# AI Hack

Technical Report

## How will developing low income neighbourhoods in Boston affect NOx levels?

**Mech Eng Defectors**

Yousef Nami

Kyriacos Theocharides

**Submitted to:**

Xing Liu

Imperial College London

February 2021¶

# Executive Summary

The Boston Housing Market dataset is ubiquitous but imperfect: with problems like small size, inconsistent definitions, incorrect coordinates and many many. However, it is still a very rich dataset containing informative geographical information, powerful socioeconomic indicators, and continuous levels of Nitrogen Oxides (NOx).

This report explores the effect of developing low income neighbourhoods on NOx. This involves three logical steps: 1) Verifying that the dataset is rich enough to form clusters of economic class, 2) train a regressor for predicting NOx values, and finally 3) creating synthetic data simulating 'improved' low income neighbourhoods by bootstrapping values from higher income classes, while keeping geographical constraints fixed.

The evidence suggests that improving low income neighbourhoods does indeed decrease overall NOx levels, giving non-humanitarian reasons for supporting social uplifting policy. This project also corrects erroneous longitude and latitude values of the Boston dataset using Google's geocoder API. The code and documentation for this project can be found here.

# 1. Introduction

Wealth inequality is, and has been, a problem that is yet to be solved and getting bigger. This is of particular issue in Boston, Massachusetts where there exist large disparities in wealth across towns due to historical segmentation of neighbourhoods, known as red-lining. Therefore the question of what policy decisions to make to uplift low income neighbourhoods is always a topic of discussion.

The democratisation of Machine Learning techniques has provided people with incentives to collaborate on solving real life problems openly-sourced data. An example pertaining to Boston is the Boston Housing Market Dataset: ubiquitous though rarely ever used for reasons beyond predicting house prices or exploring the geospatial relationships that exist between towns using unsupervised learning. To date, there have been low to no attempts at using this dataset to derive insight has political ramifications.

This is partly due to limitations posed by the dataset, namely:

1) Median value of owner occupied homes is capped at 50k dollars
2) The dataset is relatively small, with around circa 500 data points
3) The dataset does not have any clear features that can be used for direct policy recommendations without extensive exploration

Despite this, the dataset is quite rich and contains previously unexplored avenues. One such avenue is how Nitrogen Oxides (NOx) — a collection of chemicals harmful to humans and to the environment — are distributed across Boston and how it is affected by socioeconomic factors.

This clearly has important policy implications: would uplifting these low income neighbourhoods increase the NOx levels to unreasonable amounts? The overarching research question then becomes: **How does developing low income neighbourhoods affect NOx levels?**

It's important to note that answering this question can be split into three logical steps:

1) Proving that the dataset is informative enough to discriminate between low to high income neighbourhoods
2) Finding a method for predicting NOx levels based on features from the dataset
3) Finding a way 'developing' a low income neighbouring (for example: synthetically creating a new dataset that represents 'developed' low income neighbourhoods), and then feeding the new data back into the predictor to compare with the original dataset

These three steps were deemed to be project objectives.

The structure of the report is as follows: first a brief account of the management of the project is given in Section 2. This is later followed by a section on Data Exploration, where the data used for the project is outlined, preliminary research and analysis highlighted. This section also shows that the research question can indeed be answered with the data collected. The next section highlights the methodology of the project and key results obtained. This is followed by an evaluation and discussion, where key insights from the research are delivered along with implications and policy recommendations. An evaluation of the project as a whole is also provided. In the final section, a concise conclusion to the project is given, and future work and recommendations are highlighted.

## 2. Project Management and Planning

Despite the tight time schedule, the team spent the first couple of hours defining the project well to ensure an efficient use of time. To do this, the team took inspiration from John Rollins' methodology.

