LISBON DATA SCIENCE ACADEMY

Reducing Bias on Stop-and-search Operations

Analysis Report

Prepared for:
United Kingdom Department of Police

Prepared by:
João Sá
Data Science Specialist
DS Team @ Awkward Problem Solutions

28th of May 2023

Table Of Contents

1. Client requirements

1.1 Summary

The United Kingdom Department of Police (UKDP) has been dealing with accusations of racial, gender, and age discrimination in its stop-and-search operations. The UKDP is composed of several stations that are distributed geographically. Patrolling police officers are supposed to stop and search individuals or vehicles based on probable cause. The officer will usually state the object of the search and the legislation that supports such object before interpellating the vehicle or individual. Identifying probable cause can be subjective, and affected by pre-existing bias. As such, the UKDP has requested our assistance on two different fronts.

First, we need to provide an unbiased analysis of the data that they have gathered for past stop-and-search situations and identify whether there is any evidence of discrimination against the protected classes - ethnicity, and gender. They would like this analysis to be broken down by department. The UKDP would also like to understand whether there is any bias on requests to remove more than outer clothing against the following protected classes - age, ethnicity, and gender.

The second request involves creating a system, in the form of a resting Application Programming Interface (API) that approves stop-and-search operations based on the patrolling officer’s input. This system is aimed at leveling the success rate of stop-and-search activities across protected classes and search objectives, providing an extra layer of confidence against discrimination. Furthermore, the leveling of discovery across protected classes should not diminish the ability to detect offenses significantly.

1.2 Requirements clarifications

As established in the above section we have two distinct requests by the client, each with its set of requirements, but the main concepts to establish these requirements are - stop-and-search success, precision, and recall.

- **stop-and-search success** - we consider the stop-and-search to be a success when the outcome is different than ‘A no further action disposal’ and when it is linked to the object of the search. This means that if the officer decides to stop and search an individual it should result in some sort of legal action and that action should be linked to the reason why the individual was stopped and searched. Searching an individual for suspected possession of firearms and taking legal action because of possession of drugs is not considered a success for our purposes.
- **precision** - in Machine Learning, precision is defined as the fraction of correct predictions out of all predictions of success [from Wikipedia]. For our purposes, precision is the fraction of stop-and-search successes as established above, out of all the instances where the model predicted that the search would be a success. As an example, if the model suggests a search for 100 (number of True Positive plus False Positive) cases but it was only correct in 60 (number of True Positive) of those, the precision of this model is 0.6 - or 60%.

- **recall** - similarly, recall is defined as the fraction of correct predictions from all possible successes [from Wikipedia]. For our purposes, recall is the fraction of correct predictions out of all stop-and-search successes in the data. As an example, if there were 100 stop-and-search successes (number of True Positive plus False Negative) cases but the model only requested a search for 60 cases correctly (number of True Positive cases), the recall of this model is 0.6 - or 60%.

The first request is to provide evidence of discrimination against the protected classes, across stations and search objectives, plus evidence of excessive requests to remove more than outer clothing. The technical requirement for this request is defined as such:

- We will consider proof of discrimination when the rate of success between the protected class with the highest rate and the one with the lowest rate is higher than 5 percentage points. If the success rate of search for Female individuals is 5% and the success rate for Male individuals is 15%, the difference between these is 10% which indicates there could be some bias that leads to more unwarranted searches of Female individuals. This difference should be level across all stations and objects of search.
- We will also consider that the difference between the rate of requests to remove outer clothing across all protected classes should not be higher than 5 percentage points. This difference should also stay level across all stations and objects of search.

The second request is to provide a system that approves stop-and-search operations.

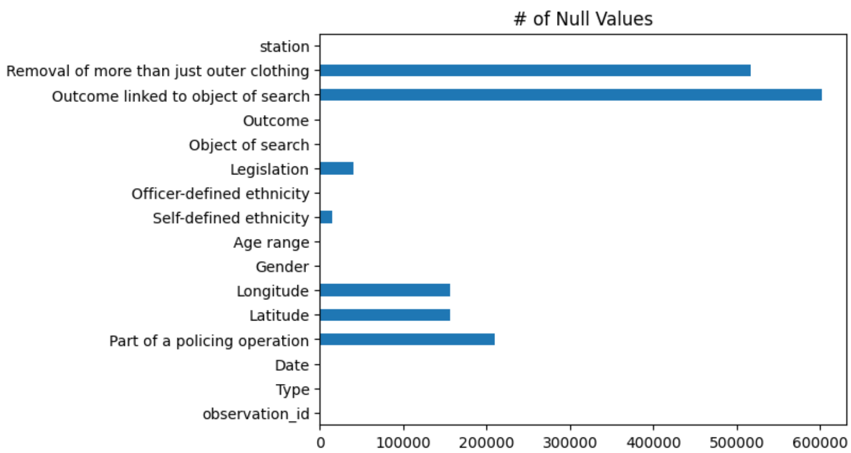
- We will consider that the system should approve stop-and-search whit at least 80% recall. This means that the model will be able to capture the large majority of offenses.
- We also consider that the model's precision should not vary by more than 5 percentage points across all protected classes, for each of the stations.
- Finally, the client has requested that searches are performed only when there is more than a 10% of likelihood of success. This means that we should keep the decision threshold above 0.1.

2. Dataset analysis

2.1 General analysis

The UKDP has provided us with a dataset that is composed of 856 610 records and 16 features. The description and data type of the 16 features is presented below:

- Observation_id - [text] Unique identifier for a particular instance of stop-and-search
- Type - [categorical] the type of search that was conducted (e.g. Person, Person-and-vehicle, Vehicle)
- Date - [timestamp] date when the operation occurred
- Part of a policing operation - [boolean] whether the search was part of a large scale policing operation
- Latitude - [numeric] latitude coordinate where the search occurred
- Longitude - [numeric] longitude coordinate where the search occured
- Gender - [categorical] gender of the individual as established by the officer
- Age range - [ordinal] age range of the individual as established by the officer
- Self-defined ethnicity - [categorical] ethnicity of the individual as defined by themselves
- Officer-defined ethnicity - [categorical] ethnicity of the individual as defined by the officer
- Legislation - [categorical] legislation that sustains the stop-and-search operation
- Object of search - [categorical] probable cause that led the officer to conduct a stop-and-search
- Outcome - [categorical] the final outcome of the stop-and-search (e.g. 'A no further action disposal', 'arrest', etc.)
- Outcome linked to object of search - [boolean] whether the outcome of the stop-and-search was linked to the object that led to it
- Removal of more than just outer clothing - [boolean] whether the office requested the individual to remove more than just and outer piece of clothing. Should not include situations such as a request to remove a heavy cloak during winter.
- Station - [categorical] the station where the officer is based at.



The provided data includes features that have missing values, as presented in the image to the left. For instance, the feature Outcome linked to the object of search has a significant amount of missing values. In fact, we have found that some stations have no, or a low percentage of, values for this feature. The complete list of such stations is shown in Table 1 of section 5. This compromises their usability as this feature is used to establish the success of a stop-and-search operation (refer to section 1.2). As such, we decided to remove from the data stations that had missing values for more than 90% of their records. Other missing values were filled as False. Similarly, for the Removal of more than the outer clothing feature, there were also some stations with a high percentage of missing values which were dealt with in the same fashion as for Outcome linked to the object of the search. These are also shown in Table 1 of section 5.

We have also found that the majority of the searches are of the type of Person search and that there is a very low minority of instances of Vehicle searches (image 1 of section 5).

Image 2 of section 5 provides an overview of the number of occurrences for each protected class. We can see that the Female gender has a very low number of instances, whereas the Other gender is almost negligible. The same can be said about Mixed ethnicity and the under-10 age range, which have very low representation. We can observe that the majority of the records are for individuals of the Male gender, White ethnicity, and in the 18-24 age range. Image 3 of section 5 provides an overview of the number of records for each protected class tuple. We can see that the large majority of records are for White Males in the 18-24 age range.

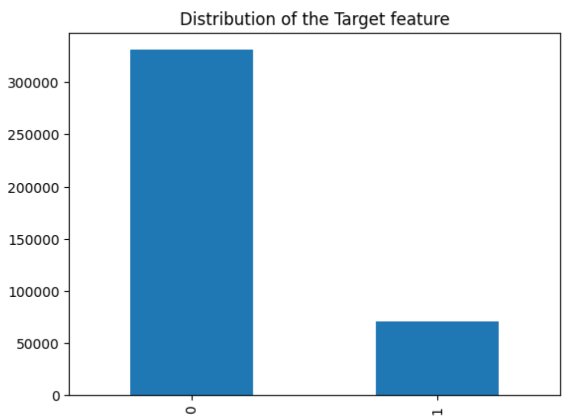
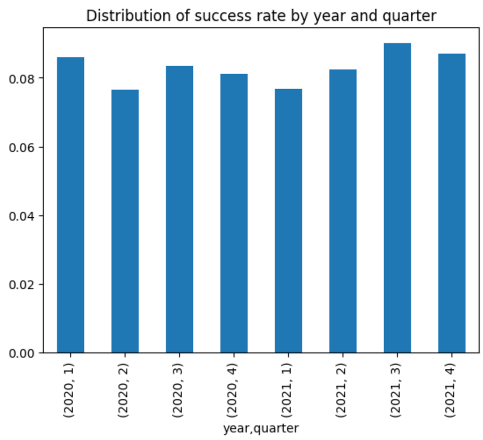


Image 4 of section 5 provides the geographical distribution of stop-and-search operations colored by the station. Comparing this image with the UK map we can see that most of the entries are centered in England and Wales, with a low number of occurrences in North Ireland and Scotland.

As we established above, there are some stations that have no values for Outcome linked to the object of the search. After removing these we arrive at the distribution shown in the image to the left. We can clearly see that the success of the stop-and-search (1 in the image) is the clear minority class - which makes this an unbalanced dataset.

The image to the right provides the evolution of the success rate (instances where the stop-and-search was a success divided by the total number of stop-and-search actions) over the years and quarters of the dataset. Through its observation, we established that there seem to have been some changes in the success rate. This could be due to a number of factors, but we still consider the differences to be quite low. We ran similar analyses for days of the week (image 5 of section 5) and hours of the day (image 6 of section 5) and reached similar conclusions.



After encoding the categorical features, we were also able to do some statistical analysis. Image 7 of section 5 presents the correlation between each of the features. It is interesting to note the fairly high correlation between the self-defined and officer-defined ethnicity features (0.73), leading to the assumption that the officers are apt at identifying the ethnicity of individuals. But, if we look a little deeper into the data we can see that is not the case. In fact, if we remove all records of the White ethnicity (the majority of records as established above) from the correlation analysis, this value drops to 0.32, which might suggest that agents struggle in correctly identifying the ethnicity of an individual if they are not White.

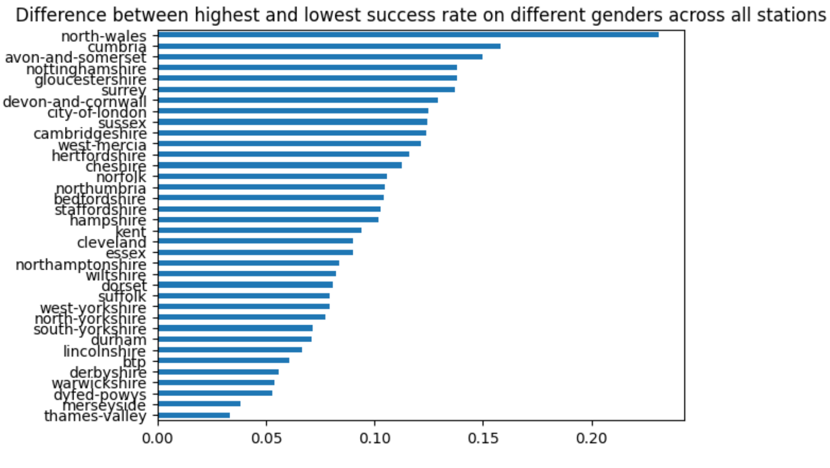
Image 8 of section 5 provides a look into possible outliers in the data. We can see that the data is mostly well-behaved. There are some entries with very low representation, such as the “Seals of hunting equipment” object of the search, but seeing as these mostly happen in categorical features we won’t consider them outliers as much as low-frequency entries.

Finally, image 9 of section 5 provides the distribution of all features. This once again corroborates the idea that most features are well-behaved. We have some features with large tails such as ‘self-defined ethnicity’ and ‘object of search’, and Latitude and Longitude which are closer to a normal distribution.

