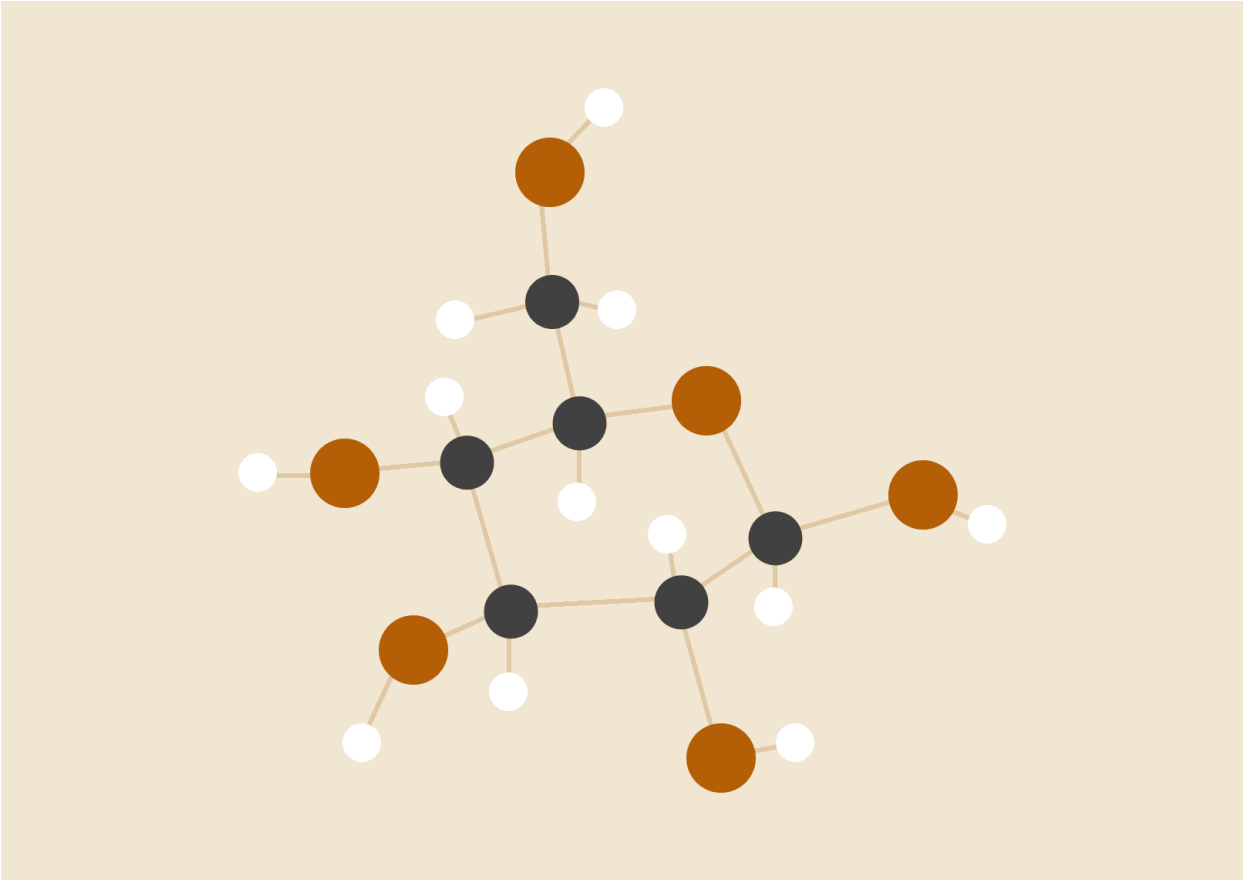
דוח מסכם



Adult Annual Income Dataset

Ohad Lavie - 209805787
Ronen Hristoforov - 318639028

Recap

In this project, an attempted clustering analysis was conducted with the goal of identifying population groups with similar characteristics, based on employment patterns and education level. The dataset contains demographic and economic information about adults, including variables such as age, education, occupation, type of work, years of education, and more.

The aim of the research is to examine how the population can be divided into meaningful groups according to employment and education characteristics, thereby uncovering insights about the relationships between education, types of employment, and income. Clustering algorithms (such as K-Means, DBSCAN, and others) were applied after data cleaning, transformation, and vectorization of categorical variables.

Introduction

The original purpose of this dataset was to predict whether a person's income exceeds $50,000 per year, but it also allows for a variety of other analyses, such as identifying social and occupational patterns. In this project, we focus on clustering methods to explore how the population can be grouped based on employment patterns and education levels, in order to gain insights into socioeconomic structures.

Socioeconomic patterns have been the subject of extensive research, particularly in relation to education and employment. While the *Adult Income* dataset has primarily been used for classification tasks—predicting income based on demographic and occupational features—it also holds significant value for exploratory analysis through clustering. By shifting the focus from prediction to grouping, we aim to uncover hidden population structures that reflect deeper social and economic divisions.

Literature Review

Clustering is an unsupervised machine learning technique that groups data points based on similarity, without predefined labels. It is widely used to explore large, complex datasets and identify underlying structures. In the context of socioeconomic and demographic data, clustering can reveal meaningful population segments that may not be apparent through traditional analysis.

