World Bank Open Data Analysis

Michał Grela, Mateusz Kocot, Maciej Trątnowiecki

May 2022

1 Introduction

2 Dataset description

- rozkład średnich, median, itd.

3 Autoencoders

Autoencoders are neural networks consisting of two modules: encoder and decoder. An encoder encodes an input vector into a latent (embedding) space. Then, a decoder decodes this vector back to the original dimensionality so that it resembles the input vector with the least possible error. After training an autoencoder, the encoder module can be used for feature extraction. We used this approach to extract time-independent features from our time series.

3.1 Data preprocessing

Before playing with neural networks, we had to preprocess the data. We could not simply standardise the data, column by column, since it would have ruined the neighborhood relationship between samples. Therefore, we normalised the data so that their norms would be equal to 1. We treated their default norms as one of the final features, apart from the features retrieved from an autoencoder. Therefore, if an autoencoder has 4 neurons in the bottleneck, the final size of the latent space is 5.

3.2 Experiments

At first, we used a single time series type ('Population growth (%)') to perform experiments and choose the best autoencoder architecture.

We tested many different architectures with numbers of the neurons in the bottleneck, that is the embedding dimensionality, from 1 to 8 (later referred to as `n_bottleneck`). We describe

the architectures briefly below. The best architecture will be described in-depth in the next section.

The first three models are regular networks with different numbers of dense-connected layers and the ReLU activations:

- `Autoencoder v1` with one layer (`n_bottleneck` neurons) in the encoder and the decoder,
- `Autoencoder v2` with two layers (20, `n_bottleneck` neurons) in the encoder and the decoder,
- `Autoencoder v3` with four layers (80, 40, 20, `n_bottleneck` neurons) in the encoder and the decoder.

The other three models are convolutional networks composed of convolutional layers with increasing numbers of filters, the ReLU activations and, respectively, max pooling or upsampling layers to decrease or increase the length of a time series. In the bottleneck we used two dense-connected layers without an activation to transform the features created by the convolutional layers into the latent space and then back to the input for the decoder. Again, the models have different numbers of convolution blocks (convolution, activation and max pooling or upsampling):

- `Autoencoder v4` with one convolution block (32 filters) in the encoder and the decoder,
- `Autoencoder v5` with two convolution blocks (32, 64 filters) in the encoder and the decoder,
- `Autoencoder v6` with three convolution blocks (32, 64, 128 filters) in the encoder and the decoder,

Before training the models, we investigated the complexity of the data using the Principal Component Analysis (PCA). We show the explained variance ratio plot in Figure 1. We also applied the inverse PCA transformation for different numbers of first PCA components and compared the original time series with the reconstructed ones by measuring the mean absolute error (MAE). The MAE values are presented in Figure 2. Note that the MAE values are multiplied by 10,000 for clarity. Since we use the MAE metric as a loss function, these values can serve as a reference in order to determine how much we gain from employing autoencoders with respect to the simple, PCA approach.

We used the Adam optimizer. We trained each of the models with a dynamic learning rate that was configured to decrease on a loss plateau, that is when the loss is not improving for a selected number of training epochs. The models were trained for the numbers of epochs corresponding to their complexity. Note that we did not split the dataset into training and testing set. We did not need the models to generalise so the split necessary. We present the comparison of achieved loss values in Figure 3.

The loss values look reasonable – the more complex the architecture and the more neurons in the bottleneck, the smaller the loss is. We also see higher performance of the convolutional approach in this problem. Therefore, we decided to use one of the convolutional models for

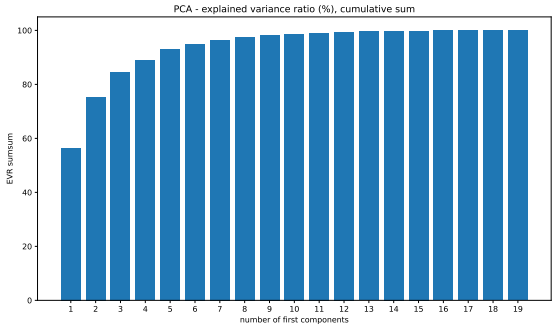


Figure 1: PCA explained variance ratio cumulative sum of first components

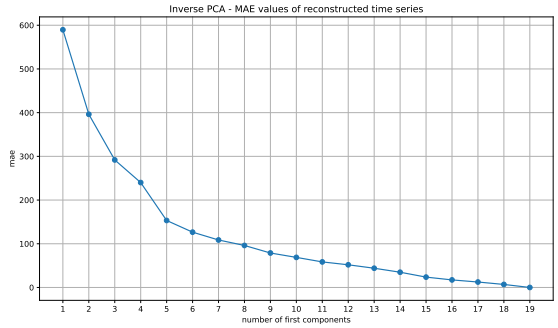


Figure 2: MAE values for the time series reconstructed by inverse PCA

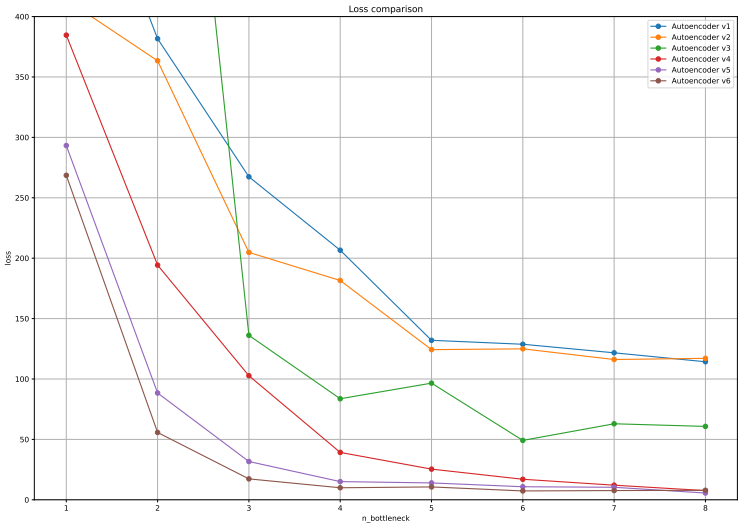


Figure 3: Comparison of the loss values achieved by the autoencoders

feature extraction. The efficiencies of `Autoencoder v5` and `Autoencoder v6` are very similar, so it would seem more efficient to use the smaller model. However, the ultimate goal of the model was to embed multiple time series types, so we decided to select the one with a higher capacity, that is `Autoencoder v6`.

Now, the losses of the autoencoders can also be compared with the MAE values of the inverse PCA reconstructions. While the loss curves of the first two models and PCA are similar, superiority of more complex architectures is clearly visible. The ultimate `Autoencoder v6` reaches the MAE value of around 20 while PCA requires about 14 dimensions in the latent space to achieve similar performance.

In the end, we compared the methods visually too. Figure 4 shows the difference of reconstruction quality between PCA and the best model. For this particular time series, autoencoder with `n_bottleneck=2` has a smaller MAE value than inverse PCA using 8 first components.

All training details can be found in the project repository [1].

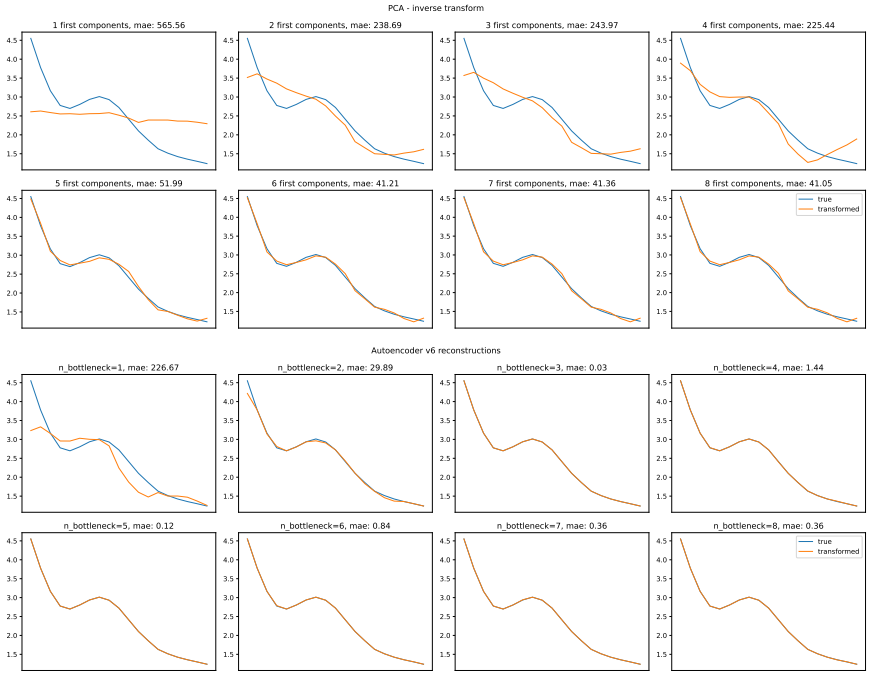


Figure 4: Comparison between reconstructions made by inverse PCA with different numbers of first components used and autoencoder (v6) reconstructions for different numbers of the neurons in the bottleneck

3.3 Final autoencoder model

The final autoencoder model takes time series with 19 samples as an input. It uses convolutional and max pooling layers to scale it to the length 10, 5 and 3 at the end. Number of convolution channels is increasing from 32, through 64, to 128. In the bottleneck, it uses a dense layer to encode the data in the latent space (`n_bottleneck` neurons) and another dense layer to go back to `3 * 128 = 384` numbers. Now, convolutional and upsampling layers are used to decode the data to the length 24 and 32 channels. The length is different than 19, since 19 cannot be reached from 3 only by multiplying by 2. Therefore, we added a dense layer at the end with 19 output neurons. The summary of the final model (with 4 neurons in the bottleneck) is printed in Appendix A.

We trained the model with all the time series we had. Later, we copied the weights and fine-tuned the autoencoder for each of the time series types. We performed this procedure for models with 2, 4 and 8 neurons in the bottleneck.

For instance, the network with 4 neurons in the bottleneck achieved MAE equal to 27.2 (Figure 5, left). Then, we fine-tuned it for each type of the time series. 'Inflation, GDP deflator (annual %)' turned out to be the easiest type, as the network reached MAE very close to 0 (Figure Figure 5, middle). On the other hand, 'Immunization, DPT (% of children ages 12-23 months)', with MAE equal to 6.5 was the most difficult one (Figure 5, right).

In Figure 3 we can observe so-called elbows for 2, 3 and 4 neurons, and after 4 there is a plateau, therefore we use only the model model with 4 neurons in the bottleneck during the

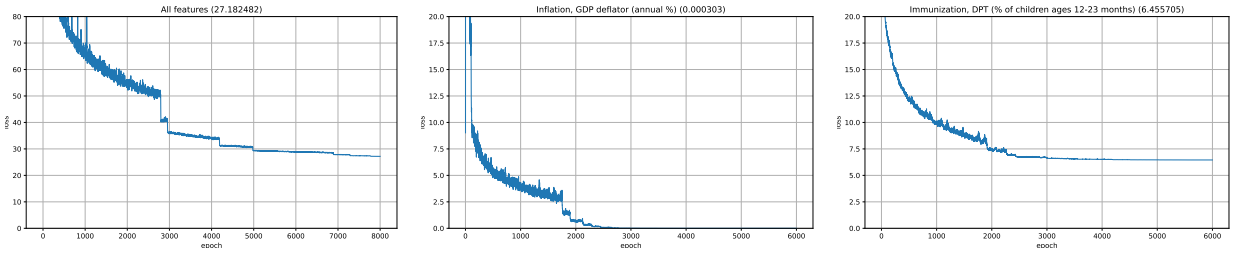


Figure 5: Loss curves for the autoencoder with 4 neurons in the bottleneck. Left: training on all features, middle: fine-tuning for the easiest time series type, right: fine-tuning for the most difficult time series type. The values of the loss functions are printed in parentheses. Note that the learning rate is automatically decreased during training, and thus the curves have a staircase shape.

analysis.

All training details can be found in the project repository [2].

3.4 Analysis of single time series types

At first, we analyse each time series type independently. Every time series is encoded into 5 values: norm and 4 values retrieved from the autoencoder. Each country is thus represented as a 5-number vector. These vectors are passed to the t-SNE and UMAP algorithms which are configured adequately for this type of data. Figure 6 shows some of more interesting results. A brief analysis is included below. The specific graphs from World Bank Open Data on which we base the analysis can be found in the appendix B. All the plots can be found in the project repository [3].

