UNIVERSITY OF NOTRE DAME

CSE40467 / CSE60467: Data Science

Milestone 2

Prof. Diego Gomez-Zara

TAs: Mariana Fernandez-Espinosa, Khiem Le

Team Number	7
Members	Claudia Kuczun, Eva Gorzkiewicz, Samantha Nagel, Tom O'Connor
Date	May 2, 2024

Project Repository

1. Introduction

1.1. Executive Summary. *A good executive summary is a concise and clear overview of a larger document that highlights its key points, findings, and recommendations, aimed at giving readers a quick and comprehensive understanding of its content.*

Our goal with this project was initially to find a way to predict a set of cities that would match a set of specified attributes. However, after guidance from the Professor and our TA Mariana, we decreased the scope of our work to focus on unsupervised clustering to group cities that are similar, focusing on a specific set of United States city attributes.

We derived our dataset from an existing city clustering project on GitHub. We initially attempted to locate and combine datasets by hand, but we realized this would be a lengthy and complex process so we focused on finding pre-existing datasets. The dataset we found was perfect for this project since it contained about 59 different attributes for 700 cities, meaning it contained over 41,000 individual data objects. The features were a combination of categorical and numerical, but for this project, since we were focusing on clustering methods and required numerical features, we narrowed down the dataset to only specific attributes. After removing categorical attributes (apart from city and state, since this is necessary for identifying data elements after clustering), we manually reviewed and selected the attributes that we thought would be most relevant to people wanting to relocate or travel. The attributes include the city's mean temperature, percentage of the population that is Democratic and Republican, average commute type, work type (whether people work at home/remotely or in-person), average household income, crime rate, and others.

After narrowing down the features, we cleaned up the data; this included replacing missing values with zeros or the column average values and standardizing. We also performed PCA to reduce the dimensionality and complexity of our data, make it easier to cluster and visualize and remove noise and irrelevant features since some of our clustering methods are sensitive to noise and outliers.

Post-processing, we designed and evaluated the performance of four different clustering methods: DBSCAN, K-Means, t-SNE, and Spectral. DBSCAN performed poorly due to the high dimensionality and the variation in density in the dataset. Spectral clustering also performed fairly poorly, likely due to the fact that the clusters/data had complex shapes and this prevented it from clustering effectively. K-Means performed slightly better, but it was difficult to find the optimal number of clusters due to the elbow graph not having a clear point for an optimal number, and since the clusters in our data are not perfectly spherical it may have negatively impacted its performance. Finally, t-SNE had the best overall performance, probably because it is well-equipped to handle high-dimensional data, and its output is easy to analyze visually. As a result, we selected t-SNE to use in our final product.

Our final product is a tool that enables a user to input a target city, and our algorithm outputs similar cities based on t-SNE clustering. By visualizing the detailed attributes of each city for the user, they can easily do a side-by-side comparison and decide whether the city meets their needs!

1.2. ***Problem definition****. Explain what this problem is about. Discuss why it should be studied/analyzed.*

Our problem is relevant for all individuals who are looking to relocate or travel within the United States. If an individual wants to move for work, family, or quality of life improvements, they would be able to use our clustering analysis to easily find cities similar to a target city (that they know they like/are currently living in) for more options. For travel destinations, if an individual has traveled, for example, to New York City, and they want to travel to a city that is similar but not the same city, they can use our analysis to deduce that a city such as San Francisco would be good for them to visit.

After performing our analysis, we output the features of the target city and its match(es) side-by-side for easy comparison by the user. This enables them to learn the similarities of features that may be more important to them in finding the best candidate city for their relocation or travel.

In future work, we would enhance our tool by enabling users to select specific attributes that they require in the cities they are looking for. This would require us to change our current model, and ensure that it can target specific attributes in cities as opposed to just overall 'similarity.' We would also create a more official website (user interface) that would make the search and comparison processes easier to increase ease of use.

2. Related Work

2.1. Describe related work with this data source.

There are a couple of related works available online that are related to our project. First, there is a "Where Should I Live?" quiz[1] available online for free. The user fills out a questionnaire, where they are asked to choose whether they prefer warm weather over a short commute or good restaurants over a larger population. After filling out a long set of questions, the tool outputs a ranked list (with percent matches) with cities that suit the criteria initially selected.

Secondly, there is a GitHub repository[2] called "city-picker" that attempted to solve a similar problem: they focused on selecting target attributes that are important to the user and choosing cities based on these specified parameters (the associated website for this project seems to be down, so we could not actually see/test their tool). This is where we sourced our data from, as this project had the data that we were looking for already aggregated and available! We note that they used K-Means in the solution to their problem, so our solutions will likely vary a bit.

Additionally, there is an academic paper[3] that analyzes how popular regions within a city are for commuters (but does not look at multiple cities). This is kind of similar to analyzing cities for travel preferred by people or what attributes are preferred by users in cities with shorter or longer average commute times. Clearly, this is not a direct comparison, but it is similar in that it looks at region popularity.

We did not find any academic studies that are more directly related to our problem, but the repository and website that we did find are close matches that show that this is a common problem that multiple groups of people have attempted to solve.

3. Data description

3.1. ***Data Collection****. Explain how data collection was conducted. If your dataset comes from a website, describe the source and explain how the original researchers potentially obtained the data.*

Our problem required data from multiple sources spanning various fields, so we could not simply use a pre-existing dataset on Kaggle or collect data manually. Manual data collection (researching the city ourselves or writing web scrapers to access specific data from different sites) would have been time-consuming given the timeline of this project. So, we first attempted to find datasets containing information that we wanted to use for this project online. For example, we searched for city datasets of weather patterns, demographics, economics, crime and safety, and politics. It was difficult to find all of these datasets, and it would have been difficult to aggregate as the datasets often had very different scopes (very few of the United States cities overlapped, or the dataset pertained to cities in Europe, etc.). We shifted our search to look for pre-existing datasets and found the dataset used in the city-picker repository[2] to be perfect for our needs.

In the repository description, the creators state that they aggregated data from the following eight datasets: DP03 - Economic/Demographic Population Estimates, City/County Population Estimates, U.S Census Demographic Data, U.S Violent Crime Dataset, Voter Share by County, IRS Tax Data, Weather Data, and COVID-19 Data. After synthesis, they selected 59 relevant attributes that they wanted to focus on when clustering and selecting target cities. Their resulting dataset is exactly what we were trying to synthesize initially and eliminated the work required to create a dataset since it contains the information we want for this project.

3.2. ***Data Preprocessing.*** *Explain all the steps that you performed to transform the raw data into a manageable dataset (e.g., merge, impute missing values, standardize, normalize). If your data did not require any data pre-processing, please discuss any potential quality issues of the data (e.g., how representative is the dataset?, who pre-processed the dataset?)*

In the development of our city recommender system, the dataset underwent a series of preprocessing steps to ensure it was analytically viable. The initial dataset was diverse with a variety of features reflecting city characteristics such as demographics, climate, and socio-economic data. The first step in preprocessing was data cleaning, where we filtered out non-numeric data from approximately 70 initial features. This was a crucial step, as non-numeric data would not have been compatible with the K-means and other clustering methods we intended to use.