2.2 Business questions analysis

We were tasked with finding two instances of possible discrimination/bias. First, we will establish whether there are any facts that support instances of discrimination against any of the protected classes across the stations. Secondly, we will establish whether there are any stations that are more prone to requesting the removal of more than outer clothing across all protected classes. For this analysis, we removed the stations in table 1 of section 5 if the feature under analysis (stop-and-search success or removal of more than outer clothing) was missing for 90% of the records. Also, we only considered tuples whit a minimum of 30 samples for each of the stations, so that underrepresented classes don’t impact the results.

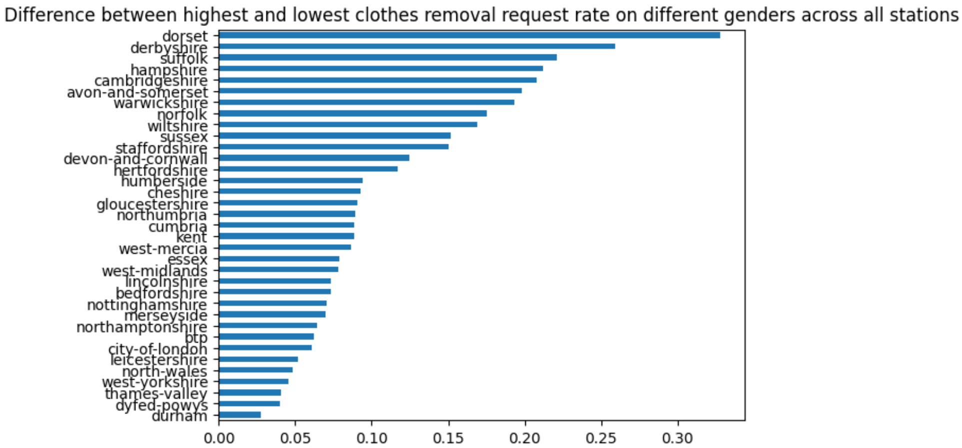
By observing the figure below we can conclude that most stations show a difference in stop-and-search success rate higher than the threshold established in section 1.2, which lets us conclude that there is evidence of discrimination across most departments.



In fact, only 2 departments are below the threshold - Merseyside and Thames-Valley. Referring to table 2 of section 5 we can see the full list of discovery rates for each tuple and the tuples that show the highest and lowest success rates in each station. The department with the highest difference is North-Wales - due to a low success rate for individuals of the Other ethnicity and Female gender and a high success rate for Asian-Male individuals. This indicates that Other-Female individuals are clearly oversearched in this station. By observing images 10 and 11 of section 5 we can see that the tuples that are overserached across more stations are Black-Female, Other-Male, and White-Female individuals, which indicates that there is a clear bias against these classes, whereas the Asian-Male tuple is the one with the highest success rate for 10 of the stations.

We also checked if there was evidence of discrimination across search objectives. Once again we only considered objectives where each race-gender tuple had at least 30 records. By observing image 12 of section 5 we can conclude that there is evidence of discrimination. The objective with the highest difference is Controlled drugs, due to a very low success rate for Black-Other individuals that are clearly oversearched, and a high success rate for Mixed-Males. The complete list of differences and success rates for each Race-gender tuple can be found in table 3 of section 5.

In regards to requests for the removal of more than outer clothing, we can see in the figure below that there is less cause for concern, but still, most stations show a difference in request rate higher than the threshold established in section 1.2, which lets us conclude that there is evidence of discrimination across most departments.



In this instance, the differences for most stations are closer to the threshold, although in some cases the difference between tuples is much higher in some stations, such as Dorset. By looking at images 13 and 14 of section 5 we can see that the tuple that shows the highest request for removal of clothing for more stations are Black-Male individuals in the 18-24 age range, whereas the tuple that has the lowest requests for more stations are White-Females in the 10-17 age range. In fact, by looking at these images we can conclude that agents tend to request removal of clothing less for underage individuals and more for Male individuals. It's also interesting to notice that White-Females over 25 show as the ones with the highest number of requests for 6 stations while they are the ones with the lowest success rate as established above. This could mean that this class is being overly targeted by these requests.

Finally, we checked whether requests for the removal of outer clothing have any impact on the success rate of search-and-stop operations. By looking at image 15 of section 5 we can conclude that there is no evidence for such a link. In fact, the station with the lowest number of requests for the removal of outer clothing has the highest search success rate. Even if we consider this as a singularity, it's very visible that with a decrease in requests for the removal of clothing, there is no similar decrease in the success rate of stop-and-search operations.

2.3 Recommendations

As established in section 2.2 there is reason to believe that the officers of some stations choose to stop and search some protected classes more than others, and this is especially true for some search objectives such as Controlled Drugs. Although we believe that there is cause for concern, we still agree that policing is a high-risk profession, where the decision to stop or not stop someone can have dire consequences. Because of this, we won't suggest that the results could be due just to biased officers, but also to ambiguous situations that they face. While it is acknowledged that some classes or communities will have higher levels of delinquency, the rate of success should not be expected to have huge variations when removing all possible subjective criteria or unconscious biases. Our suggestion here is to do a deeper dive into two reasons. Whether this difference is caused by over-zealousness in communities that could be deemed at risk, or whether it is caused by conscious/unconscious bias that the officers might have. To check the former, we suggest using exogenous features such as average income to check if there is a correlation with the success rate. In regard to the latter, we suggest adding officer data to the dataset and checking whether there are any particular officers that could be contributing more to the difference and promoting debriefing sessions to check for reasons that could escape this analysis.

On the topic of over-requesting for the removal of more than outer clothing, we also established that there are some stations that contribute significantly to skewing this metric, specifically for White Females over 25. As in the above paragraph, we suggest adding officer information and checking whether the differences might be caused by specific agents, and debriefing them on the reason that would lead to such differences.

3. Modeling

3.1 Model specifications

The deployed model was obtained through an iterative process of trial-and-elimination based on the requirement metrics established in section 1.2. We started from a simple baseline and iteratively tested different approaches for missing values imputation, feature engineering, and dealing with data unbalance. We also tested several Machine Learning Models. At each step, we chose the hypothesis that provided the highest value of Recall and the lowest average precision difference between each race-gender tuples across all stations. If these metrics didn't paint the full picture we also looked to metrics such as the Area Under de Receiver Operating Curve (AUROC) to make a decision on what model to choose for the next step. All models tested and the respective performance metrics can be found in table 5 of section 5, and a detailed description of each of the steps mentioned and alternative models considered can be found in section 3.3.

The final version of the model was based on an implementation of the Naive Bayes model, considering the following features:

- Date
- Type
- Object of search
- Part of a policing operation
- Latitude
- Longitude

We considered the target label as all records where the outcome feature was different than 'A no further action disposal' and the Outcome linked to the object of search feature was True.

We removed all records from the training dataset where the station was Metropolitan, Humberside, Leicestershire, and Lancashire. This was due to a lack, or very low number, of values in the Outcome linked to the object of search feature that would prevent correct labeling of the data. Later we found out that other stations also had no values for these features, but at the time of deployment, they were not removed from the training data.

We created a preprocessing pipeline that applied separate transformations for date columns, categorical columns, boolean columns, and numerical columns.

The date feature was transformed into separate features for year, quarter, day of the week, and hour. This was done to reduce the variability of this feature and to ensure that the model is able to correctly capture time-related contributions from the data. These features were then scaled using an implementation of min-max scaling.

For the categorical features Type and Object of search, we filled any missing values with the most common value for that feature. Since these were categorical features we had to encode them into numeric values. As such, we used an implementation of the target encoder, that converts each category to a scaled version of the average target value of the corresponding category. We decided on this encoder not only because it provided the best overall metrics, but also because it is appropriate to deal with unordered categorical variables such as these, ensuring that the model won't pick on any unwanted order relation between the data. Any unseen categories during training will be labeled with the average value of the target label. No scaling was needed since the target encoder already provided the necessary scaling.

For the boolean feature Part of a policing operation, we simply filled missing values with the False boolean value and transformed the values into integers - True as 1 and False as 0. No scaling was needed here also.

Finally, for the numerical Latitude and Longitude features, we filled the missing values with the average Latitude or Longitude value that the feature had for that particular station. This feature has high variability, but ultimately we decided on keeping them since we established that they had high feature importance using a baseline decision tree model. If the station of a new record was not available in the training data set, the missing value will be filled with the most common value. Finally, these features were scaled using an implementation of min-max scaling.

Since we were dealing with an unbalanced data set - as established in section 2.1 - before transforming the features and training the model we applied resampling by using a Random Under sampler. This resampling technique randomly removes records from the majority class until a balance between target classes is reached. We considered a random seed to keep results constant across iterations.

To measure the performance of the model we applied a train-test split with 30% of records in the test set. We did not consider stratifying the data. We also considered a random seed here to keep results consistent across testing.

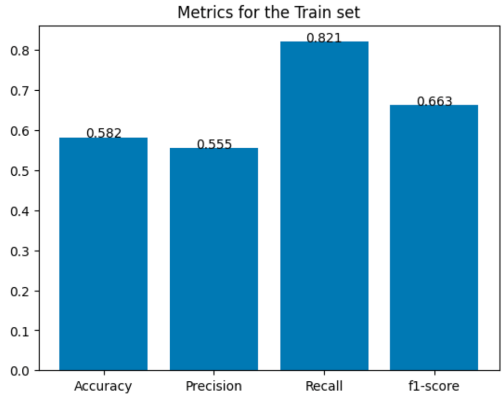
We chose the Naive Bayes Machine Learning model seen as it provided the highest values of Recall (0,821) as well as the lowest average precision difference across all stations (0,113) with the lowest amount of overfitting. It also proved to be a very efficient model, producing predictions fairly quickly.

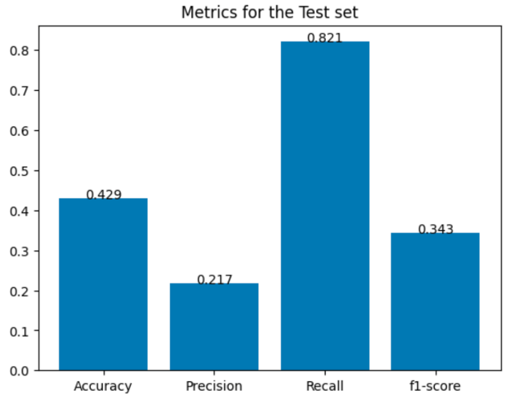
We did not perform an analysis on the decision threshold at this point, so we kept it at 0.5. This still ensures the client's requirement of requesting searches only when there is a probability of success higher than 10%, but we will try to optimize it at a later stage.

Finally a note on addressing model bias. Ultimately, after several different attempts that we will expand upon in section 3.3, we found that the best solution to reduce model bias was to simply remove the features related to the protected classes, Gender, Age range, and Officer-defined ethnicity.

3.2 Model Performance and expected outcomes

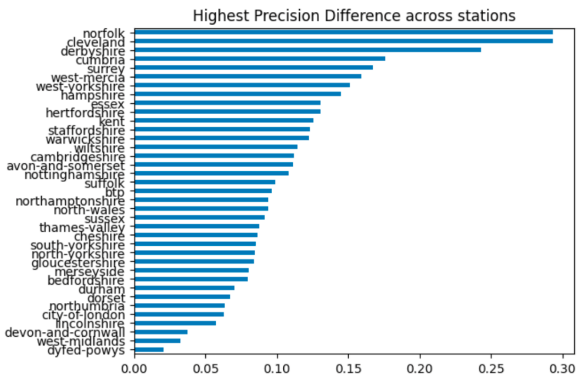
The image below provides a breakdown of the major performance metrics for the final version of the model. The models were trained using 70% of the dataset (train set) and tested on the remaining 30% (test set). The deployed model was trained on the whole dataset, so the metrics for the deployed model can be slightly different.

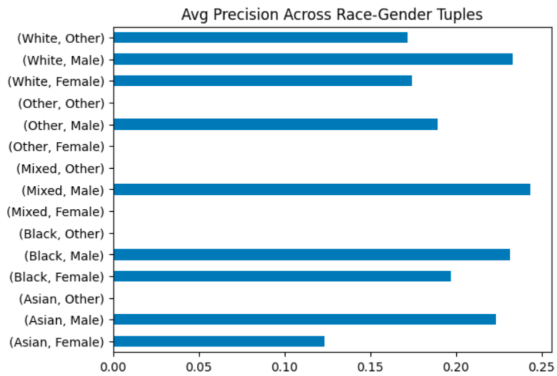




As can be observed from the image above on the right, we were able to obtain a recall of 0.821 on the test set, which clears the requirement established in section 1.2 (0.8). It's also possible to notice a decrease in the metrics from the train set to the test set, especially the Precision value, which might be indicative of overfitting. Nevertheless, we feel that the model still provides the required performance since the drop in accuracy and recall wasn't as severe.