- Top left – Adolescent fertility rate. Poland is next to countries which do not seem to have much in common, e.g. Chile, Russia, Ukraine, New Zealand, France or Estonia. However, in all of these countries we can observe a decrease in the value of adolescent fertility rate with a more dynamic trend at the end. The magnitude is different (Chile – 40 in 2018, Poland – 10 in 2018), but the trend is similar. New Zealand, Ukraine and Russia form a smaller cluster of countries that had a small increase between 2002 and 2007. There is also an interesting cluster in in the top left corner. These countries share a huge decrease of the indicator. Again, the magnitude is a little different, but the plots are parallel, e.g. Honduras goes down from 116 in 2000 to 70 in 2018, and Jamaica – from 88 to 50.
- Top right – Exports of goods and services. We could expect that Poland would be next to countries like Czech Republic and Slovak Republic, but this is not the case. Poland, North Macedonia and Georgia that are next to each other share almost the same curves, they start from about 30% in 2000, and reach about 50% in 2018. On the other hand, the curves of Czech Republic and Slovak Republic are higher, while the curve of Bulgaria is, as expected, between the Polish and Czech groups.
- Bottom left – Fertilizer consumption. One of the clusters is located in the top left corner. It starts with Germany, and goes through UK, France, ending on Sweden. All

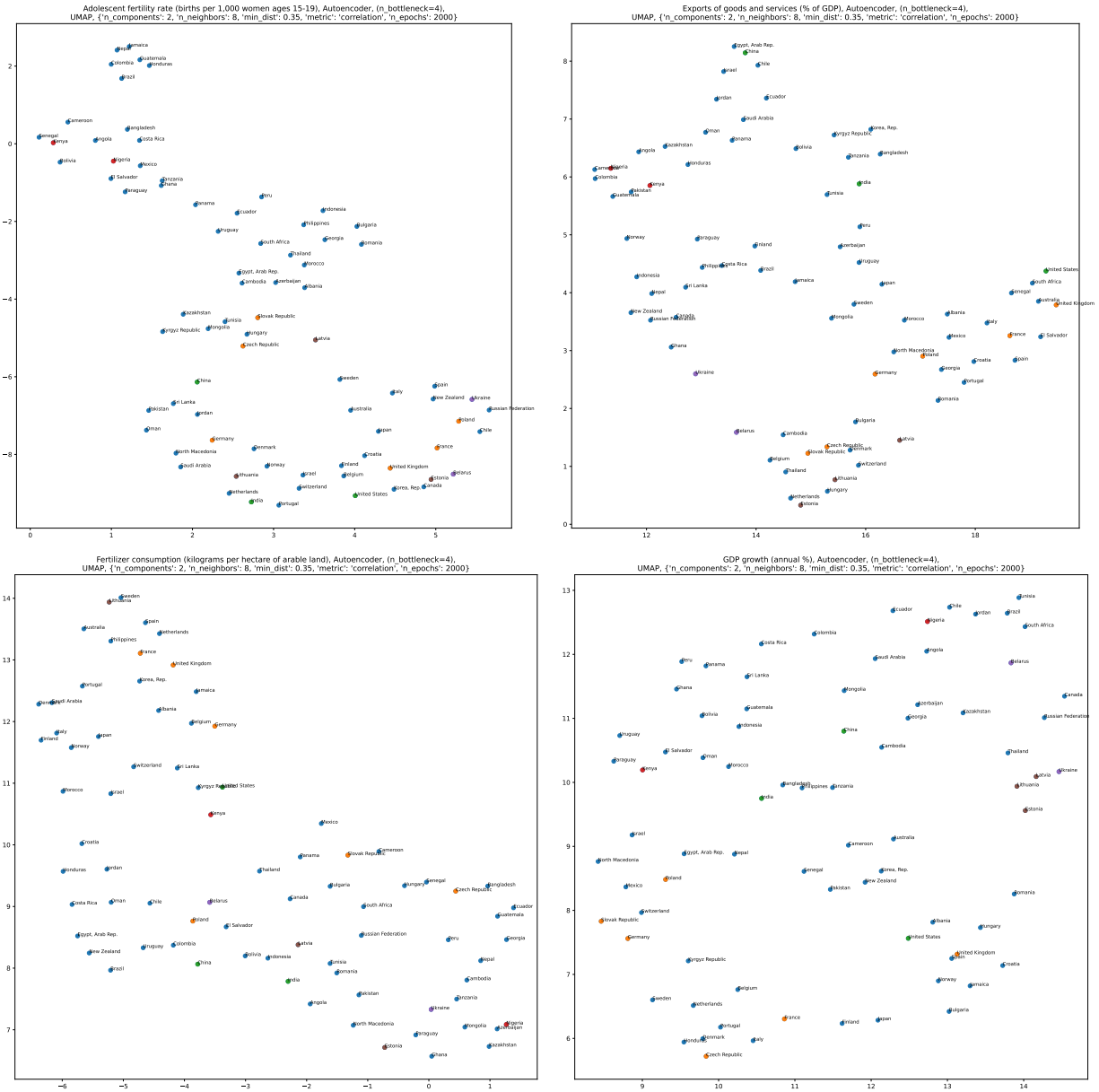


Figure 6: Some of more interesting visualisations of single time series types. Some countries are marked with different colours to increase clarity.

of these curves are similar, but their magnitude is a little different. The countries are not sorted in this order, however. What seems to be the order on the visualisation is the size of the 'pit' between 2006 and 2010. We can also compare these curves with the Polish cluster in the middle of the plot. The curves of Poland and El Salvador rise at the beginning of the twenties, while the values top-left-corner countries stay constant or decrease. On the other hand, the cluster in the bottom-left corner is composed of the countries using very small amounts of fertilizer, especially between 2000 and 2010.

- Bottom right – GDP growth. While Poland and Switzerland could never be next to each other in a GDP per capita plot, they share the same cluster in the GDP growth plot. We can compare them with Italy or Czech Republic, which are in another, but not distant, cluster. The main difference seems to be the size of the 'pit' near 2012 – Czech Republic and Italy lost more GDP than countries like Poland, Switzerland, Israel, Mexico or Slovac Republic. Another interesting fact is a presence of Albania next to USA and UK. Their curves are very similar starting from 2011 – they did not suffer that much as the previously mentioned countries in 2012.

3.5 Analysis of time series groups

Now we analyse the time series groups which also include the group of all the time series. Again, each time series is represented as a 5-number vector. We concatenate these vectors for each country, which results in vectors of a length 5 times longer than the number of time series types in a group. As previously, we use t-SNE and UMAP for visualisation purposes. Now, UMAP has two configurations though – one supporting more dense (`min_dist = 0.1`) and the other – less dense clusters (`min_dist = 0.4`). The results are show in Figure 7. Only one visualisation per group is shown, but all the plots can be found in the project repository [4].

- Top left – Agricultural indicators. We can observe a lot of clusters with neighbouring countries, e.g.: Poland, Germany; Belgium, Luxembourg, Netherlands or Latvia, Estonia, Lithuania. These countries have similar agricultural conditions. There are some nonobvious relations. For instance, Iran is located between Czech Republic and Sweden.

- Top right – Health indicators. It looks like the bottom of the plot (below -2.5) is filled with countries that do not suffer from extreme poverty, and the quality of health services is generally good, but varies. It seems, that the more we go to the right side of the plot, the better the health services are. So, on the left-hand side, there are Eastern Europe countries with life expectancy around 77. Life expectancy rises as we go to the right side of the plot. A presence of of Trinidad and Tobago with life expectancy of 73 years next to Sweden where people, on average, live almost 10 years longer is interesting. One possible explanation of this phenomenon could be the fact that some of the Trinidad and Tobago time series are very similar to their Swedish counterparts, just with a different magnitude. For example, the life expectancy curves are almost parallel. It is also important to remember, that the health group includes indicators like fertility rate or population growth which do not always depend on the quality of health services in a particular country.

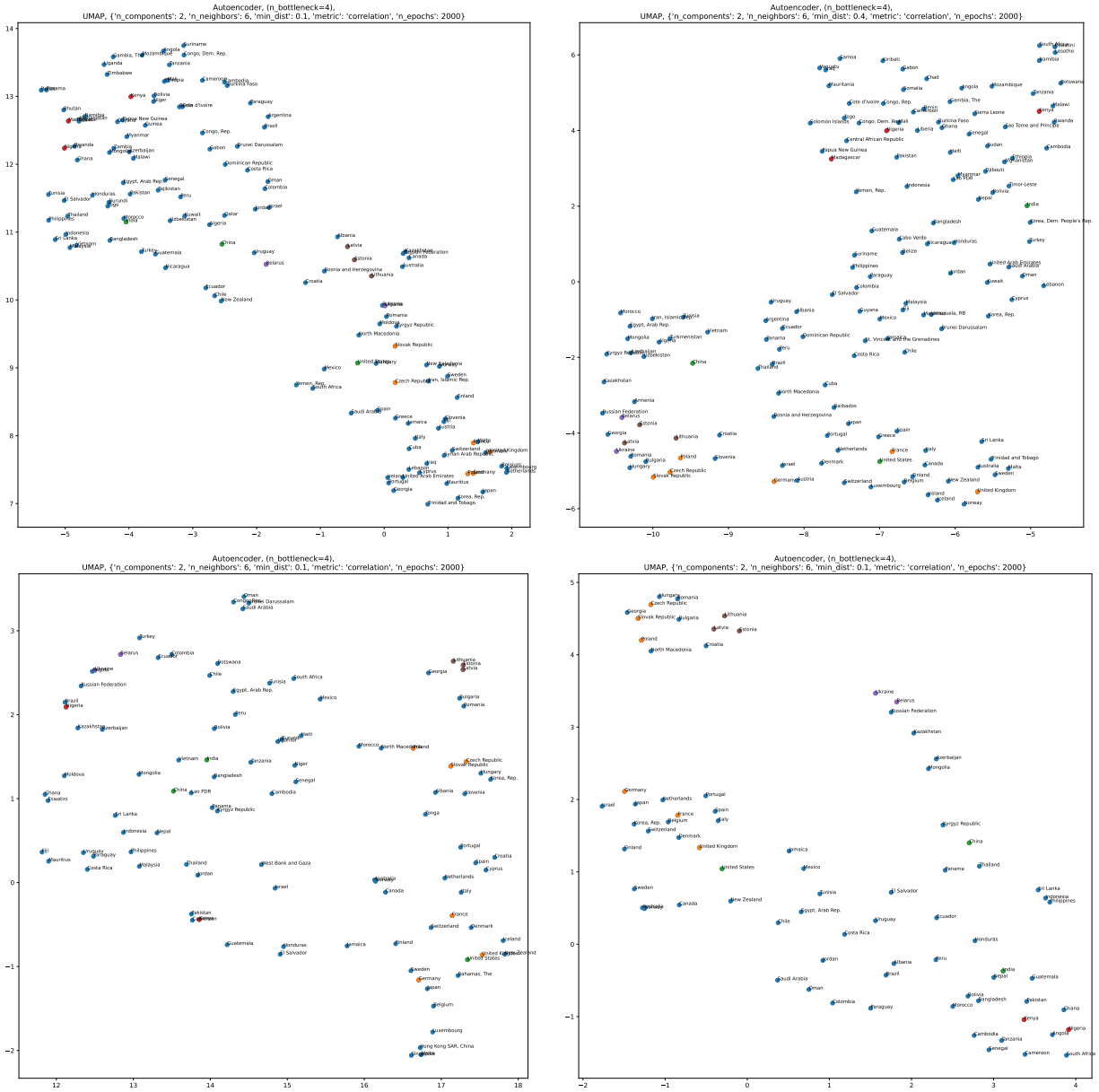


Figure 7: Time series groups visualisations. Top left: agriculture indicators, top right: health indicators, bottom left: economy indicators, bottom right: all indicators. Some countries are marked with different colours to increase clarity.

