MathWorks Math Modeling Challenge 2024

Watford Grammar School for Boys

Team #17870, Hertfordshire, England

Coach: Yachna Tailor

Students: Dominic De Jonge, Meyer Louka, Neil Nair, Jakub Skop, Kshitij Tyagi

M3 Challenge TECHNICAL COMPUTING THIRD PLACE—$1,000 Team Prize

JUDGE COMMENTS

Specifically for Team #17870—Submitted at the close of triage judging

COMMENT 1: I like how you evaluated different models to determine the best one to use.

COMMENT 2: The executive summary is meant to include discussion of your models and how they help to answer the questions being posed, and your report does well with this. It is important that mathematical models have formulas that can provide insight into how the different terms influence the model to promote policy changes. Your answer to Question 1, where it is broken up into different housing unit types is a very good example of how a more sophisticated model can identify useful details. You do a good job of explaining more sophisticated statistical techniques that allow you to check for validity.

MathWorks Math Modeling Challenge
A program of SIAM

***Note: This cover sheet has been added by SIAM to identify the winning team after judging was completed. Any identifying information other than team # on a MathWorks Math Modeling Challenge submission is a rules violation. Further, this paper is posted exactly as submitted to M3 Challenge. Typos, odd formatting, or other mistakes may be attributed to the 14-hour time constraint.

A Tale of Two Crises: *The Housing Shortage and Homelessness*

March 2, 2024

Executive Summary

To Minister of State for Housing and Planning,

In the wake of the rising cost of living crisis, the pressing issue of housing affordability and homelessness, are critical challenges facing our society today. For decades, the cost of housing in both the United States and the United Kingdom has outpaced income growth, placing immense economic strain on families and individuals across our nations. This disparity has led to a situation where many struggle to afford basic shelter. Furthermore, the shortage of available housing has reached crisis levels, exacerbating the problem of homelessness. Despite living in some of the world's wealthiest societies, homelessness remains a persistent problem.

We predicted the housing supply in cities in the UK in 2034, 2044 and 2074 using an amalgamation of multiple uni-variate logistic regression models. Using definition for 'Housing Supply' provided by the UK government[1], and by regressing different sorts of accommodations, we were able to predict that in Brighton, 293 new dwellings in arise in 2034, 207 in 2044, and 67 in 2074. In Manchester, 1407 in 2034, 1594 in 2044 and 2369 in 2074. These trends are able to show that Manchester is on a positive growth trajectory and Brighton is nearing a plateau in its urban growth, by this metric. We believe these could be important factors to consider when developing strategies to mitigate large-scale homelessness.

To further reinforce our study, we followed with a consideration of how homelessness will rise over the same spans of time, and in the same UK locations. Our approach was to gather predictor variables, and create a two-pronged approach: First, use those predictor variables to determine the amount of homeless people at a given point in time, and secondly, extrapolate those predictor variables into the future using a regression algorithm. We dealt with multi-colinearity and over fitting in our data by using ridge regression, and achieved the following figures: In Manchester, 3,474 in 2034, 4,536 in 2044 and 8,657 in 2074, while in Brighton, 1,021 in 2034, 1,176 in 2044 and 1,643 in 2074. These figures are quite startling, and we suggest an immediate study into the efficacy of various government schemes in tackling these numbers.

To corroborate with your findings, we have conducted a smaller scale investigation, examining 3 different actionable schemes (Building new flats, increasing the minimum wage and increasing rent on social houses) and also the effect of migration on our homeless population, through a combination of various simple techniques - this should provide a good starting point for a more in-depth future study into these issues. However, all policies show promise and have varying cost-to-benefit ratios, with increased social rent demonstrating a potential halving in homelessness.

Contents

1. Part I: It Was the Best of Times

1.1. Defining the Problem

This problem challenged us to create a model that predicts changes in the housing supply in the two U.K. regions, Manchester and Brighton and Hove in the next 10, 20, and 50 years. Indicate your level of confidence in your predictions.

1.2. Assumptions and Justifications

1.1. **The 'unknown' number of houses do not fall in a category of house type.**
Justification: It would be impossible to account for since the distribution of 'unknown' houses is unknown. However, the trend in this feature could indicate that methods of house-type registration have improved over time. Therefore, the limitations of poor data collection will be, in the long term mitigated, and there is no need to adjust the proportions of data from the deep past to account for this discrepancy.

1.2. **Data from 'Annexes' and 'Caravan/ Houseboat/ Mobile Home' are negligible**
Justification: The overwhelming majority of households in the UK do not fall into these categories. Thus, when considering the holistic growth in supply, these factors may be ignored.

1.3. **Time is the only necessary parameter for projecting the growth in the number of households in each household type.**
Justification: Time series forecasting requires finding a time-dependent function from which results can be drawn. In reality, a number of external influences may exist for each feature. Under our assumption, each of these factors is assumed to depend on time, such that, the sum of the time derivatives of each influence is equal to the time derivative of the model for the feature considered. Hence observing the trend of the number of households in each household type, negates any need to consider influences on these factors when developing the model

1.4. **The government will not create additional drastic legislation on the purchase of households in the timescale being considered.**
Justification: Such regulations would have a large impact on housing supply. In order to develop a concise model, these unpredictable changes cannot be accounted for. It is natural that some legislation will be implemented in the next 50 years either directly or indirectly pertaining to housing supply, however, any significant legislation can not be speculated and therefore must be ignored for the purposes of our model.

1.5. **The is no time lag between the creation of a new household and its addition to the market.**
Justification: Although net additions typically take some lag time to appear on the market, all net additions are most likely be in the housing market within the same year. Since our data is discretised by year, the time lag will be trivial in predicting 'Housing Supply'.

1.6. **Certain features are assumed to follow a logistic trend.**
Justification: It stands to reason the population over time will plateau and a stable housing arrangement will be reached for certain features

1.7. **The data given is the total number of houses of each type in a UK city, regardless of status of occupancy.**
Justification: The time-series data collected by the UK government is postcode dependent,

which doesn't require a permanent occupant. Thus the number of houses of each type can be considered to be a total, including both occupied (owned) and vacant houses.

1.3. Defining Parameters

Type	Symbol	Definition	Units
Time	t	Years after 1993	Years
Number	x_i	Number of housing units of a particular type	Unitless
Number	$\mathbf{x}$	Total number of housing units	Unitless
Number	L_i	Carrying capacity of housing units of a particular type	Unitless
Number	k_i	Intrinsic exponential growth rate of housing units of a particular type	Unitless
Number	$\mathbf{y}$	The 'Housing Supply' for a particular year	Unitless

Table 1.1: Summary of Problem 1 Variables & Constant Parameters

As shown in the table in Table 1.1

1.4. Model Development

1.4.1 Logistic Model

The yearly increase in 'Housing supply' is defined as the net additions to dwelling statistics per year in the UK, according to UK government[1]. Thus, there may exist a function $\mathbf{x}(t)$, describing the number of housing units at time t, the number of years after 1993, where x_i represents the number of housing units of a particular type.

$$\mathbf{x}(t) = \sum_{i \in A} x_i(t)$$

The model employed is a sum of multiple uni-variate logistic regression models with respect to each distinct accommodation type, along with a piece-wise logistic regression for the 'Flat / Maisonette' type.

Figure 1.1 presents as a linear relationship, however, issues may arise when considering this simple a model.

- Homoscedasticity - a condition whereby the variance of the error term (calculated by taking the squared distance between data point and predicted value) is approximately constant.
- Failure to meet this condition would suggest the model may be inadequate in explaining the trend seen in the predictor variable, resulting in a weaker model for the long term.[2]

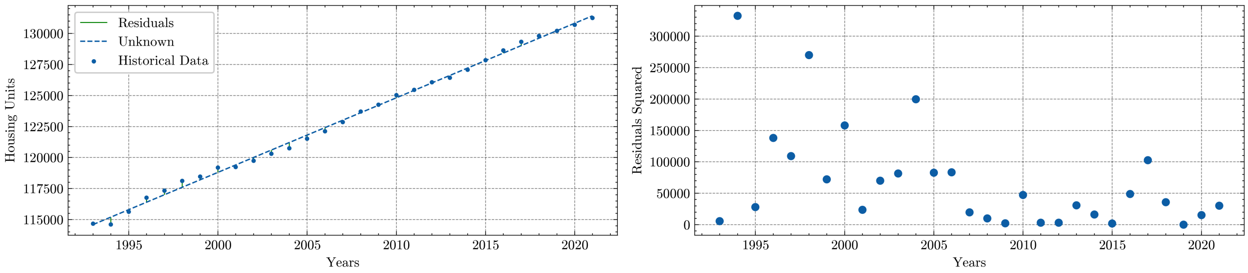

Figure 1.1: On the left is a graph to show the results of a simple univariate linear regression of the total housing units against time, for Brighton and Hove. On the right is a scatter plot of squared residuals against time.

Using this model, the squared residuals in both cities have a high variance, suggesting there are non-negligible effects from underlying factors in this model. Instead of a reductive uni-variate model, a further heavier investigation must be conducted. To unveil a relation, a logistic regression of each variable was considered, according to:

$$x_i(t) = \frac{L_i}{1 + e^{-k_i t}}$$

Values for L_i and k_i were generated with the scipy library using the curvefit function.