Following data cleaning, we engaged in a feature selection process. This step involved manually reviewing the remaining features as a group to remove those that were not relevant to our project's goals. For instance, data points like `county_fips`, `foreignborn_pct`, and `lesshs_whites_pct` were excluded from the analysis. Depending on the clustering technique applied, such as in spectral clustering, we adapted our approach to include transformation of some non-numeric data into vectors, which could be processed by the algorithm to test its effectiveness.

To address any missing values in the dataset, we implemented imputation, replacing missing data with the mean value of respective columns. This ensured that no data point was disregarded due to missing information, maintaining the integrity and continuity of our dataset. In addition to handling missing data, we standardized the dataset to normalize the data across different scales, thereby minimizing any bias introduced by various units of measurement.

A significant concern initially was the potential need to merge various data sources to create a comprehensive dataset covering aspects like weather, demographics, and crime rates. However, we

discovered a dataset on GitHub that had already integrated these elements, greatly simplifying our data integration process.

These preprocessing steps were essential in refining the dataset, making it manageable and well-suited for the sophisticated analyses that our project required.

3.3. ***Data Documentation.*** *Describe all the features/variables of your dataset. Please describe their type of data, ranges, min, max, or mode.*

Our dataset consists of a range of features that describe various aspects of city life, aimed at providing a comprehensive overview to recommend cities based on user preferences. Below are the key features used in our analysis:

- Demographic Features: These include `city_population`, `density`, and age distribution (`age29andunder_pct`, `age65andolder_pct`). These features are numeric, providing a quantifiable measure of each city's demographic makeup.
- Ethnicity and Employment: Percentages of ethnic groups (`pct_hispanic`, `pct_white`, `pct_black`, `pct_native`, `pct_asian`, `pct_pacific`) and employment types (`pct_professional`, `pct_service`, `pct_office`, `pct_construction`, `pct_production`) are critical in reflecting the cultural and economic diversity.
- Transportation and Work: Data on commuting patterns (`pct_drive`, `pct_carpool`, `pct_transit`, `pct_walk`, `pct_other_transportation`, `pct_work_at_home`) and work types (`pct_private_work`, `pct_public_work`, `pct_self_employed`, `pct_family_work`) provide insights into the mobility and occupational landscape of the population.
- Economic Indicators: Features such as `avg_income`, `income_per_cap`, along with `income_error` and `income_per_cap_err`, offer a detailed view of the economic status of city residents.
- Climate and Crime: Long-term mean temperature (`LTM_mean_temp`), precipitation (`LTM_mean_percipitation`), and crime rates (`crime_per_100`) are included to give a sense of the living conditions and safety of the area.

These variables have been selected for their relevance to the task at hand, each contributing to a robust model for city recommendation.

3.4. ***Variables.*** *Describe which columns are features (independent variables) and which ones are your target/class/dependent variables.*

In the context of our city recommender system, the selected features serve as independent variables, each providing essential data that influences the recommendation outcome. These features include demographic statistics, economic indicators, climate data, and more, as listed in the Data Documentation section.

The dependent variable, or the target variable in our analysis, is the suitability of a city as a recommendation. It is indirectly defined by the clustering outcome, where each cluster represents a group of cities with similar characteristics. The recommendation system leverages these clusters to suggest cities that match a user's preferences based on their similarity to the features of cities within a cluster.

4. Modeling

4.1. ***Description of Algorithm 1 and its model.*** *Please include details about the parameters. Did you fine-tune the parameters?*

The first algorithm we tried was DBSCAN, which clusters data based on density. Initially, we performed DBSCAN considering all of the relevant numerical attributes, with a variety of epsilon (0,10.5,1,1.5) and min_neighbors (5,10,15,20) values based on the standard ranges that other DBSCAN models typically use. Note that we first calculated the gap statistic to find the optimal number of clusters (n=9 in this case). However, after visualizing the clusters and output, we realized that DBSCAN is likely to always perform poorly with our dataset due to the changing density in our data distribution (difficult to find suitable values for epsilon) and the high dimensionality (DBSCAN has difficulty with high-dimensional data) as can be seen in Figure 1. So, we transitioned to attempting to fine-tune the eps/min_neighbor parameters by running DBSCAN with each possible feature pair, as opposed to the entire set of features, to reduce dimensionality and possibly improve performance. Overall, this still performed quite poorly as many of the features have varying densities. Interestingly, we did get a best overall Silhouette Score of approximately 95% with 10 clusters as can be seen in Figure 2, but getting good performance in a couple of pairings of this dataset is not good enough for the performance we need to successfully match a target city to similar neighbors. We did try evaluating DBSCAN using other parameters, like BetaCV, SSE, the Calinski–Harabasz index, and the Davies-Bouldin score, but these performed even more poorly than the Silhouette Score, as you can see in our code. Ultimately, we moved forward with other algorithms that would be more adept at clustering while considering multiple features.

Figure 1. "Best" Silhouette Score considering all numerical features

```
Best silhouette score: 0.03555240990895669
Best eps: 1
Best min_samples: 20
```

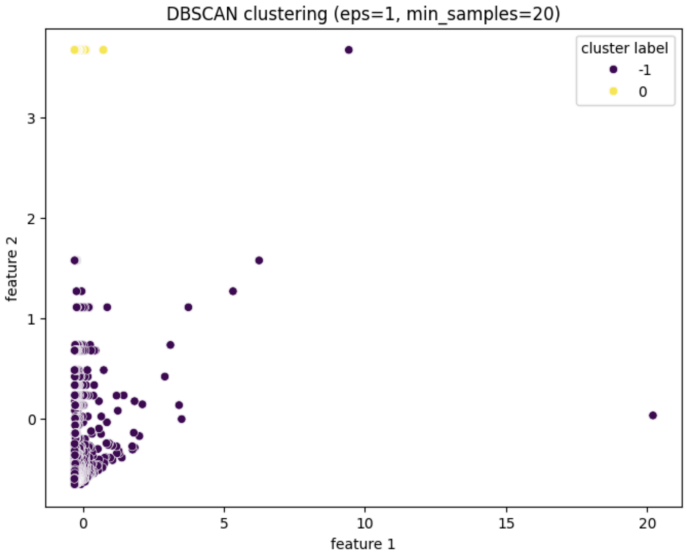
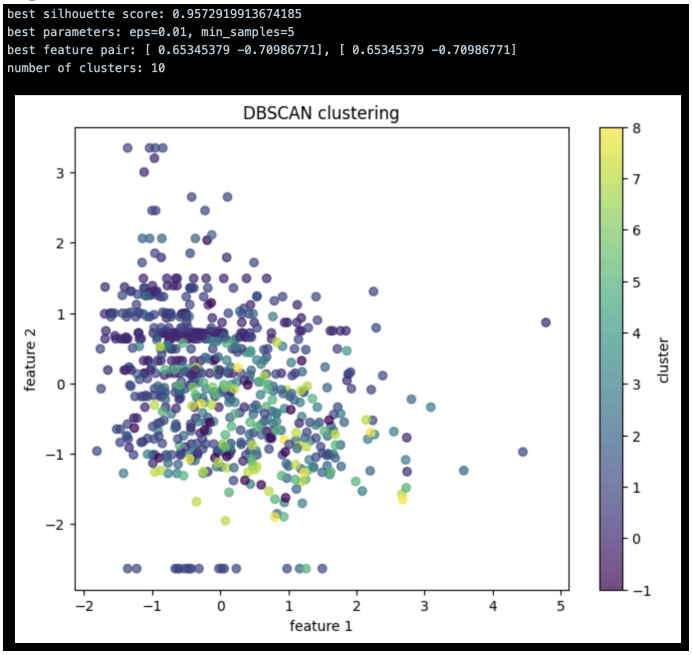