- Bottom left – Economy indicators. Again, there are a few obvious clusters like Lithuania, Estonia, Latvia, but Poland, for example, is in a cluster with North Macedonia (which is common) and Morocco. The GDP of Poland is almost 5 times bigger than the GDP of Morocco, and the GDP growth plots are different. A possible explanation could be the fact that Poland and Morocco are almost identical when it comes to a few indicators, e.g. imports of goods and services (% of GDP) or value added of industry (% of GDP).
- Bottom right – All indicators. This plot includes many obvious relations. The Eastern Europe countries (and Georgia whose GDP (annual %) is growing similarly to Poland) are in the top left corner. Then, the western countries with developed countries like USA or Canada occupy the left middle side. The post-soviet countries are located in the top right corner. Finally, the bottom right corner belongs to poor countries from Africa.

4 Hierarchical clustering

Hierarchical clustering is an unsupervised clustering approach which builds a hierarchy of clusters. The most common type of hierarchical clustering is agglomerative hierarchical clustering. In this strategy, each observation is initially treated as a separate cluster, then two closest clusters are merged into one. This iterative process repeats until all clusters are merged into one which contains every data point.

In order to decide which clusters should be combined, it is necessary to define distance metric and linkage criterion. The linkage criterion is a measure of the similarity between the two clusters. The results of hierarchical clustering are presented in a tree diagram showing the hierarchy of clusters to which each observation belongs, called a dendrogram. Final clustering is achieved by cutting the dendrogram with a horizontal line at a certain height, usually it is where the line can traverse the maximum distance up and down without intersecting the merging point.

4.1 Analysis

In our hierarchical cluster analysis, we chose euclidean distance as distance metric and Ward's method as linkage criterion. In this method, the choice of two clusters to merge is made on the minimum increment of total within-cluster variance. As it was said previously, each observation is initially treated as a separate cluster, so the sum of squares starts out at zero and grows with merging clusters. Ward's method allows keeping this growth as small as possible.

At the beginning, we clustered countries by each indicator in each group separately. Hierarchical relationships between countries are shown in the dendrograms: selected indicators in Figure 8, agriculture indicators in Figure 9, health indicators in Figure 10 and economy indicators in Figure 11. Then, looking at achieved diagrams, we determined the number of clusters. In the dendrograms each color represents a different cluster.

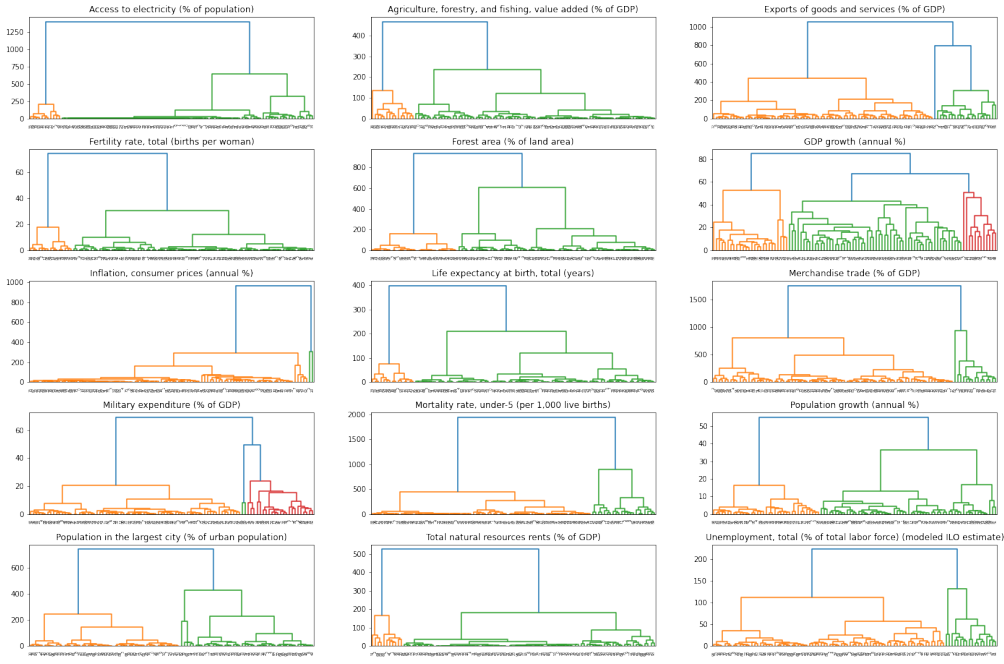


Figure 8: Dendrograms presenting hierarchical relationship between countries by selected indicators

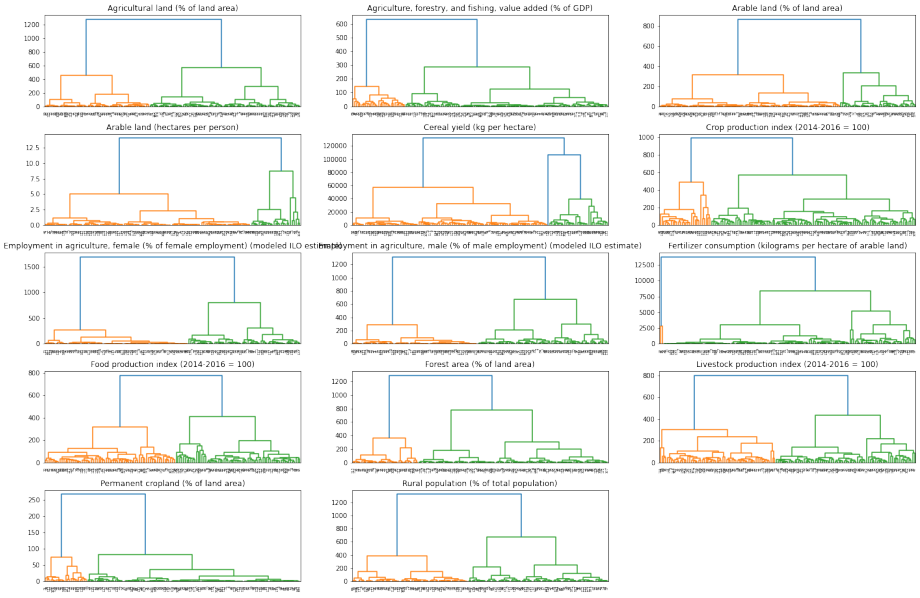


Figure 9: Dendrograms presenting hierarchical relationship between countries by agriculture indicators

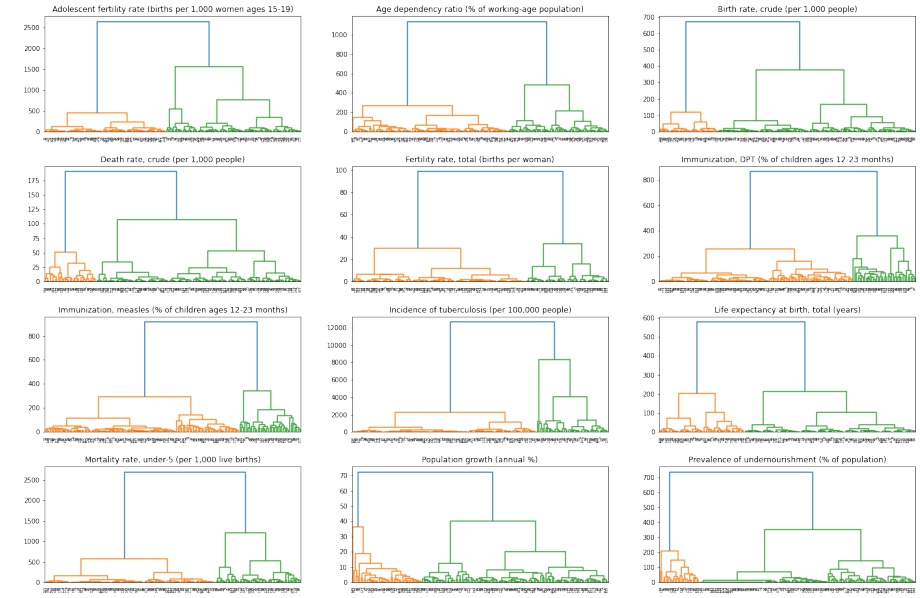


Figure 10: Dendrograms presenting hierarchical relationship between countries by health indicators

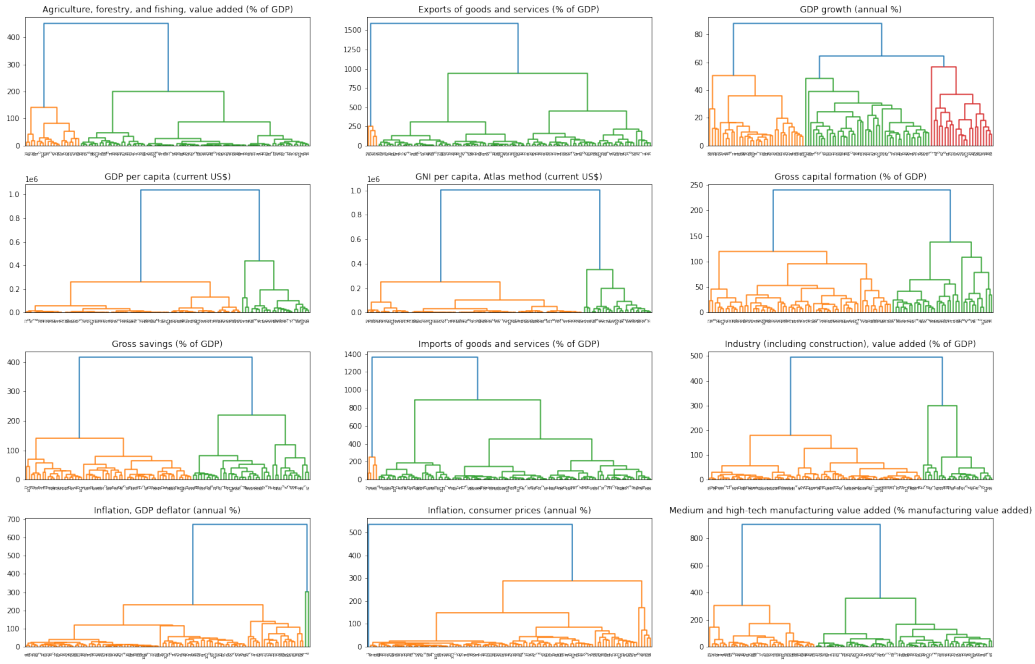


Figure 11: Dendrograms presenting hierarchical relationship between countries by economy indicators

After that, we used the Uniform Manifold Approximation and Projection (UMAP) method in order to reduce dimension. In this way, we were able to visualize the data in two dimensions and see clustering results. Due to the large number of visualizations, we decided not to include all of them in this paper. Also, clustering by each indicator separately was not our main goal, so we decided to focus on groups of indicators analysis. As an example, one of the clustering result is shown in Figure 12.

