Clustering and Regression on EU Data

Shubhang Tyagi, Alexandro Cabanillas, Zhaojing Wang, Zhen Niu, Lang Liu
MATH 108C Spring Quarter 3:30PM
Instructor: Maria Isabel Bueno Cachadina
June 12th, 2025

1 Abstract

Fertility is a crisis that threatens to undo entire cultural groups, and the worst part is that many positive things seem to reduce fertility. It's well known that factors like education, income, and human rights do not do great things to fertility, and so a lot of the best societies to live in appear to be unsustainable, since they limit their own fertility so much that societies with these things naturally screen themselves out. As such, we chose a set of features to see if, when we introduce more granularity, the micro effects differentiate from the macro effects, and instead of manually sorting, our K-Means did this to have a high degree of fidelity.

2 Background and Methodology

In this section, the mathematical principles that this project is based on is discussed and the procedure is explained.

2.1 Elbow Method

Purpose:

Determine the optimal number of clusters for clustering algorithms.

How it Works:

Run clustering algorithm for a range of different k values, and for each k, calculate the sum of squares within each cluster. Then plot k on the x-axis and sum of square on the y-axis.

As k increases, the sum of squares always decreases. However, the rate of decrease slows down dramatically after a certain point, and it is called the "elbow" of this plot.

Limitations:

Sometimes, the elbow is not visually obvious, which requires other methods.

2.2 K-means Clustering

Purpose:

Group an unlabeled dataset into k distinct clusters, where data points within the same cluster are as close to each other as possible.

How it Works:

Select k points randomly as initial cluster centroids. Then assign each data point to the nearest centroid, and calculate the new centroid of each cluster. Repeat this process until the clustering reaches a balance.

Limitation:

It is highly dependent on the choose of k and initial cluster centroids. Also, it only works for convex datasets.

2.3 Regression

Purpose:

Model the relationship between one or more independent variables and a dependent variable by finding a equation that best predicts the relationship between variables.

How it Works:

Find the best-fitting line by minimizing the sum of squared residuals.

Limitation:

Regression is sensitive to outliers, and may have the problem of overfitting.

2.4 Procedure

The language in which the mathematical backbone is coded in is Python. Below is the explanation of the attached code itself.

1. Data Retrieval:
   The data was taken from the EU website. In order to ensure a high quality of data, NUTS 3 Region was used which means the fidelity of the data is high resulting in data per county of Europe. The features extracted represented facets of fertility. Not all features had the same countries. In order to fix this issue, the intersection of all the counties that fit in all the features were subset and features for each county were labelled.

2. Data Cleaning:
   In order to ensure accuracy in the analysis, the data was cleaned by removing rows with any missing values as the regression requires a complete data set. The counties were also discarded as the K-means algorithm written required only numerical data. This was done with the pandas library. To ensure fair weight in distance-based algorithms, each feature was normalized. This was done with the scikit library.

3. Writing the algorithms:
   The K-means algorithms procedure was written. Along with personal enrichment, doing this by hand guarantees customizability. A pitfall is numerical stability which is what was encountered. Iterative multiplication were used to implement the K-means function. Spectral clustering and linear regression was implemented with scikit which ensures stability and quicker runtime.

4. Constructing the pipeline:
   The pipeline begins with the data cleaning. After generating a data-frame of the data, K-means was run 10 times to ensure a high level of precision in finding the optimal level of clusters. Using the elbow method discussed above, the optimal level of clusters which turned out to be 5. Graphs were plotted to verify the difference between K-means and Spectral means. Then the cluster labels were added to the data frame in order to perform linear regression on each cluster. The functions implemented were used for linear regression. The feature averages were multiplied by the coefficients to predict the label. The outputs of the feature averages, coefficients and predicted and actual label were printed to test validity.

3 Results

3.1 Effect of Preprocessing

We preprocessed the data, with a lot of the dataset being centered on 0. This had minimal effects except blowing up these coefficients. See, for the highly collinear ones like the family types—when normalized, they become tiny; the average features are 0.00X or something of that magnitude. That makes it so an additional whole 1-unit increment is wildly large. This makes it difficult to work with on these, but really not much trouble. In the future, it should be done with a log regression to make it in terms of percentages, but it works fine for our other features.

