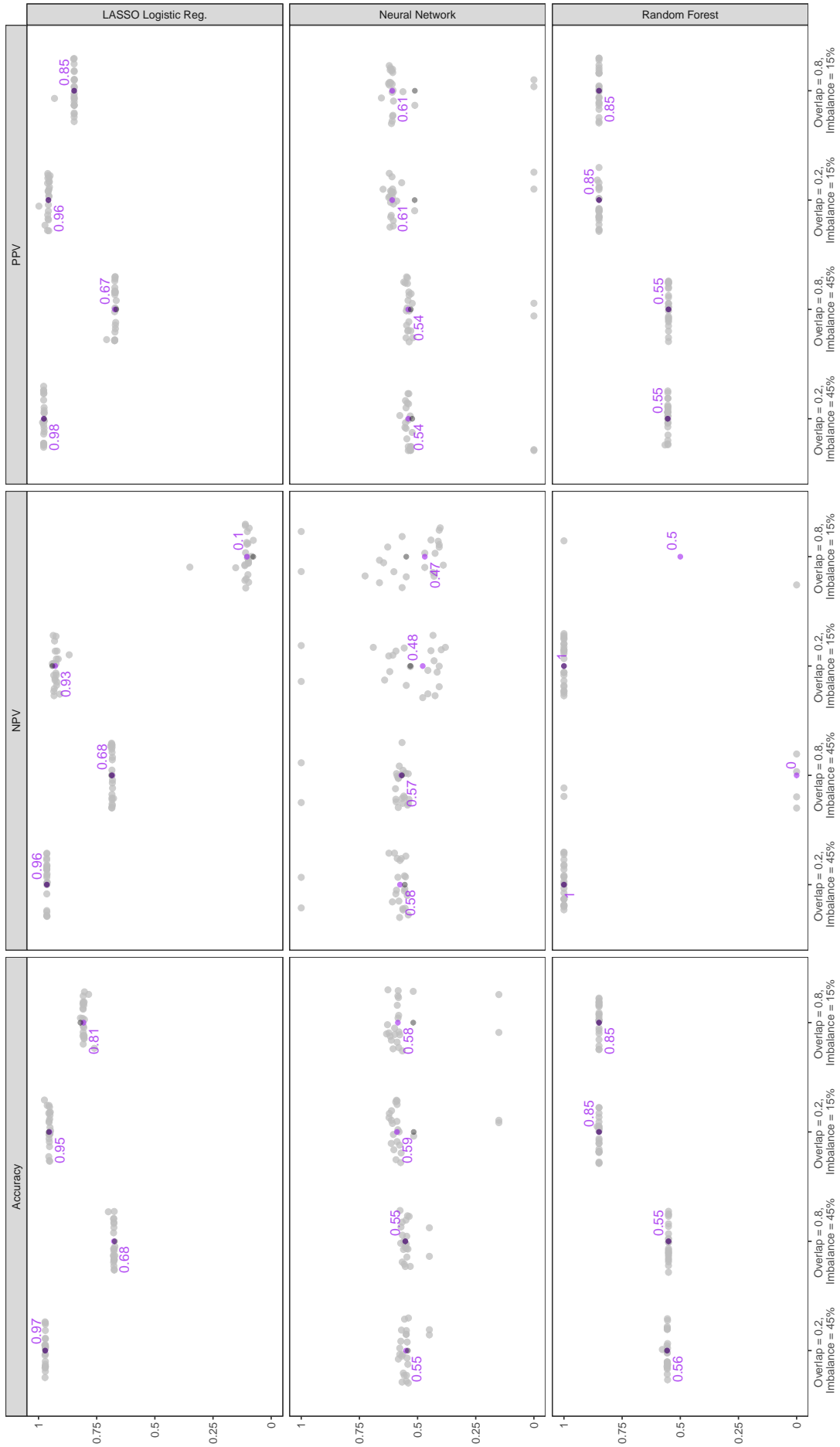
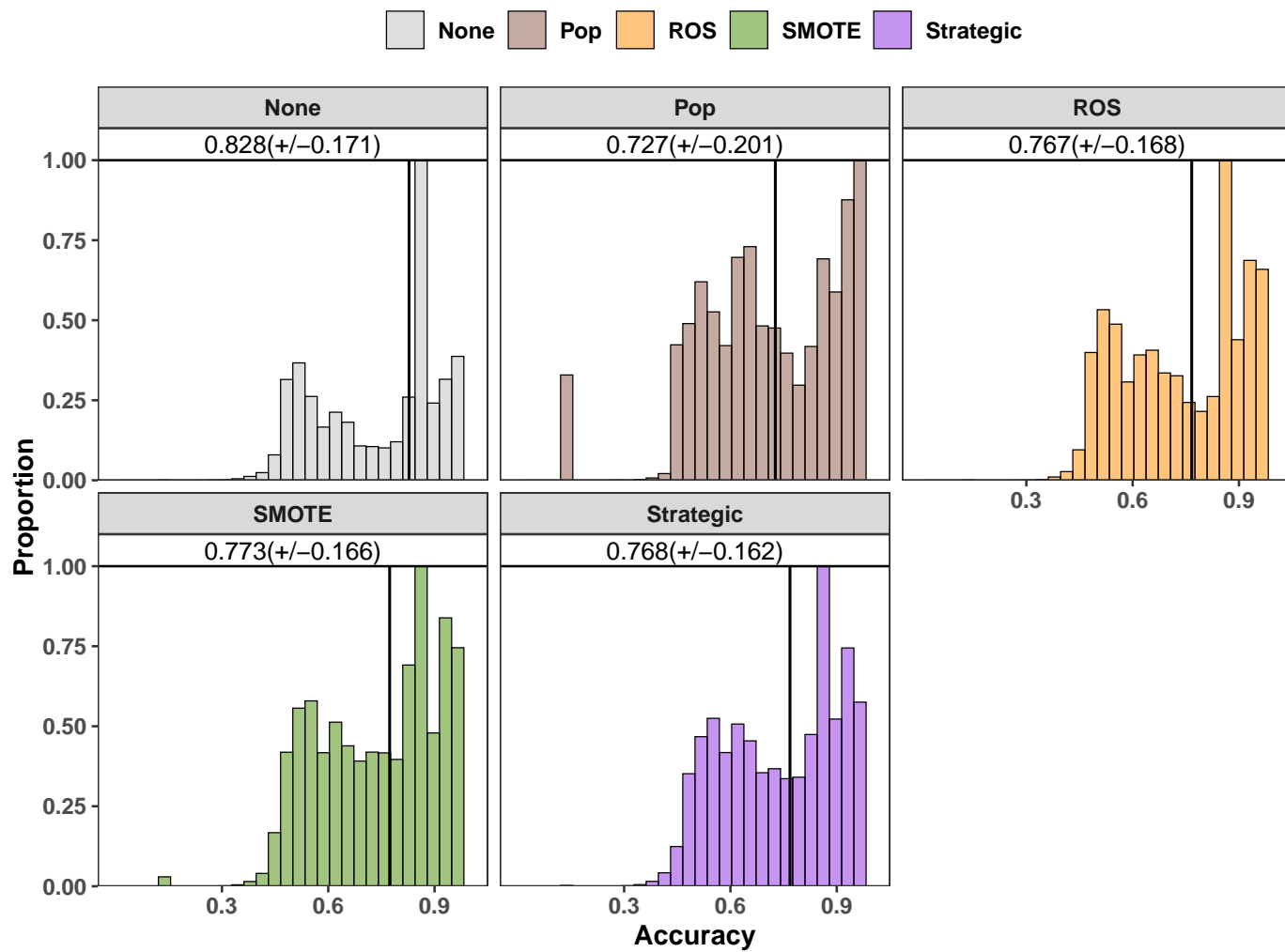
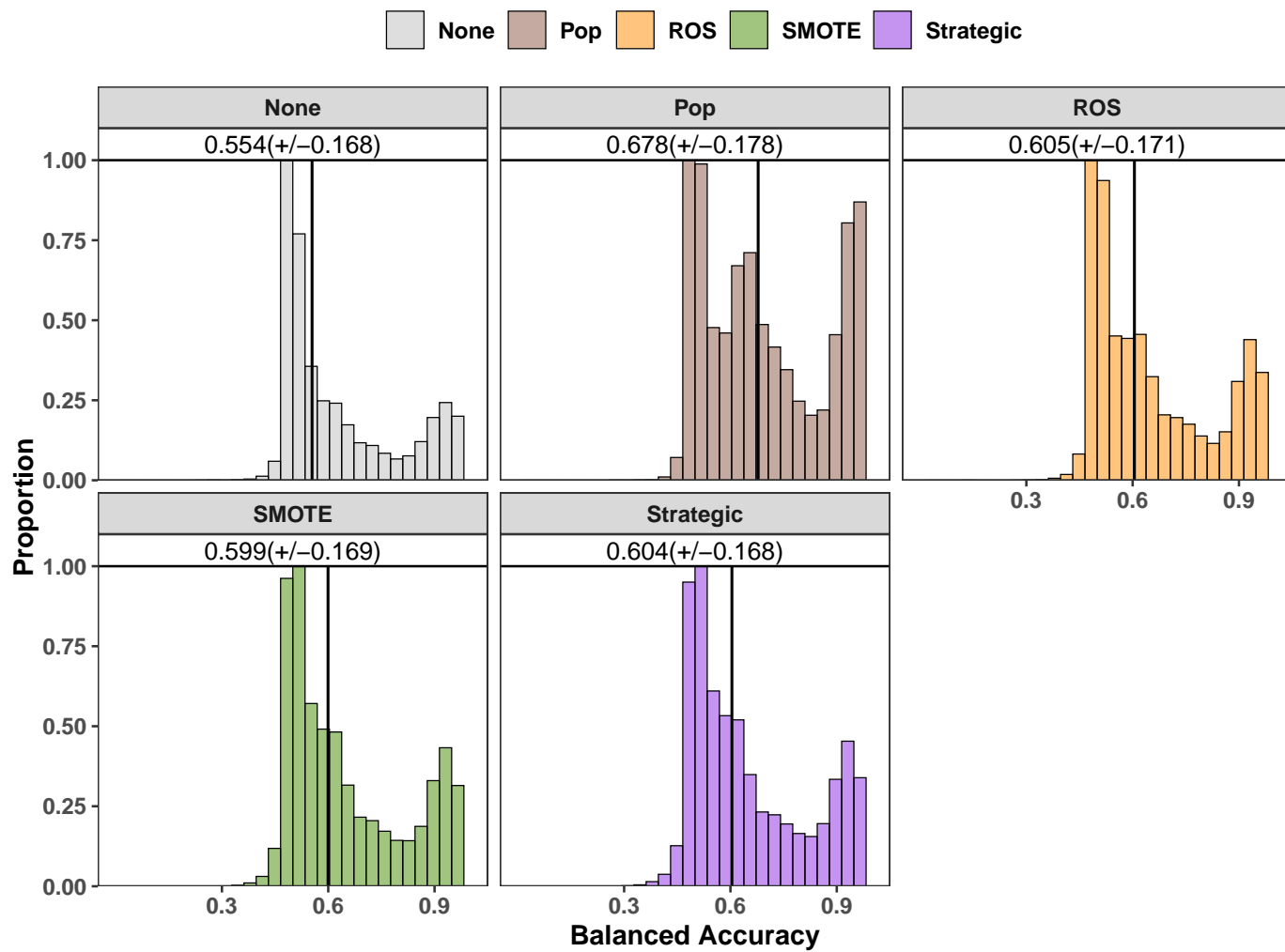
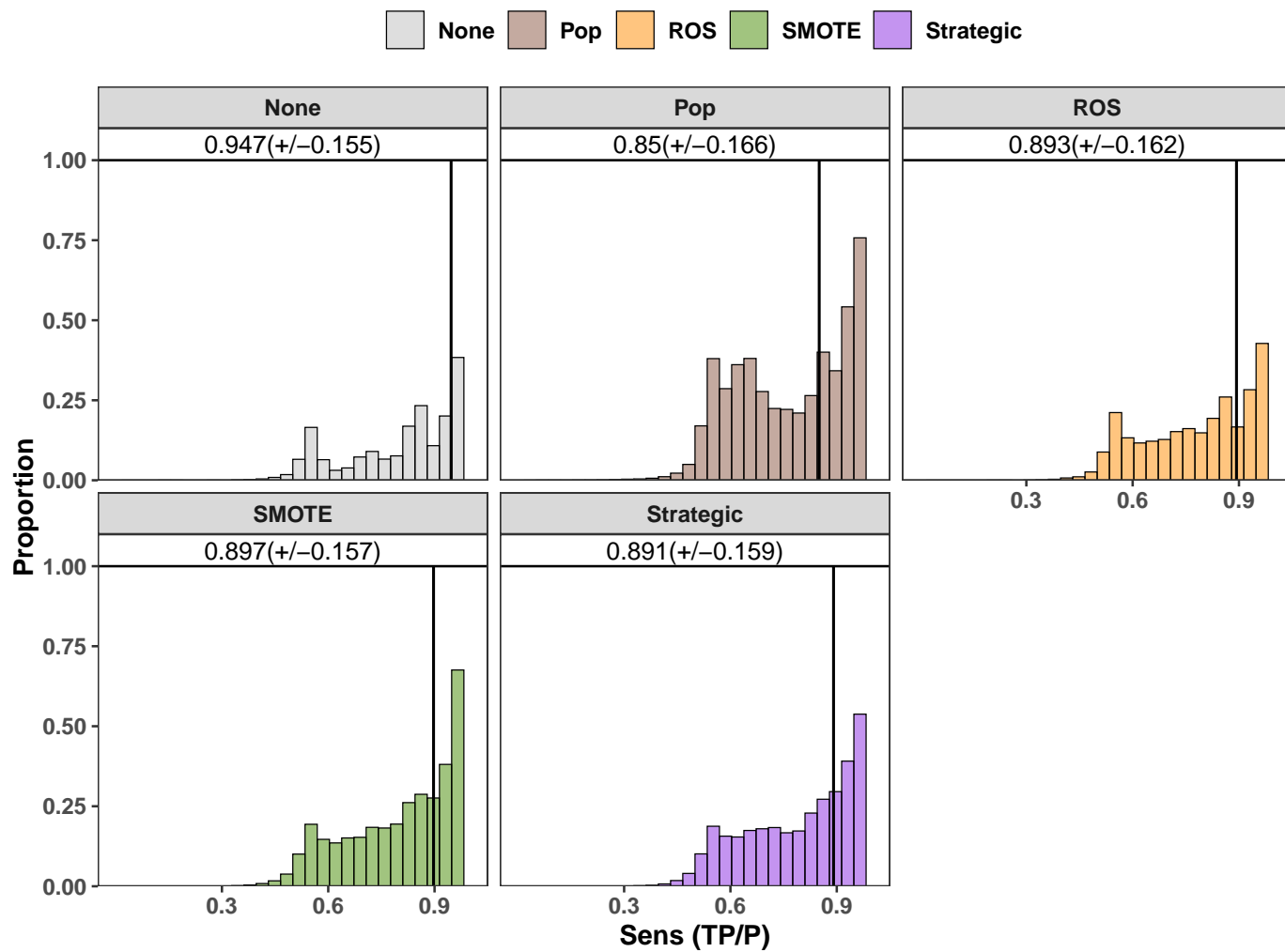
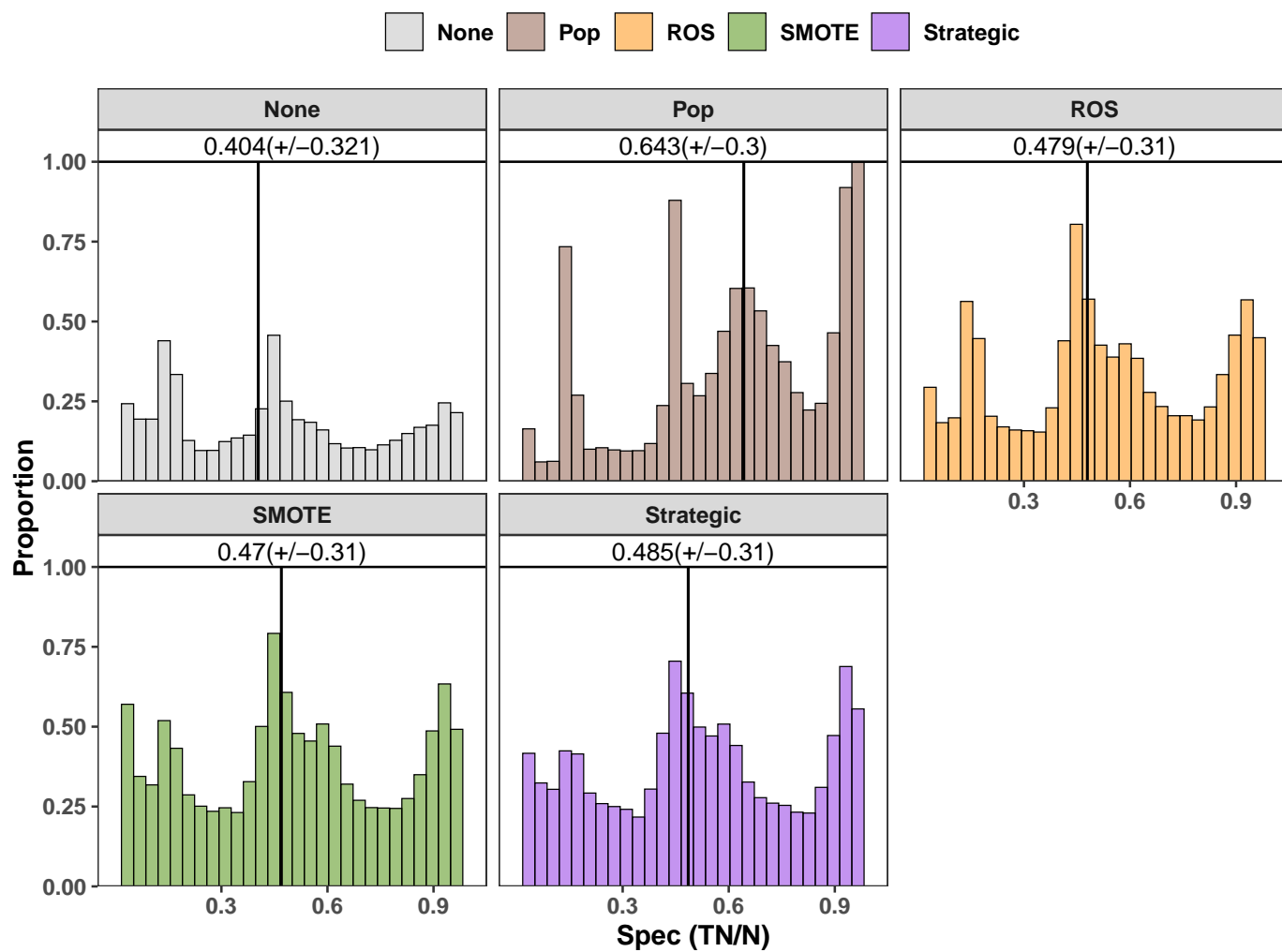


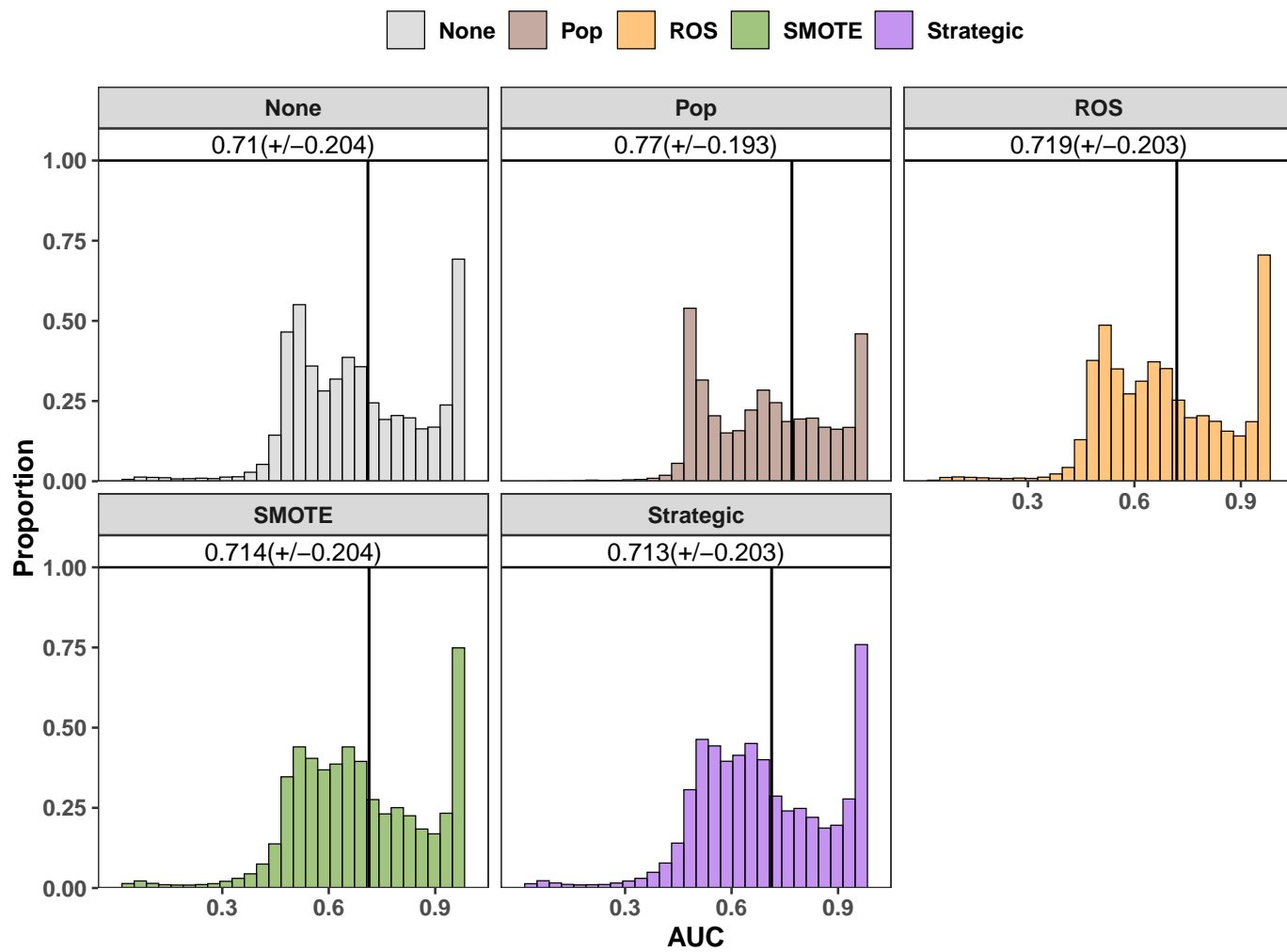
Figure B.2: Distribution of Performance Metrics Solely By Oversampling Method: Distribution of balanced accuracy and Kappa coefficient based on oversampling method. Data were aggregated over all simulations to make an initial determination about whether the oversampling method used has an impact.

