Business Conclusions

Summary

We begin the analysis by saying that there were no middle eastern car riders observations in the production run and as such we will exclude this class from the analysis.
We can see the relevant metrics in the following table:

Metric	Before production (Report 1)	In production
Global precision	0.59	0.58
Global recall	0.71	0.66
Male precision	0.59	0.58
Female precision	0.58	0.58
Hispanic precision	0.57	0.55
Not applicable ethnicity precision	0.60	0.59
White precision	0.61	0.60
Black precision	0.55	0.55
Asian precision	0.61	0.50
Indian precision	0.55	0.50

Discrepancy in SexCode feature	1%	1% (male and female precisions were rounded)
Discrepancy in EthnicityCode feature (disregarding middle eastern class)	3%	5%
Discrepancy in RaceCode feature	7%	10% (unrounded: 9,94%)

The first client requirement was achieved since the global precision was 0.58, higher than the minimum value of 0.5.
The second reformulated client requirement was also achieved, discrepancy between the protected classes was kept below 10%.
Finally, the third requirement of maximizing recall constrained to the previous objectives was also achieved, with a recall of 0.66 (approximately 2 out of 3 cars with contraband were caught).
Concluding, all requirements were achieved, but the second reformulated requirement was achieved by a very thin margin. A detailed analysis of why this happened is presented in sub-section “Fairness”.
We consider the project to have been a huge success and would welcome any further opportunity to work with the police department of Cincinnati.

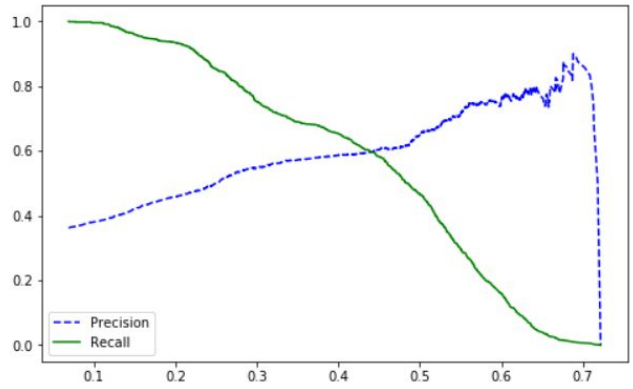
Results Analysis

Model Performance

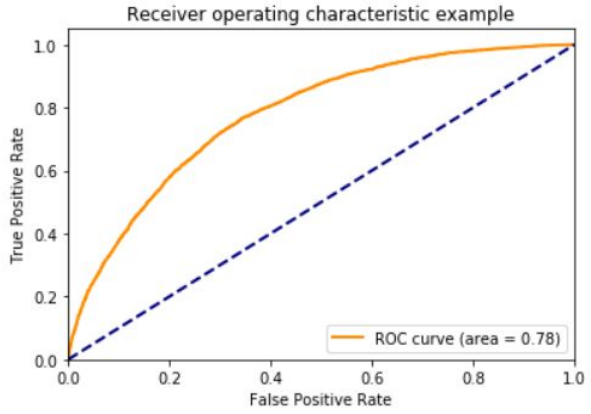
In the following table we can see the relevant metrics:

Metric	Before production (Report 1)	After production	Decrease in production
Global precision	0.59	0.58	0.01
Global recall	0.71	0.66	0.05

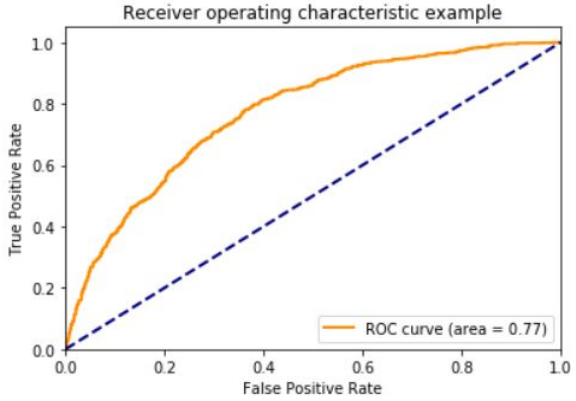
These relatively small decreases of precision and recall in production prove that our machine learning model was not overfitting the training data.
The 0.58 precision has a 0.08 buffer that reliably guarantees that the precision would be above 0.5 when the number of observations further increases.
Due to the precision recall tradeoff, we believe that the value of 0.66 recall is a very good one.
We can also compare the production precision and recall curve vs threshold with the one obtained before production:



This is a very similar curve to the one obtained before production, with a small difference: this curve's precision goes to 0 as the threshold approaches its maximum value. This has to do with the fact that, in production, the maximum value of the probability of having contraband is exactly equal to the maximum value of the threshold, while in the curve obtained before production, that same probability was a little bit higher than the maximum value of the threshold. Thus, in production, for the last threshold the model does not search any vehicle and precision is 0.
Although not shown in the previous report, it is also interesting to analyze the receiver operating characteristic before and in production.
This is the curve before production:



And this is the curve in production:



We can, again, see that the curves are similar. In these curves we can also compare another relevant indicator, the roc auc. We can see that there was a decrease of 0.01 in the roc auc, inline with the decreases of precision and recall.

