A Tale of Two Crises: The Housing Shortage and Homelessness

Team #17379
Mathworks Math Modeling Challenge 2024

March 3, 2024

Executive Summary

To the Secretary of the United States Department of Housing and Urban Development,

As the lack of affordable housing continues to plague the United States (U.S.), homelessness remains a devastating issue that countless cities encounter. Throughout the U.S., housing prices have roughly doubled in the last 15 years[1] while homelessness has increased by 6% since 2017[2]. It's evident that these problems will persist unless public policy is catered towards change. Our team aims to predict - in Albuquerque, New Mexico and Seattle, Washington - the number of homeless individuals and the number of homes that will be available on the housing market 10, 20, and 50 years into the future. We also determine the most important factors that lead to homelessness and provide recommendations to mitigate the losses due to homelessness.

We first predict the housing supply in both housing markets. We approach the problem with an agent-based model to simulate the interactions of four types of agents: families, renters, investors, and homeowners. With each agent, we assign a purpose, income, and behavior based on various factors such as house price growth, mortgage interest rate, rent price, and market conditions. Based on current income brackets, we use a representative sample size of 1000 homeowners in the market to model the proportions of available homes. In Seattle, we obtain that in 10 years, 170 houses will be available to rent, 422 houses will be up for sale, and 987 total houses will be available. In 20 years, 173, 89, and 325 houses, respectively. And in 50 years, 161, 398, and 888 houses, respectively. In Albuquerque, we obtained that in 10 years 204 houses will be available to rent, 466 houses will be up for sale, and 354 total houses will be available. In 20 years, 230, 113, and 354 houses, respectively. In 50 years, 230, 432, and 1094 houses, respectively. Generally, we have a medium-high confidence in our results. Running a sensitivity analysis on our results based on income and housing price growth rate, we found that while both factors impacted both cities significantly, Albuquerque was relatively less sensitive to shifts in these factors.

We then predict the future trends of homelessness. Using a Fourier transform we are able to model the fluctuations of homelessness in Seattle and Albuquerque, and obtain the homeless population percentage trend in the future. This also requires us to regress the total populations in these two cities as well, which we did with a linear regression. We find that in 10, 20, and 50 years, Seattle would have homeless populations of 16963, 31763, 39520 people respectively. Albuquerque would have homeless populations of 1829, 2114, 2883 respectively.

Lastly, we utilize a random forest feature analysis to determine the most important factor relevant to homelessness in Seattle: civilian participation in the labor force (0.19). After determining the most important factor, we determine the importance of the first factor compared to other factors via a threshold value. From 6 years of complete states, we can create a holistic transition matrix resistant to shocks like natural disasters. Overall, we recommend taking steps to promote the civilian labor force by mandating employee benefits such as a health insurance plan, paid time off (PTO), retirement benefits, and life insurance within companies can incentivize people to take up jobs [3]. In conclusion, by understanding the dynamics of housing supply and homelessness, policymakers can implement targeted interventions to foster more resilient communities.

Sincerely,
Team 17379

Contents

Global Assumptions

1. **Homelessness is defined as an individual or family who lacks a fixed, regular, and adequate nighttime residence.** We take the definition in the official U.S. Code from Title 42[4] in order to establish uniformity and abide by governmental policies. Primary nighttime residence that is a public or private place not designed for or ordinarily used as a regular sleeping accommodation for human beings, including a car, park, or camping ground are considered inadequate.

2. **All homeless people want to buy a home and the reason they cannot is because of monetary constraints.** We consider from Maslow's hierarchy of needs[5] that shelter is a basic physiological need for human survival. Thus, all homeless people are motivated by survival instincts to find housing and allocate their income to prioritize this need.

3. **There will be no major legislation or polices are put into effect in the next 50 years that will impact the housing market.** Legislation can impact the number of homes bought and the size of the homelessness population through subsidies, emergency/transitional housing, land grants, etc. However, it is beyond the scope of our paper to predict and account for political decisions and agendas.

4. **All economic decisions made by consumers are financially optimal and the real estate industry will cater towards their needs[5].** The average consumer always has their financial interests in mind. It is reasonable to assume that such a consumer makes the most optimal decision. Thus, businesses need to adhere to the demands of these consumers to sustain themselves.

5. **There will be no major shifts in the U.S. economic landscape.** It is possible for the economy of the U.S. to take a drastic turn, thus altering the housing market greatly. However, as this is unlikely to happen, we assume the general economic makeup of the U.S. remains constant, and no major inflationary or recessionary periods occur.

1 Part I: It Was the Best of Times

1.1 Restatement of the Problem

In this problem, we are tasked with predicting the changes in housing supply in Seattle, Washington and Albuquerque, New Mexico. We predict trends for 10, 20, and 50 years in the future and indicate our level of confidence in our predictions.

1.2 Assumptions

1. **The ratio of houses to people will stay approximately constant throughout the next 50 years.** There are currently no large-scale housing construction contracts or population fluctuations. Thus, we can assume that the population inflow and new constructions will be balanced, and there will be no significant change in the ratio.

2. **There is no psychological cost of renting a house.** Due to the housing crisis and increasing debt, home ownership is infeasible for many and thus, it is a choice based on flexibility and personal preference rather than severe financial burden. Thus, there is no psychological difference/advantage to buying a house. [6]

3. **Renters and buyers have the same desired bid mentality.** All individuals purchasing a house or any asset prioritize saving money and increasing profit. Thus, there is no significant difference in the cost-related mindset of a renter compared to a buyer.

4. **Buyers are willing to take the maximum loan amount offered.** [7] Rising prices and consumer stress have financially burdened Americans, making loan-taking extremely prevalent. Thus, we can assume that to mortgage a house, buyers will accept the maximum loan offered.

5. **Renters are typically in the lowest income bracket, buyers in the middle, and buy-to-let (BTL) investors in the highest.** Since renters cannot rent or provide a down payment for a house, buyers mortgage a house, and buy-to-let investors buy multiple houses to rent to others, we can assume the groups fall in the respective income brackets.[8]

1.3 Variables/Equations

Our model requires the assignment of the various types of intent for people entering the housing market. We define each person and their purpose by the following label, corresponding purpose and the income bracket the majority of the group is located in.

Table 1: Types of People

Label	Type	Bracket
0	Families	Middle Class
1	Renters	Lower Class
2	Investors	Upper
3	Homeowners	Middle Class

After assigning such purposes, we will analyze each individual's possible behavior and the variables affecting such behavior. To do this, we need to consider the most direct and impactful causes and their corresponding behaviors.

1.3.1 Families

Those in families primarily seek to live in a house and claim full ownership from the previous owners. They typically come from the middle class. They have enough income to find a permanent residence by submitting down payments and mortgages. However, they are not wealthy enough to own multiple of such properties for others to rent out[9]. Therefore, we can estimate a general income range in which we provide those in the middle class group. While this is the primary internal factor affecting buyers, we also need to consider external factors such as the fluctuations in mortgage payments, house price growth, and possible loans that homeowners can utilize.

By far the most important factor, income determines the affordability of house for a given person. Those with higher incomes will generally accept a higher housing price. We can define the price they are generally willing to accept as the buyer's desired bid. By adjusting the general bid equation [], we come up with following formula:

$$p = \frac{i \cdot e^{N(0,\epsilon)}}{1-g} \tag{1}$$

In this equation, g represents the expected annual house price growth based on previous years, the N represents the random noise we implement to simulate an unpredictable real-world environment, and i representing the buyer's income. This equation will give us a buyer's point of bidding which will help determine their willingness to pay, thus influencing the dynamic of the market.

Whether or not a family is willing to rent or buy a home depends on this point of bidding and the willingness to sell of sellers in the local housing district[9][10]. After all, there are many families living in rental units, especially in Seattle. We can denote this probability as P and use a the following equation:

$$\frac{1}{1+e^{-K(Cr-(m-g\cdot p))}} \tag{2}$$

Cr is the median current market price for renting per month, m is the median annual mortgage payment for houses in the area, g is the same as in Equation 1, the annual expected house price growth, K is an adjustable parameter that we hyper-tuned while testing, and p is calculated from Equation 1, the buyer's desired point of bidding.