Unfortunately, we were not able to succeed on the second requirement - precision should not vary by more than 5 percentage points across all protected classes. The images below provide an overview of the average precision for each protected class (on the right) and the average difference between the highest and lowest precision value for each protected class (on the left). To ensure that the results made sense, for each Race-Gender tuple and station we considered a minimum of 30 samples, which is why some tuples don't have an average precision value. It can be observed that instead of the required maximum difference of 0.05, we got differences ranging from 0.02 to 0.3. By looking at the breakdown by subclasses we can observe higher precision for tuples with higher representation (male individuals), and the lowest precisions for underrepresented classes (Female genders and Other ethnicities).





Analyzing results for the test dataset we observed that while the original data had 115 766 searches performed, the model would only suggest searches for 79 536 cases, which means a reduction of 31.3%. This reduction did not translate to a comparable reduction in the number of cases missed. In fact, with the model only 3764 cases would have been missed - 17.91% of all successes in the original data. We feel that if the discovery success had been correctly leveled across subclasses, this drop would have been acceptable. It's also interesting to note that with the model, the success rate of search-and-stop would increase from 18% to 22%. Taking into account the reduction in the number of searches performed this would translate into a clear increase in search-and-stop efficiency.

Finally, we can see the complete success rate across stations for each race-gender tuple in table 6 of section 5 and compare these with table 2 of section 5. We can see that even though we didn't achieve the objective we were still able to reduce the average success difference from 0,1 to 0,09. Also, we were able to reduce the success difference for some stations - we now have more stations with a difference below 0.05 (8 stations instead of 2) and also below 0.10 (18 increased to 23).

3.3 Alternatives considered

As stated in section 3.1, we ran an iterative process to land on the best model, trying to maximize recall and reduce the average precision difference for each race-gender tuple across all stations.

For the first step, we established a very simple baseline model, considering all features, simple transformations - such as ordinal encoding for all categorical classes, and filling missing values with the most common values. We then trained several models to establish the best one. The models tested included the Gaussian Naive Bayes, Logistic Regression, kNN, Decision Tree classifier, Random Forest Classifier, and Gradient Boost classifier. Table 4 of section 5 shows the complete breakdown of metrics for these models. Ultimately the Naive Bayes provided the best results with a recall of 0.043. Even though it wasn't the one with the highest recall value, we still selected this one because it showed less evidence of overfitting and better AUROC (0.61).

We also checked the bias of this baseline model against the protected classes and established that it did quite poorly. Image 17 of section 5 shows the highest precision difference between protected classes for each station. The average value didn't seem that high (0.21), but this was simply due to the fact that the precision was 0 for a significant number of stations where it was unable to generate predictions, as can be seen in the aforementioned image.

From this point on, the performance metrics for the models considered at each step can be found in table 5 of section 5. We will refer to this table in the following paragraphs.

The first step to improve the model was feature selection. We decided to remove several features based on inter-feature correlation, variability, and feature importance established with a Decision Tree model. The results of the feature importance analysis can be found in Image 16 of section 5. We tested this hypothesis against also dropping the Latitude and Longitude features since they also had high variability. We decided on the first option even though it had a slightly lower recall simply because it had a lower average precision difference.

Afterward, we went through a step of feature engineering where we established the best features to extract from the Date column. Here we also tried alternatives for missing value imputation of the Latitude and Longitude features. We compared the following two hypotheses

- using One Hot Encoder for all categorical features and an Ordinal Encoder with explicit order for the age range
- using a custom Inputer for Latitude and Longitude, Target encoder for categorical features and the same Ordinal Encoder for Age range.

Once again we chose to sacrifice Recall and chose the model that had a better average precision difference.

To handle class unbalance we compared resampling through random oversampling and undersampling. We decided to use Undersampling since it provided the best value of Recall, at the expense of doing slightly worse on the average precision difference - although this was compensated with now having 6 stations under the 0.05 threshold for the precision difference.

We then ran a step of model selection, training once again several models with the newly established transformations and balancing. Gaussian Naive Bayes was still the model that provided the highest recall, lowest average precision difference, and least overfitting.

To reduce the bias in the model we tried several different approaches. The first option was to remove stations with the highest success rate difference across all classes and keep only the “less biased” ones. We tried this approach with and without the sensitive features, and with and without Latitude and Longitude features to try and reduce overfitting with a simpler model. Ultimately, simply removing the sensitive features led to the best results - a recall of 0.821 and an average precision difference of 0.113.

Finally, we went through a step of hyperparameter tuning. Since Naive Bayes models don’t have major hyperparameters to tune, we simply checked what implementation of Naive Bayes provided the best results, arriving at the Gaussian Naive Bayes that showed a recall of 0.82 on the test set. The Recall and Precision metrics on the test set for each model considered are shown in Table 7 of section 5.

As mentioned in section 3.1 we did not consider different decision thresholds, keeping it at the default value of 0.5.

As a sanity check, we also ran hyperparameter tuning for the gradient boosting model and tried to find the parameters that reduced overfitting, with no success. The best hyperparameters for the gradient boosting that we found were using 10 estimators, a max depth of 10, and a learning rate of 0.1. This provided a recall of 0.741, still lower than what we got with Gaussian Naive Bayes. A list of Recall and Precision metrics on the test set for this can also be seen in Table 7 of section 5. We also ran an Overfitting analysis for this model, considering different values for the number of estimators. The results can be seen in image 18 of section 5, and it’s pretty clear that there were no hyperparameters that could reduce overfitting for this Machine Learning model.

4. Model Deployment

4.1 Deployment specifications

The final model described in section 3.2 was deployed as an API on railway.app. We chose to use the flask framework for deployment since it was lightweight and simple to implement, but still provided the necessary performance and features the client requested. The API consisted of two modules - /should_search/, which handles requests for prediction, and /search_result/ that handles updates to a record with the true outcome result. The requests are stored on a postgreSQL database.

The database that will store predictions has the following columns, and their respective data types, with a description of the features that weren't present in section 2.1:

- observation_id - [text] [unique]
- type - [text]
- date - [text]
- part_of_a_policing_operation - [boolean] [nullable]
- latitude - [float] [nullable]
- longitude - [numeric] [nullable]
- gender - [text]
- age_range - [text]
- officer_defined_ethnicity - [text]
- legislation - [categorical]
- object_of_search - [text]
- station - [text]
- proba - [float] [nullable] the predicted probability
- outcome - [boolean] [nullable] the predicted outcome
- true_outcome - [boolean] [nullable] the true outcome

When /should_search/ receives a new request for prediction it will run a set of checks to verify the integrity of the request. If there are any issues with the received data one of the following messages will be returned by the app:

1. "(field)": (field type) error: (field) field is missing from request" - returned if (field) is missing from the request
2. "Missing columns: (field)" - returned if (field) is missing from the request.
3. "Unrecognized columns provided: (field)" - returned if (field) in the request contains an unknown field.
4. "Invalid value provided for (field): (input value). Allowed values are: (list of values)" - returned if the input value provided for a categorical column that should not have new values (e.g. age range) is other than the known values.

The checks done are listed below:

- Checks that the incoming request has an observation id that matches the necessary format. If the check is unsuccessful, the API returns error message 1.
- Checks that the incoming request has all necessary fields to produce a prediction (observation_id, date, part_of_a_policing_operation, latitude, longitude, gender,

age_range, officer_defined_ethnicity, legislation, object_of_search, and station). If not, produces error message 2.

- Checks that incoming request has no unknown fields. Otherwise, produces an error message 3.
- Checks that each of the fields has the correct data type. For fields that can be nullable, it will accept null values. A special note on part_of_a_policing_operation: since this field is supposed to be a boolean value and nullable entries should be accepted, as requested by the client, we had to establish that any incoming missing values are set to false when received since we could parse a null value as boolean. If this check does not pass message 1 is returned.
- Checks that incoming values for type, gender, age_range, and officer_defined_ethnicity are present in the known categories. It shouldn't be expected for these categories to change or be null. All other fields accept new values. If the check is unsuccessful the API will return message 4.
- Finally there is a check on whether the incoming observation_id is already present in the database. In this situation, the following message is returned by the API: "Observation ID: (id) already exists".

If all checks are passed, the API will predict the model on the new record, predict the outcome and probability, and store the values in the database. The API will then return the following message: "outcome: (value)" - where the value corresponds to a boolean that indicates whether the search should be performed, or not.

As for /search_result/, when a new request is received, the API will check whether the input observation_id exists in the database, if it doesn't an error will be returned with the following format: "Observation ID: (value) does not exist". If it does exist in the database, the true outcome will be stored in the database, and the observation_id, outcome, and true_outcome will be returned in a message with the following format:

"observation_id": (id),
"outcome": (true outcome stored),
"predicted_outcome": (predicted outcome stored in the database)

4.2 Known issues and risks

Regarding the application, some known issues have to do with the integrity checks that are being done. If the integrity checks aren't passed, the input is not being stored. Unfortunately, we couldn't come up with a solution where weird values would be stored just for safety. Also, there could be some inputs that prevent a prediction from being returned. We are assuming that some features such as age_range won't have new categories, but it could be that the users decide on further breaking the data with new categories - this will cause the data to be lost and no predictions to be returned.

We should also mention that there are some security concerns to the way the API is being deployed. The deployment is publicly available, and anyone with access to the domain will be able to send requests, as is the case for querying the database. This should be acceptable during the testing phase, but the API should be moved to a more secure deployment for production deployment. Finally, the API was deployed in the free tier of the railway, which has several limitations, such as the amount of time it is available (500 monthly hours) as well as the performance. This should also be acceptable during testing, but once more agents are expected to send requests the performance limitations might prove catastrophic for the API.

As for the model, we exposed the main issue in section 3.2. Comparing the average difference of precisions across protected classes we can observe that the model will introduce more bias for some stations. This could be due to the model being trained on biased data, as established in section 2.2, or simply due to our inability to improve the model and reduce bias. To deal with this issue, we have established some next steps to improve the model and reduce bias for our next iteration.

1. We did not try to optimize the decision threshold for this deployment, keeping it at the default value of 0.5. This still delivered the client's requirement of it being above 0.1, but during redeployment, we will try different decision thresholds.
2. We did not remove all stations that had missing values on the Outcome linked to the object of search for more than 90% of the records. This could skew the training results, and as such we will test removing them during redeployment.
3. It seems like the approaches we took to balance the data might be causing some overfitting. We will try to check different approaches or simplify the model and reduce overfitting.

Finally, we would be remiss not to mention concerns about data privacy. This dataset includes sensitive information such as age range, gender, and ethnicity and also private information such as the combination of location and time. We acknowledge that the conjunction of these features can be used to personally identify an individual and that we are storing them in a publicly accessible database and processing it without explicit consent from the individual to who the data pertains to, so we suggest a deep analysis of the necessity of the personal information that is being used here.

5. Annexes

	Total Records	Missing Outcome	Outcome is True	Missing Clothes Removal	Missing Outcome Ratio	Missing Clothes Removal Ratio
humberside	10306,00	10306,00	0,00	0,00	1,00	0,00
lancashire	15254,00	15254,00	0,00	15254,00	1,00	1,00
metropolitan	436867,00	436867,00	0,00	436867,00	1,00	1,00
west-midlands	9271,00	9108,00	123,00	20,00	0,98	0,00
leicestershire	8298,00	7551,00	5,00	0,00	0,91	0,00
cleveland	7907,00	6237,00	1301,00	7907,00	0,79	1,00
north-yorkshire	3563,00	2588,00	757,00	3563,00	0,73	1,00
surrey	10774,00	50,00	2899,00	10774,00	0,00	1,00
south-yorkshire	24300,00	0,00	2820,00	23955,00	0,00	0,99

Table 1 - Stations with high ratio of missing values for “Outcome linked to object of search” or “Removal of more than just outer clothing”