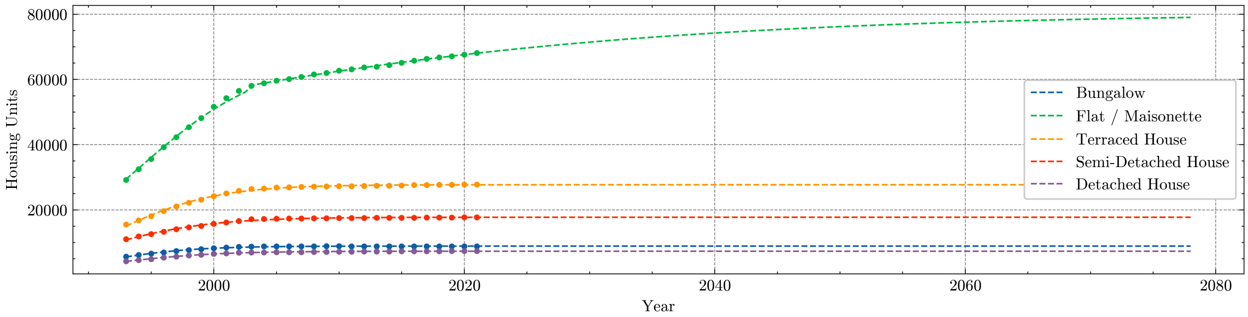

Figure 1.2: Plot of logistic regression on individual components in Brighton and Hove. A plot of Manchester is similar in fashion.

The majority of trends show that the variables are past the point of inflection in its growth cycle, in agreement with Assumption 1.6. Since the UK is becoming saturated, it is not entirely unreasonable to assume the numbers of houses in that particular category will reach their carrying capacity in the deep future.

1.4.2 Adjusting the model

In Brighton and Hove, the exception to the plateau in the number of houses per housing type is ‘Flat / Maisonette’ indicating that around 2003, there was a change in framework of sorts favouring the construction of flats over other types of property. This is evidenced by the 2003 report of the 2001-2006 housing strategy in Brighton and Hove showing it is ’cheaper to rent than to buy in Brighton & Hove’. It is not unjustified, albeit optimistically, to adapt our model in light of this information from this point in time.[3]

Likewise, in 2009 there were policy changes in Manchester that shifted the dynamic of accommodation to a more flat-based system[4]. As such, with the same justification and considerations as Brighton and Hove, the model accounts for this.

Thus, a piece-wise function was deemed more applicable in predicting the growth of 'Flat / Maisonette' data, where the overall model is an amalgam of the two logistic regressions; before the discontinuity and after. This is discussed further in Model Evaluation
Furthermore, initially a large portion of the 'Number of housing units' was classed under 'Unknown'. Under Assumption 1.1, any error in our model caused by this lack of information would only manifest in the past values, and not be relevant to future predictions.

To find the number of new accommodations that were created in the elapsed year, and thus the Housing Supply for the current year, we use the formula:

$$\mathbf{y}(t) = \mathbf{x}(t) - \mathbf{x}(t-1)$$

1.5. Results and Discussions

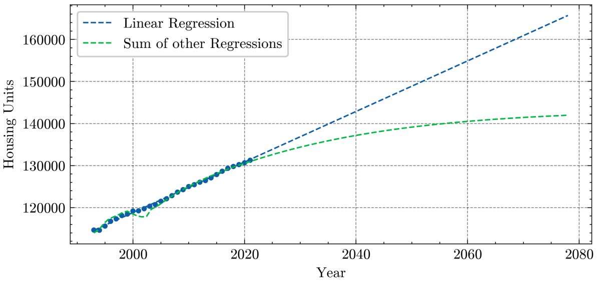

Figure 1.3: Sum of Multiple Uni-variate Logistic Regression Models (Brighton and Hove)

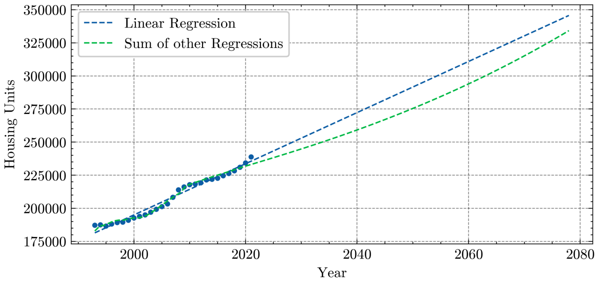

Figure 1.4: Sum of Multiple Uni-variate Logistic Regression Models (Manchester)

The graph presents the stark differences between the reductive linear model and the adjusted logistic model in the number of houses total. The majority of the features considered were on a steady asymptotic rise, justified contextually, since the UK is already considered a developed country, and urbanisation processes contribute less to the interplay between the different household types.

The calculated inflection points for the logistic regression being behind the time frame for certain features of the data makes sense. Potentially the logistic growth model would be more encompassing in a developing country, where the issue of homelessness is more pressing, and the need to construct new forms of housing, of all sorts, is more prevalent, where the exponential 'boom' is still ongoing.
Table 1.2 shows there is Housing Supply, more new accommodation types are being built than decommissioned, in both cities. In Brighton, this occurs in decreasing amounts as time goes on, however the opposite is true in Manchester. This could be due to the fact that Manchester is a larger city in the UK, and hence will experience growth rates that are higher due to factors such as immigration, so in turn, the Housing Supply must increase to meet the increased demands.

Year	**2034**	**2044**	**2074**
Brighton	293	207	67
Manchester	1407	1594	2369

Table 1.2: 'Housing Supply' in Brighton and Manchester (Rounded to nearest whole)

1.6. Sensitivity Analysis

The sensitivity analysis for the models were done by adding random noise to the data points, in order to test the models noise robustness. Each historical data point is scaled by a factor, λ, s.t. $\lambda \sim N(1, 0.05)$. The model is then re-trained on the artificial data. This process is iterated one hundred times, and a 95% confidence interval is plotted, between the 2.5 percentile and 97.5 percentile models. The graph for average percentage error against time, with noise application demonstrates the robustness of the model. Even across a 50-year time interval, the average percentage uncertainty remains under 7% for Brighton and Hove, and under 15% for Manchester - a relatively low number considering Manchester's upwards growth trajectory. This serves as evidence highlighting the flexibility of our model, in addition to the relative confidence we can place in our final predictions.

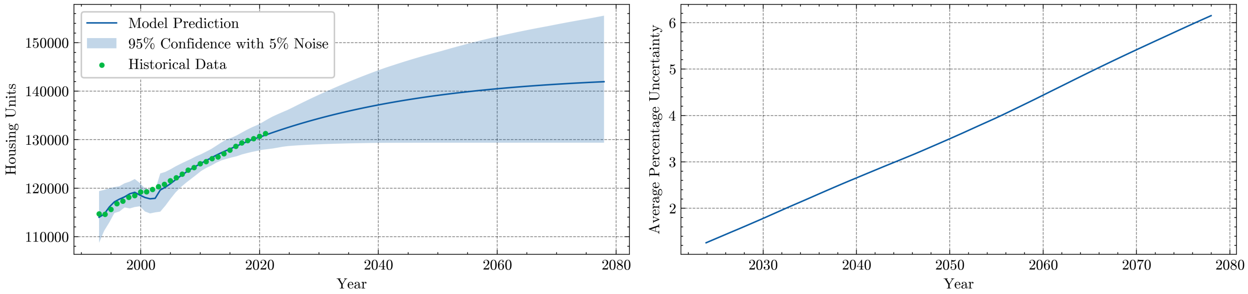

Figure 1.5

1.7. Model Evaluation

1.7.1 Model Strengths

- The discontinuity of our functions adds to its realism. In the real world, no growth model is purely logistic in nature, and oftentimes random events, such as government legislation, must be accounted for. In this case, the clear change in trend can be justified when considering real-world rationale.
- The carrying capacity and growth cap of each feature is accounted for. Accordingly, our model may be more suited to extensive future extrapolation.
- The model is accurate in the short-term because changes in the exponent coefficients cause minimal changes (and thus minimal error) in Housing Supply in the short run.

1.7.2 Model Weaknesses

- Outlier data may be present due to economic circumstances, particularly the housing market crash in the UK in 2008. Hence, the sum of absolute deviations could have been minimised instead of the sum of the squared residuals for the regressions because it is less prone to outlier influences.[5] This was ultimately deemed too computationally expensive for our models.

- Defining a piece-wise function for number of 'Flat / Maisonette' can be seen as over-fitting the data, in the absence of real-world context. It was our aim to reduce this through the addition of noise in the sensitivity analysis. Furthermore, the implementation of this piece-wise function demonstrates significant changes in our model due to urban planning approaches. Over long enough timescales, the continuous model may not be true, as there no doubt will be government legislation inhibiting the previously seen growth trends. The model assumes this does not occur within the time span over which the model is being applied, yet it is likely this assumption will be falsified.

