Group Project:
Machine Intelligence and Society

Tomas Bueno dos Santos Momcilovic
Isaac Bravo
Fabiola Schwarz
Haoran Cheng
Zitong Wang

August 31, 2021

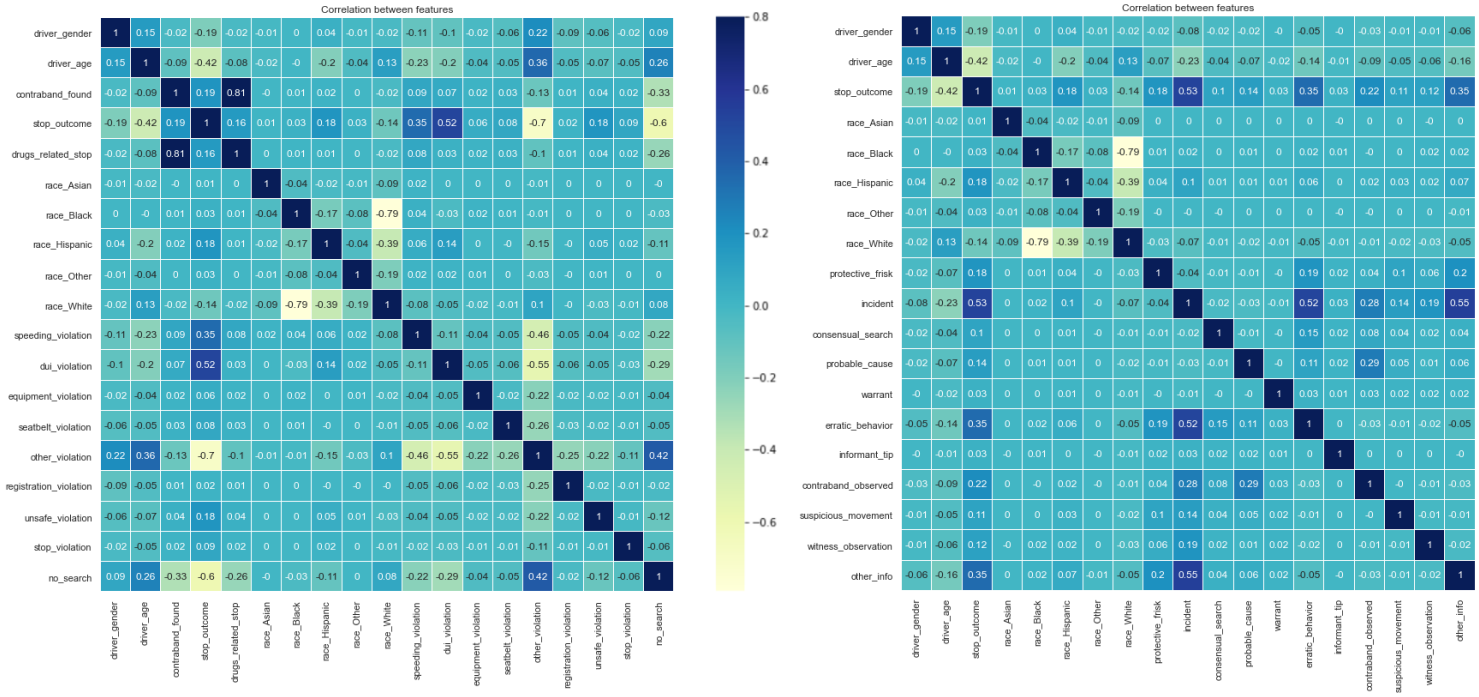
1 Introduction

This project is about basic machine learning (ML) methods and fairness in ML. In the following text, we guide the reader through the process of training, testing, and validating an ML model. The given dataset was a subset of the North Carolina Policing Dataset. It is a product of the Stanford Open Policing Project which collects data from about 21 state patrol agencies about police traffic stops, in order to investigate interactions between citizens and the police. Further information can be retrieved from: https://openpolicing.stanford.edu. The data at hand for this project only encompass the state of North Carolina (NC). It comprises **402,087 observations** with **14 attributes**.