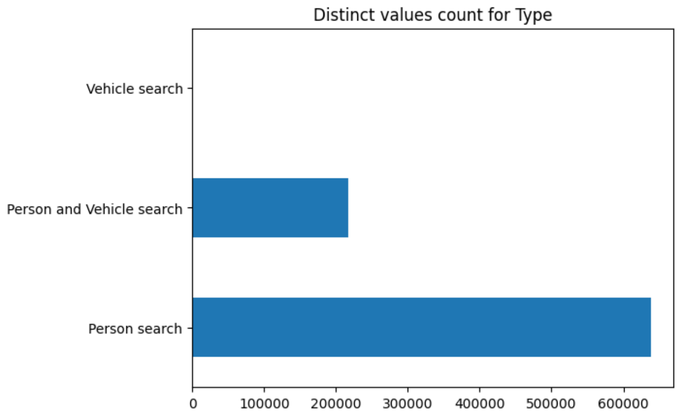


Image 1 - Distinct value count for Type

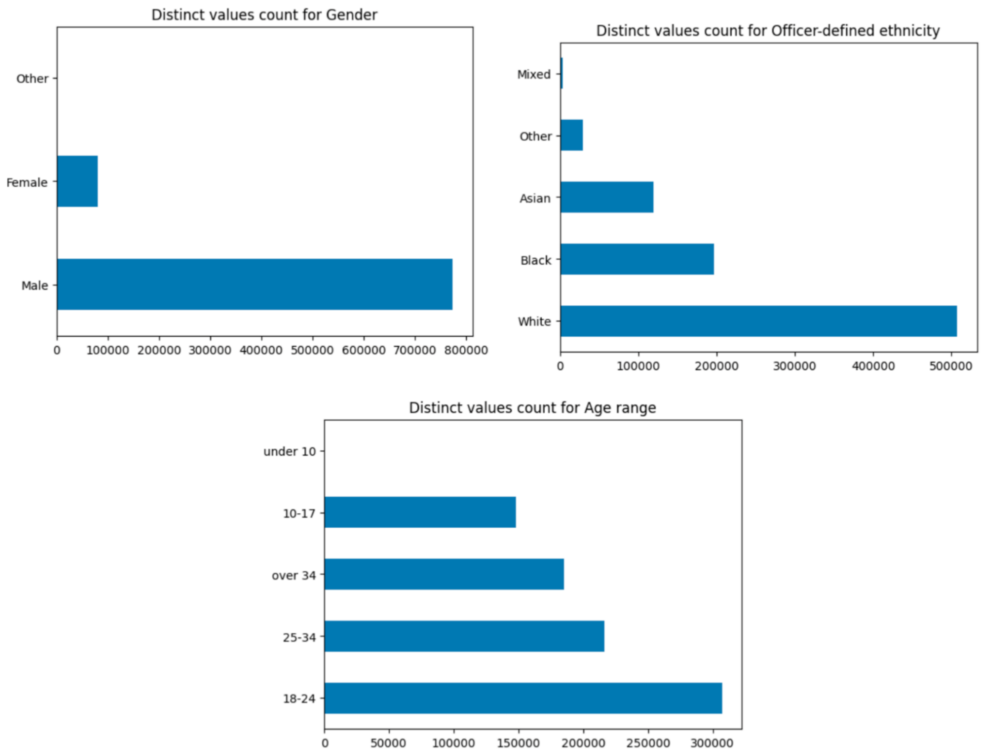


Image 2 - Distinct Record count for each protected class

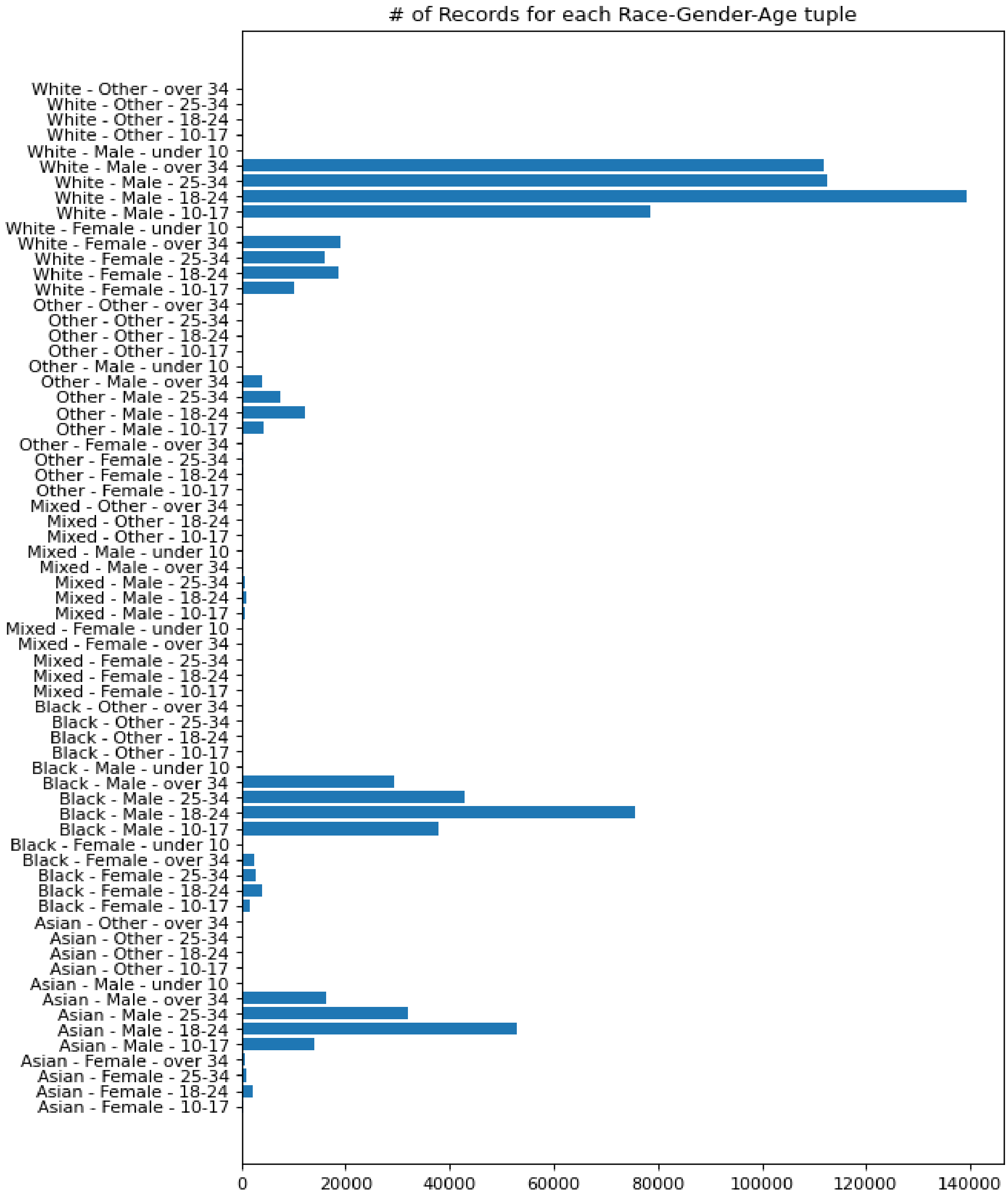


Image 3 - record distribution for each Race-Gender-Age tuple

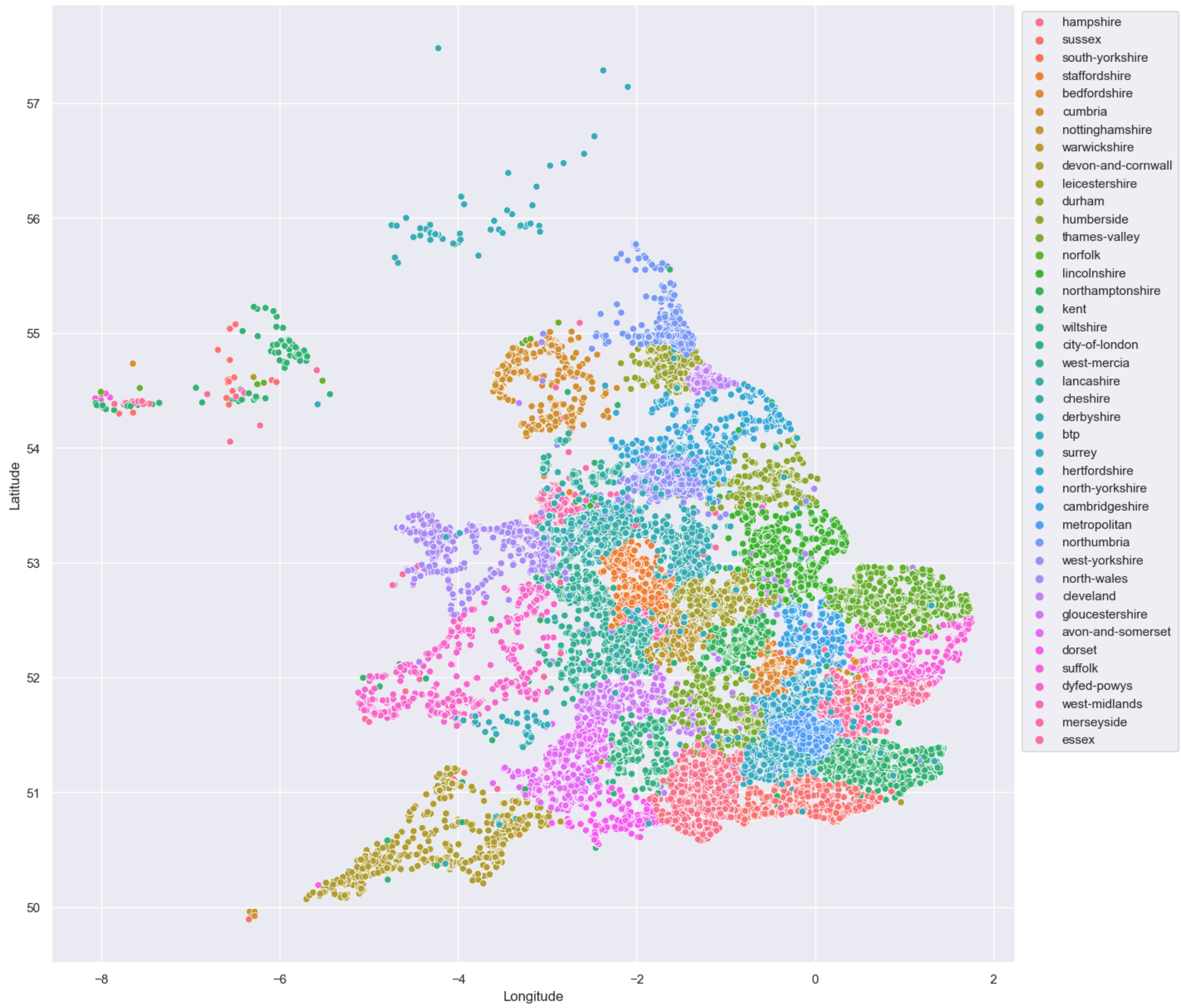


Image 4 - latitude and longitude distribution of search-and-stop action, colored by station

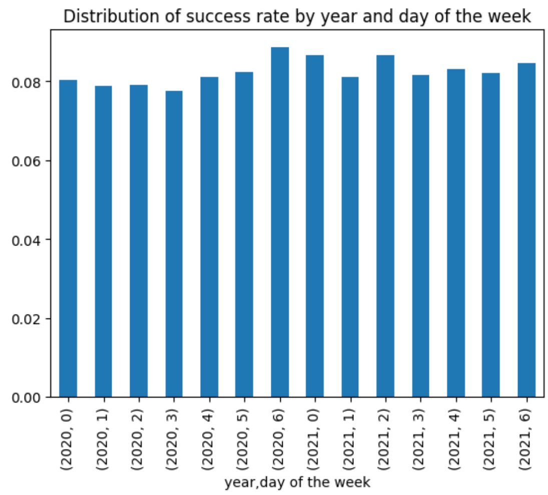


Image 5 - Rate of success over years and day of the week (0 for Monday and 6 for Sunday)