- The partial derivative with respect to a feature can tell us the locally explained variance according to the gradient along one of the inputs - provided the function is locally linear. Given that the considered timescales are large, this 'constant gradient' assumption would break down, due to external factors such as population stagnation, and a lack in demand for new housing, resulting in poorer predictions in the far future. A more complex model would lend itself to more complex methods of sensitivity analysis, for example the employment of the first-order sobol index with respect to each feature, could have allowed us to deeper gain insight over which of the house-types contributes the most to the housing supply against time.

2. Part II: It Was the Worst of Times

2.1. Defining the Problem

The problem asks us to develop a model to predict the number of homeless people 10, 20 and 50 years into the future, with respect to the UK regions of Manchester and Brighton/Hove. Our model will take into consideration past data on homelessness, population, house prices, and related variables.

2.2. Assumptions and Justifications

2.1. **Future homelessness will be caused by the same problems in society as it is today.**
Justification: Major technological or societal shifts would be incredibly difficult to forecast. Since the predictor variables we use are very broad in their scope, under the assumption that no large-scale changes will impact the causes of homelessness, the relationship between our variables and homelessness stays approximately the same.

2.2. **Number of households eligible for homelessness relief duty is a proxy for the number of homeless people.**
Justification: The latter data is the only collected formerly by the local authority. Although they may have missed some proportion of the homeless people in the area, we assume the difference is not too high.

2.3. **The time at which data was collected each year marginally affects results.**
Justification: Some data points for the different predictors where collected at different times

in the year. Since most of our variables are yearly aggregates as well, we assume no inherent yearly seasonality in the data.

2.4. **Homeless people only move into social or rental properties.**
Justification: Houses prices have in recent years outpaced incomes highly. For this reason, we assume that the only dwellings that homeless people will move into and be able to afford are social housing and rental properties.

2.5. **Social rent for private social housing is sufficiently close to public social housing.**
Justification: Due to data availability, we have only collected data about social housing rent for private companies that have been commissioned to create social housing. We assume this is roughly similar to the rent for properties directly owned by the government.

2.6. **Those susceptible to homelessness are generally in the bottom 20th percentile of income.**
Justification: In order to incorporate income into our model, we concluded that mean income is a bad predictor variable as wealth is generally not normally or uniformly distributed in a society. Hence we are using a bottom percentile to better model the amount of income available to those with the least wealth. Our assumption is that these people at the highest risk of becoming homeless and wealthier people will represent a marginal amount of those becoming homeless.

2.7. **Total population, bottom 20th percentile of income and average rent for social housing are variables that will linearly grow into the future**
Justification: This is the largest assumption in our model. Using data gathered from Eurostat[6], we have observed linear growth trends in all 3 variables in the past 20 years, and initial linear regression yields promising r^2 values, ranging from 0.88 to 0.98, which indicates high linear correlation. However, some variables are difficult to justify 50 years from now - for example, populations for Western countries are beginning to drop, and the UK may likely follow. To reduce the complexity of the model, we have decided to keep this assumption.

2.3. Defining Parameters

The variables in the table below apply individually to each region examined.

Type	**Symbol**	**Definition**	**Units**
Dependent Variable	$H(t)$	Number of households eligible for homelessness relief duty	Unitless
Predictor Variable	$P(t)$	Total population	Unitless
Predictor Variable	$S(t)$	Average rent for social housing	£
Predictor Variable	$U(t)$	Total housing units	Unitless
Predictor Variable	$I(t)$	Bottom 20th percentile of income	£

Table 2.1: Summary of Problem 1 Variables & Constant Parameters

2.4. Model Development

To predict the amount of homeless people in Manchester and Brighton and Hove for the next 10, 20 and 50 years, we first gathered data for each variable in the above table between the years of 2008 - 2017.

Using this data, we created our model in 2 parts:

1. At any point in time we combine our predictor variables using a multivariate ridge regression to obtain a prediction for the amount of homeless people at that same point in time. Our final formula for $H(t)$ will be of this form:

   $$H(t) = \alpha + \beta_1 P(t) + \beta_2 S(t) + \beta_3 U(t) + \beta_4 I(t)$$

   where β represents learned multipliers for each variable in the model (β_2 and β_4 are adjusted for units), and α is a learned y-intercept value for $H(t)$.

   To learn these parameters, regular linear regression uses the residual sum of squares to calculate the error from each data point to the predicted line. However, when data has over fit due to multi-colinearity, there is often large variance in the size of the coefficients, since if 2 predictor variables are highly correlated, the the model will just favour using only one of them to explain predictions, causing it to dismiss the other variables. Hence, ridge regression uses a new estimator modified from ordinary least squares, but with an extra term that prevents the size of the coefficients from becoming too large:

   $$\sum (\hat{H}(t) - H(t))^2 - \lambda \cdot \sum \beta^2$$

   The second term is controlled by a parameter λ, which we have found the best value for in our model using a grid search, whereby we look at a list of different λ values and see which one provides the best fit.

2. After creating the estimator for $H(t)$ we use another set of regression lines to extrapolate our predictor variables into the future. Each predictor variable mentioned in Assumption 2.7 will be fit using linear regression as a variables solely against time, while total housing units predictions will be utilized from Question 1. We then plug our variables into the ridge regressor and obtain future estimates for $H(t)$.

2.4.1 Anomaly Detection

Detecting and removing anomalies constitutes an important part of ensuring our model provides a good fit into the future, as outliers can heavily skew regression algorithms. We suspected that the 2008/9 Financial Market Crash and other historical events would create such results. To find these anomalies, we employed an a technique known as Isolation Forest.

This is a method of detecting anomalies through the employment of decision trees. In centers around the idea that anomalies are easy to separate from the rest of the data. At each node there is a randomly selected split point with respect to our variable. An ensemble of these random trees create the isolation forest, whereby an anomaly score is created using the path-length measure for a data point. The more binary splits required to isolate the data point, the less of an outlier it

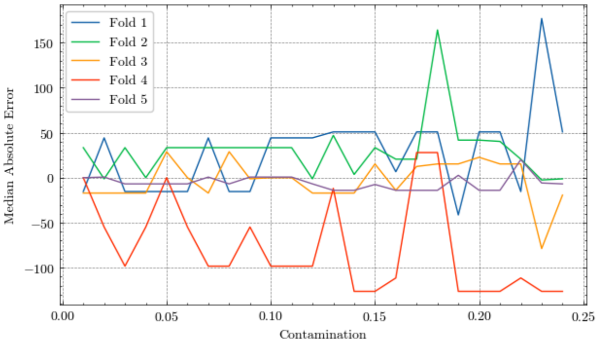

Figure 2.1: Manchester Contamination vs Error

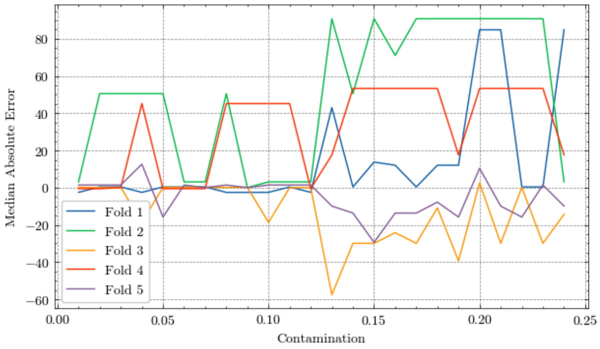

Figure 2.2: Brighton and Hove Contamination vs Error

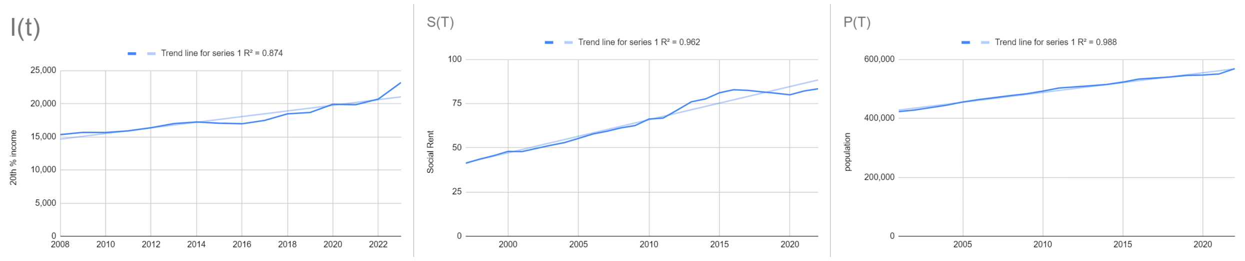

Figure 2.3: The R^2 values for the plots are 0.874, 0.962, 0.988 respectively

is considered to be. On average, the outlier data can be described in less information. In order to isolate an outlier, the algorithm recursively generates partitions on the sample by randomly selecting an attribute and then randomly selecting a split value in the feature appropriate range.

Fortunately, the results of the anomaly detection only revealed one major anomaly, for Manchester's homeless population in 2018: Removing this brought our validation error down by 1.2% (from 13.3% to 12.1%), indicating that otherwise the data is fairly consistent. This is validated by splitting the model into 5 folds and only choosing the 'contamination' value when the median absolute error of all 5 folds had decreased relative to no outliers, as seen in Figure 2.1 and Figure 2.2.