Figure 2. Best overall Silhouette score for individual feature pairs



4.2. ***Description of Algorithm 2 and its model.*** *Please include details about the parameters. Did you fine-tune the parameters?*

In our analysis using Algorithm 2, the K-Means clustering technique was selected to identify inherent groupings within the dataset. This method strategically partitions the data into clusters by minimizing the variance within each cluster, quantified as inertia. Key parameters of the K-Means algorithm include n_clusters, init, n_init, max_iter, and random_state. These parameters were carefully fine-tuned to optimize the clustering results, primarily using the Elbow method and cluster validity metrics.

The fine-tuning process began with the Elbow method, visualized in Figure 1, where inertia was plotted against a range of cluster numbers. Although this method suggested potential cluster counts, the choice was not definitive due to the subtle nature of the elbow points. To enhance our decision-making process, we employed several cluster validity metrics: the Silhouette Score, visualized in Figure 2; the Calinski-Harabasz Score, shown in Figure 3; and the Davies-Bouldin Score, detailed in Figure 4. These metrics, while useful for assessing the compactness and separation of clusters, provided mixed signals regarding the optimal number of clusters.

Given the ambiguity in these quantitative assessments, a visual assessment using Principal Component Analysis (PCA) became crucial. Figure 5 illustrates the PCA plots which helped visualize the data clustering more clearly. Upon examining these plots, it was observed that a three-cluster solution minimized overlap between clusters—a common issue with higher cluster counts. However, even with three clusters, K-Means tended to produce one significantly larger cluster that resembled a "blob," indicating a less distinct separation among some data points.

This visual observation led us to reconsider the adequacy of the K-Means algorithm for our specific dataset. Although K-Means is a powerful and widely-used clustering method, it assumes clusters of roughly similar density and size, which did not hold true in our case. The tendency to form one oversized cluster suggested that K-Means was not capturing the underlying structure of our data effectively.

In light of these findings, the decision was made to explore alternative clustering techniques that might better accommodate the varied densities and distributions observed in our dataset. This strategic shift underscores the importance of not only relying on quantitative metrics but also integrating visual evaluations to ascertain the most suitable method for data analysis.