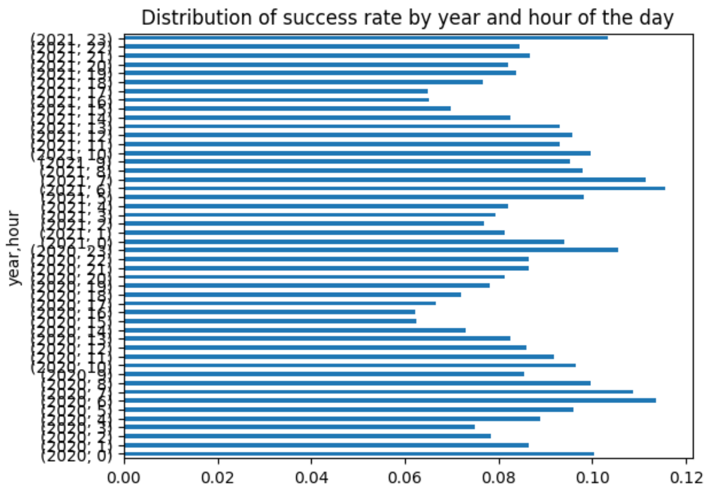


Image 6 - Rate of success over years and hour of the day

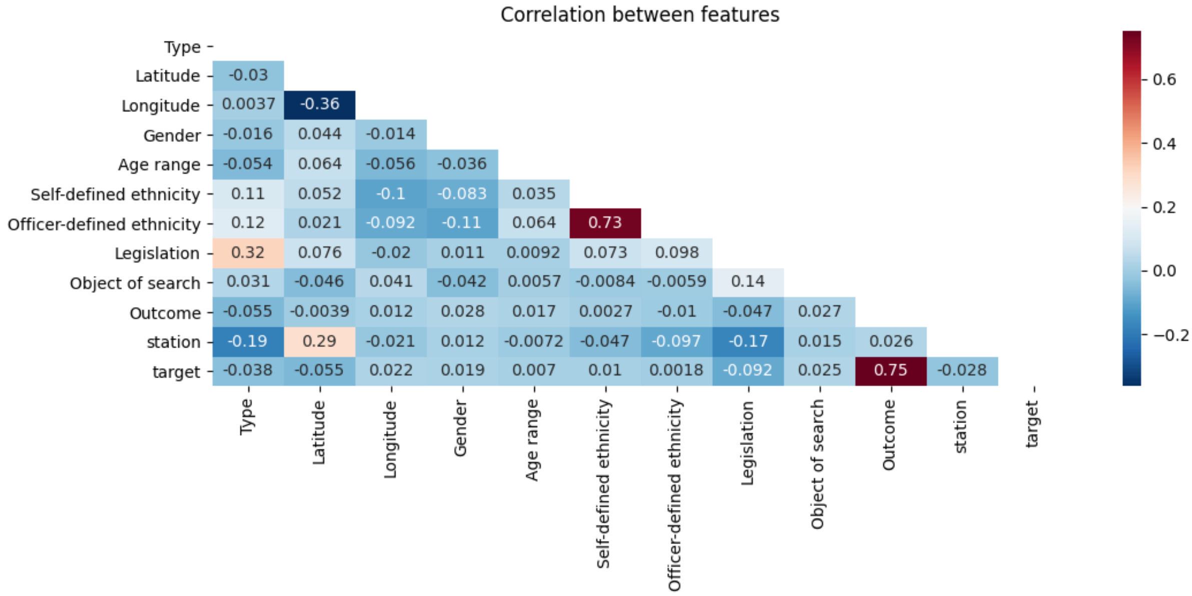


Image 7 - Correlation matrix for all features

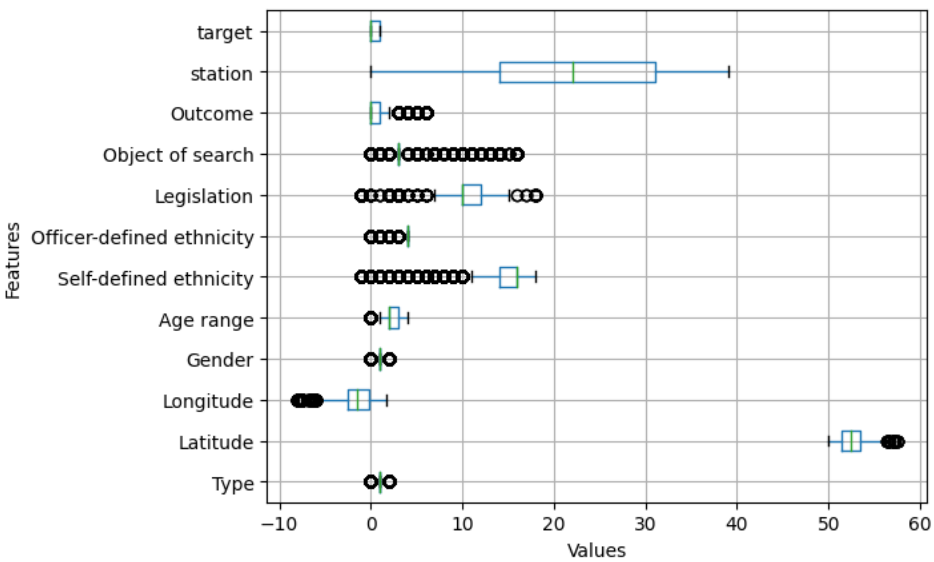


Image 8 - Box plot for all features

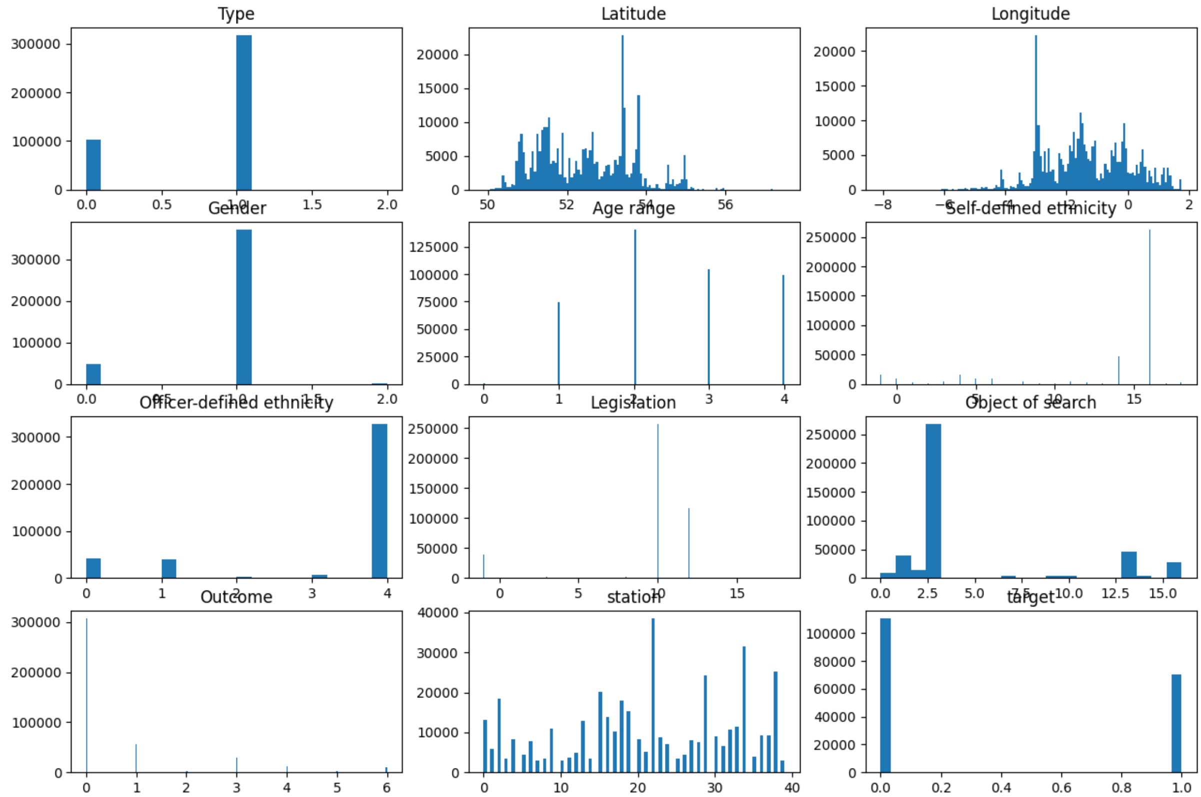


Image 9 - Histogram plots for all features

station	White Male	White Female	White Other	Other Male	Other Female	Other Other	Asian Male	Asian Female	Asian Other	Black Male	Black Female	Black Other	Mixed Male	Mixed Female	Mixed Other	diff	worst tuple	best tuple
thames-valley	0,19	0,16		0,17			0,17	0,17		0,18	0,15					0,03	('White', 'Male')	('Black', 'Female')
merseyside	0,15	0,14		0,17			0,17			0,17	0,18					0,04	('Black', 'Female')	('White', 'Female')
dyfed-powys	0,03	0,03		0,02			0,03			0,07						0,05	('Black', 'Male')	('Other', 'Male')
warwickshire	0,25	0,22		0,27			0,27			0,27			0,25			0,05	('Asian', 'Male')	('White', 'Female')
derbyshire	0,20	0,20		0,18			0,23			0,19						0,06	('Asian', 'Male')	('Other', 'Male')
btp	0,19	0,16		0,17			0,18	0,20		0,18	0,22					0,06	('Black', 'Female')	('White', 'Female')
lincolnshire	0,05	0,03		0,02			0,08			0,04						0,07	('Asian', 'Male')	('Other', 'Male')
durham	0,28	0,27	0,22				0,30									0,07	('Asian', 'Male')	('White', 'Other')
south-yorkshire	0,09	0,05		0,12			0,11	0,08		0,11	0,07					0,07	('Other', 'Male')	('White', 'Female')
north-yorkshire	0,20	0,17					0,24			0,24						0,08	('Black', 'Male')	('White', 'Female')
west-yorkshire	0,17	0,14		0,12			0,16	0,12		0,20	0,14					0,08	('Black', 'Male')	('Asian', 'Female')
suffolk	0,19	0,16		0,17			0,24			0,17			0,19			0,08	('Asian', 'Male')	('White', 'Female')
dorset	0,24	0,19					0,25			0,27						0,08	('Black', 'Male')	('White', 'Female')
wiltshire	0,20	0,16		0,22			0,21			0,13						0,08	('Other', 'Male')	('Black', 'Male')
northamptonshire	0,22	0,19					0,25			0,22	0,27					0,08	('Black', 'Female')	('White', 'Female')
essex	0,22	0,16		0,18	0,21		0,20			0,25	0,21					0,09	('Black', 'Male')	('White', 'Female')
cleveland	0,14	0,15		0,17			0,20			0,11						0,09	('Asian', 'Male')	('Black', 'Male')
kent	0,20	0,20		0,14	0,20		0,15	0,21		0,21	0,24		0,21	0,18		0,09	('Black', 'Female')	('Other', 'Male')
hampshire	0,28	0,25	0,18	0,28	0,26		0,27	0,20		0,25	0,24					0,10	('White', 'Male')	('White', 'Other')
staffordshire	0,20	0,17	0,15	0,11			0,21			0,20	0,21					0,10	('Black', 'Female')	('Other', 'Male')
bedfordshire	0,21	0,17		0,19			0,23	0,19		0,23	0,13					0,10	('Black', 'Male')	('Black', 'Female')
northumbria	0,22	0,19		0,21			0,25			0,30			0,21			0,10	('Black', 'Male')	('White', 'Female')
norfolk	0,13	0,11	0,08	0,10			0,18			0,18			0,12	0,09		0,11	('Asian', 'Male')	('White', 'Other')
cheshire	0,23	0,22		0,33			0,28			0,25						0,11	('Other', 'Male')	('White', 'Female')
hertfordshire	0,21	0,15		0,20			0,19	0,27		0,19	0,22					0,12	('Asian', 'Female')	('White', 'Female')
west-mercia	0,26	0,23		0,34			0,26	0,22		0,32			0,26			0,12	('Other', 'Male')	('Asian', 'Female')
cambridgeshire	0,24	0,19		0,15			0,28			0,27						0,12	('Asian', 'Male')	('Other', 'Male')
sussex	0,23	0,17		0,21			0,21			0,19	0,10					0,12	('White', 'Male')	('Black', 'Female')
city-of-london	0,26	0,27		0,24	0,22		0,24	0,21		0,24	0,34					0,13	('Black', 'Female')	('Asian', 'Female')
devon-and-cornwall	0,19	0,17		0,18			0,17			0,20	0,08					0,13	('Black', 'Male')	('Black', 'Female')
surrey	0,22	0,15		0,21			0,23	0,16		0,23	0,22		0,29			0,14	('Mixed', 'Male')	('White', 'Female')
gloucestershire	0,20	0,18		0,16	0,11		0,25			0,21						0,14	('Asian', 'Male')	('Other', 'Female')
nottinghamshire	0,25	0,19		0,20			0,21			0,21	0,14		0,21	0,11		0,14	('White', 'Male')	('Mixed', 'Female')
avon-and-somerset	0,23	0,18		0,26			0,26			0,23	0,33		0,24	0,24		0,15	('Black', 'Female')	('White', 'Female')
cumbria	0,22	0,14		0,30			0,27			0,26						0,16	('Other', 'Male')	('White', 'Female')
north-wales	0,18	0,14		0,15	0,11		0,34			0,15						0,23	('Asian', 'Male')	('Other', 'Female')

Table 2 - Success Rate for each Race-Gender Tuple across all stations, with largest difference for each station and the “worst” and “best” tuple

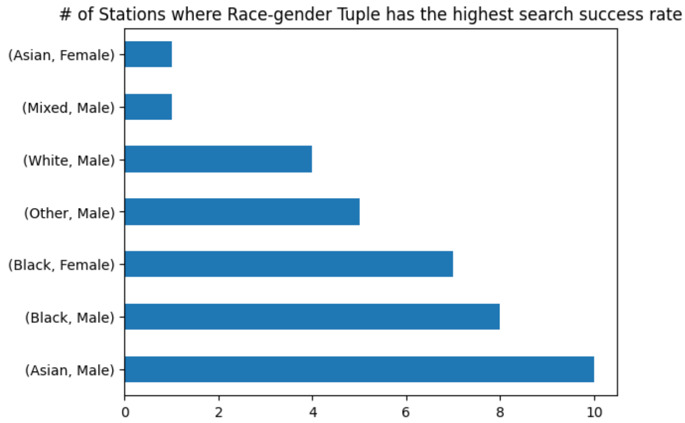


Image 10 - Number of instance where the race-gender tuple has the highest success rate

