Striking a Balance:
Performance vs. Fairness Considerations when Rebalancing Data

Anthony Rentsch and Nathan Einstein

Abstract

Extensive research has been done into developing and assessing an array of resampling methods that correct for class imbalance. But investigation into how these techniques impact algorithmic fairness is very sparse. We therefore examine how five common upsampling and downsampling techniques influence the predictive fairness, measured in terms of predictive parity and predictive equality, of a rudimentary predictive policing algorithm based on four possible model types. We find that upsampling techniques, especially those like ADASYN that introduce synthetic noise into the predictors tend to provide the best performance and fairness outcomes. This is possibly a consequence of the fact that the dataset is rebalanced in a way that weakens the relationship between the omitted, protected feature and proxy features that are included in the model and which correlate with the protected variable. However, while it is generally clear that upsampling is preferable to downsampling, the impact of each technique critically depends on the type of model being used. This result is itself highly informative in terms of the range of factors that future research into fairness must consider.

I. Background

Many real-world applications of predictive algorithms involve predicting rare events. In the machine learning context, this means that there is a class imbalance, which poses a challenge to traditional methods of training classifiers since minimization of loss tends to favor making predictions that are biased toward the majority class. To achieve reasonably high rates of correct detection in the minority class, the classifier must therefore be especially sensitive to features associated with minority observations. But there may be substantial sparsity in feature values among the available minority-class observations due just to the limited number of such observations, making the chance of "stereotyping" more probable.

To address this, much work has been done to develop methods to rebalance the data used to train these classifiers. The goal of rebalancing is to alter the composition of the data so that the model is presented with a more diverse set of examples. Ideally, this will restrict the model's tendency to "stereotype" - to find a proxy variable that correlates with both the response variable and a protected attribute and to extrapolate this relationship - by showing it more examples from

All supporting data and analysis available at:
github.com/AnthonyRentsch/ac221_final?fbclid=IwAR2675HpYZGu-eKuAq02TrfoWfW4eDR91ywbWJFtoNjk5u-WuD0xZT9G5WM

the minority class. This is analogous to how exposure to a wider range of "examples" helps address stereotyping among humans, as has been formalized by ideas such as contact theory.

Rebalancing the training data before fitting a model on it is but one method of addressing class imbalance; using variable class weights or imbalance-correcting loss functions are frequently-adopted alternatives. However, rebalancing is the likely the best approach when attempting to correct for class imbalance when using "black-box" models like those often employed for purposes such as predictive policing.

Our project will focus on the effects of rebalancing on predictive policing algorithms, one of the most notorious classes of rare event-prediction algorithms. In particular, we are motivated by CivicScape's CrimeScape, an open-source crime-prediction algorithm designed to minimize bias.[1] A key component of their design is a patented mixed upsampling/downsampling rebalancing algorithm.[2] We therefore chose to draw on the same type of data this platform was designed to ingest: open-access crime records from the Boston Police Department.

II. Research Methodology

We decomposed the effect of resampling on algorithmic fairness into two components:

1. The systematic differences in how different rebalancing techniques impact the distributions of different features in the training data, and
2. How those differences in distribution impact a particular algorithm's predictions

The second component demanded that we consider how different classes of algorithms might respond to the same change in distribution. We therefore we examined the performance of four contrasting types of classifiers on the resampled dataset: a k-nearest neighbors classifier (k=5), a linear Support Vector Machine (SVM), a random forest classifier, and a multi-layer perceptron (MLP) with 2 layers containing 100 nodes each. We adopted the default scikit-learn hyperparameters for these models, reasoning that while hyperparameter tuning could potentially improve a model's performance, it would rarely impact the fundamental trends in behavior that the model exhibited.

We trained each classifier to predict whether or not at least one murder would occur in

[1] https://www.civicscape.com/
[2] http://www.freepatentsonline.com/y2018/0096253.html

each census tract over a particular month, using a feature set that mimicked the data that CivicScape's algorithm used to predict violent crimes: total monthly counts of vandalism, drug crime, and aggravated assault in each census tract between July 2015 and April 2019, as well as a set of month dummy variables. The crime data came from the Boston Police Department's publicly-available *Crime Incident Reports*. These observations should therefore be considered not as the "ground-truth" number of crimes that did occur, as they reflect the Department's current policing strategy (e.g. where police were stationed and their vigilance in pursuing certain types of crimes) and the variable likelihood that these types of crimes would be reported to the police.