1.1 Data Preprocessing

The dataset needed to be preprocessed before training the models. To prevent errors with further analyses, we dropped the observations with missing values in driver_age from our working dataset. We encoded the binary attributes driver_gender, contraband_found and drugs_related_stop using LabelEncoder(), whereas we used a dictionary for stop_outcome, to control the order in which 'Arrest' (1) and 'No Action' (0) were encoded. Because driver_race is a non-binary categorical attribute, we applied the one-hot encoder to avoid ordering the values in any way. We also used the unique keywords from the search_basis to create new binary-encoded attributes for search reasons in each observation. Finally, we dropped driver_race_raw, district, stop_date, officer_id and state, making sure first that these attributes do not hold any unrecognized value. After pre-processing the dataset, we arrived at **401,996 observations** and **18 attributes**.

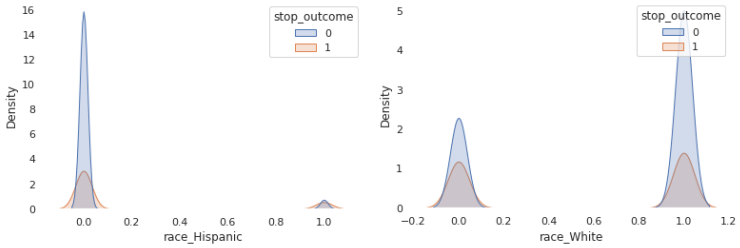
Below, we plot the correlation between attributes in 1a. Since we are classifying the probability of being arrested, we chose the attributes which correlate with the stop_outcome at any significant level. From all races, being White (r = -0.14) or Hispanic (r = +0.18) has a substantive correlation with stop_outcome, while Asian, Black or other races do not (between 0.01 and 0.03). Other relevant attributes include driver_age, driver_gender, erratic_behavior, contraband_observed, contraband_found; drugs_related_stop is also relevant, but has a very high correlation with contraband_found, so we excluded it to avoid issues with multicollinearity. These **7 attributes** were placed into a matrix that helped generate binary classifiers using two different methods, whose performance was later compared against a dummy classifier which always predicts 'No Action'.



(a) Correlation Matrix

2 Training and Evaluating Classifiers

The following table shows the type and the name of the chosen attributes which were placed into the predictor X and outcome y matrices. Although gender and race are both considered to be sensitive attributes, we only used race to test for fairness. The matrices were then divided into training (70% of the sample), validation (9%) and test (21%) sets. As the sample was large, there was no need for cross-validation.

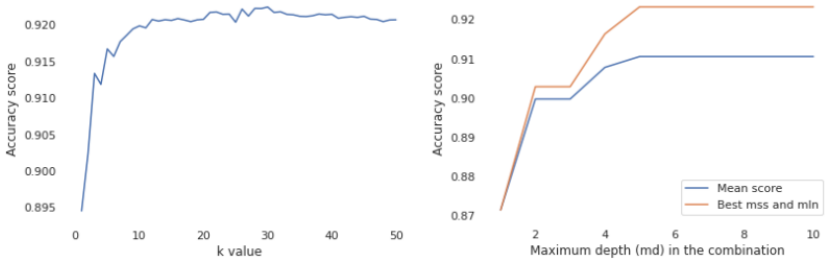


(b) Race across Stop Outcome

Attribute Type	Attribute Name
Non-Sensitive Predictors (X)	driver_age contraband_found erratic_behavior contraband_observed speeding_violation dui_violation other_violation unsafe_violation no_search protective_frisk incident probable_cause
Sensitive Predictors (X)	driver_gender race_Hispanic race_White
Predicted Outcome (y)	stop_outcome

2.1 Classifiers: kNN and Decision tree

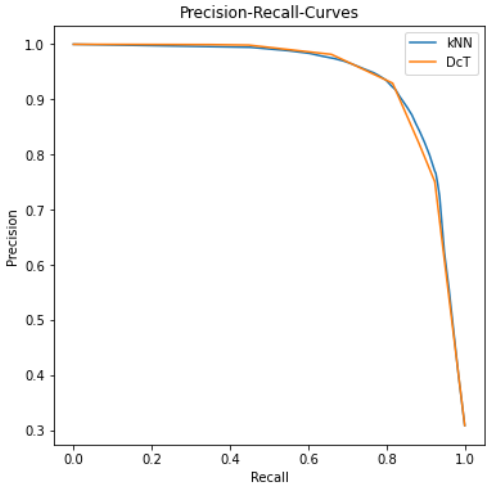
Two binary classifiers were chosen for this project: the k-Nearest-Neighbor (kNN) classifier and the Decision-Tree (DcT) classifier. We determined the hyperparameters for both models* by methodically testing their accuracy when used on the validation set. The results are represented in 1c, where we determined the best k = 30, and the best combination of md-mss-mln = 5-2-8†. On this basis, we trained both models and calculated their Recall, Precision and F1-score.