2.5. Results and Discussions

Firstly, we will display the results from linearly extrapolating the 3 aforementioned predictor variables: $P(t)$, $S(t)$ and $I(t)$, for both Manchester and Brighton/Hove below.
As was mentioned in Assumption 2.7, the data is promising and suggestive of a good linear fit. Another indicator of good fit was also that the residuals of the data were fairly random and didn't tend to grow as time increased. This property is known as homoskedasticity, and it is encountered when the independent variable completely explains the dependent variable. While not conclusive evidence, it does validate our findings and assumptions.

Using this we then obtained the following graphs for homelessness over time:
From the graph we can observe a linear growth in homeless people for both cities, which is corresponding to our predictor variables being linear in nature. As we will discuss in model evaluation, this can be refined.

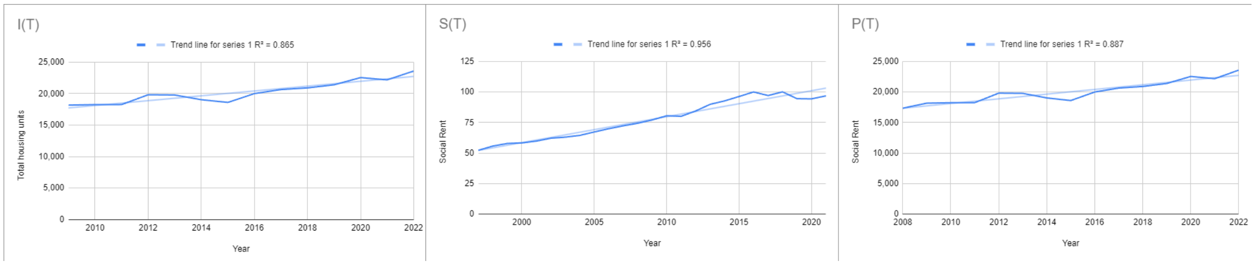

Figure 2.4: The R^2 values for the plots are 0.865, 0.956, 0.887 respectively

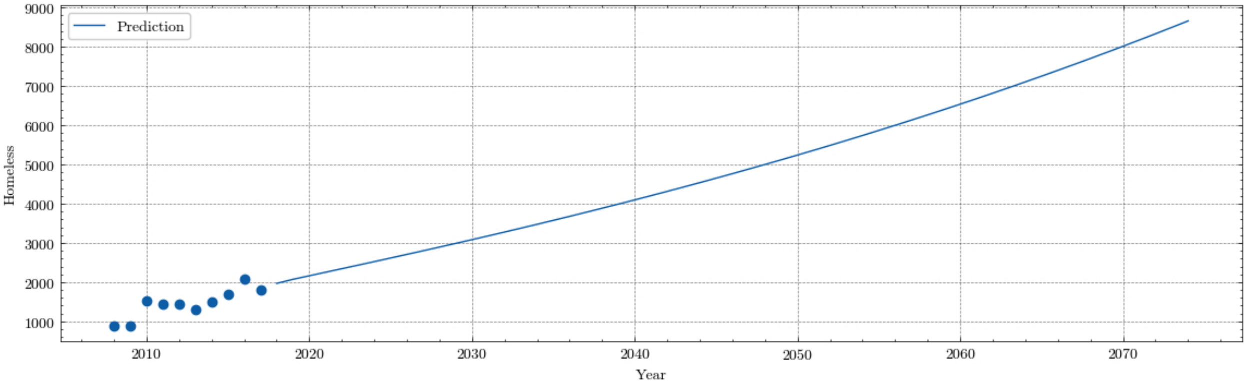

Figure 2.5: $H(t)$ for Manchester

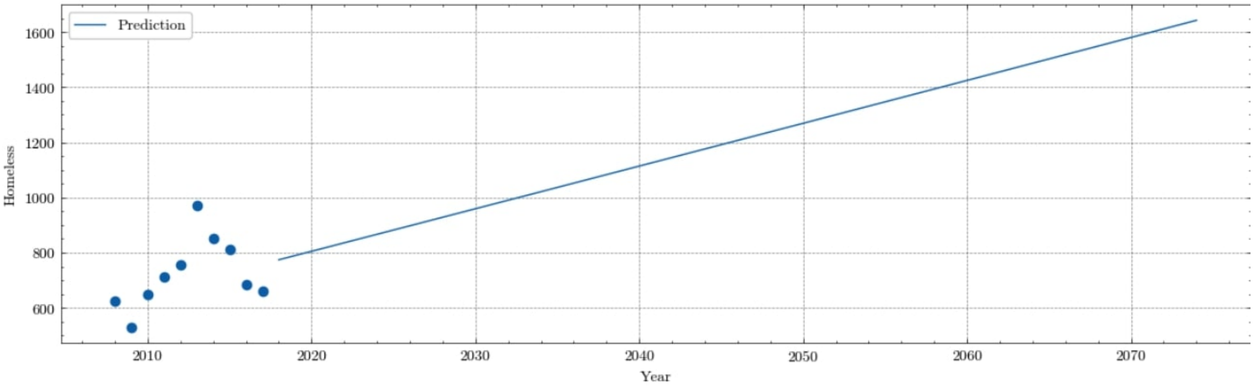

Figure 2.6: $H(t)$ for Brighton

Year	$H(t)$ **for Brighton**	$H(t)$ **for Manchester**
2034	1,021	3,474
2044	1,176	4,536
2074	1,643	8,657

Table 2.2: Results for $H(t)$ 10, 20 and 50 years from now

2.6. Sensitivity Analysis

In order to assess the accuracy of the coefficients in our multivariate linear regression, we have applied ±5% noise to all our coefficients, and re-extrapolated. From this, we find the spread and calculate the percentage uncertainty in each of our estimates.

Year	$\Delta H(t)$ **for Brighton**	$\Delta H(t)$ **for Manchester**
2034	8.2%	8.7%
2044	9.1%	10.5%
2074	15.3%	19.3%

Table 2.3: Percentage uncertainty in $H(t)$ 10, 20 and 50 years from now

Considering the timescales involved, the uncertainty does grow fairly gradually over time. This implies our coefficients are robust and if there is some error is selecting them, the outputs should be fairly consistent, at least for the short term. Unfortunately this data can not be utilised to a full extent in question 3, which is explored in greater detail later.

2.7. Model Evaluation

2.7.1 Model Strengths

- **Reducing multi-colinearity**: By using Bayesian ridge regression to estimate homelessness via our predictor variables, it is particularly useful to mitigate the problem of multi-colinearity in linear regression, which commonly occurs in models with large numbers of parameters, as is in our problem. We suspect that parameters such as social rent and the income of the bottom 20% to be correlated for reasons including inflation, economic growth, etc. Using regular linear regression would undermine the statistical significance of an independent variable.
- **Improved fitting capabilities**: The splitting of our model into 2 different regressive components allows us to model various non-linearities that otherwise would be difficult to if we only forecasted using homelessness data against time. This is further enforced by the r^2 value for homelessness against time being 0.16, meaning that at the very least a solely linear model would be a bad fit.
- **Splitting up the model**: If we ever get better estimates for our predictor variables that aren't linear, we can always separate the model and use the ridge regressing head on our better estimates - we have essentially decoupled and time series data

- **Easily extendable**: Our linear and ridge regressions can incorporate any type of time series data and can flexibly expand to account for additional predictor variables. This allowed us to test different predictor variables quickly and settle on a few strong one, and it would also allow quick modification if we were going to apply it to the American cities as well.

2.7.2 Model Weaknesses

- **Generality**: The number of assumptions we have loaded into the model may reduce its efficacy in other regions or countries. If we were to apply it to the USA, we would have to likely alter the predictor variable extrapolation with some new assumptions - The USA has a stronger economy, a less stagnating population, different socio-cultural factors, etc.
- **Noise**: The estimate from the ridge regressor has a mean error of 11% for $H(t)$. While this is understandable given the rate of homelessness is dependent on many more individual smaller factors than our 4, it does pose a barrier to the significance of our findings.

3. Part III: Rising from This Abyss

3.1. Defining the Problem

Our final problem requires us to determine the best course of action to reduce homelessness into the future. For government officials to best allocate resources for creating infrastructure and legislation in this area, it is necessary to evaluate multiple plans and their impacts indirectly into homelessness. In this section, we quantified the impacts of 3 different plans, as well as the effect of migration on homelessness.

3.2. Assumptions and Justifications

3.1. **The time taken to implement our strategies is insignificant**
Justification: We want to determine the long-term changes to homelessness each policy has, and so any startup times for the policies are assumed to be absorbed into the time-scale.

3.2. **Employed homeless people are working minimum wage jobs**
Justification: Due to the data availability, we don't know exactly what types of jobs homeless people are working. However, this is a reasonable assumption given their living situation.

3.3. **The proportion of homeless people against their potential income left on rent follows a linear decreasing distribution**
Justification: The range of values of this distribution is from 0 to R (defined below). This is once again due to data availability issues, but generally the distribution should be decreasing, since the majority of homeless people don't have any income.