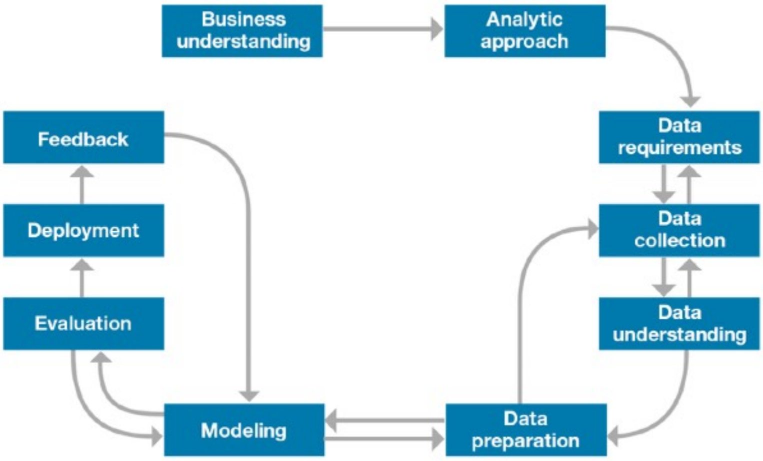


Figure 2.1: John Rollins' data science methodology

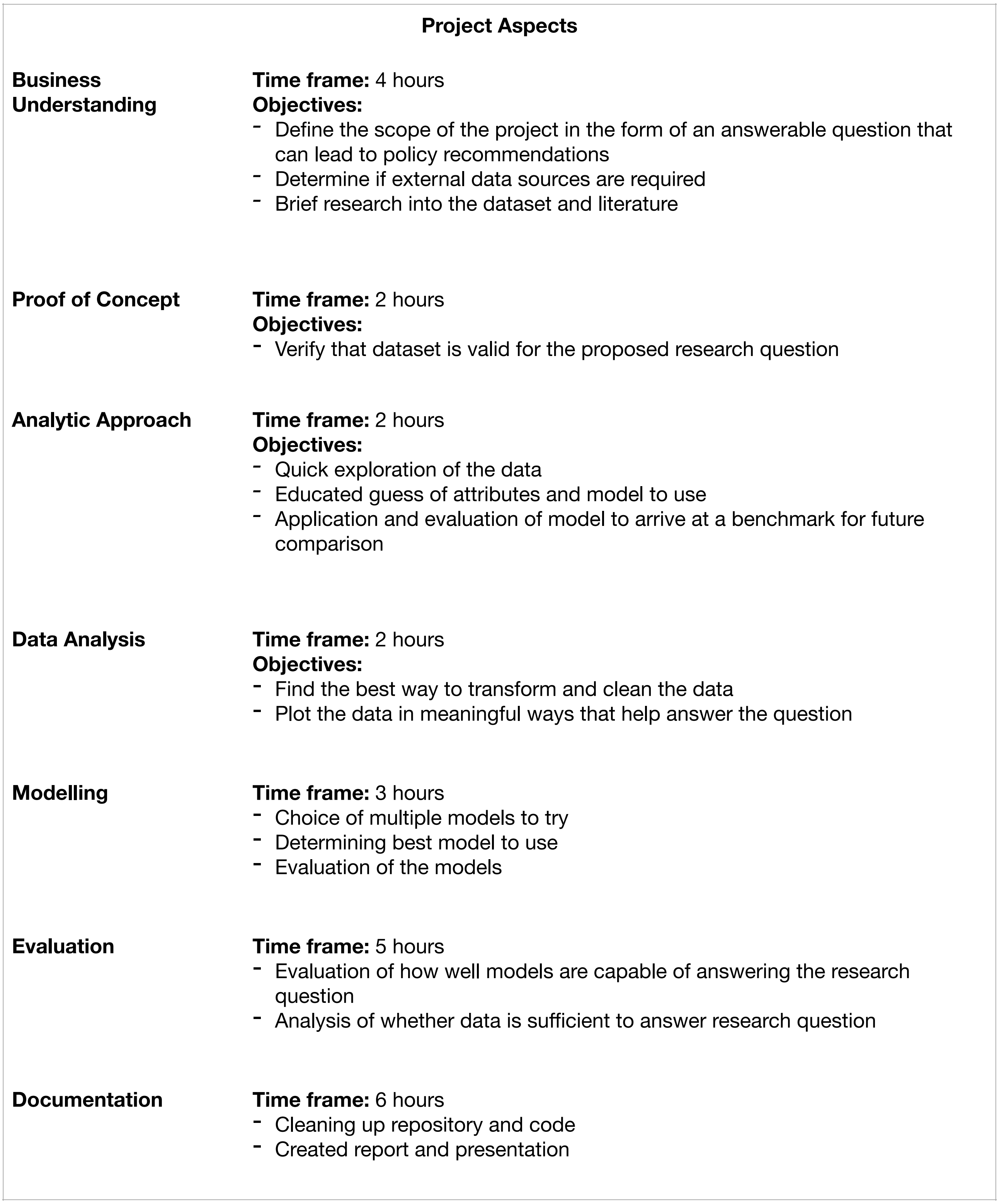
An adjusted version of the methodology (adjusted to the AI Hack's scale and time frame) was created.

It's worth highlighting that the project was highly iterative, to ensure that, at every stage, all of the assumptions were met and valid. This meant that the team would reconvene every hour to monitor progress and see if the results are as expected.

The team decided to use GitHub as a central place to keep track of development. Issues were used as a project management tool, as well as for highlighting enhancements or bugs. A link to the repository showing the work done can be found here.

Most of the work was conducted through Jupyter Notebooks.

Table 2.1: a table showing a high level plan of the project based on the John Rollins' methodology



| Project Aspects | |
|---|---|
| **Business Understanding** | **Time frame:** 4 hours<br>**Objectives:**<br>- Define the scope of the project in the form of an answerable question that can lead to policy recommendations<br>- Determine if external data sources are required<br>- Brief research into the dataset and literature |
| **Proof of Concept** | **Time frame:** 2 hours<br>**Objectives:**<br>- Verify that dataset is valid for the proposed research question |
| **Analytic Approach** | **Time frame:** 2 hours<br>**Objectives:**<br>- Quick exploration of the data<br>- Educated guess of attributes and model to use<br>- Application and evaluation of model to arrive at a benchmark for future comparison |
| **Data Analysis** | **Time frame:** 2 hours<br>**Objectives:**<br>- Find the best way to transform and clean the data<br>- Plot the data in meaningful ways that help answer the question |
| **Modelling** | **Time frame:** 3 hours<br>- Choice of multiple models to try<br>- Determining best model to use<br>- Evaluation of the models |
| **Evaluation** | **Time frame:** 5 hours<br>- Evaluation of how well models are capable of answering the research question<br>- Analysis of whether data is sufficient to answer research question |
| **Documentation** | **Time frame:** 6 hours<br>- Cleaning up repository and code<br>- Created report and presentation |

## 3. Data Exploration

The Boston dataset is almost ubiquitous and has been routinely used to test new models in practice. The original dataset is comprised of 14 attributes, but there exist corrected ones that add features such as latitude and longitude. Before verifying that the dataset is valid to answer the research question, a correlation matrix for all continuous and ordinal values was created. Fig. 3.1 below shows this.

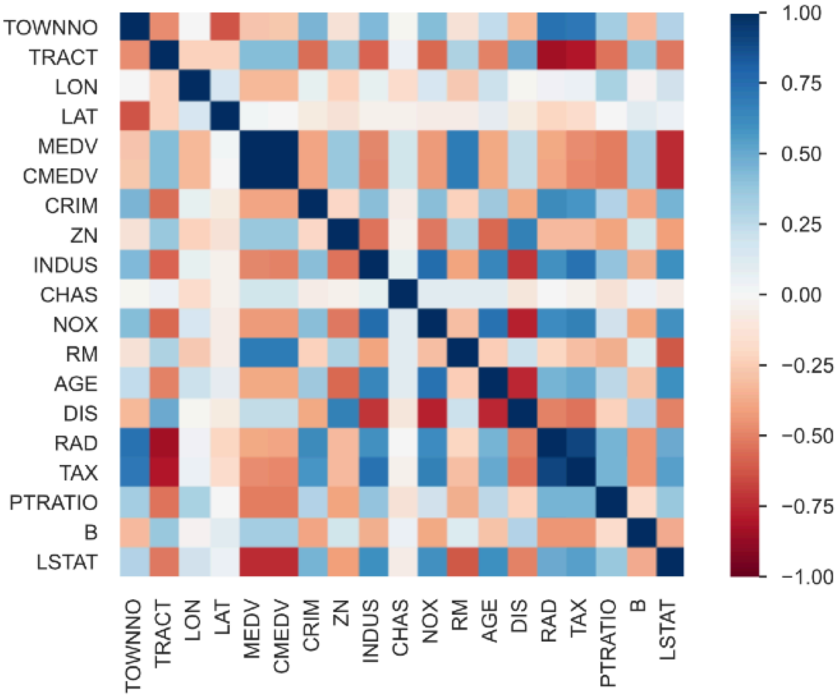


Figure 3.1: a correlation matrix created using the library `pandas-profiling`

From this, it is worth noting the following:

- There seems to be no correlation between latitudes and longitudes and NOx emissions
- 'AGE', 'DIS', 'RAD' are all highly correlated with NOx emissions (this makes sense: older houses produce more NOx through chimneys, houses further from employment centres are in less industrial towns, roads)
- 'CHAS' is not correlated with anything
- This correlation matrix was used to choose the first features whenever modelling

A closer inspection of the latitudes and longitudes using `folium` showed that these are in fact incorrect, as much of the items appear in water. Seeing as fixing this was central to this project (in terms of visualisation and verification), these values were calculated correctly using Google's `geocoder` API.