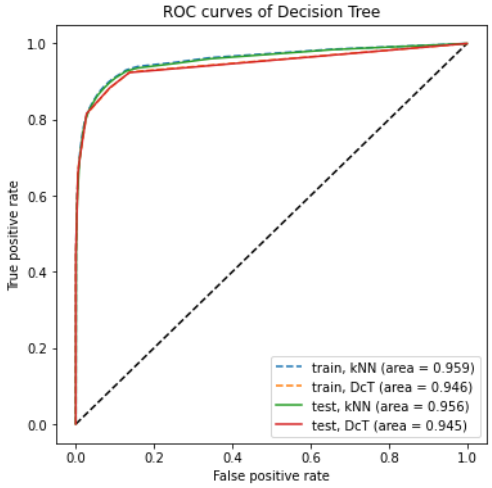
Metric/Score	kNN	DCT
True Positive:	30,647	30,348
True Negative:	80,624	81,093
False Positive:	2,770	2,301
False Negative:	6,558	6,857
Recall:	0.82	0.82
Precision:	0.92	0.93
F1:	0.87	0.87

(c) Hyperparameter selection

The following precision-recall curves in 1d show the trade-off between precision (low false positive rate) and recall (low false negative rate) of the kNN model and the DcT model. The F1 scores indicate that the optimum (harmonic mean of recall and precision) for both models is at the 0.87 threshold. The Receiver Operating Characteristics (ROC) curve in 1e also helps compare the models' performance, where the false and true positive rates map onto the x and y axes, respectively. A dummy classifier that always predicts 'No Action' has an Area Under the Curve (AUC) measure of 0.5, as indicated by the dotted line. Taking all these measures into account, we can see that both models can distinguish between positive and negative classes, and that the differences are marginal.



(d) Precision-Recall Curves



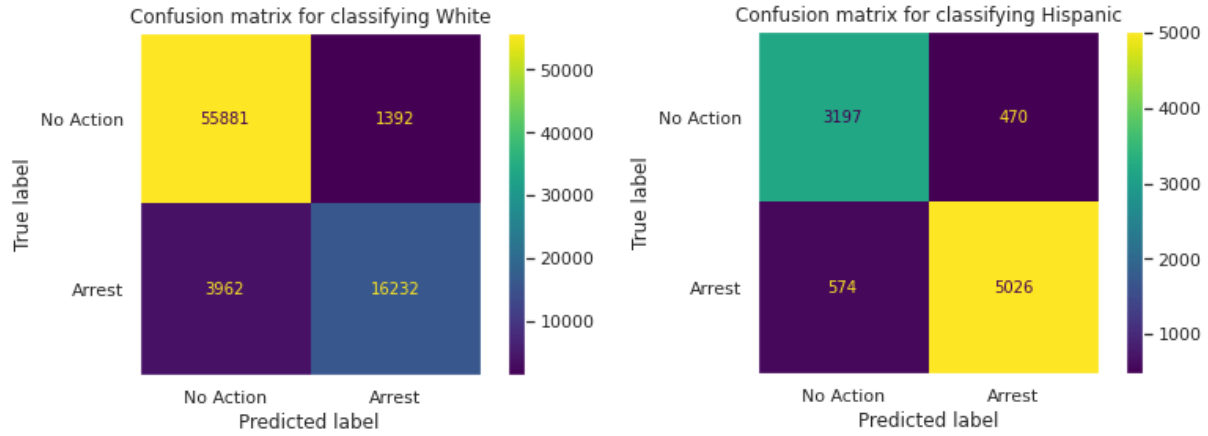
(e) Receiver Operating Curves

*For kNN the project-relevant hyperparameter is k (the number of nearest neighbors). For DcT, the relevant hyperparameters are md (minimum depth of tree), mss (minimum sample split) and mln (maximum leaf nodes).

†Best lowest; the md provided the largest changes, while mss and mln provided marginal changes

2.2 Independence, Separation, and Sufficiency

Due to the significant, yet opposite correlations of race_White and race_Hispanic with stop_outcome, we chose those two races to test our kNN model for fairness. Testing for fairness involved calculating the rates of correctly (true positive and negative) and incorrectly (false positive and negative) predicted outcomes for each race, and assessing the difference between these rates in measures of fairness determined by Barocas, Hardt & Narayanan (2019)[‡]: independence, separation and sufficiency.