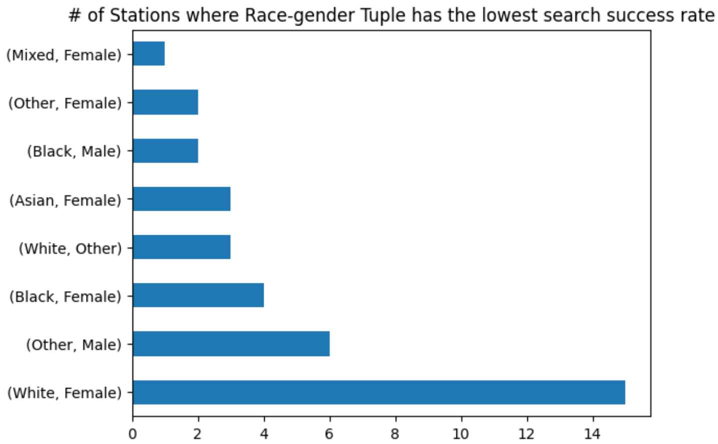


Image 11 - Number of instance where the race-gender tuple has the lowest success rate

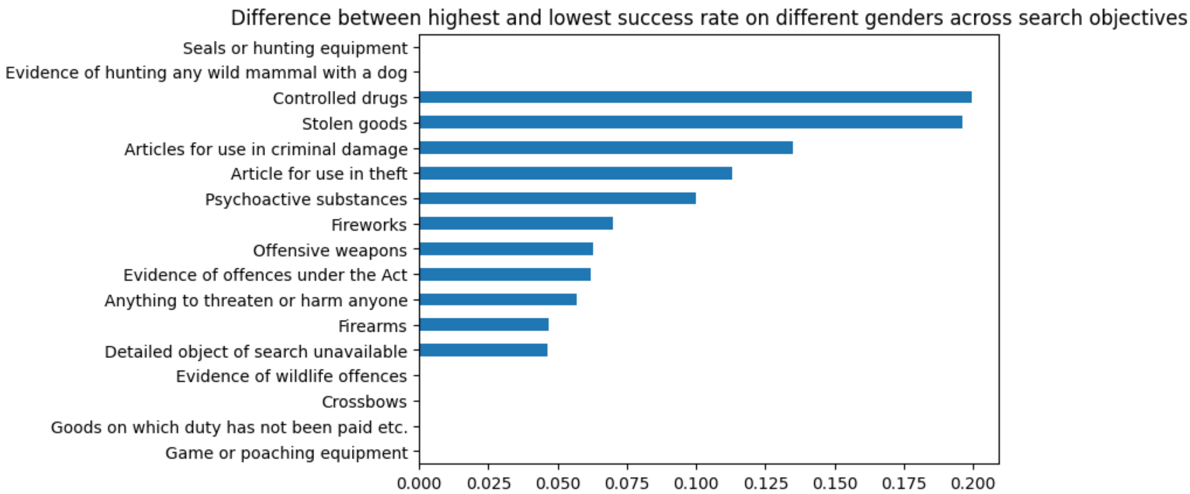


Image 12 - Difference between highest and lowest success rate for each race-gender tuple across search objectives

search objective	White Male	White Female	White Other	Other Male	Other Female	Other Other	Asian Male	Asian Female	Asian Other	Black Male	Black Female	Black Other	Mixed Male	Mixed Female	Mixed Other	diff	worst tuple	best tuple
Article for use in theft	0,09	0,09	0,04	0,10	0,06		0,07	0,09		0,10	0,15		0,09			0,11	('Black', 'Female')	('White', 'Other')
Controlled drugs	0,23	0,17	0,21	0,21	0,16		0,22	0,19		0,24	0,21	0,06	0,26	0,22		0,20	('Mixed', 'Male')	('Black', 'Other')
Stolen goods	0,17	0,23	0,23	0,17	0,30		0,15	0,19		0,17	0,23		0,19	0,34		0,20	('Mixed', 'Female')	('Asian', 'Male')
Offensive weapons	0,11	0,08	0,06	0,11	0,12		0,09	0,08		0,10	0,09		0,12	0,07		0,06	('Mixed', 'Male')	('White', 'Other')
Anything to threaten or harm anyone	0,08	0,05		0,08			0,06			0,07	0,08		0,11			0,06	('Mixed', 'Male')	('White', 'Female')
Articles for use in criminal damage	0,09	0,09		0,13			0,07			0,05			0,19			0,13	('Mixed', 'Male')	('Black', 'Male')
Firearms	0,10	0,09		0,13			0,10			0,08						0,05	('Other', 'Male')	('Black', 'Male')
Game or poaching equipment	0,17															0,00	('White', 'Male')	('White', 'Male')
Psychoactive substances	0,26	0,18		0,17			0,19			0,27			0,19			0,10	('Black', 'Male')	('Other', 'Male')
Fireworks	0,04	0,09					0,08			0,02						0,07	('White', 'Female')	('Black', 'Male')
Evidence of offences under the Act	0,12	0,08		0,12			0,11			0,12			0,14			0,06	('Mixed', 'Male')	('White', 'Female')
Goods on which duty has not been paid etc.	0,23															0,00	('White', 'Male')	('White', 'Male')
Detailed object of search unavailable	0,05	0,00														0,05	('White', 'Male')	('White', 'Female')
Crossbows	0,17															0,00	('White', 'Male')	('White', 'Male')
Evidence of hunting any wild mammal with a dog																		
Evidence of wildlife offences	0,11															0,00	('White', 'Male')	('White', 'Male')
Seals or hunting equipment																		

Table 3 - Success Rate for each Race-Gender Tuple across all search objectives, with the largest difference for each station and the “worst” and “best” tuples

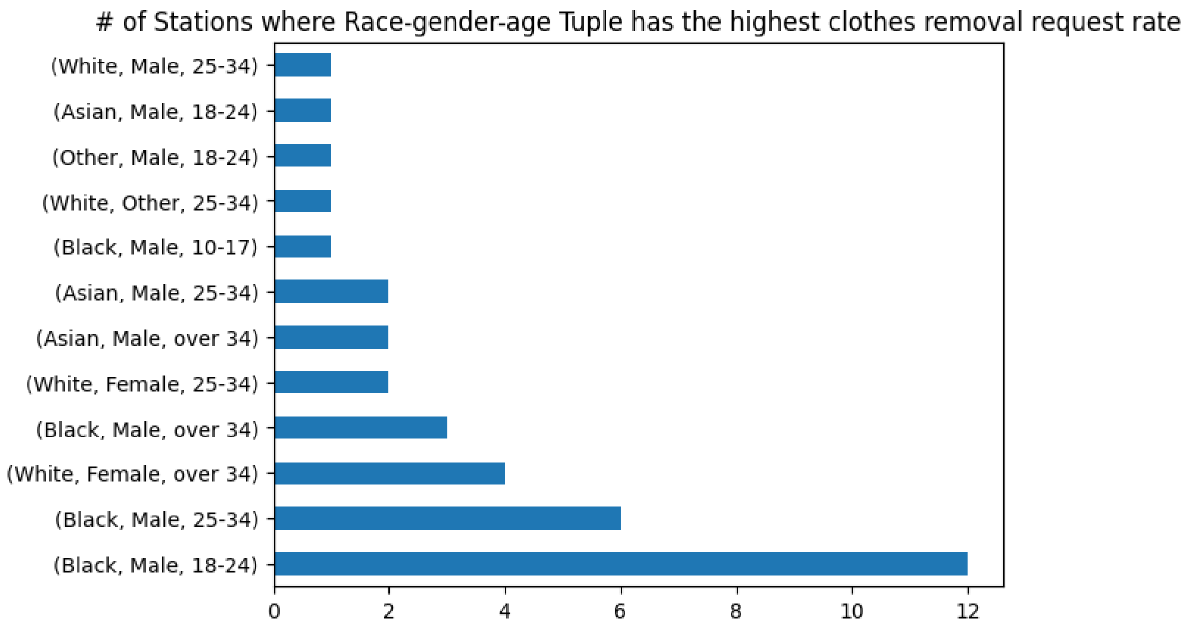


Image 13 - Number of instances where the race-gender-age tuple has the highest rate of requests for removal of more than outer-clothing

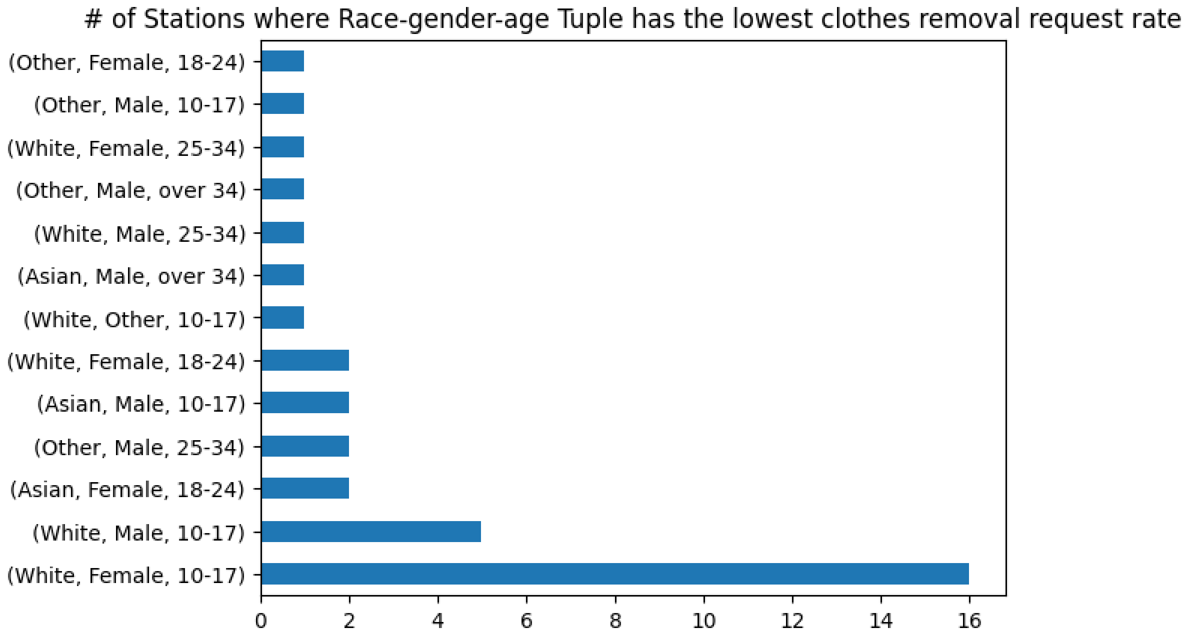


Image 14 - Number of instances where the race-gender-age tuple has the lowest rate of requests for removal of more than outer-clothing