3.4. **Natural Disasters don't affect housing or homelessness in reasonable ways in the UK**
Justification: The reason we are not modelling natural disasters is that the only major such events are flooding and heavy storms, both of which generally displace people temporarily, and people who do lose their home we assume are covered by insurance. We understand that other regions, like the US, would differ in this assumption.

3.5. **Should there be a small increase in the rent of a social house, the residents would be able to still afford it or there would be enough competition to ensure it is not**

vacant
Justification: The

3.6. **Homeless people will attempt to leave homelessness is they have the means**
Justification: Since this is an undesirable state for any human being to be in, it is a very mild assumption to make.

3.7. **All 'migrants' are immigrants**
Justification: This assumption is made for the simplicity of our models. In reality, while this may not be the case, most migration occurs due to an accumulation of push and pull factors, and the difference in quality of life between two cities in the UK is deemed too insignificant a factor for long-term intranational migration. Thus international migration is the only form of migration considered.

3.8. **Statistics from comparable UK cities reflect trends seen in Manchester**
Justification: Major cities in the UK possess more similarities than differences. Hence, statistics on proportions in other similar cities are reflective of Manchester proportions also.

3.3. Defining Parameters

Type	Symbol	Definition	Units
Variable	B_S	Cost in pounds to reduce homeless population by 1 for a strategy S	$\frac{Pound}{People}$
Constant	C	Cost of building private social flat	$(£91,290)$ [7]
Variable	R	Average cost of social housing per week	£
Distribution	$I(x)$	PDF of proportion of homeless people against income left for rent	N/a

Table 3.1: Summary of Problem 1 Variables & Constant Parameters

As shown in the table in Table 3.1

3.4. Model Development and Results

In this model we've considered 3 different government policies to tackle housing:

- Build new free homeless housing.
- Reduce social rent
- Increase minimum wage

We will also consider the situation of migrants moving to the UK, and how that will affect homelessness.

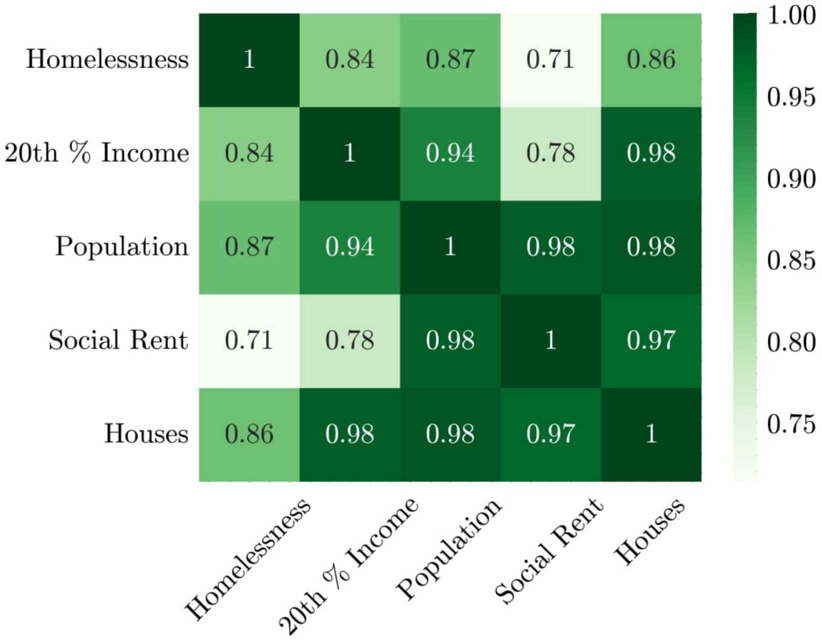

Figure 3.1: The correlation matrix for our predictor variables from Q2.

Our initial attempt at this problem involved the utilisation of the ridge regression model to predict $H(t)$ using our predictor variables from Question 2. This would be used to calculate the increases in the predictor variables created by each plan, and it would quantify the effects on homelessness figures. However, the problem encountered with this approach was that the predictor variables were highly correlated (shown in Figure 3.1), which resulted in the unrealistic changes to homelessness when variables where changed individually. One example of this was that an increase in the amount of housing led to a predicted increased homelessness - a nonsensical result. This could have arised from the high interdependence between housing and population.

Furthermore, the high correlations exhibited between variables themselves are not strong enough sources of evidence to imply any sort of causal link. This is further explored in Model Evaluation. With these pitfalls in mind, we made the decision to answer the question via simpler analysis of various plans and policies, each modelled individually, and determined a B_S for each strategy.

3.4.1 Building new flats

In question 1, the long-term growth in number of houses per housing type was only seen in 'Flat / Maisonette' for both cities in the UK. The allocation of a certain proportion of these newly built flats as free homeless housing, would naturally cause a decrease in the number of homeless people. Therefore, the cost of building new private social flats (C) can serve as a concrete benchmark for the efficacy, or lack thereof, of the consequent strategies explored.

3.4.2 Increasing minimum wage

For this model, it is useful to consider the distribution of homeless people against income left for rent. Under assumption Assumption 3.3, this will be a linear decreasing probability distribution of the following form, with the constraint that the integral of the distribution is 1 (such is the case of all probability distributions):

$$\text{I(x)} = \begin{cases} 0 & x < 0 \\ \frac{-2}{R^2}(x - R) & 0 \leq x < R \\ 0 & R \leq x \end{cases}$$

People who have a more than R income left for rent to spend, will spend it on rent, and by implication will stop being homeless, which is why the distribution is 0 past R.

It is useful to also consider an increase to the minimum wage. Since all employed homeless people are minimum wage by Assumption 3.2, this shift will affect only the end "nose" of the distribution. The range of income left for rent those working minimum wage will have can be estimated - around 23.4%[8] of homeless people are employed both full-time and part-time. If they constitute the final 23.4% of the distribution, then the range of rent incomes they sit in can be calculated, and one arrives at $42.5 \leq x \leq 82.4$.

According to this model, any increase to the minimum wage of up to $39.9 = (82.4 = 42.5)$ would reduce homelessness. Values less than this will result in linear increases of homeless people removed, and values over will yield no further improvements as all minimum wage working unhoused people would have already yielded benefits.

While an increase that high is unreasonable, a smaller increase would likely serve an easy and cost-effective strategy to reduce homelessness.

3.4.3 Increased Rent of a Social House

As per Assumption 3.5 the limiting factor for social housing occupation is its availability, as opposed to its cost. Therefore, it stands to reason that although counter-intuitive, increasing the price of rent, and reinvesting the proceeding profit into more social housing, can be effective strategies to decrease homelessness in the long term.

The model operates by first assuming that at time $t = t_0$ (where t_n dictates the nth year), there are H_0 (houses at t_0) houses. Then if all revenue is reimbursed directly into the construction of more social housing, the number of houses would be updated through the relation $H_{n+2} = H_{n+1} + \frac{Hn*12*20}{C}$, where $\frac{Hn*12*20}{C}$ represents the annual number of houses built annually. This number is justified when examining the minimum wage (£10 an hour); £20 a month represents an extra 2 hours of monthly work, a figure that seems reasonable considering the surrounding circumstances of the homeless population. The number of council houses increasing will cause the number of homeless people (calculated in question 2) to decrease at an approximately constant rate. By 17 years, the homeless population will approximately halve, as seen in Figure 3.3.

3.4.4 Migration

Although not previously considered, migration is a contributing factor in the increase in number of homeless people. It is still worth investigating, as it can serve as an avenue towards more interesting statistical considerations regarding conditional probability. Let $P(A)$ be the probability of an individual in the population being homeless and $P(B)$ be the probability of an individual in the population being an immigrant. These values can be calculated by looking at the proportion

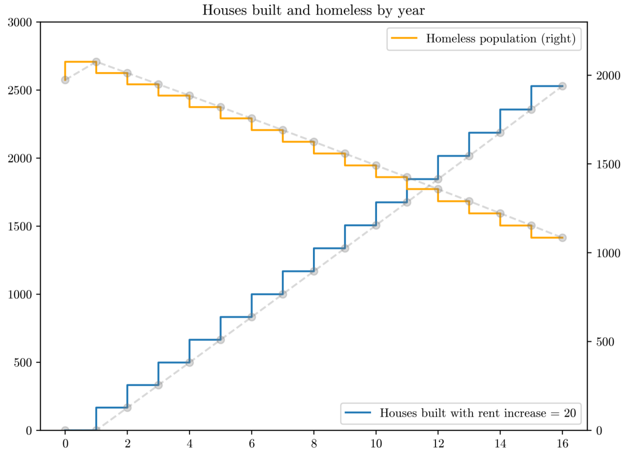

Figure 3.2: New houses built and homeless population vs time since policy introduced

$P(A)$ **[9]**	$P(B)$ **[10]**
0.0135	0.31

Table 3.2: Table caption

of Manchester's total population that fall under these subcategories. In reality, migrant income is distributed across all percentiles of income in the UK according to an unknown. Estimating the minimum income percentile required to afford shelter is no trivial task. Luckily, this value does not need to be estimated, as Bayes' theorem is useful in finding the proportion of homeless migrants. 52% of homeless people are migrants in London[11], a comparable UK city. Assuming the same is true for Manchester Assumption 3.8. This provides a value for $P(B|A)$. The Bayes' Theorem formula can tell us the $P(A|B)$ Depending on this proportion, an increase in population due to migration factors will be causally linked with an increase in the homeless population.