We adopted the same "fairness through blindness" approach that CivicScape embraces by omitting the protected feature -- the racial composition of each tract -- from being explicitly used for classification. However, several other features that correlated with race were included, as Figure 1 illustrates. Moreover, Figure 2 illustrates that murders were especially highly-concentrated in minority-heavy tracts -- defined as having a lower share of white non-hispanic residents than the citywide average of 47% -- which means that any set of predictors that adequately modeled crime would also together be correlated with race.[3]

To mimic the training and testing conditions adopted by the CivicScape algorithm, we divided the dataset into training and test sets temporally, reserving the last year of data as the test set. The test set includes roughly 30% (38/131) of all murders in the dataset, and 26.7% (2136/8010) of all observations.

We performed five common resampling methods to even out the relative imbalance in number of observations in the positive class (for which there was a murder in the tract) and the negative class (for which there was not a murder in the tract) in the training set. These methods are summarized in Table 1. We then evaluated how the four models described above fared at predicting the presence of a murder in each tract in 2019.

[3] Data on racial composition came from the U.S. Census Bureau's 2013-2017 American Community Survey 5-Year Estimates.

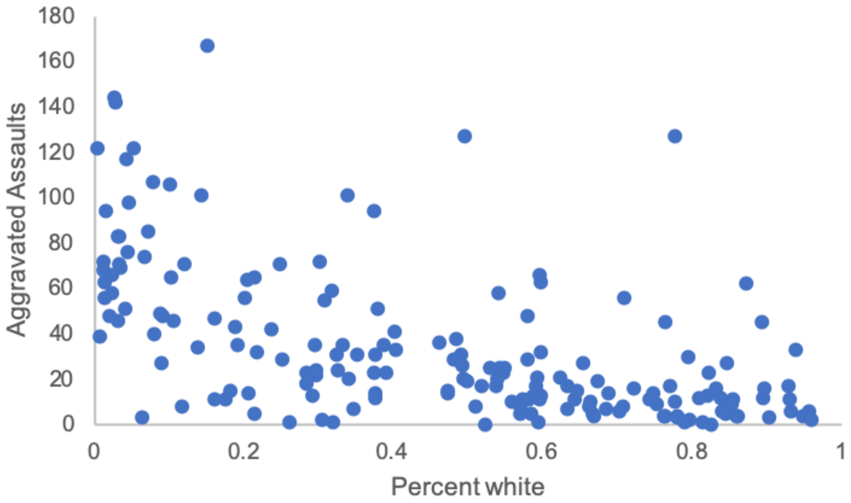


Figure 1. Relationship between tract-level racial demographics and number of reported aggravated assaults.