1.3.2 Renters

Renters primarily seek a temporary place for them to stay while they build up their wealth and eventually move out[9]. For this reason, it is safe to assume that renters make up the lowest bracket of income and typically do not have enough money to afford mortgages, down payments, and may even need to take out loans to afford rent. Similar to families, renters' decision to rent out a living space from a landlord depends primarily on income. Because fluctuations in rent depend on such landlords, it is safe to assume that the change in rent revolves solely around the income of the renter in question. This income accounts for inflation, the primary cause for changes in rent outside of the housing market[]. With this in mind, we came up with the following equation

$$pr = 0.3 * I \tag{3}$$

The 30 percent rule is a general rule of thumb for those seeking to rent [11], so we can determine their willingness to pay for rent as simply 30 percent of their monthly income.

1.3.3 Investors

Investors are those who choose to buy a house, not for the sake of living, but for the purpose of re-renting it out for other buyers[9]. Because of this extremely quick process of transitioning from buying to selling, we can simply count an investor as a buyer who, when successfully purchases a home, converts to a seller instantaneously. However, these sellers are different from homeowners, because investors are incentivized to rent their homes while homeowners look to sell their house entirely. Therefore, we will only consider investors who rent. The renting price depends solely the median renting price from other investors in the area, as an investor cannot go against the supply-demand curve. Therefore the following equation is produced with m representing the median price, and N representing the Gaussian noise.

$$p = m + N(0, \epsilon) \tag{4}$$

This will account for randomness, but stay pretty consistent with the market price. We will explore more about the factors influencing sellers in the next section.

1.3.4 Homeowners

Homeowners are those who own a home and have a choice to sell their house or keep ownership. After this decision is made, if they have chosen to sell the house, they would also need to determine what price point to set it at. The following equation solves this preliminary decision of whether a homeowner lists their house for sale.

$$p = c \cdot (1 + a \cdot (0.05 - N_h) + b \cdot (0.03 - i)) \tag{5}$$

In this equation, p, the price point the house is set, is determined by N_h, the number of houses on the market per capita that month, i, represents the mortgage interest rate that month, and a, b, and c are constants that we determined through thorough testing. We adjusted these parameters to limit all possible values to be between the interval $[0, 1]$, because this value is a probability. After determining this probability, if the seller is willing to sell their home, we can proceed to the price in which they will list their home on the market. This equation is very similar to Equation 4, with the only difference being selling price instead of renting price.

$$p = m + N(0, \epsilon) \tag{6}$$

The equations in this section describe the behaviors of the different categories of people and how we calculate the With all this in mind, all we need to do is combine these behavior influences and substitute values based on the month we are currently analyzing. We will describe how we plan to simulate these agents in the following section.

1.4 Model Development

To predict changes in the housing supply over time, we utilize an agent-based model (ABM). This computational model simulates the operations and interactions of autonomous agents within a given environment. Each agent is assigned initial values (starting conditions) and then placed into a spatial field. The agents then follow a set of rules that determine their interactions with other agents including movement, resource allocation, and decision-making. Over time, the simulation generates feedback loops that give rise to complex patterns at a macroscopic level. Our ABM consists of four agents: families, investors, renters, and homeowners.

Before diving into the specifics, please keep in mind all functions referenced below are explained in closer and more technical detail in the section prior. For the sake of efficiency, we will avoid reexplaining each formula and rather paraphrase their effects to reference them.

Moving from here, after coding these agents in, we adjust their exact behaviors by employing the function mentioned prior. For families, they will search for a home depending on willingness to buy a home, calculated via the equation mentioned prior. After determining they wish to search for a home, they will then set their maximum price for a home, again based off of the function prior based on their income and annual price growth rate.

Investors on the other hand also search for a home, but with the few restrictions mentioned prior. We track their years since investment, and only past 3 years can they reinvest in a home. Then, if they choose to do so, they again employ the same functions as families, of course with just a significantly higher income. The price they choose to then rent out their newly acquired home is calculated via the rent price function, a function of income and market conditions. However, every few years, like sellers, based off the same probability function they will decide to sell their homes. This is justified as for investors, if market conditions are very good and home prices are high, even though they entered the investment searching for return through rent, they have a chance of taking the quick route and selling the home for quick profit.

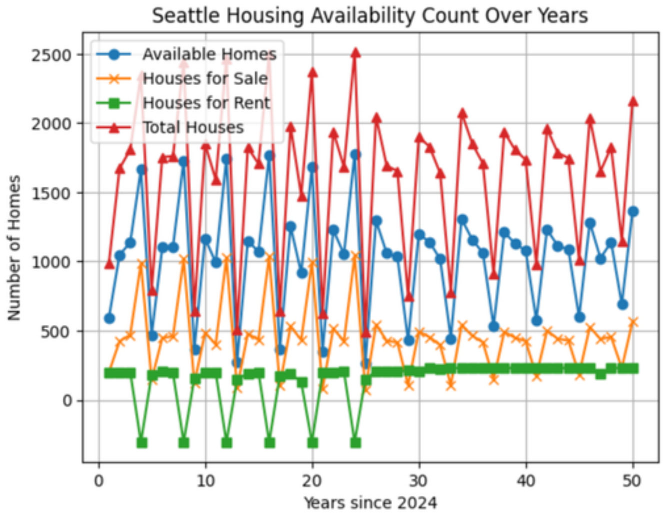
Renters are quite simple. They search only for a home to rent based on their max rent price, and then rent that home if one from an Investor is available. Then, in 2 years, they are sent back to the market, once again searching for a home to rent.

Finally, home-owners are the ones putting their homes on the market. From our function regarding probability of a homeowner placing their home on sale, we test that each year and obtain a handful of home-owners ready to sell. When they do wish to sell, they then run our function to find selling price, determined via market conditions during that year. Finally, their home stays up for sale for an investor or family to buy.

From here, we merely need to introduce the initial starting sample to be representative of the housing market of Seattle and Albuquerque, and iterate from there. Looking at Seattle's housing market, from their last census we found their income distributions were 10.1% low income, 34.4% middle income, and 55.5% high income[12], meaning our distribution of families, investors, and renters would be 101, 344, and 555 respectively given 1000 people, of which a same group would be homeowners to match them. Albuquerque's starting conditions would be 165, 545, and 290[12] respectively From this, letting the simulation run 50 years, we get the following.

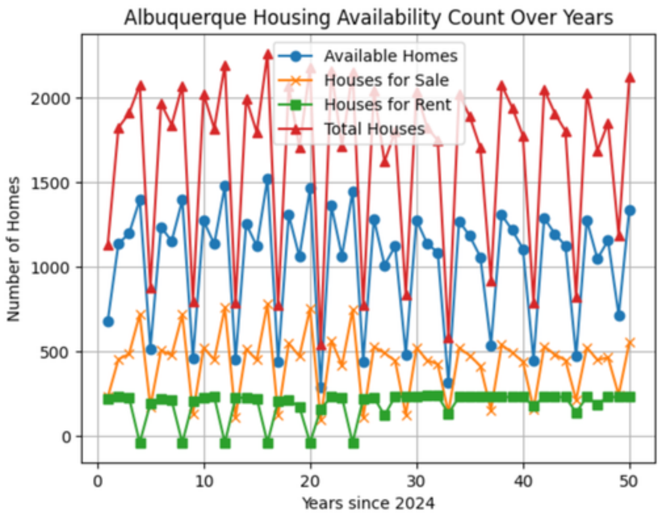
1.5 Results

The results we obtain are shown in the below graph. As you can see, there is rough oscillation, as we did not take into account introduction of new homes being built into our housing analysis. However, this would be a quick addition given an accurate model to represent homes built yearly, taking into account government policy and sentiment alongside construction costs and investment. If added, it would likely change out model a bit, altering the houses to sell number while also likely incentivizing more people to come to the market.



Years Since 2024	Houses to Rent	Houses for Sale	Available Homes
10	170	422	987
20	173	89	325
50	161	398	888

Model and Predictions for Seattle



Years Since 2024	Houses to Rent	Houses for Sale	Available Homes
10	204	466	1109
20	174	113	354
50	230	432	1094

Model and Predictions for Albuquerque

We chose to run our simulation on a sample size of 1000 people as our proportions represent a representative population from the respective cities. This, paired with our assignment being random, allow us to use this sample size instead of the entire population of each city. Thus, the proportions present in these results also apply to the entire city housing market - it just needs to be scaled up to the entire population of the city instead of just 1000 people. We were unable to obtain true numbers as our computing power and time was insufficient, but our same methodology can be run with scaled up initial population parameters - and thus results would be some multiplier of our current results - to obtain such values.