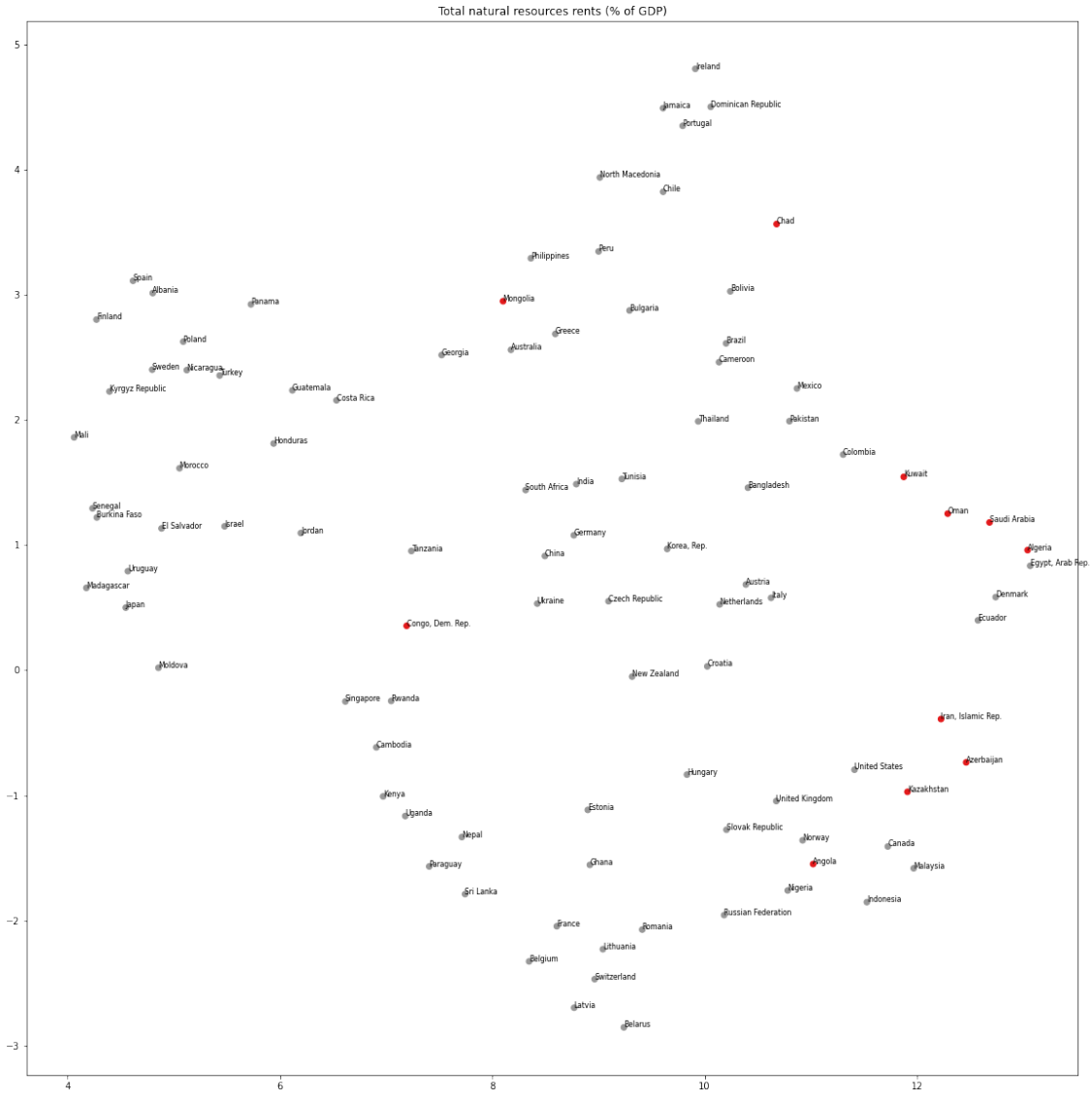


Figure 12: Countries clustered by total natural resources rents (% of GDP), each color represents a separate cluster

Subsequently, we repeated steps described above, but with feature extraction using one of the previously trained autoencoders (`Autoencoder v4`). Hierarchical relationships between objects from each indicator group, after feature extraction, are shown in dendrograms: selected indicators in Figure 13, agriculture indicators in Figure 14, health indicators in Figure 15 and economy indicators in Figure 16.

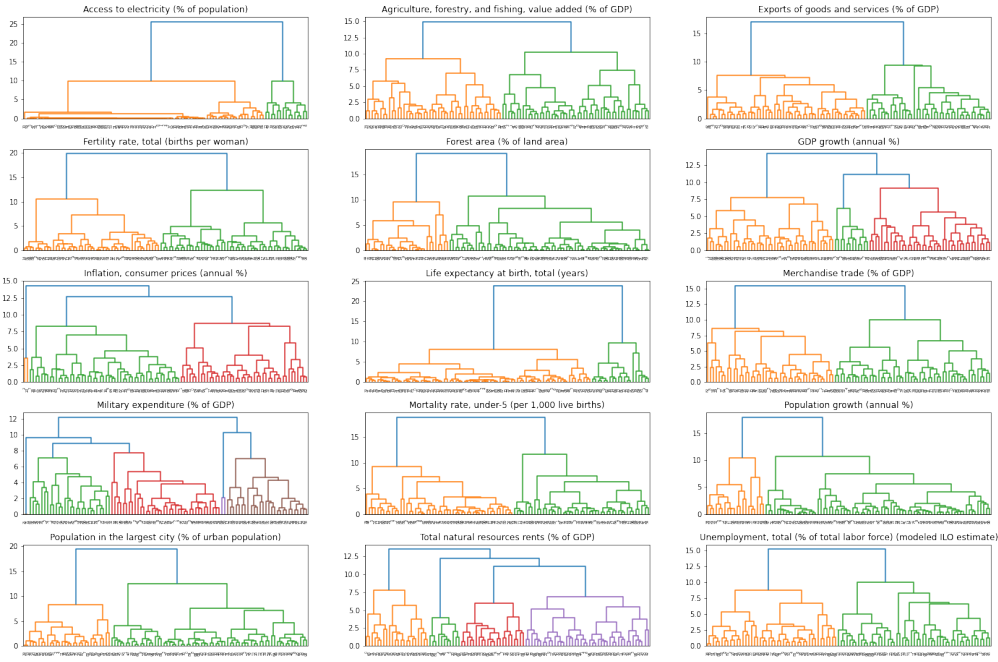


Figure 13: Dendrograms presenting hierarchical relationship between countries by selected indicators, with feature extraction

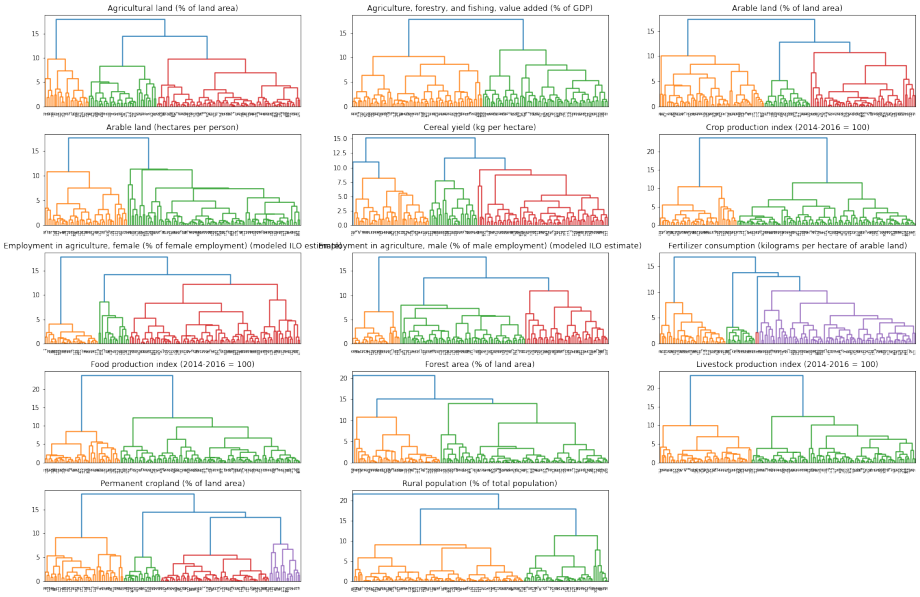


Figure 14: Dendrograms presenting hierarchical relationship between countries by agriculture indicators, with feature extraction

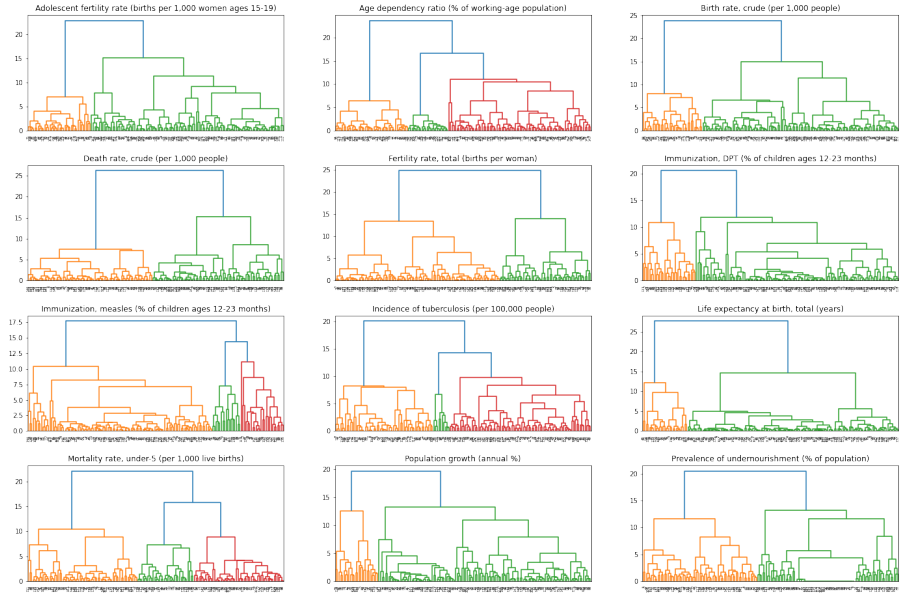


Figure 15: Dendrograms presenting hierarchical relationship between countries by health indicators, with feature extraction

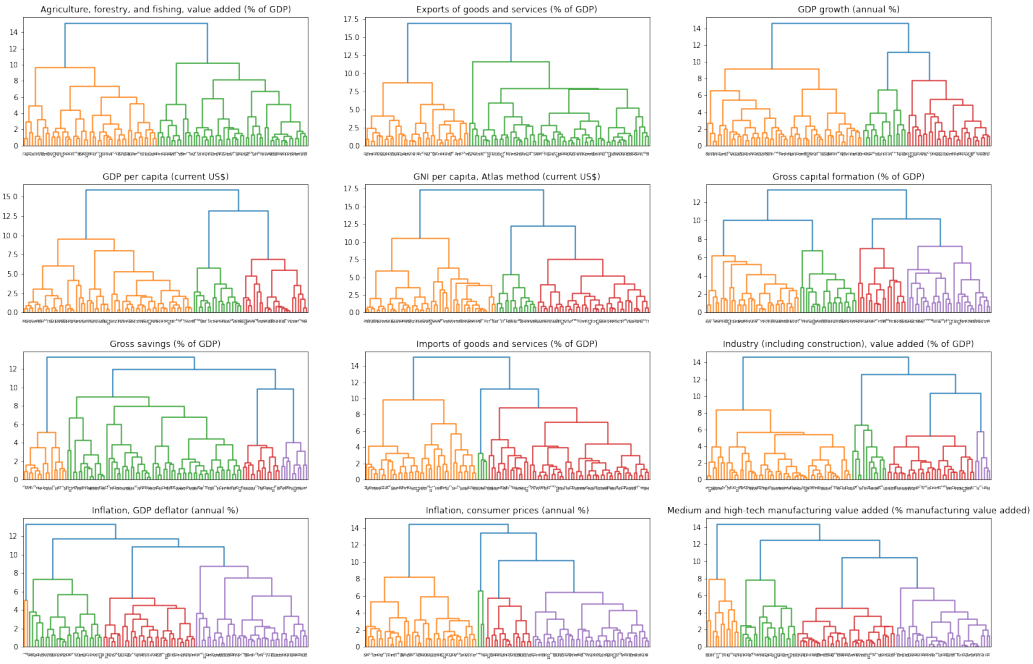


Figure 16: Dendrograms presenting hierarchical relationship between countries by economy indicators, with feature extraction

As diagrams show, results differ from those without feature extraction presented before. Structure of clusters has changed, as well as the number of clusters for some indicators. Clustering by total natural resources rents (% of GDP) is shown in Figure 17. In case of clustering without feature extraction only two clusters were found. By looking at data we can tell that one cluster, marked in red, includes countries with higher total natural resources rents and second cluster, marked in grey, includes countries with lower total natural resources rents, so interpretation is pretty simple. Feature extraction revealed much more complex information about data. In this case, four significant clusters were found. We can assume that cluster marked in brown includes countries in which the value of this indicator was very low (below 2.5% of GDP), compared to other countries, in the analyzed period of time. In cluster marked in grey we can find countries in which value of this indicator decreased in the period from 2000 to 2018. Third cluster, marked in red, includes countries in which total natural resources rents were unstable over that time. In last cluster, marked in purple there are countries for which the value of the analyzed indicator was very high (over 20% of GDP), compared to other countries in analyzed period of time.

