Business Conclusions

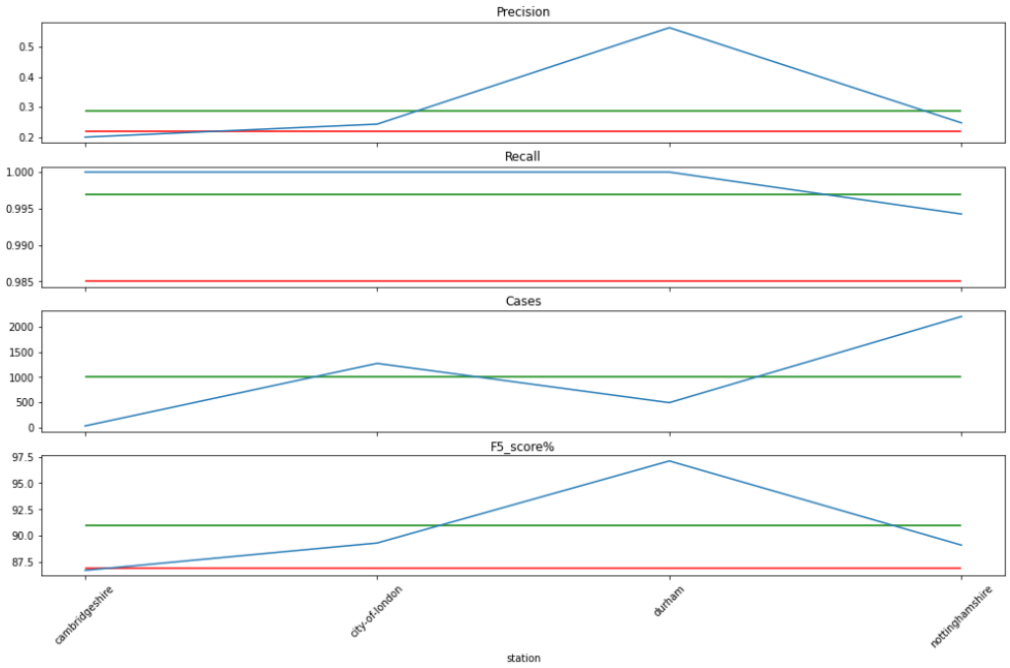
Summary

There were two main requirements for our model: to have a high discovery rate (Recall) while maximizing an overall ability to detect offences (Precision). In order to analyse these two parameters simultaneously, the F5 score was used. We used 80% of initial data (train set) to train our model and 20% to evaluate (validation set) its initial performance:

- Precision: 0.22
- Recall: 0.985
- F5: 86.9%

After having our API fully functional, 4000 requests were made with new data. API requests ran smoothly and we got the following performance (averages from val/test sets in red/green):

- Precision: 0.286
- Recall: 0.997
- F5: 91.0%



Despite these positive results, the model under-performed in ‘cambridgeshire’ probably due to the low number of cases (only 31).

Regarding performance between stations, we had a Department mean rate of 20.4% success searches with 31% stations (12 out of 39) below the mean margin (18.36%) in our validation set, not counting the 3 stations with an average of 0. In the test set, we got a mean successful search rate of 28.6% across 4 stations and none of them had an average below initial mean

margin of 18.36%. Thus, we can conclude that our model is not discriminating any specific station in the test set.

Result Analysis

Model Performance

Initially, the performance of our model was evaluated in the validation set (VAL SET) to assess the percentage of wrongdoer suspects we were able to correctly identify (Recall) and the overall ability to detect offenses (Precision). Additionally, F5_score (F_β with **β**=5 to increase the recall's relative contribution to precision, thus minimizing the number of suspects we miss) was used to choose and tune the model. The new data (TEST SET) got good results with our model:

station	Precision	Recall	cases	F5_score
cambridgeshire	0.200	1.000	31	86.7
nottinghamshire	0.247	0.994	2203	89.1
city-of-london	0.243	1.000	1272	89.3
durham	0.564	1.000	493	97.1

station	Precision	Recall	cases	F5_score
mean VAL SET	0.220	0.985	6574	86.9
mean TEST SET	0.286	0.997	1000	91.0

$$F_\beta = \frac{(1+\beta^2)\cdot(\text{precision}\cdot\text{recall})}{(\beta^2\cdot\text{precision}+\text{recall})}$$

The performance of 'cambridgeshire' data turned out to be worse than our mean VAL SET values. However, since we only have 31 cases, it is recommended to continue collecting data and then do another analysis before changing anything in the model and/or making a decision regarding this station.