1.6 Confidence

The following are our estimates for our level of confidence in the various portions of our model.

1.6.1 Structure and Code Accuracy: High Confidence

Our code is structured sensibly into classes for each agent and function, and has clear locations for decision making. The functions make sense and simulation seems to work as expected, leading us to be highly confident in the fact that our code works the way we intend it to. The architecture of both our environment and agents are accurate in representing the behaviors through various formulas which are calculated and justified.

1.6.2 Agent Behavior: Medium to High Confidence

The behaviors of agents like homeowners deciding to sell, families looking for homes, investors putting homes to rent, and renters looking for renting seem reasonably implemented based on the provided functions and level of complexity we were able to implement. While our decision making parts do have some oversimplification, leading us to have slightly lower confidence, it still should accurately represent the model to a decent degree. In the real world, there are very little to none agents that behave poorly in following ethics and the the theoretical model of supply-demand.

1.6.3 Market Dynamics: Medium Confidence

The simulation tries to implement market dynamics such as house price growth, changes in housing availability, and rental demand. These dynamics are influenced by external factors native to our agents and simulation like income and house price growth rates. While the dynamics do seem decently accurate, we do recognize that modeling the full economy is extremely difficult and therefore cannot comfortably say that our modeling of market fluctuations is extremely precise or empty of potential flaws.

1.6.4 Graphical Representation: High Confidence

The graphs we have are representative of our data, meaning we are extremely confident in the graphical representations submitted.

1.6.5 Mathematical Functions: Medium Confidence

Mathematical functions like calculating house prices, mortgage payments, and rent are used in our decision making processes. While these functions are reasonable and based on general US data, there is possible error as Seattle and Albuquerque likely have a few different parts or constants due to city-specific things. This means that while our model is roughly accurate, there is a slight lowering of confidence due to our non-specific use of math functions. These functions are backed by creditable research papers.

1.6.6 Random Processes: Medium Confidence

Random processes are used in many parts of the code, specifically such as generating noise or tracking random chance of a homeowner to list their home for sale each year. While randomness is essential for simulating real-world uncertainty, the exact effects of these random processes on the simulation outcomes might need further analysis, and averaging over more iterations and attempts of the model.

1.7 Sensitivity Analysis

For our sensitivity, we looked at changing two factors that did not see change in out previous modelling. Looking at the two cities, we saw their difference in income composition effect the results. Therefore, for sensitivity, we opted to look into change of model results depending on initial income starting locations, looking at impact of how poor a community is, and also the growth rate of home prices, looking at possible inflation effects or gentrification threats. We set up 2 arrays with potential varying values for these 2 factors, and ran the model through them. We varied between the following:

Income Levels: \$20,000 to \$100,000 in increments of \$1000.
Growth Rate: Ranging from 1% to 5% by 0.5%..
Average Availability: Spanning from 50% to 90% by 1%.

Looking at some samples from our testing, note that when Income was 55000 and Growth Rate was 0.04, Average Home Availability was 8714.62. Another example is that when Income was 50000 and Growth Rate was 0.04, Average Home Availability was 5118.86. With our vast amount of values, we ran a polynomial regression on them to find their sensitivities, resulting in the following results.

Seattle Sensitivity:

- *Low Growth Rate Sensitivity*: A 1% change in economic growth results in a 0.72% change in housing demand.
- *Medium Growth Rate Sensitivity*: A 1% change in economic growth leads to a 1.13% change in housing demand.
- *High Growth Rate Sensitivity*: A 1% change in economic growth results in a 1.65% change in housing demand.
- *Low Income Sensitivity*: A 1% change in income results in a 0.76% change in housing demand.
- *Medium Income Sensitivity:* A 1% change in income leads to a 1.03% change in housing demand.
- *High Income Sensitivity*: A 1% change in income results in a 1.33% change in housing demand.

Albuquerque Sensitivity:

- *Low Growth Rate Sensitivity:* A 1% change in economic growth results in a 0.78% change in housing demand.
- *Medium Growth Rate Sensitivity:* A 1% change in economic growth leads to a 0.88% change in housing demand.
- *High Growth Rate Sensitivity:* A 1% change in economic growth results in a 0.97% change in housing demand.

- *Low Income Sensitivity:* A 1% change in income results in a 0.26% change in housing demand.
- *Medium Income Sensitivity*: A 1% change in income leads to a 0.44% change in housing demand.
- *High Income Sensitivity:* A 1% change in income results in a 0.83% change in housing demand

1.8 Strengths and Weaknesses

A strength of the ABM is that it can capture emergent phenomena. Its ability to provide holistic information about the system's dynamics and not just the apparent behavior of individual agents makes it especially insightful. The model is also capable of representing heterogeneous agents with diverse characteristics and decision-making processes, thus reflecting the variability within real-world populations of renters, buyers, BTL investors, and sellers.

Another strength of the ABM model is its ability to be easily adjusted upon further improvements. If given more time or a stronger computer, we could easily implement bits and pieces of complex decision making, and even incorporate more agents, to further make our model more realistic while preserving the original framework which persists in use throughout these improvements. It is a model easily built upon, without needing to often change old parts or do major structural overhauls.

One weakness is that our model does not address the relation between affordable housing and homeless population locations. House prices vary with geographic location. However, we assume that homeless people have more freedom in moving distances to purchase affordable housing, as they have comparatively less factors tying them down. Thus, the impact of geographical location on our results should be near negligible. ABMs may also be sensitive to initial conditions,requiring precise sensitivity analysis to assess the robustness of the results.

Additionally, another weakness of our model is our lack computation power. Due to this, the decision making processes we implemented for our agents were significantly simplified versions of systems we wanted to implement, yet were unable as upon implementation computational time exceeded our present capacity. This means that our decision making is simplified, yet it should still decently model a real life scenario.

One final weakness of our model is the generalized nature of our formulas. As our data was not exact to Seattle nor Albuquerque, both share the same formulas for buyer price, sentiments, probability of selling homes, and more. This means that while our initial starts for the two cities were different, and a few constants were altered based on city-specific data we could locate, a part of the model was consistent. If given more specific data on both cities, we could change our formulas in decision making to be much more city-specific, likely allowing us to get much more accurate results along with reliable sensitivity results.

2 Part II: It Was the Worst of Times

2.1 Restatement of the Problem

For Seattle, Washington and Albuquerque, New Mexico, we predict changes in the homeless for the next 10, 20, and 50 years.

2.2 Assumptions

1. *Homeless populations fluctuate constantly over the years*[12] Homelessness is a problem in which policy towards mitigating the issue tightens when the problem worsens and loosens as it gets better. Thus, it would fluctuate accordingly as when homelessness becomes extremely problematic, politicians will seek to solve the problem. Meanwhile, when homelessness eases up, policies will loosen as a result, leading to near-sinusoidal behavior[13].

2. *The level of complexity and order in our Fourier series is sufficient.* While increasing the complexity would still slightly improve the accuracy of our model, we note that further increases in complexity in the Fourier series provides severely diminishing marginal return in accuracy for computation time and resources, making the move not worth the time.

2.3 Model Development

2.3.1 Fourier Transform

Our longitudinal data for the total number of homeless people spans from 2012 to 2022 and varies greatly in a sinusoidal-esque pattern. Therefore, given this fluctuating data set - justified in Assumption 1 - we decided upon a Fourier transform to formulate a pattern to generalize and model these trends. This works better than other options such as polynomial, linear, and logistic regressions as these other models do not account for this sinusoidal behavior. However, there still exists a general trend upwards or downwards even if the specific data points fluctuate on a year-by-year basis. Thus, utilizing a Fourier transform alongside the linear increase will be able to best model the above features, and the general equation for our model is as such:

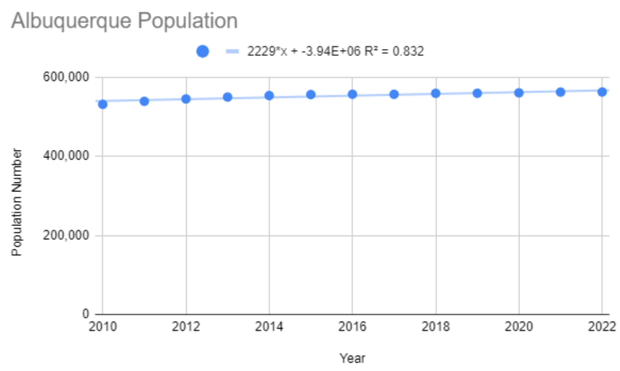
$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{k} \left[a_n \cos\left(\frac{n\pi x}{L}\right) + b_n \sin\left(\frac{n\pi x}{L}\right) \right]$$

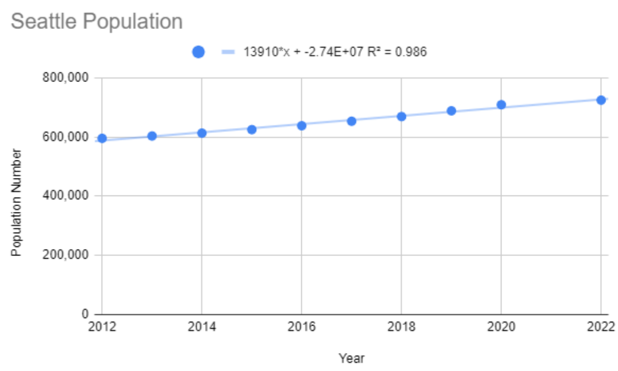
The code for which can be found in the Appendix.