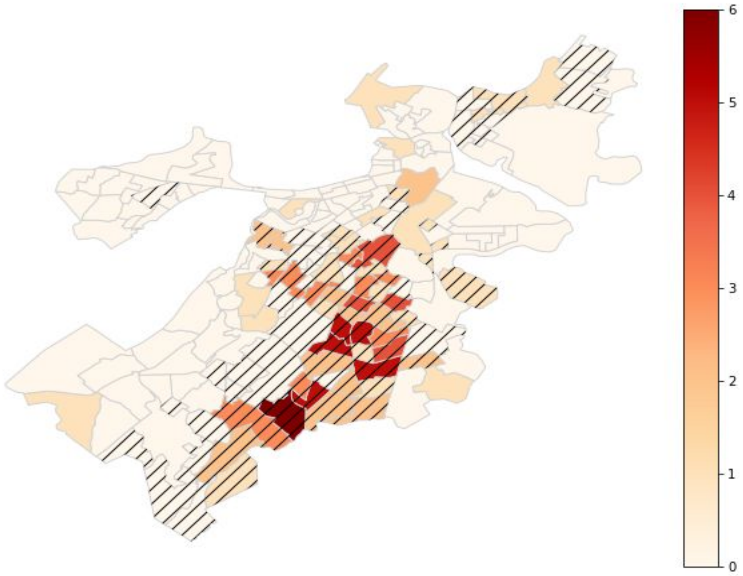
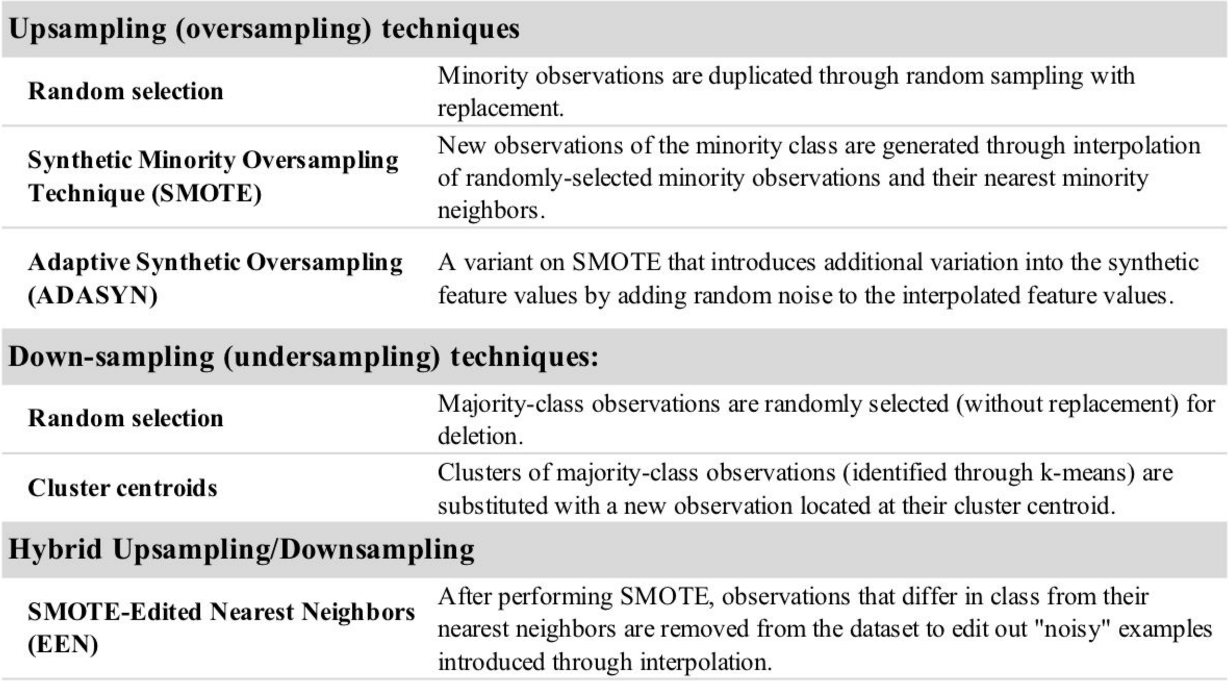


Figure 2. Co-occurrence of minority-heavy census tracts (cross-hatched) and murders. Shading indicates number of months with a murder.



Upsampling (oversampling) techniques	
Random selection	Minority observations are duplicated through random sampling with replacement.
Synthetic Minority Oversampling Technique (SMOTE)	New observations of the minority class are generated through interpolation of randomly-selected minority observations and their nearest minority neighbors.
Adaptive Synthetic Oversampling (ADASYN)	A variant on SMOTE that introduces additional variation into the synthetic feature values by adding random noise to the interpolated feature values.
Down-sampling (undersampling) techniques:	
Random selection	Majority-class observations are randomly selected (without replacement) for deletion.
Cluster centroids	Clusters of majority-class observations (identified through k-means) are substituted with a new observation located at their cluster centroid.
Hybrid Upsampling/Downsampling	
SMOTE-Edited Nearest Neighbors (EEN)	After performing SMOTE, observations that differ in class from their nearest neighbors are removed from the dataset to edit out "noisy" examples introduced through interpolation.

Table 1. Resampling techniques used to rebalance the training data.

III. Results

III.i. Distributional Effects of Resampling

While we are unable to directly measure the impact of the resampling methods on the distribution of the protected variable, as this variable is withheld from the training set, we can instead consider how they adjust the distributions of predictors that correlate with the protected variable. This is shown in Figure 3.