As mentioned previously, the first step in our analysis was to determine whether the dataset contained enough data to be able to distinguish the languishing low income areas from the more affluent towns. Immediately, the attribute 'LSTAT' (a numeric vector of percentage values of lower status population) stood out as a strong indicator for socioeconomic disparity. One option would have been to perform some background research on socioeconomic classes and attempt to define boundaries based on LSTAT alone. However, we did not think this was a sensible avenue as the data on LSTAT may be subjective and unrepresentative as some areas may be considered 'run down' despite not having many lower income people living there (e.g industrial estates).

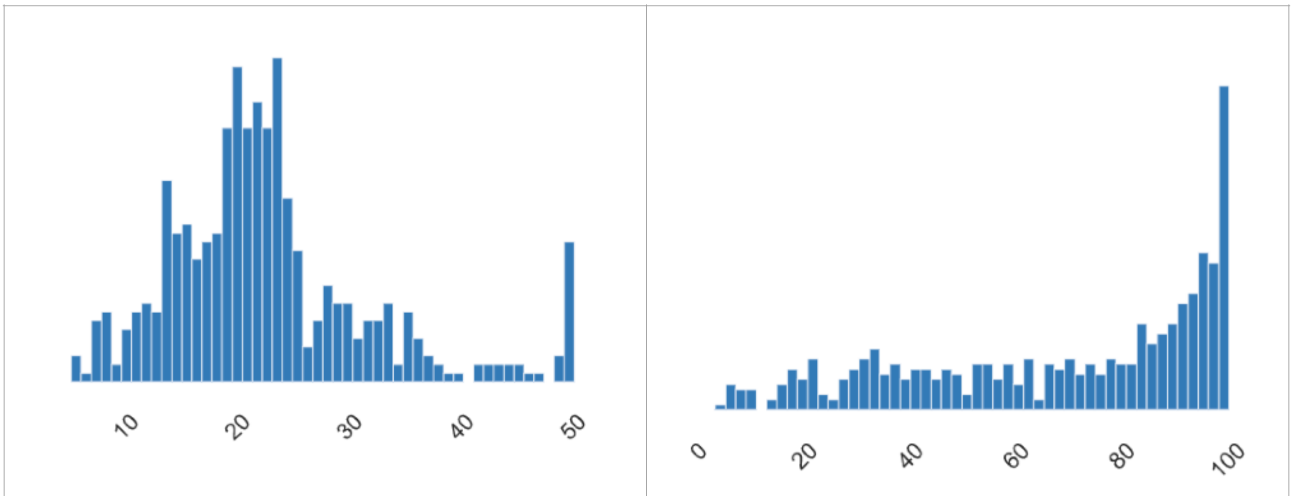
Instead, we opted for a k-means clustering algorithm which would define the socioeconomic classes for us. Given k-means simplicity, it is the obvious choice for a first model to choose. The question now was : "Which variables should we use that are good indicators of socioeconomic class?". `pandas` produced histograms of the spread of each variable as well as descriptive statistics. We ended up using the following 4 variables:

- 'CMEDV' - a numeric vector of corrected median values of owner-occupied housing in USD 1000
- 'INDUS' - a numeric vector of proportions of non-retail business acres per town (constant for all Boston tracts)
- 'AGE' - a numeric vector of proportions of owner-occupied units built prior to 1940

- 'LSTAT' - a numeric vector of percentage values of lower status population

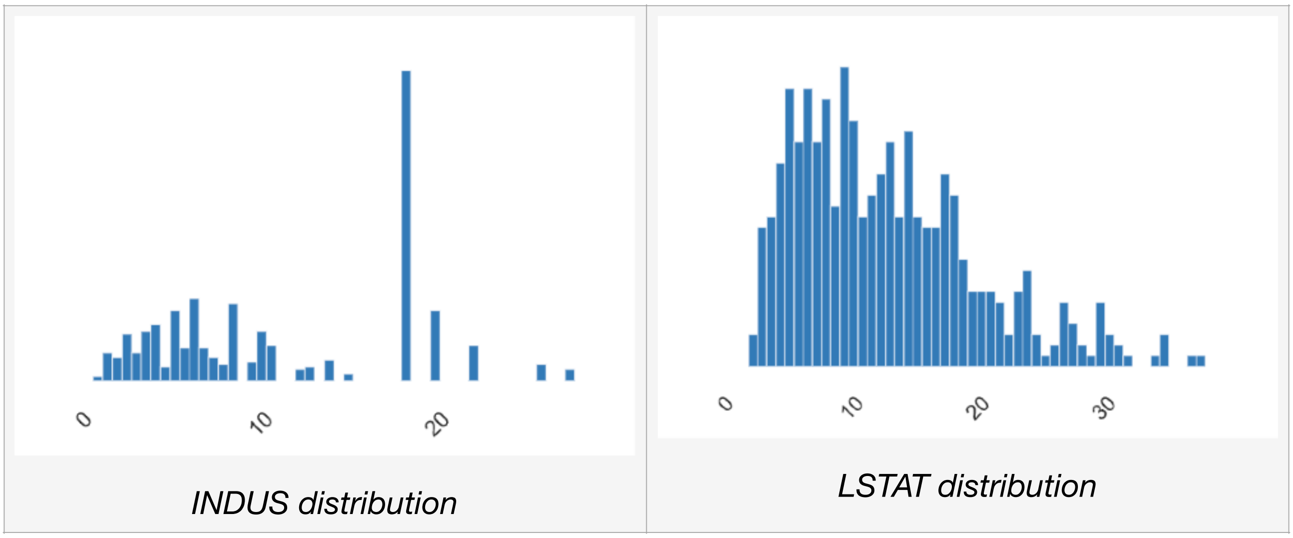
'CMEDV' was selected as people of lower socioeconomic class will almost certainly be earning salaries on the lower end of the spectrum. 'INDUS' would most likely refer to factories and other 'blue collar' workplaces which are usually dominated by the working class. Lastly, we assumed that people of poorer backgrounds would be living in older accommodation as richer citizens would prefer to live in more modern and spacious housing. The one caveat with 'AGE' is that there may be a proportion of affluent citizens who live in older homes which have been extensively renovated, but we assumed that if this was a factor it would be a minority and the overwhelming majority of citizens living in older housing would be of poorer backgrounds.

Fig 3.2 below shows the distributions of each of our variables used for clustering.