2.3.2 Prediction for Total Population of Seattle and Albuquerque

To run our Fourier Transform requires us to model the trend of the percentages of homeless people in the two cities rather than the absolute number of homeless people. This is because given our limited number of data points (around 10 years), regressing the trend purely based upon absolute number of homeless people would lead to extremely biased results as the absolute number of homeless people can be influenced by various factors such as population growth, economic changes, and policy interventions[14][15][16], which may not be captured accurately within a short time frame. By modeling the trend of percentages, we normalize the data with respect to the population size, providing a more consistent basis for comparison over time. This approach allows us to focus on the relative changes in homelessness rates, which are more indicative of underlying societal trends, rather than being confounded by external factors.

Thus, we being to regress the total populations of Seattle and Albuquerque, as percentage homeless only makes sense in the context of the entire populations of the cities. Due to how little data we have, we decide to utilize a linear regression as the end behavior of a linear regression is less extreme than other regressions like exponential or polynomial, especially when regressing a large amount of years forwards (50 years forwards compared to our 10 years of data). Thus, we regress our data and achieve the following lines of best fit:





2.4 Results

2.4.1 Regression Results

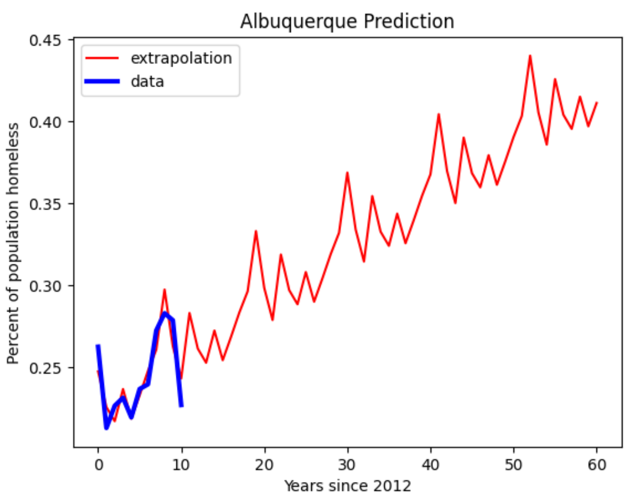
Table 2: Albuquerque Population Prediction

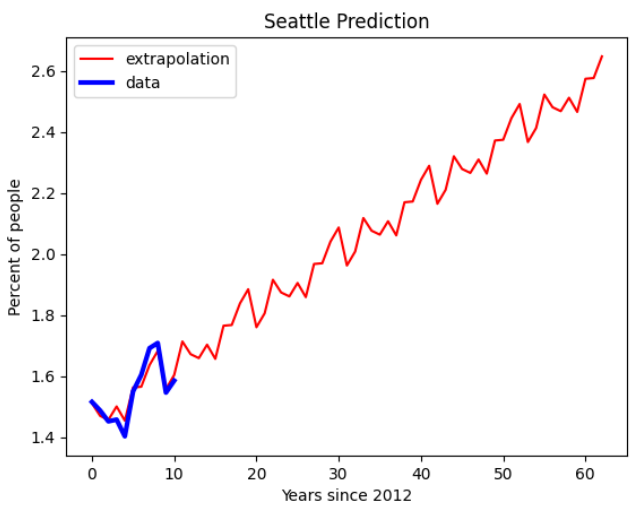
Year	Population
2034	593786
2044	616076
2074	682946

Table 3: Seattle Population Prediction

Year	Population
2034	963826
2044	1106716
2074	1535386

2.4.2 Fourier Results





The r^2 value for Albuquerque is 0.769 while the r^2 for Seattle is 0.921. We settled upon using $k = 4$ harmonics as it returned the best r^2 value while not being an over-fit (5 harmonics returned r^2 values of 1.0, a clear sign of that). This essentially means we run through the summation function 4 times. Thus, our predictions for 10, 20, and 50 years are as follows:

Years Since 2024	**Percent Homeless(%)**	**Number Homeless**
10	0.308	1829
20	0.343	2114
50	0.422	2883

Table 4: Predictions for Albuquerque

Year Since 2024	**Percent Homeless(%)**	**Number Homeless**
10	1.760	16963
20	2.087	31763
50	2.574	39520

Table 5: Predictions for Seattle

2.5 Strengths and Weaknesses

The Fourier transform method accounts for fluctuations within our data while still revealing the general trend. Moreover, it is a flexible model and can be extended to model longer time series such as 50 years into the future. The Fourier transform method is robust against outliers and noise in the data. By focusing on frequency components rather than individual data points, it can effectively filter out disturbances and anomalies, providing a clearer view of the underlying structure and dynamics. For our Fourier, we purposefully choose to normalize the homeless population to total population which makes our estimate more resistant to changes that proportionately effect both populations.

The model assumes parameters of an ideal scenario as it assumes periodicity. However, in many real-world scenarios, the function may exhibit non-periodic behavior or experience extreme changes in factors such as environmental conditions. This can pose challenges for the Fourier transform model, as it may not adequately capture the transient or non-stationary aspects of the signal. Consequently, the model's applicability and accuracy may be limited in dynamic or non-linear systems where such extreme changes are prevalent. Our linear regression also assumed similar ideal scenarios, resulting in similar weaknesses.

3 Part III: Rising from This Abyss

3.1 Restatement of the Problem

Using the results from Part I and II, we develop a model to target homelessness prevention based on different factors contributing to the problem. Our model will also account for possible unforeseen circumstances as shocks to the overall model.

3.2 Assumptions

1. *Homeless is experienced at a roughly equal level for all people, with all homeless people wishing to escape homelessness through any means possible.* While there is a slight possibility of people wishing to stay in homelessness, we assume this percentage is such a slight and rare piece of the overall homeless composition, that it can be disregarded.
2. *The results we have are applicable to the US.* While we considered both cities, we ultimately decided upon looking at the most volatile of the two cities. This allows us to be more confident in applying this model to the larger US, as having it function for a very volatile city assures us of its validity for the large part of the US's cities, save for perhaps a select extreme few.

3.3 Data

Source	Description
[17][18]	Describes monthly unemployment percentage
[19][20]	Describes number of active house listings per month
[21][22]	Describes median house price listing per square foot per month
[23][24]	Describes # of new private housing units authorized by building permits per month
[25][26]	Describes median home size in square feet per month
[27][28]	Describes median home listing price per month
[29][30]	Average weekly earnings of all employees per month
[31][32]	Average hourly earnings of all employees per month
[33][34]	Total civilian workforce per month
[12]	Population
[12]	Counts of 12 different unit types
[12]	Counts for Emergency, Transitional Housing, Sheltered or Un-sheltered Housing
[35]	30-year fixed rate mortgage average in the United States

Table 6: Data Used for Part 3

3.4 Model Development

When considering the data present and Part 1's sensitivity analysis, we choose to focus our model for this section on Seattle, Washington. This is because Seattle's housing market is significantly more volatile than Albuquerque's, and thus would be more in need of recommendations to mitigate the losses and risks associated with homelessness.