Overall our model performance was completely inline with what was expected and reported in the first report.

Fairness

Let us look at fairness metrics before and after production:

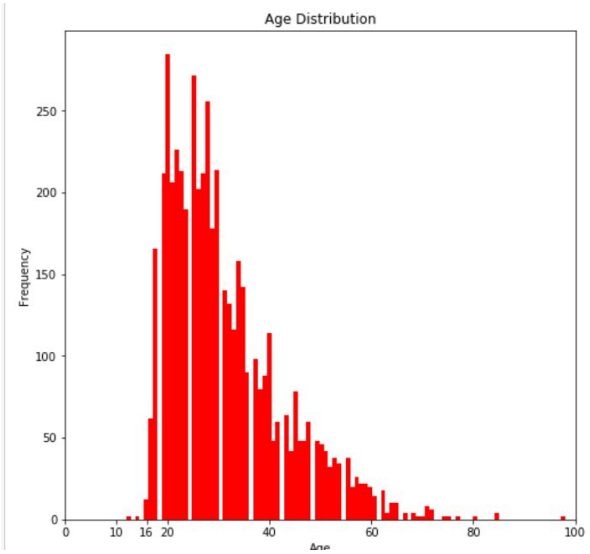
Metric	Before production (Report 1)	After production	Decrease in production
Male precision	0.59	0.58	0.01
Female precision	0.58	0.58	0
Hispanic precision	0.57	0.55	0.02
Not applicable ethnicity precision	0.60	0.59	0.01
White precision	0.61	0.60	0.01
Black precision	0.55	0.55	0
Asian precision	0.61	0.50	0.11
Indian precision	0.55	0.50	0.05
Number of Asian observations	44419	46	N. A.
Number of Indian observations	17623	10	N. A.
Discrepancy in sex class	0.01	0	0.01
Discrepancy in ethnicity class	0.03	0.04	-0.01(increased)
Discrepancy in race class	0.06	0.1 (0.05 excluding asian and indian classes)	-0.04 (increased)

All the precision values after production were very similar to the ones calculated before production. This is yet another sign that our machine learning model was not overfitting the training data.
However, both the asian and the indian classes had a noticeable decrease in precision which resulted in a race class discrepancy very near the limit of 10 percentage points.

If we look at the number of observations for these classes, we see that in the final dataset there were 46 asian observations and 10 indian observations (for comparison purposes, these values were 460 and 191 in the dataset before production, respectively).
These amounts are so small that, e.g., if our model would have searched and missed an additional indian observation, the precision would drop from 0.5 to 0.4.
Thus, even though the race class discrepancy was kept lower than the objective of 10 percentage points, we cannot reliably say that this would hold true for small increases in the number of observations. However we would expect the discrepancy to approach 7% (the value that we got before production) as the number of observations increases.
Thus, it does not make sense to evaluate the discrepancy of precision in the race feature with these two classes.
The race class discrepancy excluding asian and indian observations is 5 percentage points. In this situation, every protected class had a discrepancy equal or lower than 5 percentage points.
Overall our fairness performance was inline with what was expected and reported in the first report, except for the Asian and Indian precisions, for which we did not foresee the problem related to the very low number of observations of these classes.

Population Analysis

Let us evaluate the population feature by feature:
Every department in the production data was also present in the training data. However, 12 departments present in the training data were not present in the production data.
Regarding the intervention location feature, there was one location in the production data that was not present in the training data ('YALESVILLE') and in reverse, 524 locations of the training data were not present in the production data.
Regarding the InterventionDateTime feature, the training data was from 2014 to 2017 and the production data was from May to July 2018. It was expected that the production data would be more recent than the training data.
Regarding the ReportingOfficerIdentificationID feature, there were 86 new officers in the production data. Also, 4070 officers of the training data were not present in the production data.
Regarding InterventionReasonCode values and frequency order of the values (from most frequent to least frequent: V, E, I) stayed the same.
The ResidentIndicator feature stayed a boolean feature with only True or False values. Both datasets had the same frequency order.
The SearchAuthorizationCode feature had the same values and the same frequency order.
Regarding the StatuteReason feature, every class present in the production data was also present in the training data. The reverse was not true, 4 classes of the training data were not present in the production data (including null values). The frequency of the classes changed in a relevant way (e.g. the least frequent class in training "Stop Sign" with 3 observations was relatively much more frequent in the production data, with 230 observations).
Regarding age, let us look at the distribution:



The distribution is similar to the one in the first report. However, there are some differences in the values, with 87 unique age values in the training data and only 64 in the production data.

As explained in the “Summary” sub-section, the ethnicity code feature in the production data didn’t have middle eastern observations. The frequency order was the same, more non identified ethnicity observations than hispanic observations.

The race code, the sex code and the TownResidentIndicator features had the same classes with the same frequency orders in both datasets.

Overall, many features had values in the training data that were not present in the production data. This was expected because of the big difference in the number of observations (69879 vs 5000).