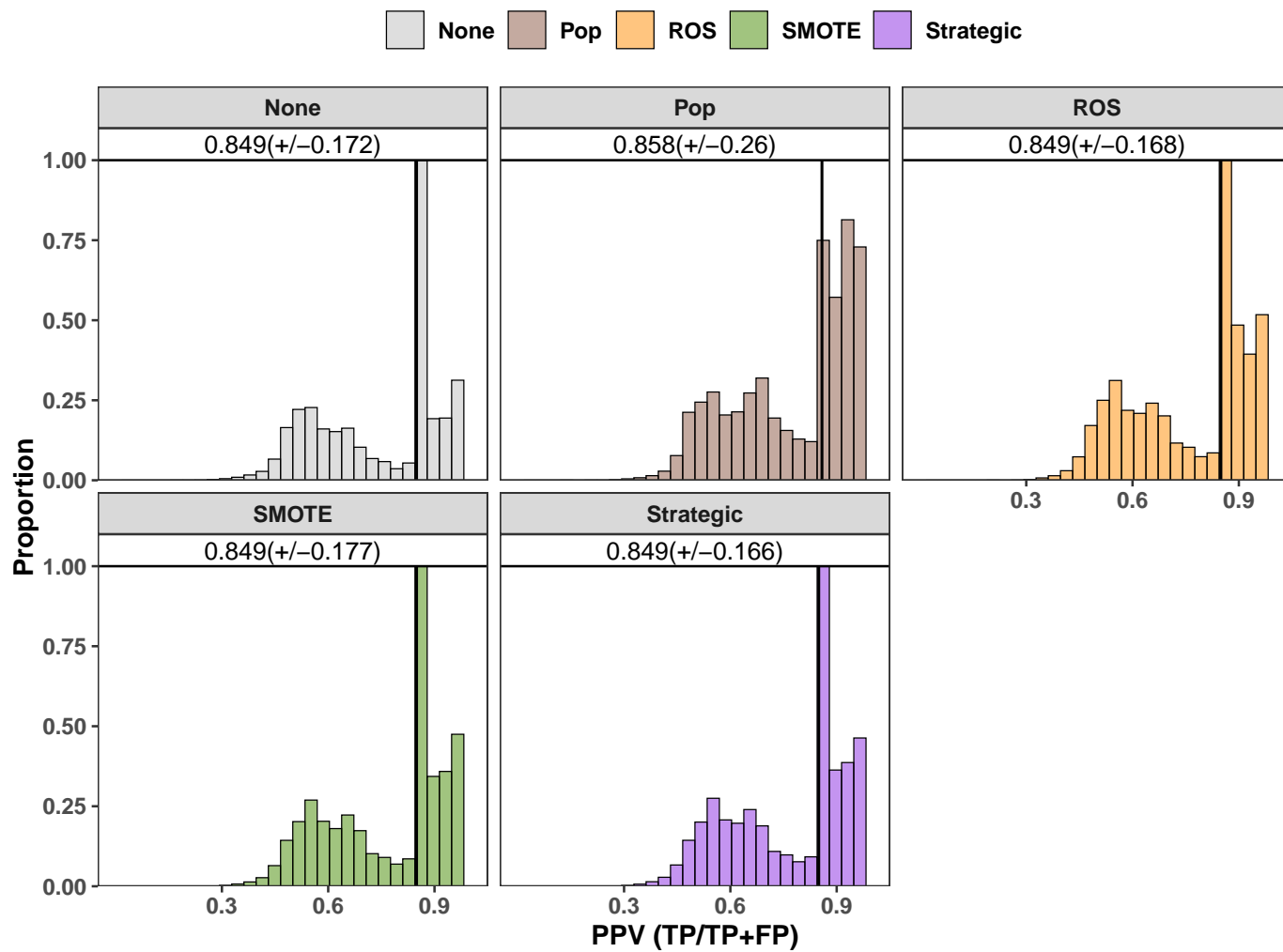


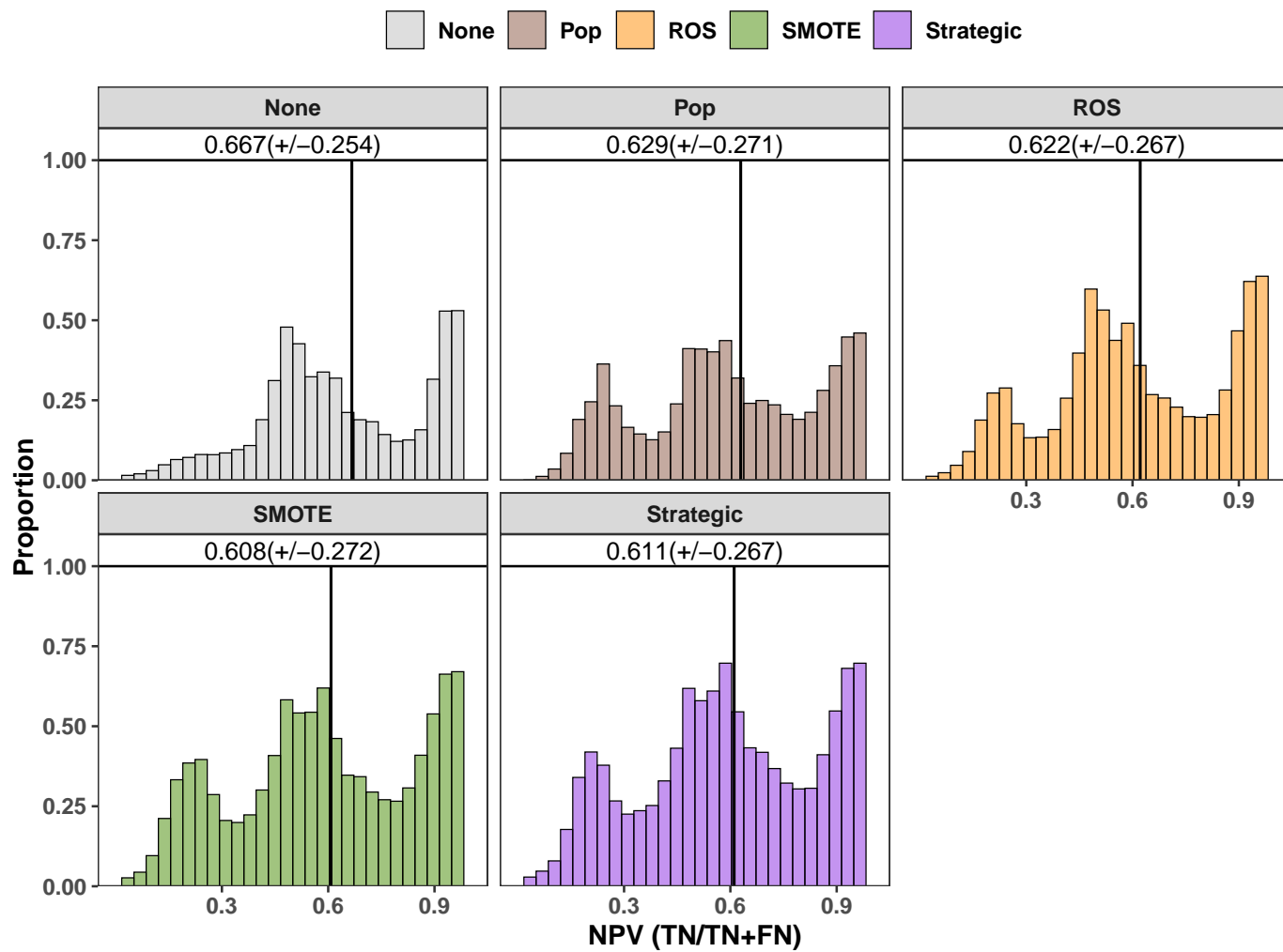


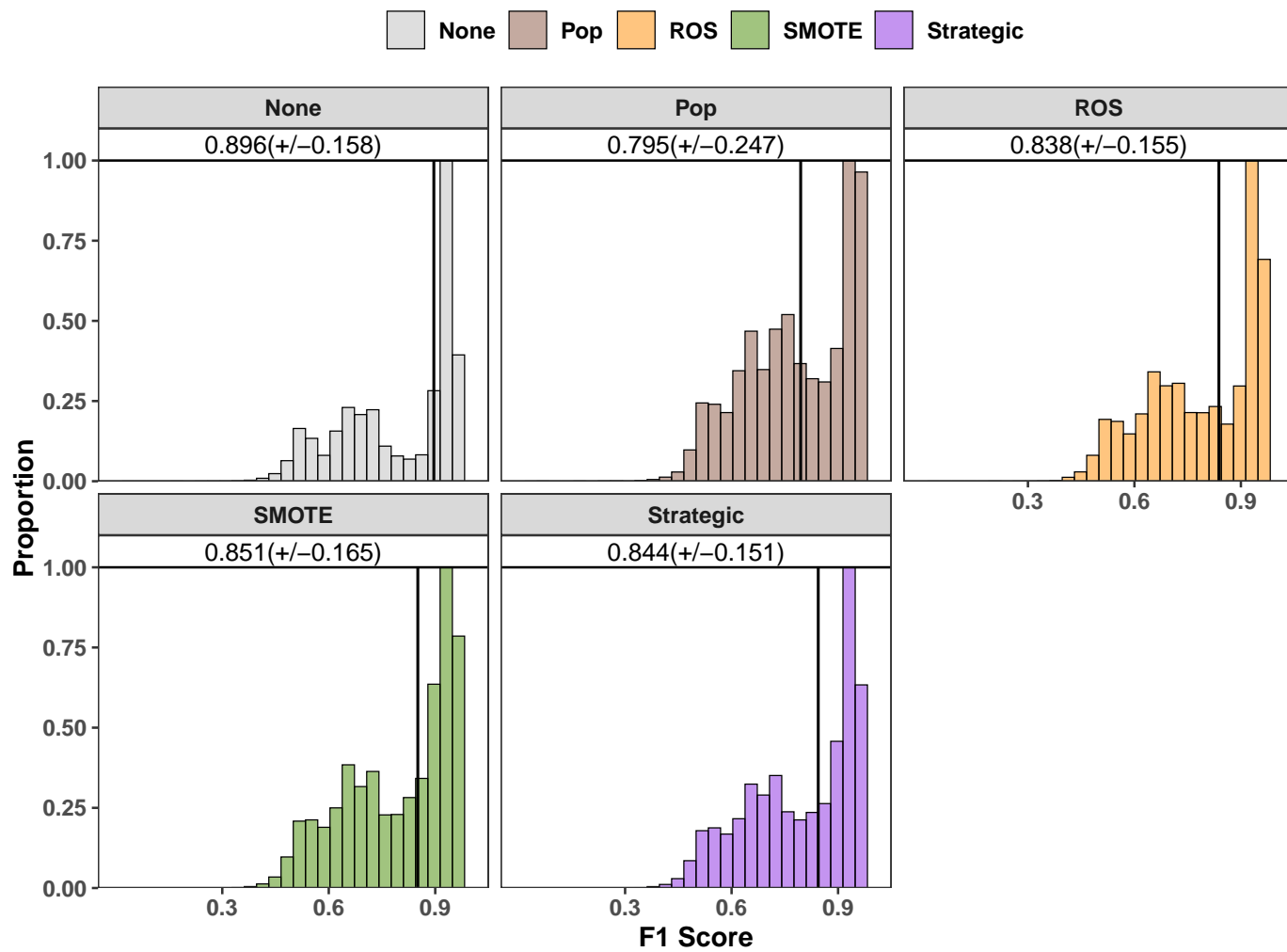












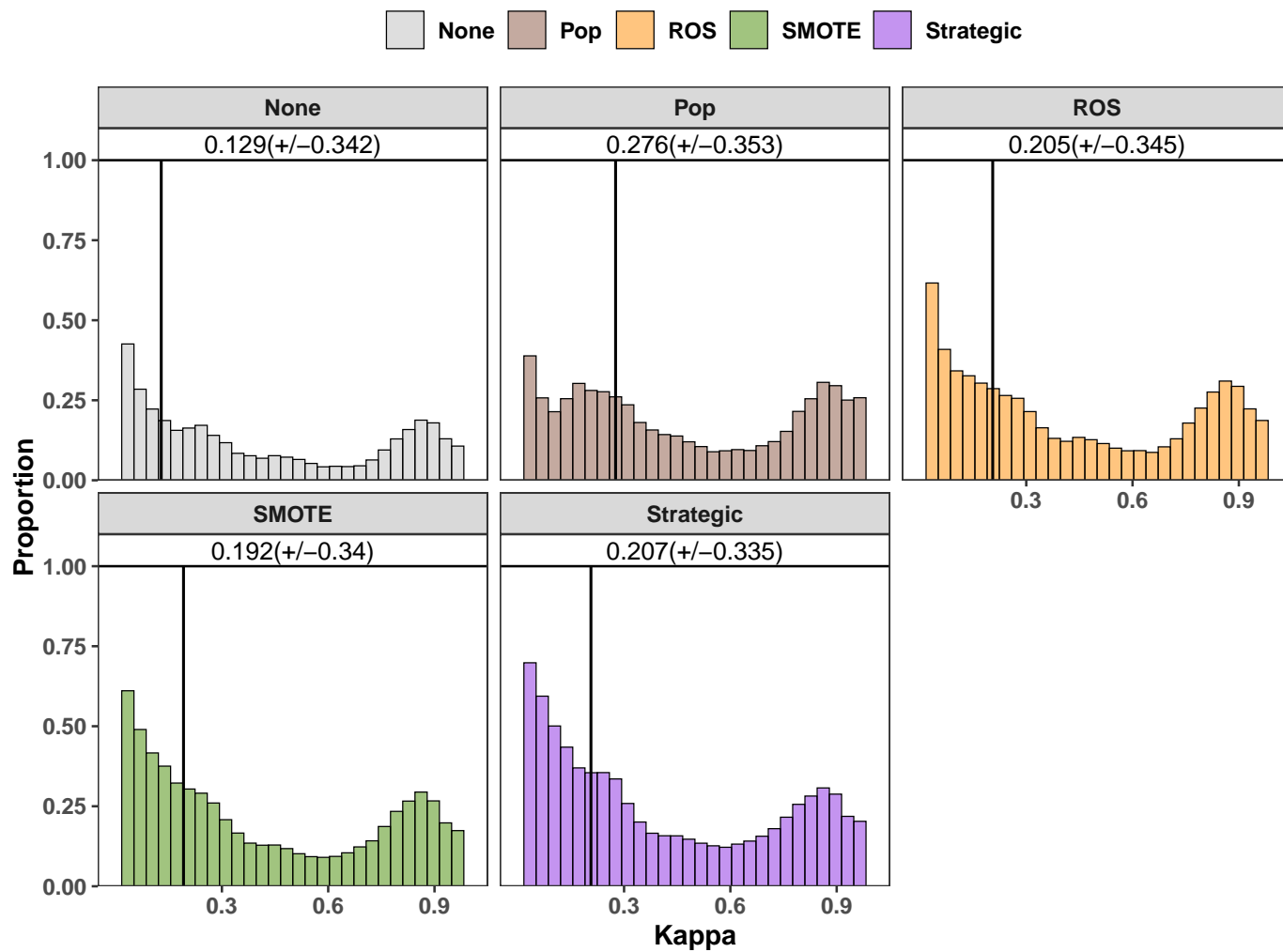
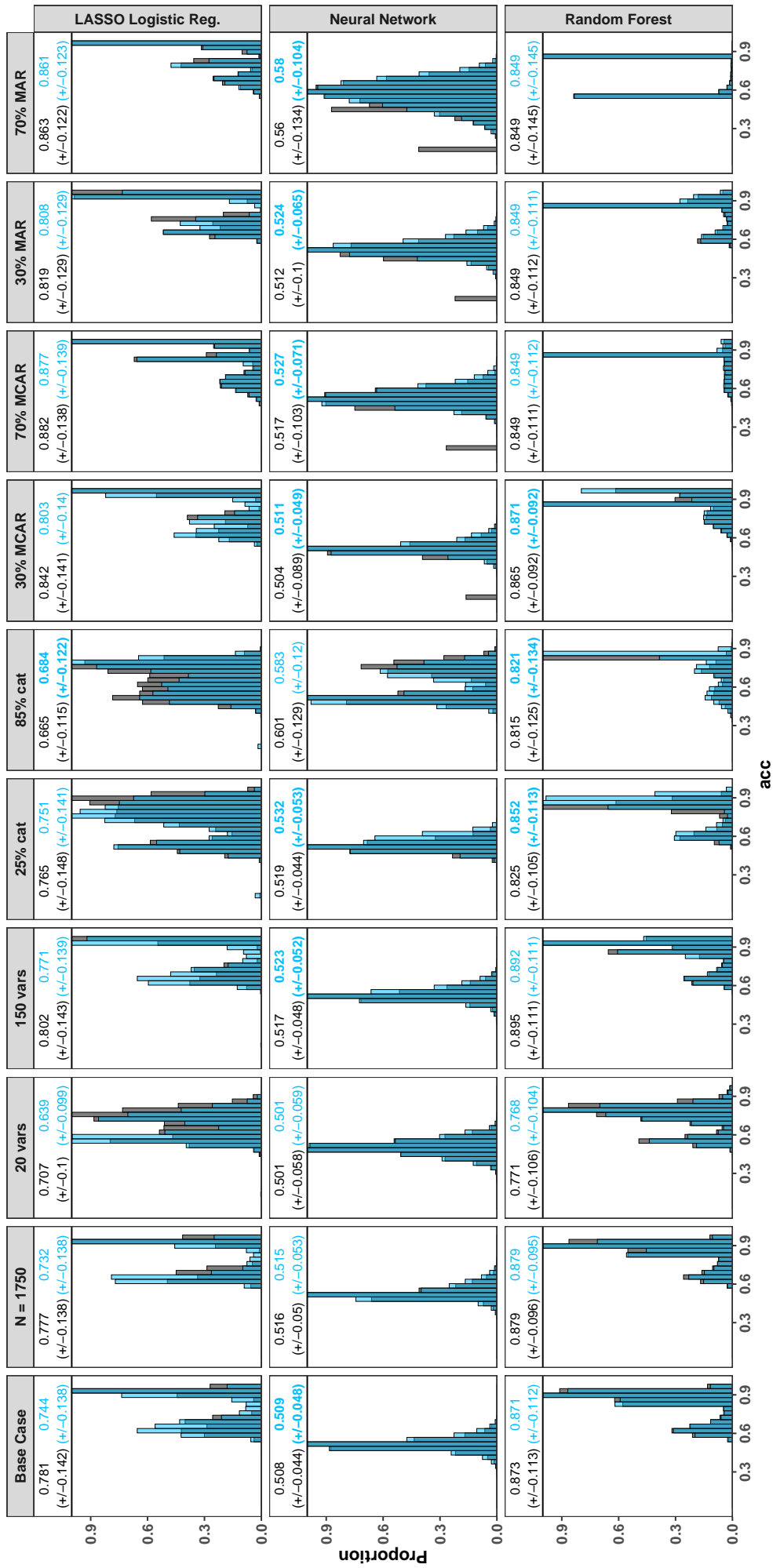
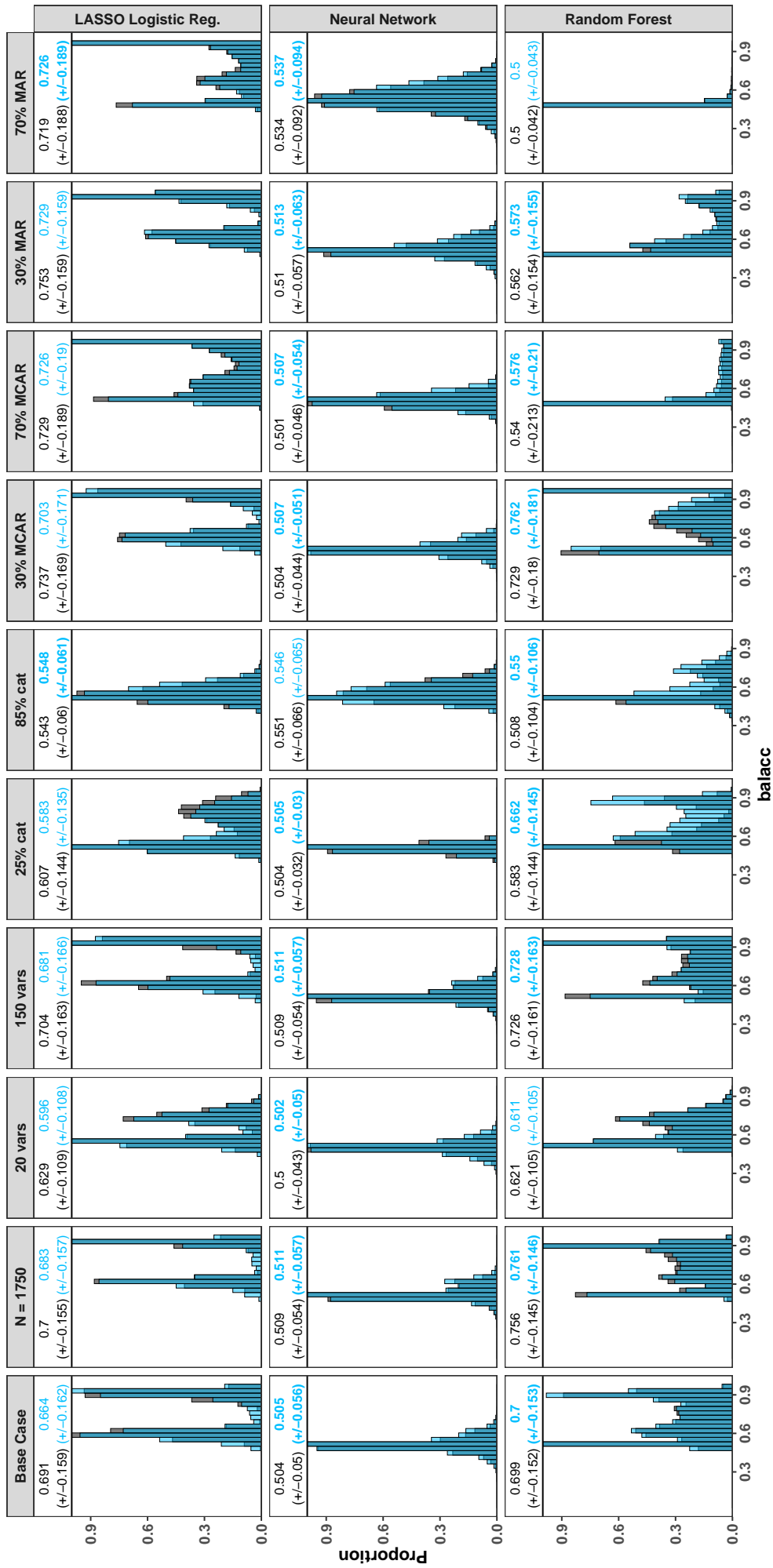


Figure B.3: Distribution of Performance Metrics For SMOTE and S-SMOTE By Data Case and Model: Distribution of performance metrics by oversampling method (S-SMOTE and SMOTE). Data were aggregated over all distributions for w and overlap/imbalance amounts in order to make initial comparisons between the two methods.

