Assessing Accuracy Disparities in Discrimination-Free Naïve Bayes Classification

Amina Abdu
amina.abdu@gmail.com

Lavanya Singh
lsingh@college.harvard.edu *

Abstract

Employment pre-screening is increasingly becoming automated. Given the potential for algorithms to produce discriminatory results, this is a cause for concern: can we rely on these algorithms to be fair? In this work, we explore whether U.S. employment law has encouraged the algorithmic hiring industry to ignore racial disparities in model performance. We implement a binary naïve Bayes classification scheme in which we drop variables highly correlated with race, a common tactic in algorithmic hiring for ensuring compliance with non-discrimination laws. We compare the resulting model to a basic naïve Bayes model with no fairness constraints and to an approach from Calders and Verwer for discrimination-free classification that violates Title VII. An experimental comparison of these models on ACS data shows that all the models perform equally as well and that differential accuracy impacts are limited. We highlight the implications of our results for policy makers and data scientists as the market for algorithmic pre-employment assessments continues to grow.

1 Introduction

As artificial intelligence is increasingly used to make consequential decisions once considered to be under the purview of human judgment, ensuring algorithms make fair, unbiased decisions is more important than ever. Despite growing interest in this question in the computer science literature, scholars have yet to come to a consensus regarding what we mean when we say an algorithm ought to behave "fairly".[1] In the absence of such a consensus, the law may provide the clearest available guidance on domain-specific social and ethical values.

In this work, we use the example of commercial hiring software to explore the ways in which law can shape ethical practice in emerging technologies. In particular, we consider the question of whether the legal emphasis on differential treatment and differential impacts come at the cost of another form of injustice: differential accuracy.

1.1 U.S. Employment Law and Algorithmic Hiring

Tile VII of the Civil Rights Act of 1964 is the cornerstone of U.S. anti-discrimination law in employment, prohibiting discrimination on the basis of a number of protected attributes—"race, color, religion, sex, or national origin" and establishing the Equal Employment Opportunity Commission (EEOC) to oversee compliance.[2] The EEOC later issued the Uniform Guidelines on Employment Selection Procedures, which lay out in greater detail the regulatory standards for fair employment practices. In particular, the Uniform Guidelines establish two avenues for bringing a lawsuit on the basis of discrimination: disparate treatment and disparate impact.

Disparate Treatment Disparate treatment concerns the relatively straightforward case of explicit discrimination. Employers cannot treat job candidates differently based on attributes protected by

*Code can be found at https://github.com/lsingh123/ac221finalproject

Title VII. In the case of hiring software, this has been taken to mean that, because machine learning models are designed to explicitly discriminate based on their inputs, algorithmic pre-employment assessments should not take protected attributes as inputs.[3]

Disparate Impacts The Uniform Guidelines go further than simply outlawing explicit discrimination based on inputs— they also consider unfair differences in outcomes, or disparate impacts. In particular, the Uniform Guidelines propose a rule of thumb called the "4/5 Rule" as a legal standard for when a disparate impact lawsuit can be brought against an employer. The "4/5 Rule" states that the selection rate (in other words, the proportion of people receiving job offers) for any protected group should be at least 4/5 of the selection rate of the group with the highest selection rate. If an employer falls below this standard, they become liable to a discrimination lawsuit, at which point they are required to prove the validity and necessity of these differential impacts if a suit is brought. [4]

Employers have strong incentives to remain 4/5 compliant in order to avoid liability. Because employers are by far the biggest client base for algorithmic pre-employment assessments, companies that design such assessments have similarly strong incentives to design their models for 4/5 compliance, and a 2020 survey of company policies suggests that they do.[5]

1.2 De-Biasing Methods

While there are many ways to audit and design algorithms for 4/5 compliance, we focus here on two of the biggest players in the market, HireVue and pymetrics, both of whom have released relatively detailed accounts of their "de-biasing" processes.

Both companies use similar methods focused on addressing bias introduced through variables highly correlated with protected attributes. [6, 7] One way in which an algorithm with no knowledge of protected attributes may introduce disparate impact is through the use of these proxy variables. Take for example last name, which is often correlated with ethnicity. A hiring algorithm with no knowledge of race or ethnicity may still exhibit bias against a particular ethnicity if it uses last name as an input.

To avoid this problem HireVue and pymetrics test their models for disparate impacts using the 4/5 Rule as a guideline and, if they find evidence of bias, look for model inputs that are highly correlated with protected classes, remove these variables [2], and re-train their models. They repeat this process, dropping new variables each time, until they determine that their model is 4/5 compliant.