When looking at the criteria between ethnicities we found a balanced distribution, which indicates that generally our model does not discriminate any ethnicity.

ethnicity	Precision VAL	Precision TEST	Recall VAL	Recall TEST	cases VAL	cases TEST	F5_score VAL	F5_score TEST	F5_progress
Black	0.216	0.240	0.985	0.994	4345	654	86.7	88.6	BETTER
White	0.220	0.321	0.984	0.997	36977	2528	86.8	92.3	BETTER
mean	0.220	0.286	0.985	0.997	9204	800	86.9	91.0	BETTER
Asian	0.219	0.210	0.987	1.000	3237	546	87.0	87.4	BETTER
Mixed	0.218	0.212	1.000	1.000	374	176	87.9	87.5	WORST
Other	0.234	0.239	1.000	1.000	1086	95	88.8	89.1	BETTER

Since the TEST data is more recent, F5_progress generally indicates that the performance of the Department has improved over time, possibly due to some change in the conduct of agents. Only 'Other' got a worst F5 score when compared to the validation set. However, this difference is less than 0.5% which represents little relevance.

Success on requirements

Similarly as we made on Report 1, we created sub-groups for each combination of ´Gender' + 'station' + 'Officer-defined ethnicity' with more than 30 occurrences. This approach reduced our test set from 28 to 12 sub-groups representing 96.3% of searched cases (model predicted True).

station	Gender	Officer-defined ethnicity	cases	success	success_rate
durham	Female	White	52	31	59.6
	Male	White	417	231	55.4
city-of-london	Female	White	79	25	31.6
nottinghamshire	Male	White	1197	326	27.2
city-of-london	Male	Black	306	81	26.5
		White	549	135	24.6
		Other	58	14	24.1
nottinghamshire	Male	Black	303	68	22.4
		Asian	270	58	21.5
		Mixed	150	32	21.3
city-of-london	Male	Asian	237	49	20.7
nottinghamshire	Female	White	124	25	20.2

Results from KPI-1 show that in 'nottinghamshire' all sub-groups (5 out of 5) follow outside the +/- 5% margin from station mean search rate (24.9%). This behaviour is identical to that observer in the validation set (we had 8/8 sub-groups outside mean margin). Search rate in 'city-of-london' dropped from 27.9% to 24.7% and in 'durham' it improved from 35.8% to 55.9%.

Test set mean search rate by sub-group is 28.7% and all 3 stations follow outside +/- 10% margin (cambridgeshire do not have any sub-group with more than 30 cases). Despite these results, it should be noted that the search rate at all 3 stations is above the average observed in Report 1 (20.4%).

KPI-1 (station indicator)

- Evaluates discrimination using a success rate criteria per sub-group in each station
- 'True' means that the sub-group fails the discrimination criteria (is being discriminated)
- '100%' means that all sub-groups fail the criteria

KPI-2 (global indicator)

- Evaluates discrimination using a success rate criteria per station
- 'True' means that the station fails the discrimination criteria (is being discriminated)

station	sub-groups	mean search rate %	cases	discrimination% KPI-1
nottinghamshire	5	24.9	2044	100.0
city-of-london	5	24.7	1229	60.0
durham	2	55.9	469	50.0

station	station rate	discrimination KPI-2
city-of-london	24.7	True
nottinghamshire	24.9	True
durham	55.9	True