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

Figure 3.3: Bayes' Theorem

Ultimately our result for $P(A|B) = 0.022$ to 3 significant figures. One way in which the issue of homelessness may be tackled is through an adjustment in governmental policies regarding migration. That is to say, for every 44 migrants, 1 of them is homeless. Reducing migration will reduce homelessness linearly, in the short term.

3.5. Evaluating the Model

3.5.1 Model Strengths

- A majority of our strategies operate independently from external factors. This aids in the establishment of a clear causal relationship between the implementation of the strategy and the associated decrease in the homeless population. This stands in contrast to an indirect approach using more aggregated models, where it would prove difficult, and potentially speculative to establish a link beyond a correlation.

- Although initially counter-intuitive, the suggestion to increase social housing prices, represents a largely novel, and potentially very effective methodology to reduce the number of homeless people on England's streets. It can also be simply lagged in the case of migration, and is therefore very adaptable.

3.5.2 Model Weaknesses

- The primary limitation of the strategies is an uncertainty in efficacy in long time scales. Despite the majority of our statistics exhibiting relative stationary behaviour, especially in the short term, the disregard towards their fluctuations with time inherently limits the extent to which conclusions can be drawn from the results.
- The simplicity of our models, in comparison to the more intricate nature of implementing city-wide policies, may fail to adequately account for the logistical complications that arise from practical implementation. Conversely, the simplistic, yet versatile nature of the models on which we have built our suggestions provides a concrete framework, on which the individual circumstances of the implementation can be added.
- The 'migration' model is only valid when considering a reduction in the rate of migration due to government legislation in the short term. In reality, there could could be adverse long term effects, alongside large-scale social repercussions associated with its implementation.

4. Conclusions and Perspectives

The problem of homelessness is more prevalent than ever. In contemporary society, this paper sheds light on pervasive structural inequalities and systemic obstacles encountered by numerous individuals and communities. Continuously, a growing number of individuals confront housing instability, grappling with a range of intricate challenges, frequently entwined with economic adversity.

Question 1 demonstrated that various cities in the UK can present varying trends in presenting in the housing supply. This aspect warrants careful consideration when formulating strategies to alleviate widespread homelessness.

Question 2 delved into the predictor variables growth in homeless people for UK cities. We struggled with finding a way around the high dependence between predicting variables, but ultimately used ridge regression to complete this task, finding a linear relationship for homelessness over time.

Question 3 presented a number of different solutions to tackle the increasing challenge of climate change. We considered building new flats, the effect of migration, increasing social rent prices and increasing minimum wage. Each presented their own unique perspective, and we suggest the Minister of State for Housing and Planning employ an array of the proposed methods, dedicating funds to improve the metrics that they wish to optimise.

While the road to ending homelessness may be long and arduous, it is not insurmountable. By working together and addressing the underlying factors that contribute to homelessness, we can create a society where everyone has a place to call home. It is only through collective action and compassion that we can truly make a meaningful difference in the lives of those who are experiencing homelessness.

5. References

[1] H. Department for Levelling Up and Communities, "Housing supply: Net additional dwellings." https://www.gov.uk/government/collections/net-supply-of-housing, Nov 2023.

[2] W. Kenton, "Homoskedastic: What it means in regression modeling, with example." https://www.investopedia.com/terms/h/homoskedastic.asp.

[3] https://shorturl.at/ekHIZ.

[4] https://shorturl.at/bjBMZ.

[5] D. Birkes and Y. Dodge, *Alternative methods of regression.* John Wiley Sons, 1993.

[6] D. Commons, "Data Commons: Place Explorer for the UK." https://datacommons.org/place/country/GBR?utm_medium=explore&mprop=count&popt=Person&hl=en, March 2024. Accessed: 2024-02-03.

[7] https://bit.ly/48GmEUi.

[8] https://www.gov.uk/government/statistical-data-sets/live-tables-on-homelessness.

[9] N. Haigh, "Unveiling the crisis of homelessness in manchester [2024]." https://www.mastermanchester.co.uk/is-homelessness-a-problem-in-manchester/, Aug 2023.

[10] M. C. Council, "The council and democracy census 2021 - migration summary." https://www.manchester.gov.uk/info/500388/census_2021/8585/census_2021_-_migration_summary#:~:text=The%20Council%20and%20democracy%20Census%202021%20%2D%20Migration%20Summary&text=31%25%20of%20residents%20were%20born,residents.

[11] L. Geraghty, "Rough sleeping in london rises to highest since records began." https://www.bigissue.com/news/housing/rough-sleeping-london-homelessness-chain/#:~:text=The%20number%20of%20non%2DUK,Haddad%2C%20St%20Mungos%20chief%20executive., Nov 2023.

6. Appendix: Code Listing for Technical Computing Consideration

6.1. Part I: It Was the Best of Times

Question1.py

```python
# -*- coding: utf-8 -*-
"""Question 1.ipynb

Automatically generated by Colaboratory.

Original file is located at
    https://colab.research.google.com/drive/1e6XJJh86TXGuJRgAbmP3swQL9ct-A_xS
"""

import numpy as np
import pandas as pd

from scipy.optimize import curve_fit
from sklearn.metrics import r2_score

# Used for styling of graphs
import matplotlib.pyplot as plt
from matplotlib.collections import LineCollection

import scienceplots
plt.style.use(['science', 'grid', 'no-latex'])

# Reads data from excel - swap "Brighton" & "Manchester" to switch
df = pd.read_excel('/content/Question 1.xlsx', sheet_name='Brighton Data')

featureNames = df.columns.values[1:]
data = df.values.T
xData, yData = data[0], data[1:]

# Data that you are predicting for
xPrediction = np.array([2034., 2044., 2074.])

# Codes the data to be in a more manageable range
minXData = np.min(xData)
xData -= minXData
xPrediction -= minXData

df.head(5)

# Defines the logistic function
def logistic(xData, L, k, x0):
  return L / (1 + np.exp(-k * (xData - x0)))

# Defines the linear function
def linear(xData, m, c):
  return m * xData + c

# Creates the logistic models for the param
def createLogisticModels(xData, yData):
  models = np.empty([len(yData)], dtype=np.dtype(np.object_))

  # Loops over each of the variables
  for i, data in enumerate(yData):

    # For each the parameters of the model are calculated using OLS
    popt, pcov = curve_fit(logistic, xData, data, p0 = (np.max(data), 0, 0),
                        bounds=([0, 0, -np.inf], [np.inf, np.inf, np.inf]), maxfev=10000)

    # Creates function for the model with the found parameters
    models[i] = lambda xData, popt=popt: logistic(xData, *popt)

    print(f'___{featureNames[i + 1]}___')
    print(f'Paramater Values: {popt}')
    print(f'R^2 value: {r2_score(data, models[i](xData))}\n')

  return models

# Combines the linear and logstic section of the functions
def combineFlatsPiecewise(logisticSection, linearSection, switchYear):
  def model(xData):
    points = np.zeros(len(xData))

    # If year earlier than switch year, use logistic - otherwise linear
    for i, value in enumerate(xData):
      if value < switchYear:
        points[i] = logisticSection(value)
      else:
        points[i] = linearSection(value)
    return points
  return model

def createFlatsModel(xData, yData, logisticSection):
  # 2003 for Brighton, 2009 for Manchester
  switchYear = 2003 - minXData

  # Creates a linear regression after the switch year
  popt, pcov = curve_fit(logistic, xData[switchYear:], yData[switchYear:],
                         p0 = (np.max(data), 0, 0), maxfev=10000,
                         bounds=([0, 0, -np.inf], [np.inf, np.inf, 0]))
  linearFlatsModel = lambda xData, popt=popt: logistic(xData, *popt)

  print(f'___{featureNames[2]}___')
  print(f'Paramater Values: {popt}')

  # Combines this with the initial logistic porion
  flatsModel = combineFlatsPiecewise(logisticSection, linearFlatsModel, switchYear)

  print(f'R^2 value: {r2_score(yData, flatsModel(xData))}\n')
  return flatsModel

def createTotalHousingModel(xData, yData):
  # Creates the linear regression for the total housing
  popt, pcov = curve_fit(linear, xData, yData)
  totalModel = lambda xData, popt=popt: linear(xData, *popt)

  print(f'___{featureNames[0]}___')
  print(f'Paramater Values: {popt}')
  print(f'R^2 value: {r2_score(yData, totalModel(xData))}\n')
  return totalModel

def modelSum(models):
  # Combines all of the individual models into one final one for predicting the houses sold
  def model(xData):
    total = np.zeros(len(xData))

    # For each of the models - add them to the main one
    for model in models:
      total += model(xData)
    return total
  return model

logisticModels = createLogisticModels(xData, yData[1: -1])

# Adds the piecewise logistic function for flats
```

```python
logisticModels[1] = createFlatsModel(xData, yData[2], logisticModels[1])

# Linearly interpolates for the unknown values
unknownModel = lambda x: np.interp(x, xData, yData[-1])

# Combines the inidvidual models
model = modelSum(np.append(logisticModels, unknownModel))

# Defines basic linear regression
totalHouseModel = createTotalHousingModel(xData, yData[0])

print("Prediction of Housing Supply")

# Loops through all the predictions and calcualtes the difference in housing
# units for "housing supply"
yPredictions = np.zeros(len(xPrediction))
for i, prediction in enumerate(xPrediction):
  data = model(np.array([prediction, prediction - 1]))
  yPredictions[i] = data[0] - data[1]
  print(f'{prediction + minXData}: {yPredictions[i]}')

# Initialises constants
X_LABEL = "Year"
Y_LABEL = "Housing Units"
LINE_LABEL = "Line Label"
LOWER_X, UPPER_X = 0, 85

# Initialises settings of the graph
plt.figure(figsize=(13, 3), dpi=500)

plt.xlabel(X_LABEL)
plt.ylabel(Y_LABEL)

xPlot = np.linspace(LOWER_X, UPPER_X, 100)

# Plots the regression
for i, logisticModel in enumerate(logisticModels):

  plt.plot(xPlot + minXData, logisticModel(xPlot), linestyle='dashed', label=featureNames[i + 1])
  plt.scatter(xData + minXData, yData[i + 1], marker="o", s=7)

plt.legend(loc='center right')
plt.show()

# Initialises settings of the graph
plt.figure(figsize=(6.5, 3), dpi=500)

plt.xlabel(X_LABEL)
plt.ylabel(Y_LABEL)

# Plots the regression
xPlot = np.linspace(LOWER_X, UPPER_X, 100)
plt.plot(xPlot + minXData, totalHouseModel(xPlot), linestyle='dashed', label='Linear Regression')
plt.plot(xPlot + minXData, model(xPlot), linestyle='dashed', label='Sum of other Regressions')

# Plots the data points
plt.scatter(xData + minXData, yData[0], marker="o", s=7)

plt.legend()
plt.show()

fig, axes = plt.subplots(ncols=2, figsize=(13,3), dpi=500)

# Creates line segments connecting the points to their prediction on the regression
segments = [[[xData[i] + minXData, yData[0][i]], [xData[i] + minXData, totalHouseModel(xData[i])]] for i in range(len(xData))]