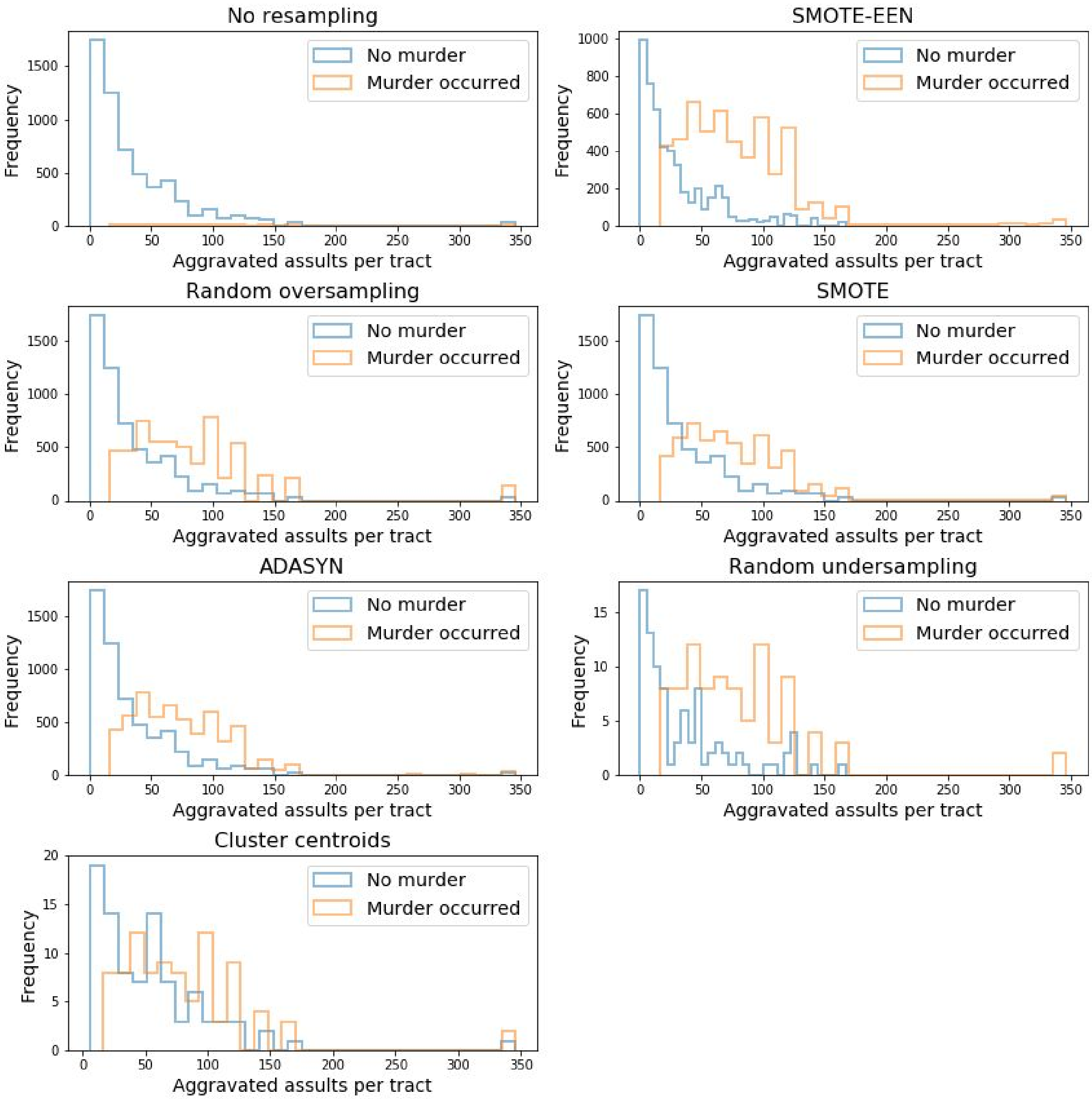


Figure 3. Distribution of aggravated assaults per tract in training data before and after applying various resampling techniques.