The InterventionDateTime, InterventionLocationName and the ReportingOfficerIdentificationID were the only features that had values in the production data that were not present in the training data. In the case of the InterventionDateTime, this was expected.

Let us also perform a duplicates analysis: out of our 5000 observations, 2510 are duplicates. If we extend this analysis to all observations (the ones for which we got the true_class and the ones for which we do not), we can see that out of the total 10000 observations, 5020 are duplicates.

Next steps

Next steps

There are some technicalities that could improve the model: a more rigorous data cleaning process, where we would iterate back and forth with the police department to understand what are the valid values and what are data imputation problems and a proper hypertuning parameter optimization.

Another way of going forward would be to explore other ways to guarantee fairness. In the model we relied on what is technically known as post-processing (find the threshold that minimizes discrepancies in precision of the protected classes). Other ways, like

pre-processing (create a new representation of the data that removes the information correlated to the protected classes and preserves the other information as much as possible) and optimization at training time (add a constraint or a regularization term to the existing optimization objective) could further improve the fairness metrics.
Regarding the way to move forward, we believe that the police department has to do a reflection about our results. This reflection must cover at least the following two questions:

Is the current definition of fairness (difference between the precision of the protected classes) the most appropriate one for this particular case? For example, is it fair for the classes to have a big recall difference?

Is the maximum difference of 10 percentage points in precision good enough? If it isn't, should the police department focuses on improving the current model or should it create a new dataset that isn't so biased, sensibilizing the police officers for the problems of the current one?

Our consultancy would be available to work with you on these issues.

Deployment Issues

Redeployment

We did not redeployed the model. To understand why, let us look at the data:

Metric	21 Jan.	22 Jan.	23 Jan.	25 Jan. (Final)
Global precision	0.58	0.57	0.58	0.58
Global recall	0.64	0.64	0.66	0.66
Male precision	0.61	0.52	0.58	0.58
Female precision	0.58	0.58	0.58	0.58
Hispanic precision	0.51	0.52	0.55	0.55
Not applicable ethnicity precision	0.61	0.58	0.59	0.59
White precision	0.58	0.56	0.60	0.60
Black precision	0.60	0.59	0.55	0.55
Asian precision	0	0.4	0.5	0.50
Indian precision	0	0	0.5	0.50

The global precision stayed above 0.50 for the whole duration of the production run.
The discrepancy in the sex class was also lower then 10 percentage points for the whole duration of the production run.
The discrepancy in the ethnicity class was 10 percentage points for the first day. In this situation, we decided that we would redeploy if the discrepancy grew in the second day. However, the discrepancy decreased in that day and managed to stay under 10 percentage points for the rest of the production run.
The discrepancy in the race class, considering only the white and black observations, was lower than 10 percentage points for the whole duration of the production run. However, if we include the asian and indian observations, the discrepancy was higher than 10 percentage points in the first two days and equal (without roundings it is actually a little bit lower) in the last two days.
Nonetheless, we never thought of redeploying because of this fact. Please look at the section “Fairness” for a detailed explanation of why we disregarded Asian and Indian observations in the assessment of the race code class discrepancy.
However, we should have foreseen this issue and made it clear to the client that the reformulated second requirement would only apply if the classes were representative of the dataset.

Unexpected problems

There were some unexpected problems when we were trying to deploy the model. Mainly they were related to the fact that our code was in a different directory that the one that was being deployed to heroku. We solved that by deleting the heroku app and deploying everything from the same directory (single source of truth).
During the production run itself there were no unexpected crashes, everything went smoothly and we were able to analyse all the 10000 rows of data.
Regarding unexpected data, as can be seen in more detail in the section “Population Analysis”, the population was representative of the data that we had before production (a fact that confirms this is the very small decreases in production of our performance metrics). However, we were not expecting the high number of duplicates. Even though they do not affect the metrics (since they are basically percentages and thus are not affected by duplicates), we expected the data to be free of duplicates.
Other problem that we didn’t expect was the difficulty to parse the data from the DB into a pandas dataframe. This was due to the fact that we saved the json string into the DB and that made it difficult to reconstruct it into a dataframe.
Finally, as further elaborated in the sub-section “Fairness”, we were not expected (although we should have) that the Indian and Asian classes had so few observations that it was unreasonable to calculate the discrepancies of the race feature with these two classes.

What would you do differently next time

If we could repeat the construction of the model, here is what we would have done differently:

Align the expectations of the client: with 460 asian and 191 indian observations in the training set and knowing that our production set would have only 5000 labeled observations, we should have foreseen that including these classes precisions in the calculation of the race discrepancy would be problematic. We should have explained this to the client and calculate the race discrepancy as just the difference between the white and the black drivers precisions.

Regarding the difficulty of accessing the json observation saved in the database, we should have saved each individual field in the database instead of the whole observation. This would avoid having to parse this information to transform it into a pandas dataframe.

We realized that the model use case only made sense it the model was trained just with observations of cars that were searched very late into development. This implied that some late stage optimizations (like hyperparameter tuning) were not done. Having another chance, we would rectify that.