3.4.1 Random Forest Model

To complete this task, we choose to use a random forest model. Random forest is an ensemble learning model that uses multiple decision trees. Decision trees partition the feature space into regions until we are left with pure leaf nodes and finds the best split by maximizing entropy gains. In Random Forests, each tree is trained on a random subset of the training data and a random subset of features. The number of features split is according to $\sqrt{p}$ where p is the total number of features, aiming to produce diversity among trees. Increasing the number of trees generally leads to better generalization, but comes at computational costs. Each individual decision trees contributes a vote to the overall result. The most populous vote takes precedence. This ensemble learning methods reduces variance and improves performance over other learning models [36].

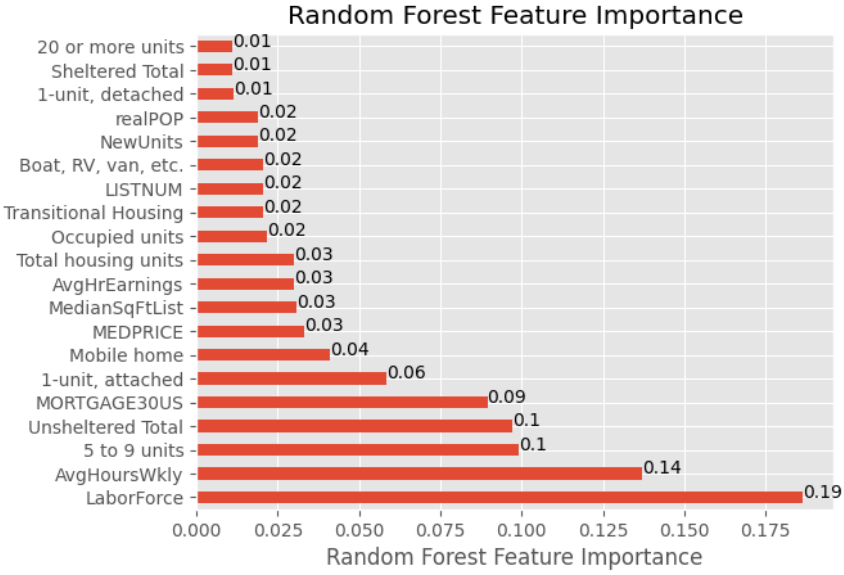
The random forest feature importance algorithm determines the relative importance of each of the 28 factors relevant to determining Seattle's homelessness. We use data from 2016-2022 and apply a train-test split of 80-20 to implement the algorithm. This means we use 80% of the data set to train the decision trees and 20% of the data set to test the model and achieve the relative feature importance. This algorithm will help us determine where we need to cater our long-term plan towards based on which factor influences homelessness the most.

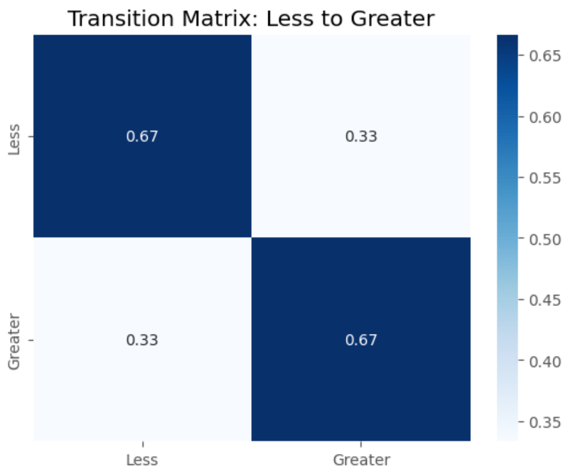
3.4.2 Transition Matrices

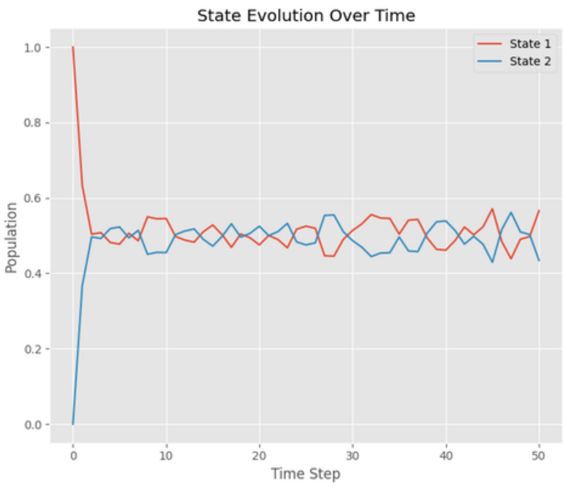
After determining the most important factors in the random forest model, we look at how much more important the first factor is compared to the other factors. Because there is a significant feature importance coefficient for the population in the labor force, and the r-squared value for the relationship between labor force and homelessness is extremely high, we can assume that homelessness is primarily affected by the number of people in the labor force. By considering this relationship only, we determine a threshold that separates the "significantly greater" homelessness levels which incorporate a significantly greater level of people in the labor force from the "significantly lesser" homelessness levels which incorporate a significantly lesser level of people in the labor force. By assigning each year in our dataset with an index, we can count how many transitions there are from a greater to lesser or vice versa. We can generalize the 6 year transitions to a holistic US homelessness transition matrix as it is recent sample. Then, we can simply multiply the initial state matrix represented by [1,0] in 2022 (because it shows lesser homelessness) by the transition matrix to see the fluctuation and general trend for homelessness.

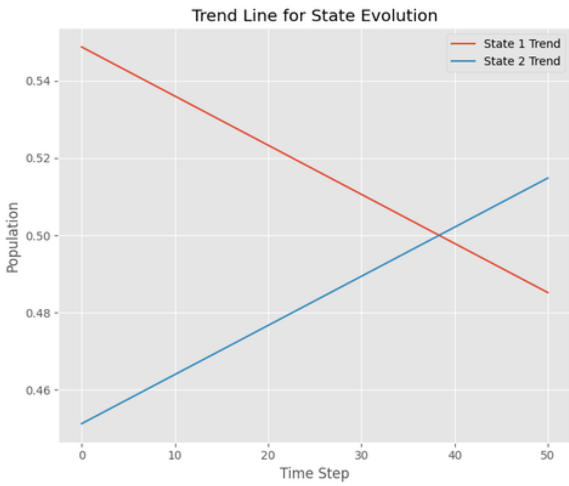
3.5 Results

The random forest feature analysis determined that the most relatively important factor was civilian labor force participation with a feature importance of 0.19. The determined regression equation is $y = 0.026x - 9879.84$ with an r-s .0quared value of 0.9887.









As we can see from the linear regression line, there is a clear threshold in the middle we can use to categorize homelessness as greater or lesser. This vertical line exists at $y = 435000$. We prove that the general transition matrix shows two-thirds of the years homelessness stays the same while in one-third of the years homelessness changes from lesser to greater or greater to lesser. Therefore, if our current year is in a greater state, over time, the homelessness will eventually converge in the middle of the homelessness magnitude, signifying a stable settling point for homelessness. We can conclude that homelessness will not continue to grow further but will oscillate between a greater and lesser state. Based on this information, we claim that although the current homelessness is rising in magnitude, homelessness in the following decades will begin to descend if we implement the correct solutions to combat such rise. The oscillation proved by the previous six years demonstrates that with a rise in issues requires a rise in public policy.

Seeing as the labor force participation rate was the most relevant in determining homelessness, there are several long-term plans the Seattle government could establish in order to lower homelessness rates. Mandating employee benefits such as a health insurance plan, paid time off (PTO), retirement benefits, and life insurance within companies can incentivize people to take up jobs [37]. This, in turn, will provide more Americans with a wage that they can use to sustain themselves and mortgage a house. Furthermore, the government can fund projects that promote industrialization such as construction and public space maintenance to create more jobs [38]. The more jobs that are available, the more people will join the work force and earn incomes. Lastly, the government can also increase access to education and promote worker training to qualify more homeless individuals for a wider range of jobs, ultimately taking them off the street and placing them in homes [39].

3.6 Strengths and Weaknesses

The random forest model is very robust in the face of noisy data, allowing it to accurately calculate results despite the existence of irrelevant and redundant data. It is also resistant to the effects of over-fitting because it aggregates the predictions of multiple decision trees, thus improving the model's generalization ability. The transition matrices allows our predictions to be more accurate as we account for changing conditions on a year-by-year basis and in the face of a shock. During a shock, which happens on a local scale, the state matrices describing the situation changes, but our transition matrices faces little movement. It also allows our model to be extended either by adding more factors or analyzing further into the future.