Whereas random oversampling retains the original distribution of the positive ("murder occurred") class, the two interpolative oversampling methods, SMOTE and ADASYN, smooth out the distribution, especially for tracts with moderately high (100-150) numbers of aggravated assaults. They also appear to reduce the relative number of outlier observations with abnormally high numbers of aggravated assaults. We interpret these distributional changes as providing the model with a more diverse set of reasonable values for the proxy feature, which in turn mutes the correlation between the proxy feature, which ends up being included in the model, and the protected feature, which is excluded from the model.

Among undersampling techniques, the "cluster centroids" method similarly increased the spread of the majority (no-murder) distribution, while random undersampling merely made that group's distribution less even.

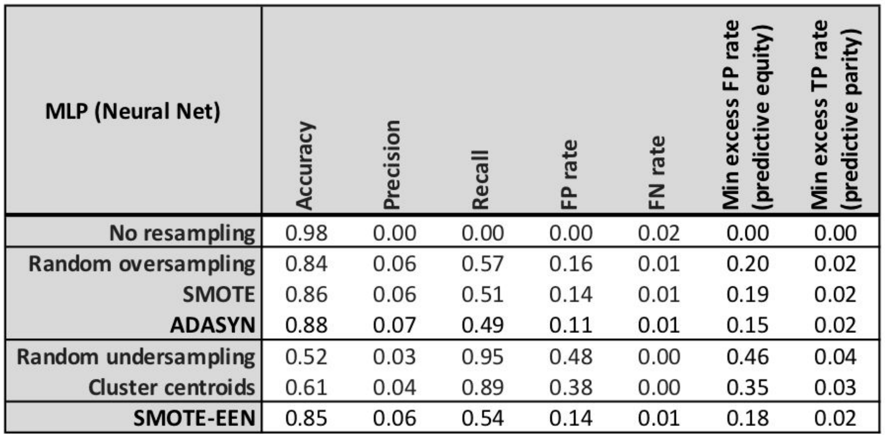
III.ii. Overall Trends in Performance and Fairness Metrics

To measure fairness in our model's predictions, we consider a handful of metrics. (Note: When we say "positive" in this context we are referring to our model predicting that a murder occured in a Census tract and when we say "minority" we are referring to Census tracts with non-white populations above the city average.)

- *Precision*: proportion of true positives out of all predicted positives. High precision indicates that the model is not too "noisy": out of all the murders predicted, many did in fact occur.
- *Recall*: proportion of true positives out of all correct positive predictions. High recall indicates that the model is successful at identifying events: of the murders that actually occurred, many were correctly identified by the model.
- *Minority excess false-positive rate (predictive (in)equality)*: Difference in the false positive rate between the minority and majority classes; we have more predictive equality as this approaches 0.
- *Minority excess true-positive rate (predictive (dis)parity)*: Difference in the true positive rate between the minority and majority classes; we have more predictive parity as this approaches 0

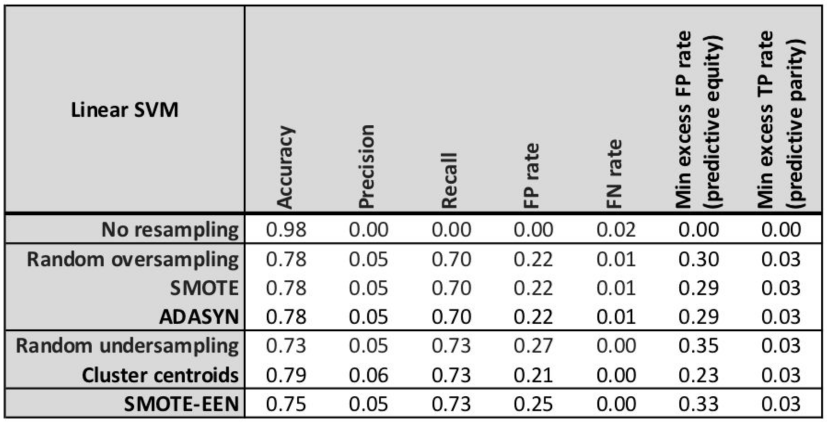
Overall, the trends that we observed in the impact of different resampling techniques (discussed in detail below) suggest that rebalancing techniques that introduce more diversity in the characteristics of crime-heavy tracts -- i.e. increase the variability of the predictors included in the model in a way that reduced their correlation with racial composition while retaining their correlation with murder -- generally achieve higher levels of both predictive equality and predictive parity. Interpolative upsampling methods like SMOTE and especially ADASYN performed best at improving the fairness criteria because they introduce new "positive" (murder-present) examples with feature values not found in the original dataset, which reduces the correlation between these features and racial composition.[4] Notice in Table 2 that predictive parity and predictive equality rates are typically lowest when SMOTE or ADASYN are applied, for all four model types.