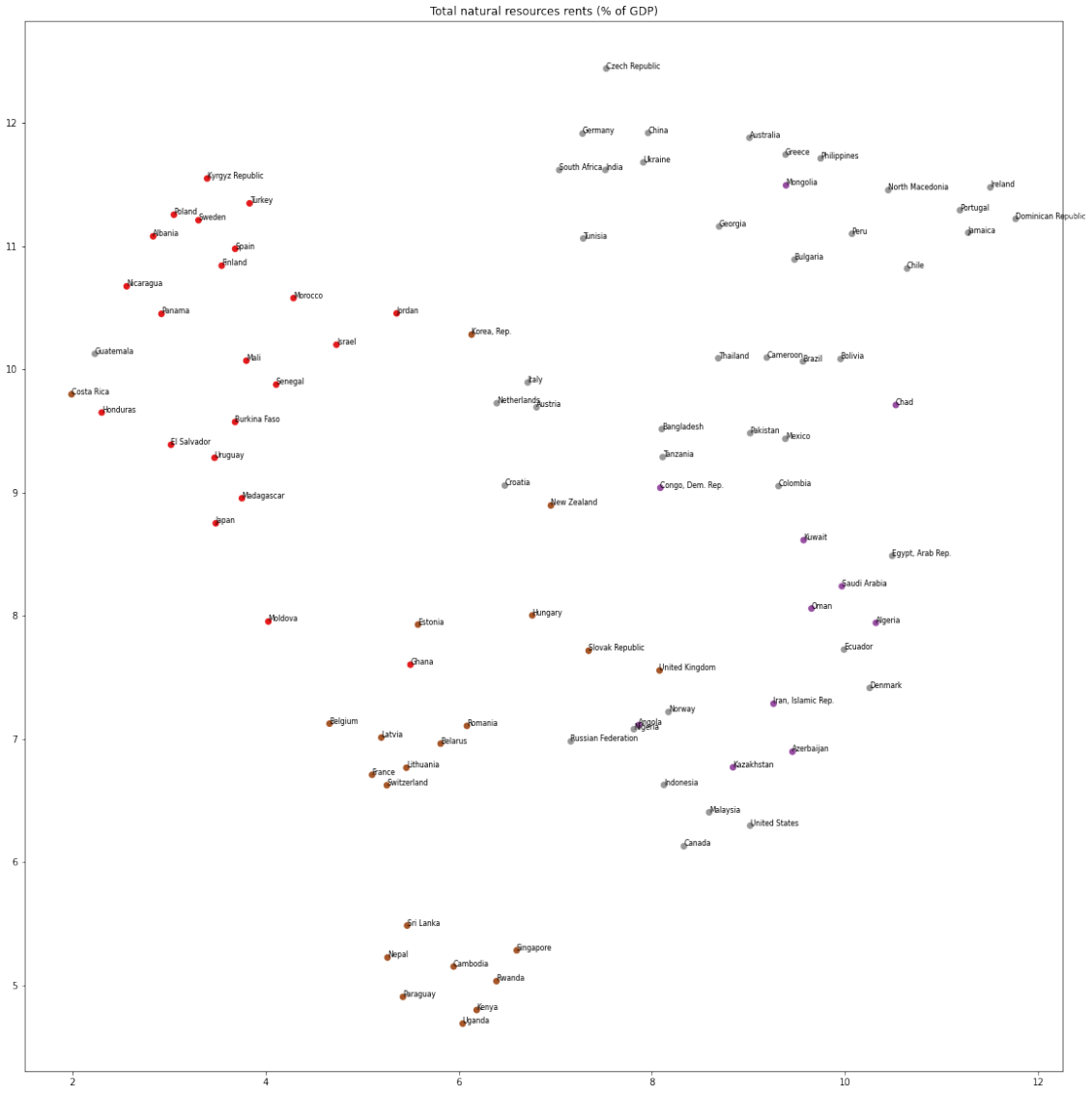


Figure 17: Countries clustered by total natural resources rents (% of GDP) with feature extraction, each color represents a separate cluster

Hierarchical cluster analysis was also carried out for combination of the indicators in four groups: selected, agriculture, health and economy, first without feature extraction, next with feature extraction. The analysis for the combination of all indicators was also performed. For feature extraction `Autoencoder v4` was used. Dendrogram for selected indicators is shown in Figure 18 and clustered data is shown in Figure 19. For these indicators data was divided into two clusters. The first one, marked in red, includes Africa and South Asia countries and the second, grey cluster includes the remaining countries.

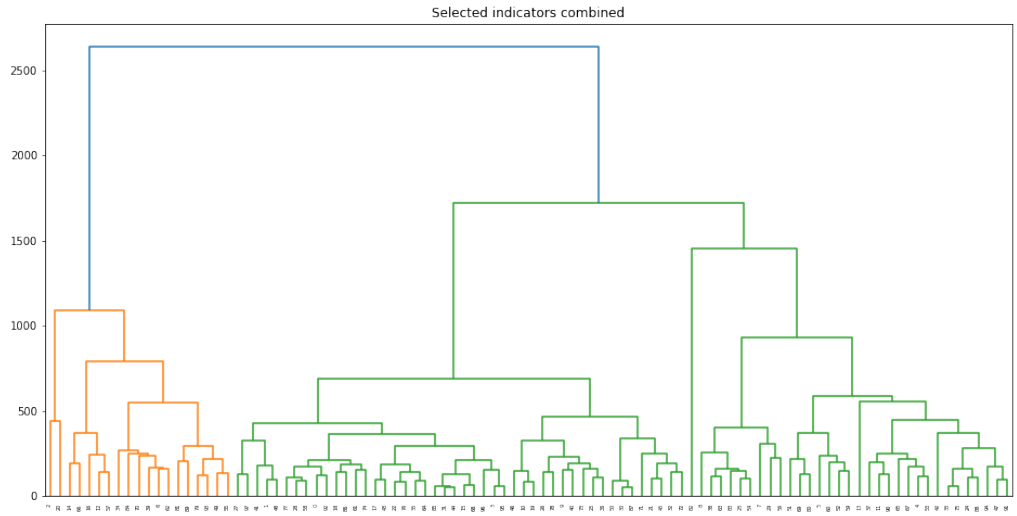


Figure 18: Dendrogram presenting hierarchical relationship between countries by combined selected indicators

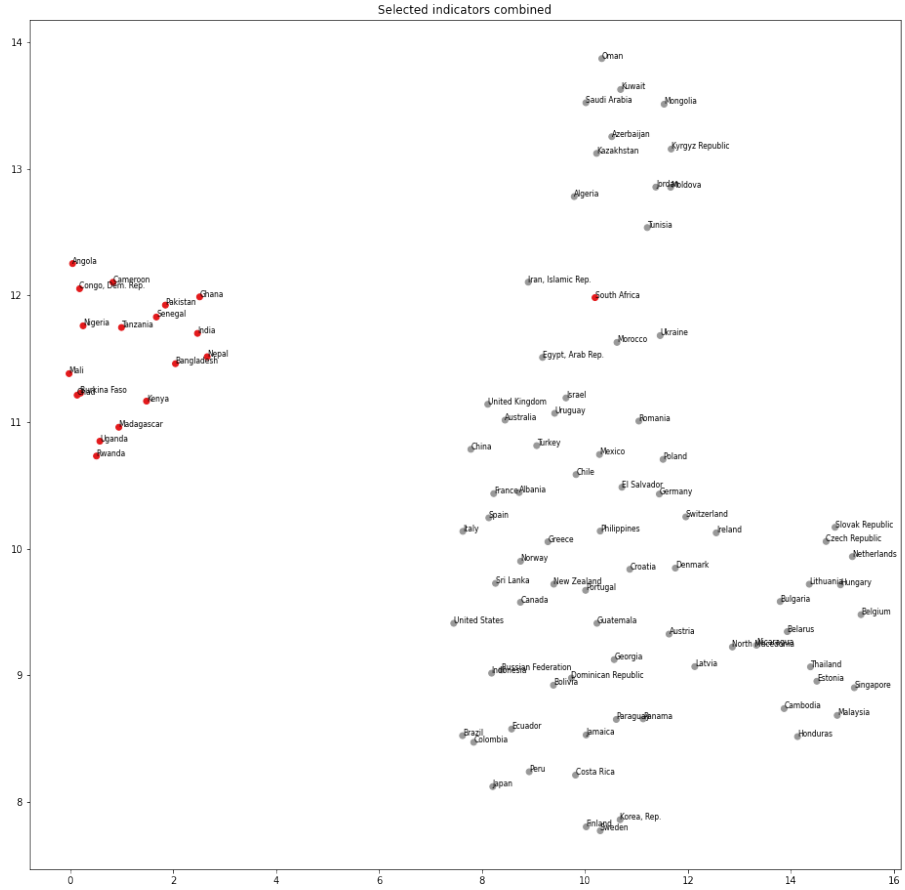


Figure 19: Countries clustered by selected indicators combined, each color represents a separate cluster

In case of clustering by combined selected indicators with feature extraction three clusters were obtained, as we can see in Figure 20. Final result of clustering is presented in Figure 21. First cluster, marked in red, includes European countries and more developed countries from other parts of the world such as: United States, Australia, Japan. The second, grey cluster includes Asia, South America and more developed African countries. Last cluster, marked in orange includes poor countries, most of which are African countries.

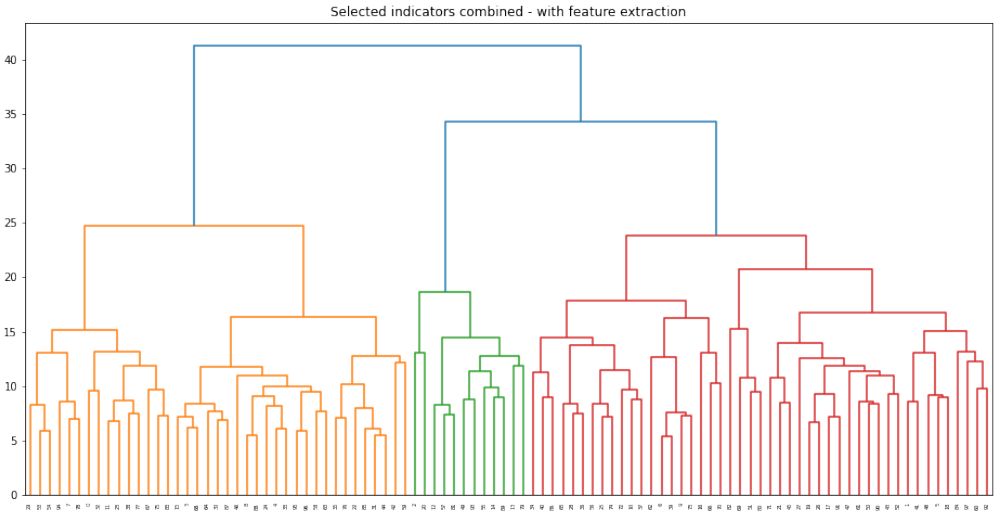


Figure 20: Dendrogram presenting hierarchical relationship between countries by combined selected indicators with feature extraction

Figure 21: Countries clustered by selected indicators combined with feature extraction, each color represents a separate cluster

Clustering countries by combined agriculture indicators allowed for the designation of three clusters, as it is presented in Figure 22. As it can be seen, one of them includes only one country. which is the United Arab Emirates. This country is very different from the rest when it comes to agriculture, because despite the low importance of this sector, it was characterized by very high values of the cereal yield index in the analyzed period. Clustering result is presented in Figure 23. Cluster marked in orange includes countries for which agriculture is mostly not an important sector. Red cluster is the opposite. Single-element cluster, marked in Grey includes the United Arab Emirates.