3.2 Age

Regression analysis: some of these are rather unsurprising. The coefficient for median female age is negative for all clusters except 4, which more or less aligns with fertility peaking rather early. In cluster 4, it is 0.1136, which is admittedly more strange. Perhaps this has to do with there being a limit to the lifespan, so an additional year means more babies, but the older folks dying means that the effect looks positive. For median age of males, it's a bit more all over the place, with some being rather significant at 1.28 in cluster 0 to -10 in cluster 3. I suspect this is a bit harder to parse out because this would have to do with incomes and other economic factors. Males almost universally need to build up a nest egg as a necessary prerequisite to having children, and this is

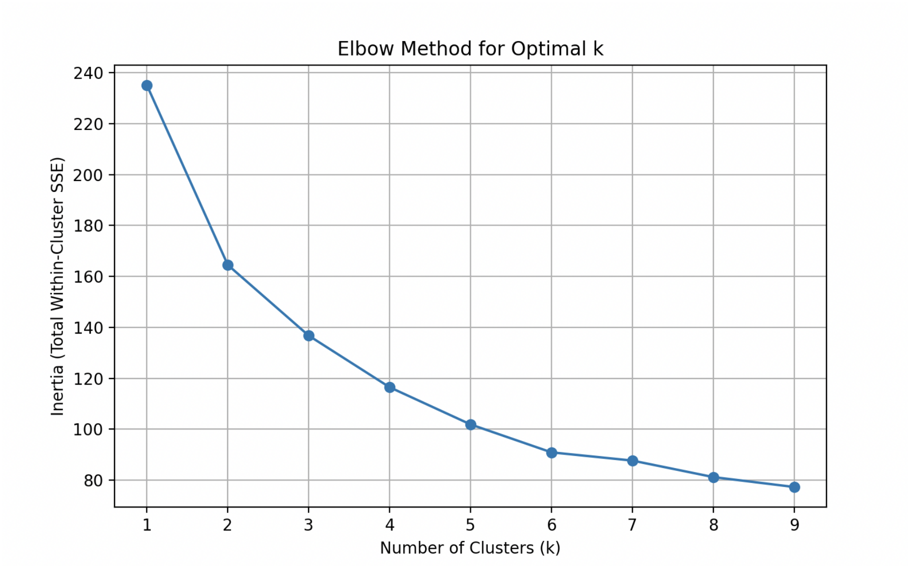

Figure 1: Elbow Method

thus interacting with the GDP variable in ways that are outside the scope of this regression structure. Interestingly, the coefficient for median age of the population seems basically independent of anything which is not the average age. It is positive and negative for different clusters, but I suspect it's because the average age in these clusters varies widely.

3.3 Family Types

Here, the preprocessing has affected things the most. Because these are normalized as well as divided by the population, they all have tiny observations but massive coefficients. We see the feature one-family nucleus has very differential effects—close to zero for some—but importantly, in our North Sea cultures, we see the largest positive effect by far. For our Catholics and Central Europeans, not so; but for our modernized Eastern Europeans, very strong positive effects on fertility, while in our Southeastern Europeans (cluster 4), not so—super negative. This has to do with cultural modernization. In Southeastern Europe, raising children is super communal, and so one-family nucleuses are doing so without family support, without the cultural systems that ones with more of these have to compensate. While our Eastern Europeans (cluster 3) with the one-nucleus families are a bit more modernized, as well as having higher incomes than the surrounding population, which explains some of the positive fertility effect.