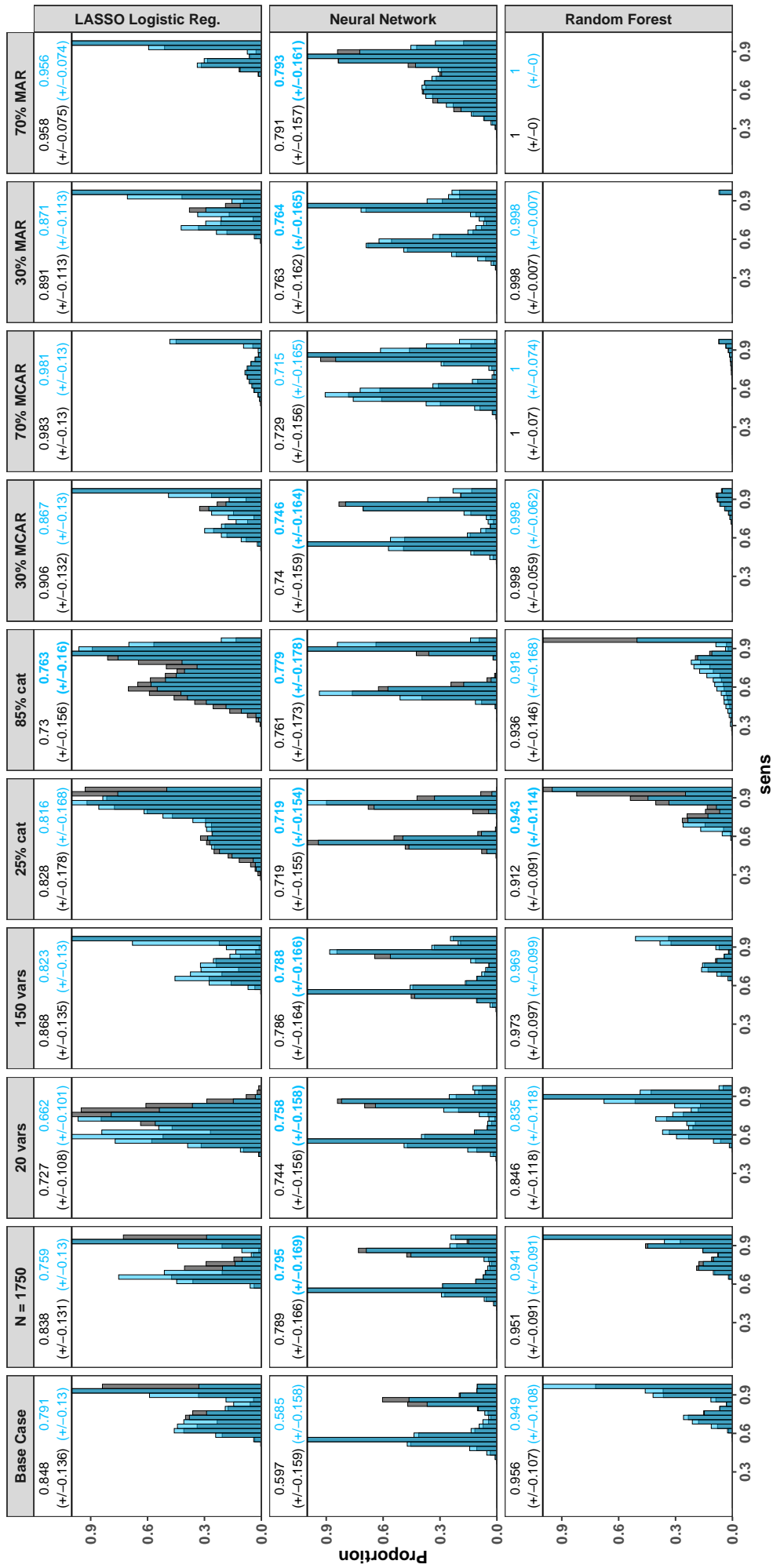
a SMOTE Strategic



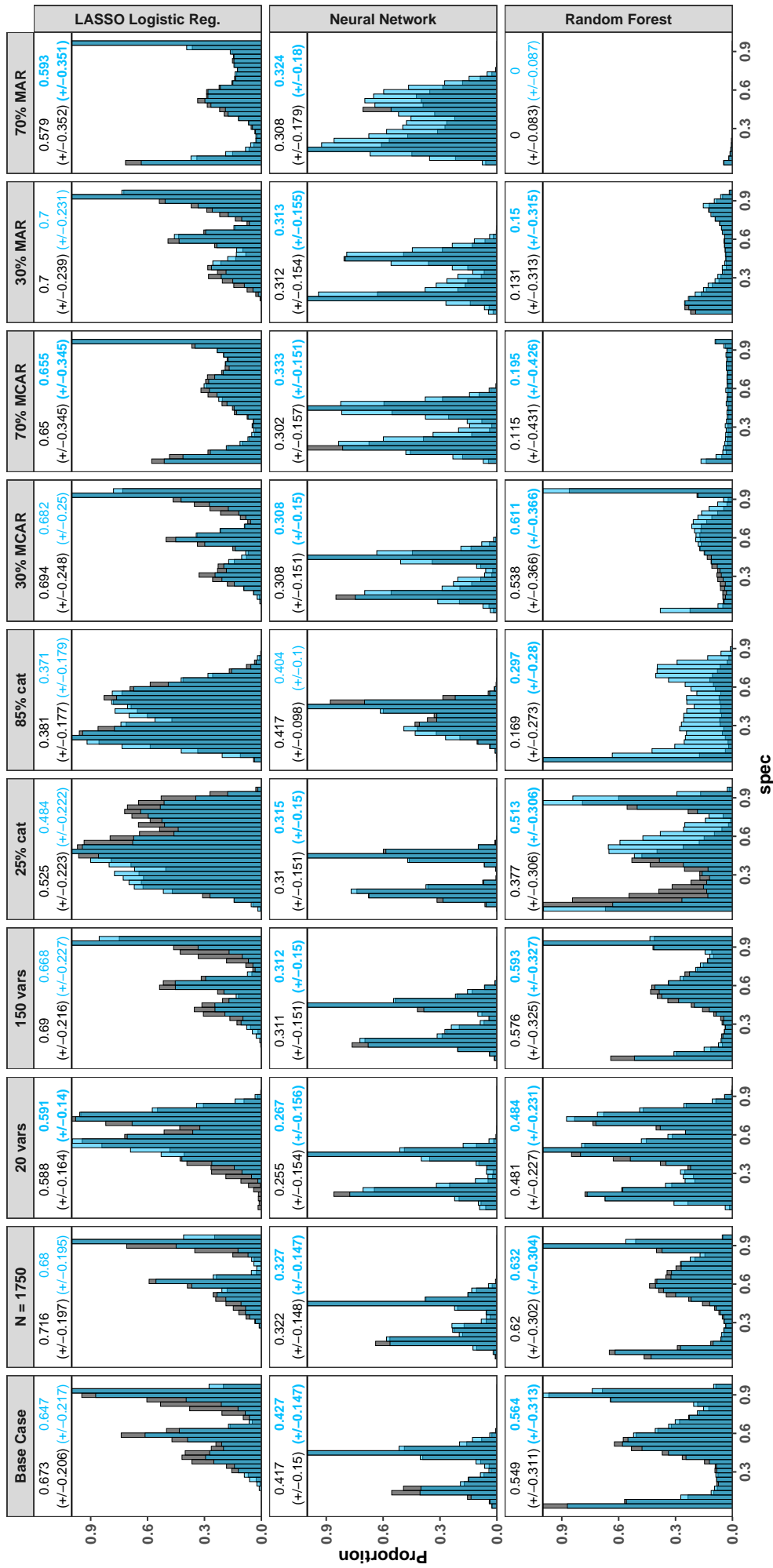
a SMOTE Strategic



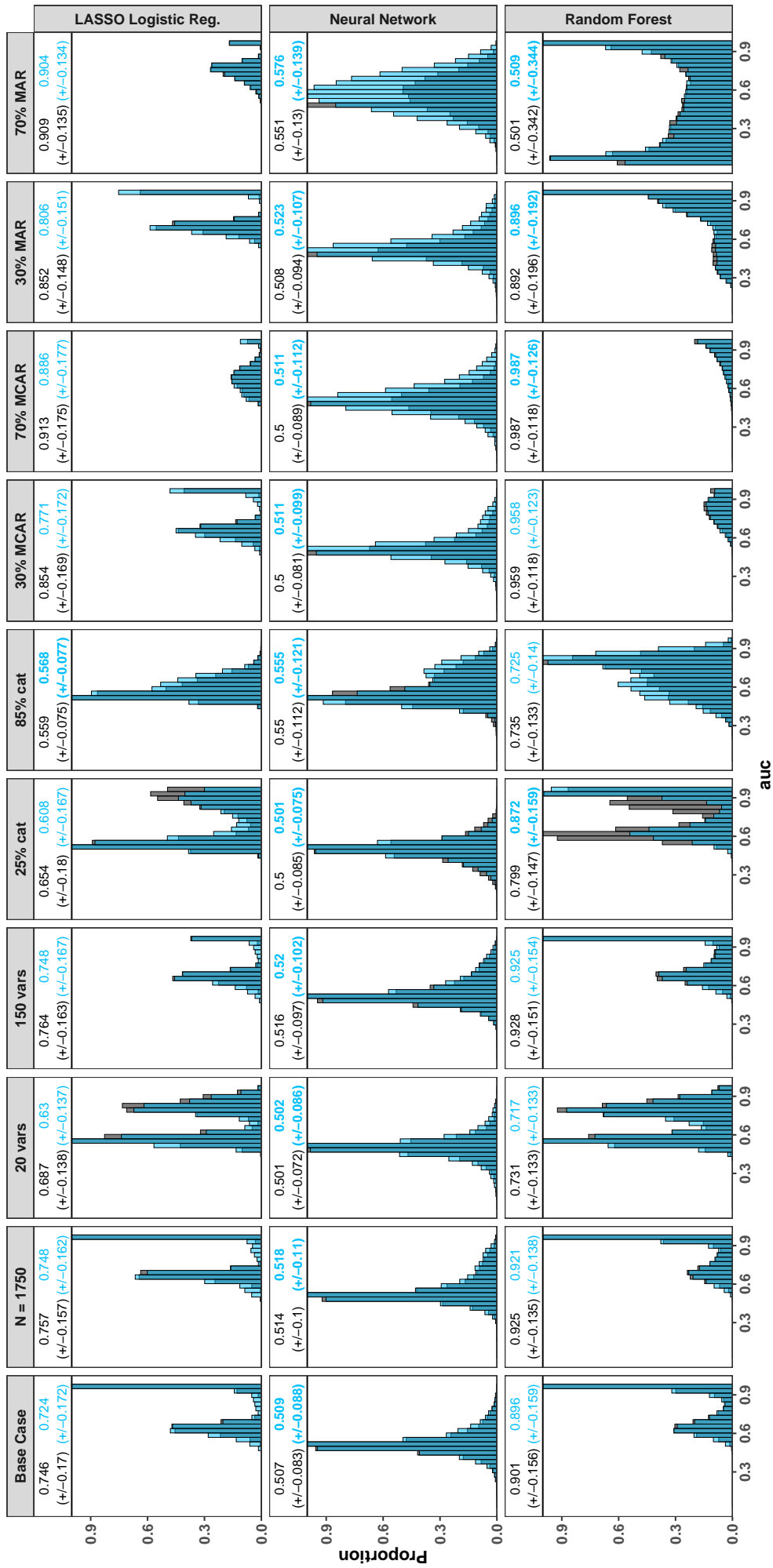
a SMOTE Strategic



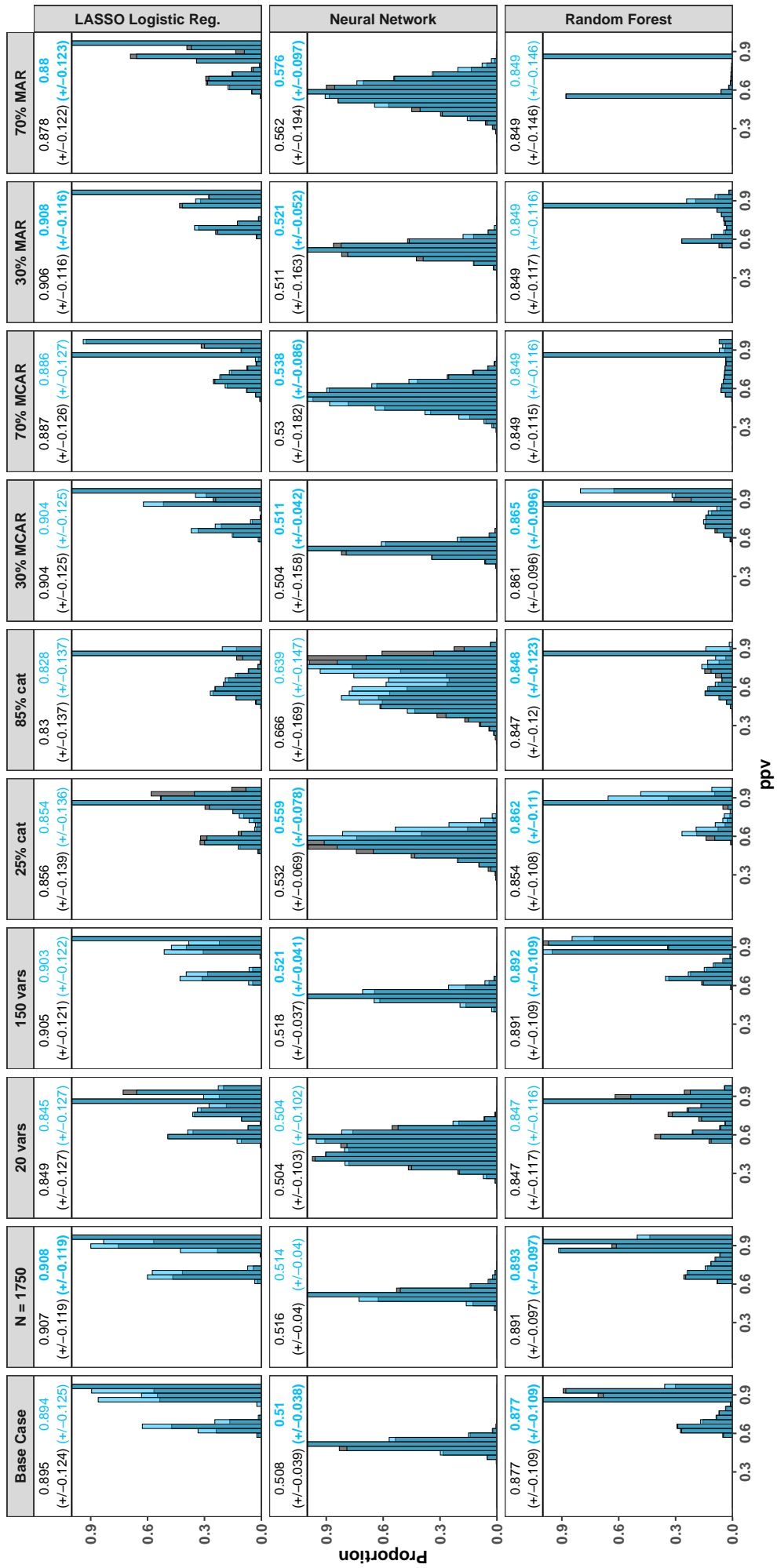
a SMOTE Strategic



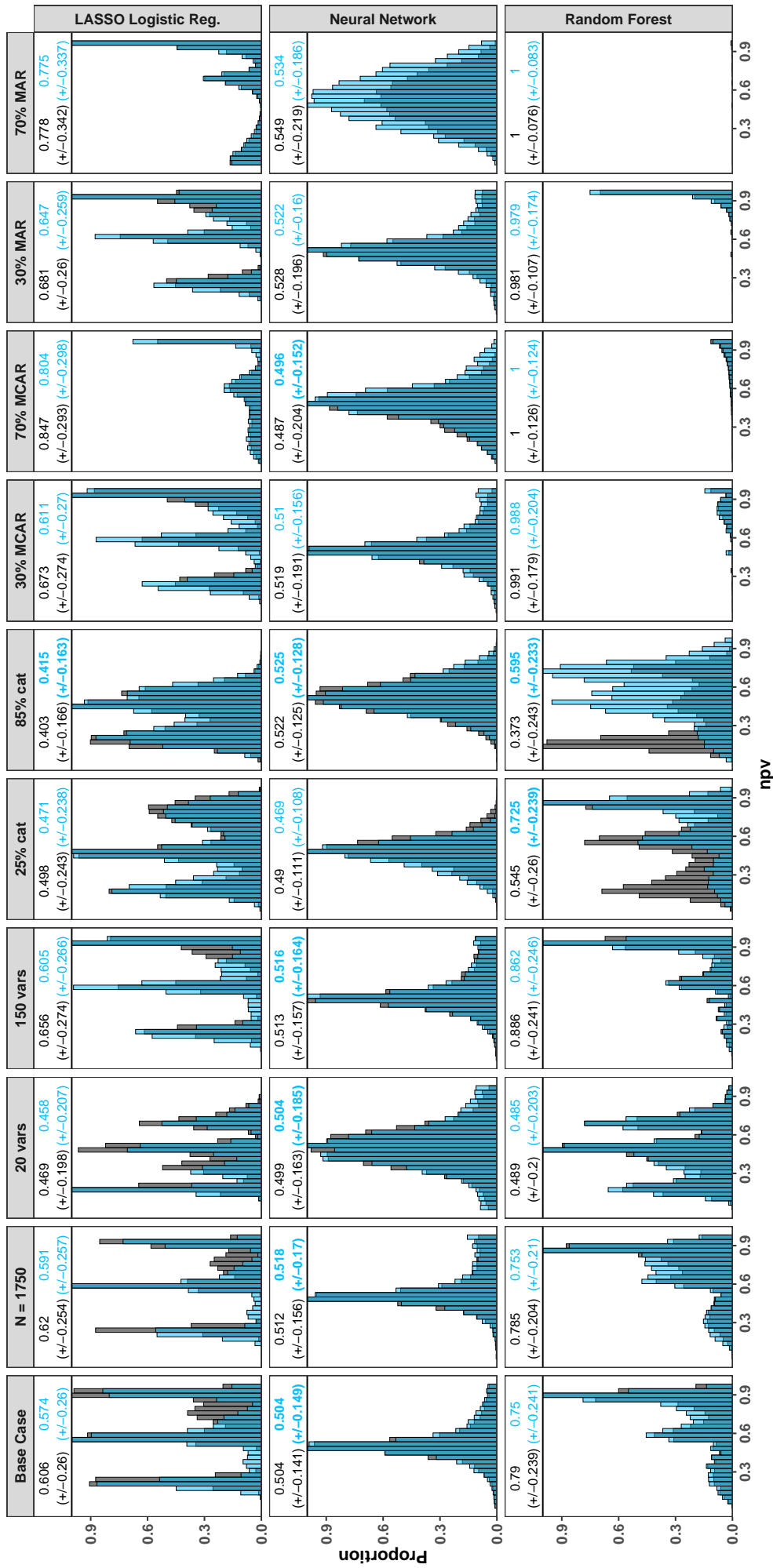
a SMOTE Strategic



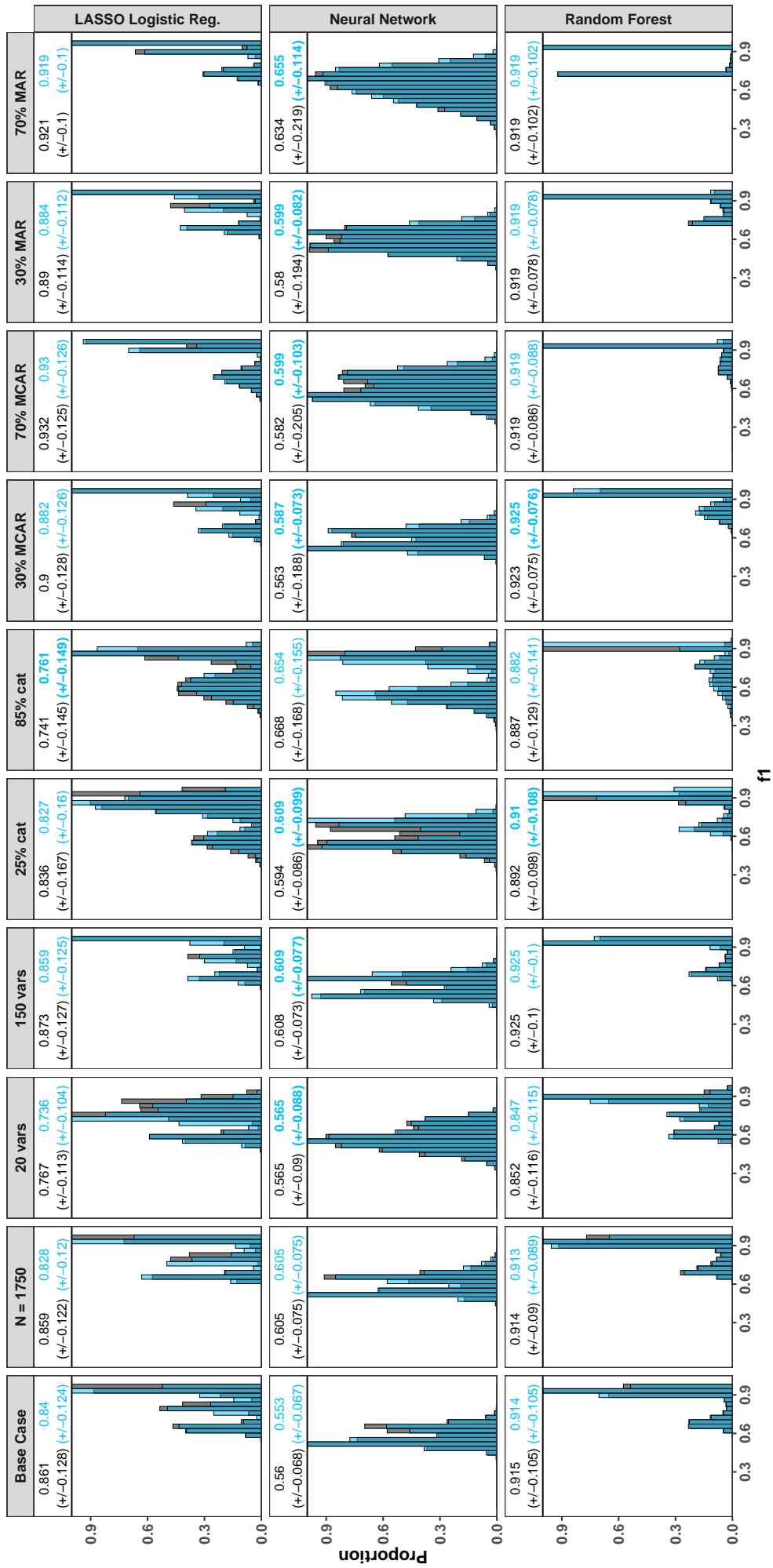
a SMOTE Strategic



a SMOTE Strategic



a SMOTE Strategic



a SMOTE Strategic

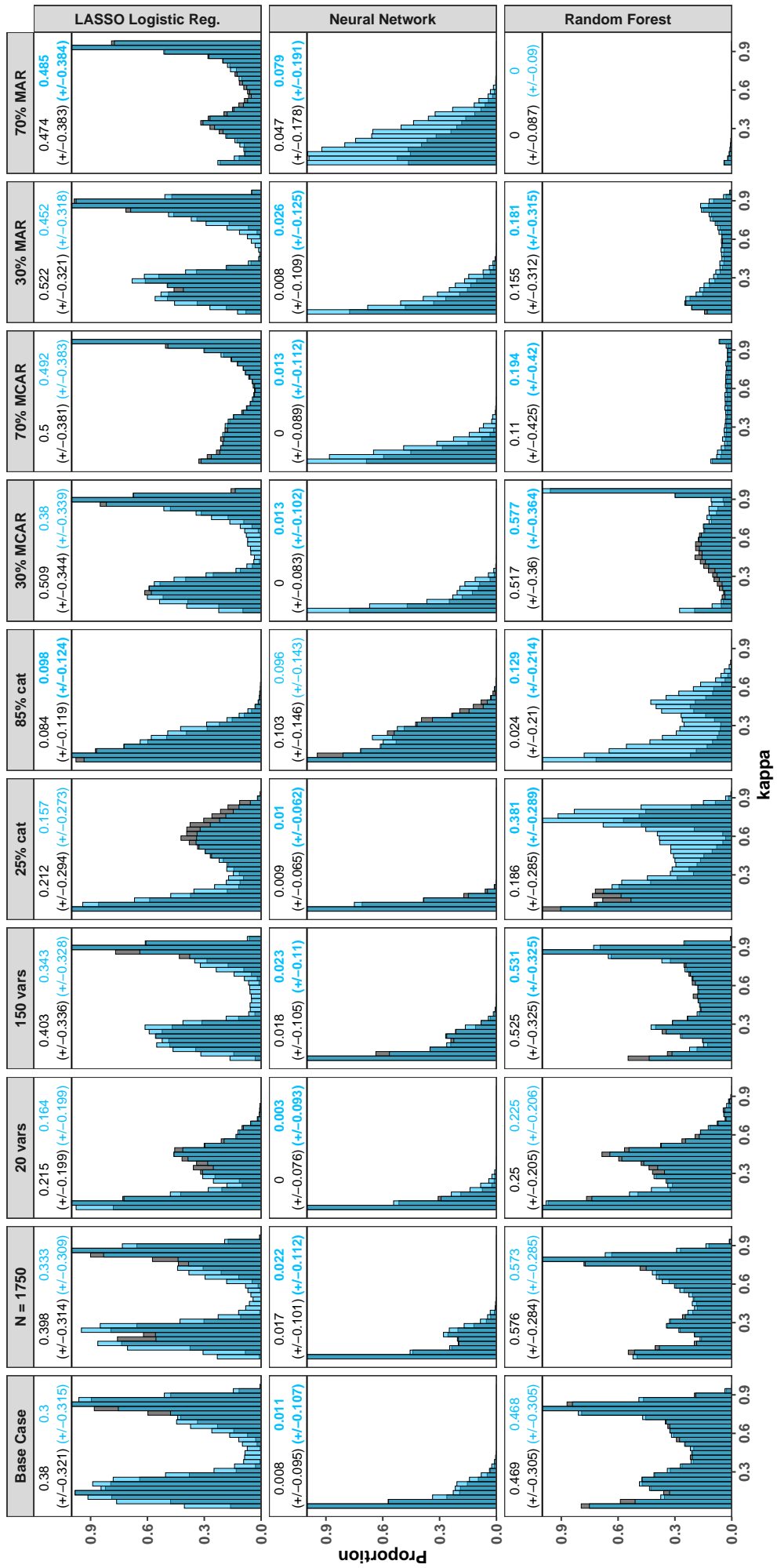
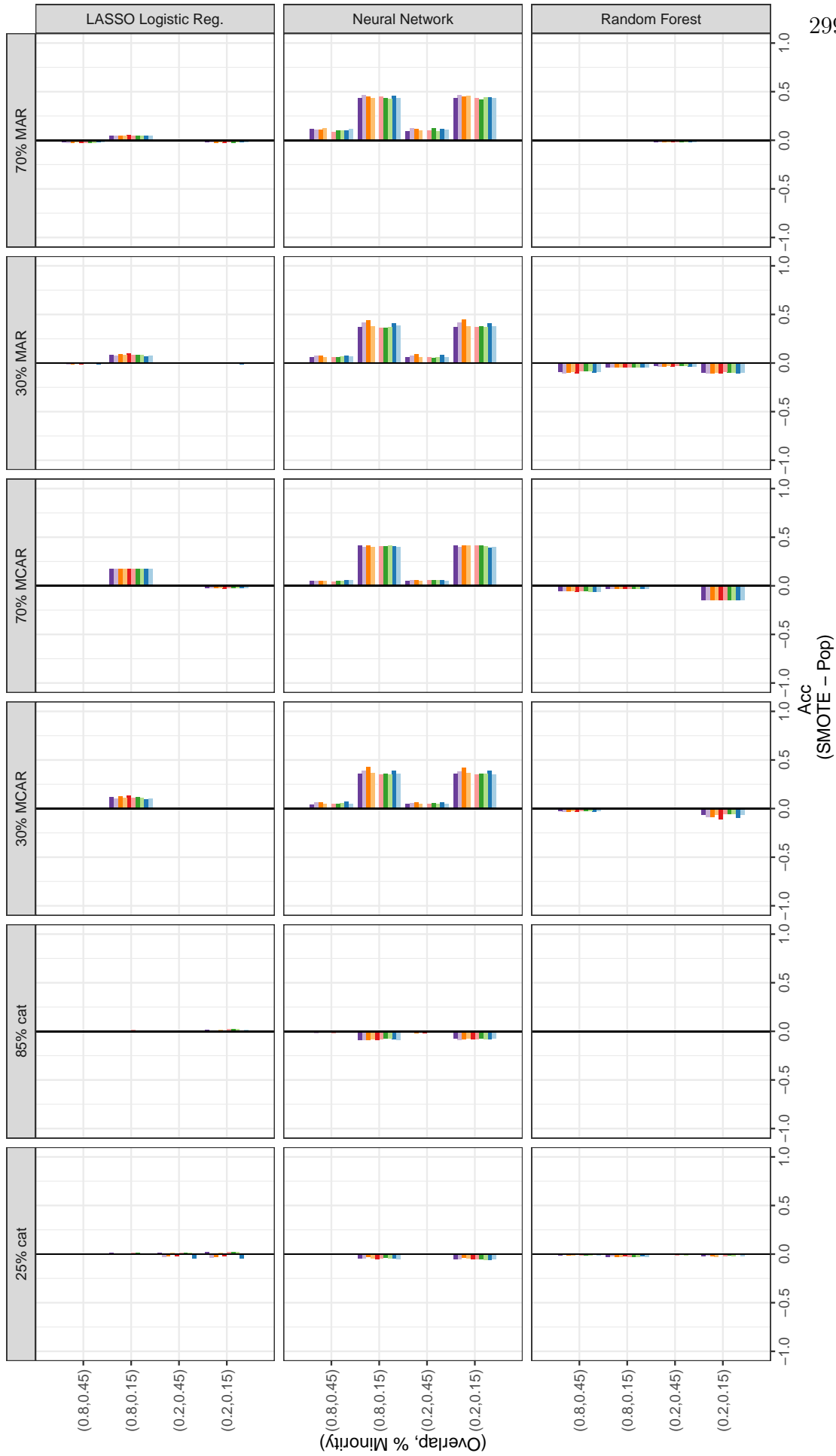
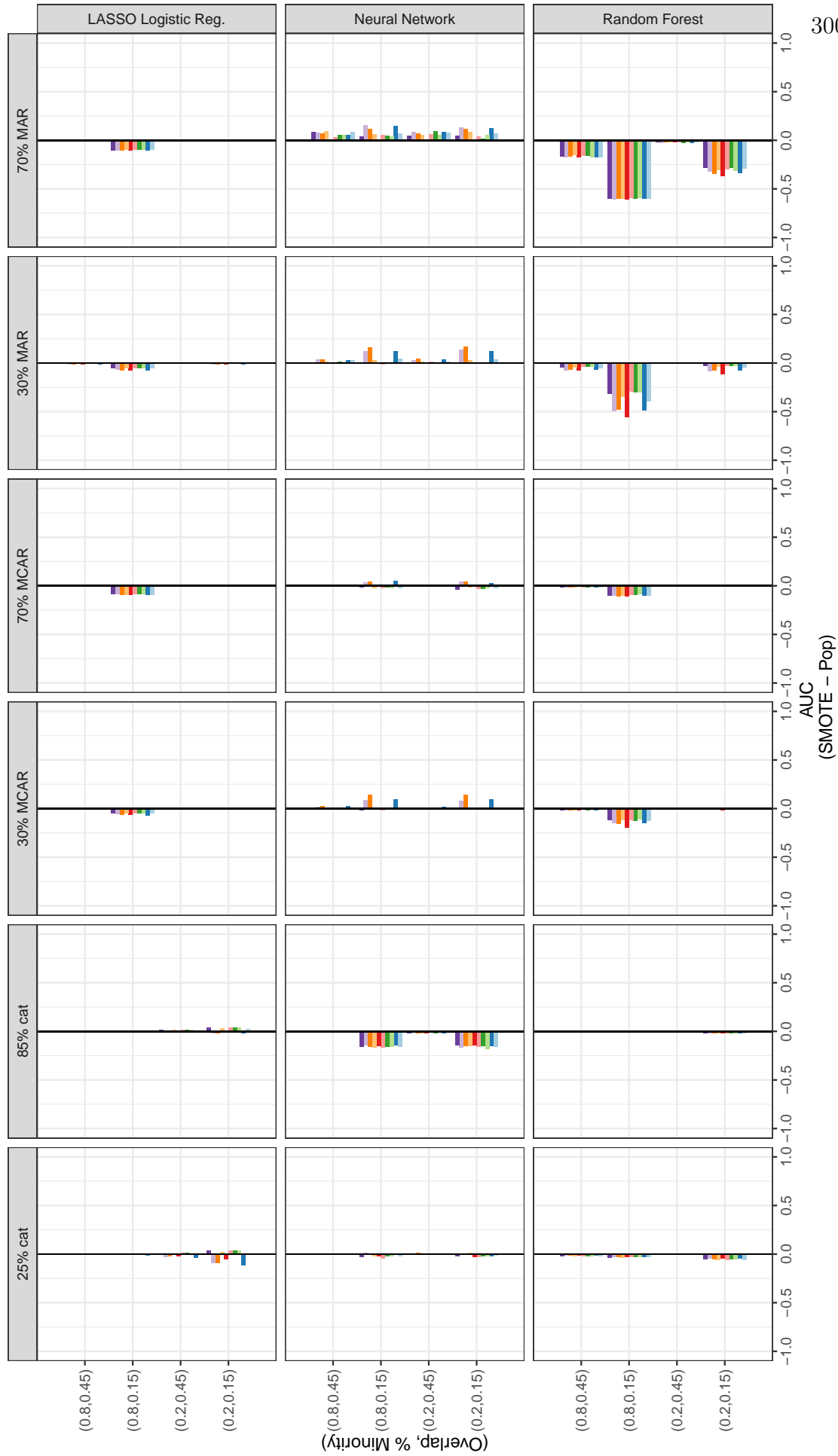
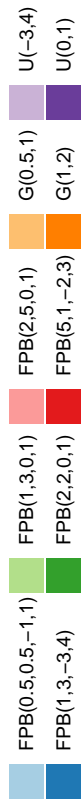
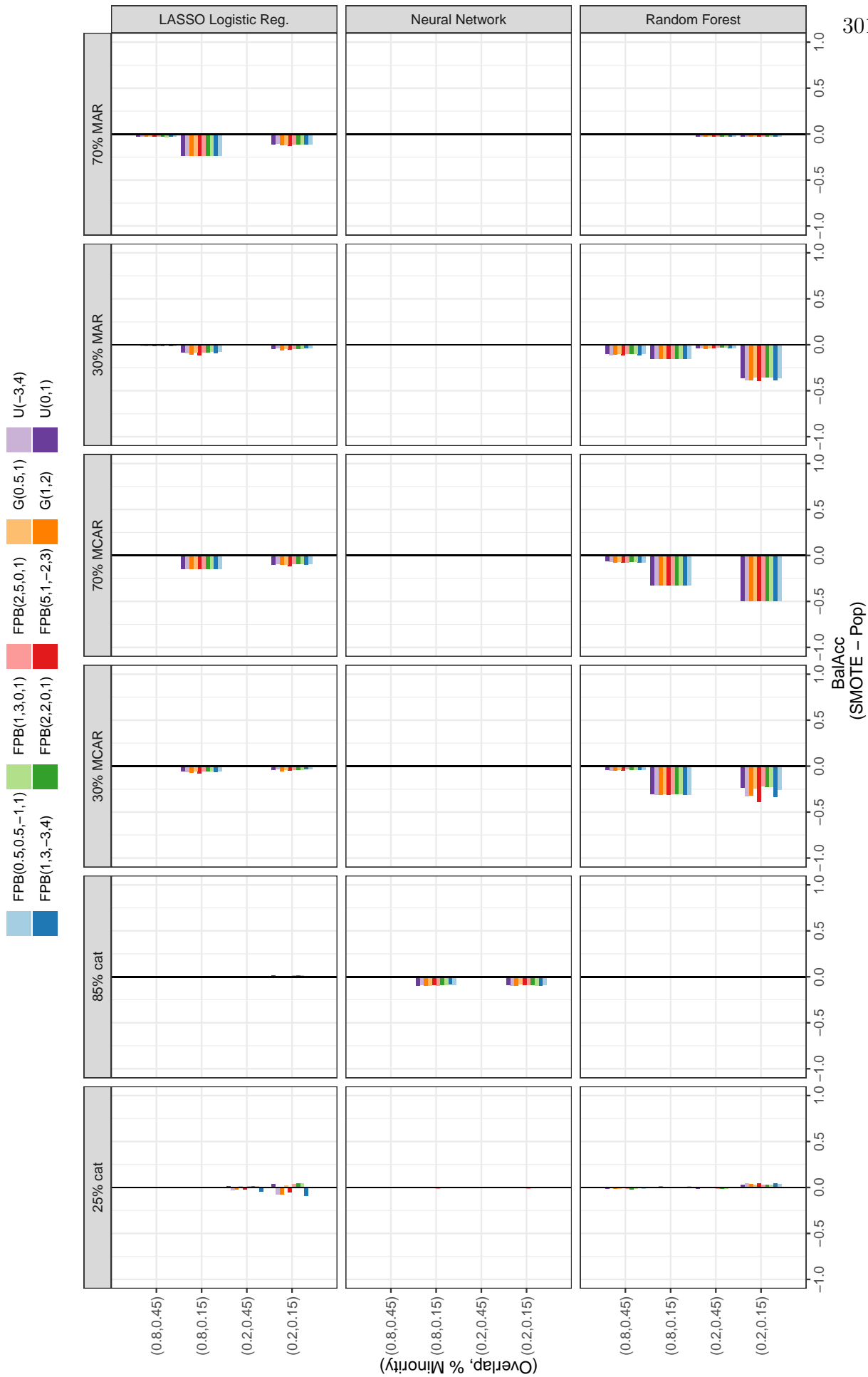
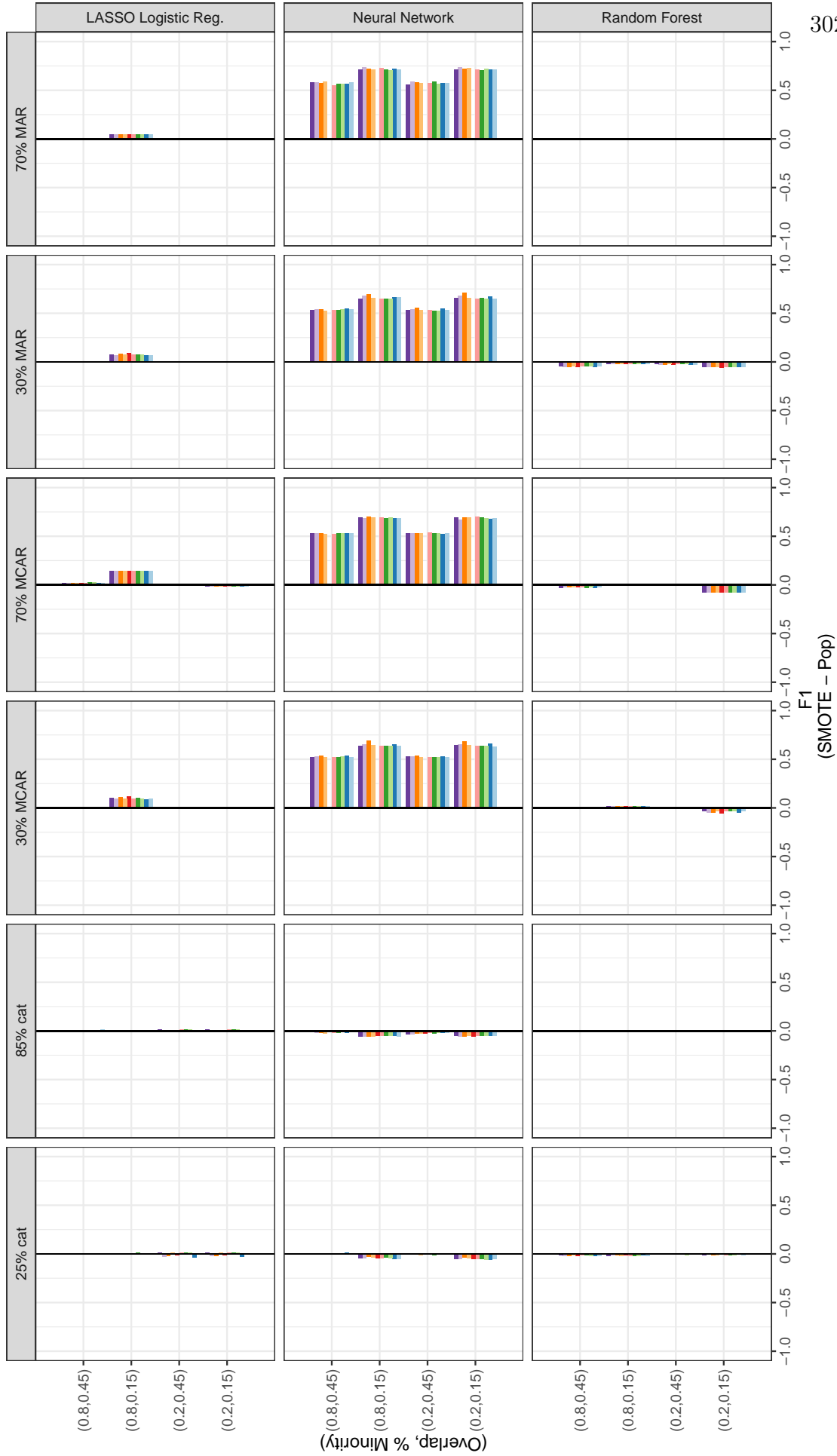


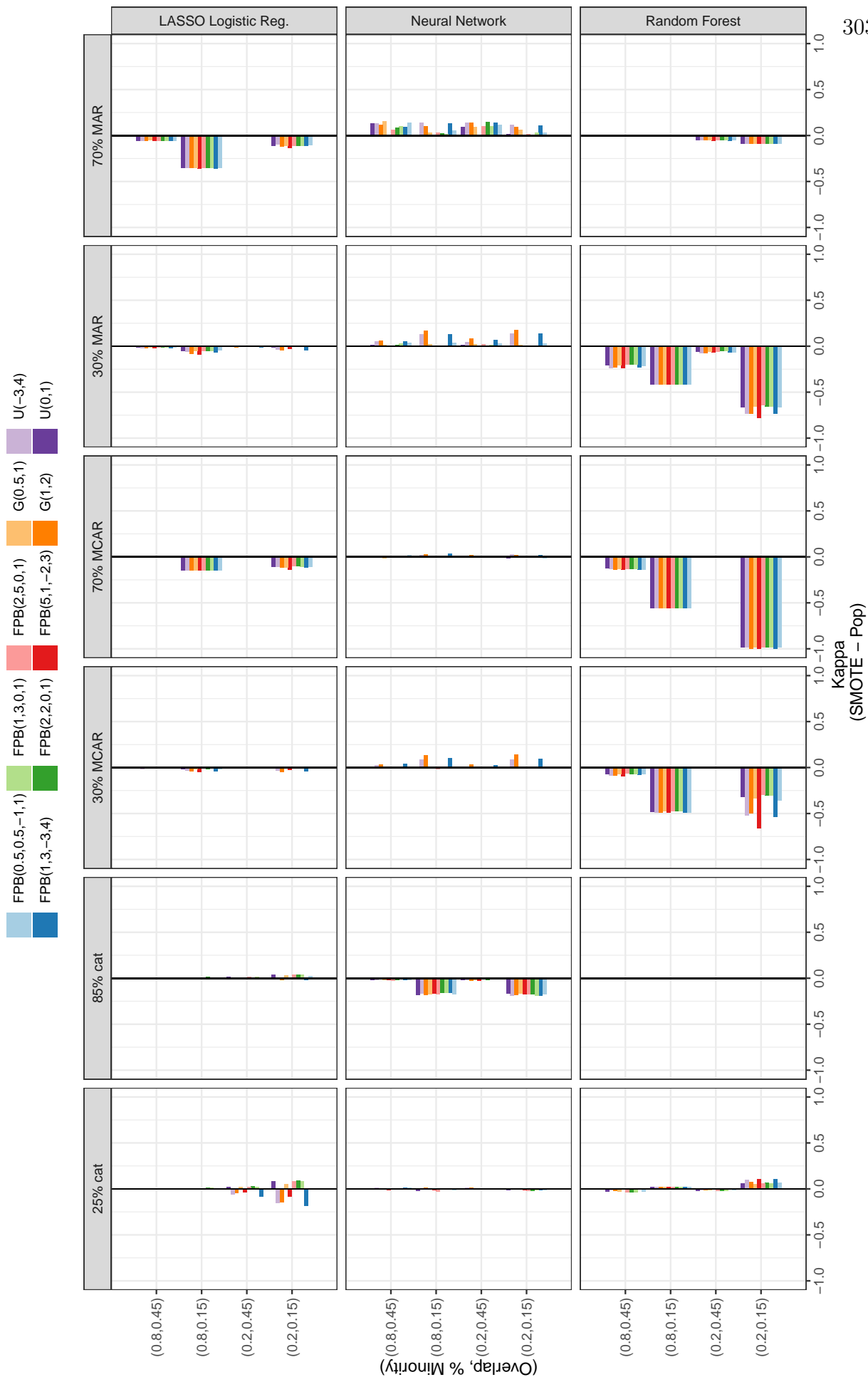
Figure B.4: Differences In Performance Metrics By Oversampling Method: Comparisons were made between S-SMOTE, SMOTE and Population oversampling. Differences were calculated with respect to data case, model, amount of overlap and imbalance, and which distribution was used for w .

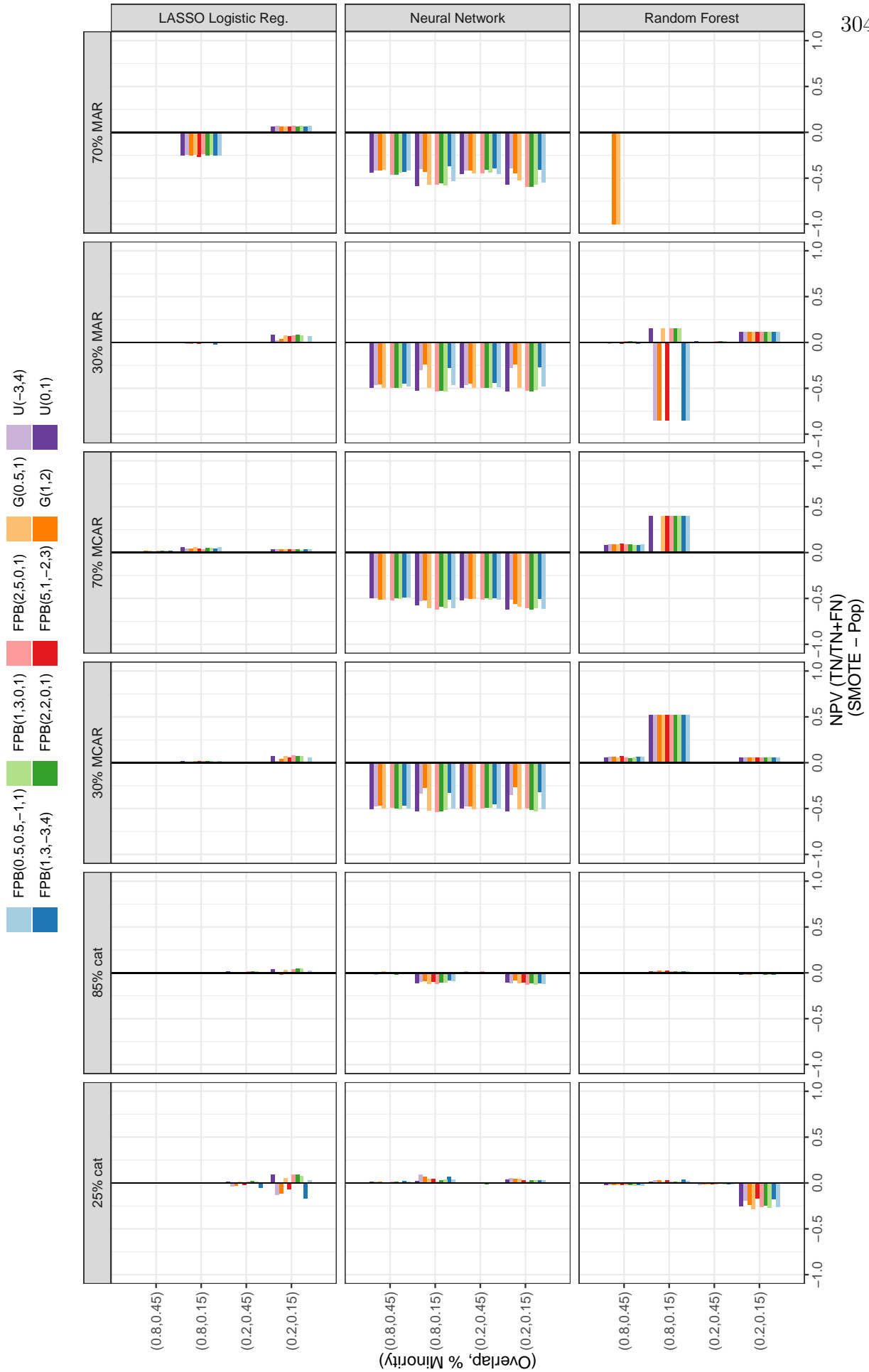


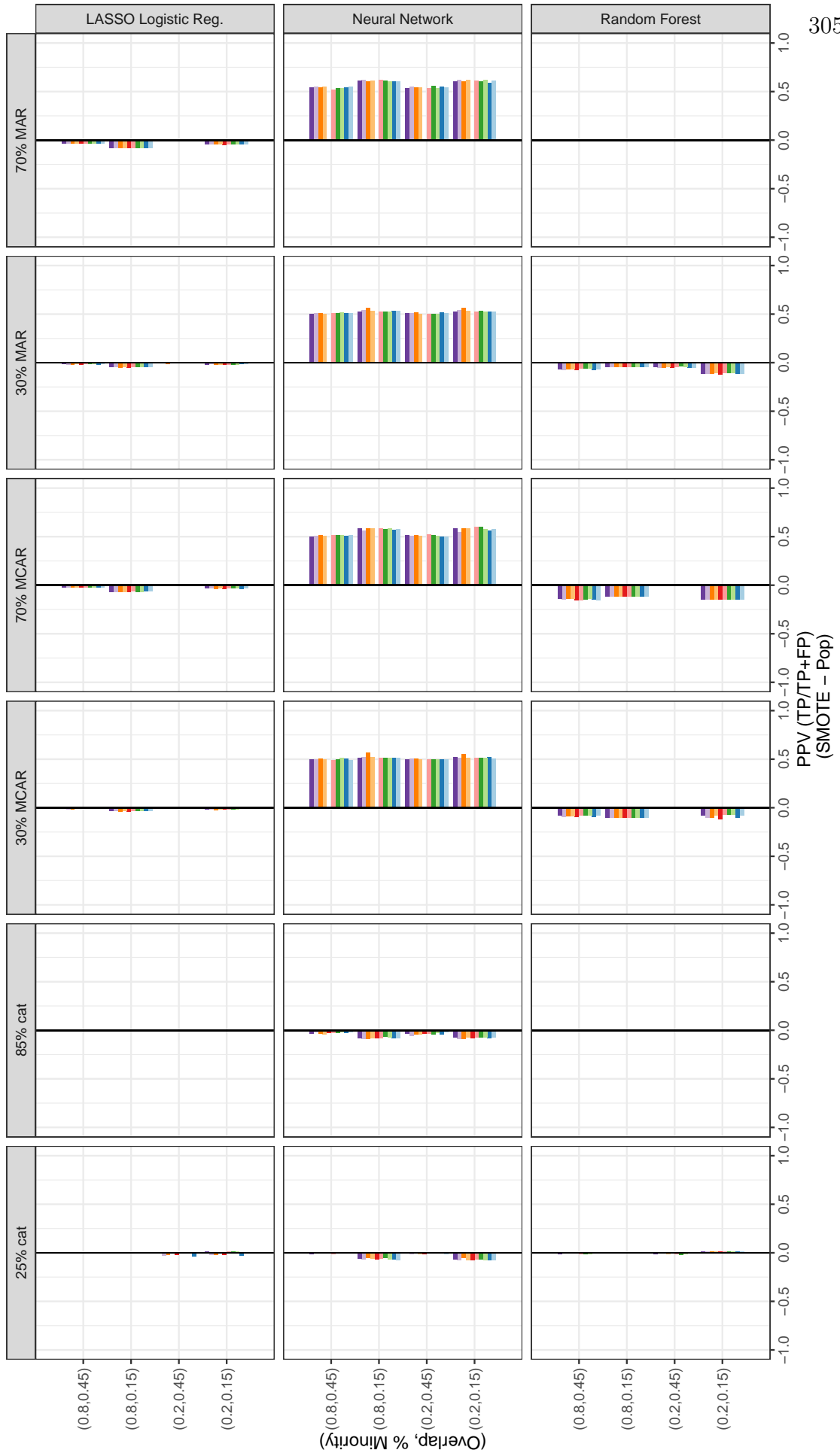


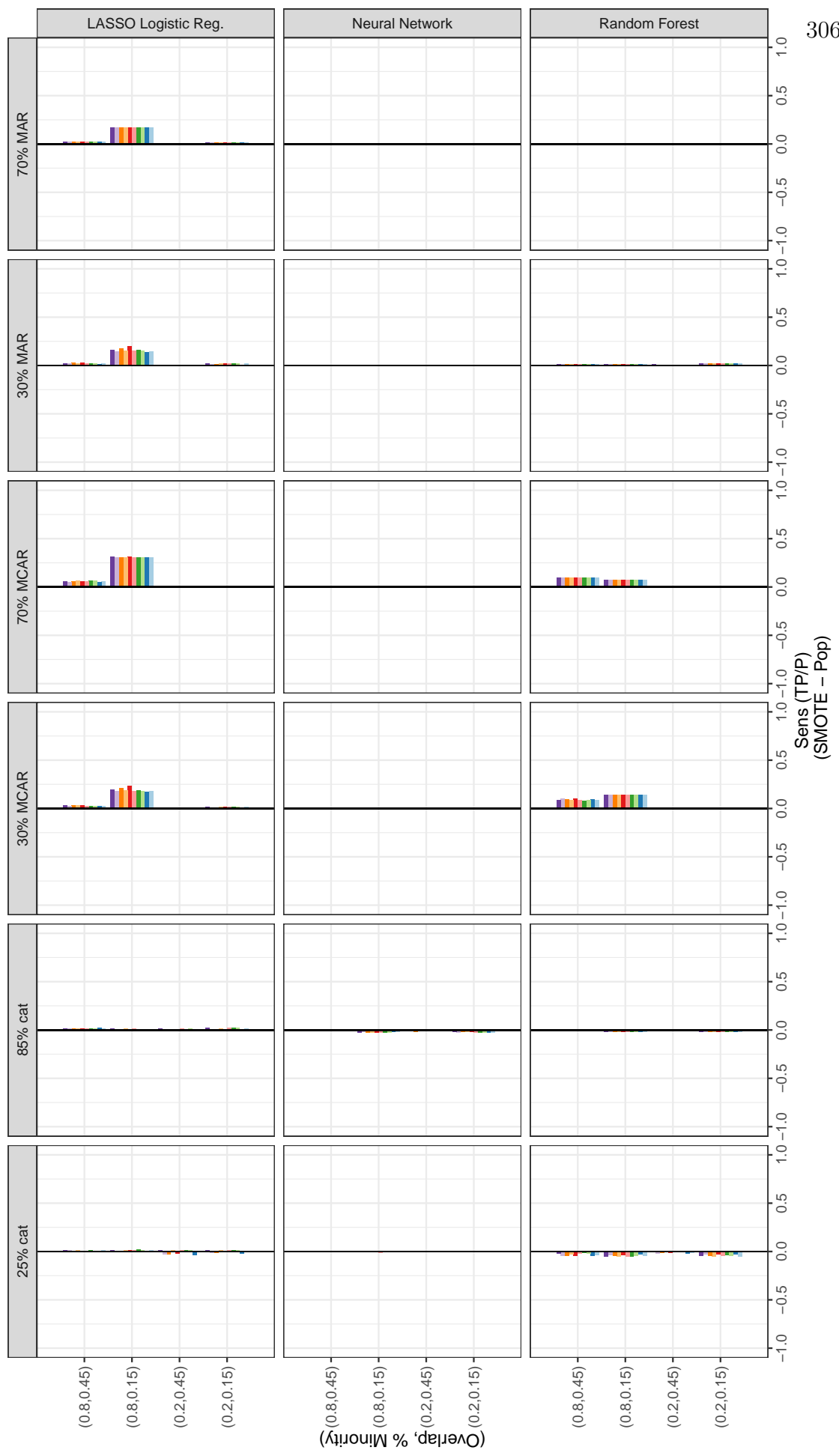
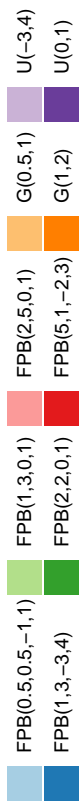