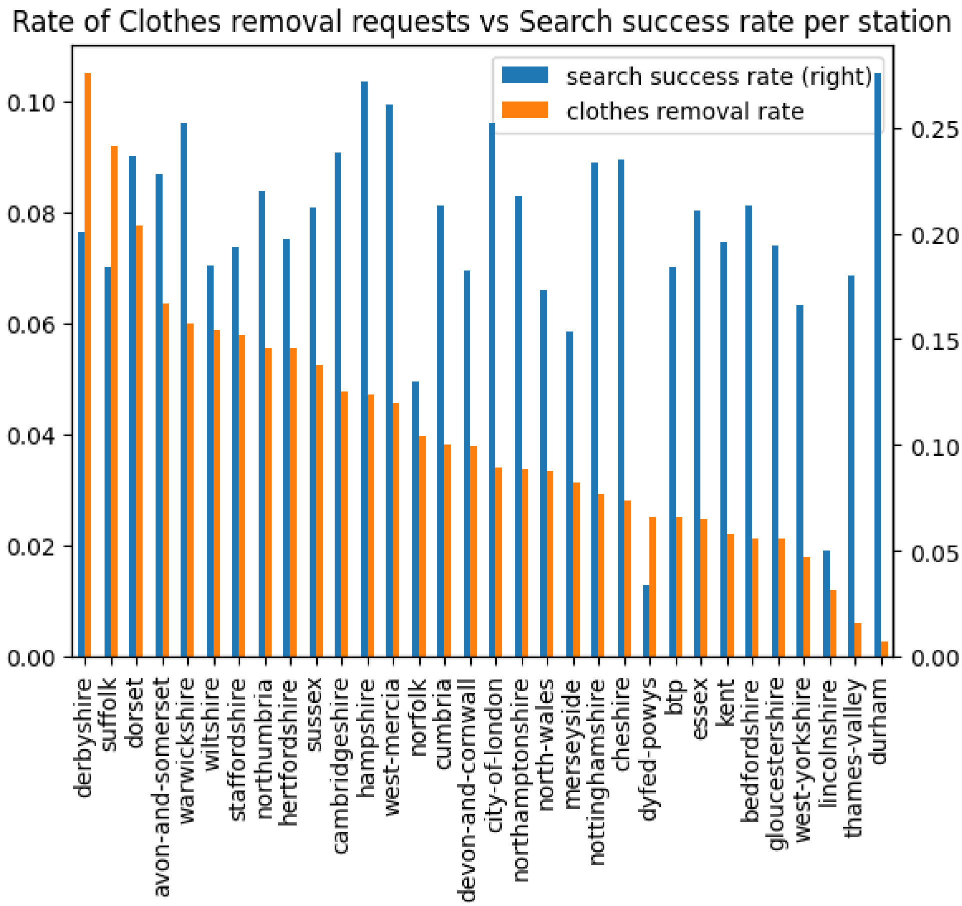


Image 15 - Clothes Removal rate vs Search success rate across stations

	Train Set				Test Set				AUROC
ML Model	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	
Gaussian NaiveBayes	0,8	0,24	0,043	0,073	0,802	0,239	0,043	0,072	0,61
Logistic Regression	0,817	0,017	0	0	0,818	0,042	0	0	0,58
Decision Tree Class.	0,988	0,998	0,939	0,967	0,729	0,266	0,28	0,273	0,56
Gradient Boosting Class.	0,817	0,833	0	0	0,818	0,5	0	0	0,67
Linear SVC	0,817	0	0	0	0,818	0	0	0	NA
k Nearest Neighbors Class.	0,841	0,66	0,267	0,38	0,79	0,301	0,12	0,171	0,6
Random Forest Class.	0,988	0,986	0,951	0,968	0,809	0,382	0,081	0,133	0,68

Table 4 - Performance Metrics for Baseline Models

		Train Set				Test Set					Discrimination		
Step	Model	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	AUROC	Number of Good Departments	Number of Problematic Departments	Average Precision Difference
Baseline	Gaussian Naive Bayes	0,8	0,24	0,043	0,073	0,0802	0,239	0,043	0,072	0,61	16	21	0,22
Feature Selection	Keep Only Feat with High Importance	0,804	0,238	0,033	0,059	0,805	0,231	0,032	0,057	0,58	19	18	0,16
	Also Drop Lat and Lon	0,804	0,245	0,035	0,062	0,805	0,24	0,035	0,06	0,57	19	18	0,2
Feature Engineering and Missing Values Imputation	One Hot Encoder and Ordinal Enc for Obj of Search	0,77	0,254	0,133	0,175	0,77	0,252	0,134	0,175	0,59	4	33	0,32
	Custom Imputer for Lat and Lon, Ordinal Encoder for Age and Target Encoder	0,799	0,254	0,005	0,084	0,8	0,251	0,051	0,084	0,62	17	20	0,188
Balancing	Random Undersample	0,586	0,563	0,771	0,651	0,47	0,223	0,772	0,346	0,62	6	31	0,162
	Random Oversample	0,587	0,565	0,757	0,647	0,479	0,224	0,76	0,346	0,62	4	33	0,152
Model Selection	Gaussian NaiveBayes	0,586	0,563	0,771	0,651	0,47	0,223	0,772	0,346	0,62	6	31	0,162
	Logistic Regression	0,581	0,558	0,775	0,649	0,46	0,22	0,776	0,343	0,62	4	33	0,153
	Decision Tree Class.	0,983	0,994	0,972	0,983	0,573	0,227	0,563	0,324	0,57	4	33	0,117
	Gradient Boosting Class.	0,625	0,607	0,707	0,653	0,57	0,252	0,7	0,371	0,67	3	34	0,127
	Linear SVC	0,581	0,558	0,78	0,651	0,456	0,219	0,781	0,343	NA	5	32	0,145
	k Nearest Neighbors Class.	0,718	0,714	0,728	0,721	0,563	0,225	0,578	0,324	0,59	2	35	0,116
	Random Forest Class.	0,983	0,98	0,986	0,983	0,603	0,258	0,634	0,367	0,66	2	35	0,137
Adjusting Bias	Remove Sensitive Features	0,582	0,555	0,821	0,663	0,429	0,217	0,821	0,343	0,61	3	34	0,113
	Remove Stations with Highest Avg Difference	0,591	0,571	0,729	0,641	0,501	0,229	0,737	0,349	0,62	4	33	0,19
	Remove Both	0,585	0,569	0,699	0,628	0,494	0,225	0,733	0,345	0,61	3	34	0,125
	Remove Both and also drop Lat and Lon	0,58	0,555	0,802	0,656	0,442	0,217	0,795	0,341	0,6	3	34	0,117

Table 5 - Performance Metrics for Models trained during each step of the improvement process

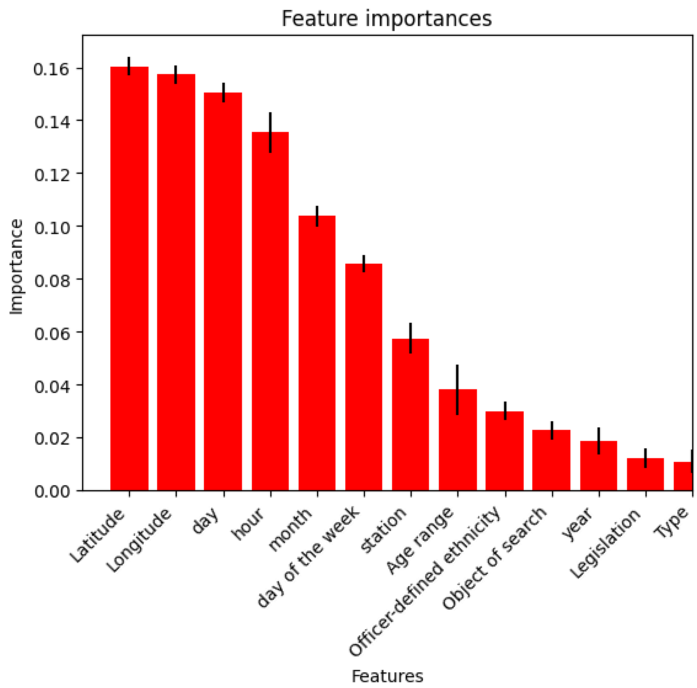


Image 16 - Feature importances for the Baseline Model

stations	White Male	White Female	White Other	Black Male	Black Female	Black Other	Asian Male	Asian Female	Asian Other	Mixed Male	Mixed Female	Mixed Other	Other Male	Other Female	Other Other	diff	worst tuple	best tuple
dyfed-powys	0,037423847	0,027777778														0,00964	('White', 'Male')	('White', 'Female')
devon-and-cornwall	0,188649081	0,17693837		0,185714286			0,173913043									0,01474	('White', 'Male')	('Asian', 'Male')
west-midlands	0,011246485	0,017045455		0,0155902			0,017489712	0								0,01749	('Asian', 'Male')	('Asian', 'Female')
bedfordshire	0,211706102	0,186440678		0,195488722			0,204678363									0,02527	('White', 'Male')	('White', 'Female')
lincolnshire	0,057471264	0,021621622		0,026666667			0,057142857									0,03585	('White', 'Male')	('White', 'Female')
cheshire	0,221585482	0,22265625		0,263157895			0,263157895									0,04157	('Black', 'Male')	('White', 'Male')
city-of-london	0,249158249	0,219178082		0,232081911			0,24609375						0,206521739			0,04264	('White', 'Male')	('Other', 'Male')
thames-valley	0,183300199	0,164759725		0,185499673	0,147058824		0,157466589	0,191489362					0,16			0,04443	('Asian', 'Female')	('Black', 'Female')
merseyside	0,155789048	0,135714286		0,180451128			0,18974359						0,195804196			0,06009	('Other', 'Male')	('White', 'Female')
dorset	0,243451464	0,198473282		0,25862069												0,06015	('Black', 'Male')	('White', 'Female')
warwickshire	0,269333333	0,204545455		0,26446281			0,269230769			0,241935484						0,06478	('White', 'Male')	('White', 'Female')
south-yorkshire	0,092128981	0,053475936		0,115517241	0,085714286		0,118292683	0,097560976					0,098765432			0,06481	('Asian', 'Male')	('White', 'Female')
northamptonshire	0,210023866	0,177304965		0,223744292			0,242424242									0,06512	('Asian', 'Male')	('White', 'Female')
surrey	0,224162341	0,16626506		0,184713376	0,157894737		0,183391003									0,06627	('White', 'Male')	('Black', 'Female')
sussex	0,222776392	0,156321839		0,203508772			0,205673759						0,208333333			0,06646	('White', 'Male')	('White', 'Female')
hertfordshire	0,209621993	0,141666667		0,184496124	0,214285714		0,168161435						0,169491525			0,07262	('Black', 'Female')	('White', 'Female')
gloucestershire	0,198966408	0,158730159		0,185185185									0,12			0,07897	('White', 'Male')	('Other', 'Male')
btp	0,186761964	0,170418006		0,183982684	0,227272727		0,177033493						0,147783251			0,07949	('Black', 'Female')	('Other', 'Male')
north-wales	0,172372372	0,123636364											0,092198582			0,08017	('White', 'Male')	('Other', 'Male')
avon-and-somerset	0,243949289	0,187145558		0,228506787			0,229508197			0,267379679			0,27027027			0,08312	('Other', 'Male')	('White', 'Female')
suffolk	0,185018727	0,171003717		0,123188406			0,162162162			0,21			0,146341463			0,08681	('Mixed', 'Male')	('Black', 'Male')
wiltshire	0,23089172	0,1484375		0,197368421			0,142857143									0,08803	('White', 'Male')	('Asian', 'Male')
staffordshire	0,192225772	0,163822526	0,133333333	0,211640212			0,223404255									0,09007	('Asian', 'Male')	('White', 'Other')
nottinghamshire	0,264399722	0,168674699		0,221453287			0,206278027			0,17699115						0,09573	('White', 'Male')	('White', 'Female')
kent	0,197654941	0,182885906		0,239035088	0,21875		0,161825726			0,235294118			0,137055838			0,10198	('Black', 'Male')	('Other', 'Male')
north-yorkshire	0,211734694	0,187096774		0,294117647			0,243243243									0,10702	('Black', 'Male')	('White', 'Female')
hampshire	0,284774665	0,250334672		0,254071661	0,181818182		0,252491694						0,296969697			0,11515	('Other', 'Male')	('Black', 'Female')
derbyshire	0,194444444	0,290322581		0,174418605			0,1796875						0,176470588			0,1159	('White', 'Female')	('Black', 'Male')
cambridgeshire	0,24537037	0,164835165		0,282608696			0,225806452									0,11777	('Black', 'Male')	('White', 'Female')
northumbria	0,213903743	0,178082192		0,29787234			0,248175182						0,188679245			0,11979	('Black', 'Male')	('White', 'Female')
west-yorkshire	0,167828685	0,14619883		0,200988468			0,148131673	0,15942029					0,077922078			0,12307	('Black', 'Male')	('Other', 'Male')
essex	0,216479401	0,132315522		0,256410256			0,220338983						0,211678832			0,12409	('Black', 'Male')	('White', 'Female')
durham	0,276422764	0,203389831	0,144927536													0,13149	('White', 'Male')	('White', 'Other')
cumbria	0,224899598	0,090909091														0,13399	('White', 'Male')	('White', 'Female')
west-mercia	0,259699948	0,226351351		0,353293413			0,258741259			0,203703704						0,14959	('Black', 'Male')	('Mixed', 'Male')
cleveland	0,132277835	0,147410359		0,018867925			0,156862745						0,21875			0,19988	('Other', 'Male')	('Black', 'Male')
norfolk	0,120456906	0,100278552		0,181208054			0,291666667			0,085365854			0,088235294			0,2063	('Asian', 'Male')	('Mixed', 'Male')

Table 6 - Success Rate for each Race-Gender Tuple across all stations, with largest difference for each station and the “worst” and “best” tuple for the test set using the Final Model