Figure 1. Elbow Method Plot

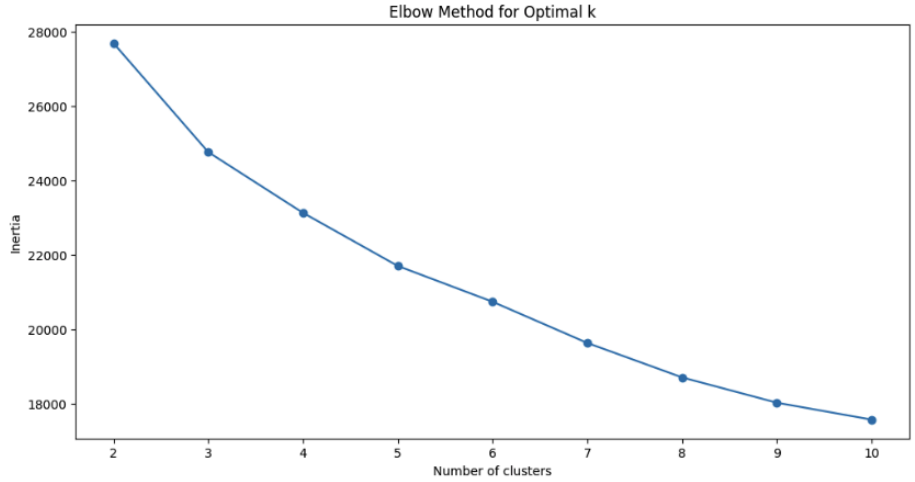


Figure 2. Silhouette Coefficient Plot

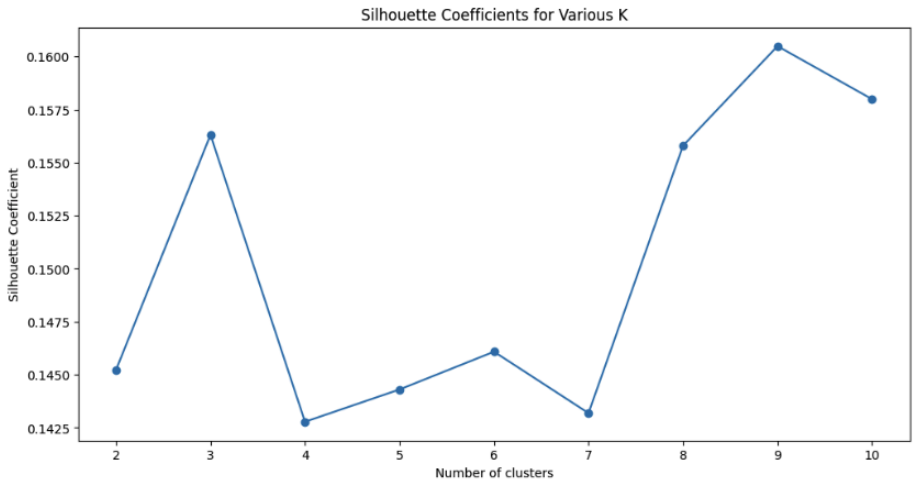


Figure 3. Calinski-Harabasz Scores Plot

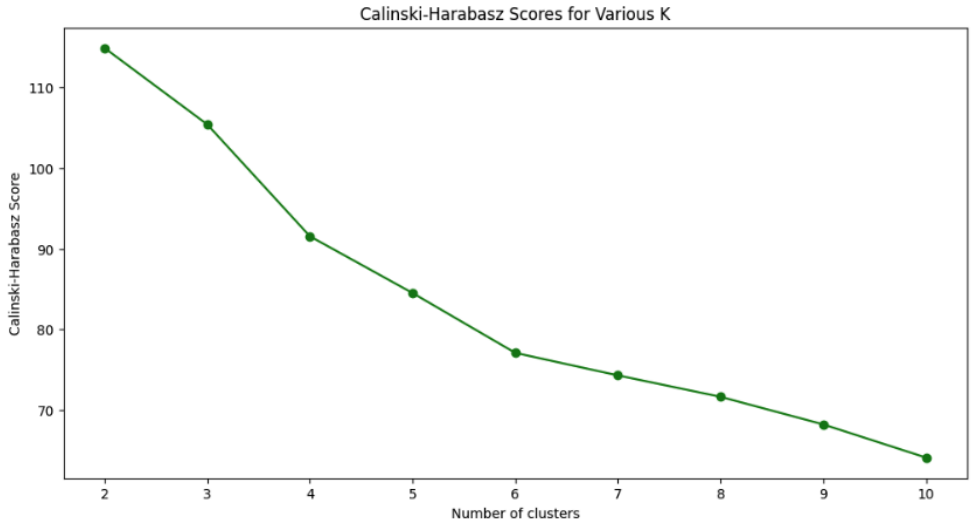


Figure 4. Davies-Bouldin Scores Plot

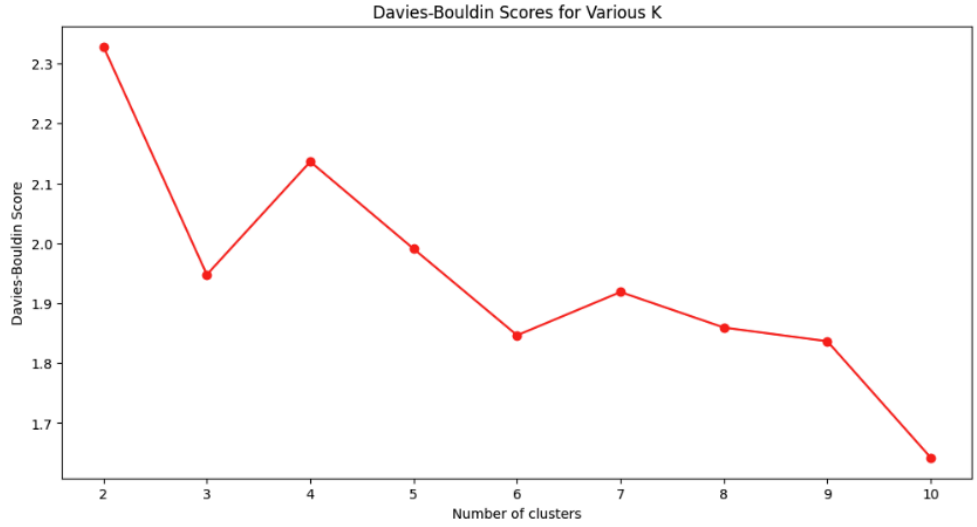
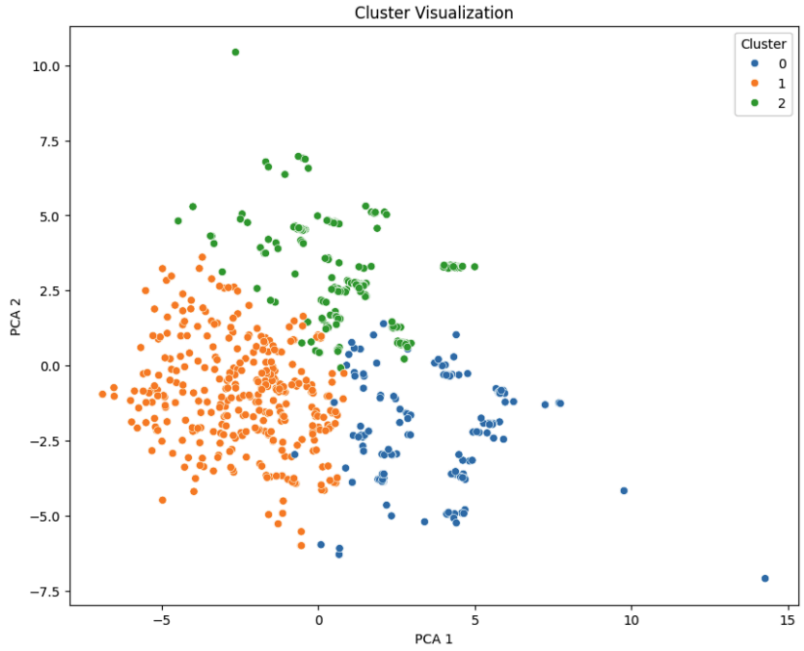


Figure 5. PCA Cluster Visualization



4.3. ***Description of Algorithm 3 and its model.*** *Please include details about the parameters. Did you fine-tune the parameters?*

Our third algorithm used for the data was spectral clustering. This algorithm was chosen due to its ability to highlight complex nonlinear shapes and structures when clustering. In order to do this, spectral clustering uses eigenvalues and eigenvectors of a similarity matrix of the data to separate the data into clusters. Spectral clustering reduces the dimensionality of the data and then uses another clustering method on the reduced dimensionality data in order to get the final result.

Spectral clustering has two main parameters: the number of clusters and the affinity. The affinity is the traditional clustering algorithm to be used once the dimensionality of the data is reduced. In order to fine tune the number of clusters, first the pairwise Euclidean distances between the points in the points in the data were calculated. Then, these distances were converted into an affinity matrix to get the similarity between points using the Gaussian kernel function. Afterwards, the Laplacian matrix was made using the difference between the degree matrix and affinity matrix. After making the Laplacian matrix, the eigenvalues of the Laplacian matrix were calculated and plotted as seen in Figure 1. Then, the eigengaps were calculated by computing the difference between consecutive eigenvalues. Finally, the optimal number of clusters was found by using the index of the maximum eigengap plus one. This resulted in three clusters being the optimal amount of clusters to specify as a parameter.

In order to fine tune the affinity, nearest_neighbors, laplacian, and rbf were tested and evaluated based on silhouette score, Calinski-Harabasz index, and Davies-Bouldin index. Overall we found that laplacian worked the best out of the three tested affinities. The plot of the clusters after using PCA found using laplacian as the affinity and three clusters are seen in figure 2.