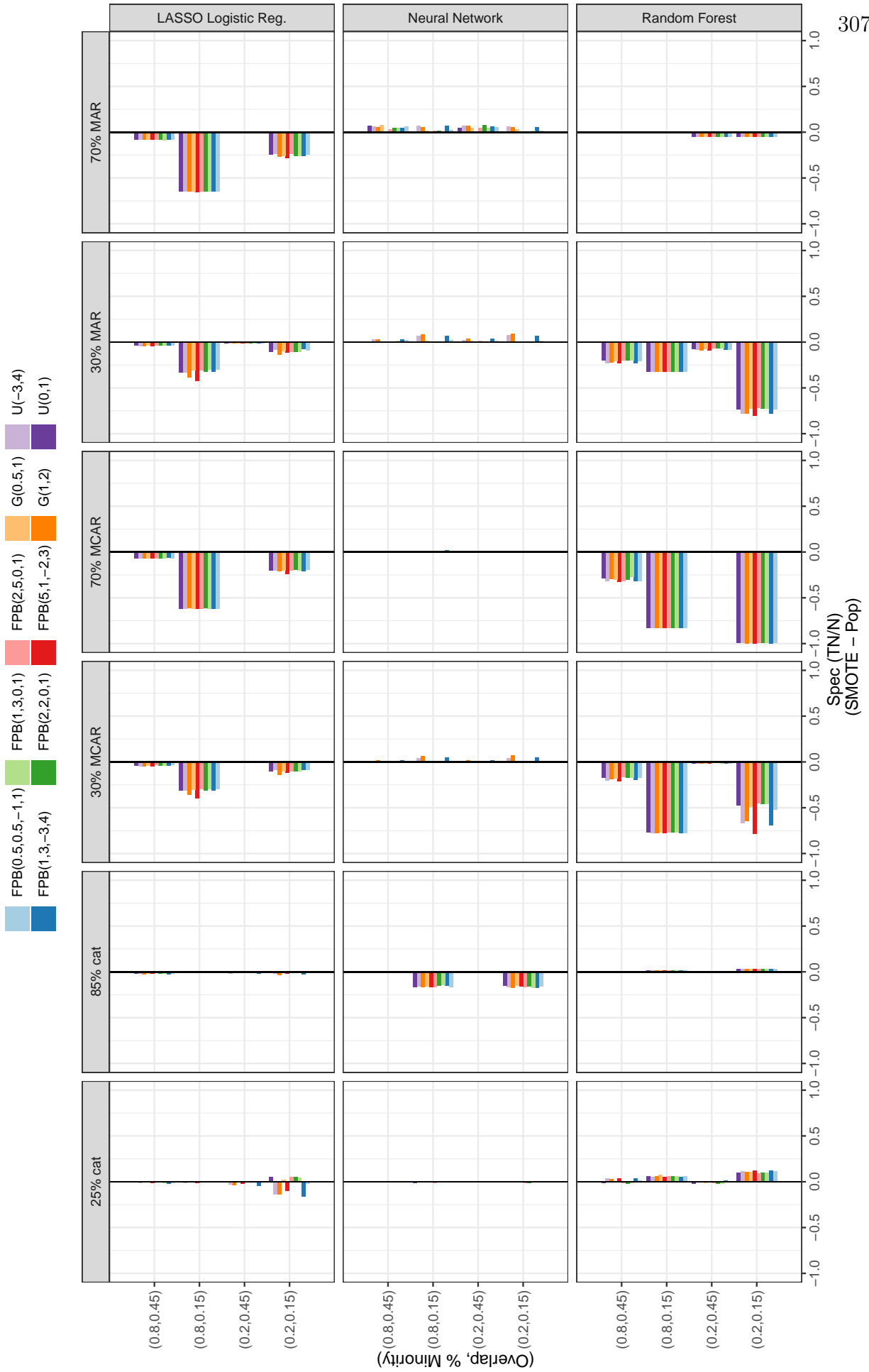


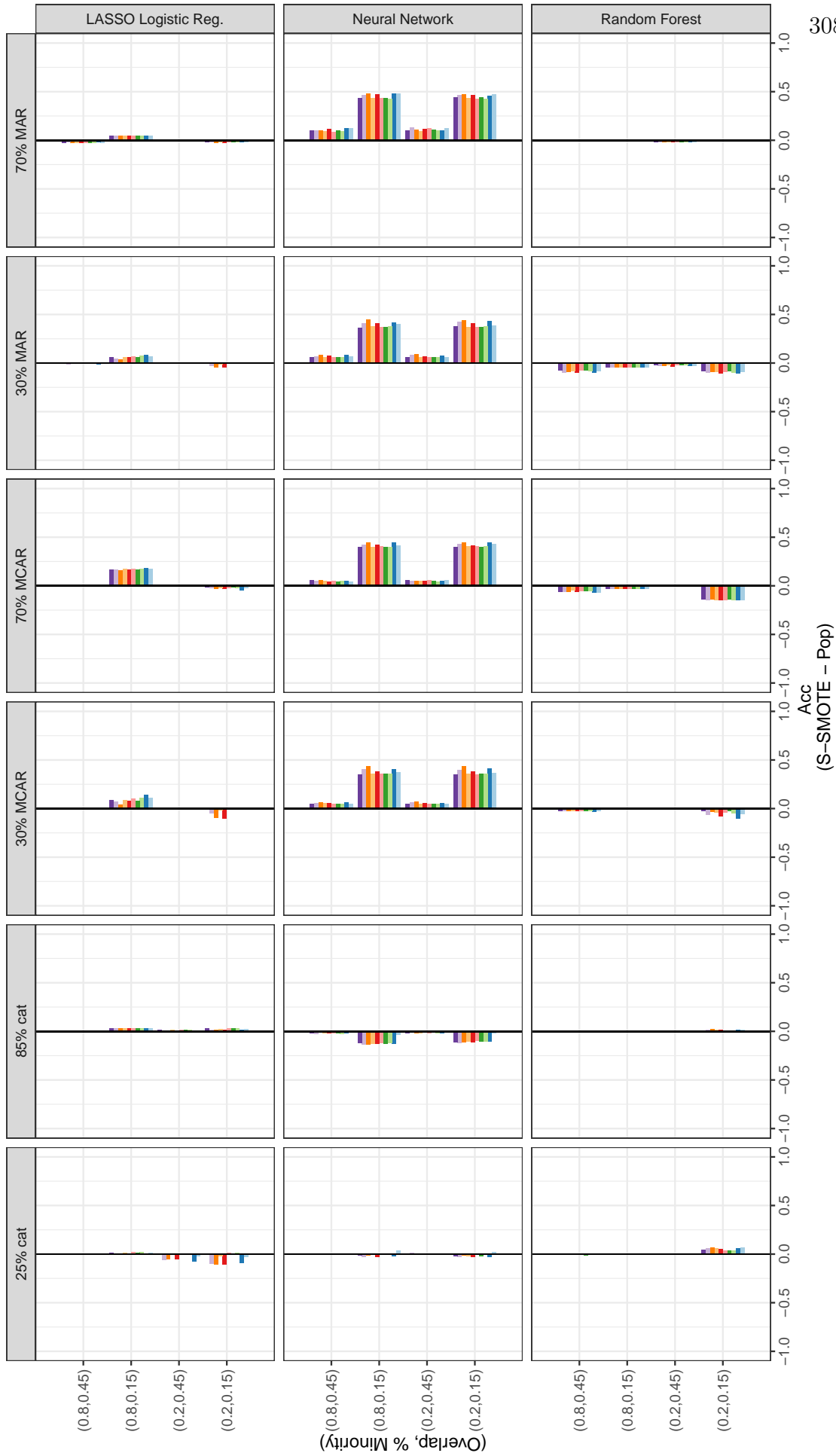


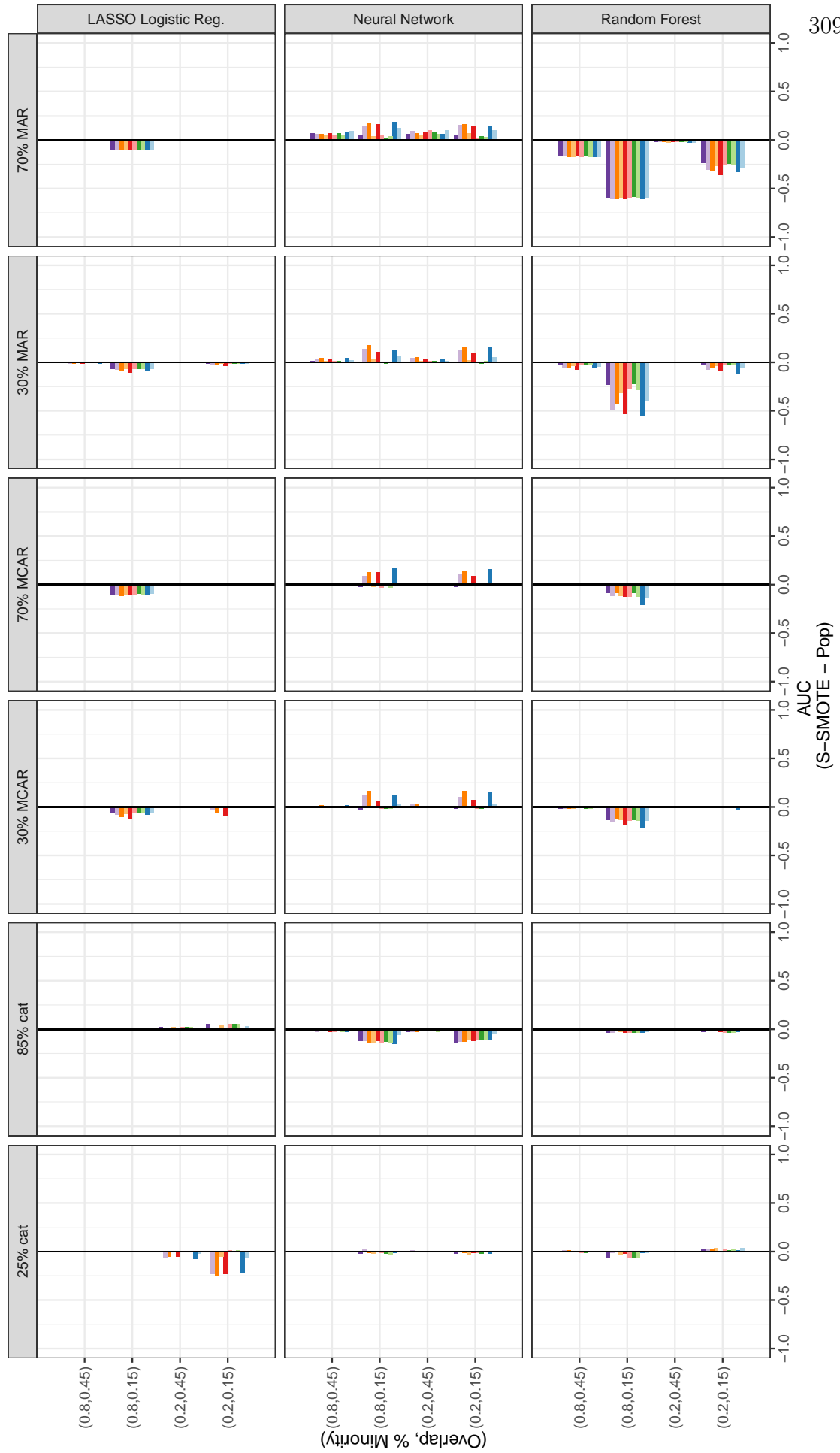
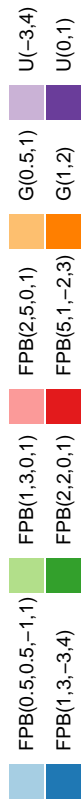


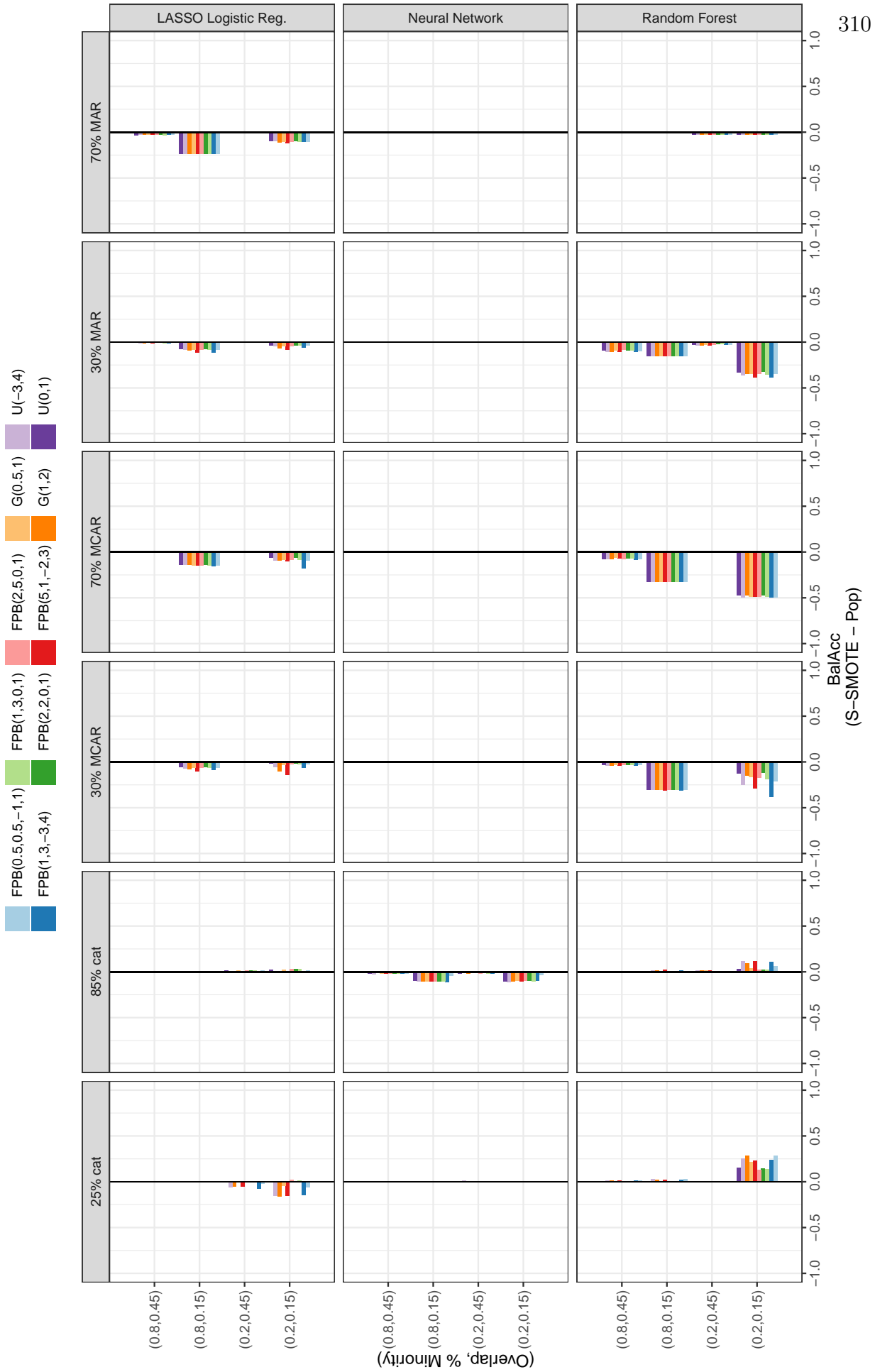


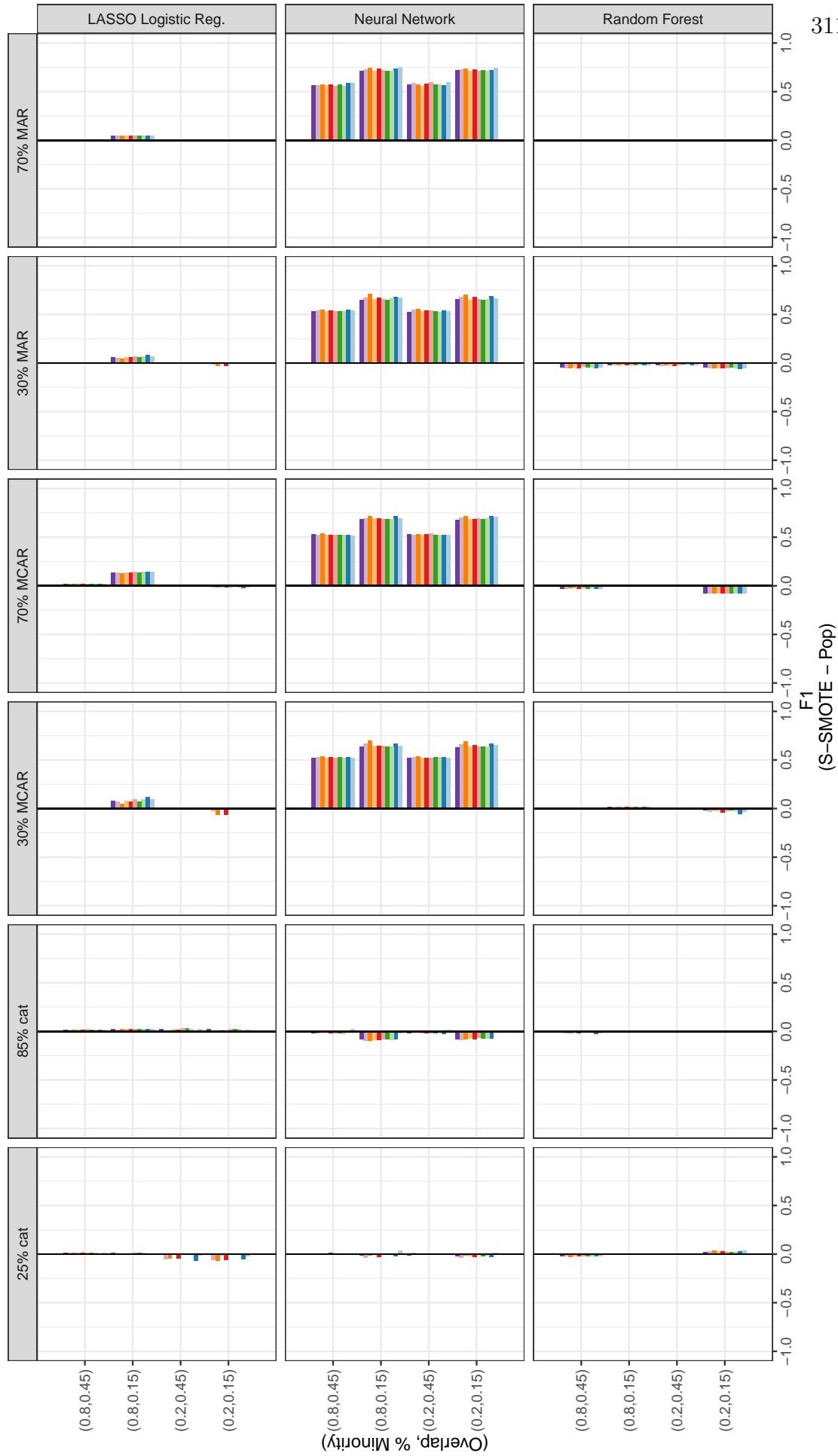
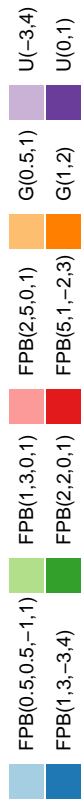


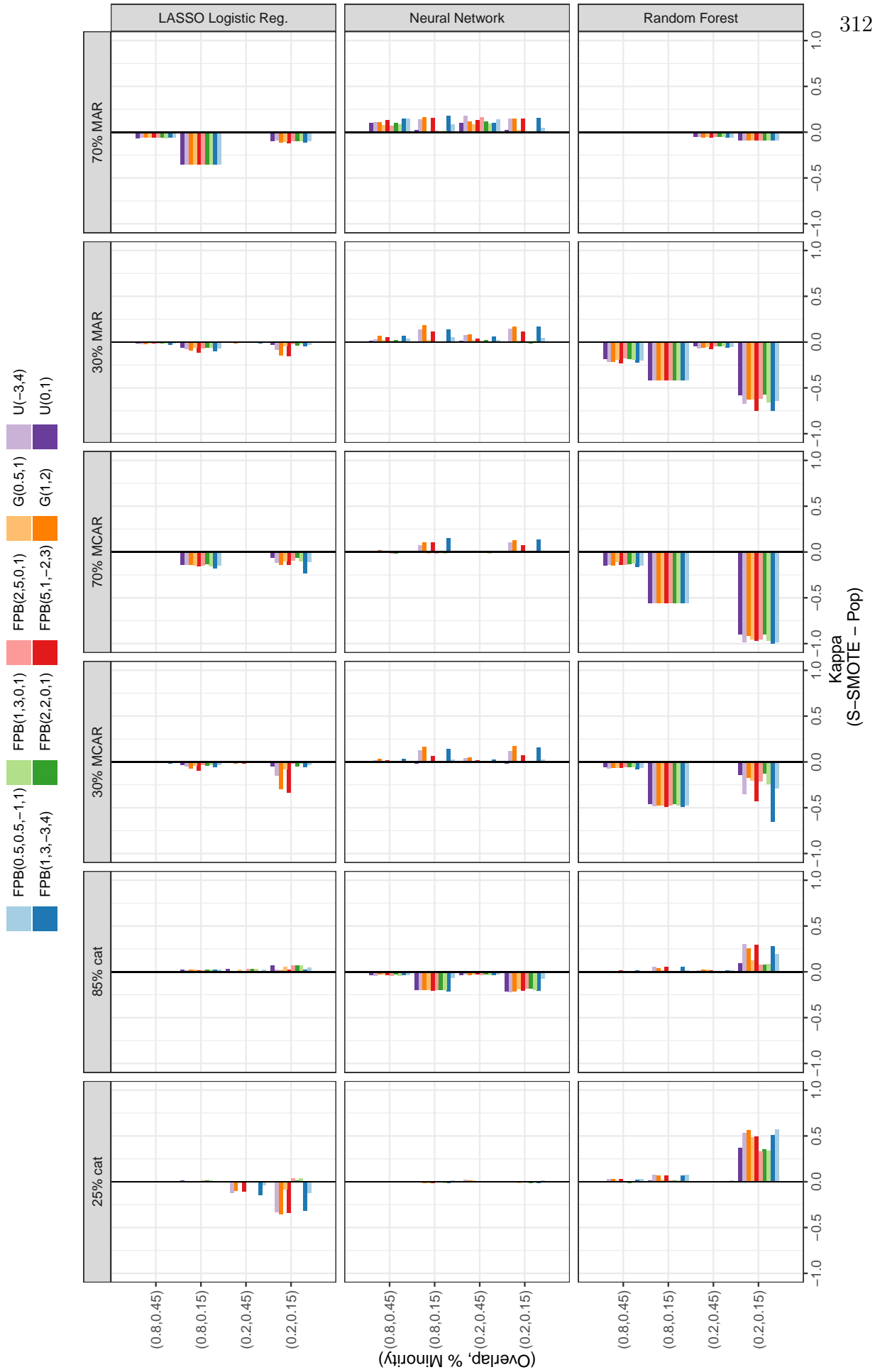


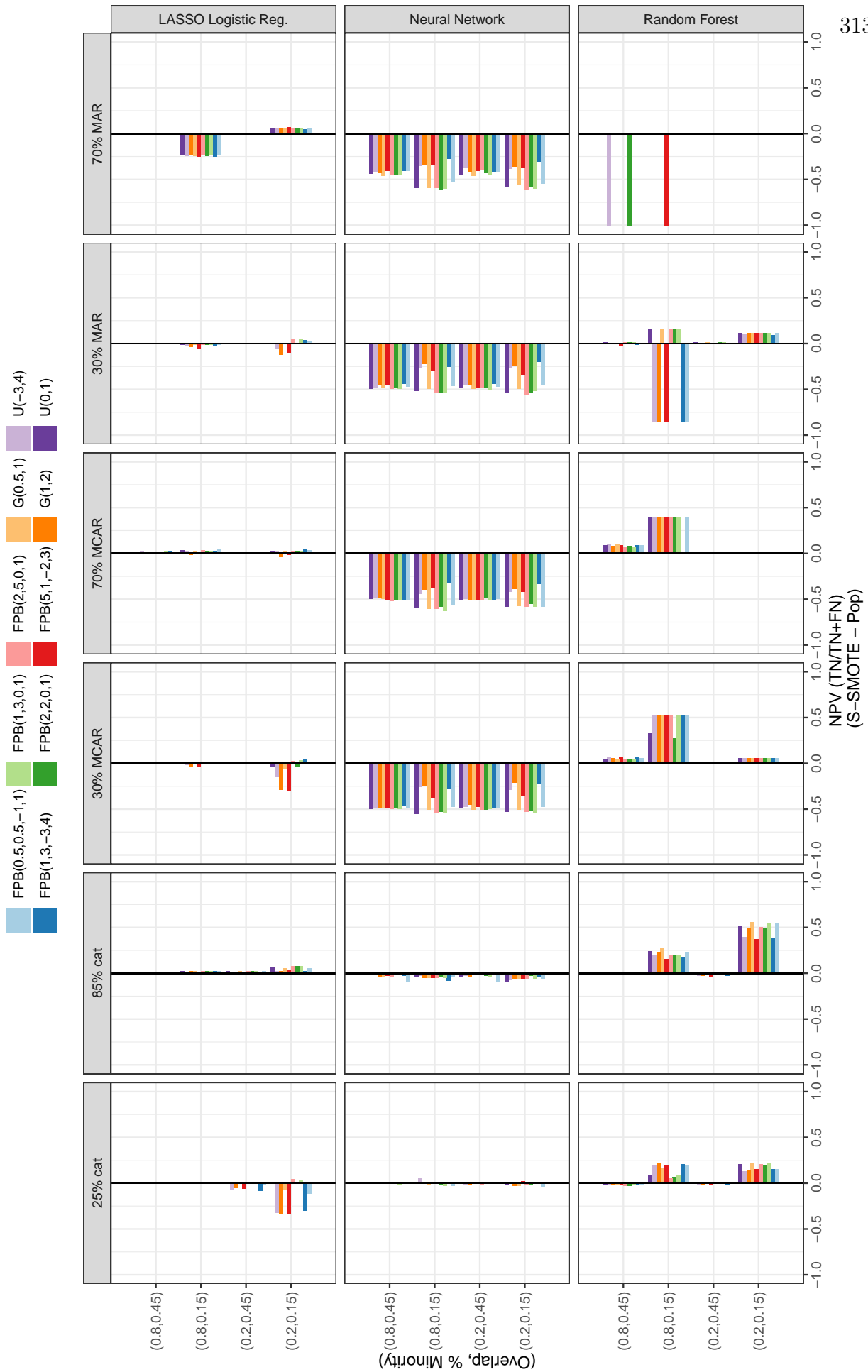


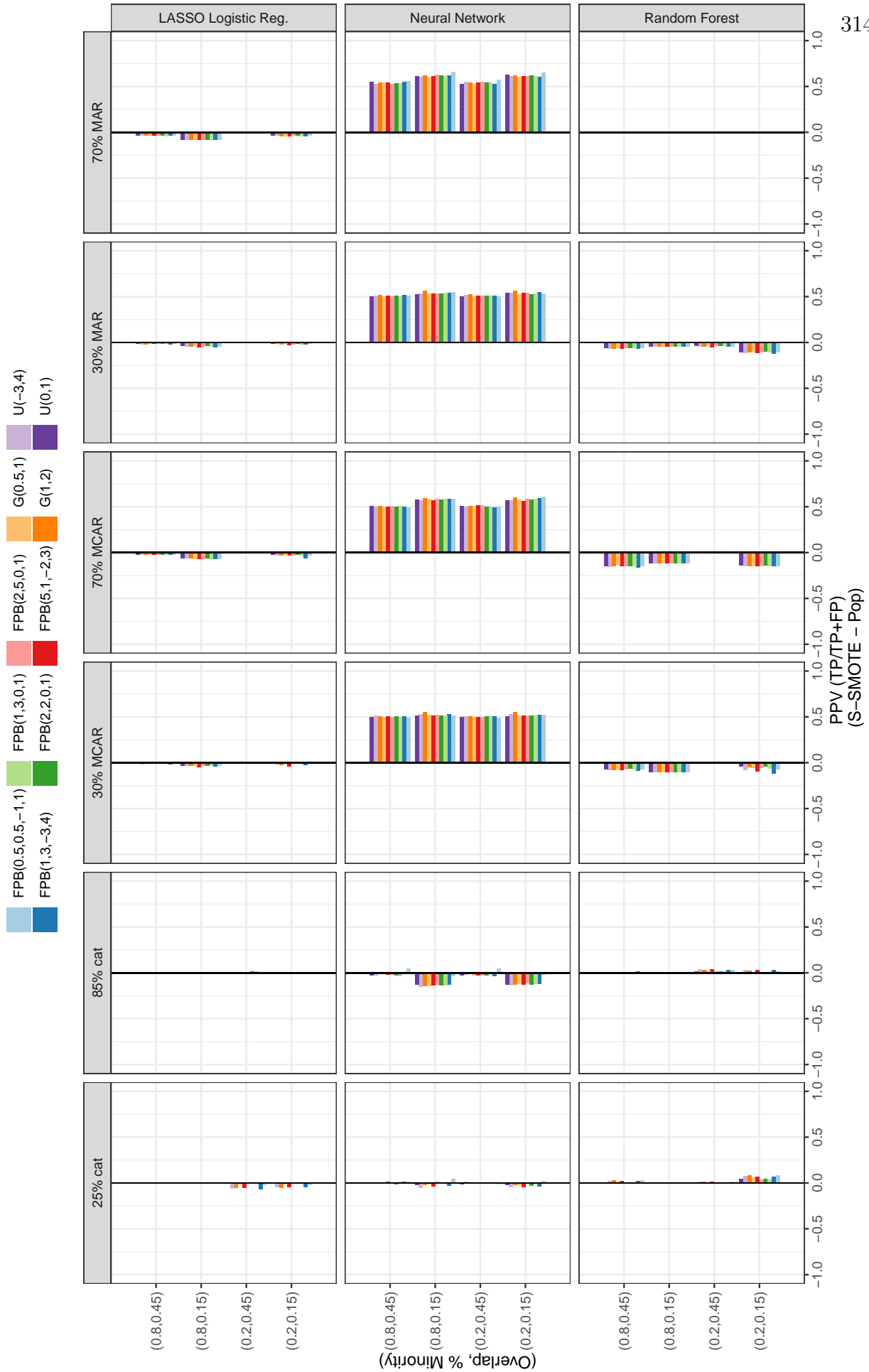


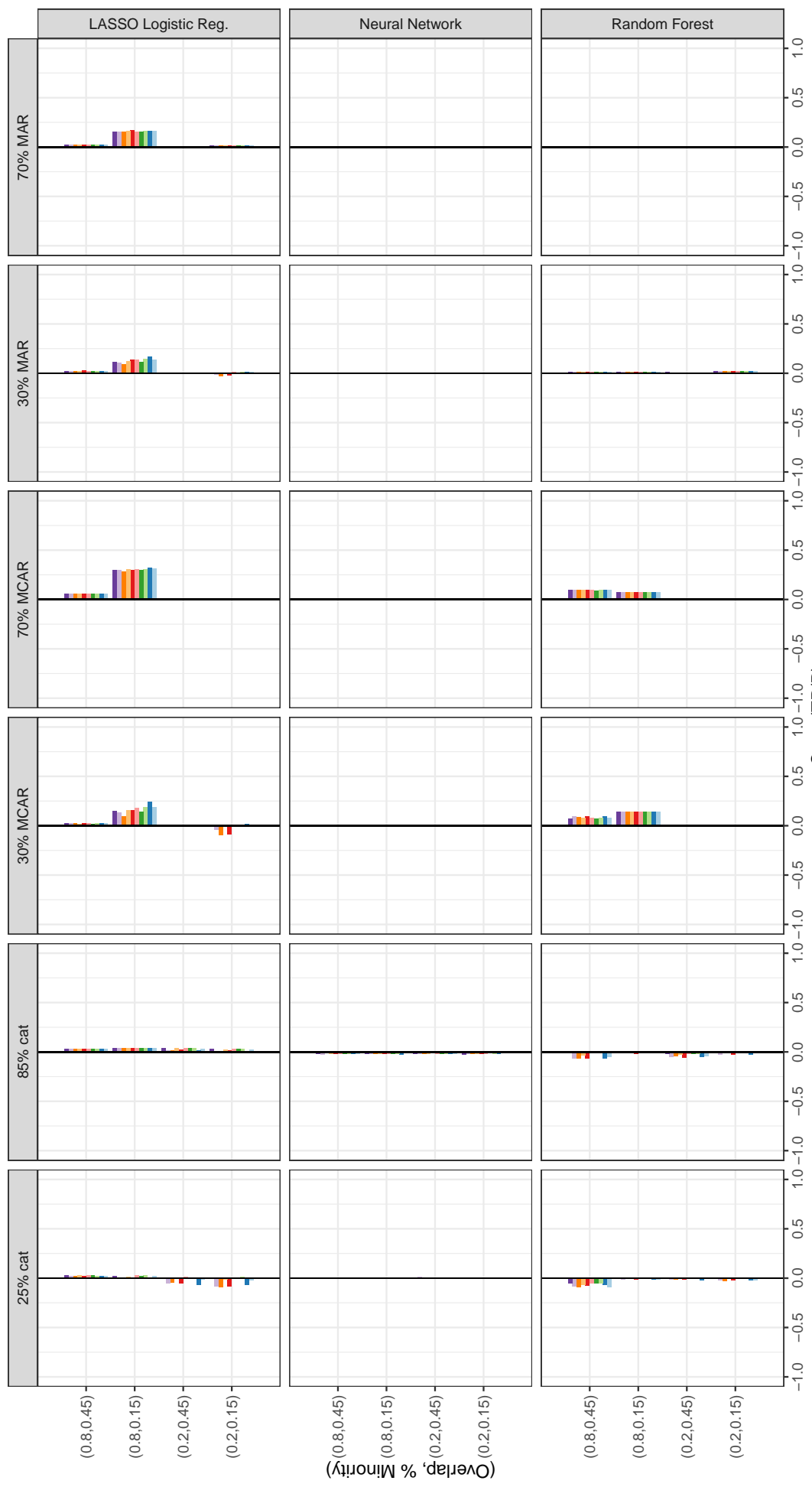
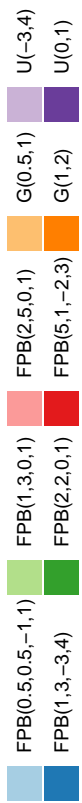