Conversely, resampling methods that effectively reduce diversity of the proxy variable of high-crime tracts appeared to induce more bias in the predictions by "confirming" the correlation between crime and the proxy variable. Looking at Table 2, we see that predictive equality and predictive parity rates tended to be the highest when we applied random undersampling or cluster centroids.

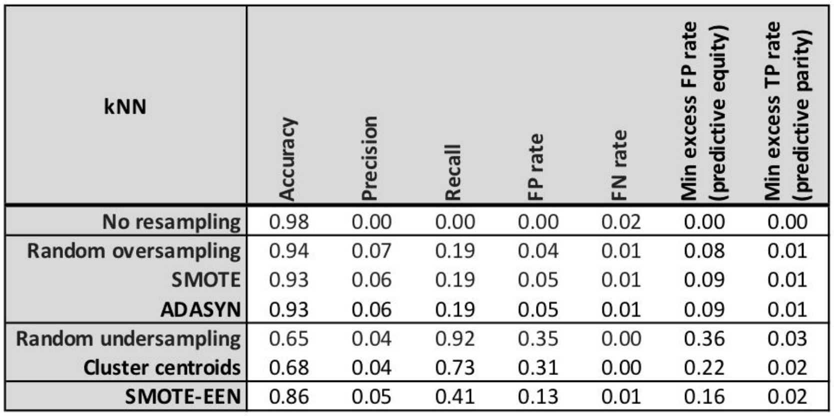


MLP (Neural Net)	Accuracy	Precision	Recall	FP rate	FN rate	Min excess FP rate (predictive equity)	Min excess TP rate (predictive parity)
No resampling	0.98	0.00	0.00	0.00	0.02	0.00	0.00
Random oversampling	0.84	0.06	0.57	0.16	0.01	0.20	0.02
SMOTE	0.86	0.06	0.51	0.14	0.01	0.19	0.02
ADASYN	0.88	0.07	0.49	0.11	0.01	0.15	0.02
Random undersampling	0.52	0.03	0.95	0.48	0.00	0.46	0.04
Cluster centroids	0.61	0.04	0.89	0.38	0.00	0.35	0.03
SMOTE-EEN	0.85	0.06	0.54	0.14	0.01	0.18	0.02

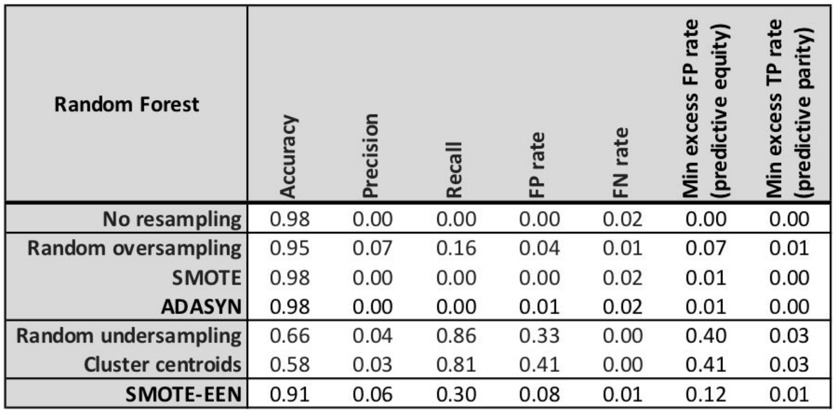
[4] This result is consistent with Rokach and Maimon's (2005) assertion that "the synthetic examples [created from SMOTE] cause the classifier to create larger and less specific decision regions." Chawla (2005) similarly notes that "SMOTE provides more related minority class samples to learn from, thus allowing a learner to carve broader decision regions, leading to more coverage of the minority class." See Rokach and Maimon. 2005. *Data Mining and Knowledge Discovery Handbook*. (p. 860) and Chawla. 2005. *Data Mining for Imbalanced Datasets: An Overview*. (p. 862).