This process of iteratively removing proxies necessarily means that the de-biased models are not using a number of variables the original training scheme found useful, but both companies deny that this technique affects their products' ability to accurately predict job performance. pymetrics states that they are able to "improve the fairness of our models, without sacrificing predictive power,"[7] while HireVue claims that it is "possible to remove problematic factors and still maintain a highly accurate predictive capability,"[6] citing the large number of predictive variables to which they have access.

1.3 Differential Accuracy

While HireVue and pymetrics argue that dropping proxies does not affect overall model performance, they say nothing about impacts on model performance for subgroups. It is possible to achieve high global accuracy with high accuracy on the majority class and significantly lower accuracy on minority classes (which may correspond with protected groups). We are interested in exploring whether this may be an unintended consequence of dropping proxy variables.

We are concerned that the existing legal framework in the hiring system, as a result of its focus on disparate treatment and disparate impacts, has encouraged algorithmic hiring companies to overlook a third, more subtle, form of unfairness: disparate accuracy. Taken to its extreme, we can imagine a world in which, for example, 10% of male candidates were chosen for hire based on an in-depth interview process and 10% of female candidates were chosen for hire completely at random. While there is technically no disparate impact, as defined by the 4/5 Rule, the fact that the qualifications of the female candidates become irrelevant clashes with our notions of individual-level fairness. Moreover, the men who were hired because of their qualifications may experience greater long-term success their female counterparts, leading to adverse impacts down the line.

[2] pymetrics indicates that they sometime use feature regularization instead of removal.

This toy example illustrates the importance of checking not only for disparate outcomes, but also for disparate model validity, which we aim to do in this work. More specifically, we evaluate what happens to accuracy, overall and broken down by race, in a proxy-dropping model in the spirit of HireVue and pymetrics. We compare the proxy-dropping technique to a de-coupled classifier technique from the algorithmic fairness literature that violates Title VII, using disparate treatment in order to correct for bias in outcomes. Through this comparison, we explore the trade-offs between disparate treatment and disparate accuracy and consider whether U.S. employment law has pushed us too far in one direction.

The remainder of this paper is organized as follows. In section 2, we review existing work on decoupled models and outline the methods we use in our experiment. In Section 3, we implement our own proxy-dropping algorithm for binary classification and compare it with alternative classification schemes. In Section 4, we discuss some of the implications of our experiment on the interplay between employment law and algorithmic hiring assessments. Finally, we conclude with closing remarks and areas for further exploration in Section 5.

2 Methods and Related Work

2.1 Related Work

In 2012, Calders and Verwer[8] presented three possible ways to create discrimination-free naïve Bayes classifiers. They briefly considered the idea of dropping proxies, but instead opted for a "2NB" classifier: a separate naïve Bayes classifier trained for each of the two subsets of a binary sensitive attribute. The target variable was also binary. They argued that this model would preserve the information that the proxy variables provided, and should therefore increase accuracy.

Calders and Verwer tested their methods using 10-fold cross validation on Census data. The task was to classify individuals as low or high income, and the sensitive attribute was sex. Out of the three discrimination-reducing models that they tested, the 2NB model far outperformed its counterparts. They concluded that the 2NB model was able to maintain high accuracy while achieving 0 discrimination.

In 2018, Dwork et al.[9] provide a theoretical and practical generalization of the 2NB model, coining the term "decoupled classifiers," in which subsets of the dataset divided by sensitive attribute value are "decoupled." They experimented with a wide range of classification tasks and techniques on over 47 datasets, and found that decoupling often improved performance more often than it decreased it. They conjectured that,"Using sensitive attributes may increase accuracy for all groups and may avoid biases where a classifier favors members of a minority group that meet criteria optimized for a majority group."[9] Their method involved calculating a joint loss function that penalized differences in classification across groups, pushing the overall model towards homogeneous classification statistics.

2.2 Goals

Related work in the field has traditionally presented new, sophisticated models, measured their discrimination and accuracy, and compared these values to a basic model. We would like to extend this process to include the HireVue method of dropping proxy variables until a certain fairness standard is reached. We would also like to combine both Dwork et al's and Caders and Verwer's methods to create a classifier with 0 discrimination that can handle non-binary sensitive attributes, like race.