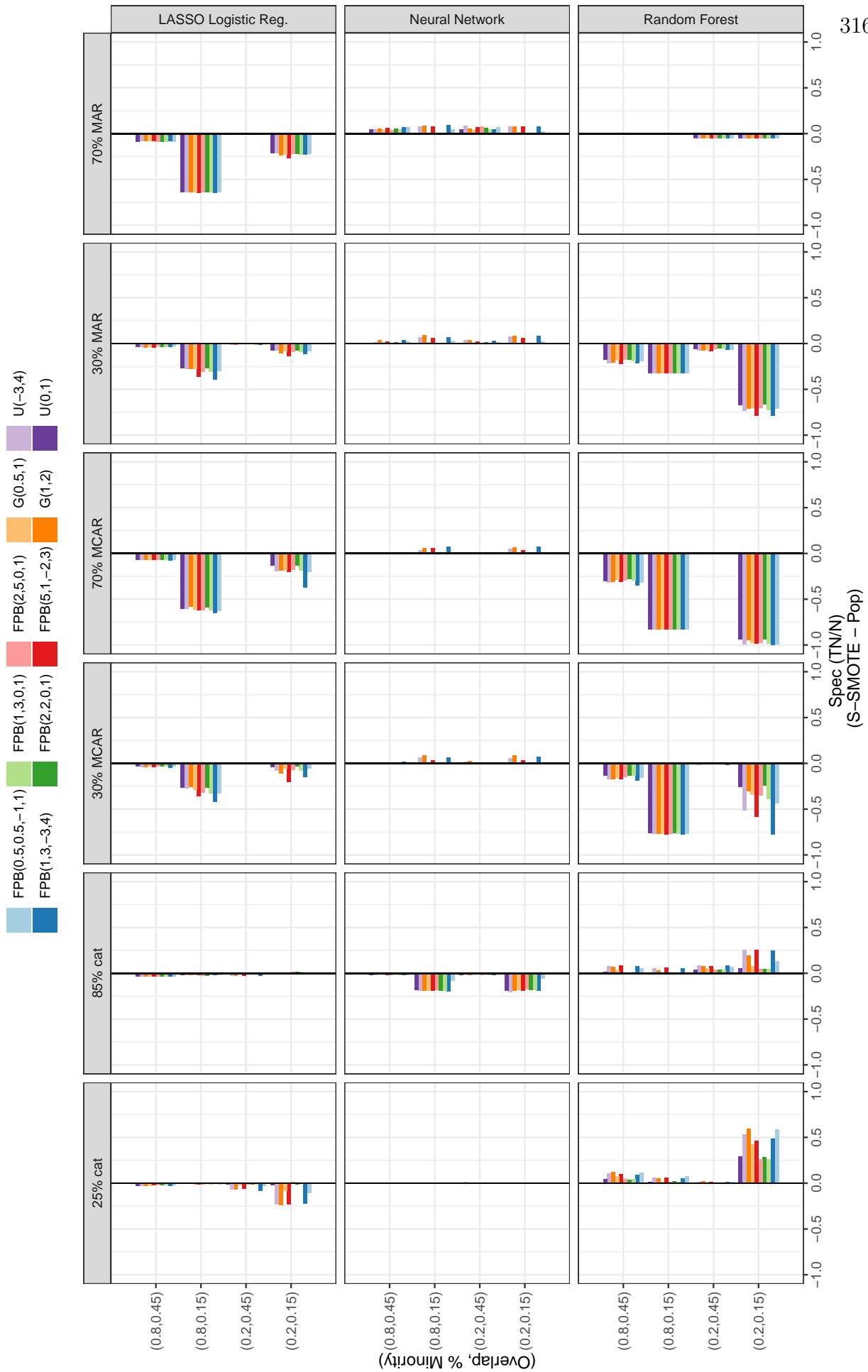


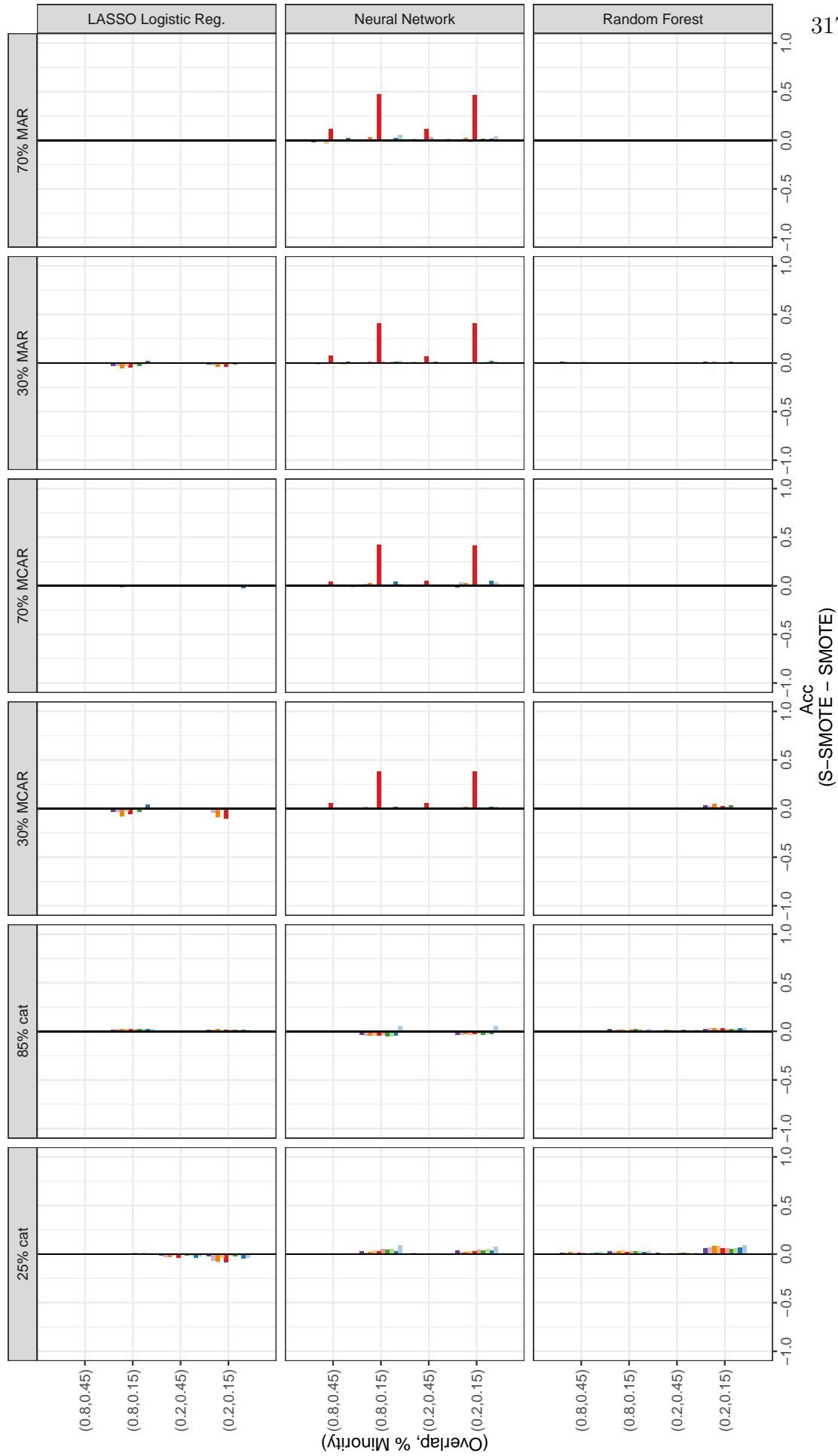
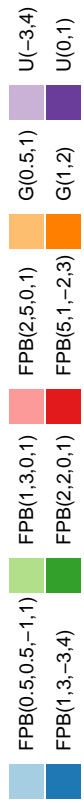


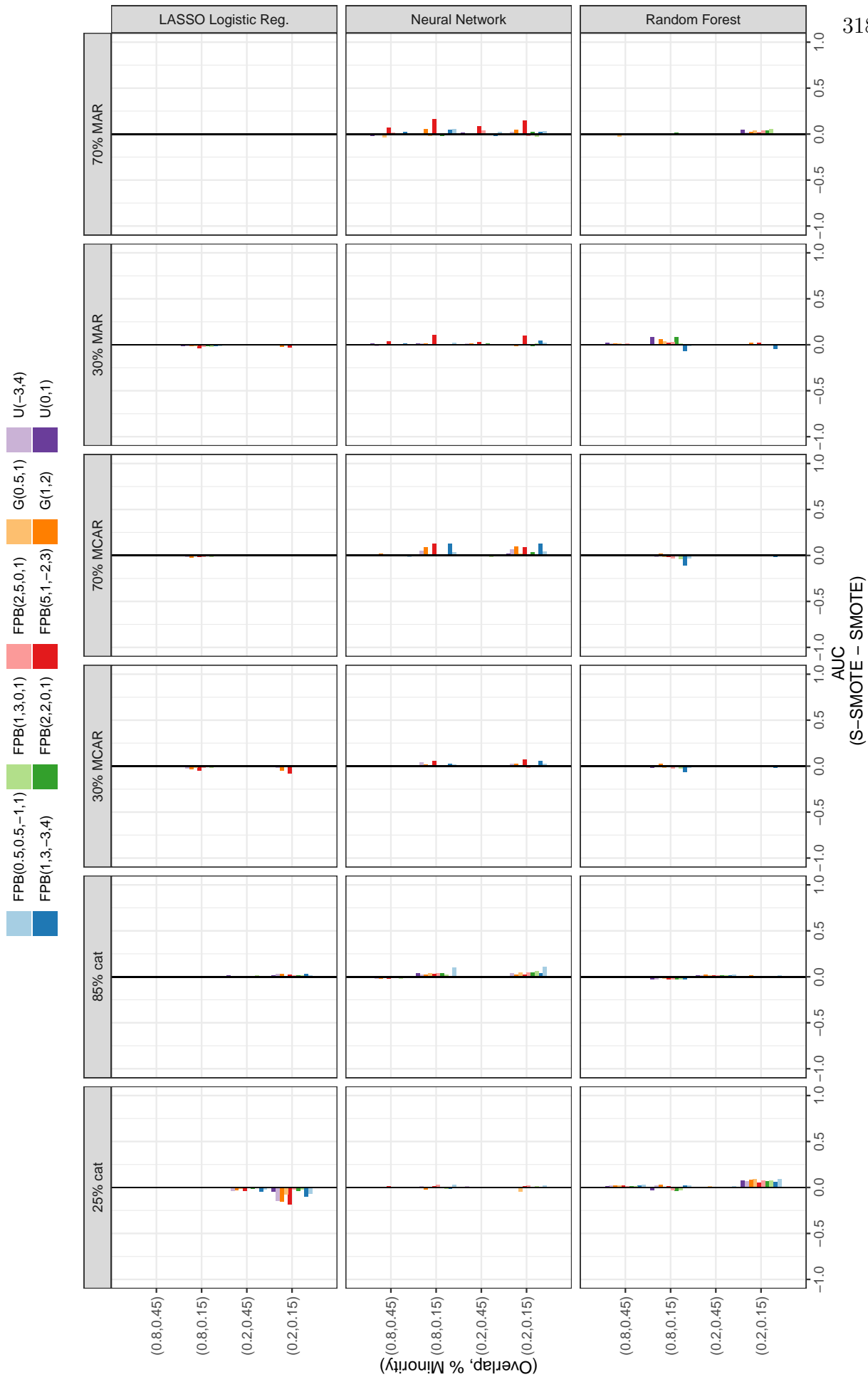


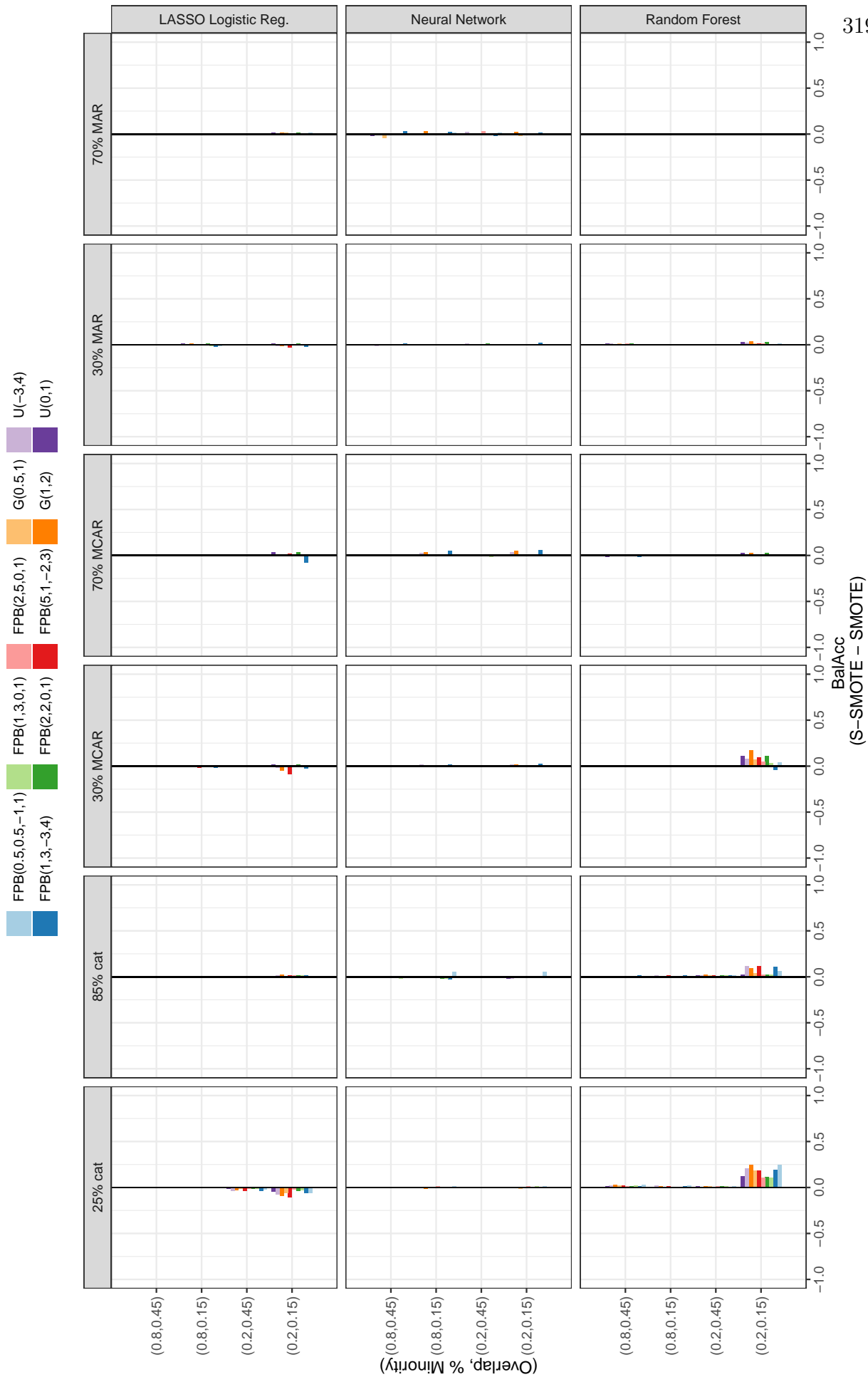


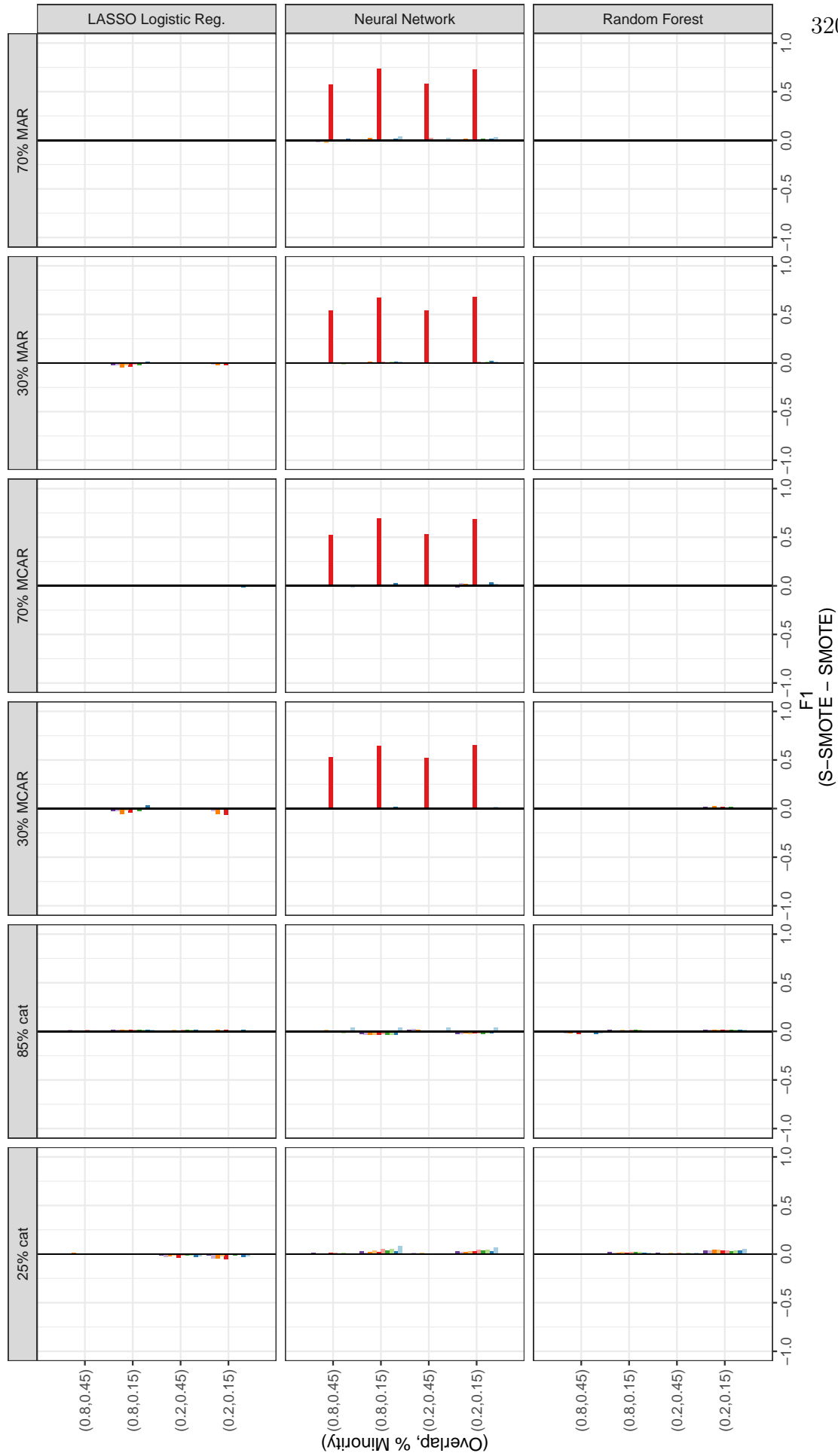
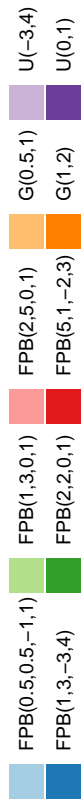