*CMEDV distribution*

*AGE distribution*



*INDUS distribution*

*LSTAT distribution*

Figure 3.2: initial distortions of the 4 features used for k-means (before scaling), created using `pandas-profiling`

One may wonder why we did not include 'CRIM' - the crime rate, as one of our variables. Firstly, 'CRIM' does not correlate as strongly with 'LSTAT' as INDUS does for example, according to the correlation matrix perhaps because crime is loosely defined. While poorer areas will experience more violent crimes, richer areas are more likely to experience thefts such as pickpocketing and burglaries, so we did not believe that crime rates would be a clear indication of low socioeconomic class. Moreover, the data for 'CRIM' is the most skewed of all the variables in the dataset, which could pose problems for the k-means clustering algorithm even after transforming it. As for the 4 aforementioned variables, these were transformed by first square rooting all the values and using `PowerTransformer()` from the `sklearn.preprocessing` module, with the results shown below in Figure 3.3.

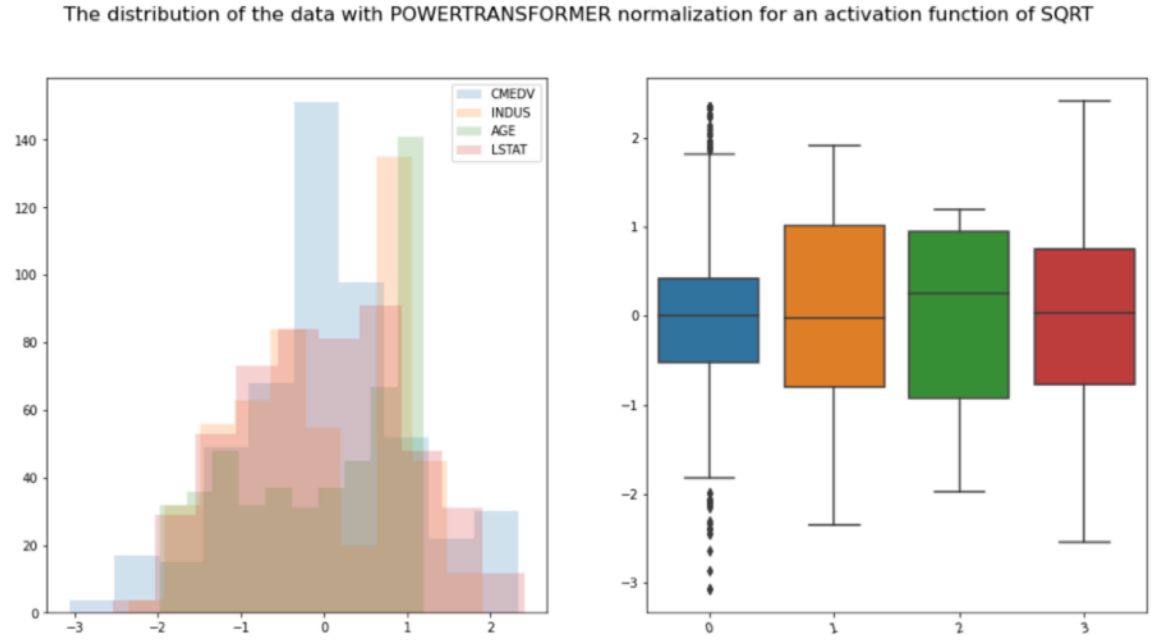


Figure 3.3: boxplot and histograms of the four features after a SQRT transformation and then normalised using `PowerTransformer()`. The plots are made using `matplotlib` and `seaborn` respectively.

This combination was picked based on an optimiser that minimises the kurtosis and skewness of the distributions. It should be noted that although there are outliers for the leftmost box plot (which shows CMEDV), we decided to keep these as they are symmetrical and because our dataset is small, so we did not want to start omitting values. Besides, it is not always wise to remove outliers when it isn't clear why these are outliers (see here).

To decide on the number of clusters to use, an elbow plot was employed which shows the decrease in squared error with increasing cluster number. The idea is to use a cluster number which shows the greatest decrease in squared error, shown by the edge or 'elbow' in Figure 3.4 below:

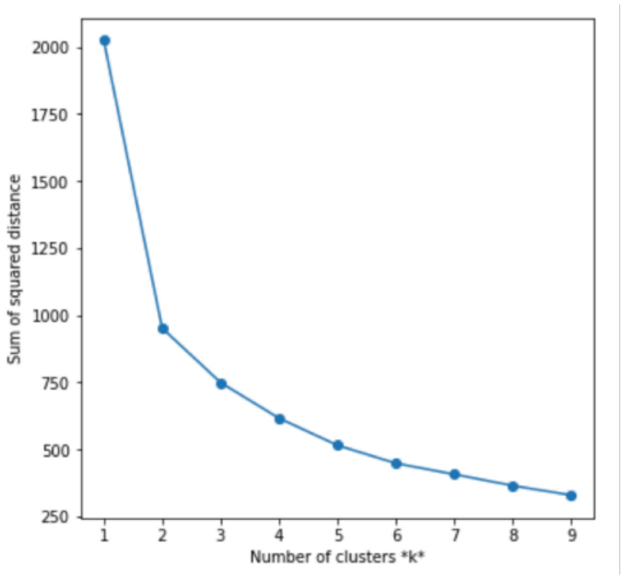
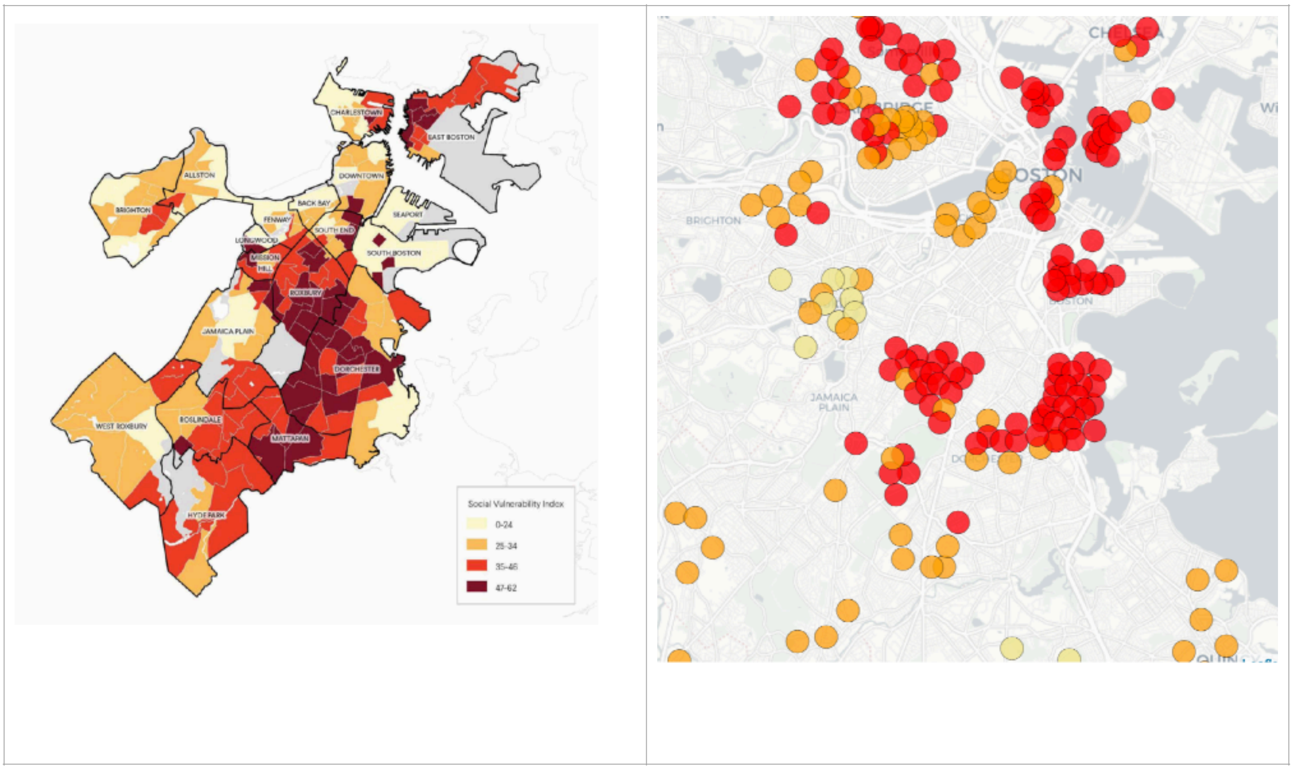