- Anil K. Jain provides an overview of clustering algorithms and highlights their effectiveness in high-dimensional datasets, especially when combined with careful preprocessing and feature selection—essential steps when working with both numerical and categorical data.
- Sanjay Chakraborty and N.K Nagwani applied clustering techniques to census-like data and successfully grouped individuals based on education, occupation, and income, demonstrating how these methods can provide interpretable insights into workforce structures and broader socioeconomic trends.

Methods and Results

EDA

For each step of the project we used a variety of techniques to try and get the best results. After we cleaned our data of irrelevant data or null values that did not contribute to anything, we plotted some of the values to see what other data was irrelevant. From the plots, we saw that we could use feature engineering to better manipulate our data in order to get better insights and results.

After that we used a correlation matrix to try and see if there was any meaningful correlation between the numerical columns. There wasn't.

We tried to view from violin plots, if we could see any correlation that we couldn't otherwise see from the matrix, and to try and get a better understanding of how the data is distributed. These helped us gain many observations on our data which we used for more feature engineering and it helped us think of a research question for our project.

Dimensional Reduction

The next step was to try and cluster our data. We ran PCA and then evaluated the best threshold for variance for our clustering algorithms.

We ran PCA on various variance thresholds from 0.70 – 0.90 got:

- 6 components explain ≥70% of variance.
- 10 components explain ≥80% of variance.
- 13 components explain ≥85% of variance.
- 17 components explain ≥90% of variance.

After multiple tries on 90% and 85% thresholds, to be less objected to dimensionality problems we decided to do our PCA on 10 components, which means our 10 most significant components will capture 80% of the total variance.

K-MEANS

How it works: Partition N observations into K groups (clusters) so that points within each cluster are as similar as possible, and points in different clusters are as dissimilar as possible.

To try and find clusters we started running clustering algorithms. We started with K-MEANS and using the elbow method found that the best ‘k’ for use would be 3. We attached cluster labels to our dataframe and ran K-MEANS using ‘cluster’, one time as a hue, and another time with ‘Education-Group’ as a hue. We then plotted this and saw that the feature engineering that we did on education, ‘Education-Group’, was pretty much aligned with PC1, meaning it is basically a principle component.

To verify our assumptions we ran a silhouette score test on ‘Education-Group’ and got a very low score (0.096) for k=3, which is a very low score for clustering algorithms. We deduced from this that in n-dimensional data, even though one column can capture ~26% of the variance (PC1), it is not a clear separator of the other n-1 dimensions.

To show this, we plotted scatter plots of PC3 and PC4 (arbitrary PC’s), colored by

‘Education-Group’ and got plots that were very ‘fuzzy’ and there was no clear separation.

After getting a silhouette score of 0.203, that was as optimal as possible, we decided to move on to a different algorithm.

DBSCAN

How it works: A clustering algorithm that groups points based on density — not distance to a centroid like K-Means.

DBSCAN uses two main parameters:

- ε (epsilon): Radius of neighborhood around a point.
- minPts: Minimum number of points required to form a dense region.

We chose minPts as number of dimensions + 1 (common rule of thumb)
To decide on the right epsilon we used a k-distance graph, and determined the right epsilon by the elbow of it. The process of determining the right epsilon was tricky, as from the K-Distance graph, it was unclear which epsilon would be best.

Overall we tried running DBSCAN but got very low silhouette scores (0.064), even though visually, some of the plots seemed promising. We decided to move on to Hierarchical Clustering.

Hierarchical Clustering

Hierarchical clustering builds a tree-like structure (called a dendrogram) that shows how data points can be grouped at different levels of granularity and uses Linkage criteria for merging clusters. We tried the Agglomerative method which starts with individual points as clusters, and merges them step by step. We then tried the single, complete, average and Ward’s method and all of these were measured with Euclidean distance. The best result was using Ward’s method which tries to minimize the increase in total within cluster variance. For Ward’s method we got a silhouette score of (0.17599), which isn’t that great, so we decided to try a different method for distance.

Cosine Distance

After trying multiple algorithms, without any seen improvement, we then started researching ways to improve our algorithms. One way was to use a different distance metric. The metric that was recommended to use for high dimensional data as it performs well with high dimensional data where Euclidean distance can become less meaningful.

Cosine distance is one minus the cosine of the angle between two vectors in feature space. For this to work, we had to normalize our data. We then proceeded to run K-Means clustering.

K-Means

After running a silhouette score test we got a silhouette score of 0.251, which is ~25% higher than K-Means with Euclidean distance metric. We concluded that this is a good sign and we will try to run Hierarchical clustering with cosine distance to see if the score can get better. Spoiler alert, it did!

Hierarchical clustering

After running a silhouette score test we got a silhouette score of 0.3316 which is almost 2x higher than HC with Euclidean distance metric, and overall this is our best result yet.

We then tried to run this algorithm on PCA embedding of 85% variance with motivation to get an even higher score, but got a score of 0.297 and concluded that the best result was 0.3316.