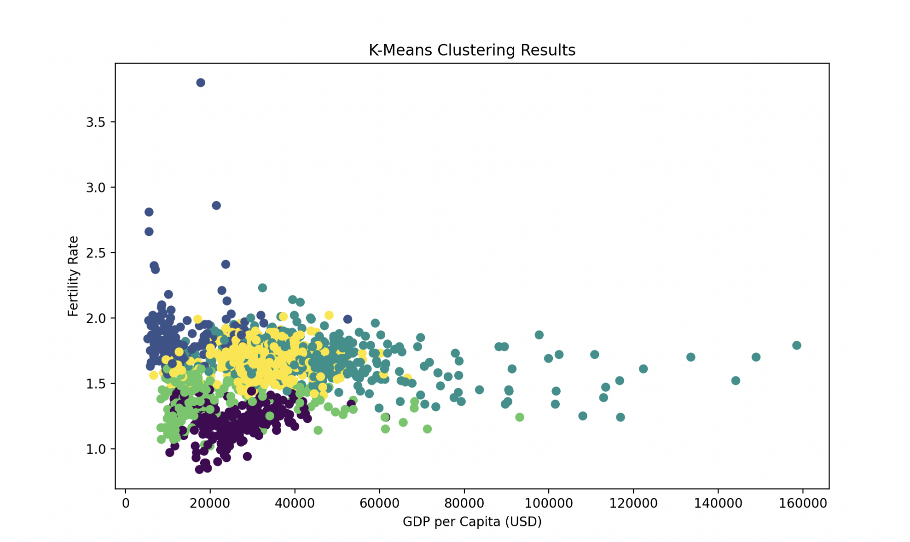

Figure 2: K-Means Cluster

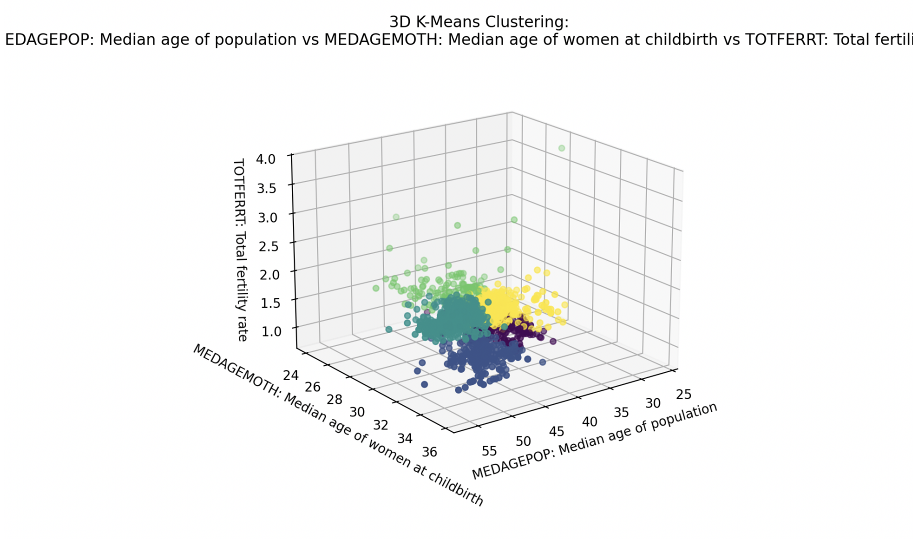

Figure 3: 3D Clusters

3.4 GDP

GDP in all groups positively correlates, which is interesting since, on a national level, higher incomes lower fertility rates. So it is a cultural thing—richer countries have cultures that deprioritize fertility in favor of education, which raises incomes—but within the country, more income means more kids. It is rather interesting that it holds like that because it could have to do with labor mobility, and anywhere there are no jobs, young folks will tend to move, which is hard to see without interaction variables.

```
Cluster 0:
Feature                                Coefficient
Intercept                               1.9889
Median age of females                  -0.0767
Median age of population               -1.4993
Median age of males                     1.2838
Couples                                -106.8404
Couples without resident children       52.7973
One resident child under 25 years old   63.9832
One - family nucleus                    20.1902
Total households                       -23.9759
GDP per capita                          0.0969
Mean age of women at childbirth        -0.2830
Median age of women at childbirth       0.1011

Cluster 1:
Feature                                Coefficient
Intercept                                1.3562
Median age of females                   -3.1276
Median age of population                 6.0748
Median age of males                     -3.0445
Couples                                 -28.8242
Couples without resident children        16.3492
One resident child under 25 years old    11.8922
One - family nucleus                    -0.3210
Total households                         5.4974
GDP per capita                           0.0010
Mean age of women at childbirth         -0.3992
Median age of women at childbirth        0.4852

Cluster 2:
Feature                                Coefficient
Intercept                                1.9510
Median age of females                   -6.0818
Median age of population                 10.1029
Median age of males                     -4.0149
Couples                                 -7.9738
Couples without resident children        6.6571
One resident child under 25 years old    11.5650
```

```
One - family nucleus                  -11.3150
Total households                       1.8050
GDP per capita                        -0.1461
Mean age of women at childbirth       -1.2198
Median age of women at childbirth      0.8970

Cluster 3:
Feature                                Coefficient
Intercept                              2.8566
Median age of females                 -10.5540
Median age of population               19.9173
Median age of males                   -10.3771
Couples                               -280.8563
Couples without resident children      82.4663