A weakness of the random forest is that it tends to be biased towards features with more unique values than others because such features tend to have more opportunities for split-points. Additionally, this model ignores the possible dependence between features as it is a uni-variate method that assumes independence - an aspect untrue in the real world. Our transition matrices being limited to only two states may also lead to bias within our findings.

References

[1] "Average Sales Price of Houses Sold for the United States". [Online]. Available: https://fred.stlouisfed.org/series/ASPUS

[2] State of Homelessness: 2023 Edition. [Online]. Available: https://endhomelessness.org/homelessness-in-america/homelessness-statistics/state-of-homelessness/

[3] A Menu of Options to Grow the Labor Force . [Online]. Available: https://www.americanactionforum.org/solution/menu-options-grow-labor-force/

[4] 42 U.S. Code § 11302 - General definition of homeless individual. [Online]. Available: https://www.law.cornell.edu/uscode/text/42/11302

[5] Essentials of Economics in Context. [Online]. Available: https://www.bu.edu/eci/files/2020/04/Chapter-4_Sample-Chapter.pdf

[6] The psychology of renting vs. buying. [Online]. Available: https://www.earnest.com/blog/rent-vs-buy/

[7] Delinquencies rise for credit cards and auto loans, and it could get worse. [Online]. Available: https://www.washingtonpost.com/business/2023/08/30/delinquencies-credit-auto-loans/

[8] Share of Income Needed to Pay Rent Increased the Most for Low-Income Households From 2019 to 2021. [Online]. Available: https://www.census.gov/library/stories/2023/03/low-income-renters-spent-larger-share-of-income-on-rent.html

[9] Real Estate Economics Course. [Online]. Available: https://ocw.mit.edu/courses/11-433j-real-estate-economics-fall-2008/

[10] Real estate economics. [Online]. Available: https://en.wikipedia.org/wiki/Real_estate_economics

[11] [Online]. Available: https://www.chase.com/personal/banking/education/budgeting-saving/how-much-income-should-go-to-rent

[12] A Tale of Two Crises, MathWorks Math Modeling Challenge 2024, curated data,. [Online]. Available: https://m3challenge.siam.org/kdfrldh/.

[13] Simulating the Evolution of Homeless Populations in Canada Using Modified Deep Q-Learning (MDQL) and Modified Neural Fitted Q-Iteration (MNFQ) Algorithms . [Online]. Available: https://www.researchgate.net/publication/341366247_Simulating_the_Evolution_of_Homeless_Populations_in_Canada_Using_Modified_Deep_Q-Learning_MDQL_and_Modified_Neural_Fitted_Q-Iteration_MNFQ_Algorithms

[14] Homelessness in California: Causes and Policy Considerations. [Online]. Available: https://siepr.stanford.edu/publications/policy-brief/homelessness-california-causes-and-policy-considerations

[15] Housing Supply and the Drivers of Homelessness. [Online]. Available: https://bipartisanpolicy.org/report/housing-supply-and-homelessness/

[16] Factors That Cause Homelessness. [Online]. Available: https://online.campbellsville.edu/social-work/factors-that-cause-homelessness/

[17] Unemployment Rate in Albuquerque, NM (MSA) . [Online]. Available: https://fred.stlouisfed.org/series/ALBU735URN

[18] Unemployment Rate in Seattle-Tacoma-Bellevue, WA (MSA) . [Online]. Available: https://fred.stlouisfed.org/series/SEAT653URN

[19] Housing Inventory: Active Listing Count in Albuquerque, NM (CBSA) . [Online]. Available: https://fred.stlouisfed.org/series/ACTLISCOU10740

[20] Housing Inventory: Active Listing Count in Seattle-Tacoma-Bellevue, WA (CBSA) . [Online]. Available: https://fred.stlouisfed.org/series/ACTLISCOU42660

[21] Housing Inventory: Median Listing Price per Square Feet in Albuquerque, NM (CBSA) . [Online]. Available: https://fred.stlouisfed.org/series/MEDLISPRIPERSQUFEE10740

[22] Housing Inventory: Median Listing Price per Square Feet in Seattle-Tacoma-Bellevue, WA (CBSA) . [Online]. Available: https://fred.stlouisfed.org/series/MEDLISPRIPERSQUFEE42660

[23] New Private Housing Units Authorized by Building Permits for Albuquerque, NM (MSA) . [Online]. Available: https://fred.stlouisfed.org/series/ALBU735BPPRIVSA

[24] New Private Housing Units Authorized by Building Permits for Seattle-Tacoma-Bellevue, WA (MSA) . [Online]. Available: https://fred.stlouisfed.org/series/SEAT653BPPRIVSA

[25] Housing Inventory: Median Home Size in Square Feet in Albuquerque, NM (CBSA) . [Online]. Available: https://fred.stlouisfed.org/series/MEDSQUFEE10740

[26] Housing Inventory: Median Home Size in Square Feet in Seattle-Tacoma-Bellevue, WA (CBSA) . [Online]. Available: https://fred.stlouisfed.org/series/MEDSQUFEE42660

[27] Housing Inventory: Median Listing Price in Albuquerque, NM (CBSA) . [Online]. Available: https://fred.stlouisfed.org/series/MEDLISPRI10740

[28] Housing Inventory: Median Listing Price in Seattle-Tacoma-Bellevue, WA (CBSA) . [Online]. Available: https://fred.stlouisfed.org/series/MEDLISPRI42660

[29] Average Weekly Earnings of All Employees: Total Private in Albuquerque, NM (MSA) . [Online]. Available: https://fred.stlouisfed.org/series/SMU35107400500000011

[30] Average Weekly Earnings of All Employees: Total Private in Seattle-Tacoma-Bellevue, WA (MSA) . [Online]. Available: https://fred.stlouisfed.org/series/SMU53426600500000011

[31] Average Hourly Earnings of All Employees: Total Private in Albuquerque, NM (MSA) . [Online]. Available: https://fred.stlouisfed.org/series/SMU35107400500000003

[32] Average Hourly Earnings of All Employees: Total Private in Seattle-Tacoma-Bellevue, WA (MSA) . [Online]. Available: https://fred.stlouisfed.org/series/SMU53426600500000003

[33] Civilian Labor Force in Albuquerque, NM (MSA) . [Online]. Available: https://fred.stlouisfed.org/series/ALBU735LFN

[34] Civilian Labor Force in Seattle-Tacoma-Bellevue, WA (MSA) . [Online]. Available: https://fred.stlouisfed.org/series/SEAT653LF

[35] 30-Year Fixed Rate Mortgage Average in the United States . [Online]. Available: https://fred.stlouisfed.org/series/MORTGAGE30US

[36] D. W. Gareth James, Trevor Hastie and R. Tibshirani, *An Introduction to Statistical Learning.* Springer, 2013.

[37] What are employee benefits? [Online]. Available: https://www.peoplekeep.com/blog/what-are-employee-benefits

[38] Labor force participation rate: Purpose, formula, and trends. [Online]. Available: https://www.investopedia.com/terms/p/participationrate.asp

[39] The evolution of labor force participation. [Online]. Available: https://www.brookings.edu/articles/labor-force-nonparticipation-trends-causes-and-policy-solutions/

A Code Used

```
#Part 1
import numpy as np
import random
import matplotlib.pyplot as plt
import math
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier, RandomForestRegressor
from sklearn.metrics import r2_score

class Agent:
    def __init__(self, agent_type, income, initial_savings=0):
        self.agent_type = agent_type
        self.income = income
        self.savings = initial_savings

class HomeOwner(Agent):
    def __init__(self, income, initial_home_value, mortgage_payment_function,
    house_price_function):
        super().__init__('homeowner', income)
        self.home_value = initial_home_value
        self.mortgage_payment_function = mortgage_payment_function
        self.house_price_function = house_price_function
        self.listed_year = 0

    def decide_to_sell(self, current_market_price, house_price_growth_rate, years,
     nh, income):
        mortgage_payment = self.mortgage_payment_function(years)
        price_willing_to_sell = self.house_price_function(self.income,
    house_price_growth_rate)
        probability_to_sell = sellerProb(nh, income)
        return random.random() < probability_to_sell

    def set_home_value(self, new_value):
        self.home_value = new_value

class Family(Agent):
    def __init__(self, income, house_price_growth_rate, E):
        super().__init__('family', income)
        self.house_price_growth_rate = house_price_growth_rate
        self.E = E

    def buy_home(self, current_market_price, house_price_growth_rate, years):
        N = np.random.normal(0, self.E)
        p = (self.income * math.exp(N)) / (1 - house_price_growth_rate)
        max_price = calculate_sell_price(self.income, self.house_price_growth_rate
    )
        if p >= current_market_price and max_price >= current_market_price:
            return True
        else:
            return False