Population Analysis

After cleaning the initial dataset, we got 230094 cases to train our model (train_val). Our API stored information from 3999 requests (test). These dataframes have the following characteristics:

train_val	missing	uniques	missing %	uniques %
Type	0	3	0.0	0.0
Date	0	165481	0.0	71.9
Part of a policing operation	0	2	0.0	0.0
Latitude	0	65324	0.0	28.4
Longitude	0	65656	0.0	28.5
Gender	0	3	0.0	0.0
Age range	0	5	0.0	0.0
Officer-defined ethnicity	0	5	0.0	0.0
Object of search	0	16	0.0	0.0
station	0	35	0.0	0.0
outcome	0	2	0.0	0.0

test	missing	uniques	missing %	uniques %
Type	0	2	0.0	0.1
Date	0	3450	0.0	86.3
Part of a policing operation	1796	3	44.9	0.1
Latitude	2526	656	63.2	16.4
Longitude	2526	658	63.2	16.5
Gender	0	3	0.0	0.1
Age range	0	5	0.0	0.1
Officer-defined ethnicity	0	5	0.0	0.1
Object of search	0	9	0.0	0.2
station	0	4	0.0	0.1
outcome	0	2	0.0	0.1

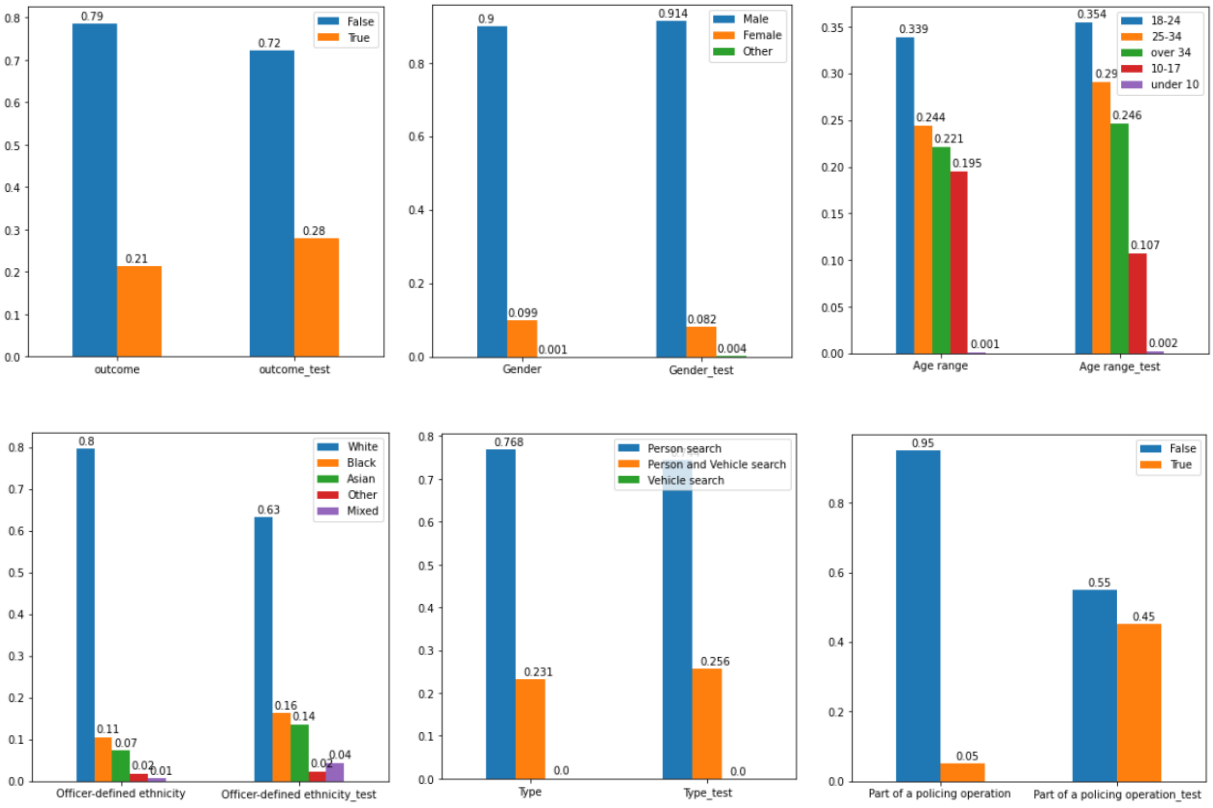
It is observed that there are some categories missing in some features. It is to be noted that missing value also counts as a category, this is the reason why we have 3 uniques in ‘Part of policing operation’. Out test set have the following missing / extra categories (station list is too large to show):