One resident child under 25 years old  112.2270
One - family nucleus                   23.9579
Total households                       72.4030
GDP per capita                         0.3612
Mean age of women at childbirth        0.6957
Median age of women at childbirth     -1.3805

Cluster 4:
Feature                               Coefficient
Intercept                              1.9710
Median age of females                  0.1186
Median age of population              -0.5090
Median age of males                    0.0040
Couples                                38.9255
Couples without resident children     -3.4075
One resident child under 25 years old  4.6929
One - family nucleus                  -48.5311
Total households                       11.0260
GDP per capita                         0.5119
Mean age of women at childbirth        0.3367
Median age of women at childbirth     -1.0525
```

4 Conclusion and Future Work

4.1 Conclusion

This project investigates the relationship between family structure and national outcomes across European countries. By applying K-means, we identified five distinct clusters that group countries with similar family-related indicators. The clustering enabled us to perform regression analysis within more homogeneous subsets of countries, which reduced the impacts caused by confounding variables.

Our regression results revealed that certain variables, such as the median age of the population and household composition, show varying levels of impact on fertility rate across different clusters. For example, the number of one-family households exhibited a shift in its effect—positive in some clusters and negative in others—suggesting that its influence depends on the broader demographic context. These variations highlight the importance of clustering as a way to account for structural differences between countries before modeling.

Overall, the combination of unsupervised clustering and regression analysis provided a complicated understanding of how family structure interacts with economic and demographic outcomes, and provided an idea to study such relationships with reduced confounding effects.

4.2 Future Work

While K-means provided useful insights, this method has limitations such as assuming convex clusters and sensitivity to initialization. To improve robustness and flexibility, future work could explore alternative algorithms. On the regression side, using models like non-linear regression could better capture complex relationships between family structure and national outcomes.

Another valuable extension would be to apply this analysis to datasets from other regions beyond Europe. Family structures and demographic patterns differ widely across cultural and economic contexts, and comparing results across datasets could help test the generalizability of our findings. This would also allow us to see whether the observed clustering patterns are unique.