Figure 1. Eigenvalues of Laplacian Matrix

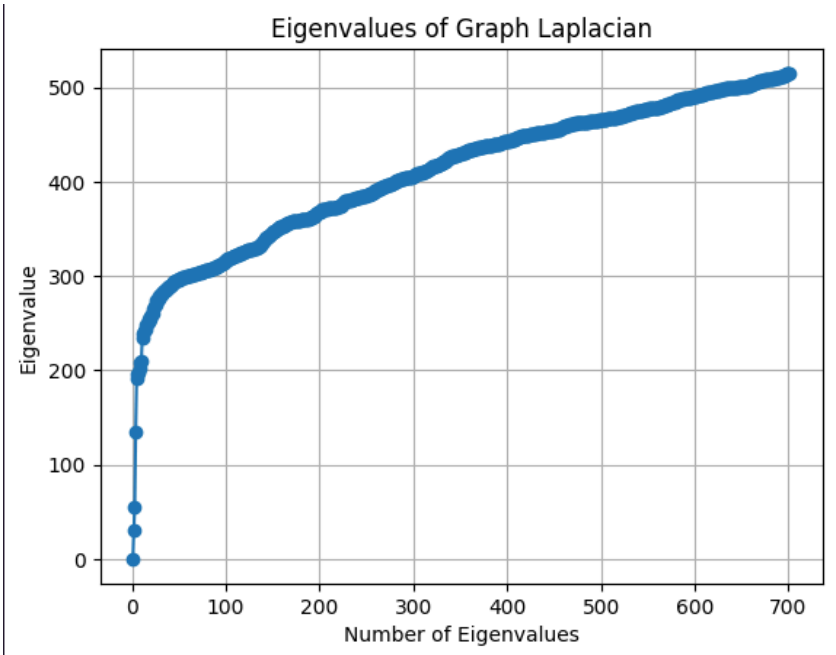
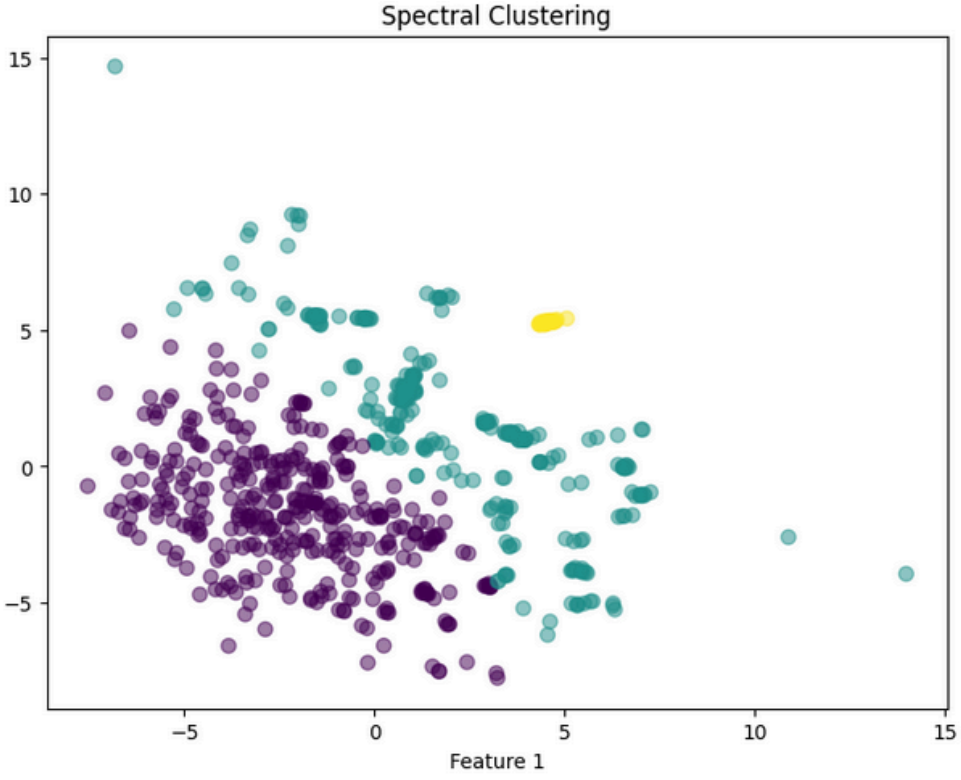


Figure 2. Spectral Clustering Visualization



4.4. ***Description of Algorithm 4 and its model.*** *Please include details about the parameters. Did you fine-tune the parameters?*

Our final model used was t-distributed Stochastic Neighbor Embedding (t-SNE). t-SNE is a nonlinear dimensionality reduction algorithm. Typically, t-SNE is "used to understand high-dimensional data and project it into low-dimensional space."[4] Essentially, it models the similarity between pairs of high-dimensional points then projects it into lower-dimensional space.

For this model, we found that the MinMaxScaler worked best for scaling the data. We then used Principal Component Analysis (PCA) to reduce the dimensionality and preprocess the data. Figure 1 is what the clustering looked like after PCA.

After PCA, we then implemented the t-SNE algorithm. We tested a few different values for the parameters. These included n_components, perplexity, verbose, and n_iter. n_components is the dimension of the embedded space. We found the most optimal number of components was 2. Perplexity is the number of nearest neighbors that is used in other manifold learning algorithms. Our optimal perplexity was 20. Verbose is the verbosity, which had the best value of 1 for clustering. Lastly, n_iter is the number of iterations which we set to 2000. This number of iterations was a good balance of keeping the algorithm efficient yet accurate.
Figure 2 is what the clustering looked like after t-SNE.

Figure 3 shows a direct comparison between the clustering produced by PCA and the clustering produced by passing the PCA reduced data into the t-SNE algorithm. PCA is on the left, t-SNE and PCA combination is on the right.

Just by looking at the plot, you can see that the t-SNE clusters the cities fairly well. There are distinct, spread out clusters. After comparing our other algorithms, we determined that t-SNE performs the best. We will discuss how we evaluated our t-SNE model below.