Most prior work seeks to optimize traditional measures of accuracy for the overall models. Our goal to examine our set of models for differential accuracy impacts. Specifically, we would like to compare accuracy impacts for the possible values across the different values that the sensitive attribute can take on, both binary and categorical. It is possible to achieve high overall accuracy values while having wildly different differential accuracies across different subgroups, and we would like to test whether or not this is the case.

2.3 Models

We will examine 4 different models over the course this paper, all focusing on a classification task with a binary target variable, a nonbinary sensitive attribute, and categorical fields. In this context, the term "sensitive attribute" refers to the field that we do not wish to discriminate on, like race or sex.

1. **Basic Model**: This is a naïve Bayes classifier that uses all of the given fields in a dataset. This model makes no guarantees about discrimination, and likely to have discrimination for classification tasks in which the sensitive attribute does in fact predict the target variable.
2. **"HireVue" model**: This model implements HireVue's preferred method for reducing discrimination. The model starts with the basic model, and iteratively removes the attribute most strongly correlated with race until the resulting classifier meets the 4/5 standard. To measure the 4/5 standard, we compare the selection rate of the group selected least often to the selection rate of the group selected the most often. Pseudocode for the algorithm we used is below:

```
fields = data
model = train(fields)
while (model does not meet 4/5 standard):
    var = field in fields most correlated with race
    fields.drop(var)
    model = train(fields)
return model
```

3. **2NB model**: This is a direct implementation of the model proposed by Calders and Verwer. The sensitive attribute is converted to a binary value for all entries, and the dataset is decoupled along the two possible values for this attribute. Then, two separate models are trained on each decoupled subset. To classify a given entry, the proper decoupled model is determined based on the value of the sensitive attribute for the test field and then the appropriate model is used.
4. n**NB model**: The nNB model is our application of Dwork et al.'s decoupled classifier algorithm to the Calders and Verwer approach. The sensitive attribute is treated as a categorical field with n possible values. The dataset is decoupled into n different subsets, each corresponding to a given value for the sensitive attribute. A separate model is trained for each of these possible n values. To classify a given entry, the proper decoupled model is determined based on the value of the sensitive attribute for the test field and then the appropriate model is used. Pseudocode for the algorithm is below:

```
# determine the n possible values of the sensitive attribute
values = set(sensitive_attribute)

# train n possible models
models = {}
for value in values:
    model = train(data[sensitive attribute == value]
    models[value] = model

# classify a test point
value = test_point[sensitive attribute]
classifier = models[value]
return classifier(test_point)
```

To standardize our implementations of the algorithms, we used scikit-learn[10], an out of the box machine learning library for Python. Because the goal of this project was to compare models, not design the most accurate model possible, we felt comfortable using an out-of-the-box solution, since in this case programmer time was more important than accuracy. We used the `categoricalNB` classifier, which is designed for categorical fields and target variables. This classifier made sense given that the sensitive attributes we are interested in are categorical.

2.4 Data and Classification Task

We used data from the 2015 American Community Survey conducted by the United States census.[3] The classification task in question is to classify individuals as high or low income, where low income is defined as the lower quartile of the income distribution, so the lower 25% of the dataset. The sensitive attribute S in question is race, which is a protected class and therefore protected by the 4/5 standard in hiring law. For the 2NB model, we divided race into white and nonwhite, and for the nNB model, we used the field RAC1P, which corresponds to the Census's least detailed race code. The more detailed race fields took on over 50 values, and decoupling the dataset into over 50 subgroups would have rendered each subgroup too small to produce accurate results. This field we chose has 9 possible values, corresponding to following possible races:

1. White alone
2. Black or African American alone
3. American Indian alone
4. Alaska Native alone
5. American Indian and Alaska Native tribes specified; or American Indian or Alaska Native, not specified and no other races
6. Asian alone
7. Native Hawaiian and Other Pacific Islander alone
8. Some Other Race alone
9. Two or More Races

For analysis purposes, we collapsed some of the racial categories, but when constructing the nNB model, we preserved these racial categories.

This dataset contains rich demographic information including citizenship status, language spoken, race, geographic information, income, and demographic details for the parents of each individual. We intentionally chose a dataset that not only included fine-grained information about each individual's race, but also included lots of proxies for race. This will give us lots of room to drop proxies to reduce discrimination. Additionally, the proxies in this dataset are "nontrivial" proxies in the sense that many of them, like citizenship status and language spoken at home, also offer valuable information relevant to the classification task at hand. Something like citizenship status should, intuitively, have an impact on an individual's employment prospects and therefore on their income classification.