	cat_missing	cat_extra
Type	1	0
Gender	0	0
Age range	0	0
Officer-defined ethnicity	0	0
Object of search	7	0
station	31	0
outcome	0	0

```
{'Object of search': ['Anything to threaten or harm anyone',
                      'Psychoactive substances',
                      'Detailed object of search unavailable',
                      'Goods on which duty has not been paid etc.',
                      'Evidence of wildlife offences',
                      'Crossbows',
                      'Seals or hunting equipment'],
 'Type': ['Vehicle search'],
```

The remaining features have practically identical distributions, with the exceptions:

- ‘Part of a policing operation’: more True cases (from 5% to 45%)
- ‘outcome’: more cases True (from 21% to 28%)
- ‘Officer-defined ethnicity’: all categories changed mainly ‘Mixed’ which has more cases than ‘Other’



These changes in the category distribution may have an impact on the results obtained.

Next Steps

Next Steps

We have some ideas on how we can complement the work presented here.

From a business point of view, we can think of creating a platform to share real-time statistics and insights with each community (station / city) as an awareness campaign and to clean the Department's public image.

On the other hand, technically we can think of some improvements:

- Implement random searches (when API predictions is lower than 10%) to collect more data and, in a way, avoid bias
- Collect more information to be used in future models, such as: meteorology, existence of a large event within a 10Km radius, amount of people in the car, etc.
- Define performance criteria to re-train API's model; i.e. successful search rate is very far from the initial expectation or simply because 1 year has passed since the last model was developed
- Define a clear objective function with presented KPIs (or others) and use other models (deep learning, AI, etc.) to improve performance, at the expense of model explainability

We can also think about implementing other tasks in our API, namely validating requests for suspects to remove clothes.

Another project that we can also develop in this area would be to study which zones and schedules have the majority of criminal activity in order to support and optimize the definition of the routes and schedules of patrol cars.

Deployment Issues

Re-deployment

After receiving the requests from the test set, we went back to training our model. The performance of the new model was very similar to our previous model so it was not necessary to update the model available in the API.

Unexpected problems

No production problems were detected. Since heroku does not save all logs (current version stores only the most recent 1500 lines of log history), it is possible that there was a problem that was not observed (see other options). This behavior can be a consequence of the decision implemented in our API, where all fields have a default value in case there is some data missing or incorrectly entered.

What would you do differently next time

We believe that managing customer expectations is essential in every project. Thus, we always seek to improve communication with all parties involved in our projects.

In this specific project, we would like to have had more opportunities to present and discuss the various decisions that were made during the development phase.

Various phases of the project allowed us to better understand the United Kingdom Department of Police reality. The collection of potentially wrong data and handling missing data are unwanted tasks but represent real difficulties.

We were able to quantify the several accusations that the Department was subject to, thus contributing to an improvement of its public image.

A model of probabilistic prediction was developed to improve the stop and search policy for every police station. We considered the issues of discrimination, bias and equality.

An API was created using the heroku service to validate the Department's stop and search requests. The collection of this data will allow us to improve our model in the future.

Above all, it was a great pleasure to work with the United Kingdom Department of Police and to be able to demonstrate the potential of Data Sciences in day-to-day tasks. We have tried to provide documentation that helps in making more informed and duly documented decisions in the future.

It was a great pleasure for us as Awkward Problem Solutions to work with you again. Greeting and see you next time.