Linear SVM	Accuracy	Precision	Recall	FP rate	FN rate	Min excess FP rate (predictive equity)	Min excess TP rate (predictive parity)
No resampling	0.98	0.00	0.00	0.00	0.02	0.00	0.00
Random oversampling	0.78	0.05	0.70	0.22	0.01	0.30	0.03
SMOTE	0.78	0.05	0.70	0.22	0.01	0.29	0.03
ADASYN	0.78	0.05	0.70	0.22	0.01	0.29	0.03
Random undersampling	0.73	0.05	0.73	0.27	0.00	0.35	0.03
Cluster centroids	0.79	0.06	0.73	0.21	0.00	0.23	0.03
SMOTE-EEN	0.75	0.05	0.73	0.25	0.00	0.33	0.03



kNN	Accuracy	Precision	Recall	FP rate	FN rate	Min excess FP rate (predictive equity)	Min excess TP rate (predictive parity)
No resampling	0.98	0.00	0.00	0.00	0.02	0.00	0.00
Random oversampling	0.94	0.07	0.19	0.04	0.01	0.08	0.01
SMOTE	0.93	0.06	0.19	0.05	0.01	0.09	0.01
ADASYN	0.93	0.06	0.19	0.05	0.01	0.09	0.01
Random undersampling	0.65	0.04	0.92	0.35	0.00	0.36	0.03
Cluster centroids	0.68	0.04	0.73	0.31	0.00	0.22	0.02
SMOTE-EEN	0.86	0.05	0.41	0.13	0.01	0.16	0.02



Random Forest	Accuracy	Precision	Recall	FP rate	FN rate	Min excess FP rate (predictive equity)	Min excess TP rate (predictive parity)
No resampling	0.98	0.00	0.00	0.00	0.02	0.00	0.00
Random oversampling	0.95	0.07	0.16	0.04	0.01	0.07	0.01
SMOTE	0.98	0.00	0.00	0.00	0.02	0.01	0.00
ADASYN	0.98	0.00	0.00	0.01	0.02	0.01	0.00
Random undersampling	0.66	0.04	0.86	0.33	0.00	0.40	0.03
Cluster centroids	0.58	0.03	0.81	0.41	0.00	0.41	0.03
SMOTE-EEN	0.91	0.06	0.30	0.08	0.01	0.12	0.01

Table 2. Evaluation of our four models' performance and fairness using various metrics and resampling techniques. (a) Multi-Layer Perceptron (MLP), (b) Linear Support Vector Machine (SVM) (c) k-Nearest Neighbors, (d) Random Forests.

As we would expect, all four models performed equally poorly when the class imbalance went untreated, defaulting to the "naive" approach of always predicting the majority class (no murders).

Random oversampling caused a dramatic change in the behavior of the linear SVM and MLP models, which predicted many of the murders correctly, though also suffered from very high false-positive rates. This result is consistent with literature noting that random oversampling tends to induce overfitting of the minority class.[5] But the nonparametric kNN and random forest classifiers were surprisingly much less sensitive to this resampling method.

The models also responded somewhat differently to the two interpolative oversampling methods, SMOTE and ADASYN. Whereas the impact of these oversampling methods was about comparable to random oversampling for the kNN and linear SVM classifiers, the random forest model was completely uninfluenced by the changed distribution.

For the MLP, SMOTE and, in particular, ADASYN, were clearly the most effective resampling techniques for improving the precision and recall (relative to performing no rebalancing) without simultaneously incurring high false positive rates. Relative to simple random oversampling, performing SMOTE and ADASYN increased the MLP's accuracy and precision and decreased its recall, suggesting that these sampling techniques caused the model to be more conservative, identifying murders only when it was more certain of its prediction.

The two undersampling techniques caused comparably high overfitting for all four models, resulting in elevated recall rates in tandem with very high false positive rates. For the SVM, these results were comparable to those achieved using random oversampling, while the results were better using oversampling for the other three models. In fact, for the MLP, random undersampling causes a complete reversal in which the model always predicts that a murder will occur. The random undersampling and cluster centroid undersampling approaches are therefore unambiguously inferior to the oversampling methods, as the lost information causes the models to mistake what were previously negative predictions instead as false positives.