```

```
lc = LineCollection(segments, zorder=0, color='green', label='Residuals')
lc.set_array(np.ones(len(yData[0])))
lc.set_linewidths(np.full(UPPER_X, 0.7))

axes[0].add_collection(lc)


# Plots the regression
xPlot = np.linspace(np.min(xData), np.max(xData), 100)
axes[0].plot(xPlot + minXData, totalHouseModel(xPlot), linestyle='dashed')

# Plots the historical data points
axes[0].scatter(xData + minXData, yData[0], marker="o", s=5, label="Historical Data")

axes[0].set_xlabel(X_LABEL)
axes[0].set_ylabel(Y_LABEL)
axes[0].legend()


# Plots the residuals squared
axes[1].set_xlabel(X_LABEL)
axes[1].set_ylabel('Residuals Squared')
axes[1].scatter(xData + minXData, (yData[0] - totalHouseModel(xData)) ** 2, marker="o", s=25)

fig.tight_layout()
plt.show()

# Create a model for given x and y data
def createModel(xData, yData):
  logisticModels = createLogisticModels(xData, yData[1: -1])
  totalHouseModel = createTotalHousingModel(xData, yData[0])
  flatsModel = createFlatsModel(xData, yData[2], logisticModels[1])
  unknownModel = lambda x: np.interp(x, xData, yData[-1])

  model = modelSum(np.append(np.delete(logisticModels, 1), [flatsModel, unknownModel]))
  return model

NOISE_PERCENTAGE = 5
PERCENTILE = 95

# Applies noise to the input data
def applyNoise(data):
  return data * np.random.normal(loc=1, scale=NOISE_PERCENTAGE / 100, size=np.shape(data))

xPlot = np.linspace(LOWER_X, UPPER_X, 100)
yPlots = []

# Creates 100 models with noisy input data and finds their predictions
for _ in range(100):
  tempModel = createModel(xData, applyNoise(yData))
  yPlots.append(tempModel(xPlot))

# Calculates the upper and lower percentiles for predictions
yUpper = np.percentile(yPlots, 50 + PERCENTILE / 2, axis=0)
yLower = np.percentile(yPlots, 50 - PERCENTILE / 2, axis=0)

# Retrieves the first order indicies
fig, axes = plt.subplots(nrows=1, ncols=2, figsize=(12,3), dpi=500)

# Plots the regression
axes[0].plot(xPlot + minXData, model(xPlot), label="Model Prediction")

# Plots the confidence interval for the percentile level
axes[0].fill_between(xPlot + minXData, yUpper, yLower,
                 alpha=0.25,
                 label=f"{PERCENTILE}% Confidence with {NOISE_PERCENTAGE}% Noise")

```

```python
axes[0].scatter(xData + minXData, yData[0], marker="o", s=7, label="Historical Data", zorder=2)

axes[0].set_xlabel("Year")
axes[0].set_ylabel("Housing Units")
axes[0].legend(loc='upper left')


# Plots the percentage uncertainty with the noise
yPrediction = model(xPlot)
axes[1].plot(xPlot[36:] + minXData, (np.abs(yPlots - yPrediction) / yPrediction).mean(axis=0)[36:] * 100)
axes[1].set_xlabel("Year")
axes[1].set_ylabel("Average Percentage Uncertainty")

fig.tight_layout()
plt.show()
```