Conclusion

Given the data we got, and our research question : “How can clustering analysis be applied to population data to form meaningful groups based on employment patterns and education level?”

We can say that :

Clustering can be applied to a population data using :

1. **Selecting** key education and employment features.
2. **Reducing** dimensionality to capture 80 % of variance while preserving structure.
3. **Normalizing** for cosine-style comparisons.
4. **Applying** a Hierarchical Agglomerative Clustering with average linkage + cosine metric on the L_2-normalized PCA-80% embedding to determine clusters .

We discovered three interpretable segments that directly reflect differences in education level (HS-Graduates vs Some-College/Assoc vs professional–role backgrounds) and occupational specialization (administrative vs craft‑repair vs professional‑specialty), along with demographic patterns—while all work full time, it’s their role and education that really set them apart.

In our data the clusters we got were:

Cluster 0: Entry-Level Administrators

- Age: on average ~36 years old (median 33), but the single most common age is 23.
- Work & Family: overwhelmingly Private-sector, full-time, in Adm-clerical; most are never-married and not-in-family.

- Education: mode = HS-Graduate, with very low capital-gain/loss and income flag (~6 % have any gain, almost none > 50K).
- Demographics: almost all White, ~46 % female (sex_mean ≈ 0.46).
- Income: income_mean ≈ 0.058 means only about 6 % fall into the high-income (> 50K) bracket.

This is a young, entry-level administrative cohort—mostly high-school grads, full-timers in clerical roles, very low capital gains, and low rates of high income.

Cluster 1: Early-Career Associates

- Age: on average ~35 years old (median 33), most common age = 20.
- Work & Family: also Private and full-time in Adm-clerical, but these tend to be spouses rather than independent "not-in-family."
- Education: mode = Some College/Assoc, with somewhat higher capital-gain frequency (13 % have any positive gain) and a higher high-income rate (~13 %).
- Demographics: still predominantly White, but more skewed female (sex_mean ≈ 0.64).

Interpretation: a younger, post-secondary-educated administrative cohort—many college/associate attendees, higher female representation, moderate capital gains, and moderate (≈13 %) high-income attainment.

Cluster 2: Mid-career Professionals

Observations:

- Age: older, average ~43 years (median 42), mode age = 38.
- Work & Family: mostly Private, full-time, but in Prof-specialty roles; nearly all are married-civ-spouse with the "spouse" relationship.

- Education: mode = HS-Graduate (surprising, but note many may have "HS-Graduate" highest degree but professional roles), and this cluster has

the highest capital-gain incidence (42 % have positive gains) and high-income rate.
- Demographics: again mostly White, ~89 % male (sex_mean ≈ 0.89).
- Interpretation: mid-career professionals—older, married, in specialized roles and highest rate of > 50K earners.

We believe (with a certain degree of certainty) our results can be seen as a blueprint to any population dataset with similar features.

Based on this type of analysis, by segmenting residents according to their education background and job profiles, cities or governments can tailor workforce development offerings: *

- Cluster 0 (high-school grads in clerical roles) → foundational digital and administrative skills workshops
- Cluster 1 (some-college associates balancing part- and full-time hours) → flexible, family-friendly certificate programs
- Cluster 2 (mid-career professionals in specialty roles) → leadership development and advanced financial planning courses.

*Cluster number and characteristics can vary per data set, but the blueprint remains.

Bibliography

We used a variety of sources to better help us understand what techniques to use and algorithms, and how to understand what they give us. These are the sources we used:

https://www.sciencedirect.com/science/article/abs/pii/S0167865509002323

https://arxiv.org/pdf/1406.4751

https://medium.com/@sachinsoni600517/mastering-hierarchical-clustering-from-basic-to-advanced-5e770260bf93

https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/

https://www.naftaliharris.com/blog/visualizing-k-means-clustering/

https://medium.com/@dishantkharkar9/k-means-clustering-algorithm-ce4fbcac8fb0

https://en.wikipedia.org/wiki/K-means_clustering

https://en.wikipedia.org/wiki/Hierarchical_clustering

StatQuest: Hierarchical ClusteringYouTube · StatQuest with Josh Starmer20 Jun 2017

https://simulationbased.com/2021/01/05/introduction-to-t-sne-in-python-with-scikit-learn/

https://scikit-learn.org/stable/modules/clustering.html

https://chatgpt.com/

What We Would Have Done Differently

Along the journey of doing this project we identified a few things we could have done differently and more efficiently. Firstly since we had to rely on ourselves for understanding the material, algorithms, what to do and how it should look, instead of looking up most of the things we needed to know during the process of the project, it would have been better to go over some things beforehand, although we didn't really know the direction the project was going in and what we were doing, there were some things we could have looked up that would have been beneficial and saved us some time.

Something else that we could have done differently was from the beginning to correct spelling mistakes as we typed them, as this would have saved us quite a lot of time.

Lastly, looking at animations of the clustering algorithms before we attempted them would have helped, but this was a bit problematic as we wanted to learn together and debate on what to do, however the war, and other courses, took up a lot of our time.

Overall we could have done things more efficiently, however under the circumstances we feel as though we did well.