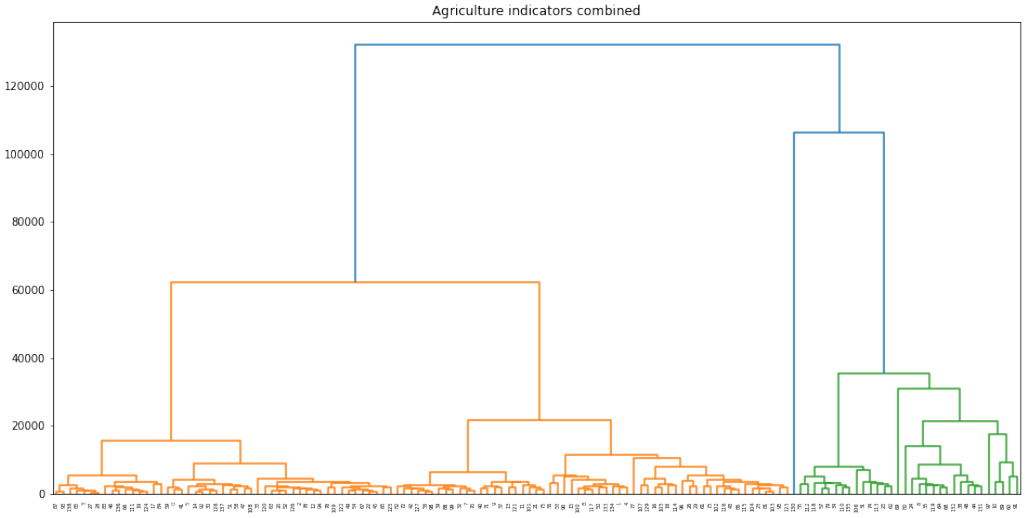


Figure 22: Dendrogram presenting hierarchical relationship between countries by combined agriculture indicators

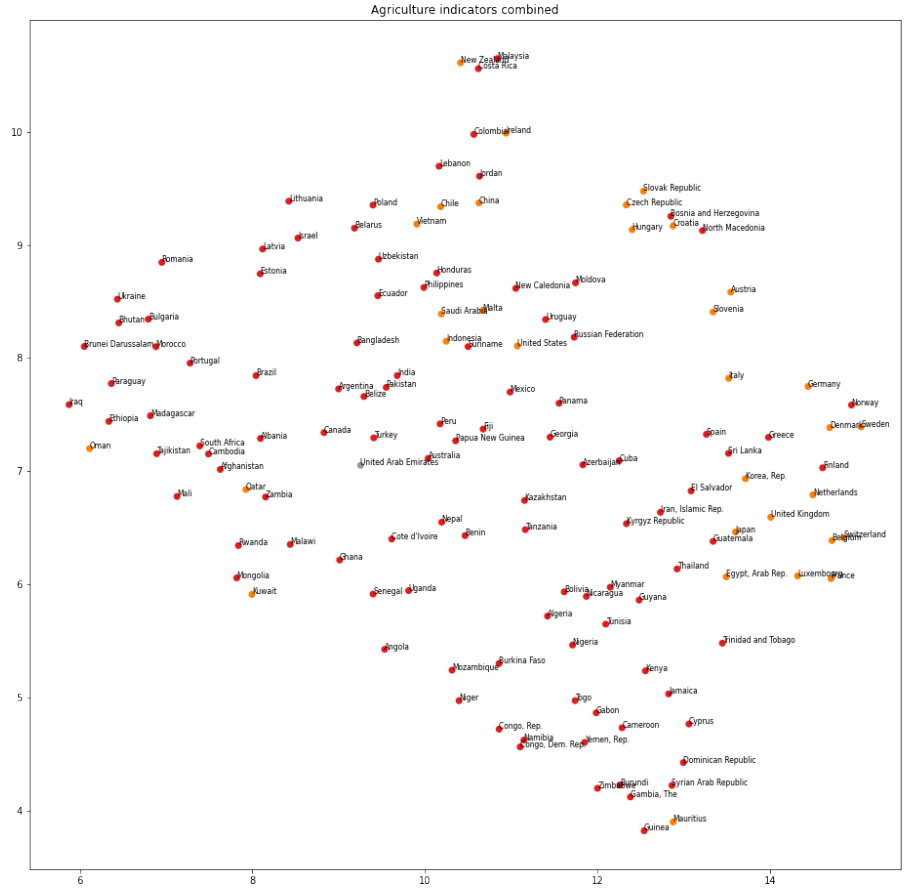


Figure 23: Countries clustered by agriculture indicators combined, each color represents a separate cluster

With feature extraction we obtained only two clusters. Structure of clusters changed, some of countries no longer belonged to the same cluster. Combined agriculture indicators with feature extraction dendrogram is presented in Figure 24 and clustering result in Figure 25.

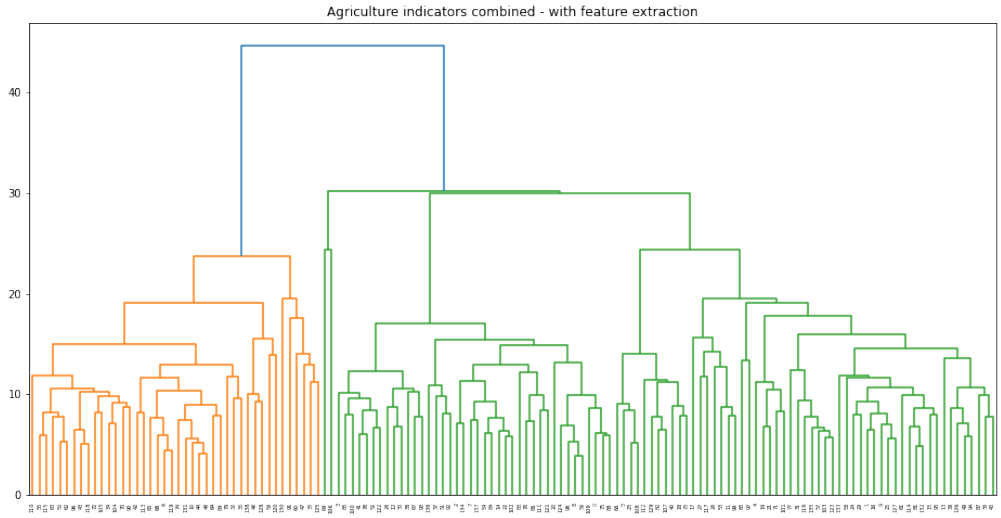


Figure 24: Dendrogram presenting hierarchical relationship between countries by combined agriculture indicators with feature extraction

Health indicators were the next analyzed group of indicators. Without feature extraction three clusters were obtained, as it can be seen in dendrogram (Figure 26). Result of clustering is presented in Figure 27. Cluster having the most elements, marked in red, includes countries with better healthcare. The second, grey cluster contains countries with worse healthcare, higher birth and death rate. The last cluster, marked in orange, is similar to second one, mainly different from it in more cases of tuberculosis.

Figure 25: Countries clustered by agriculture indicators with feature extraction combined, each color represents a separate cluster

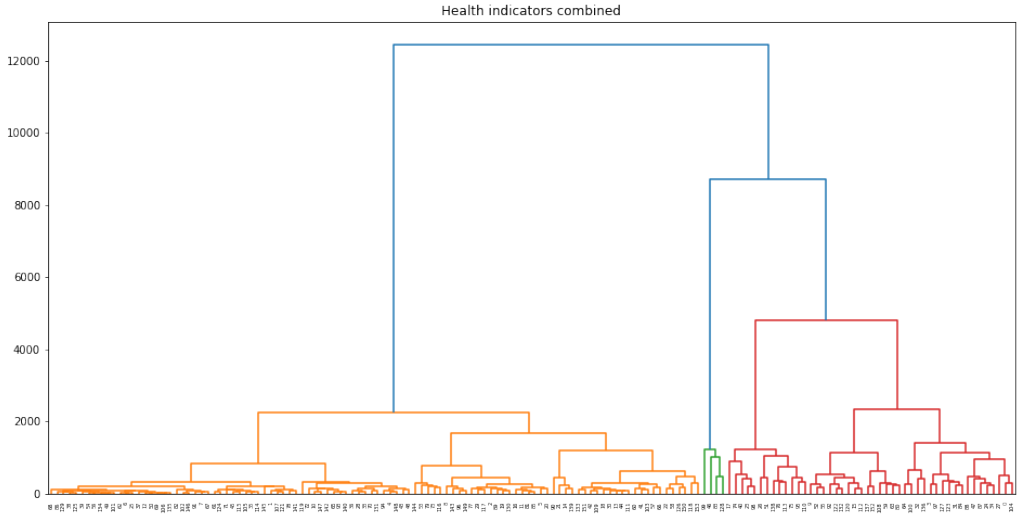


Figure 26: Dendrogram presenting hierarchical relationship between countries by combined health indicators

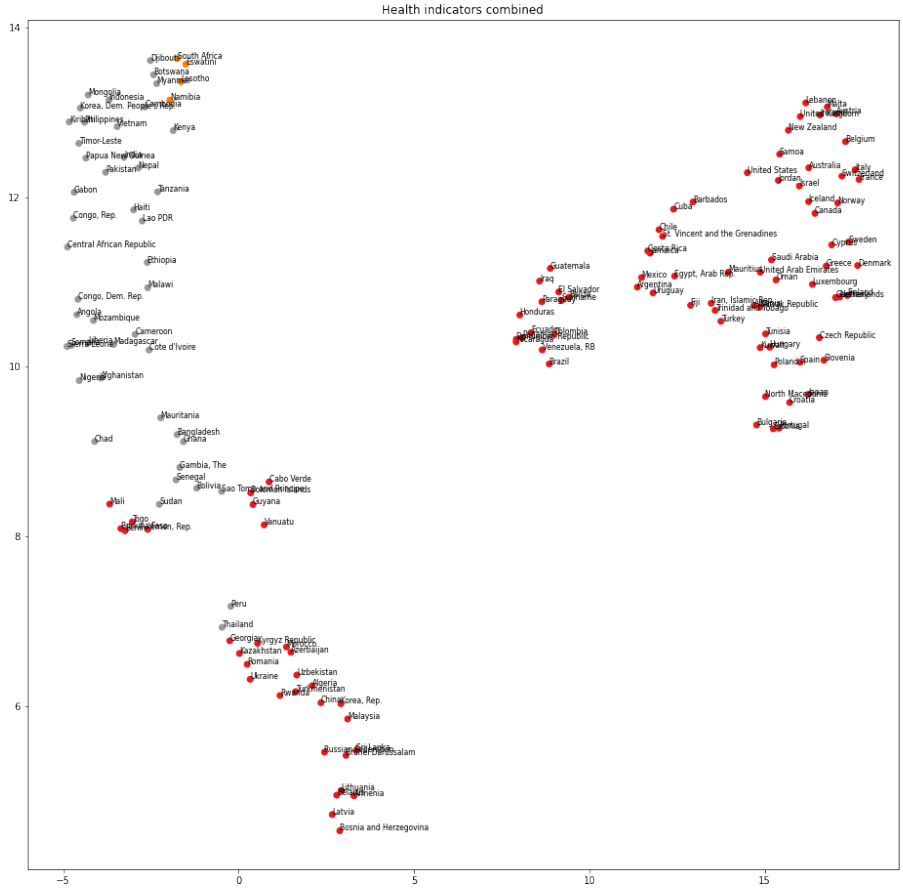


Figure 27: Countries clustered by health indicators combined, each color represents a separate cluster

Analysis of these indicators with feature extraction led to more simple interpretation. In this case only two significant clusters were found, as it is shown in Figure 28. The result of this clustering is presented in Figure 29. We can tell that the first, more numerous cluster, marked in grey, includes countries with better access to health services, higher immunization rate, higher life expectancy and lower birth and death rates. The second cluster, on the other hand, is the opposite of the first one.

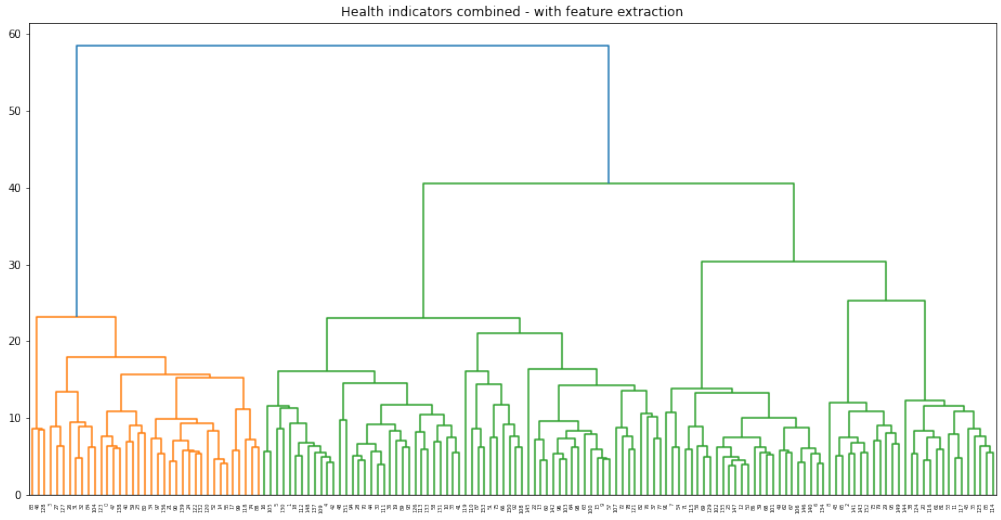


Figure 28: Dendrogram presenting hierarchical relationship between countries by combined health indicators with feature extraction