```

```
class Investor(Agent):
    def __init__(self, income, max_house_price):
        super().__init__('investor', income)
        self.max_house_price = max_house_price
        self.last_purchase_year = -1
        self.own_home = True

    def can_buy(self, current_market_price, year):
        if year > self.last_purchase_year + 3 and self.max_house_price >= current_market_price:
            self.last_purchase_year = year
            self.own_home = True
            return True
        return False

    def decide_to_sell(self, current_market_price, years):
        if self.own_home and random.random() < 0.1:
            self.own_home = False
            return True
        return False

class Renter(Agent):
    def __init__(self, income, rent_function, max_rent_ratio):
        super().__init__('renter', income)
        self.rent_function = rent_function
        self.max_rent_ratio = max_rent_ratio
        self.years_since_last_found_home = 0

    def can_afford_rent(self, current_rent):
        return current_rent <= self.income * self.max_rent_ratio

def calculate_house_price(income, house_price_growth_rate, E=0.0001, D=0.005):
    pr = income * (1 + house_price_growth_rate)
    y = house_price_growth_rate
    return pr + np.random.normal(0, E) - D * np.log((y * 365 + 1) / 31)

def calculate_sell_price(income, house_price_growth_rate, E=0.1):
    N = np.random.normal(0, E)
    p = income * math.exp(N) / (1 - house_price_growth_rate)
    return p

def sellerProb(nh, income, a=2, b=10, c=0.107575):
    x = c * (1 + a * (0.2 - nh) + b * (0.7 - income))
    return x

def calculate_mortgage_payment(years):
    x = 12 * years
    w = 2.4856557259171987
    return ((
        4.34 + 654.26 * np.cos(1 * w * x) - 0.14 * np.sin(1 * w * x) - 653.10 * np.cos(1**2 * w * x) +
        0.10 * np.cos(2 * w * x) - 1.16 * np.sin(2 * w * x) - 1160.35 * np.cos(2**2 * w * x) +
```

```
        0.07 * np.cos(3 * w * x) - 0.49 * np.sin(3 * w * x) + 0.06 * np.cos(3**2 *
    w * x) +
        1160.31 * np.cos(4 * w * x) - 0.08 * np.sin(4 * w * x) - 0.04 * np.cos
    (4**2 * w * x) +
        0.02 * np.cos(5 * w * x) - 0.14 * np.sin(5 * w * x) + 0.05 * np.cos(5**2 *
    w * x) +
        0.12 * np.cos(6 * w * x) - 0.33 * np.sin(6 * w * x) + 0.07 * np.cos(6**2 *
    w * x)
    ) / 10)

def calculate_rent(income):
    return income * 0.3

num_years = 50
initial_market_price = 200000
initial_family_income = 50000
initial_investor_income = 100000
E = 0.1

initial_family_population = 545
initial_investor_population = 290
initial_renter_population = 165
initial_homeowner_population = 1000

available_housing_counts_per_iteration = []
houses_for_sale_per_iteration = []
houses_for_rent_per_iteration = []

min_homes_for_rent = 50

families = [Family(initial_family_income, 0.03, E) for _ in range(
    initial_family_population)]
investors = [Investor(initial_investor_income, initial_market_price * 1.5) for _
    in range(initial_investor_population)]
renters = [Renter(30000, calculate_rent, 0.3) for _ in range(
    initial_renter_population)]
homeowners = [HomeOwner(75000, 250000, calculate_mortgage_payment,
    calculate_house_price) for _ in range(initial_homeowner_population)]

income_values = [45000, 50000, 55000]
house_price_growth_rates = [0.02, 0.03, 0.04]

sensitivity_results = []

for income in income_values:
    for growth_rate in house_price_growth_rates:
        available_housing_counts = []
        houses_for_sale_counts = []
        houses_for_rent_counts = []

        for year in range(1, num_years + 1):
            new_renters = [Renter(random.randint(25000, 75000), calculate_rent,
    0.3) for _ in range(5)]
```

```
        renters.extend(new_renters)

        total_population = len(homeowners) + len(families) + len(investors) + len(renters)
        current_market_price = initial_market_price * (1 + year * 0.02)
        available_housing_count = 0
        houses_for_sale_count = 0
        houses_for_rent_count = max(min_homes_for_rent, len(renters))

        for homeowner in homeowners:
            mortgage_rate = calculate_mortgage_payment(year)
            if homeowner.decide_to_sell(current_market_price, 0.02, year, available_housing_count/total_population, mortgage_rate):
                available_housing_count += 1
                houses_for_sale_count += 1

        for family in families:
            if family.buy_home(current_market_price, 0.02, year):
                available_housing_count += 1

        houses_for_rent_count = 250
        for investor in investors:
            if investor.can_buy(current_market_price, year):
                available_housing_count += 1
                houses_for_sale_count += 1
                houses_for_rent_count -= 1
            elif investor.decide_to_sell(current_market_price, year) and houses_for_sale_count > 0:
                available_housing_count -= 1
                houses_for_sale_count -= 1
                houses_for_rent_count -= 1

        if year >= 32:
            houses_for_rent_count = 230

        if year % 2 == 0:
            for renter in renters:
                if renter.can_afford_rent(current_rent) and houses_for_rent_count > 0:
                    available_housing_count -= 1
                    houses_for_rent_count -= 1
                    renter.years_since_last_found_home = 1

        current_rent = calculate_rent(random.randint(25000, 75000)) * (1 + year * 0.02)

        for renter in renters:
            if renter.years_since_last_found_home > 0:
                renter.years_since_last_found_home += 1

        available_housing_count += houses_for_rent_count
        available_housing_count += houses_for_sale_count

```

```
            available_housing_counts_per_iteration.append(available_housing_count)
            houses_for_sale_per_iteration.append([houses_for_sale_count])
            houses_for_rent_per_iteration.append([houses_for_rent_count])

            avg_availability = sum(available_housing_counts_per_iteration) /
    num_years

            sensitivity_results.append({
                'income': income,
                'growth_rate': growth_rate,
                'avg_availability': avg_availability,
            })

available_housing_counts_per_iteration.append(available_housing_counts)
houses_for_sale_per_iteration.append(houses_for_sale_counts)
houses_for_rent_per_iteration.append(houses_for_rent_counts)

for result in sensitivity_results:
    print("Income:", result['income'])
    print("Growth Rate:", result['growth_rate'])
    print("Average Availability:", result['avg_availability'])
    print()

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit

def fourier_series(x, *a):
    n = (len(a) - 1) // 3
    omega = 2 * np.pi / len(x)
    result = a[0]
    for i in range(n):
        result += a[3*i+1] * np.cos((i+1) * omega * x) + a[3*i+2] * np.sin((i+1) *
     omega * x) + a[3*i+3] * np.cos((i+1)**2 * omega * x)
    return result

data = pd.read_csv('sad.csv')[['MORTGAGE30US', 'LISTNUMperCapita', 'MEDPRICE']]
y = data['MORTGAGE30US']
x = np.arange(1, len(y) + 1)
n_terms = 6
initial_guess = [np.mean(y)] + [0.5] * 3 * n_terms
bounds = ([-np.inf] * (3 * n_terms + 1), [np.inf] * (3 * n_terms + 1))
popt, pcov = curve_fit(fourier_series, x, y, p0=initial_guess, bounds=bounds,
    maxfev=10000)

future_steps = 600
x_future = np.arange(1, len(y) + future_steps + 1)
y_future = fourier_series(x_future, *popt)

print("First 5 values generated from the fitted function:")
print(y_future[:5])