Figure 1. PCA Clustering

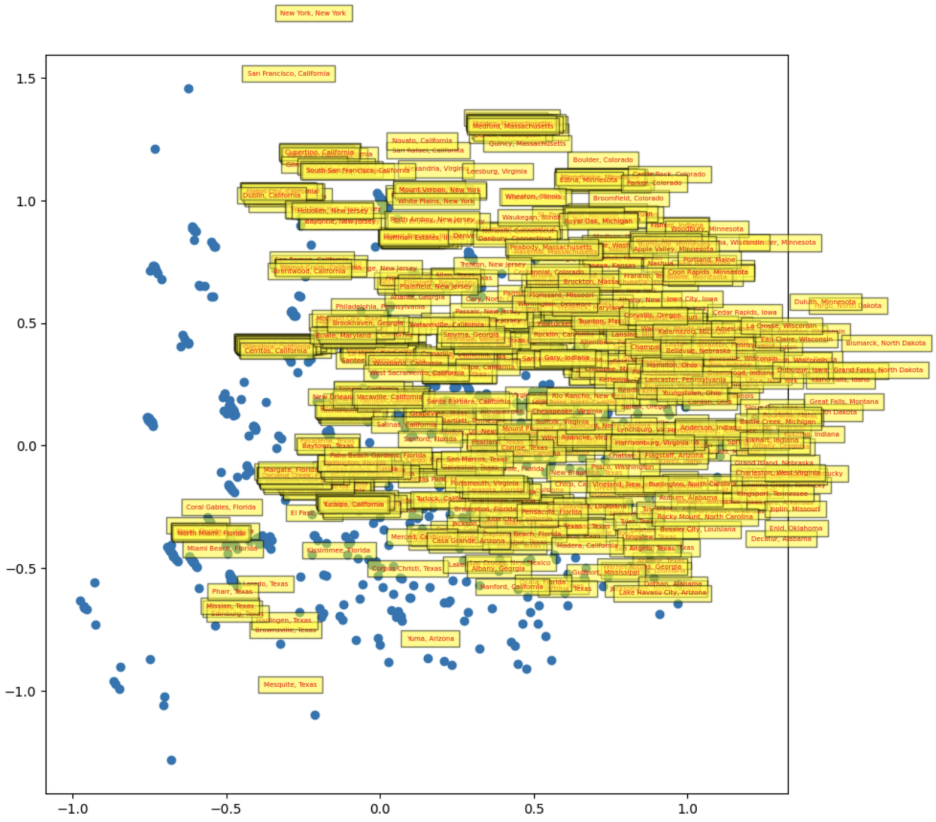


Figure 2. t-SNE Clustering

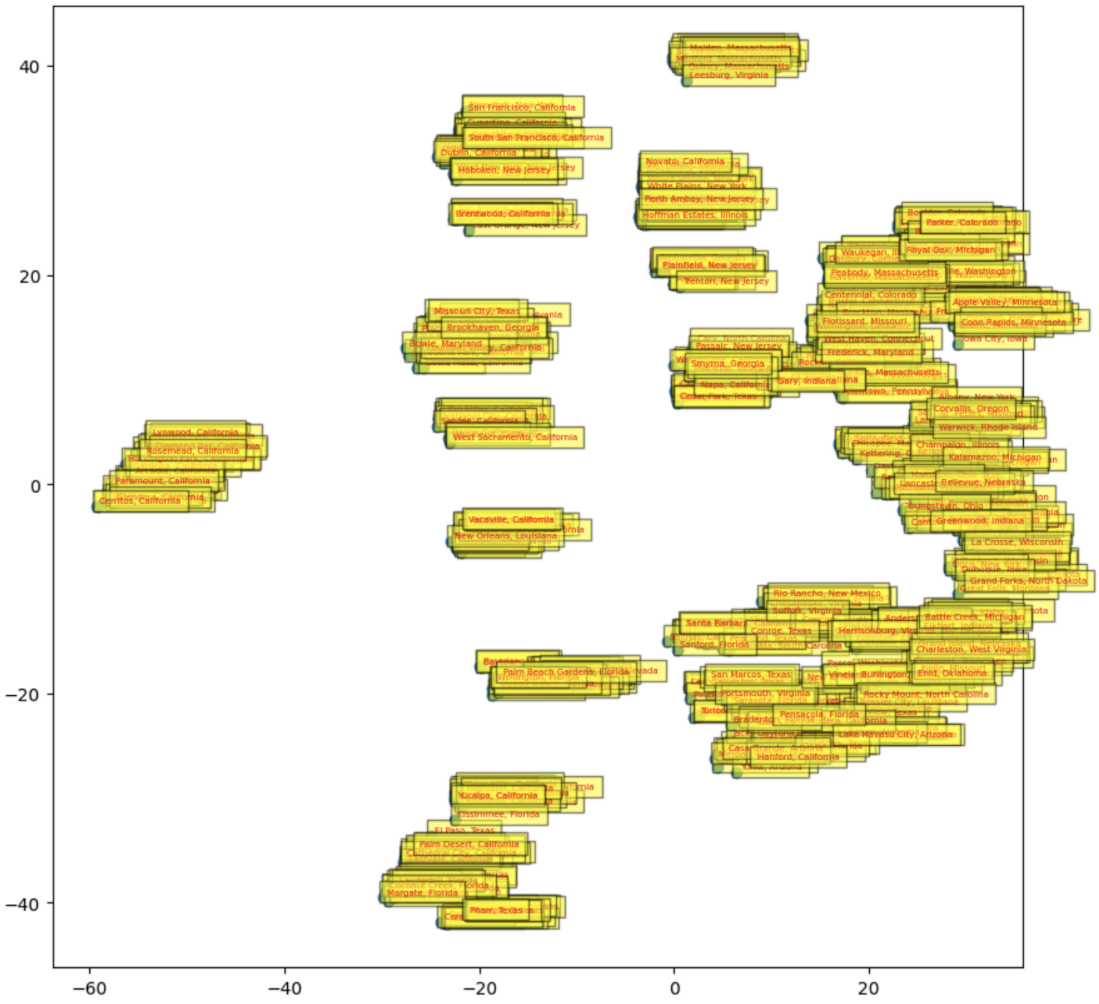
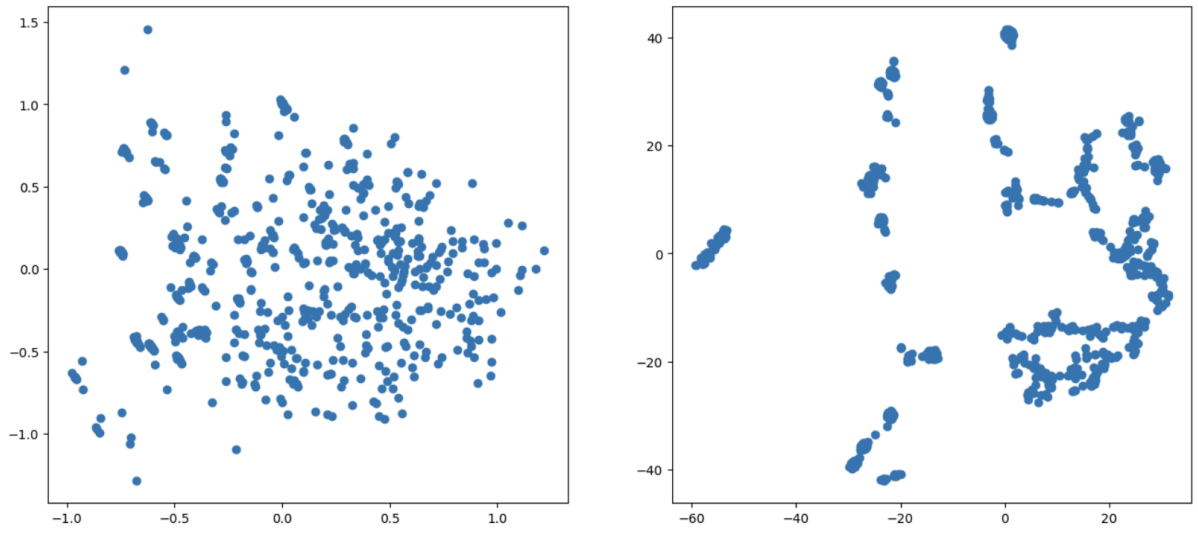


Figure 3. Comparison Between PCA Clustering and t-SNE Clustering



5. Evaluation

5.1. ***Baseline****. Please describe your baseline for the evaluation.*

For the evaluation of our city recommender system, we established a baseline method to compare the effectiveness of our model. The baseline chosen was a simple random selection approach. In this method, the recommendation of a city similar to a given city (e.g., New York) is made by randomly selecting from the list of cities in the dataset. For instance, if New York is the input, a random city such as South Bend might be suggested as similar, regardless of actual similarities in demographics, climate, or other relevant factors.

This baseline serves to highlight the improvements and precision offered by our clustering-based recommendation system. By comparing the outcomes of our model to those generated by this random baseline, we can demonstrate the value added by employing sophisticated clustering techniques like t-SNE, which groups cities based on their actual similarities across multiple dimensions.

5.2. ***Metrics****. Describe the metrics and their results.*

The metrics chosen for evaluating the performance of our city recommender system are designed to quantify the similarity between the recommended cities and the input city. To understand these metrics, we observed how well our model's recommendations compared to the user's input in terms of key attributes like population, density, and average income.

By comparing cities such as Boston, Massachusetts, to others in the dataset, we can evaluate the recommender system's precision. For example, the results illustrated in the bar chart and radar chart show that while some cities like Yonkers, New York, may have a population size and density close to Boston's, their average income may differ significantly, which impacts the overall similarity score. On the other hand, the PCA scatter plot and the t-SNE visualization provide a multidimensional perspective on city similarity, where the proximity of points reflects the likeness across multiple features.

Through these metrics, we assessed not only individual attribute matches but also the holistic similarity across all features. This approach allows for a nuanced understanding of what makes a city 'similar' beyond surface-level attributes, incorporating a broader spectrum of socioeconomic, demographic, and geographic factors.