Figure 3.4: an elbow plot

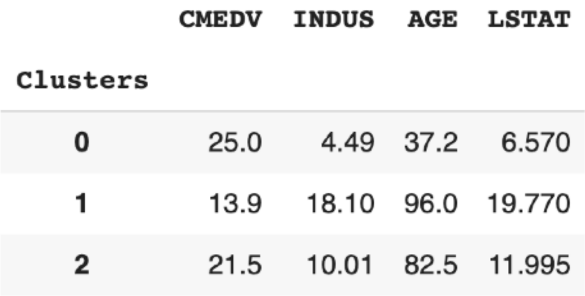
Although the plots shows 2 to be the optimum number, we found that 3 produced the best results on our `folium` plot, as 2 clusters could not differentiate the extremely poor areas from the rest. Besides, the rule of thumb is to have 3 socioeconomic classes (higher, middle, lower), so we mimicked this approach. Our results from the clustering are shown below in Figure 3.5, along with an image of a similar study conducted by Boston University on Social Equity. Our results seemed reasonable as the average values of clusters seemed to correspond with our assumptions based on the correlation matrix. Looking at the figure below, it is apparent that Cluster 1 (red) is the lowest socioeconomic class, 2 (orange) is a middle class and 0 (yellow) the highest.

Figure 3.5: (left) Boston University choropleth and (right) our clusters with the labels. On close inspection once can see that much of the neighbourhoods match



This comparison gave us confidence that our k-means clustering algorithm produced an accurate representation of the socioeconomic divide of the Boston area for our dataset[1].

Table 3.1: showing the median characteristics of each cluster (solely the features used to predict it)



| | CMEDV | INDUS | AGE | LSTAT |
|---|---|---|---|---|
| **Clusters** | | | | |
| **0** | 25.0 | 4.49 | 37.2 | 6.570 |
| **1** | 13.9 | 18.10 | 96.0 | 19.770 |
| **2** | 21.5 | 10.01 | 82.5 | 11.995 |

[1] we could not expect a perfect match, nor could we normally verify an unsupervised problem. However, in this context, this means that our data is representative enough to be able to give some indication of socioeconomic class + randomness due to the dataset itself

# 4. Modelling and Results

The second objective of the project was to find a predictor of NOx given data points. Since NOx is a continuous variable, regression was deemed appropriate. The method relied on a first good estimate for which parameters to use, based on the the correlation matrix as well as intuition. The next step involved trying out different regression models to see what works best. The flowchart in Fig 4.1. explains this process.

In fact, the best model turned out to be the SVR. This is expected due to the fact that support vector machines can capture non-linear relationships quite well, but also because one of the regressors, ‘RAD’, is an ordinal variable, and therefore the ‘distance’ measurements that a normal regression assumes would be somewhat invalid. A support vector method however is able to capture this in data.

A plot showing the output of the SVR plotted on top of that of the original data for 4 of the 5 regressor is shown. Seeing as this project places more emphasis on actual prediction as opposed to explainability of regressors, it was deemed that the SVR would be appropriate. A summary of the best three models are provided in Table 4.1 along with hyperparameters.

It’s worth noting that a neural model was also attempted, but only to see if it could exceed the accuracy of the SVR. Only a brief amount of time was spent on this, since it was deemed that the SVR was sufficient given its simplicity and relatively high accuracy.

Table 4.1: summary of top 3 models

| Model | R2 | R2-Adj | MAE | MSE |
|---|---|---|---|---|
| `SVR(kernel = ‘rbf’, epsilon = 0.1, C = 12)` | 0.885 | 0.879 | 0.153 | 0.043 |
| `LinearRegression()` | 0.811 | 0.801 | 0.203 | 0.069 |
| `Ridge(alpha = 0.001)` | 0.812 | 0.802 | 0.203 | 0.070 |