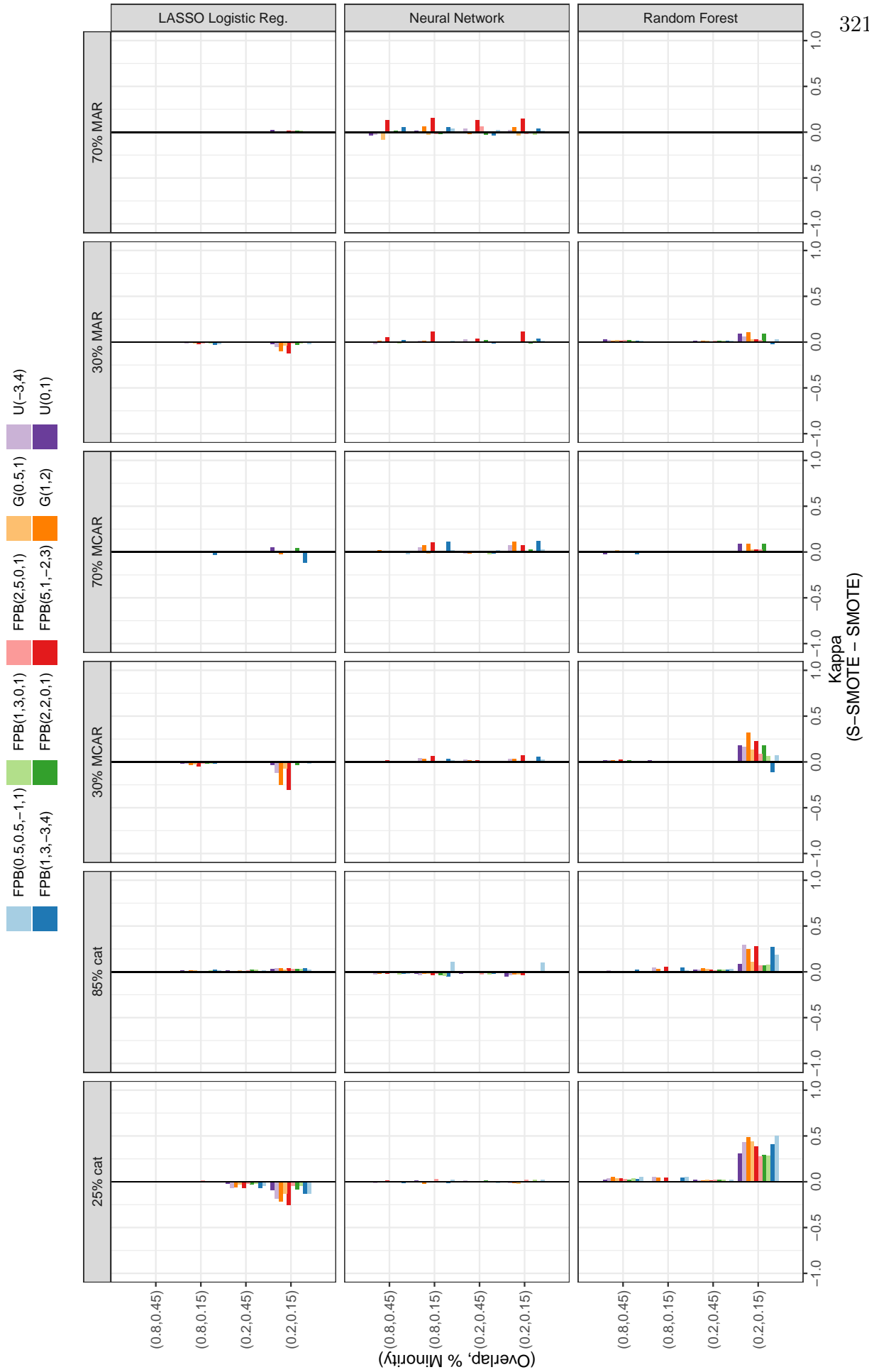


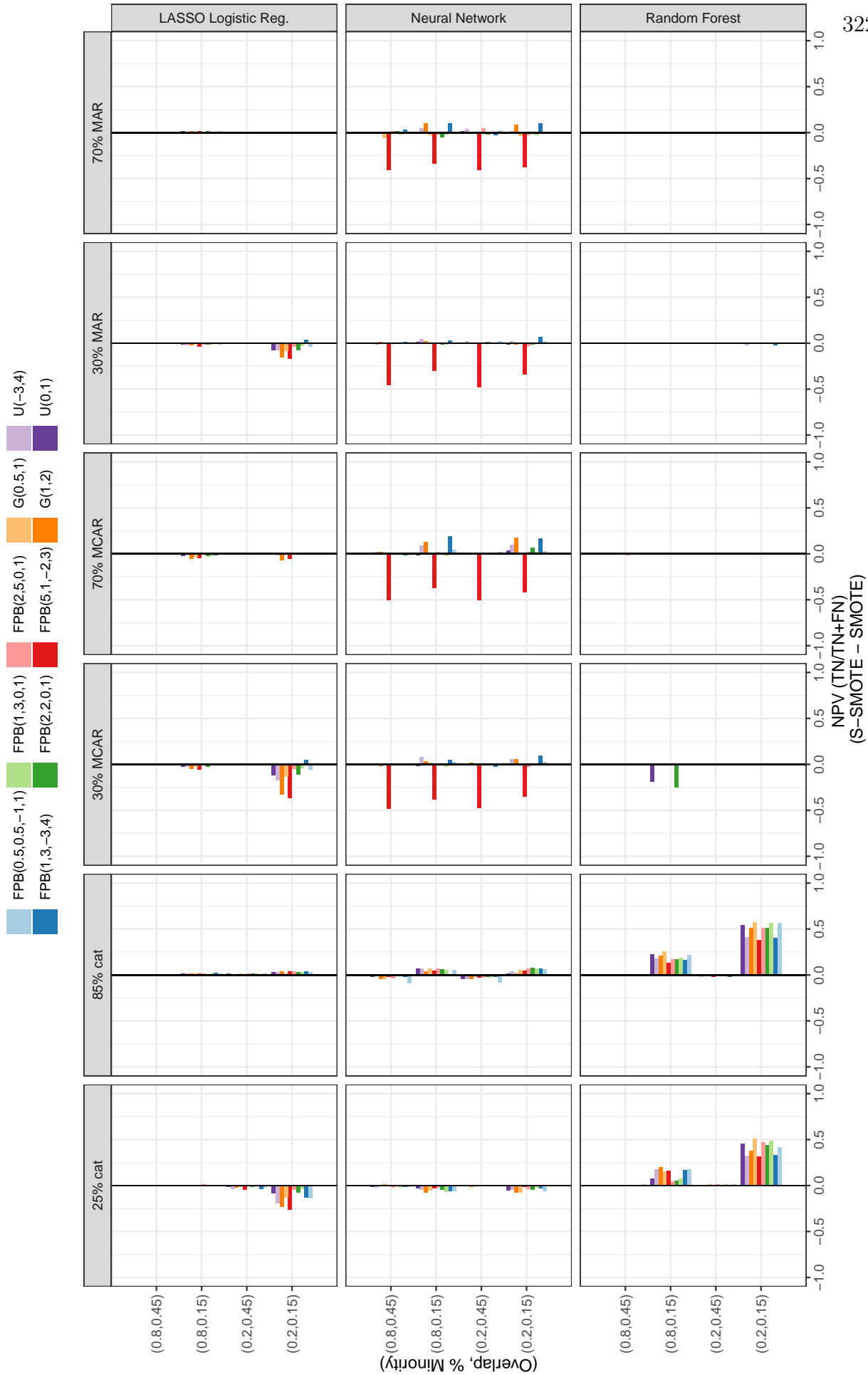


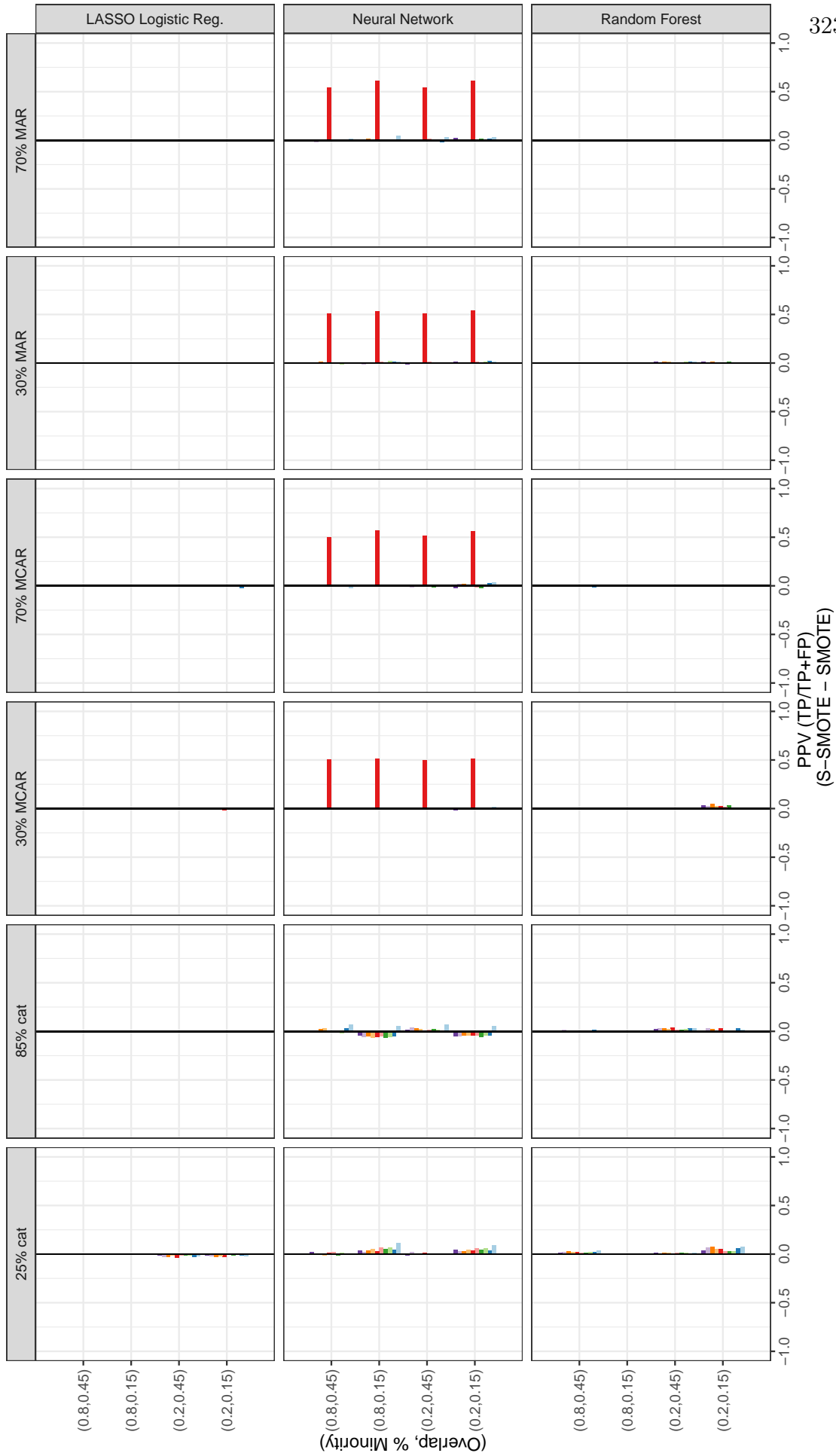


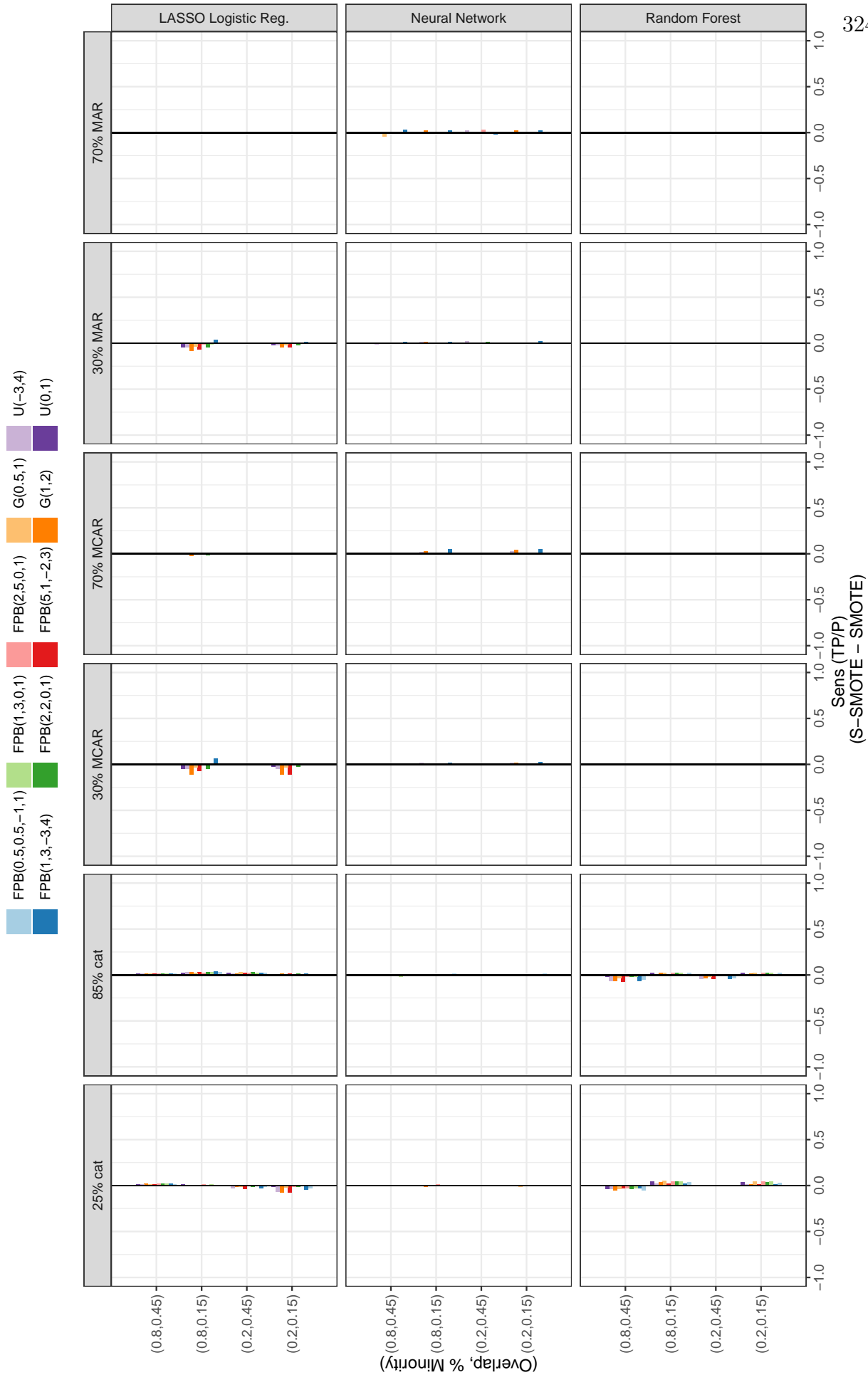












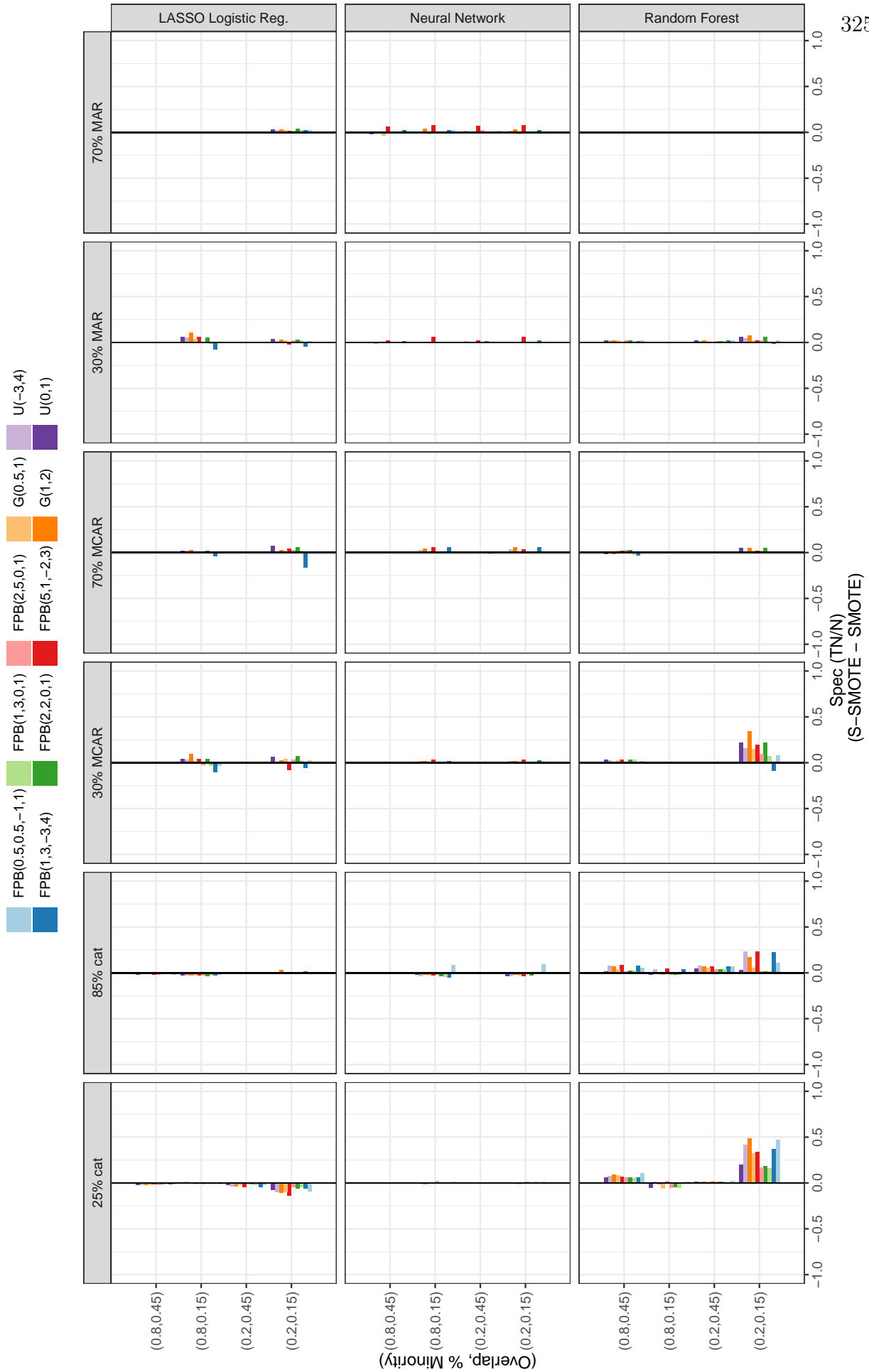
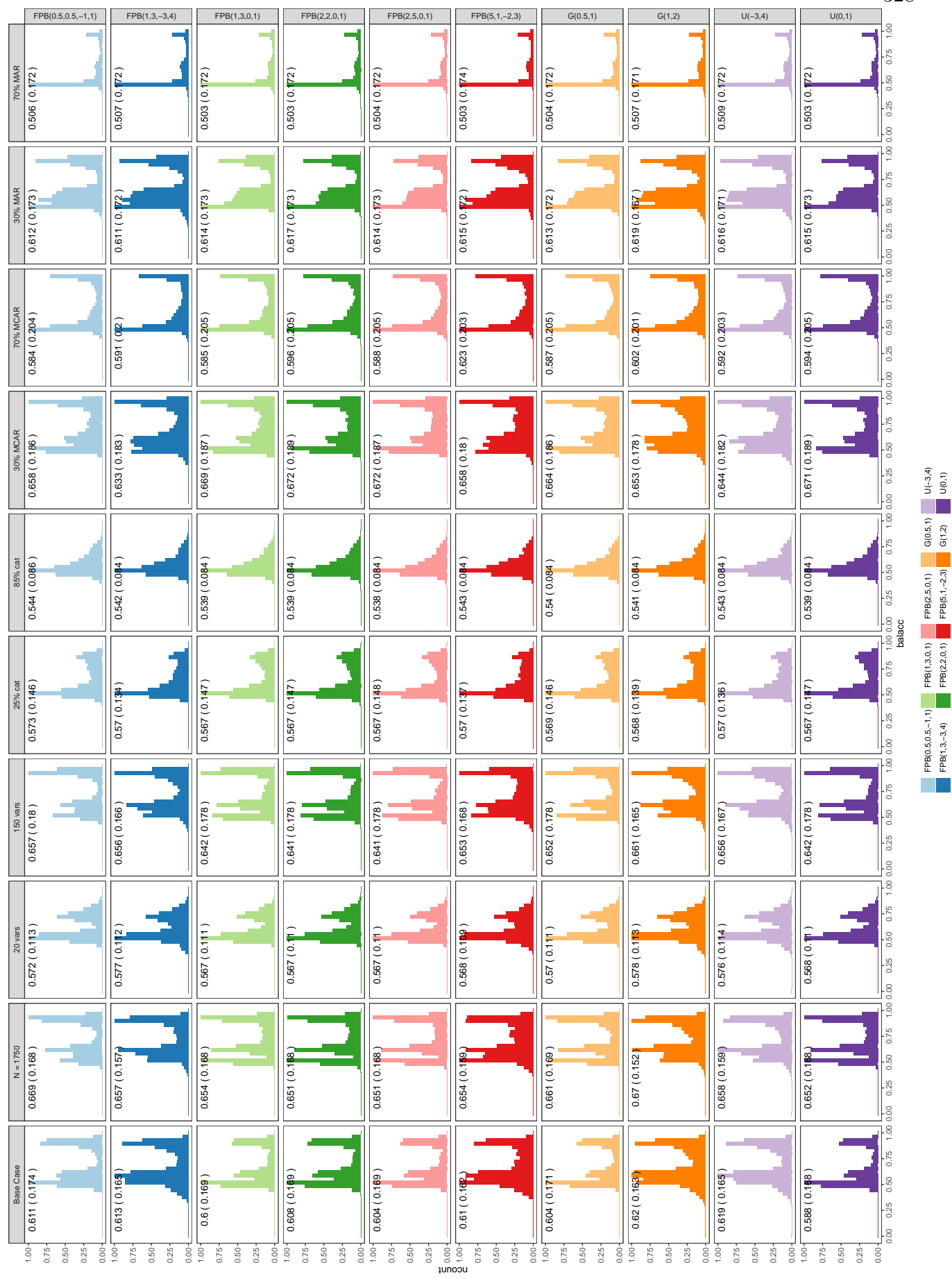


Figure B.5: Distribution For Performance Metrics By Distribution for w and Data Scenario: Distributions of each performance metrics are given aggregated over all characteristics except the data scenario and distribution used for w . These were used to determine if there were any changes in performance due simply to the distribution used for w and what data scenarios these occurred in.

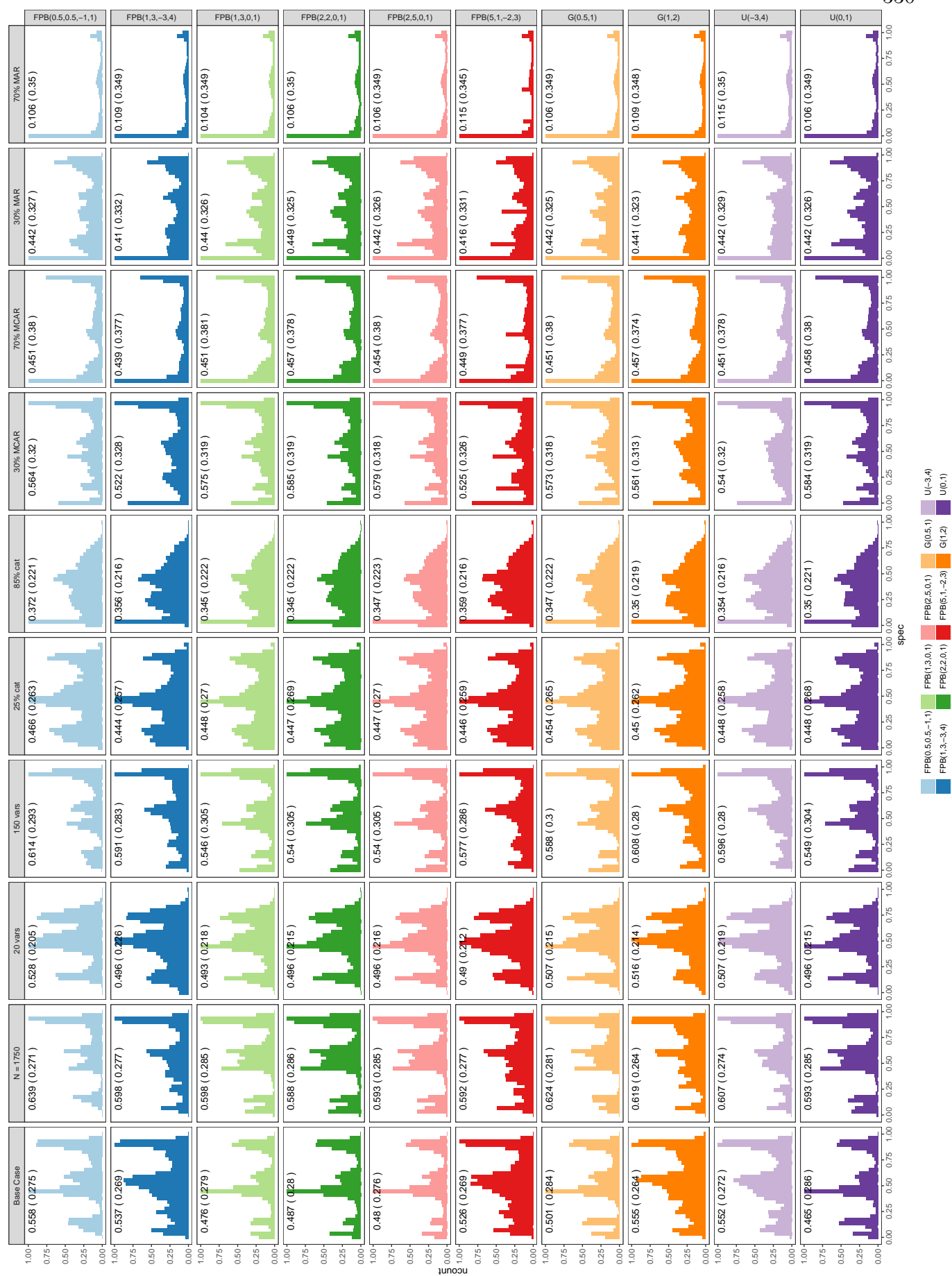


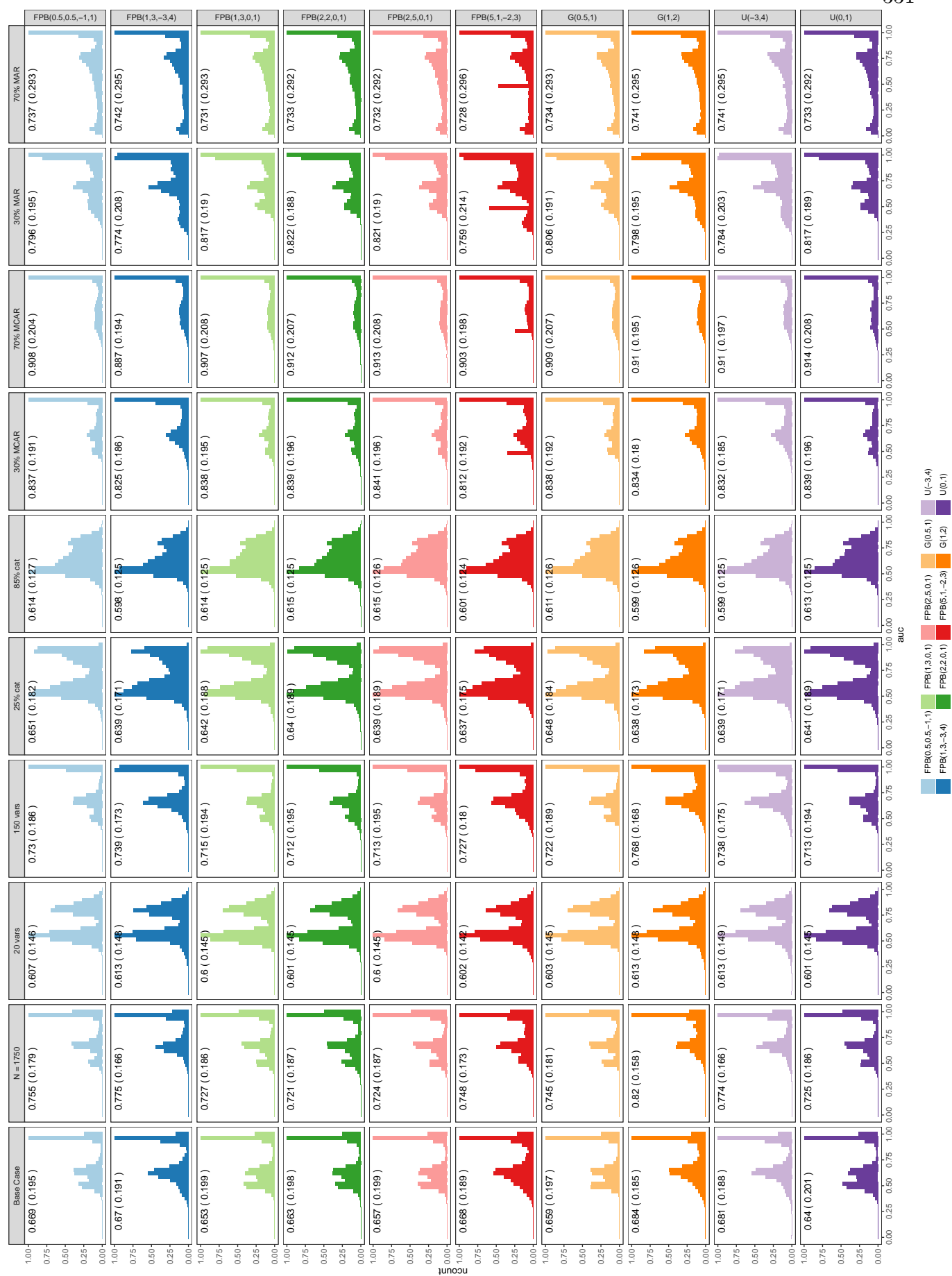


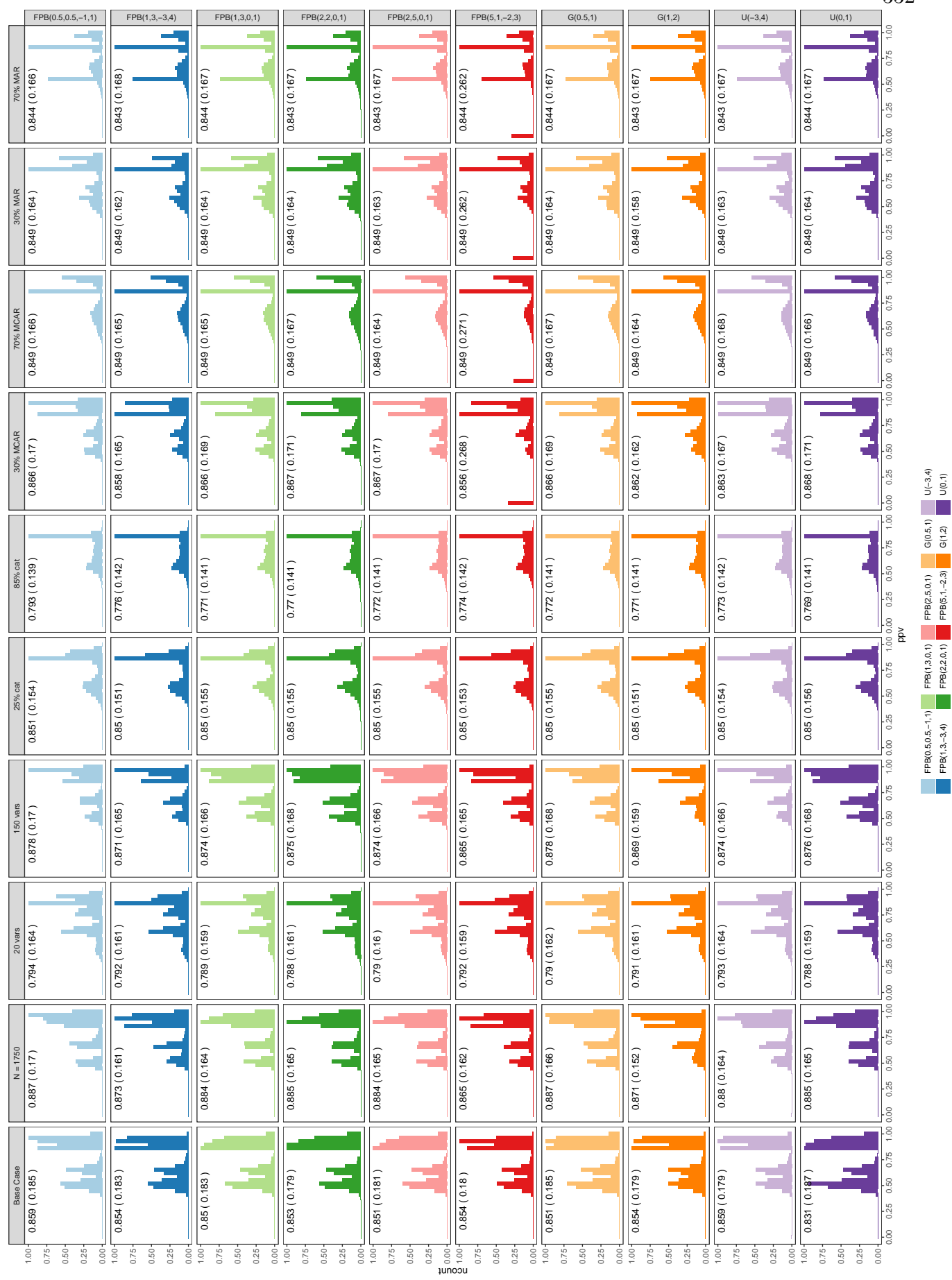
balance

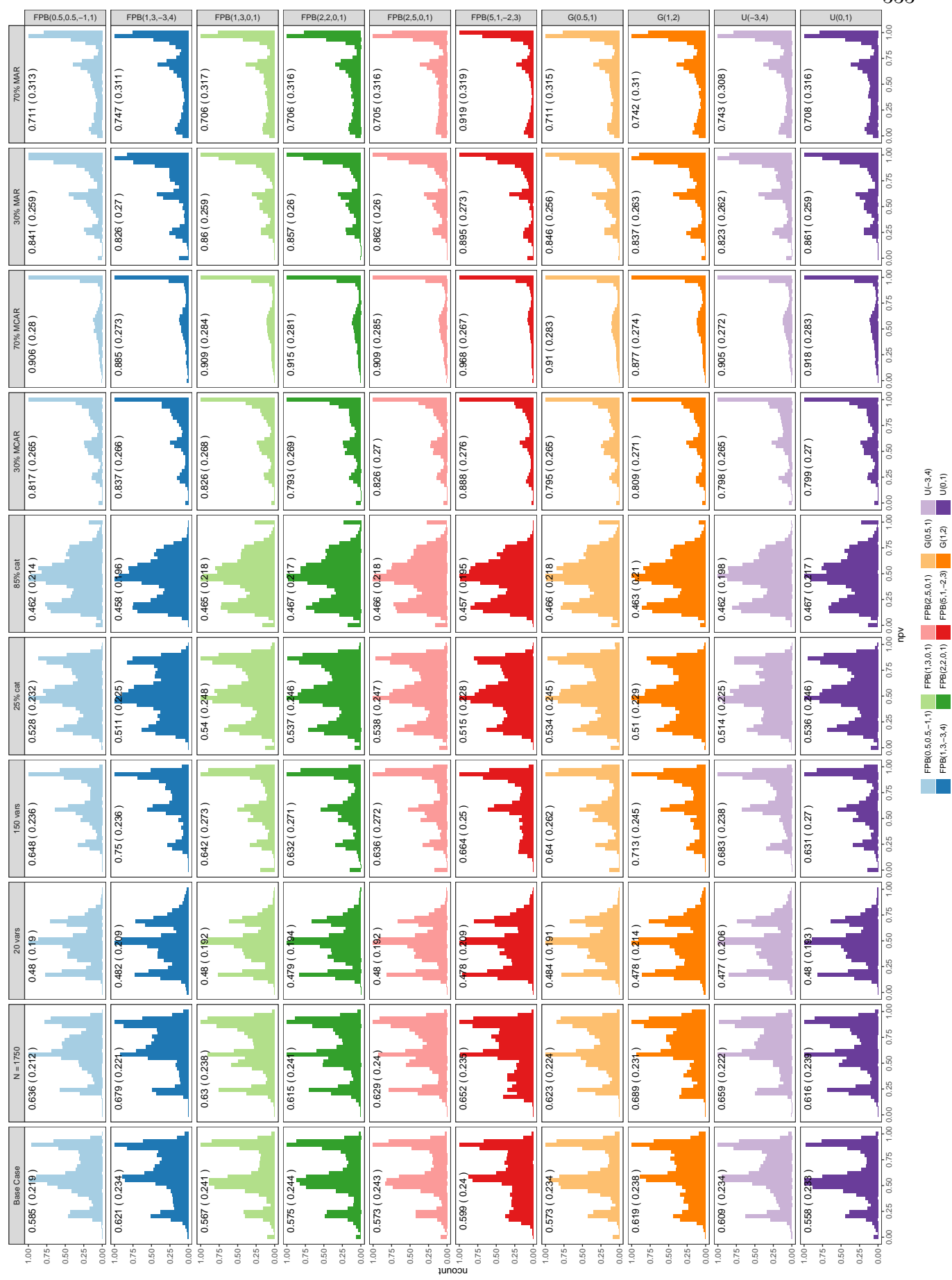
- FPB(0.5,0.5,-1,1)
- FPB(1.3,-3,4)
- FPB(1.3,0,1)
- FPB(2,2,0,1)
- FPB(2.5,0,1)
- FPB(5,1,-2,3)
- G(0.5,1)
- G(1,2)
- U(-3,4)
- U(0,1)













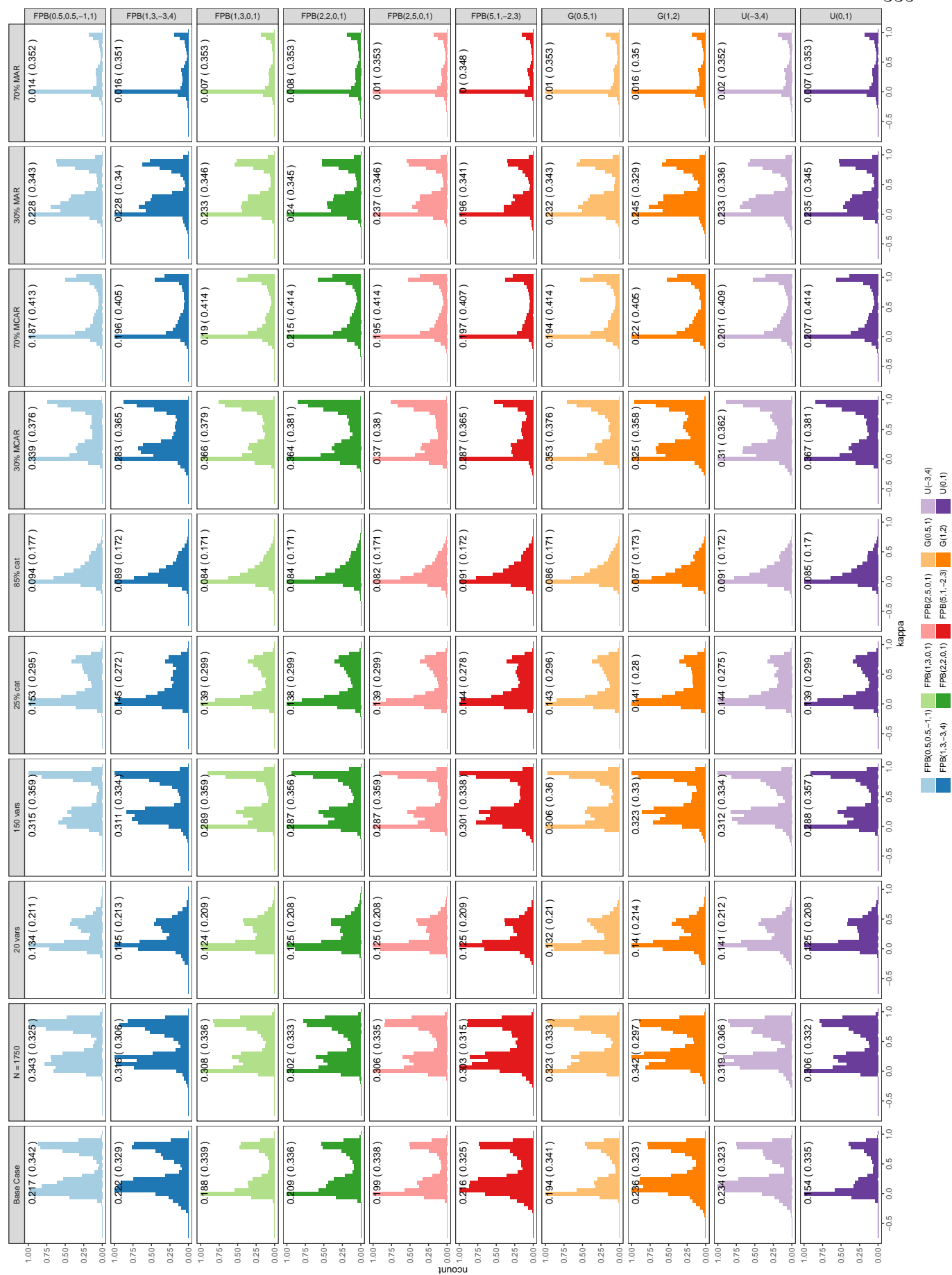
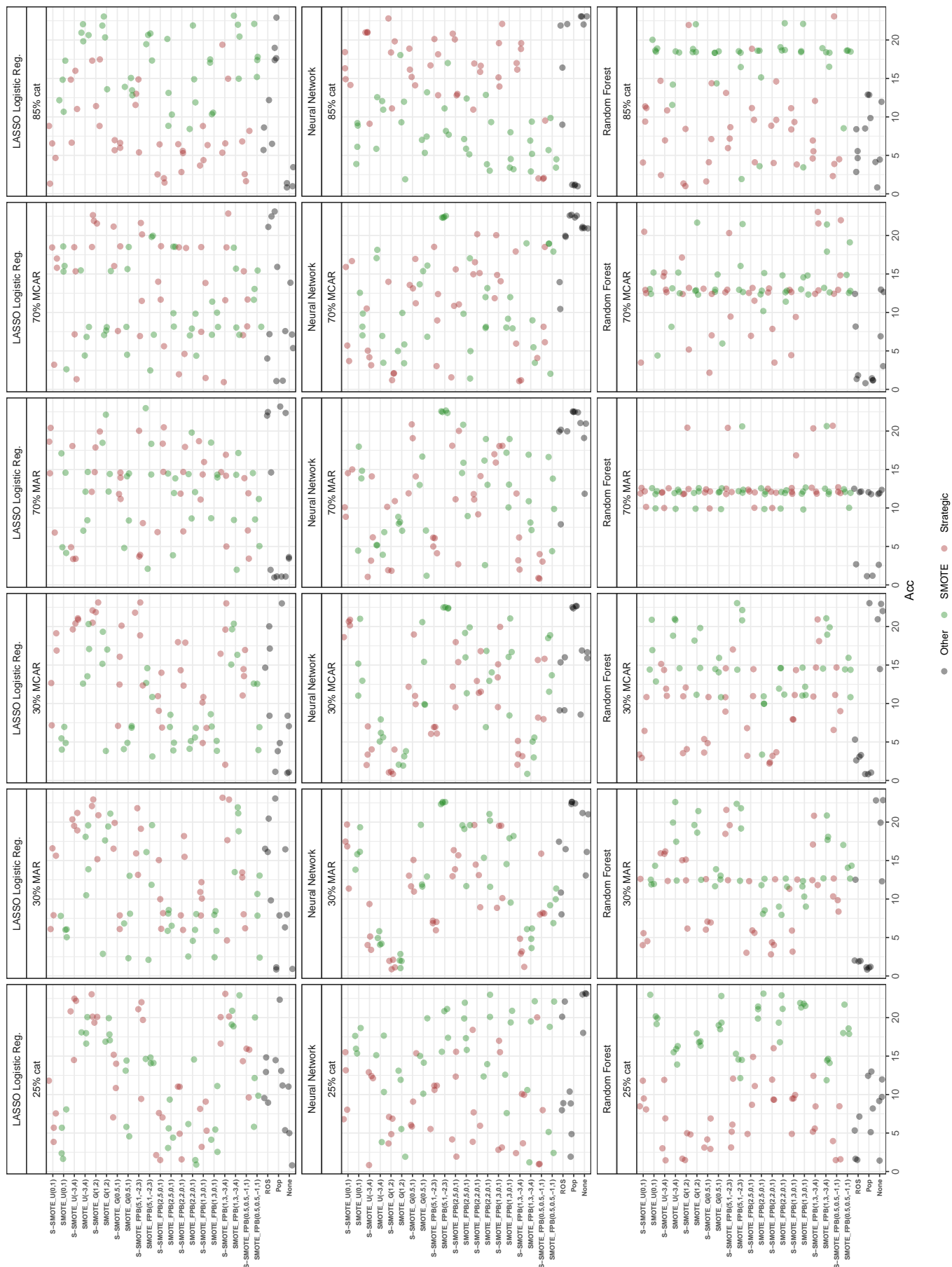
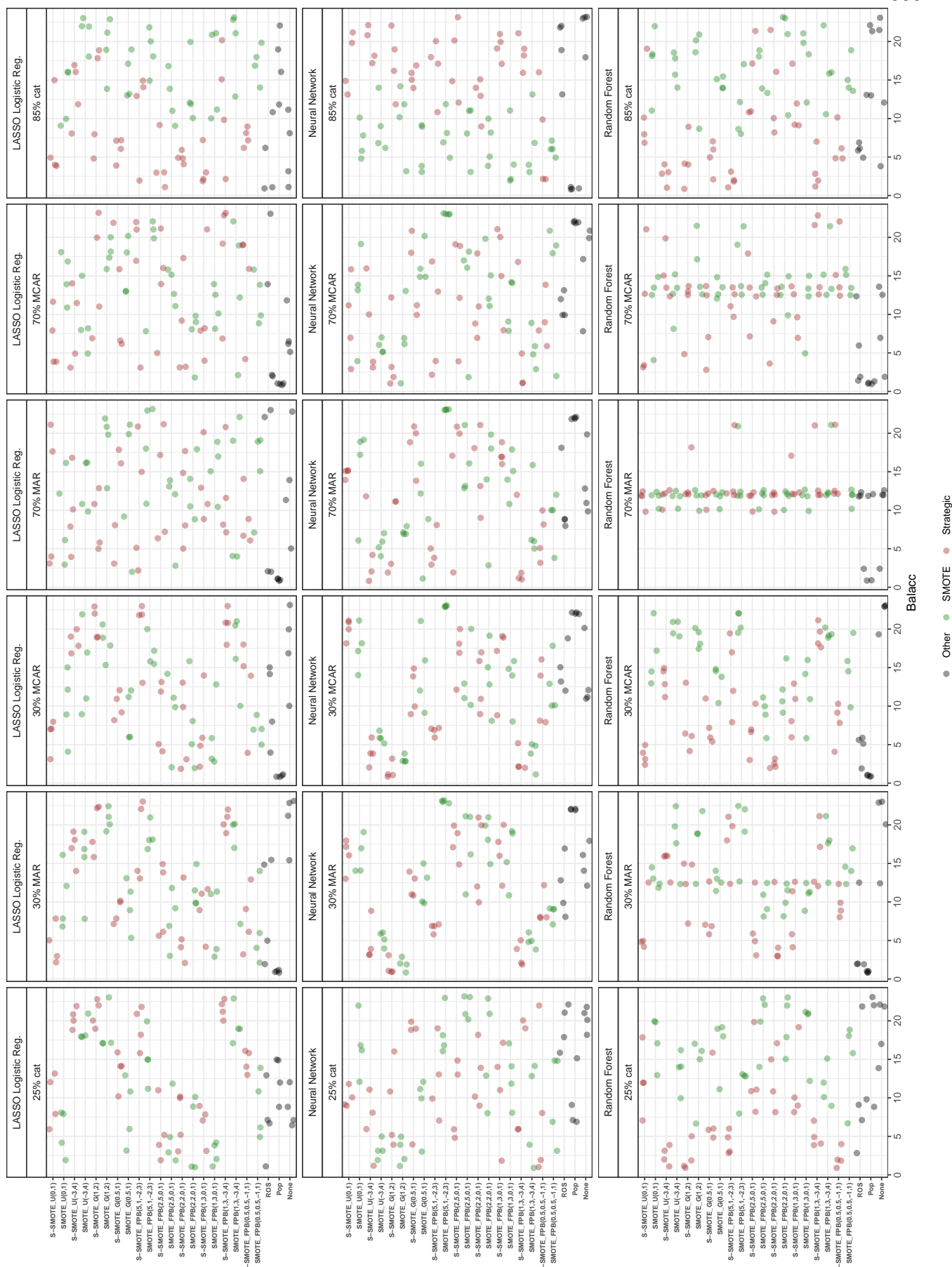
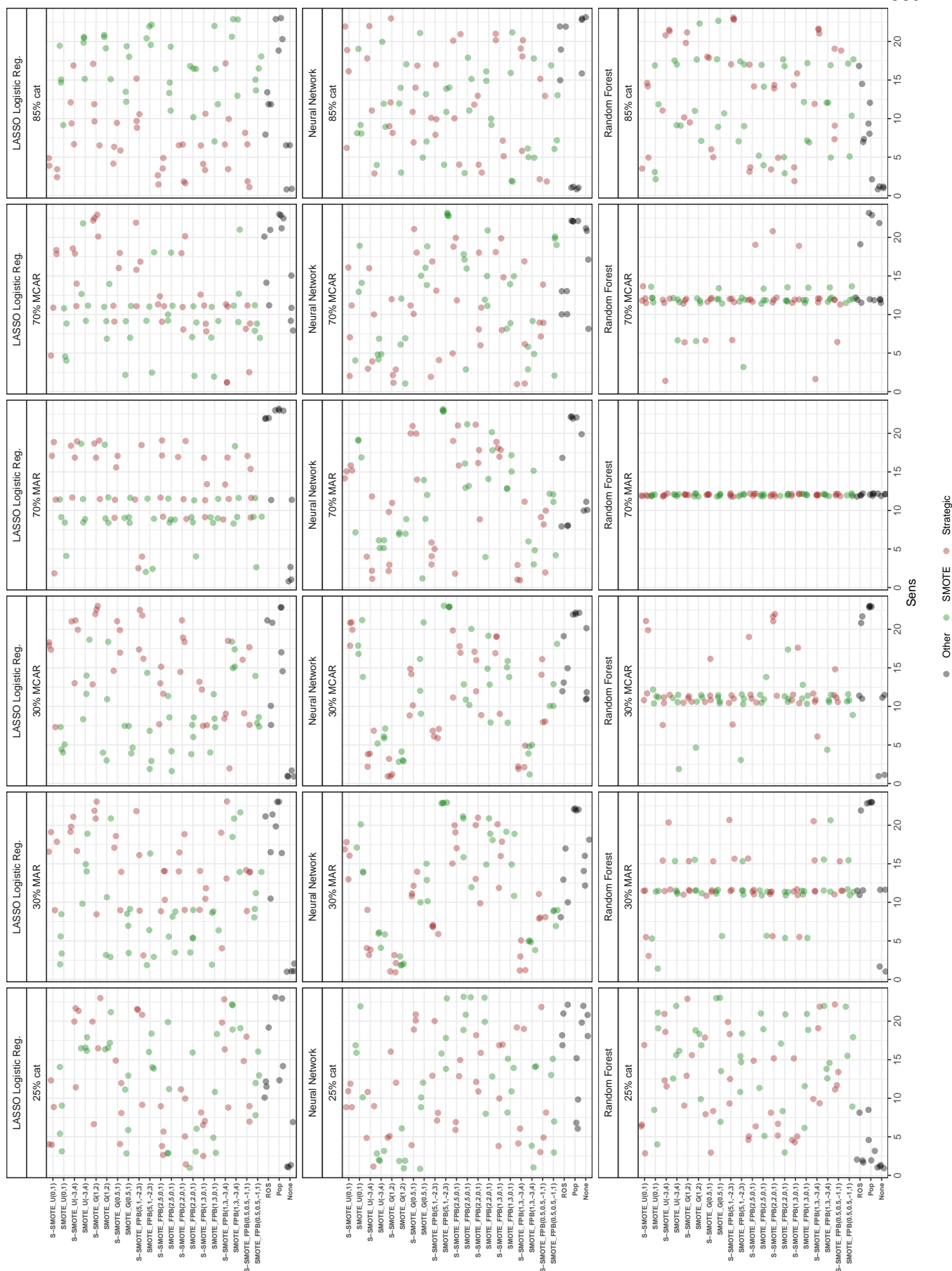