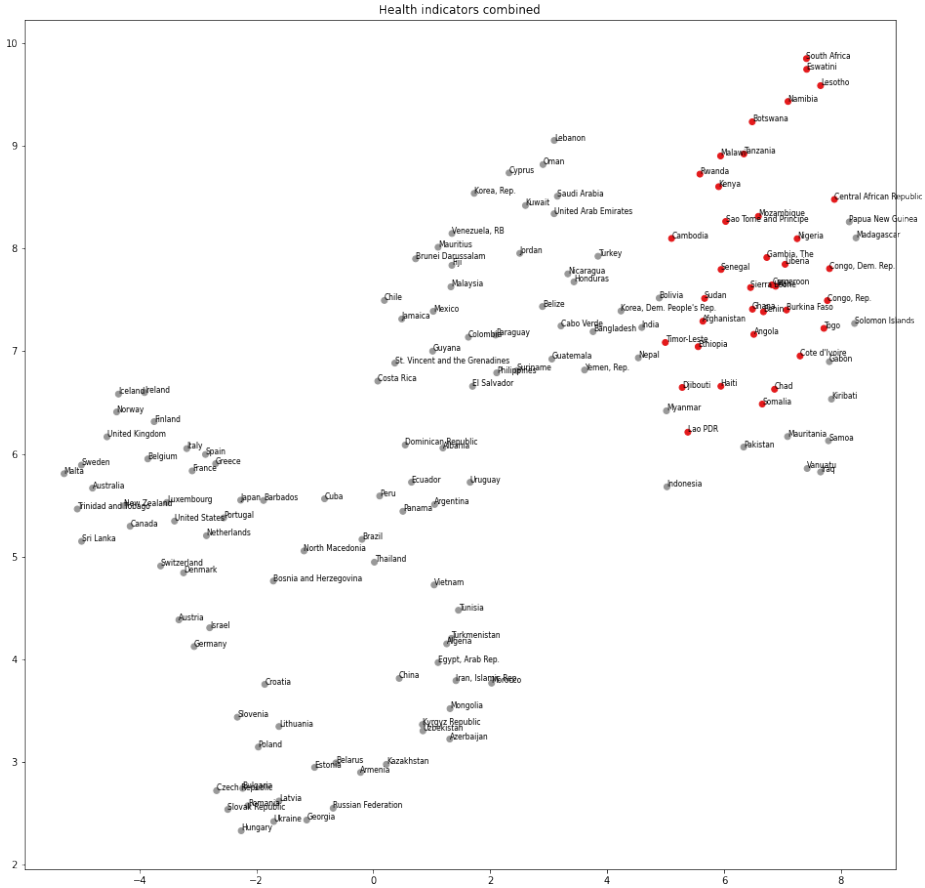


Figure 29: Countries clustered by health indicators with feature extraction combined, each color represents a separate cluster

The last group of indicators that we examined in our analysis were economic indicators. Two clusters were obtained without feature extraction, as Figure 30 presents. Clustered countries are presented in Figure 31. We can tell that smaller cluster, marked in grey, includes the richest countries in the world like China, the United States or Western European countries. The second cluster includes the remaining countries.

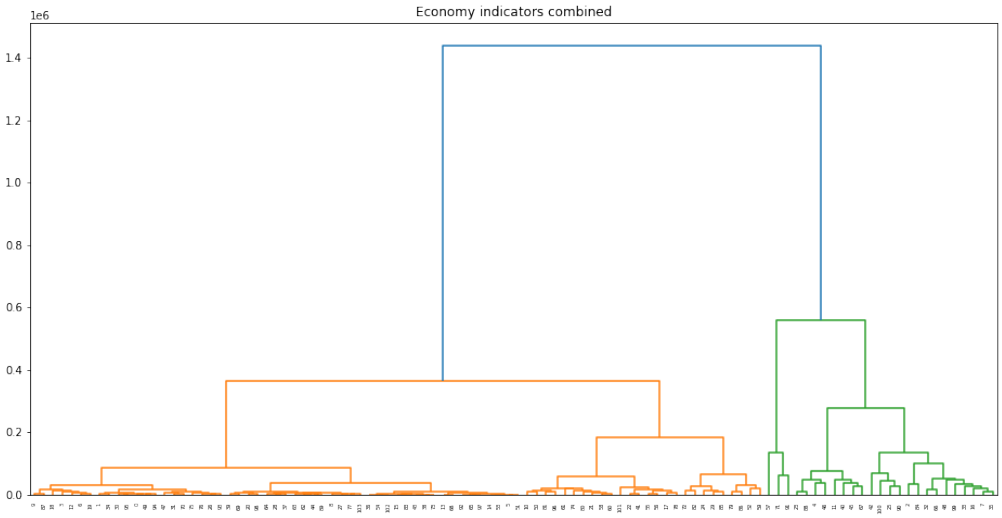


Figure 30: Dendrogram presenting hierarchical relationship between countries by combined economic indicators

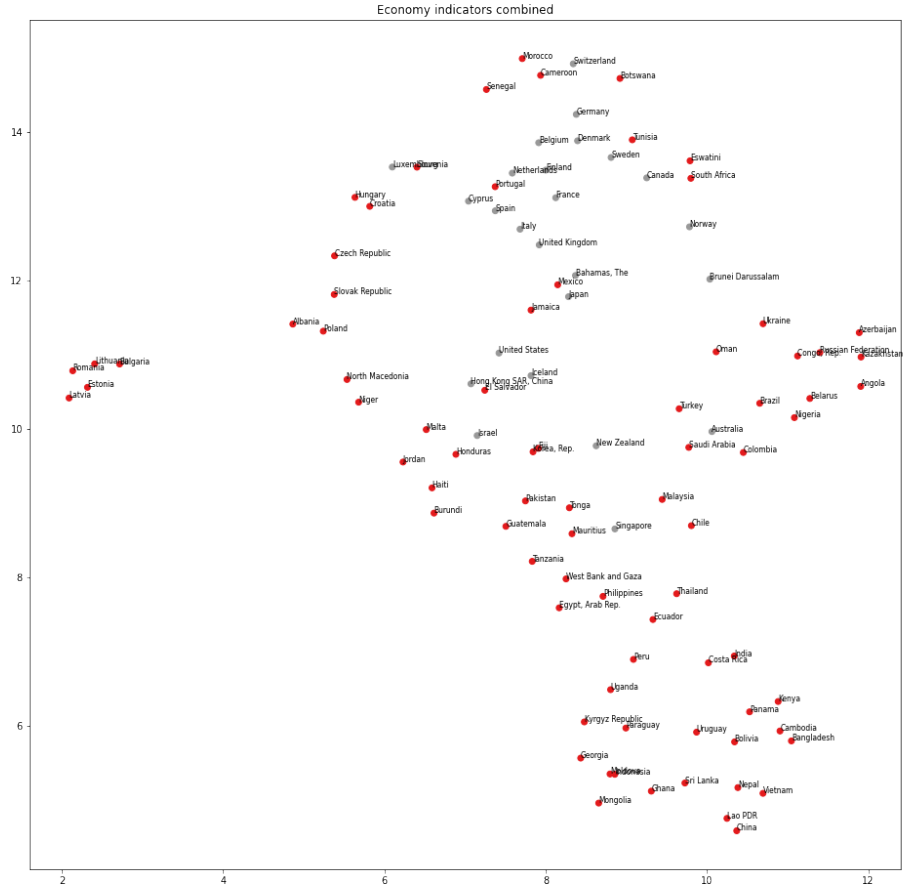


Figure 31: Countries clustered by economic indicators combined, each color represents a separate cluster

As with the previous groups, we then repeated the clustering using feature extraction. In this case three clusters where received, as it is presented in Figure 32. The clusters and the countries belonging to them are presented in Figure 33. In the first, red cluster there are mainly European countries, but also developed countries from other continents. The smallest cluster, marked in orange, includes Russian Federation, Ukraine, West Asia and some Africa countries. The last, gray cluster includes the remaining Asian, African and South American countries.

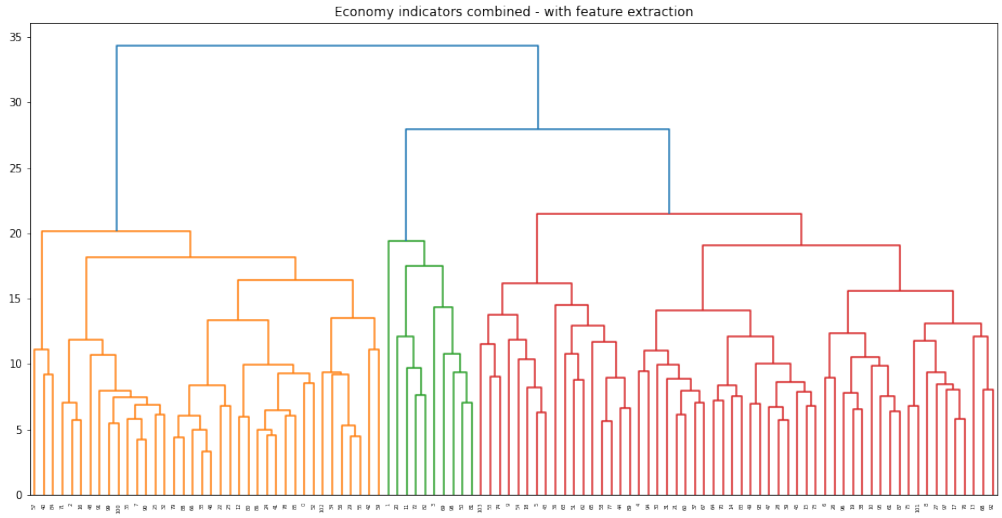


Figure 32: Dendrogram presenting hierarchical relationship between countries by combined economic indicators with feature extraction

Figure 33: Countries clustered by economic indicators with feature extraction combined, each color represents a separate cluster

TODO

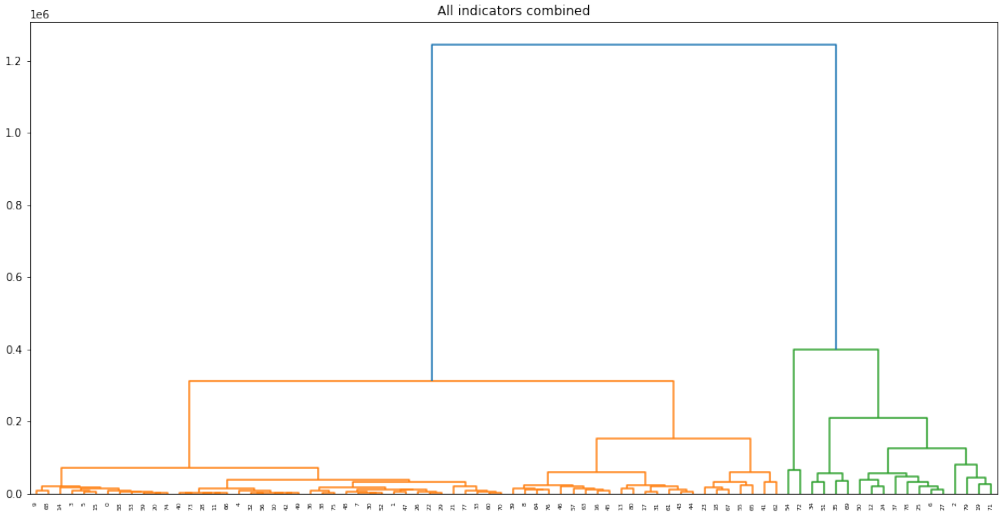


Figure 34: Dendrogram presenting hierarchical relationship between countries by combined all indicators