(f) Confusion matrices for White and Hispanic drivers

kNN Model	Independence	1.Separation	2.Separation	1.Sufficiency	2.Sufficiency
Formula:	(tp+fp)/(all)	tp/(tp+fp)	tn/(tn+fn)	tp/(tp+fn)	fp/(fp+tn)
White:	0.23	0.92	0.93	0.80	0.12
Hispanic:	0.59	0.91	0.85	0.90	0.13
Difference:	-0.37	0.01	0.09	-0.09	-0.10

In a perfectly fair model, the race-based difference between all measures would have been zero. As can be observed in the table above, this is not the case. Most notably, the predictions of our kNN classifier are **not independent** from the sensitive characteristics of race. One reason is that the model predicts a higher percentage of false positives - i.e. predicted 'Arrest' when in reality there would have been 'No Action' - in the subset of Hispanic (13%) than in the subset of White (2%) drivers in the sample. Another is that there is a larger number of arrests in the Hispanic subset (54%), compared to the White subset (21%). While the issue with the bias in the dataset cannot be fixed as easily, potential ways to fix the bias in the model are explored below.

We have assessed the fairness of the DcT model separately in the code. However, due to marginal differences between kNN and DcT performance, and different trade-offs when comparing the measures of fairness, we have decided to continue with kNN only. Please see the code for this evaluation of DcT.

[‡]Barocas, S., Hardt, M. & Narayanan, A. (2019). Fairness and Machine Learning [Pre-Published Version]. Retrieved on 20th August 2021 from https://fairmlbook.org

3 Modelling Fairness

To test whether we can generate a fairer model, we can try to fix its bias by 1) excluding sensitive variables from the training of the model, and 2) choosing different thresholds in determining false and true positives for each race.

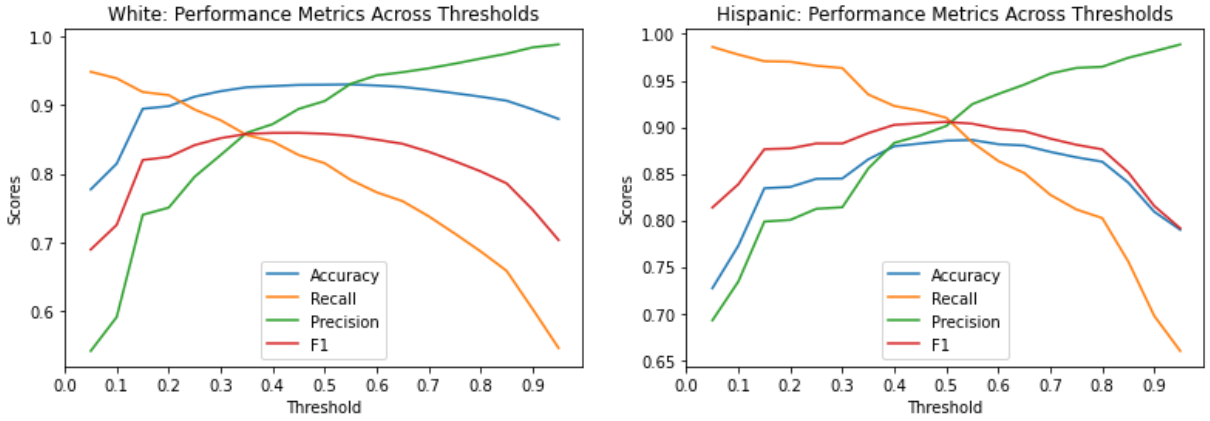
3.1 With and Without Sensitive Attributes

By excluding the sensitive attributes, we have trained our new model *kNN˙new* on a matrix that does not contain race_White nor race_Hispanic, with very similar results in performance. In the following table, we used the results of the old and new confusion matrices[§] Comparing the differences in both models, we can say that the kNN model trained without sensitive characteristics is slightly fairer according to independence and sufficiency, but involves a proportional trade-off in separation.

kNN_new Model	Independence	1.Separation	2.Separation	1.Sufficiency	2.Sufficiency
White:	0.24	0.91	0.94	0.82	0.03
Hispanic:	0.56	0.94	0.82	0.87	0.09
Difference:	-0.33	-0.03	0.12	-0.05	-0.6
kNN Diff.	-0.37	0.01	0.09	-0.09	-0.10