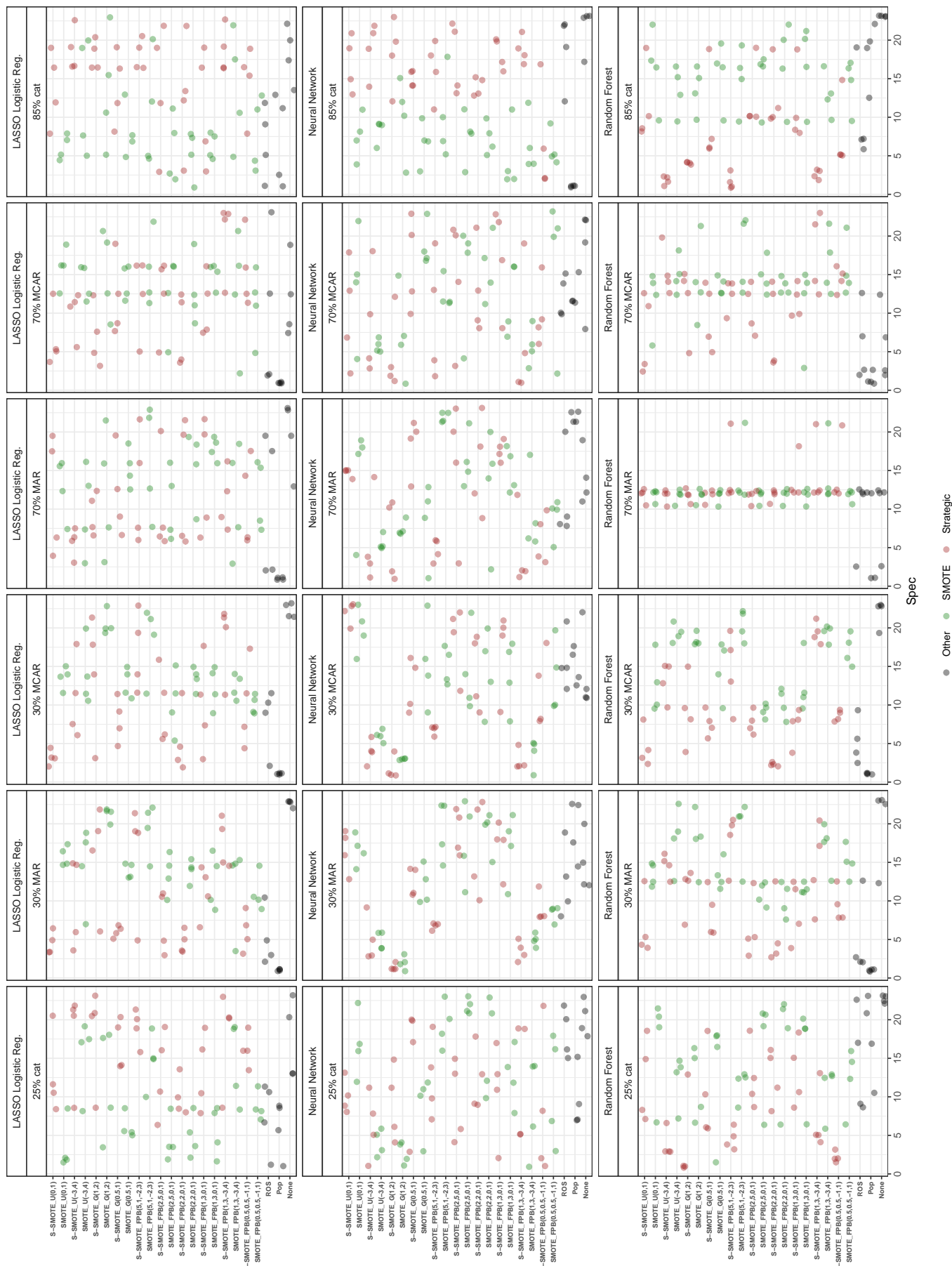
Figure B.6: Ranks of Median Performance Metrics By Oversampling Method, Model, and Data Scenario: The median performance metrics were calculated with respect to each data scenario, amount of imbalance and overlap, model applied, and oversampling method used. These were then ranked with respect to oversampling method (ranks of 1 to 23 possible). The rank is plotted on the x-axis for each oversampling method. Facets correspond to the model fit and data scenario.



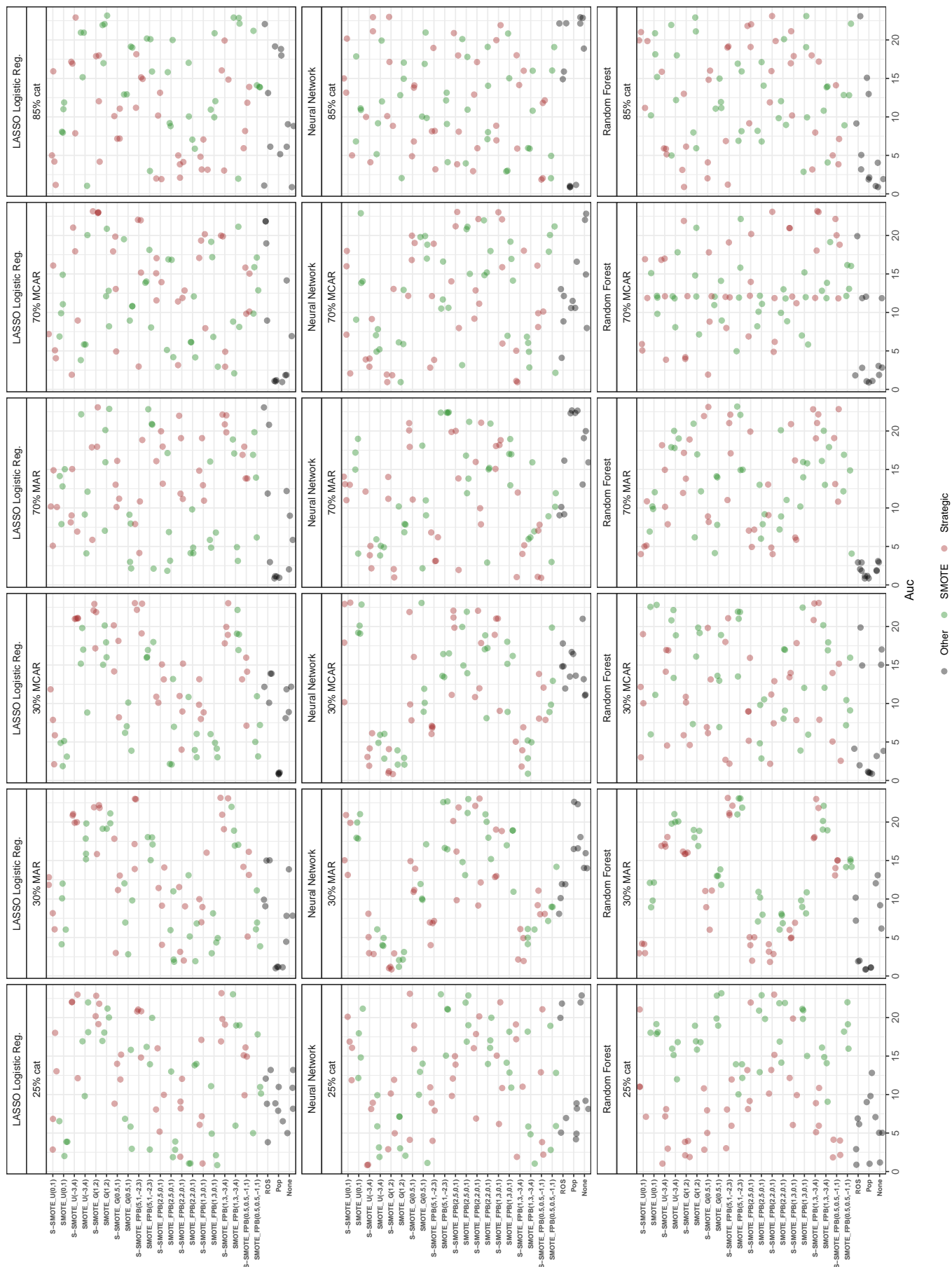


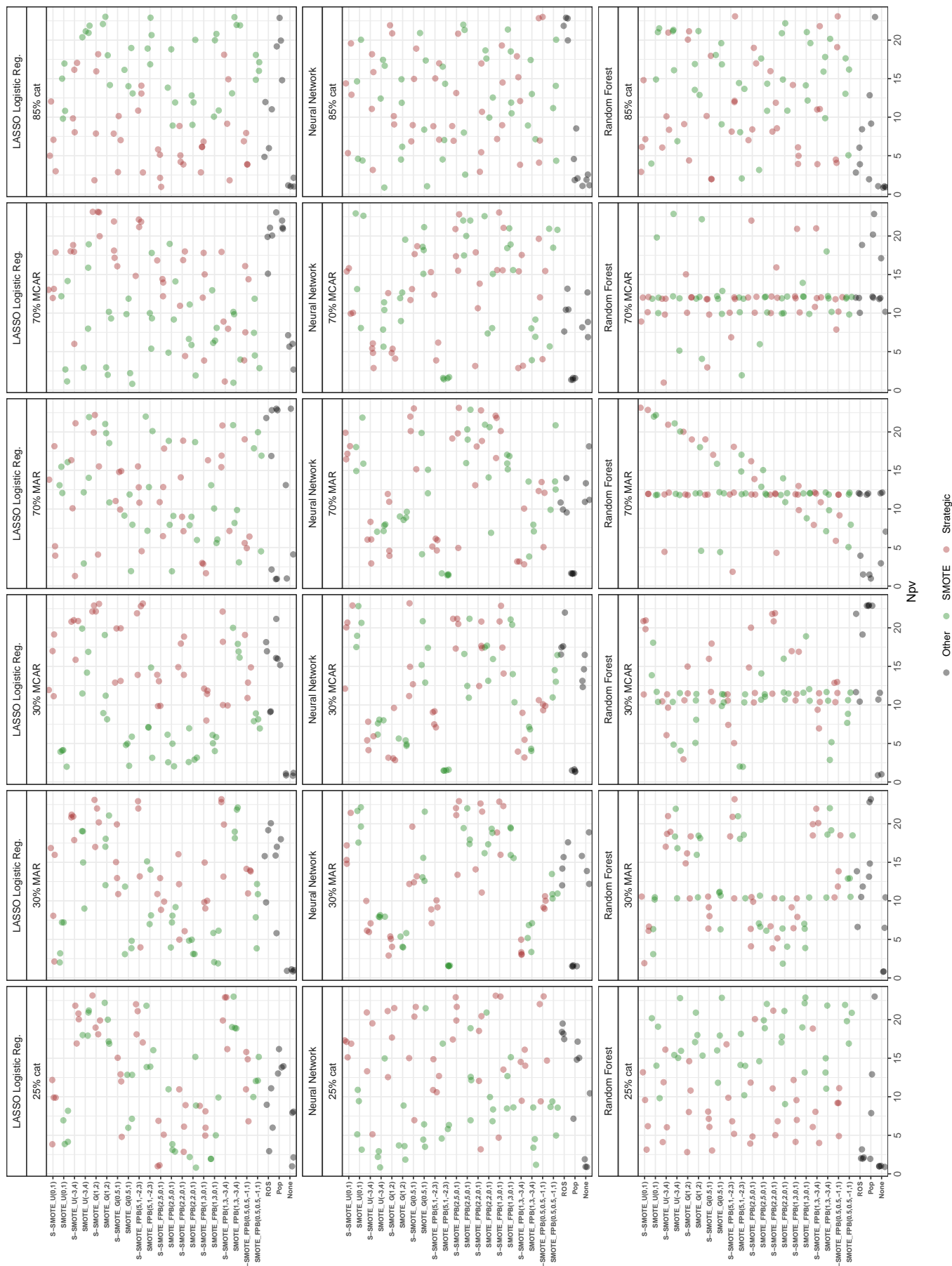


Other SMOTE Strategic

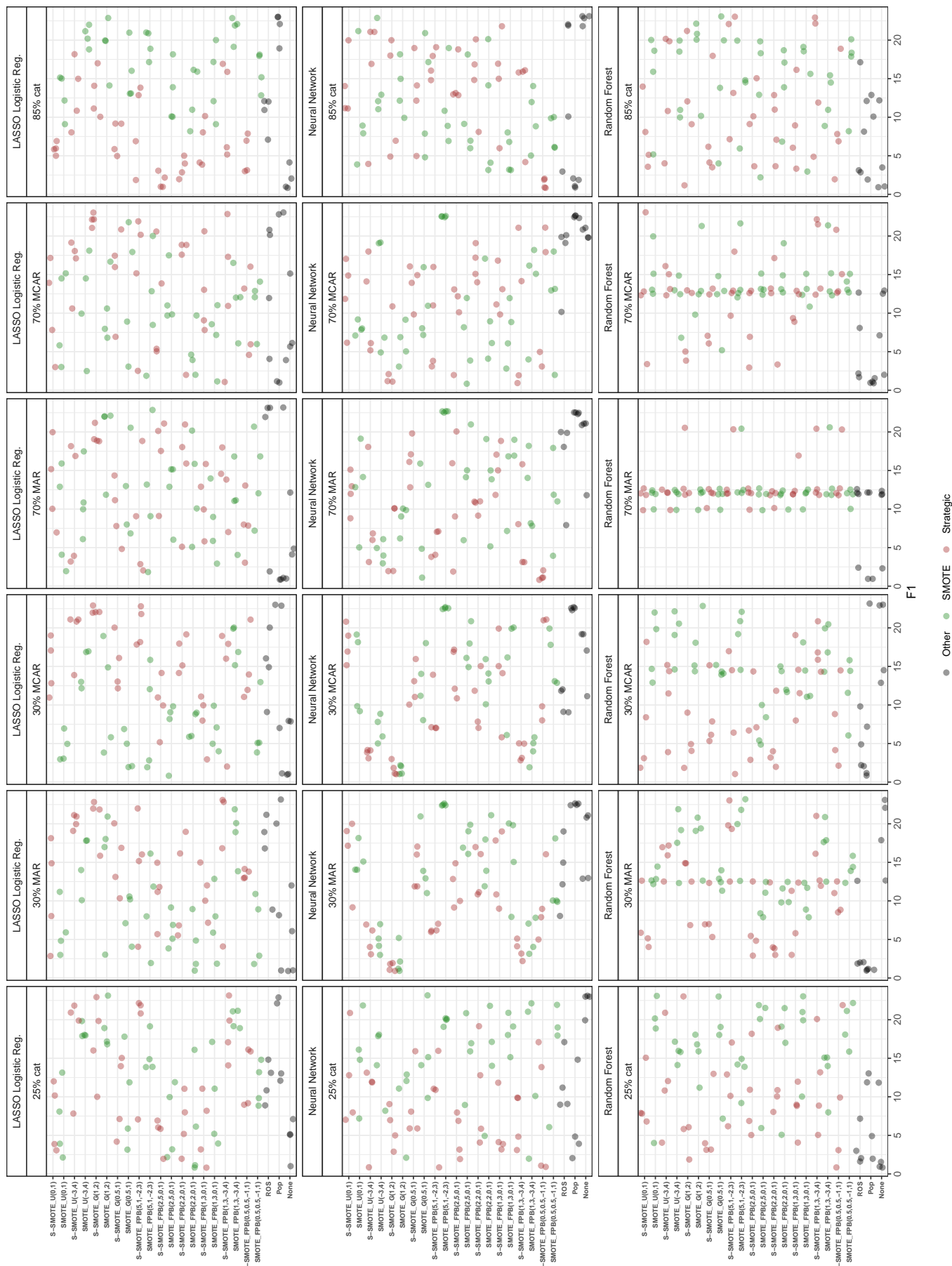


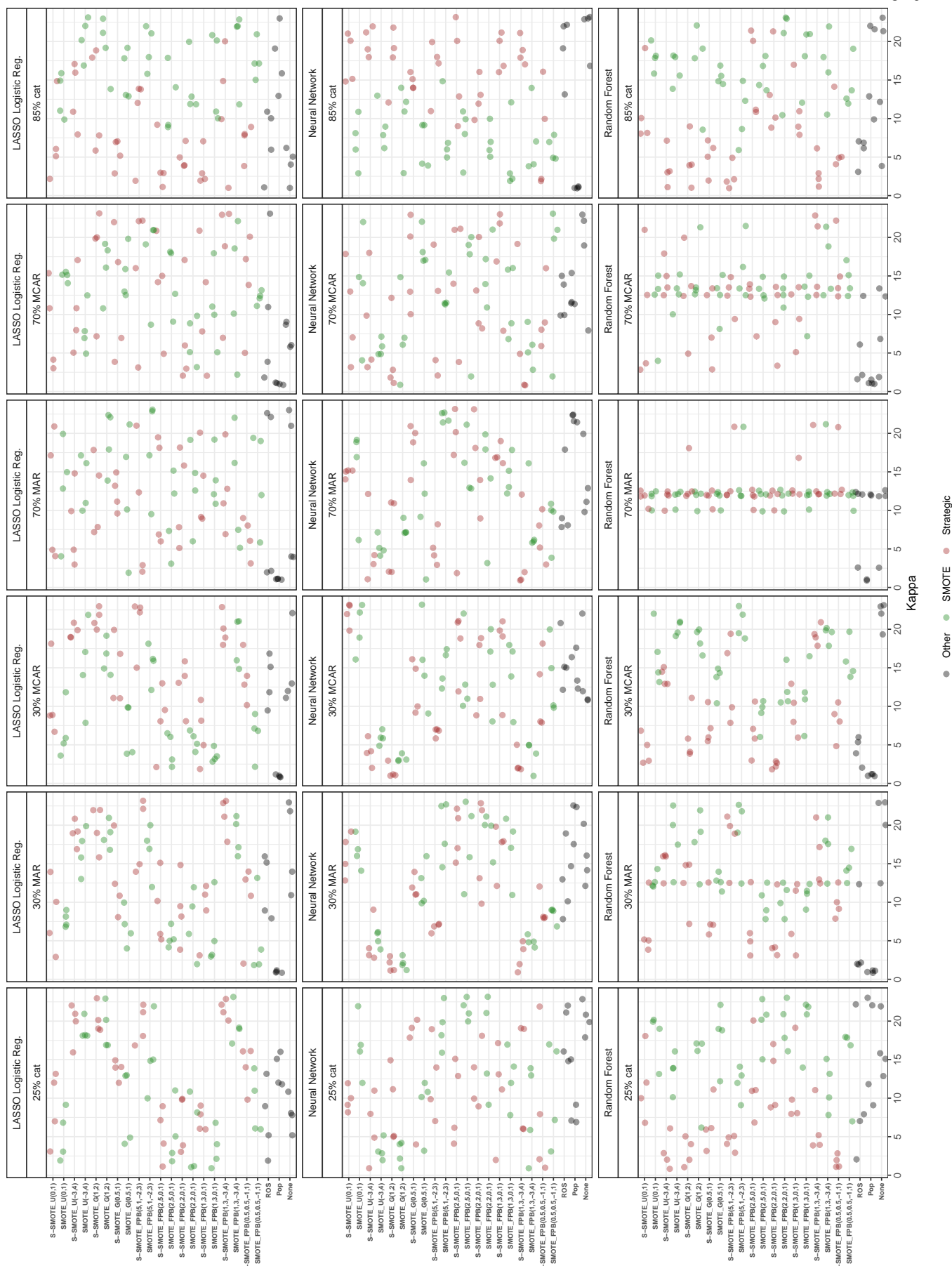
Other SMOTE Strategic

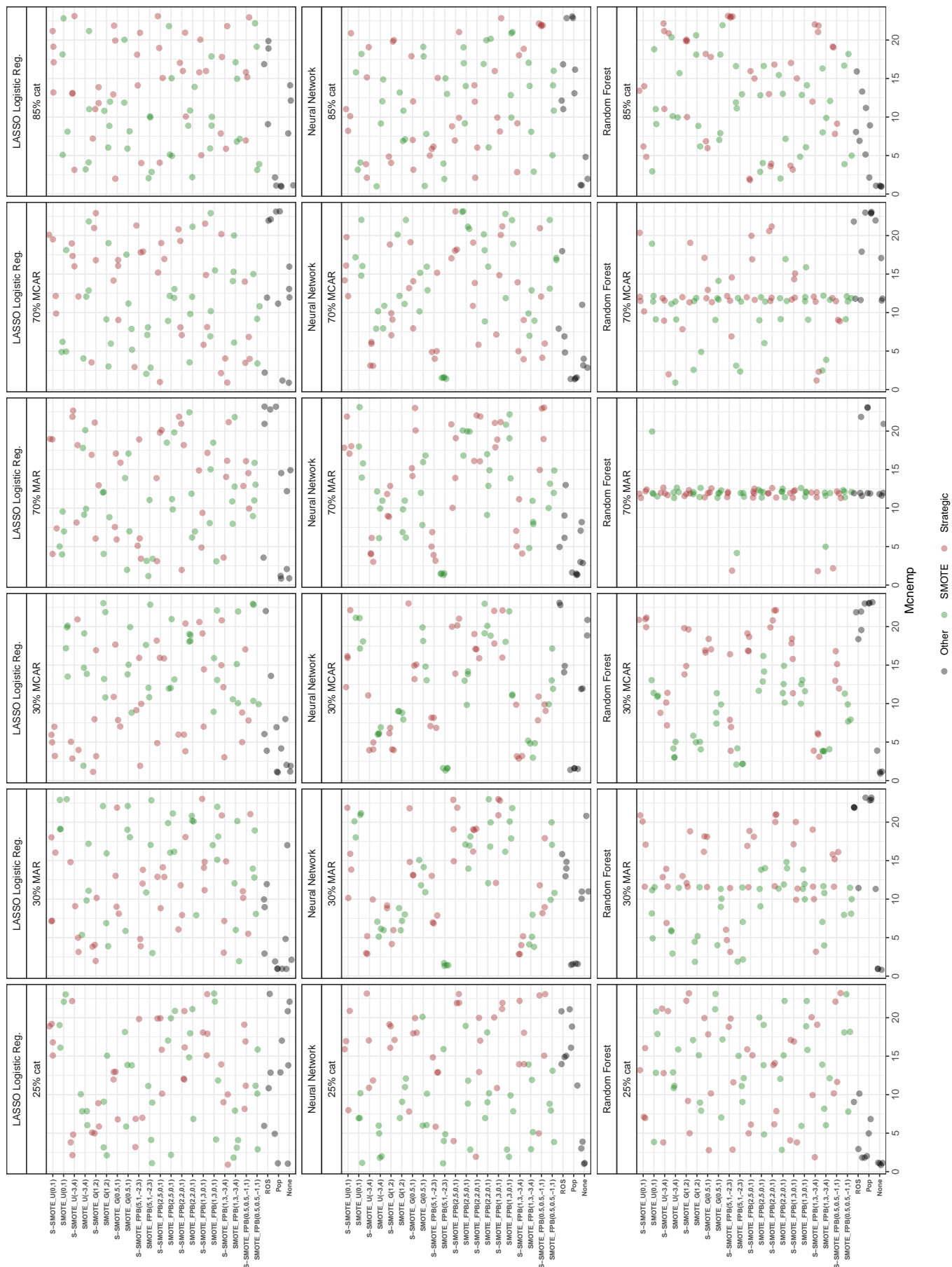




● Other ● SMOTE ● Strategic

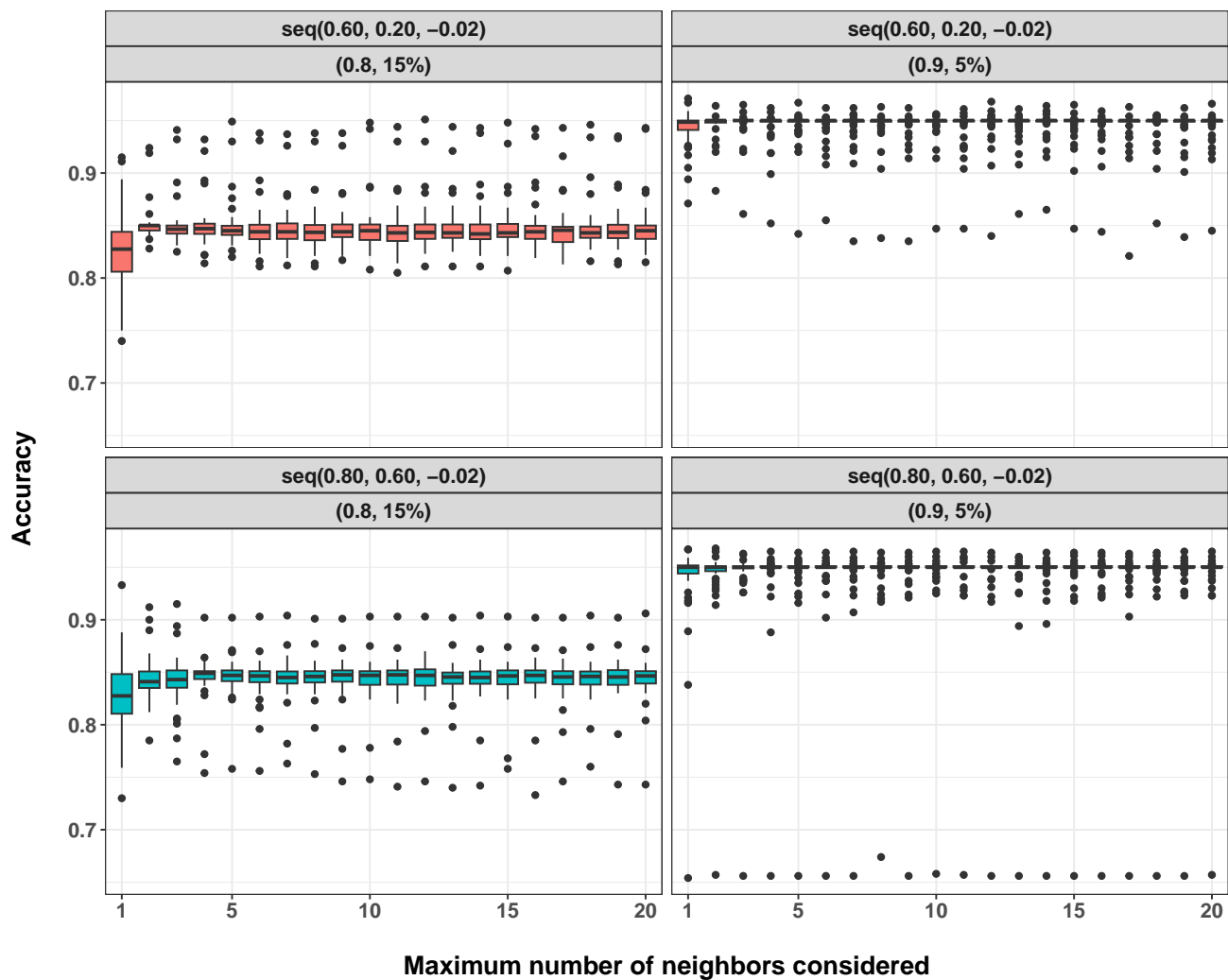


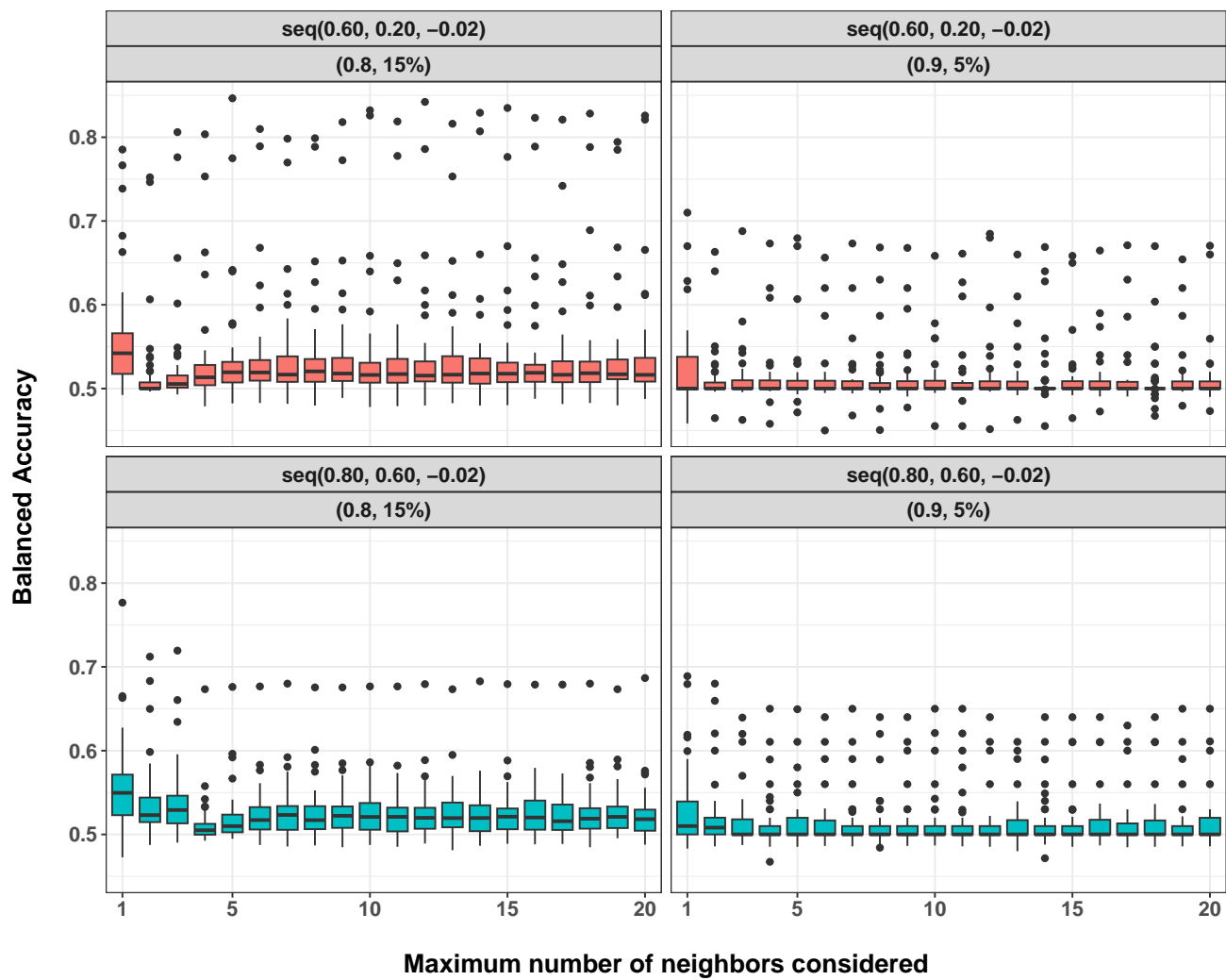


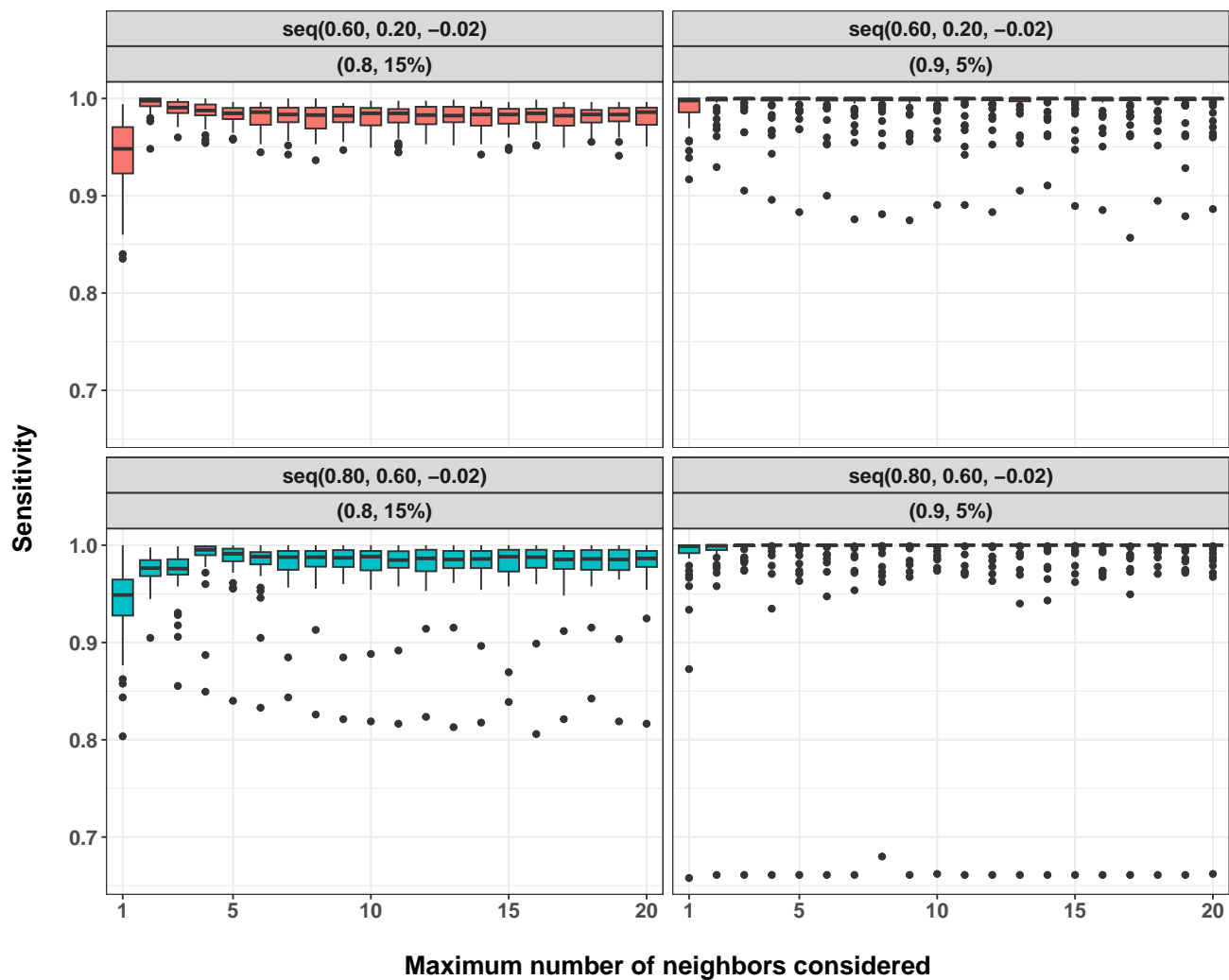


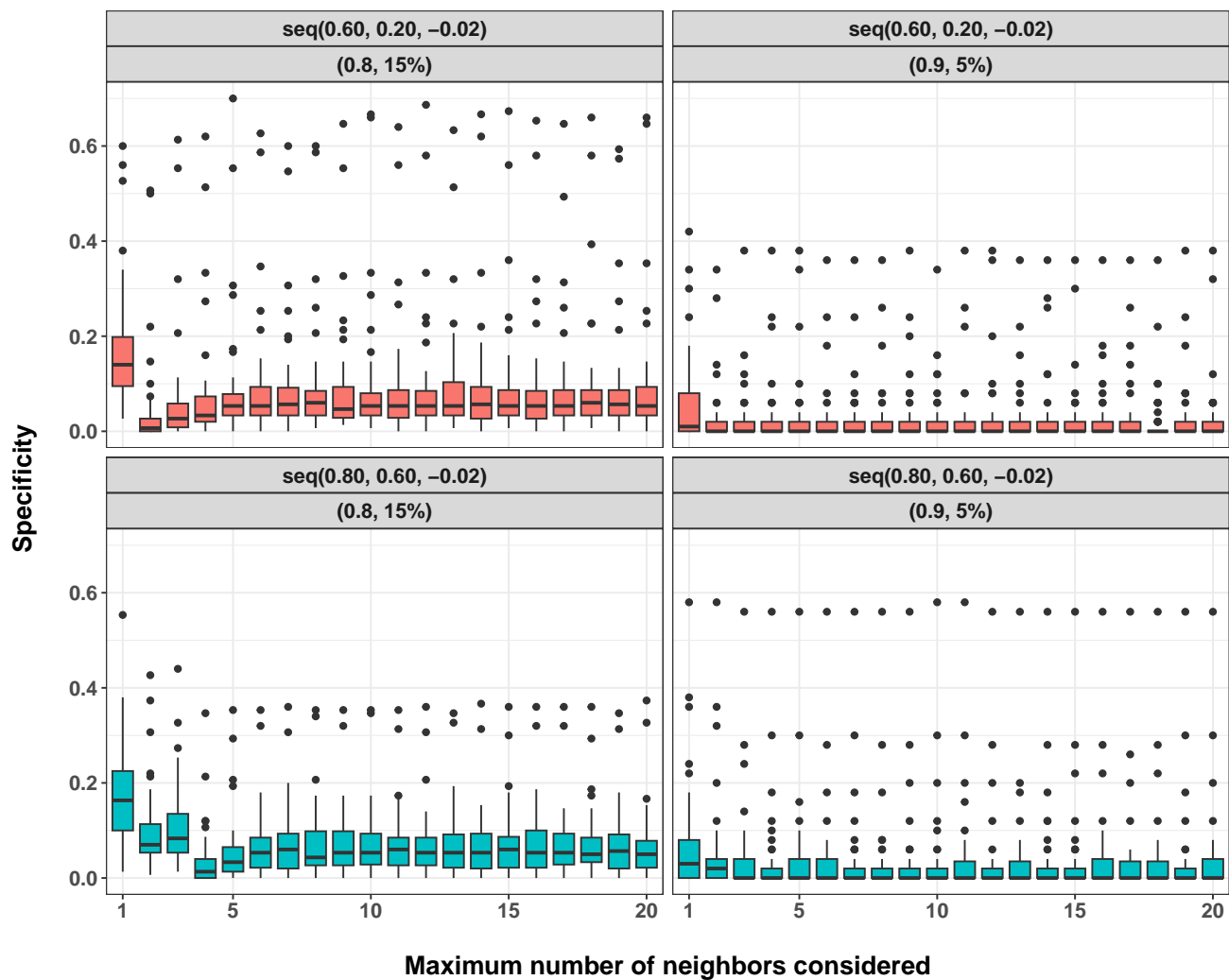
Appendix C: Supplementary Visualizations of Simulation Results Studying Hyperparameters of S-SMOTE