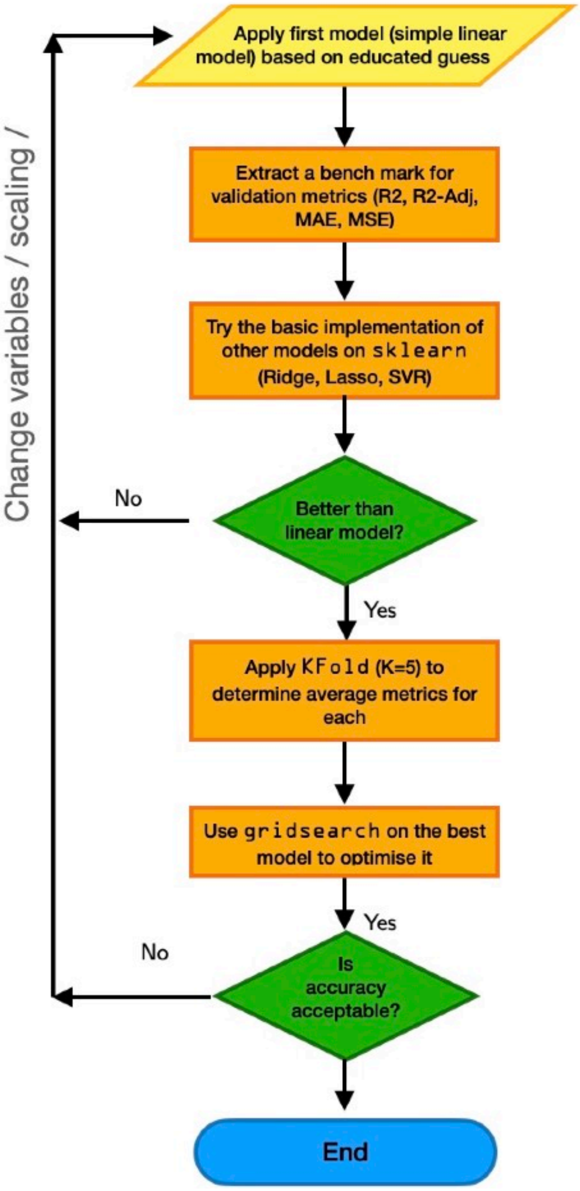

Change variables / scaling /
Apply first model (simple linear model) based on educated guess
Extract a bench mark for validation metrics (R2, R2-Adj, MAE, MSE)
Try the basic implementation of other models on sklearn (Ridge, Lasso, SVR)
No
Better than linear model?
Yes
Apply KFold (K=5) to determine average metrics for each
Use gridsearch on the best model to optimise it
Yes
No
Is accuracy acceptable?
End

Figure 4.1: a flowchart of the methodology for finding the best model

At first, the two regression models were to be compared using AIC, but seeing the superiority of SVR in terms of accuracy, it was determined that that would be superfluous. With this found, it was time to verify that the errors (residuals) are normally distributed. This assumption is necessary in order to use bootstrapping (see Section 5). As a result, the residuals were calculated and drawn on a QQ-plot. The result for the SVR is shown in the Figure 4.3.

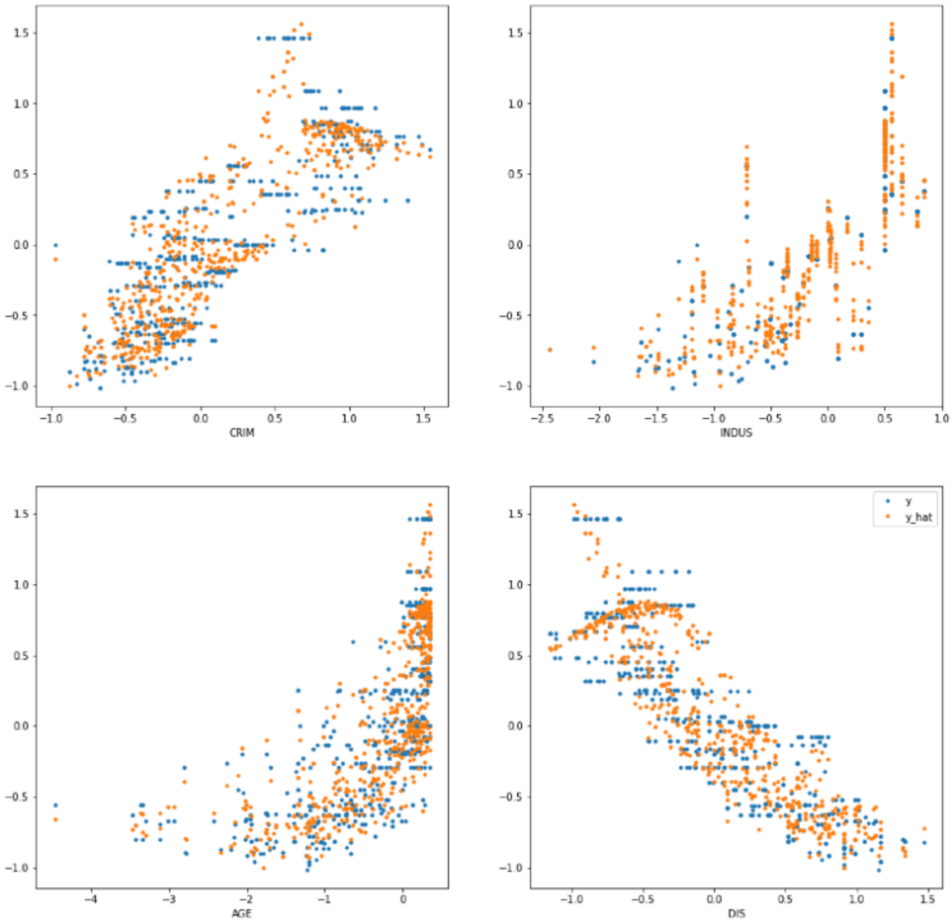


Figure 4.2: a graph of 4/5 regressors of the SVR plotted on top of the original dataset

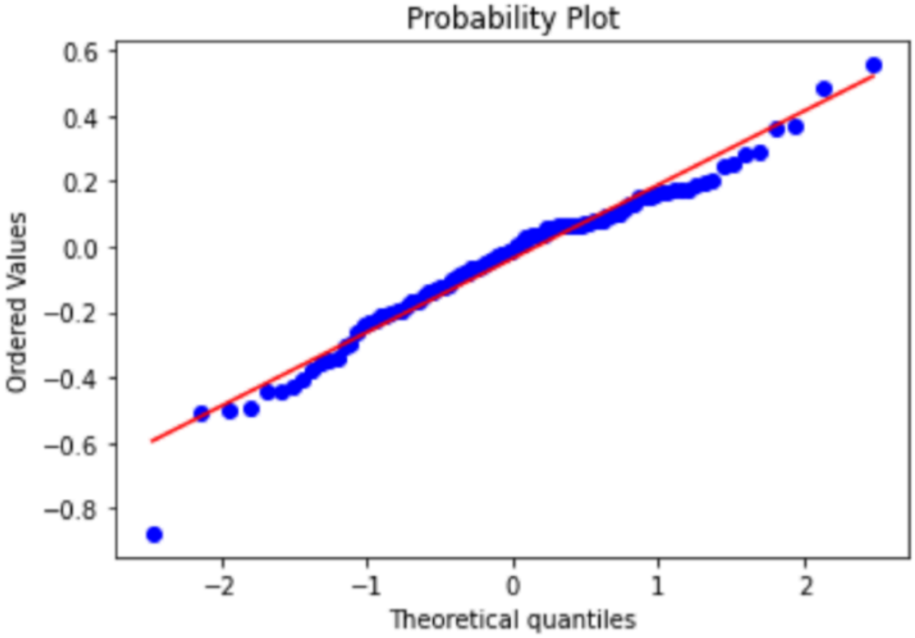


Figure 4.3: a QQ plot of the residuals of the y_hat from the SVR

As we can see from the plot, there is no indication that the errors are not normally distributed. It's also worth noting that the errors are homoskedastic.