6. Discussion

6.1. ***Discussion of the Results****. Please provide a thorough and insightful analysis of the results, integrating them with previous/related work.*

The recommender system's robust analytical framework was leveraged to evaluate urban similarities with a case study focusing on Boston, Massachusetts. To initiate the recommendation process, a set of pre-selected features was utilized, chosen for their relevance to urban living such as average income, city population, and density. The exclusion criteria mandated that recommendations omit cities within the same state as Boston. Through this process, the system identified a suite of cities that resemble Boston to various extents. Notably, Yonkers, New York emerged as the closest match, as illustrated in Figure 1. The bar chart in this figure underscores Yonkers' similarity to Boston in the pre-selected key attributes, affirming the effectiveness of our feature selection methodology.

The subjective analysis further corroborates Yonkers' analogous position to Boston, encompassing shared historical significance, urban structure, and demographic diversity—characteristics that, while salient to residents' experiences, often elude quantification. The PCA scatter plot in Figure 2 and the t-SNE visualization in Figure 3 elevate the analysis from mere numerical matching to a more sophisticated assessment of similarity, incorporating the complex tapestry of city life into the model's recommendations.

The introduction of Figures 2 and 3, which showcase the dimensionality reduction capabilities of PCA and t-SNE, illustrates how Yonkers and Boston's data points are drawn closely together, echoing the cities' resemblance. These advanced visualizations display the cities as clusters of similar features in a reduced dimensional space, offering an intuitive understanding of their likenesses and differences.

Drawing from the foundational concepts seen in existing city comparison tools, such as online quizzes and GitHub repositories, our system advances the field with refined algorithms and user-centric visualizations. The radar chart in Figure 4 exemplifies this by offering a multi-attribute visual comparison, which speaks volumes about our system's capacity for depth and customization in analysis.

The comparative evaluation of Yonkers and Boston highlights a critical aspect of our model: while it adeptly identifies cities with similar features, the assignment of weight to each feature during the similarity calculation process suggests room for refinement. Providing users with the ability to prioritize features when seeking recommendations will empower them to personalize the results to their individual criteria, a feature that our pre-selection method has started to address.

As we move towards a more interactive and user-responsive future, integrating direct feedback will be essential to fine-tune the recommendations further. This will align our model with actual user experiences, enhancing both the accuracy and reliability of the system. Ultimately, Figures 1 through 4 set the stage for an evolving recommendation system that not only matches cities statistically but also resonates with the users' preferences and lived experiences.

Figure 1. Bar Chart of Average Feature Comparisons – Highlights the comparison of pre-selected features like city population, density, and average income between Boston and its five closest matches, showcasing Yonkers as the most similar.

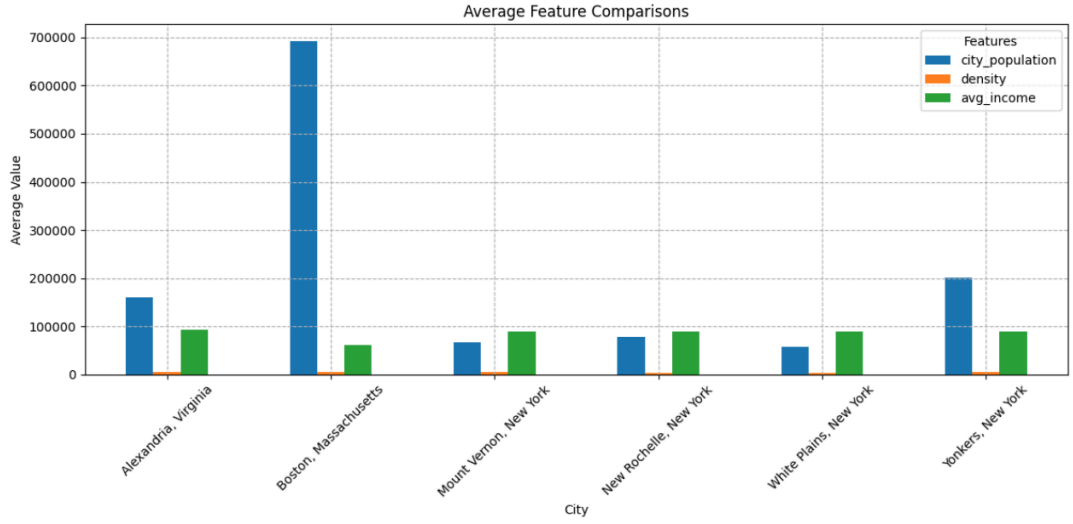


Figure 2. PCA Scatter Plot – Maps the cities based on the pre-selected features, clustering those with similar characteristics closer together.

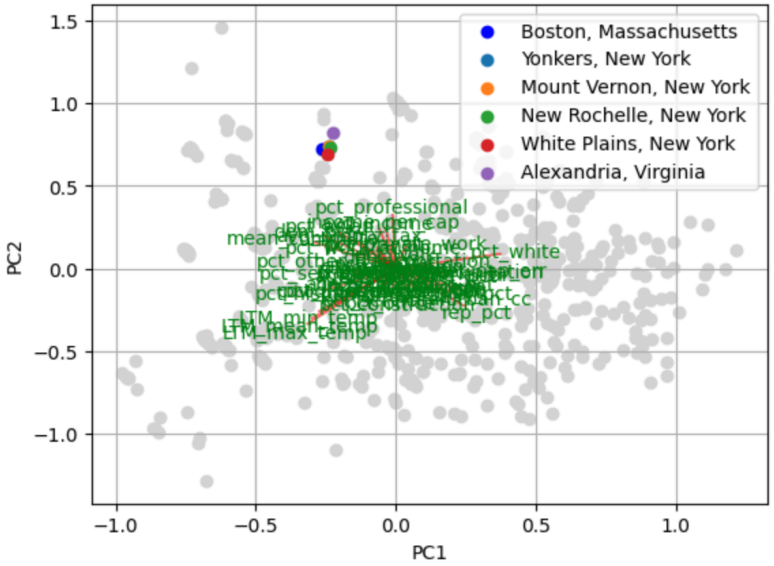


Figure 3. t-SNE Visualization – Further delves into the data's dimensionality to draw relationships between cities, with proximity reflecting a higher degree of similarity.