Finally, combining SMOTE and EEN offered noticeable improvements in recall for all models except the MLP relative to SMOTE alone, though also increased the false positive rates,

[5] Chawla. 2005. *Data Mining for Imbalanced Datasets: An Overview*.

such that the overall precision did not dramatically change (with the exception of for the random forest model, which predicted a positive number of murders after introducing EEN).

IV. Conclusions

On a broader level, our results highlight how critical the choice of resampling technique can be to a model's performance, and how difficult it may be to predict what impact it will have when applied to any particular dataset. Even if CivicScape publishes the code to its algorithm, and allows users to run the code on specific datasets that are also publicly available, members of the public will still have difficulty predicting how the CrimeScape model will behave without knowing in detail how the data are rebalanced (the specifics of which we were not able to ascertain from their source code).

Extensive research has been carried out on how different methods for collecting samples of new observations, removing outliers, and anonymizing existing datasets might bias the results of analyses performed on the dataset without due consideration of how these data-manipulation and pre-processing choices interact with the choice of algorithm. But our analysis demonstrates that different types of models will respond in markedly different ways to the same changes in the underlying data.

While there is a lot of nuance in how these different resampling techniques interact with different machine learning algorithms, there is an inherent tradeoff between precision and recall, as shown in Figure 3. With different resampling techniques we can improve our model's ability to predict positive cases but we have to concede that we will also make more errors. These types of tradeoffs exist throughout machine learning and our project demonstrates the case of resampling and fairness is no exception.

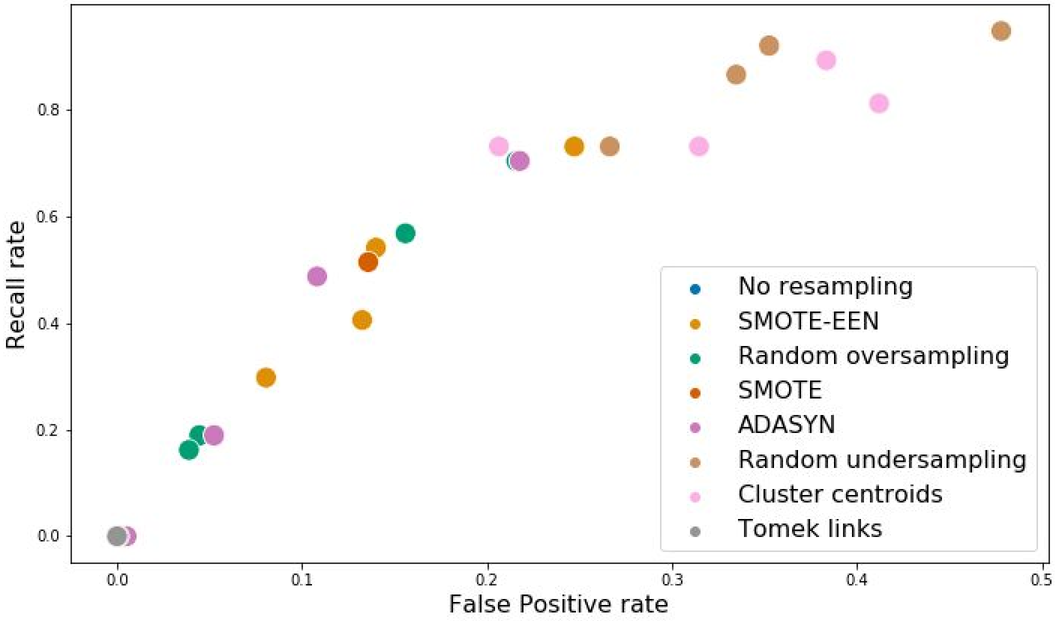


Figure 3. False positive vs. recall rate for different resampling techniques. Different colors correspond to different resampling techniques. Each dot of the same color corresponds to one of our four models. The resampling technique Tomek links is included in this plot, even though it is omitted from our discussion in this paper.

With that being said, however, we do find that techniques that oversample the underrepresented class and add some noise to those new observations tend to fare better on two widely-adopted criteria of predictive fairness. We interpret these techniques that do this -- specifically SMOTE and ADASYN -- as effectively performing regularization by restricting the model's ability to overfit to the observed “stereotypical” examples of crime.