The dataset contained hundreds of thousands of rows and included rich demographic information. During the cleaning process, we examined the dataset for any fields with missing values for the race or income field and threw them out. Fields with missing race values would be impossible to classify using the 2NB or nNB classifiers, and it would be impossible to verify the accuracy of the classification of a field with a missing income value. For all other fields, we allowed missing values and included them in our analysis (census data represents missing values with a value of "-1" so we did not need to modify missing values). We did this because missing fields can provide valuable information that might be useful to a classifier. For example, a certain demographic might be more or less likely to completely fill out the ACS, so we decided not to treat missing values like garbage.

Because we used sklearn's categoricalNB classifier, we needed to convert all of our fields to categorical data. To do so we binned any continuous information. Specifically, this required binning the fields corresponding to travel time, age, income from government assistance, and income (described above). We binned into 4 categories corresponding to each quartile, except for income, which is described above.

[3] data can be downloaded here in csv form: `https://www.kaggle.com/census/2015-american-community-survey#ACS2015_PUMS_README.pdf`

3 Results

3.1 Accuracy

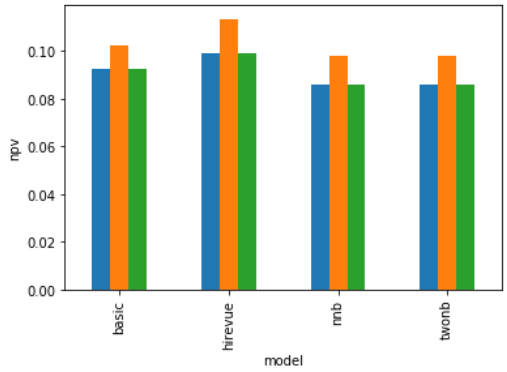
We analyzed three different measures of accuracy: positive predictive value (ppv), negative predictive value (npv), and accuracy (acc). These values are calculated using the following equations:

$$\text{ppv} = \frac{\text{\# of true positives}}{\text{\# of true positives + \# of false positives}}$$

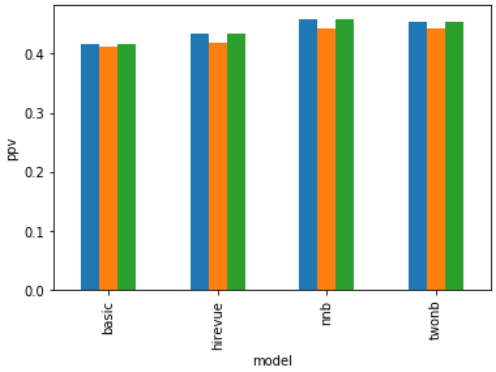
$$\text{npv} = \frac{\text{\# of true negatives}}{\text{\# of true negatives + \# of false negatives}}$$

$$\text{acc} = \frac{\text{\# of true positives + \# of true negatives}}{\text{size of test population}}$$

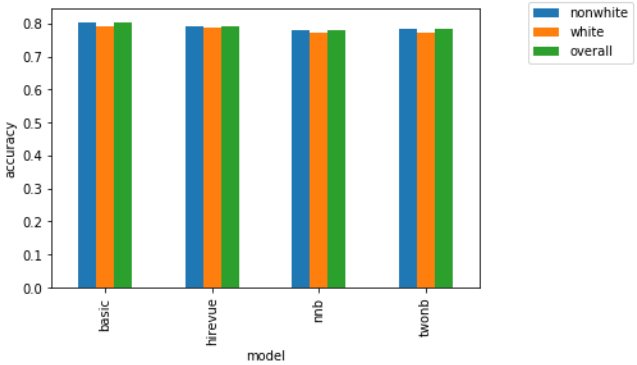
We compare how each of our 4 models performed overall based on these 3 measures of accuracy.