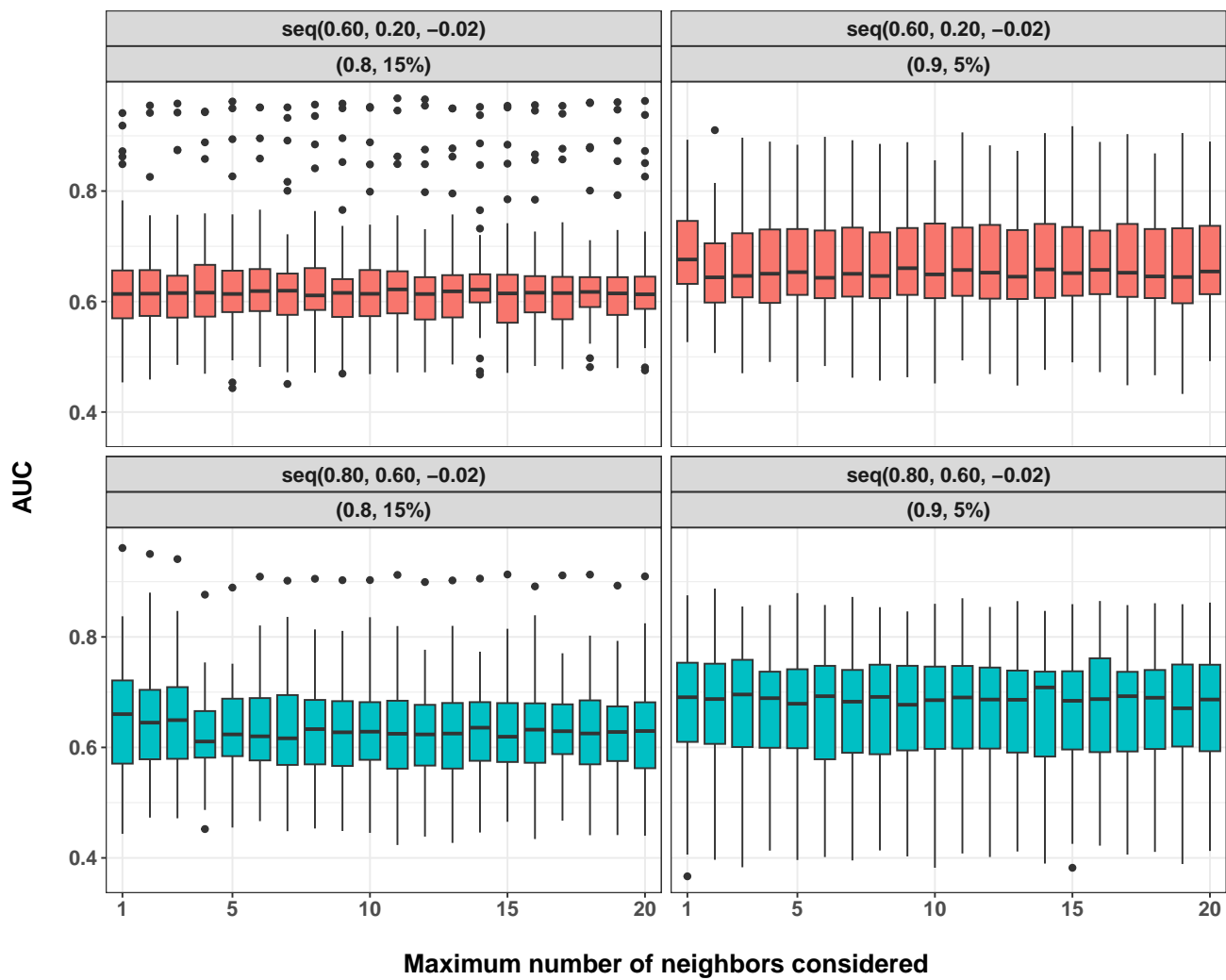
Figure C.1: Distribution of Performance Metrics For Various Parameters of S-SMOTE: The first set of plots (4 facets per page) provides the distribution of performance metrics (y-axis) as the maximum number of nearest neighbors considered ranges from one to 20. We denoted this parameter as k_{max} when describing S-SMOTE. This is faceted by the sequence of thresholds tried in S-SMOTE and the amount of overlap and percent minority examples in the simulated dataset. The second set of plots (3 facets per page) provide the distribution of performance metrics (y-axis) as the sampling weights used to select minority class examples for oversampling change. These weights determine how often we oversample minority points after they have been separated using the median dominance threshold met and number of neighbors deemed fit for oversampling. The third value corresponds to the quadrant that least dominated and crowded minority examples fall into followed by the second value, fourth value, and the first value. 50 repetitions were performed for each simulation with a different dataset each time. Datasets were simulated in the same manner that was used for the larger simulations. When the overlap amount was 0.9 and 5% of points belonged to the minority class many values for the NPV were missing, leading to more variable results (see Figure 4.16).

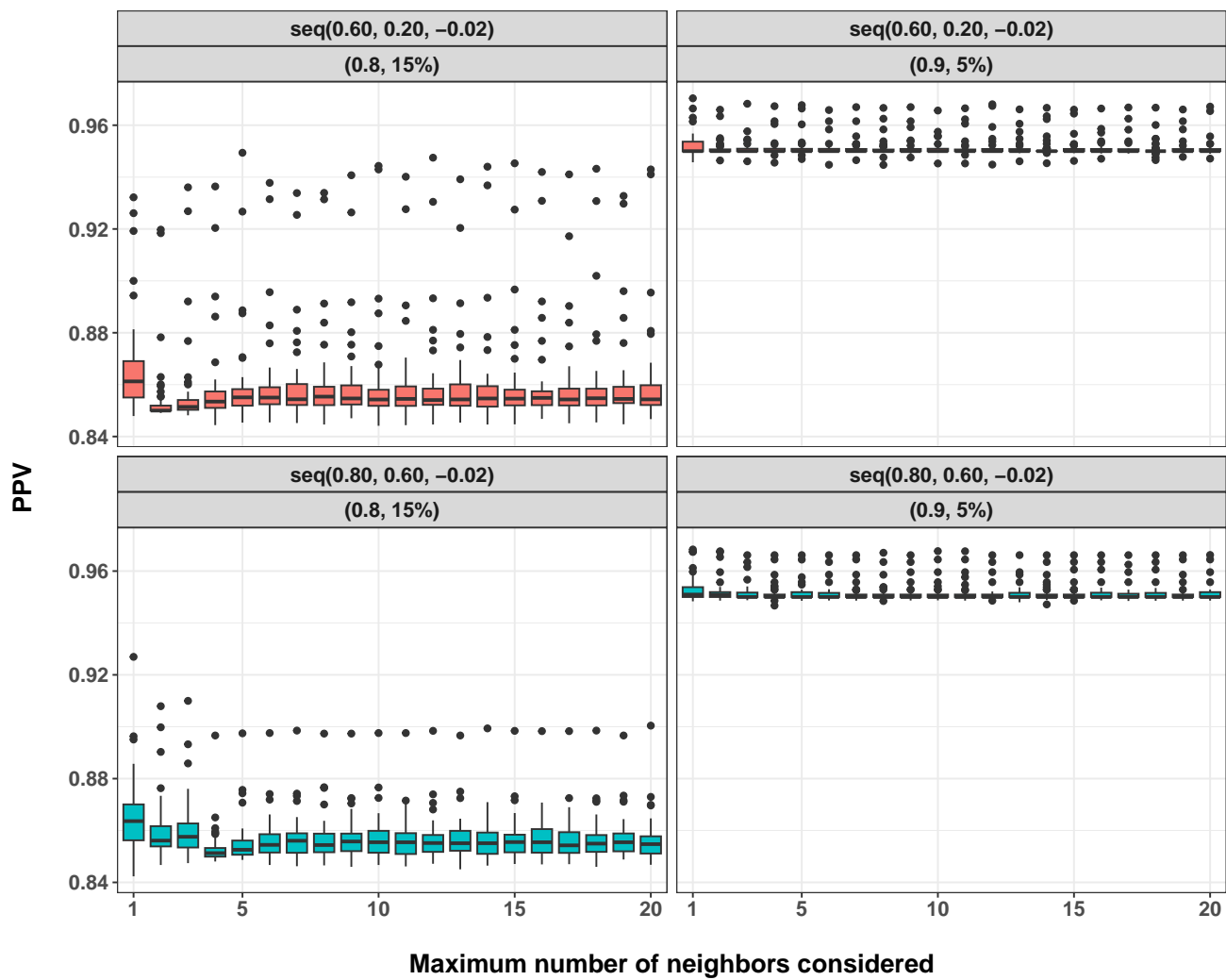


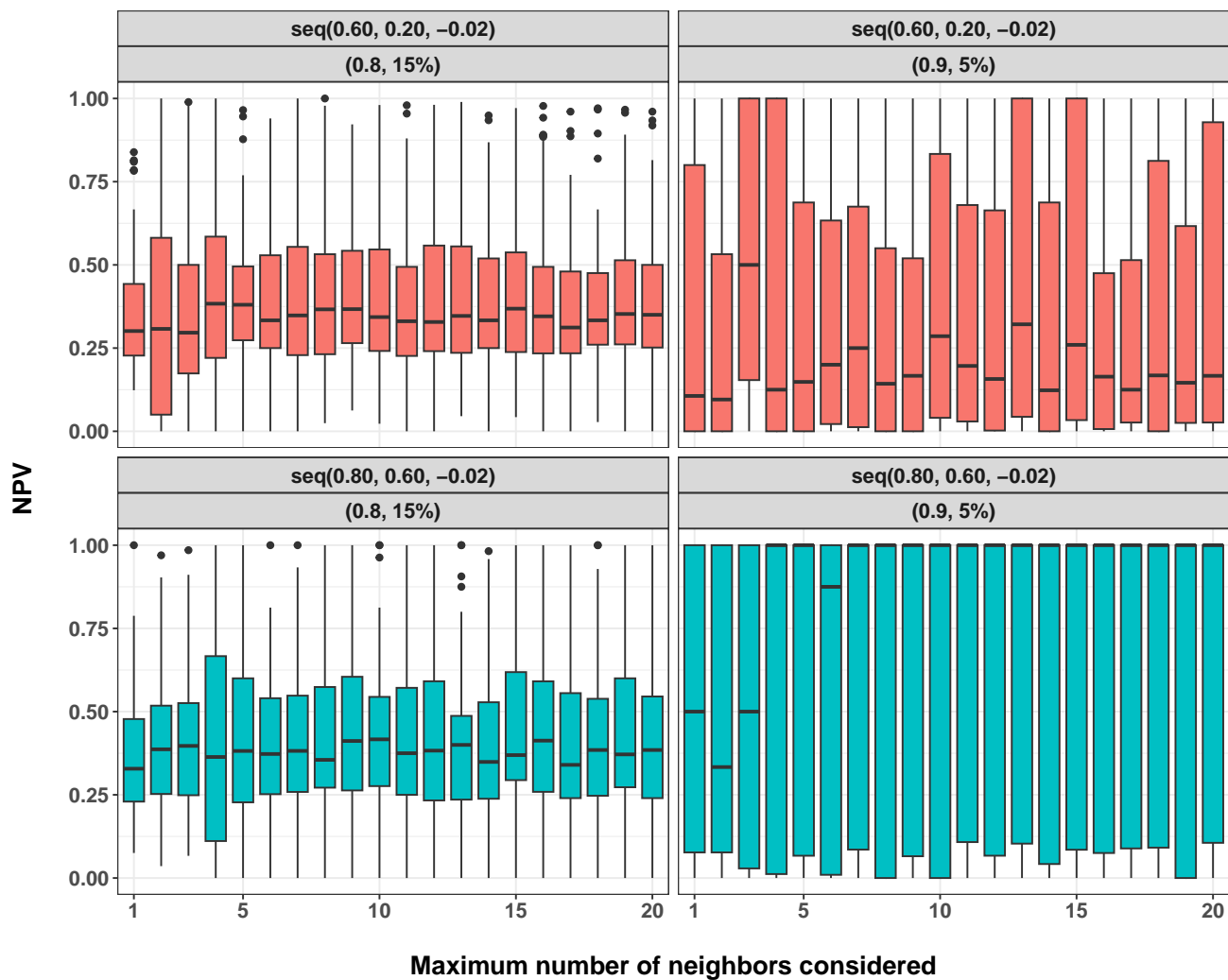


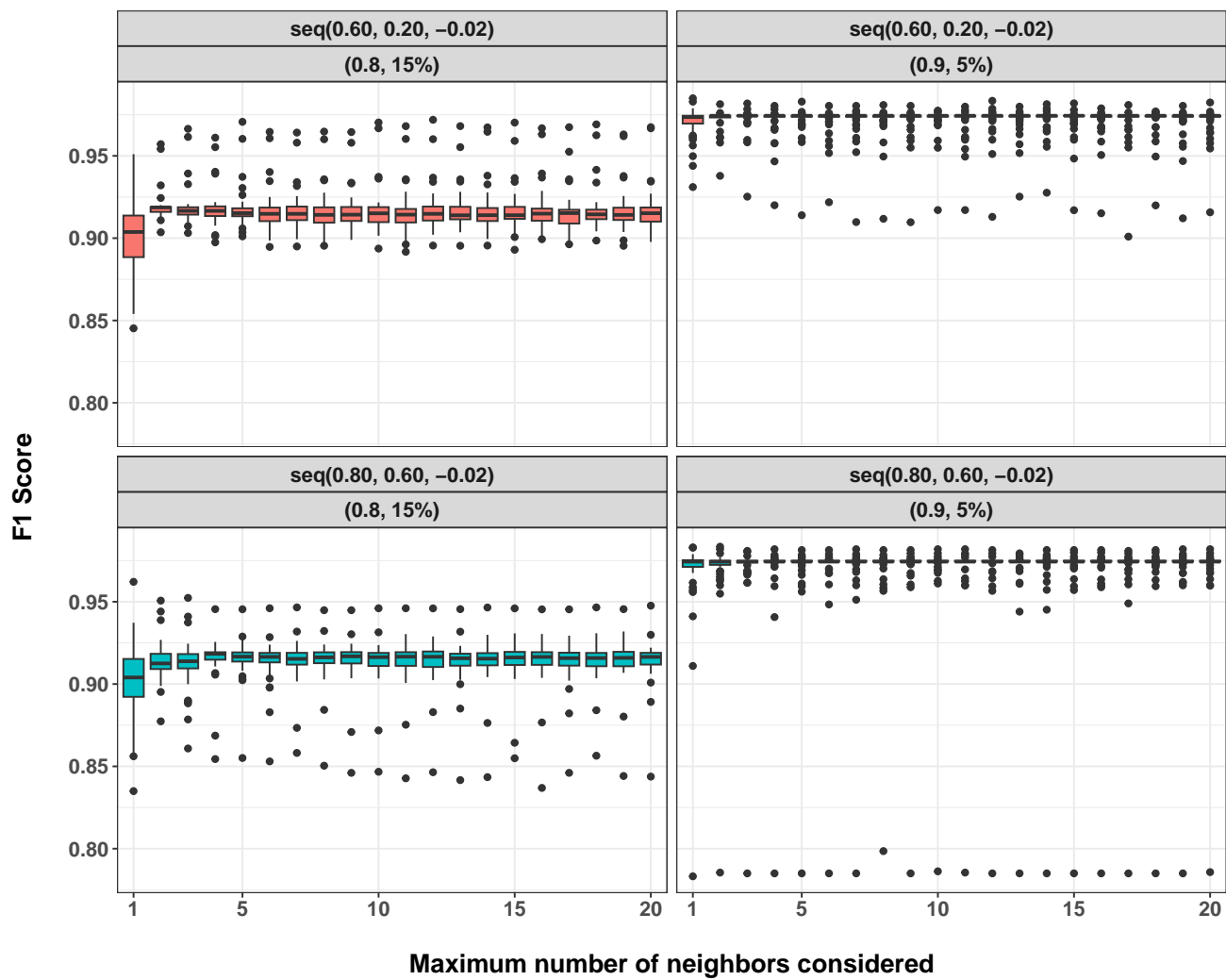


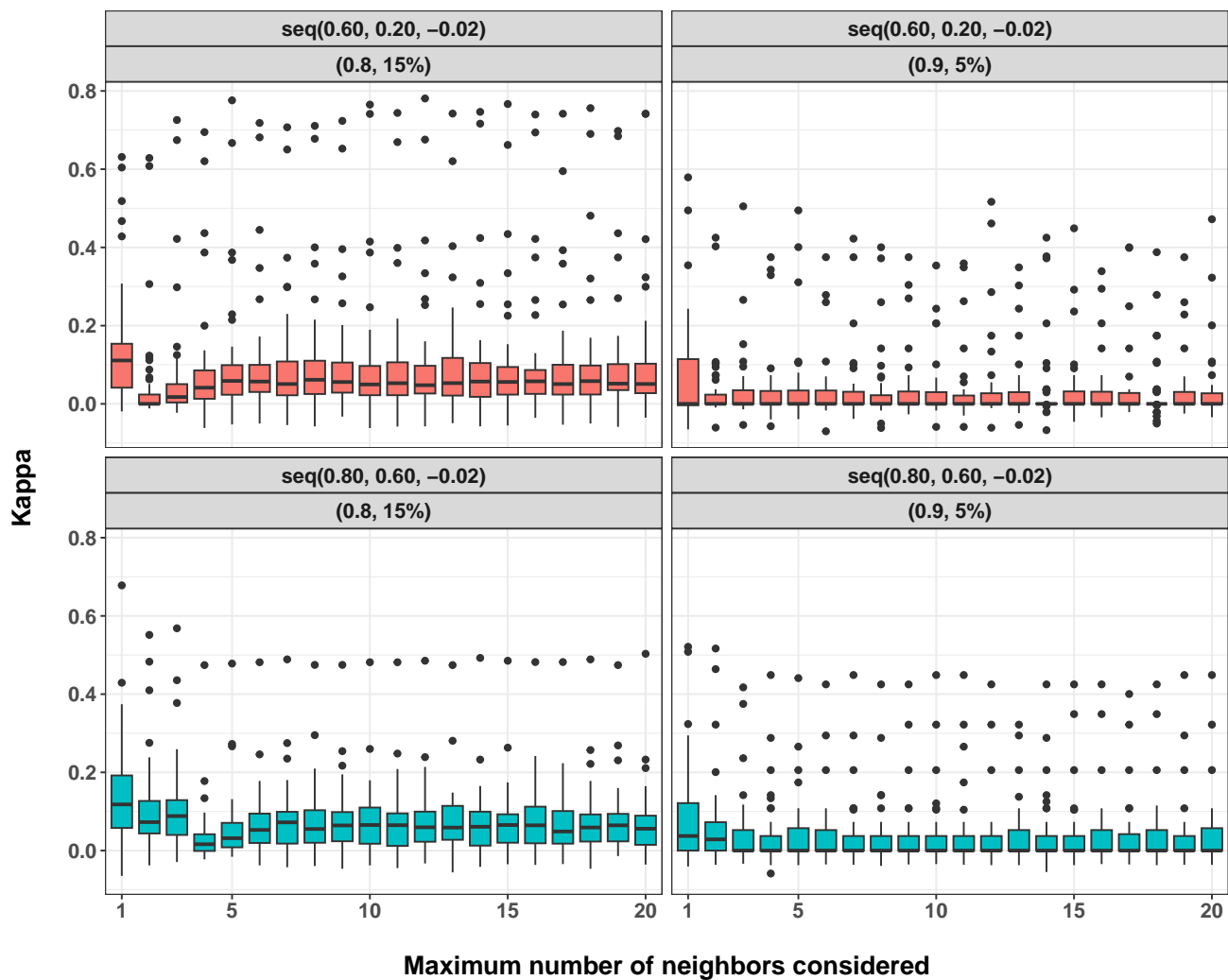


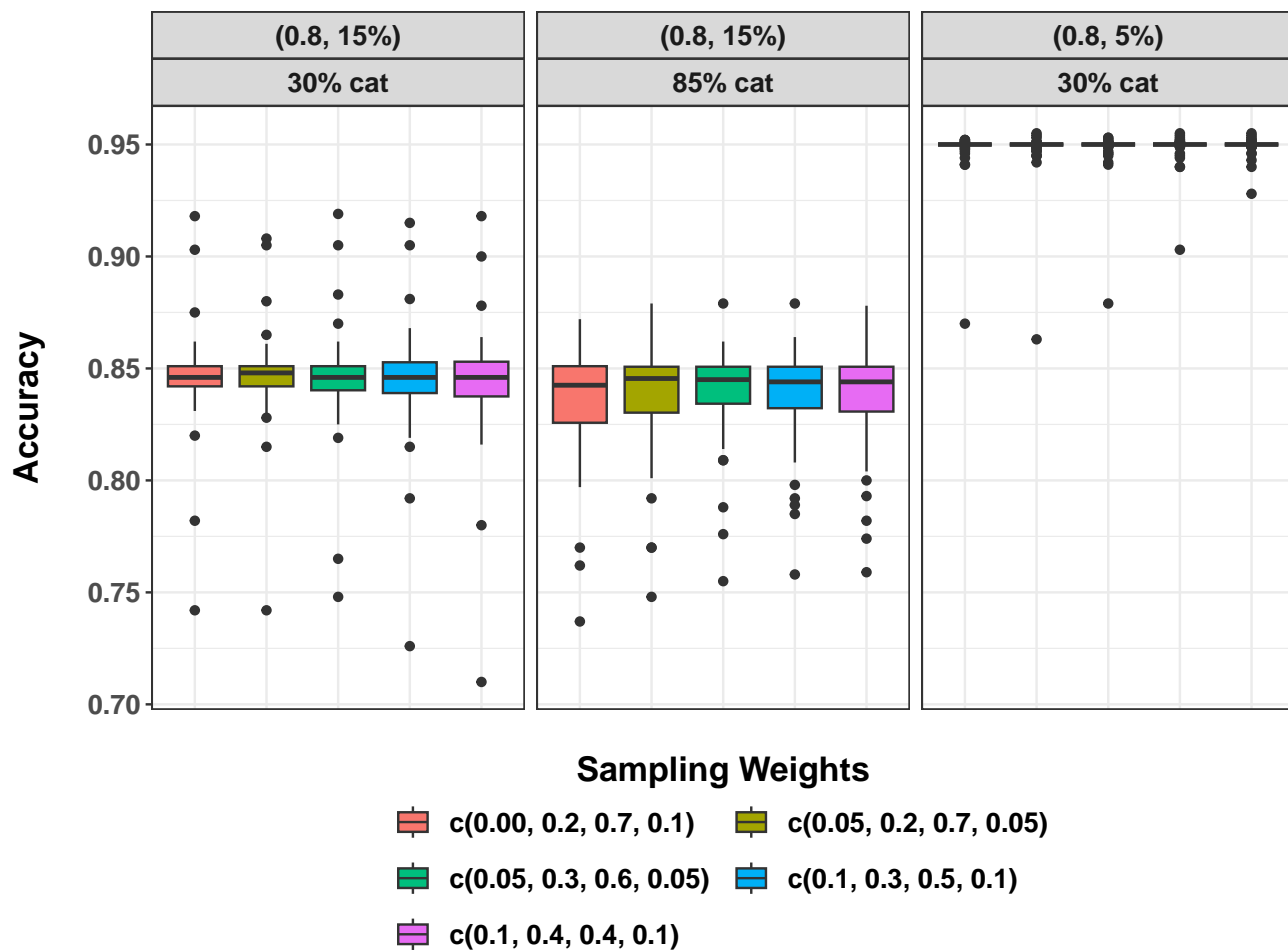


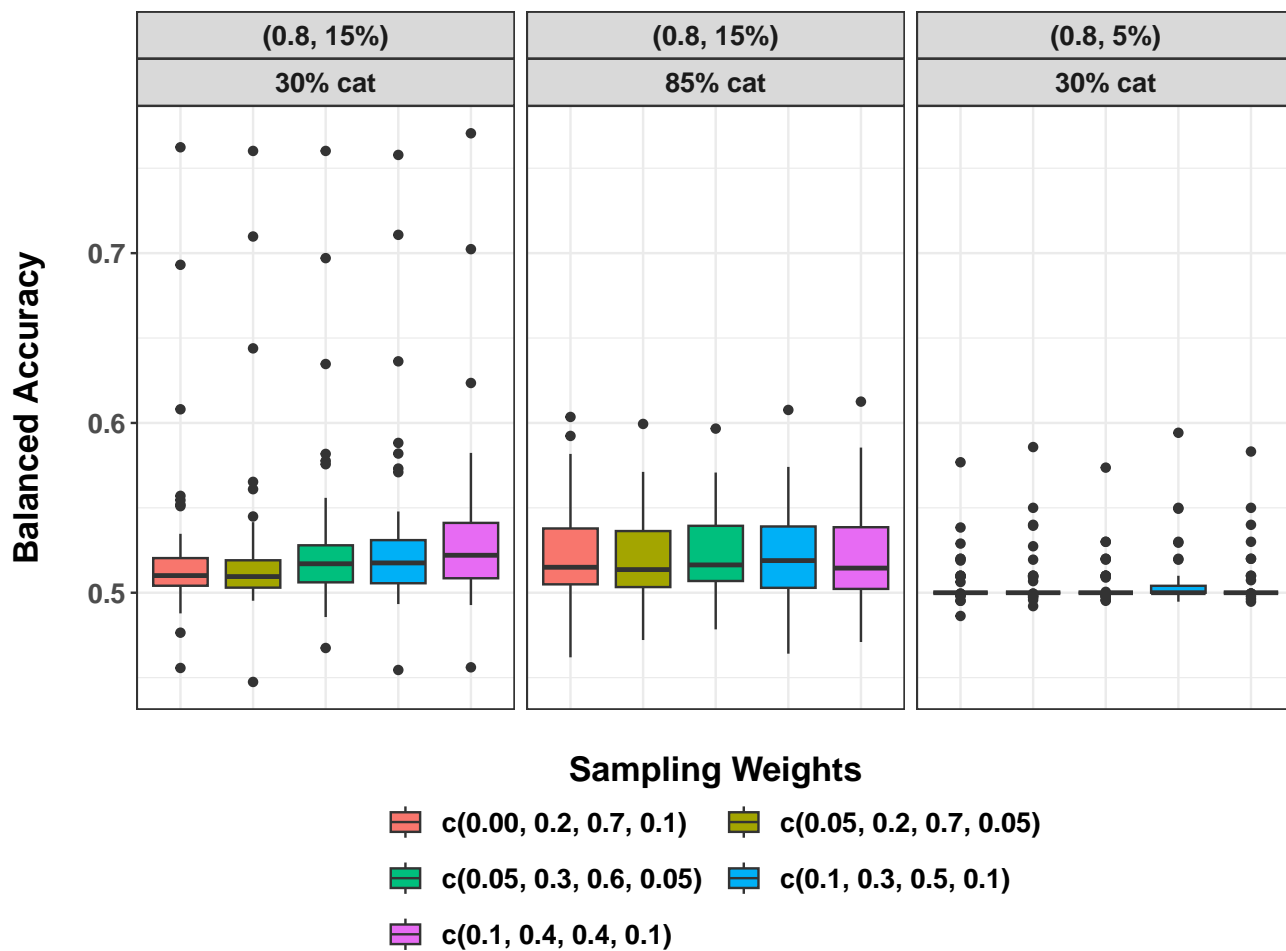


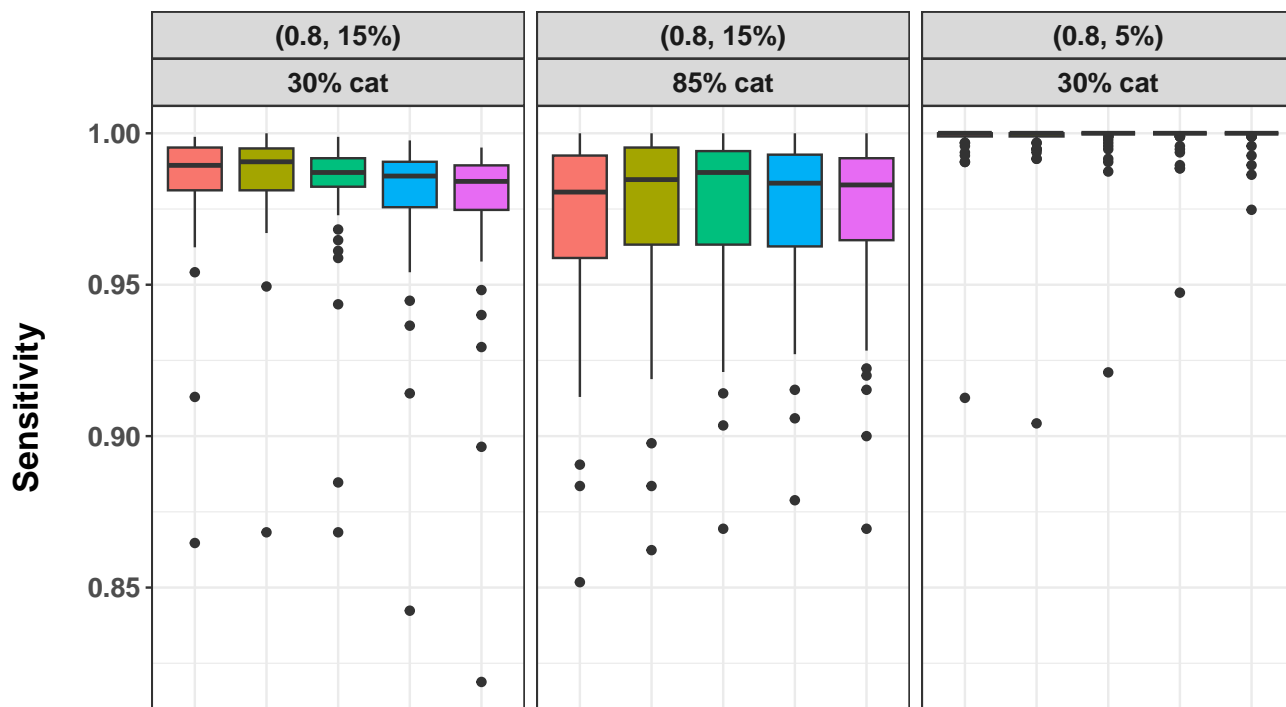












Sampling Weights

- $c(0.00, 0.2, 0.7, 0.1)$
- $c(0.05, 0.2, 0.7, 0.05)$
- $c(0.05, 0.3, 0.6, 0.05)$
- $c(0.1, 0.3, 0.5, 0.1)$
- $c(0.1, 0.4, 0.4, 0.1)$