6.2. Part II: It Was the Worst of Times

Question2.py

```python
# -*- coding: utf-8 -*-
"""Copy of Question 2.ipynb

Automatically generated by Colaboratory.

Original file is located at
    https://colab.research.google.com/drive/1elVcqehfRCNl19Do3R3peq1XOHGgK6uX
"""



"""# Importing Libraries and setup"""

from sklearn import datasets
from sklearn.model_selection import train_test_split
from gurobi_optimods.regression import LADRegression
from catboost import CatBoostRegressor
from sklearn.metrics import mean_absolute_percentage_error
import sklearn.pipeline
import sklearn.neighbors
from sklearn import linear_model
import pandas as pd
import numpy as np
import sklearn.ensemble
import sklearn

# Used for styling of graphs
import matplotlib.pyplot as plt
import scienceplots
plt.style.use(['science', 'grid', 'no-latex'])

"""# Preparing data"""

data = pd.read_csv("/content/predictors Brighton Hove - Manchester.csv") # OR use Brighton

# where homelessness data is in range
housing = data["Houses"][15:25]
lowIncome = data["20th % income"][15:25]
pop = data["population"][15:25]
socialRent = data["Social Rent"][15:25]

xData = pd.DataFrame(data={
    "housing":housing,
    "lowIncome":lowIncome,
    "socialRent":socialRent,
    "pop":pop
}).reset_index(drop=True)


yData = pd.DataFrame(data={
    "homelessness": data["Homelessness"][15:25]
}).reset_index(drop=True)

# allow conversion to numbers
xData.replace(',','', regex=True, inplace=True)
yData.replace(',','', regex=True, inplace=True)

# get values that are not NaN - this is for regressing
lowIncome = data["20th % income"][15:29]
pop = data["population"][8:29]
socialRent = data["Social Rent"][4:29]

year = data["Year"][:31]

pop = pd.DataFrame(data={
  "year": year,
  "pop": pop
}).dropna(subset=['pop']).reset_index(drop=True).replace(',','', regex=True)

socialRent = pd.DataFrame(data={
  "year": year,
  "socialRent": socialRent
}).dropna(subset=['socialRent']).reset_index(drop=True).replace(',','', regex=True)

lowIncome = pd.DataFrame(data={
  "year": year,
  "lowIncome": lowIncome
}).dropna(subset=['lowIncome']).reset_index(drop=True).replace(',','', regex=True)

"""# Reference model performance"""

losses = []

reg = sklearn.linear_model.BayesianRidge(alpha_1=1e-03, alpha_2=9e-05, lambda_1=3e-06, lambda_2=9.5e-06)

for i in range(len(yData["homelessness"])):
  xTrain = xData[xData.index != i]
  xTest = xData[xData.index == i]
  yTrain = yData[yData.index != i]
  yTest = yData[yData.index == i]

  reg.fit(xTrain, yTrain)
  preds = reg.predict(xTest)
  losses.append(mean_absolute_percentage_error(preds, yTest))

  print(preds)

np.mean(losses)

"""# outliers"""

# reference: https://www.kaggle.com/code/mateuszk013/playground-series-s3e25-mohs-hardness#-5-%7C-Modelling-%E2%86%91
# remove outliers for a given detector
def remove_outliers(data, detector):
    if not isinstance(data, pd.DataFrame):
        raise TypeError(f"'data' must be {pd.DataFrame!r} instance")

    result = detector.fit_predict(data)
    outlierIds = pd.Series(result == -1, index=data.index, dtype=bool)
    dataIds = pd.Series(np.ones_like(data.index), index=data.index, dtype=bool)
    return data[~(outlierIds & dataIds)], outlierIds, dataIds

# also can use isolation forest
detector = sklearn.pipeline.make_pipeline(
    sklearn.preprocessing.PowerTransformer(method="yeo-johnson", standardize=True),
    sklearn.ensemble.IsolationForest(n_estimators=6),
)

reg = sklearn.linear_model.BayesianRidge(alpha_1=5e-03, alpha_2=7e-05, lambda_1=1e-06, lambda_2=1e-5)

kFold = sklearn.model_selection.KFold(n_splits=5, shuffle=True, random_state=84) # maybe lower depending on data


hyperparameter = "isolationforest__contamination"
hyperparameterValues = [None] + np.arange(0.01, 0.25, 0.01).tolist()
noOutliersMedae = {}


for k, (trainIds, validIds) in enumerate(kFold.split(xData, yData), start=1):

    xTrain = xData[xData.index != validIds[0]]
    yTrain = yData[yData.index != validIds[0]]

    xTrain = xData[xData.index != validIds[1]]
    yTrain = yData[yData.index != validIds[1]]

    xValid = xData[xData.index == validIds[1]].append(xData[xData.index == validIds[0]])
    yValid = yData[yData.index == validIds[1]].append(yData[yData.index == validIds[0]])
    # default loss
    reg.fit(xTrain, yTrain)
    defaultMedae = sklearn.metrics.median_absolute_error(yValid, reg.predict(xValid))

    for hpValue in hyperparameterValues:
        if hpValue is None:
            # save default
            noOutliersMedae[f"0 - {k}"] = defaultMedae
            continue

        # new params
        detector.set_params(**{hyperparameter: hpValue})
        xNoOutliers, outlierIds, dataIds = remove_outliers(pd.DataFrame(xTrain), detector)

        yNoOutliers = yTrain[~(outlierIds & dataIds)]


        reg.fit(xNoOutliers, yNoOutliers)
        cleanMedae = sklearn.metrics.median_absolute_error(yValid, reg.predict(xValid))
        noOutliersMedae[f"{hpValue} - {k}"] = cleanMedae


# clean data

noOutliersMedaeClean = {}
outliersMedae = {}
for i in noOutliersMedae.keys():
  strTemp = i.split()

  if strTemp[0] == '0':
    noOutliersMedaeClean[strTemp[2]] = noOutliersMedae[i]
  try:
    outliersMedae[strTemp[2]].append(noOutliersMedae[i] - noOutliersMedaeClean[strTemp[2]])
  except:
    outliersMedae[strTemp[2]] = []
    outliersMedae[strTemp[2]].append(noOutliersMedae[i] - noOutliersMedaeClean[strTemp[2]])

X_LABEL = "Contamination"
```

```python

# Initialises settings of the graph
plt.figure(figsize=(7,4))

plt.xlabel(X_LABEL)
plt.ylabel('Median Absolute Error')

# Plots the regression
xPlot1 = hyperparameterValues

plt.plot(xPlot1, outliersMedae['1'], label='Fold 1')
plt.plot(xPlot1, outliersMedae['2'], label='Fold 2')
plt.plot(xPlot1, outliersMedae['3'], label='Fold 3')
plt.plot(xPlot1, outliersMedae['4'], label='Fold 4')
plt.plot(xPlot1, outliersMedae['5'], label='Fold 5')

plt.legend()
plt.show()

"""# Removing outliers for best hyperparameter"""

# set best outlier detection and remove

detector.set_params(**{hyperparameter: 0.04})

xNoOutliers, outlierIds, dataIds = remove_outliers(pd.DataFrame(xData), detector)

yNoOutliers = yData[~(outlierIds & dataIds)].reset_index(drop=True)

xNoOutliers = xNoOutliers.reset_index(drop=True)

# test efficacy with outliers removed

losses = []

reg = sklearn.linear_model.BayesianRidge(alpha_1=5e-03, alpha_2=7e-05, lambda_1=1e-06, lambda_2=1e-5)

for i in range(len(yNoOutliers["homelessness"])):
  xTrain = xNoOutliers[xNoOutliers.index != i]
  xTest = xNoOutliers[xNoOutliers.index == i]
  yTrain = yNoOutliers[yNoOutliers.index != i]
  yTest = yNoOutliers[yNoOutliers.index == i]

  reg.fit(xTrain, yTrain)
  preds = reg.predict(xTest)
  losses.append(mean_absolute_percentage_error(preds, yTest))

np.mean(losses)

"""# Forecasting"""

# linear models for other predictor variables
popReg = linear_model.LinearRegression()
popReg.fit([[pop["year"][i]] for i in range(len(pop["year"]))], pop['pop'])

rentReg = linear_model.LinearRegression()
rentReg.fit([[socialRent["year"][i]] for i in range(len(socialRent["year"]))], socialRent['socialRent'])

incomeReg = linear_model.LinearRegression()
incomeReg.fit([[lowIncome["year"][i]] for i in range(len(lowIncome["year"]))], lowIncome['lowIncome'])

# get predictions for each variable
homes = pd.read_csv("/content/homes_manchester.csv")

homes = pd.DataFrame(data={
    "years": homes["years"],
    "homes": homes["homes"]
})

populationForecast = []
rentForecast = []
incomeForecast = []

for i in homes["years"]:
  populationForecast.append(popReg.predict([[i]]))
  rentForecast.append(rentReg.predict([[i]]))
  incomeForecast.append(incomeReg.predict([[i]]))

# homelessness predictions
reg = sklearn.linear_model.BayesianRidge(alpha_1=5e-03, alpha_2=7e-05, lambda_1=1e-06, lambda_2=1e-5)
reg.fit(xNoOutliers, yNoOutliers)
homelessness = []

for i in range(100):
  homelessness.append(reg.predict([[homes["homes"][i], int(incomeForecast[i]), int(rentForecast[i]), int(populationForecast[i])]])[0])

# predictions in range

homeless = pd.DataFrame(data={
    "year":homes["years"][25:82],
    "homelessness":homelessness[25:82]
}).reset_index(drop=True)

#real data

yData_2 = pd.DataFrame(data={
    "year": data["Year"][15:25],
    "homelessness": data["Homelessness"][15:25]
}).reset_index(drop=True)

X_LABEL = "Year"
Y_LABEL = "Homeless"
# Initialises settings of the graph
plt.figure(figsize=(14,4))

plt.xlabel(X_LABEL)
plt.ylabel(Y_LABEL)

# Plots the regression
xPlot0 = np.linspace(2008, 2018, 10)

xPlot1 = np.linspace(2018, 2074, 2074-2017)
plt.plot(xPlot1, homeless["homelessness"], label='Prediction')

plt.scatter(yData_2["year"], yData_2["homelessness"])

plt.legend()
plt.show()
```

6.3. Part III: Rising from This Abyss

Question3.py

```
# -*- coding: utf-8 -*-
"""q3rentincrease.ipynb

Automatically generated by Colaboratory.

Original file is located at
    https://colab.research.google.com/drive/1wcB9uKb92ALVD0dPhutIZq1mDmoaky2r
```

```python
"""

# Used for styling of graphs
import matplotlib.pyplot as plt
from matplotlib.collections import LineCollection

import scienceplots
plt.style.use(['no-latex'])

import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
# houses built

COSTOFHOUSE = 91260
RENTINCREASE = 20
INITIALHOUSES = 63276

# houses built per year
def housesBuilt(numOfHouses):
  return 12* (numOfHouses * RENTINCREASE) / COSTOFHOUSE

houses = [INITIALHOUSES for i in range(2)]

# total houses per year
for i in range(15):
  houses.append(int(houses[-1] + housesBuilt(houses[-2])))

housesbuilt = []

# houses built (houses )
for i in houses:
  housesbuilt.append(i-INITIALHOUSES)

year = [i for i in range(17)]

# question 2 results
homeless = pd.read_csv("/content/homeless.csv")
homeless = np.array(homeless["homelessness"][:17]) - np.array(housesbuilt)

# graphs

fig, ax1 = plt.subplots( figsize=(8,6), dpi=400)
ax2 = ax1.twinx()

ax1.set_ylim(bottom=0, top=3000)
ax2.set_ylim(bottom=0, top=2300)

ax1.step(year, housesbuilt, label='Houses built with rent increase = 20')
ax1.plot(year, housesbuilt, 'o--', color='grey', alpha=0.3)

ax2.step(year, homeless, label='Homeless population (right)', color='orange')
ax2.plot(year, homeless, 'o--', color='grey', alpha=0.3)

plt.grid(axis='x', color='0.95')
ax1.legend(loc="lower right")
ax2.legend()


plt.title('Houses built and homeless by year')
plt.show()
```

```python
# Imports libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sb

# Used for styling of graphs

import scienceplots
plt.style.use(['science', 'grid', 'no-latex'])

# Reads data from excel - swap "Brighton" & "Manchester" to switch
df = pd.read_excel('/content/data.xlsx', sheet_name='data')
corrMatrix = df.corr().abs()

plt.figure(figsize=(4,3), dpi=500)

# Plots the correlation matrix
ax = sb.heatmap(corrMatrix, annot=True, cmap="Greens")
ax.tick_params(axis='x', rotation=45)
```