```

```
plt.plot(x, y, label='Original Data')
plt.plot(x, fourier_series(x, *popt), 'r-', label='Fitted Curve')
plt.plot(x_future+len(y), y_future, 'g--', label='Predictions into the Future')

plt.xlabel('X Values')
plt.ylabel('MORTGAGE30US')
plt.title('Fitted Curve with Future Predictions (Fourier Series)')
plt.legend()
plt.show()

x_values_extended = np.arange(1, len(y) + future_steps + 1)
fitted_equation = "Fitted Equation: "
for i, coeff in enumerate(popt):
    if i == 0:
        fitted_equation += f"{coeff:.2f}"
    else:
        harmonic_index = (i - 1) // 3 + 1
        if i % 3 == 1:
            fitted_equation += f" + {coeff:.2f} * cos({harmonic_index} x )"
        elif i % 3 == 2:
            fitted_equation += f" + {coeff:.2f} * sin({harmonic_index} x )"
        else:
            fitted_equation += f" + {coeff:.2f} * cos({harmonic_index}  x  )"
print(fitted_equation)
n_harmonics = (len(popt) - 1) // 3
final_omega = (n_harmonics**2) * (2 * np.pi / len(x))
print("Final Angular Frequency (  ):", final_omega)

#part 2

def fourierExtrapolation(x, n_predict):
    n = x.size
    n_harm = 4
    t = np.arange(0, n)
    p = np.polyfit(t, x, 1)
    x_notrend = x - p[0] * t
    x_freqdom = fft.fft(x_notrend)
    f = fft.fftfreq(n)
    indexes = range(n)
    indexes = list(range(n))

    t = np.arange(0, n + n_predict)
    restored_sig = np.zeros(t.size)
    for i in indexes[:1 + n_harm * 2]:
        ampli = np.absolute(x_freqdom[i]) / n
        phase = np.angle(x_freqdom[i])
        restored_sig += ampli * np.cos(2 * np.pi * f[i] * t + phase)
    y_hat = restored_sig + p[0] * t
    y_mean = np.mean(x)
    ss_tot = np.sum((x - y_mean) ** 2)
    ss_res = np.sum((x - y_hat[:n]) ** 2)
    r_squared = 1 - (ss_res / ss_tot)

```

```
    return restored_sig + p[0] * t, r_squared

def main():
    x_albuquerque = np.array([0.2625288259, 0.212981892, 0.2265271616,
    0.2314365249, 0.2194451378, 0.2367446355, 0.239627183, 0.2724474144,
    0.2829884003, 0.2786590224, 0.2270016407])
    n_predict = 50
    extrapolation_albuquerque, r_squared_albuquerque = fourierExtrapolation(
    x_albuquerque, n_predict)
    plt.plot(np.arange(0, extrapolation_albuquerque.size),
    extrapolation_albuquerque, 'r', label='extrapolation')
    plt.plot(np.arange(0, x_albuquerque.size), x_albuquerque, 'b', label='data',
    linewidth=3)
    plt.title('Albuquerque Prediction')
    plt.xlabel('Years since 2012')
    plt.ylabel('Percent of people')
    plt.legend()
    plt.show()
    print("R-squared ALB:", r_squared_albuquerque)
    years_albuquerque = [20, 30, 60]
    y_values_albuquerque = [extrapolation_albuquerque[idx] for idx in
    years_albuquerque]
    print("Albuquerque - Y values at 21, 31, and 61 years:", y_values_albuquerque)

    x_seattle = np.array([1.51569115, 1.487464645, 1.451911844, 1.457704012,
    1.402994434, 1.550036217, 1.604248493, 1.691694091, 1.708748164, 1.546171847,
    1.585292971])
    n_predict = 52
    extrapolation_seattle, r_squared_seattle = fourierExtrapolation(x_seattle,
    n_predict)
    plt.title('Seattle Prediction')
    plt.plot(np.arange(0, extrapolation_seattle.size), extrapolation_seattle, 'r',
     label='extrapolation')
    plt.plot(np.arange(0, x_seattle.size), x_seattle, 'b', label='data', linewidth
    =3)
    plt.xlabel('Years since 2012')
    plt.ylabel('Percent of people')
    plt.legend()
    plt.show()
    print("R-squared Seattle:", r_squared_seattle)
    years_seattle = [20, 30, 60]
    y_values_seattle = [extrapolation_seattle[idx] for idx in years_seattle]
    print("Seattle - Y values at 21, 31, and 61 years:", y_values_seattle)

if __name__ == "__main__":
    main()
#part 3

plt.style.use('ggplot')
target_list = ['MORTGAGE30US','MEDPRICE','realPOP','LISTNUMperCaptica','LaborForce
    ','NewUnits','MedianSqFtList','MedianPriceperSqFt','AvgHoursWkly']
df = pd.read_csv('hda.csv')
df = df.replace(',', '', regex=True)
```

```
df[['LaborForce']] = np.array([430800,432500,438300,441700,441500,430300,429700]).
    reshape(7,1)
df = df.dropna()
y = df["Homeless Total"]
df = df.drop(columns = ['DATE', 'Homeless Total','Unemployment','2 units'])
X = df
X = X.astype('float')
y = y.astype('float')
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
    random_state=0)
print(X_train)
print(y_train)
clf = RandomForestRegressor(n_estimators=100, random_state=0)
clf.fit(X_train, y_train)
clf = RandomForestRegressor(n_estimators=100, random_state=0)
clf.fit(X_train, y_train)
y_pred = clf.predict(X_test)
print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test,
    y_pred)))
importance = clf.feature_importances_
feat_importances = pd.Series(clf.feature_importances_, index=X.columns)
feat_importances.nlargest(20).plot(kind='barh')
plt.xlabel("Random Forest Feature Importance")
plt.title("Random Forest Feature Importance")
for index, value in enumerate(feat_importances.nlargest(20)):
  plt.text(value, index, str(round(value, 2)))
plt.show()
df = df[['LaborForce']]
X = df
for feature in X.columns:
    x = X[feature].astype(float)
    slope, intercept = np.polyfit(x, y, 1)
    predicted_Y = slope * x + intercept
    plt.scatter(x, y, label='Original data')
    plt.plot(x, predicted_Y, color='red', label='Linear regression')
    plt.xlabel(feature)
    plt.ylabel('Homeless Total')
    plt.title(f"Linear Regression for {feature}")
    plt.axvline(x=435000, color='green', linestyle='--', label='Threshold line')
    plt.legend()
    plt.show()
    r_squared = r2_score(y, predicted_Y)
    print(f"Slope: {slope}")
    print(f"Intercept: {intercept}")
    print(f"R-squared: {r_squared}")
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
threshold = 435000
df['State'] = np.where(df['LaborForce'] < threshold, 'Less', 'Greater')
```

```
transition_matrix = np.zeros((2, 2))
for i in range(len(df) - 1):
    current_state = df['State'].iloc[i]
    next_state = df['State'].iloc[i + 1]
    current_state_index = 0 if current_state == 'Less' else 1
    next_state_index = 0 if next_state == 'Less' else 1
    transition_matrix[current_state_index, next_state_index] += 1
transition_matrix /= transition_matrix.sum(axis=1, keepdims=True)
sns.heatmap(transition_matrix, annot=True, fmt=".2f", cmap='Blues', xticklabels=['
    Less', 'Greater'], yticklabels=['Less', 'Greater'])
plt.title('Transition Matrix: Less to Greater')
plt.show()
print(transition_matrix)
std_dev = 0.05
mean = 0
num_states = 50
initial_state = np.array([[1,0]])
current_state = initial_state
states = [initial_state.flatten()]
for i in range(num_states):
    noise = np.random.normal(mean, std_dev, size=transition_matrix.shape)
    noised_transition_matrix = transition_matrix + noise
    row_sums = noised_transition_matrix.sum(axis=1)
    normalized_transition_matrix = noised_transition_matrix / row_sums[:, np.
    newaxis]
    next_state = np.dot(current_state, normalized_transition_matrix)
    states.append(next_state.flatten())
    current_state = next_state
states = np.array(states)
plt.figure(figsize=(14, 6))
plt.subplot(1, 2, 1)
for i in range(len(initial_state[0])):
    plt.plot(states[:, i], label=f'State {i+1}')
plt.title('State Evolution Over Time')
plt.xlabel('Time Step')
plt.ylabel('Population')
plt.legend()
plt.grid(True)
plt.subplot(1, 2, 2)
for i in range(len(initial_state[0])):
    x = np.arange(len(states))
    y = states[:, i]
    model = LinearRegression().fit(x.reshape(-1, 1), y)
    trend_line = model.predict(x.reshape(-1, 1))
    plt.plot(trend_line, label=f'State {i+1} Trend')
plt.title('Trend Line for State Evolution')
plt.xlabel('Time Step')
plt.ylabel('Population')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```