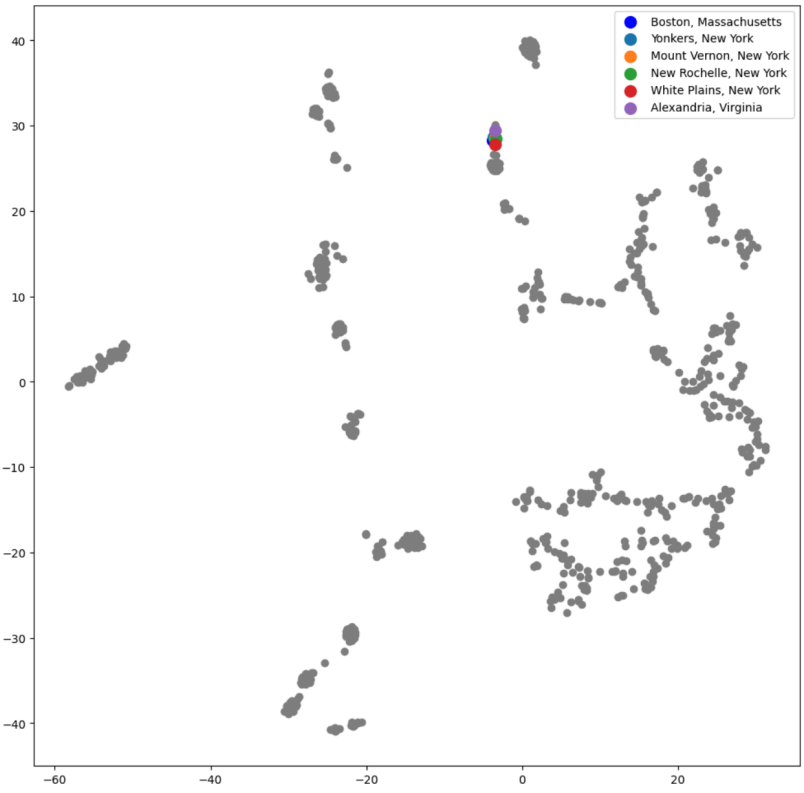
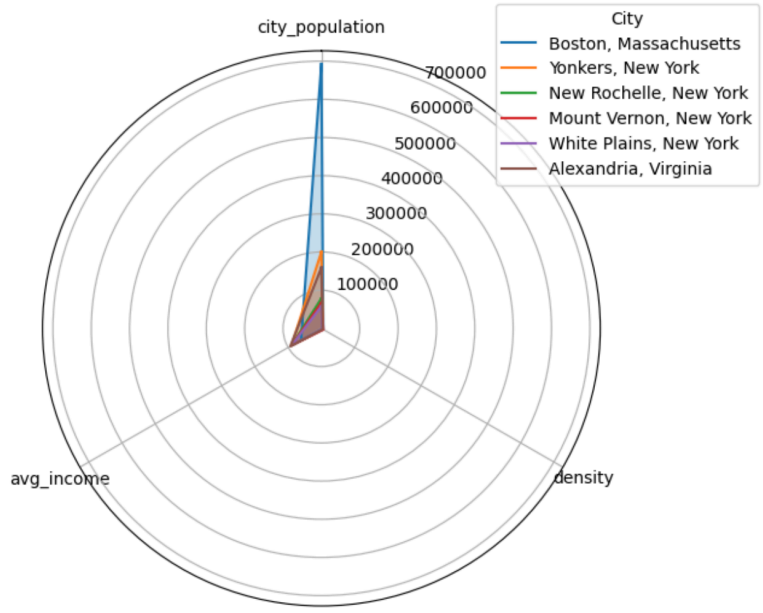


Figure 4. Radar Chart – Compares Boston with the closest cities across multiple pre-selected attributes, providing a comprehensive view of their similarities.



6.2. ***Recommendations and implications of your report****. Provide relevant recommendations, develop thoughtful implications, and suggest specific actionable directions for the future.*

The potential of our city recommender system, as illustrated in the analysis, opens avenues for enhancement and expansion. To make the system more intuitive and user-friendly, we envision developing an interactive interface. This interface would not only simplify the input of user preferences but also allow users to adjust the significance of city attributes, tailoring the recommendations to their individual needs. Moreover, expanding the database to include a wider array of cities and integrating additional variables such as cultural landmarks, economic indicators, and quality of life factors will increase the model's precision and utility.

Forging partnerships with travel and real estate platforms presents an opportunity to translate the analytical capabilities of our system into practical tools for users seeking vacation destinations or considering new homes. Such integrations promise to leverage our system's insights for real-world applications, enhancing user experience in tangible ways.

Looking ahead, we anticipate integrating predictive modeling into the system. Such features would use advanced machine learning algorithms to forecast demographic and economic trends within cities, ensuring that our recommendations remain relevant amid changing urban landscapes. Furthermore, the establishment of feedback loops would create a mechanism for continuous learning and improvement, allowing the system to adapt to evolving user preferences and behaviors.

The implications of these enhancements are multifaceted. For individuals, they mean a more sophisticated and customizable tool that streamlines complex decisions related to travel and relocation. For businesses, it translates to strategic insights into new markets, akin to those found in their current successful locations, facilitating more confident expansion plans. These strategic enhancements aim to bolster the system's current capabilities and secure its relevance for future applications, ensuring it remains a valuable resource as cities and their populations continue to evolve.

7. Members' Contributions

You must describe what each member contributed to the report. We will adopt the CRediT Taxonomy to describe each team member's individual contributions. The submitting team member is responsible for providing the contributions of all authors at submission. We expect that all team members will have reviewed, discussed, and agreed to their individual contributions ahead of this time.

Contributor Role	**Role Definition**	**Team Members (List your names here)**
Conceptualization	Ideas; formulation or evolution of overarching research goals and aims.	Eva, Claudia, Tom, Samantha
Data Collection	Activities to search for, obtain, download datasets.	Tom
Data Curation	Management activities to annotate (produce metadata), scrub data and maintain research data (including software code, where it is necessary for interpreting the data itself) for initial use and later reuse.	Samantha
Formal Analysis	Application of statistical, mathematical, computational, or other formal techniques to analyze or synthesize study data.	Eva, Claudia, Tom, Samantha
Investigation	Conducting a research and investigation process, specifically performing the experiments, or data/evidence collection.	Claudia
Methodology	Development or design of methodology; creation of models	Eva – k-means Claudia – DBScan Samantha – t-SNE Tom – Spectral
Project Administration	Management and coordination responsibility for the research activity planning and execution.	Eva
Software	Programming, software development; designing computer programs; implementation of the computer code and supporting algorithms; testing of existing code components.	Eva, Claudia, Tom, Samantha
Validation	Verification, whether as a part of the activity or separate, of the overall replication/reproducibility of results/experiments and other research outputs.	Tom, Samantha
Visualization	Preparation, creation and/or presentation of the published work, specifically visualization/data presentation.	Eva, Claudia
Writing – Original Draft Preparation	Creation and/or presentation of the published work, specifically writing the initial draft (including substantive translation).	Eva, Claudia, Tom, Samantha
Writing – Review & Editing	Preparation, creation and/or presentation of the published work by those from the original research group, specifically critical review, commentary or revision – including pre- or post-publication stages.	Eva, Claudia, Tom, Samantha

8. References

1. Waters, Dan. “Where Should I Live?” *Where Should I Live?*, www.whereshouldilive.co/.
2. LaNeve, Julian. “Jlaneve/City-Picker.” *GitHub*, 21 Nov. 2023, github.com/jlaneve/city-picker?tab=readme-ov-file. Accessed 27 Apr. 2024.
3. Alhazzani, May, et al. “Urban Attractors: Discovering Patterns in Regions of Attraction in Cities.” PLOS ONE, vol. 16, no. 4, 26 Apr. 2021, p. e0250204, https://doi.org/10.1371/journal.pone.0250204. Accessed 22 Sept. 2022.
4. Erdem, Kemal. “T-SNE Clearly Explained.” *Medium*, Medium, 22 Apr. 2020, towardsdatascience.com/t-sne-clearly-explained-d84c537f53a. Accessed 29 Apr. 2024.