In addition to this, a partial dependence analysis was conducted to measure the impact of varying parameters parameters while keeping all else equal. In this case, the regressors are: ['CRIM', 'DIS', 'AGE', 'INDUS', 'RAD']. We are not interested in varying the parameters that are fixed by geography, but rather the parameters that have policy impact. These would be 'AGE" and 'CRIM'.

Partial dependence is essentially a way of 'fixing' all other parameters while the parameter of interest (whose impact we want to measure) is being changed. Mathematically this would be something like this:

```
all_parameters = [
    [parameter_of_interest] + [*other_attributes]
]
```

To 'fix' other attributes, one might be tempted to use the average. However, this does not capture the variation across all datapoint. As a result, for each parameter_of_interest, another for loop is created where parameter_of_interest remains constant, but the other attributes are summed across all instances in the dataset. Normally, one could easily apply this by using plot_partial_difference from sklearn. However, since we were interested in calculating these changes solely across the low income neighbourhoods, but looping through all the range of possible values, we had to create this function ourselves. The result of the partial difference are shown below in Figure 4.4.

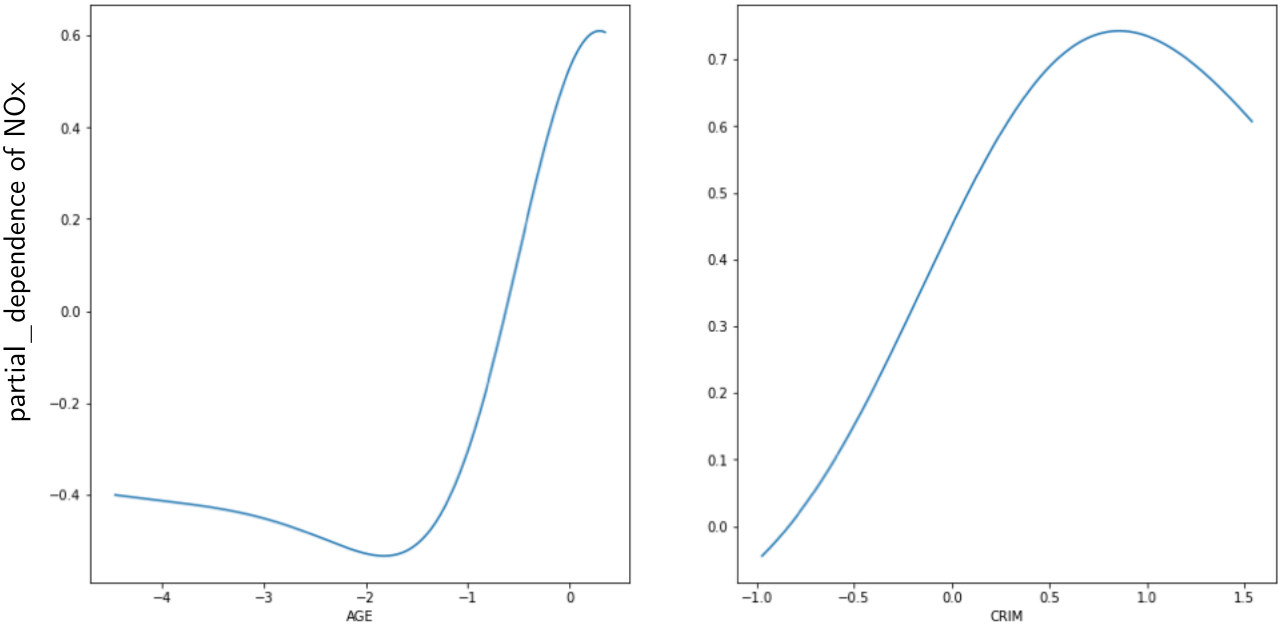


Figure 4.4: a plot of the partial difference for the two non-geographical regressor

These results are both interesting and confusing. The one for 'AGE' is mostly logical: an increase in house AGE means older infrastructure for heating, and thus more particulates. The one for CRIM is confusing, it somewhat seems to suggest, all other things equal, that increasing crime will increase the NOx levels. A notion that is rather hard to believe on it's own without checking for other dependencies.¶

# 5. Evaluation and Discussion

In order to answer the question "How will developing low income neighbourhoods in Boston affect NOx levels?" a measure for development needs to be defined. For this we propose that each datapoint that regressors comprised of geographically binding and non-geographically binding (e.g. socio-economic) features. As such, each item in the data set can be represented as:

```
item = [
    [*geographical_data] + [*non-geographical_data]
]
```

That said, one could, replace the non-geographical attributes with that sampled from bootstrapped medium / high income datapoint. Essentially, what this means is that: there are certain geographic factors that city planners and policy makers physically cannot change that correlate with highly with NOx. However, by modifying the 'non-constrained' variables, such as AGE of houses[2] and crime rate, one could **simulate** what impacts developing a neighbourhood would have on NOx. Figure 5.1 highlights this nicely:

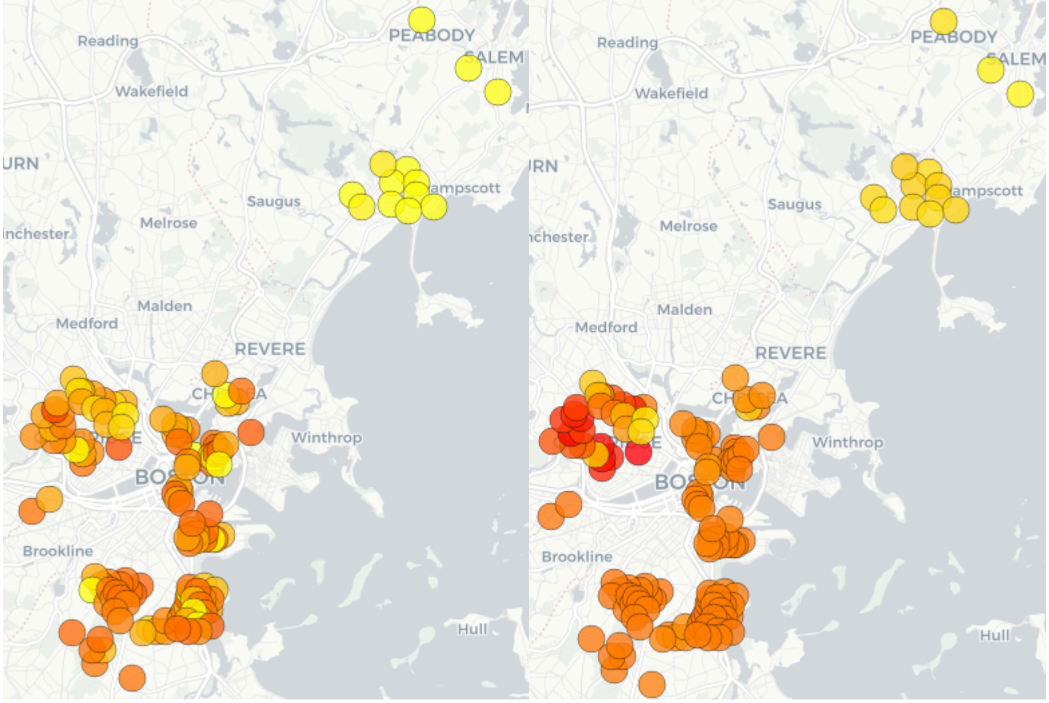


Figure 5.1: (left) the low income towns after development (by using bootstrapped non-geographical parameters from high income areas), where the color of the circles represents the intensity of NOx emissions and (high ) the low income towns without development (original dataset)

The disparity is clear: the results of this analysis suggest that there is incentive to uplift low income towns because it would potentially decrease NOx levels. This provides even non-humanitarian reasons to pursue this initiative. Figure 5.2 below shows the effect that uplifting low income neighbourhoods by bootstrapping from medium and high income areas has on the distributions. It would appear that the peak of the distribution remains the same, with little change, but that the data is more dispersed towards the left, leading to an overall positive skew. In terms of other recommendations: INDUS explains NOx quite a bit. A recommendation for governments could be encourage industries to move

[2] By means of improving infrastructure for instance

out of cities, and to build in more rural areas (as opposed to the city centre), since this affects both congestion and overall pollution density. However, more research needs to be done in determining the socioeconomic impact that this will have on the low income neighbourhoods, as many residents may lose their jobs as a result.

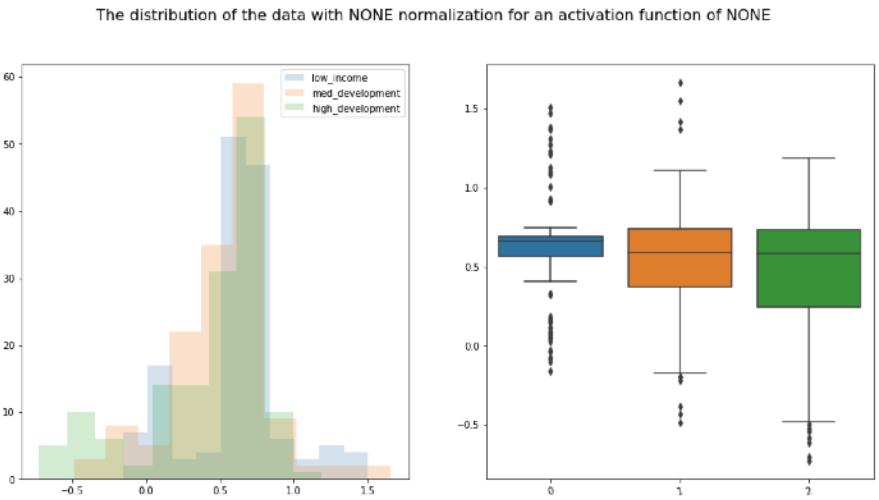


Figure 5.2: box plot (right) of the 3 datasets and histograms (left)

However, it's worth noting that there are problems with the dataset, that impact the effectiveness of this study. In addition to the reasons mentioned in the introduction, this dataset is weakened by the fact that some of its attributes have surprising correlations that don't make sense. In addition to this, the geographical parameter 'CHAS' has no effect on anything, while intuitively one would think so, and he literature seems to suggest so too. Another interesting thing to note is that the dataset is quite thin… much of the affluent neighbourhoods fall in the outskirts of the city. Though this appears to agree with literature, the dataset has imbalanced classes.

It's interesting to look at some probabilities based on the models and existing online thresholds. Based on online standards, California's max NOx levels (parts per 10 million) are 0.3, and the federal government sets the limit at 0.53. If California's limit is to be taken as the threshold low NOx, and the government's for high NOx, this means that 52% of the dataset has NOx levels that are too high., none which are actually low. Another interesting insight about the disparity… given a low income neighbourhood, the probability of having a high NOx rating is 96.6%. The table below summarises this information nicely:

Table 5.1: proportions of low, medium, high NOx levels (based on online standards) against low, medium high income towns

| Neighbourhood \ NOx | Low | Medium | High |
|---|---|---|---|
| **High** | 0 | 5 | 142 |
| **Medium** | 0 | 57 | 109 |
| **Low** | 0 | 182 | 11 |

## 6. Conclusion and Future Work

This project's goal was to explore how developing low income neighbourhoods affects NOXs levels in order to provide insight into policy making. This goal was broken down into three logical steps: 1) determining if the dataset is sufficient for the task at hand, 2) finding a model for predicting NOx levels across any data point, 3) determine the impact of developing a low income neighbourhood while keeping the geographical constraints constant. In order to verify the dataset's sufficiency, K-means clustering was used for finding 3 distinct clusters (corresponding to low, medium and high income neighbourhoods). This outcome was verified against past data showing similar distributions, meaning that the data is good enough to reasonably cluster neighbourhoods into three classes.

The second aspect of the project applied regression techniques to determine NOx emissions using 5 regressors, namely AGE, DIS, INDUS, CRIM, and RAD. This aspect found SVR to be the best predictor, giving an accuracy of 88%. The normality of the residuals was verified through QQ plots, which then allowed for a bootstrapped sample of high income neighbourhood data points to be created. Keeping geography fixed (i.e. INDUS, DIS and RAD), improvable[3] from the low neighbourhood dataset were replaced (i.e. AGE, CRIM) with those from the high income neighbourhood. It was found that caused an overall decrease in the NOx levels. The statistical effect is that the distribution of NOx in the low income neighbourhoods doesn't change its peak. but has its surroundings dispersed to the left, causing an overall positive skew as development increases. Based on the research, it appears that CRIM and AGE are strong humanly-changeable indicators of NOx pollution. The evidence suggests that improving low income neighbourhoods correlates with lower NOx values.

Future work would do well to consider the following: 1) **causality:** could you explore if AGE / CRIM can be deemed to cause, 2) **data augmentation:** this work did much in terms of augmenting the longitude and latitudes, and creating synthetic data for 'developed' low income neighbourhoods, future methods could look to using neural network driven data augmentation techniques (such as GANs).

[3] These are things that can be developed without radical changes to industry. This includes improving the condition of older houses (i.e. AGE) or decreasing the crime rate in an area (i.e. CRIM). You cannot for instance change 'DIS' so easily.