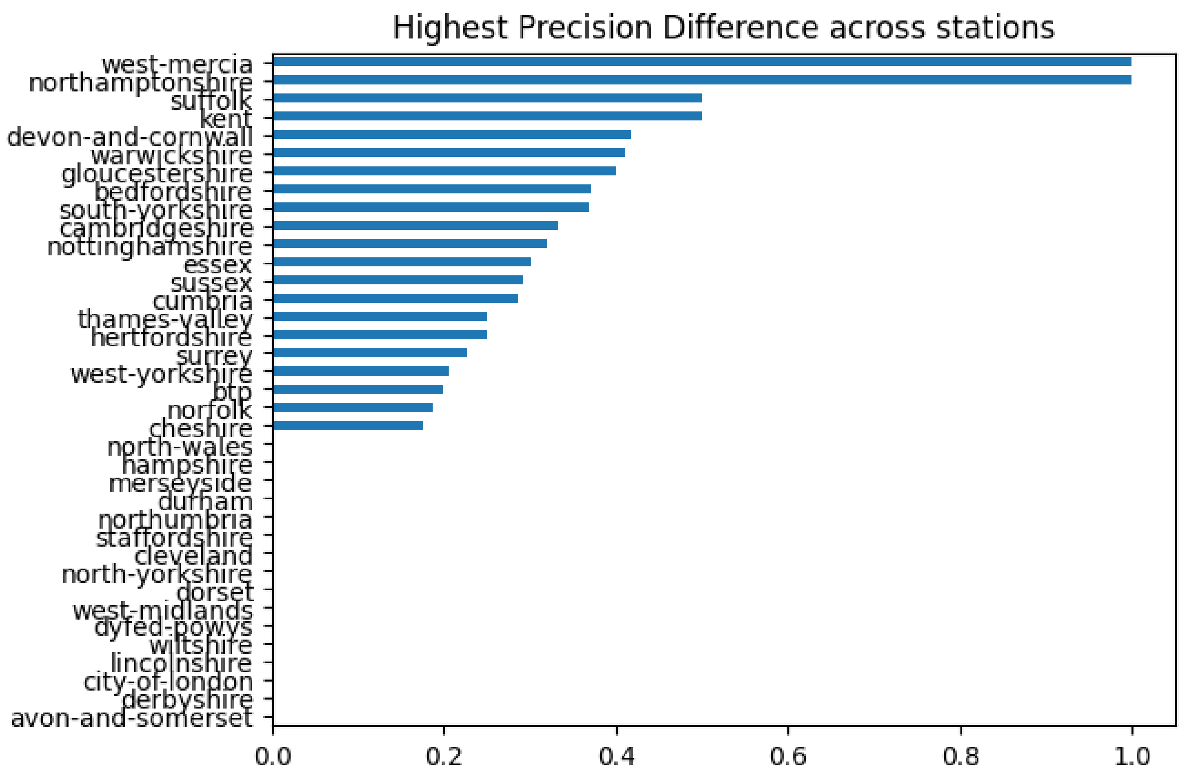


Image 17 - Highest precision difference between race-gender tuples across all stations

	Parameters	Test Set	
		Precision	Recall
NaiveBayes	GaussianNB	0,217	0,82
	BernoulliNB	0,191	0,5
	MultinomialNB	0,198	0,51
	CategoricalNB	0,192	0,497
GradientBoosting Classifier	max_depths: 5 ; learning rate: 0,1 ; n_estimators: 5	0,240	0,700
	max_depths: 5 ; learning rate: 0,1 ; n_estimators: 10	0,241	0,726
	max_depths: 5 ; learning rate: 0,1 ; n_estimators: 25	0,244	0,725
	max_depths: 5 ; learning rate: 0,1 ; n_estimators: 50	0,246	0,725
	max_depths: 5 ; learning rate: 0,1 ; n_estimators: 75	0,247	0,719
	max_depths: 5 ; learning rate: 0,1 ; n_estimators: 100	0,247	0,717
	max_depths: 5 ; learning rate: 0,1 ; n_estimators: 200	0,249	0,712
	max_depths: 5 ; learning rate: 0,1 ; n_estimators: 300	0,250	0,710
	max_depths: 5 ; learning rate: 0,1 ; n_estimators: 400	0,250	0,709
	max_depths: 5 ; learning rate: 0,5 ; n_estimators: 5	0,243	0,722
	max_depths: 5 ; learning rate: 0,5 ; n_estimators: 10	0,245	0,719
	max_depths: 5 ; learning rate: 0,5 ; n_estimators: 25	0,247	0,713
	max_depths: 5 ; learning rate: 0,5 ; n_estimators: 50	0,247	0,707
	max_depths: 5 ; learning rate: 0,5 ; n_estimators: 75	0,247	0,698
	max_depths: 5 ; learning rate: 0,5 ; n_estimators: 100	0,248	0,693
	max_depths: 5 ; learning rate: 0,5 ; n_estimators: 200	0,249	0,688
	max_depths: 5 ; learning rate: 0,5 ; n_estimators: 300	0,248	0,678
	max_depths: 5 ; learning rate: 0,5 ; n_estimators: 400	0,247	0,671
	max_depths: 5 ; learning rate: 0,9 ; n_estimators: 5	0,246	0,711
	max_depths: 5 ; learning rate: 0,9 ; n_estimators: 10	0,246	0,708
	max_depths: 5 ; learning rate: 0,9 ; n_estimators: 25	0,247	0,697
	max_depths: 5 ; learning rate: 0,9 ; n_estimators: 50	0,248	0,686
	max_depths: 5 ; learning rate: 0,9 ; n_estimators: 75	0,247	0,687
	max_depths: 5 ; learning rate: 0,9 ; n_estimators: 100	0,246	0,678
	max_depths: 5 ; learning rate: 0,9 ; n_estimators: 200	0,245	0,662
	max_depths: 5 ; learning rate: 0,9 ; n_estimators: 300	0,244	0,653

	max_depths: 5 ; learning rate: 0,9 ; n_estimators: 400	0,243	0,650
	max_depths: 10 ; learning rate: 0,1 ; n_estimators: 5	0,244	0,727
	max_depths: 10 ; learning rate: 0,1 ; n_estimators: 10	0,245	0,741
	max_depths: 10 ; learning rate: 0,1 ; n_estimators: 25	0,249	0,729
	max_depths: 10 ; learning rate: 0,1 ; n_estimators: 50	0,250	0,721
	max_depths: 10 ; learning rate: 0,1 ; n_estimators: 75	0,252	0,709
	max_depths: 10 ; learning rate: 0,1 ; n_estimators: 100	0,252	0,709
	max_depths: 10 ; learning rate: 0,1 ; n_estimators: 200	0,252	0,697
	max_depths: 10 ; learning rate: 0,1 ; n_estimators: 300	0,252	0,691
	max_depths: 10 ; learning rate: 0,1 ; n_estimators: 400	0,253	0,683
	max_depths: 10 ; learning rate: 0,5 ; n_estimators: 5	0,246	0,708
	max_depths: 10 ; learning rate: 0,5 ; n_estimators: 10	0,246	0,708
	max_depths: 10 ; learning rate: 0,5 ; n_estimators: 25	0,247	0,690
	max_depths: 10 ; learning rate: 0,5 ; n_estimators: 50	0,246	0,677
	max_depths: 10 ; learning rate: 0,5 ; n_estimators: 75	0,246	0,661
	max_depths: 10 ; learning rate: 0,5 ; n_estimators: 100	0,245	0,657
	max_depths: 10 ; learning rate: 0,5 ; n_estimators: 200	0,245	0,641
	max_depths: 10 ; learning rate: 0,5 ; n_estimators: 300	0,245	0,629
	max_depths: 10 ; learning rate: 0,5 ; n_estimators: 400	0,246	0,623
	max_depths: 10 ; learning rate: 0,9 ; n_estimators: 5	0,242	0,697
	max_depths: 10 ; learning rate: 0,9 ; n_estimators: 10	0,243	0,680
	max_depths: 10 ; learning rate: 0,9 ; n_estimators: 25	0,242	0,658
	max_depths: 10 ; learning rate: 0,9 ; n_estimators: 50	0,240	0,648
	max_depths: 10 ; learning rate: 0,9 ; n_estimators: 75	0,241	0,639
	max_depths: 10 ; learning rate: 0,9 ; n_estimators: 100	0,241	0,635
	max_depths: 10 ; learning rate: 0,9 ; n_estimators: 200	0,240	0,608
	max_depths: 10 ; learning rate: 0,9 ; n_estimators: 300	0,239	0,603
	max_depths: 10 ; learning rate: 0,9 ; n_estimators: 400	0,240	0,601
	max_depths: 25 ; learning rate: 0,1 ; n_estimators: 5	0,234	0,607
	max_depths: 25 ; learning rate: 0,1 ; n_estimators: 10	0,235	0,613
	max_depths: 25 ; learning rate: 0,1 ; n_estimators: 25	0,237	0,614
	max_depths: 25 ; learning rate: 0,1 ; n_estimators: 50	0,239	0,620

	max_depths: 25 ; learning rate: 0,1 ; n_estimators: 75	0,240	0,612
	max_depths: 25 ; learning rate: 0,1 ; n_estimators: 100	0,241	0,613
	max_depths: 25 ; learning rate: 0,1 ; n_estimators: 200	0,243	0,616
	max_depths: 25 ; learning rate: 0,1 ; n_estimators: 300	0,246	0,620
	max_depths: 25 ; learning rate: 0,1 ; n_estimators: 400	0,245	0,623
	max_depths: 25 ; learning rate: 0,5 ; n_estimators: 5	0,236	0,599
	max_depths: 25 ; learning rate: 0,5 ; n_estimators: 10	0,238	0,605
	max_depths: 25 ; learning rate: 0,5 ; n_estimators: 25	0,241	0,605
	max_depths: 25 ; learning rate: 0,5 ; n_estimators: 50	0,242	0,609
	max_depths: 25 ; learning rate: 0,5 ; n_estimators: 75	0,246	0,614
	max_depths: 25 ; learning rate: 0,5 ; n_estimators: 100	0,246	0,615
	max_depths: 25 ; learning rate: 0,5 ; n_estimators: 200	0,248	0,622
	max_depths: 25 ; learning rate: 0,5 ; n_estimators: 300	0,247	0,616
	max_depths: 25 ; learning rate: 0,5 ; n_estimators: 400	0,248	0,618
	max_depths: 25 ; learning rate: 0,9 ; n_estimators: 5	0,237	0,592
	max_depths: 25 ; learning rate: 0,9 ; n_estimators: 10	0,239	0,597
	max_depths: 25 ; learning rate: 0,9 ; n_estimators: 25	0,242	0,606
	max_depths: 25 ; learning rate: 0,9 ; n_estimators: 50	0,247	0,616
	max_depths: 25 ; learning rate: 0,9 ; n_estimators: 75	0,245	0,614
	max_depths: 25 ; learning rate: 0,9 ; n_estimators: 100	0,246	0,615
	max_depths: 25 ; learning rate: 0,9 ; n_estimators: 200	0,247	0,611
	max_depths: 25 ; learning rate: 0,9 ; n_estimators: 300	0,247	0,611
	max_depths: 25 ; learning rate: 0,9 ; n_estimators: 400	0,248	0,614

Table 7 - Hyperparameter Tuning

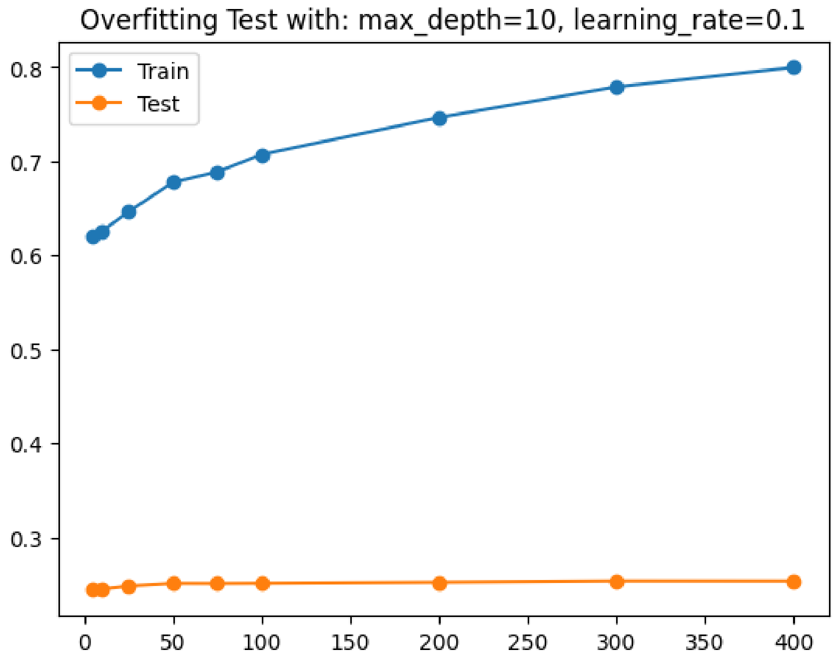


Image 18 - Overfitting Analysis for Gradient Boosting with max depth 10 and learning rate 0.1