(a) negative predictive value



(b) positive predictive value



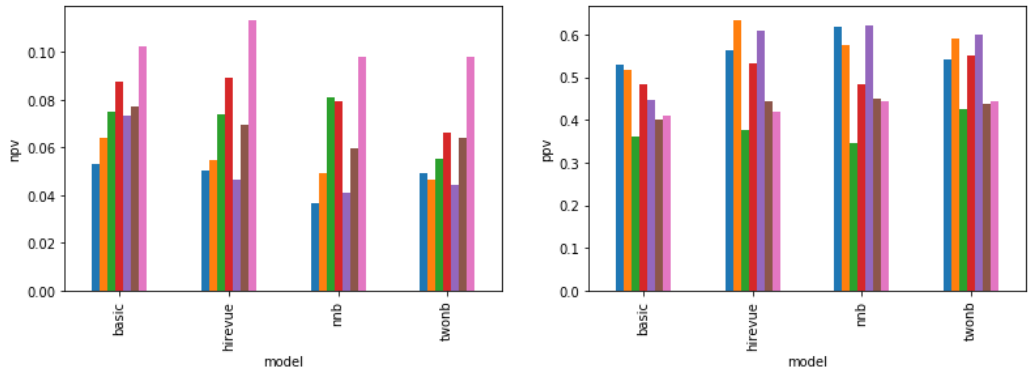
(c) accuracy

Figure 1: Accuracy Measures by Model

As seen in the figures above, the change in any given accuracy measure for any model is tiny. No change is ever greater than 0.02, and all models tend to perform equally as well. There is some fluctuation in negative predictive value and positive predictive value, with negative predictive value being highest for the HireVue model and positive predicting value being highest for the nNB model, but these changes are too small to be significant, indicating that all of the models perform equally as well in terms of accuracy.

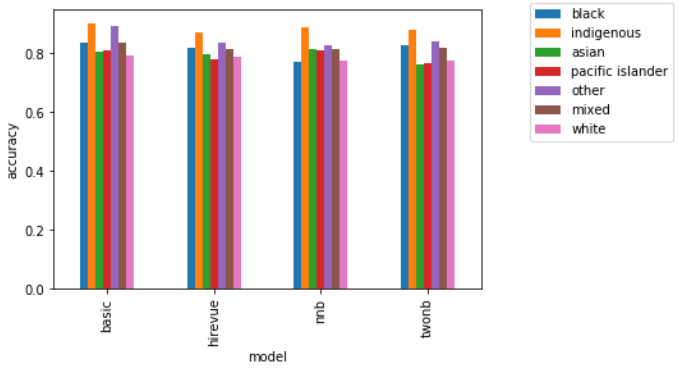
3.2 Differential Accuracy

Another one of our goals was to examine whether or not changes in accuracy vary across race. The figure below providers more fine-grained analysis, demonstrating how the different accuracy measures vary by race:



(a) negative predictive value

(b) positive predictive value



(c) accuracy

Figure 2: Differential Accuracy Measures by Model

The figure above indicates that accuracy measures tend to vary across race. The shape of these variations is similar across models, but some models appear to have larger differential impacts than others. specifically, there seems to be more variation in the nNB model's negative predictive value and more variation in the HireVue model's positive predictive value. The variances across race in the different accuracy measures for each model are displayed below:

measure	multirace	hirevue	nnb	twonb
npv	0.000252	0.000572	0.000533	0.000347
ppv	0.003948	0.009854	0.010771	0.005713
accuracy	0.001829	0.000976	0.001479	0.001978

Figure 3: Variance in Accuracy Measures

To test if these variances are significantly different, we used a Brown-Forsythe test. $p > 0.3$ no matter which combination of models and measures we compare, indicating that the differences in variances are not statistically significant.

3.3 A Closer Look at Proxy Dropping

Binary Race Model (white/nonwhite) In our experiment, we had to drop 11 (of a total 37) variables in addition to race in order to meet the 4/5 standard. These variables are listed in the order in which they were dropped:

- POBP: place of birth
- LANP: language spoken at home
- ENG: ability to speak English
- LANX: language other than English spoken at home (yes, no, N/A)
- CIT: citizenship status
- NATIVITY: native or foreign-born
- ST: state
- POWPUMA: place of work (public use microdata area code based on 2010 Census definition)
- MARHYP: year last married
- MAR: marital status
- MARHD: divorced in the past 12 months (yes, no, N/A)

Because we drop nearly a third of all explanatory variables, we expect to see declines in accuracy. However, as proxies are dropped, accuracy does not decrease monotonically and in fact there is no clear relationship between number of proxies dropped and accuracy, as illustrated in Figure 4 below.