5 Pipeline

```python
import pandas as pd
import kMeansCluster as kMeans
import numpy as np
from numpy.linalg import matrix_rank
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches
import leastsquaresbestfit as ls
from sklearn.preprocessing import MinMaxScaler, StandardScaler
import geopandas as gpd
from sklearn.cluster import SpectralClustering
from sklearn.linear_model import LinearRegression

# Load the dataset
file_path = 'eu_common_2021_indicators.csv'
df = pd.read_csv(file_path, encoding='utf-8')
print(f"Dataset shape: {df.shape}")
# Remove rows with missing values
cleaned_df = df.dropna()
# Remove the 'country' column
cleaned_df = cleaned_df.drop(columns=['country'])
# Normalize the data except for the 'Total fertility rate'
# Identify columns to normalize
```

```python
columns_to_normalize = cleaned_df.columns[cleaned_df.columns != 'TOTFERRT: Total fertility rate']
# Initialize the scaler
scaler = MinMaxScaler()
# Normalize the selected columns
cleaned_df[columns_to_normalize] = scaler.fit_transform(cleaned_df[columns_to_normalize])
# Display the head of the cleaned and normalized dataframe
print(cleaned_df.head())

#Elbow method to find optimal number of clusters
print("\n" + "="*50)
print("ELBOW METHOD FOR OPTIMAL K")
print("="*50)
kMeans.findK(cleaned_df, k_range=range(1, 10))
# Kmeans clustering with k=5
print("\n" + "="*50)
print("TESTING K-MEANS CLUSTERING")
print("="*50)
k = 5
X, C = kMeans.createMatricies1(cleaned_df, k)
print(f"Data shape: {X.shape}")
print(f"Initial centroids shape: {C.shape}")
A_final, C_final = kMeans.kMeans(X, C, k, max_iter=100)
   # Get cluster assignments
cluster_labels = np.argmax(A_final, axis=1)
cleaned_df['K-Cluster'] = cluster_labels
print(f"\nCluster distribution:")
print(pd.Series(cluster_labels).value_counts().sort_index())
# Spectral clustering
print("\n" + "="*50)
print("TESTING SPECTRAL CLUSTERING")
print("="*50)
n_clusters = 5
model = SpectralClustering(
    n_clusters=n_clusters,
    affinity='rbf',
    gamma=0.5,
    random_state=42
)
clusters = model.fit_predict(cleaned_df)
# Add spectral clustering results to the original dataframe
#cleaned_df['Spectral_Cluster'] = clusters
cleaned_df.to_csv('eu_common_2021_indicators_with_clusters.csv', index=False)
print(f"\nCluster distribution:")
print(pd.Series(clusters).value_counts().sort_index())
feature_cols = [col for col in cleaned_df.columns if col != 'TOTFERRT: Total fertility rate' and col != 'K-Cluster']
```

```python
for cluster_id in range(k):
    cluster_data = cleaned_df[cleaned_df['K-Cluster'] == cluster_id].copy()
    if len(cluster_data) > 0:
        print(f"\nCluster {cluster_id} ({len(cluster_data)} samples):")

        # Design matrix and label
        X = cluster_data[feature_cols].values
        y = cluster_data['TOTFERRT: Total fertility rate'].values

        model = LinearRegression()
        model.fit(X, y)

        # Intercept and coefficients
        intercept = model.intercept_
        coeffs = model.coef_

        # Rank check
        is_full_rank = matrix_rank(X) == X.shape[1]

        print(f"Design matrix shape: {X.shape}")
        print(f"Is full rank: {is_full_rank}")
        print(f"Coefficients shape: ({coeffs.shape[0] + 1}, 1)")

        # Compute average values
        cluster_means = cluster_data[feature_cols].mean()
        label_avg = cluster_data['TOTFERRT: Total fertility rate'].mean()

        print("Feature Averages and Coefficients:")
        print(f"{'Feature':<30}{'Average':>15}{'Coefficient':>15}")
        print("-" * 60)
        print(f"{'Intercept':<30}{'{':>15}{intercept:>15.4f}")  # Intercept
        prediction_at_mean = intercept

        for i, name in enumerate(feature_cols):
            avg_val = cluster_means[name]
            coef_val = coeffs[i]
            prediction_at_mean += avg_val * coef_val
            print(f"{name:<30}{avg_val:>15.4f}{coef_val:>15.4f}")

        print(f"{'Label average (Fertility Rate)':<35}{label_avg:>15.4f}")
        print(f"{'Prediction at mean features':<35}{prediction_at_mean:>15.4f}")
# Save the country-cluster mapping
# Plot K-Means
```

```python
colors = ['red', 'blue', 'green', 'orange', 'purple']  # up to 5 clusters
cluster_colors = [colors[label] for label in cluster_labels]

plt.figure(figsize=(10, 6))
plt.scatter(df.iloc[:, 9], df.iloc[:, 12], c=cluster_colors, s=40)
plt.xlabel('GDP per Capita (USD)')
plt.ylabel('Fertility Rate')
plt.title('Spectral Clustering Results')
plt.show()
# Plot Spectral Clustering
plt.figure(figsize=(10, 6))
plt.scatter(df.iloc[:, 9], df.iloc[:, 12], c=clusters, cmap='viridis')
plt.xlabel('GDP per Capita (USD)')
plt.ylabel('Fertility Rate')
plt.title('Spectral Clustering Results')
plt.show()
```