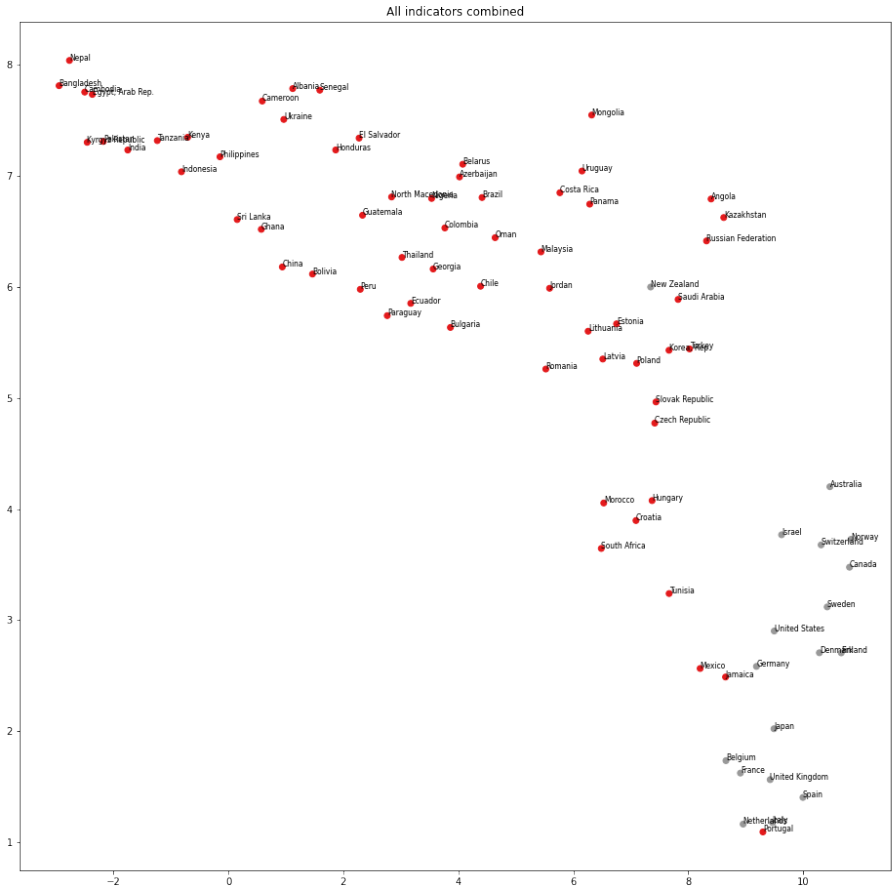


Figure 35: Countries clustered by all indicators combined, each color represents a separate cluster

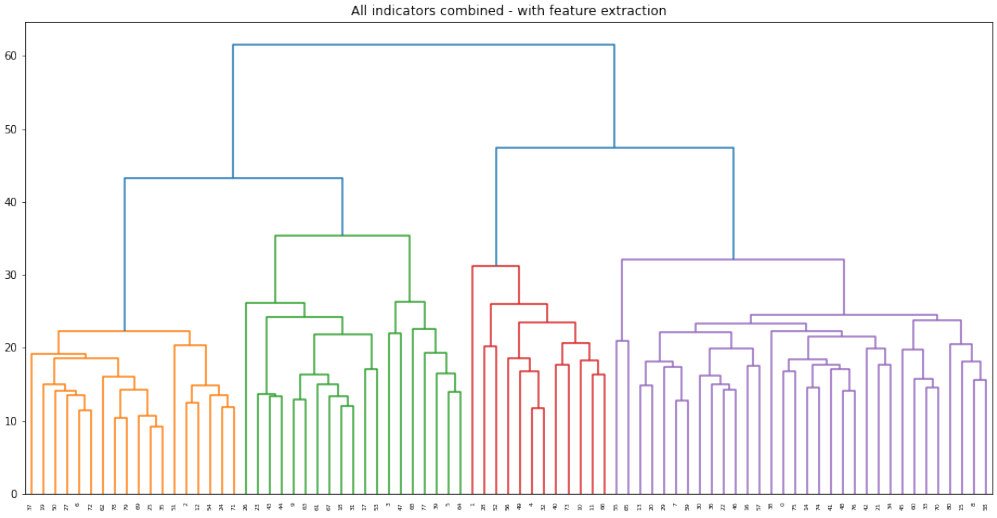


Figure 36: Dendrogram presenting hierarchical relationship between countries by combined all indicators with feature extraction

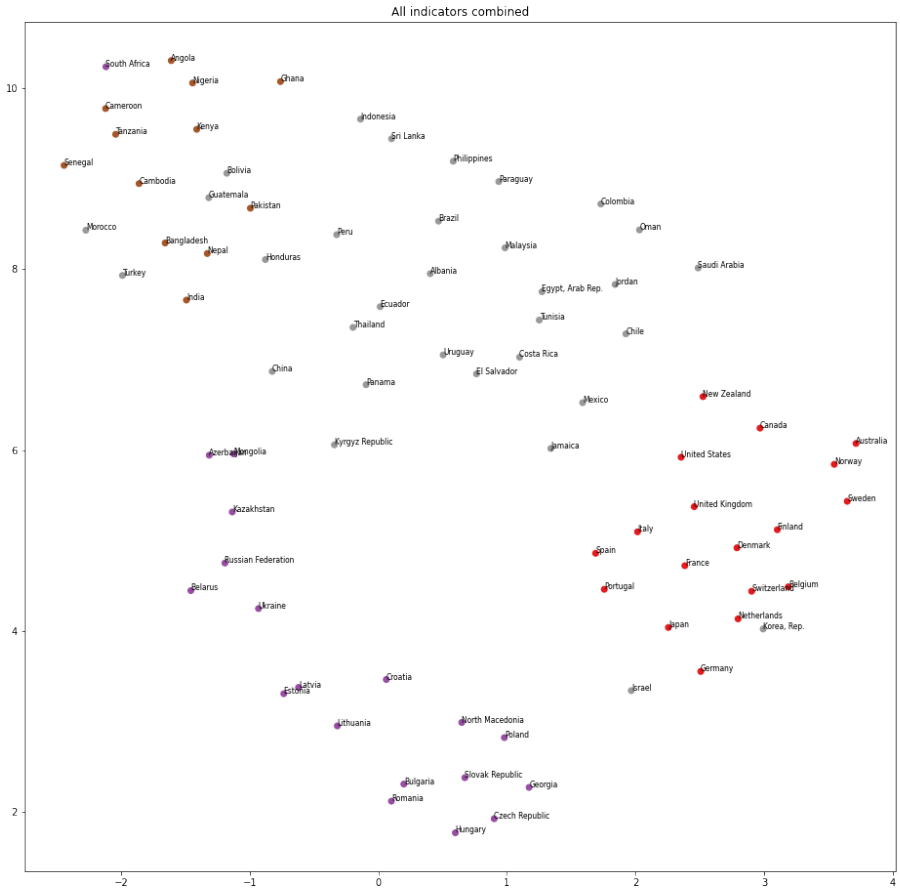


Figure 37: Countries clustered by all indicators with feature extraction combined, each color represents a separate cluster

5 Summary

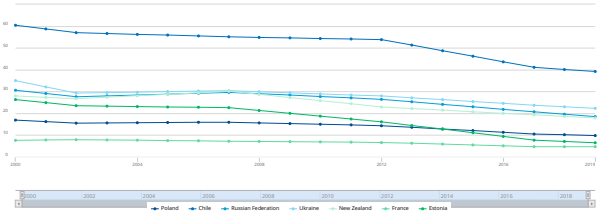
We were able to see many obvious international relations, but also discover some not obvious ones...

Appendix A. Architecture of the autoencoder used for feature extraction

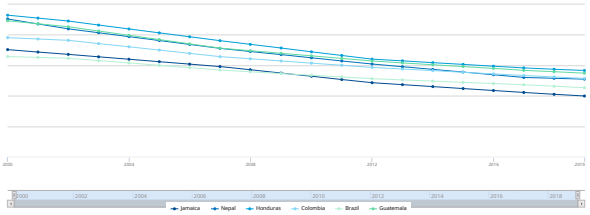
Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 19)]	0
reshape (Reshape)	(None, 19, 1)	0
conv1d (Conv1D)	(None, 19, 32)	128
max_pooling1d (MaxPooling1D)	(None, 10, 32)	0
conv1d_1 (Conv1D)	(None, 10, 64)	6208
max_pooling1d_1 (MaxPooling1D)	(None, 5, 64)	0
conv1d_2 (Conv1D)	(None, 5, 128)	24704
max_pooling1d_2 (MaxPooling1D)	(None, 3, 128)	0
flatten (Flatten)	(None, 384)	0
dense (Dense)	(None, 4)	1540
dense_1 (Dense)	(None, 384)	1920
reshape_1 (Reshape)	(None, 3, 128)	0
conv1d_3 (Conv1D)	(None, 3, 128)	49280
up_sampling1d (UpSampling1D)	(None, 6, 128)	0
conv1d_4 (Conv1D)	(None, 6, 64)	24640
up_sampling1d_1 (UpSampling1D)	(None, 12, 64)	0
conv1d_5 (Conv1D)	(None, 12, 32)	6176
up_sampling1d_2 (UpSampling1D)	(None, 24, 32)	0
flatten_1 (Flatten)	(None, 768)	0
dense_2 (Dense)	(None, 19)	14611

Total params: 129,207
Trainable params: 129,207
Non-trainable params: 0

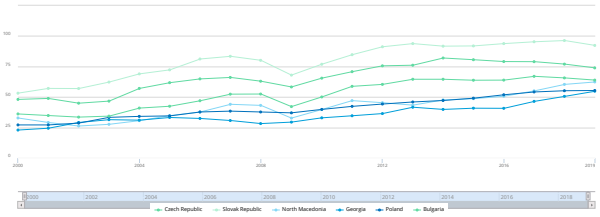
Appendix B. Time series previewed in on the World Bank Open Data website on which we base the analysis in Section 3



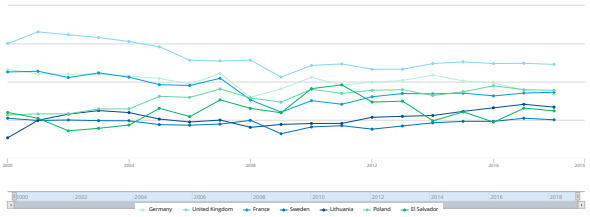
Series : Adolescent fertility rate (births par 1,000 women ages 15-19)
Source: World Development Indicators
Created on: 06/04/2022



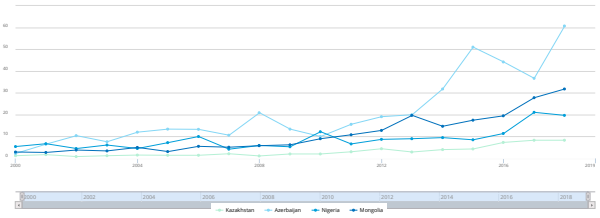
Series : Adolescent fertility rate (births per 1,000 women ages 15-19)
Source: World Development Indicators
Created on: 06/04/2022



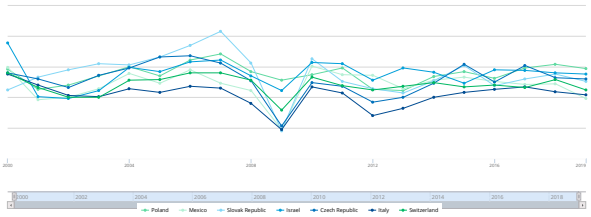
Series : Exports of goods and services (% of GDP)
Source: World Development Indicators
Created on: 06/04/2022



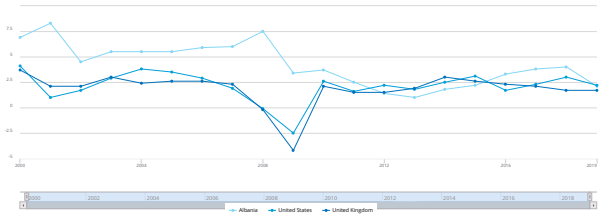
Series : Fertilizer consumption (kilograms per hectare of arable land)
Source: World Development Indicators
Created on: 06/04/2022



Series : Fertilizer consumption (kilograms per hectare of arable land)
Source: World Development Indicators
Created on: 06/04/2022



Series : GDP growth (annual %)
Source: World Development Indicators
Created on: 06/04/2022



Series : GDP growth (annual %)
Source: World Development Indicators
Created on: 06/04/2022

References

[1] *The project repository – autoencoder experiments.* URL: https://github.com/maciektr/worldbank_data_exploration/blob/main/notebooks/feature_extraction/autoencoding_experiments.ipynb.

[2] *The project repository – training the autoencoders.* URL: https://github.com/maciektr/worldbank_data_exploration/blob/main/notebooks/feature_extraction/train_autoencoders.ipynb.

[3] *The project repository – single time series types analysis.* URL: https://github.com/maciektr/worldbank_data_exploration/blob/main/notebooks/feature_extraction/single_types_feature_extraction.ipynb.

[4] *The project repository – time series groups analysis.* URL: https://github.com/maciektr/worldbank_data_exploration/blob/main/notebooks/feature_extraction/groups_feature_extraction.ipynb.