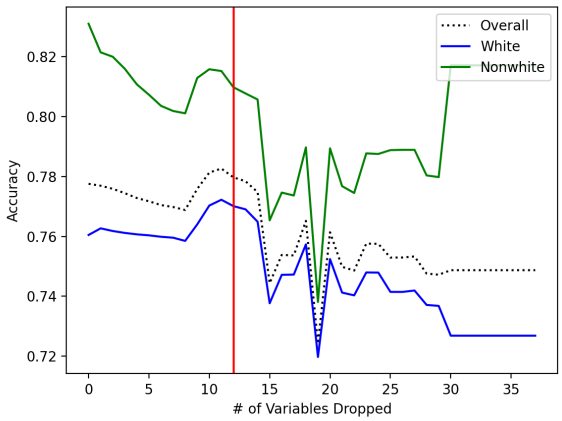
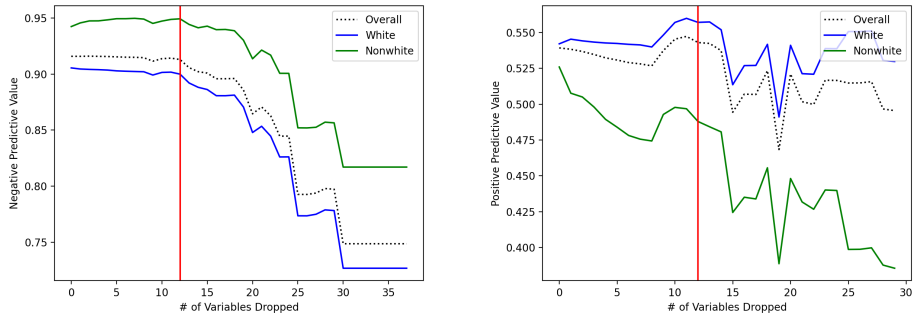


Figure 4: Accuracy as a function of number of variables dropped. The red line represents the point at which the 4/5 rule was respected.

We also looked at negative predictive value and positive predictive values as proxies are dropped. These results, displayed in Figure show that that both predictive negative value and predictive positive value fall as proxies are dropped, and that this effect is especially clear for the nonwhite group.



(a) Predictive Negative Value (b) Predictive Positive Value

Figure 5: Negative Predictive Value and Positive Predictive Value as a function of number of variables dropped. The red line represents the point at which the 4/5 rule was respected.

In order to get a fuller picture of how the model was performing as proxies were dropped, we also looked at the difference in mean income between those selected for the high income group and those not selected. The idea is that we don't care as much if someone in the 26th percentile of income is classified as being in the high income group as we do if someone in the 3rd percentile is misclassified. We expect that if the model is doing a relatively good job sorting between high and low income group, the difference in mean incomes between these groups will be quite high. The results, shown in Figure 6 show that the difference in mean income between the selected and not selected groups falls almost monotonically as variables are dropped, and that this effect is stronger for the nonwhite group.

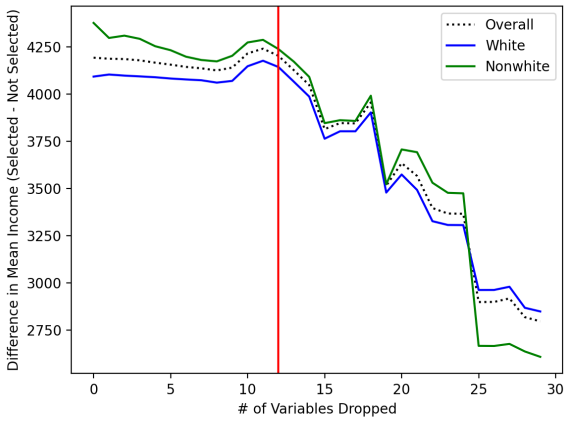


Figure 6: Difference in mean income between the selected and not selected groups as a function of number of variables dropped. The red line represents the point at which the 4/5 rule was respected.

Multi-Race Model The multi-race HireVue model never respected the 4/5 rule. Even as we dropped proxies for race, the group with the lowest selection rate was still selected at a rate of less than 80% of the highest selection rate, up until the model stop selecting anyone in any group.

4 Discussion

4.1 Accuracy

We found that all the models performed about as well as each other. This is surprising: our hypothesis was that the HireVue model would lose accuracy because of the loss of information from the dropped proxies. One possible reason for the stability in accuracy may be that race was not a strong predictor

of the bottom quartile of income, even if race and income are generally correlated. If this is the case, dropping race and its proxies may not affect accuracy.

We also observed that differences in accuracy measures across race tend to be similar across the models. While there is some variance in accuracy measures across race, this variance holds for all the models and the general shape of these variances does not change. This indicates that all the models perform about as well as each other even differentially.