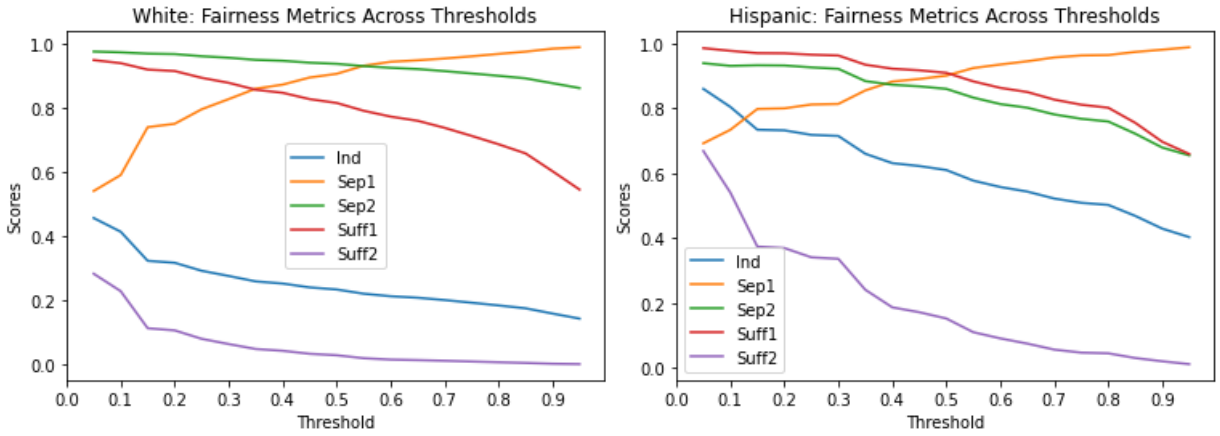
3.2 Aligning Fairness with Thresholds

Models prior to this one used the predictors to estimate the probability of a driver being arrested, and then decided on which outcome will be predicted based on a threshold of 0.5 probability. To account for bias in the dataset and the model, we can adjust the thresholds at which our kNN model predicts the outcome for the White and Hispanic race separately. To do so systematically, we have mapped difference metrics of performance and fairness on each 0.05 level between the 0.05 and 0.95 thresholds. The results are displayed in the graphs 1g and 1h below.



(g) Performance across thresholds for White and Hispanic subsets

[§]Because the difference in performance between the old and the new model is marginal, we do not include the new confusion matrices here. Please consult the code for the graphical representation of values.



(h) Fairness across thresholds for White and Hispanic

While the performance metrics generally converge on one threshold for each race (0.35 for White, 0.5 for Hispanic), the fairness metrics feature a trade-off, not only between races, but also within. For example, a higher score in independence is mutually exclusive with a higher score in first separation; similarly, a change in threshold leads to marginal change in one metric, but a significant one in another.

Because of this issue, we have decided on a heuristic that helped us choose a threshold for each race systematically. First, for each fairness metric and each threshold, we subtracted the value in Hispanic from the value in White, in order to get a metric-relevant difference from every possible threshold combination. Second, we looked for the smallest possible difference in each metric[¶]. Finally, we took the mean of those differences across all metrics, and decided on a threshold value of 0.49 for White and 0.71 for Hispanic. We show the results of this new kNN model that thresholded the outcomes separately for the two races; performance was once again not affected to a large extent. The limitation of this heuristic is that it emphasizes balance between the metrics, but we did not have a good reason to weigh either metric as more relevant.

kNN_new Model	Independence	1.Separation	2.Separation	1.Sufficiency	2.Sufficiency
White:	0.23	0.91	0.94	0.82	0.03
Hispanic:	0.52	0.96	0.78	0.83	0.06
Difference:	-0.29	-0.05	0.16	-0.01	-0.03
kNN Diff.	-0.37	0.01	0.09	-0.09	-0.10

3.3 Conclusion

The dataset on arrests of drivers in North Carolina is imbalanced, as it features proportionally more arrests of Hispanic drivers, and less of White drivers. The kNN and DcT models that stem from this dataset contain this bias, and as we have tested by excluding race altogether or trying out different decision thresholds and comparing fairness metrics, they cannot be debiased to a large extent. While our approach partially succeeded, to properly train the model, we would need to validate our model across different states and countries. Data-level, not model-level, interventions seem to be the way forward.

[¶]Negative and positive differences were compared by taking the absolute of the difference.