One feature of our accuracy measures is that we used the exact same testing and training data for each model. This approach makes our results more robust to variations within the dataset as well, but this is unlikely to reflect implementations of the model in the real world. Real implementations of our models would likely use k-fold cross-validation or some other metric to choose the best set of training data possible. We tried 10-fold cross validating each of our models and found that the testing and training data for each model ended up differing significantly. That being said, our basic results about the stability of accuracy and variance held even with 10-fold cross validation, but there is room for further work to explore how differences in the selection of training data may affect accuracy measures across models.

In particular, using k-fold cross validation will change the proportion of each racial group included in the training sample for the 2NB and HireVue models. Specifically, the nNB model will have more data for minority racial groups, so may be more accurate for them. Preliminary work demonstrated that this property did not hold, but there is room for further exploration.

4.2 Falling Selection Rates

One possible reason that accuracy does not continue to fall as proxies are dropped is the model's tendency to assign everyone the modal label (in this case, "low income") in the absence of good predictors. This behavior results in approximately 75% accuracy by design, which is not much lower than initial accuracy. This is worthy of further exploration as the size of the selected group for a pre-employment assessment (analogous to our "high income" label) would likely have to be held constant, and therefore could not be allowed to fall to 0 as it does in our experiment.

4.3 Effects of Dropping Proxies

Considering overall accuracy race as proxies were dropped surprisingly revealed little change in model validity in spite of reduced information. However, using other measures of validity exposed some trends that suggest there might be more going on beneath the surface. In particular, looking at predictive negative value, predictive positive value, and the difference in mean income between the selected and not selected groups showed that the model might be losing its ability to make meaningful predictive distinctions about income as variables were removed from the inputs. Additionally, these impacts seem to be particularly concentrated among the nonwhite population, perhaps suggesting the existence of disparate accuracy effects. This is evidence that perhaps the accuracy impacts of HireVue's and pymetrics's methods are not as minimal as they seem and greater work should be done to evaluate the method of dropping proxies.

Overall accuracy measures may have stayed stable because proxy-dropping does not begin to drastically affect accuracy measures until far more proxies are dropped than are required to respect the 4/5 rule. For classification tasks for which race is a strong predictor, more proxies will have to be dropped to achieve compliance, so the accuracy impacts will be stronger. Hiring data is not, generally, available to researchers, so whether or not more proxies need to be dropping in Hirevue's model invites further work.

4.4 Failure to Achieve 4/5 Compliance in the Multi-Race HireVue model

In our results, we found that when using the full categorization of race (rather than the white/nonwhite binning), no number of proxies dropped made the model 4/5 compliance in our tests. While we don't know the details of HireVue's data or model, this was an interesting finding because it calls into question whether the de-biasing methods outlined on HireVue's website work all of the time. Because the incentives for compliance are reasonably high, we assume that they implement alternative solutions to generate fair outcomes. If so, HireVue should be clear about this when describing their de-biasing process.

4.5 Research Ethics

In the spirit of the many conversations about research ethics that took place over this course, we have chosen to justify and document some of the ethical observations we made over the course of this project. This project was inspired by the Calder and Verwer paper, and we wanted to replicate their results to the best of our ability. Unfortunately, the information that they included on their methodology and the datasets that they tested on was too sparse to replicate exactly. We did know that they used Census data and classified income, so we conducted our experiments in a similar fashion. In the spirit of transparency and replicability in research, we are making our code available on Github, and have included a link to our dataset at the beginning of this paper.

During this course, we discussed the tendency of the scientific community to only publish positive, surprising, or significant results. In an effort to combat that tendency, we are including an explanation of our first (failed) attempt to conduct this experiment. We initially started with a different dataset, the Current Population Survey (CPS)[4]. While this dataset also includes Census data, it has far fewer fields and rows than the ACS dataset (because this survey is conducted monthly).

Over the course of our analysis with the CPS dataset, we found that our models consistently achieved accuracies almost as low as 0.4–worse than a coin flip! We concluded that the dataset was not well-suited for a naïve Bayes classifier for income. We feel comfortable making the switch to ACS because its more robust data better reflects the depth of data that hiring services like HireVue have access to. That being said, we feel obligated to share our first, failed attempt.

Lastly, as were constructing our models, we initially used scikit-learn's `gaussianNB` classifier instead of the `categoricalNB` classifier. The Gaussian classifier is the wrong model for these datasets: it assumes continuous fields instead of categorical fields, which is why we switched to the Categorical classifier instead. However, we found that, curiously, the Gaussian model actually cohered with most of our hypotheses. As proxies were dropped, accuracy decreased almost monotonically, and the nNB and 2NB models exhibited significantly less variance in accuracies across race. We are unsure how to interpret these results. The model was likely overfitting to the data, but it is still surprising that it behaved as expected, while the correct (Categorical) model behaved unexpectedly. There is a possibility that these results are meaningless because of the use of the incorrect model[5].

5 Conclusions

This paper examined the disparate accuracy impacts of several debiasing methods. We compare four models: a basic model using full racial information, the Hirevue model of dropping proxies until 4/5 compliance is achieved, the simple Calders and Verwer decoupled classifier, and a more fine-grained decoupled classifier that trains a separate model for each racial group. Accuracy measures remained surprisingly stable across these models, indicating that all four models perform equally as well. Accuracy counts across races displayed some variation, but no statistically significant amount.

Our results disprove our hypothesis that training separate models for populations with different characteristics may reduce disparate accuracy impacts. Some of our findings do leave room for future work. Specifically, using the full categorization of race (as opposed to the binary 'white' and 'nonwhite' values), The Hirevue model failed to achieve 4/5 compliance. This leaves room for further research on whether proxy dropping is sufficient to achieve 4/5 compliance for different kinds of datasets.

This work has two major ethical implications. First, we examine a kind of fairness that the law does not regulate: disparate accuracy. For this experiment, there was no tradeoff between disparate accuracy and disparate impact, but our hypothesis (which still invites further research) was that such a tradeoff may exist, inviting questions about the structure of the law. Regulating such accuracy impacts is difficult, but one can imagine a 4/5 rule for accuracy: the accuracy of the group with the lowest accuracy must be no more than 4/5 that of the group with the highest accuracy. This suggestion is

[4]Data can be found here: `https://www.census.gov/programs-surveys/cps.html`

[5]Lesser researchers may have engaged in "model hacking" and made the Gaussian model the focus of their paper in order to publish more surprising results, but we have taken AC221 and therefore know all about the ethics of research! We feel that our integrity is evidence of our deep commitment to research ethics and our internalization of the course's research ethics module.

very preliminary, as accuracy measures vary across models and classification tasks, but there is room for further work in this area.

Second, out of the models we examined, only the Hirevue model complies with hiring law. The fact that differential accuracy did not change for the 2NB and nNB models indicates that disparate treatment in the context of algorithmic hiring does not produce a disparate impact. This is an interesting ethical question: if treating different racial populations differently does not adversely impact these populations and may in fact positively impact minority populations, is such disparate treatment wrong? The history of American hiring law is closely related to the history of segregation and civil rights in this country, but machine learning is changing the hiring landscape so rapidly that perhaps we must consider new definitions of fairness and justice.

References

[1] Sorelle A Friedler, Carlos Scheidegger, and Suresh Venkatasubramanian. On the (im) possibility of fairness. *arXiv preprint arXiv:1609.07236*, 2016.

[2] U.S. Congress. Civil rights act. 1964.

[3] Zachary Lipton, Julian McAuley, and Alexandra Chouldechova. Does mitigating ml's impact disparity require treatment disparity? In *Advances in Neural Information Processing Systems*, pages 8125–8135, 2018.

[4] Equal Employment Opportunity Commission. Uniform guidelines on employee selection procedures. *Federal Register*, 1978.

[5] Manish Raghavan, Solon Barocas, Jon Kleinberg, and Karen Levy. Mitigating bias in algorithmic hiring: Evaluating claims and practices. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 469–481, 2020.

[6] Loren Larsen. Train, validate, re-train: How we build HireVue assessments. `https://www.hirevue.com/blog/train-validate-re-train-how-we-build-hirevue-assessments-models`, 2018.

[7] pymetrics inc. audit-AI: How we use it and what it does. `https://github.com/pymetrics/audit-ai/blob/master/examples/implementation_suggestions.md`, 2020.

[8] Verwer S. Calders, T. Three naive bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, 2012.

[9] Cynthia Dwork, Nicole Immorlica, Adam Tauman Kalai, and Max Leiserson. Decoupled classifiers for group-fair and efficient machine learning. In *Conference on Fairness, Accountability and Transparency*, pages 119–133, 2018.

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.