**Clustering**

SV42/2020 Danilo Babić, SV61/2020 Jelena Miković

# Problem

The goal is to predict the continent of country based on attributes.

# Data

The training data set consisted of 219 countries and their data.

# Pre-processing

Each row had 7 attributes, but we deleted ‘Year’ because it wasn’t relevant. We added missing data using KNNImputer with 20 neighbours. Also we used StandardScaler for all of numerical data.

# GMM

GaussianMixture is a probabilistic model that represents a mixture of multiple Gaussian distributions. Each Gaussian distribution in the mixture has its own mean and covariance, and the entire dataset is modeled as being generated from these distributions with certain probabilities. The algorithm iteratively estimates the parameters of these Gaussian distributions and the probabilities that each data point belongs to each distribution.

**Using GridSearch we tried over 720 combinations of parametars and got these as best**:

* **n\_components**: The number of mixture components (i.e., Gaussian distributions) to fit. In this case, 27 means the model will fit **27** different Gaussian distributions to the data.
* **covariance\_type**: Specifies the type of covariance parameters to use. The options are:
  + 'full': Each component has its own general covariance matrix.
  + 'tied': All components share the same covariance matrix.
  + 'diag': Each component has its own diagonal covariance matrix.
  + 'spherical': Each component has its own single variance. Here, **'diag'** indicates that each Gaussian distribution has its own diagonal covariance matrix.
* **init\_params**: Determines the method used to initialize the parameters of the Gaussian distributions. Options are:
  + 'kmeans': Use k-means clustering to initialize.
  + 'random': Randomly initialize the parameters. Here, **'random'** indicates that the parameters are initialized randomly.
* **max\_iter**: The maximum number of iterations for the Expectation-Maximization (EM) algorithm. Here, it is set to **50**.
* **n\_init**: The number of initializations to perform. The best results are kept. Here, it is set to **1**, meaning the algorithm will run only once with a single initialization.
* **random\_state**: The seed used by the random number generator. Setting this parameter ensures reproducibility. Here, it is set to **42**.
* **reg\_covar**: A regularization term added to the diagonal of the covariance matrices to ensure they are positive definite. This helps prevent numerical issues during the computation. Here, it is set to **0.01.**
* **tol**: The convergence threshold. The EM algorithm will stop if the log-likelihood of the model does not improve by at least this amount between iterations. Here, it is set to **0.01.**
* **warm\_start**: If set to True, the solution of the last fitting is used as the initialization for the next call to fit. This can speed up convergence when fitting the model incrementally. Here, it is set to **False**.
* **weights\_init**: The initial weights for the mixture components. If None, they are initialized using the method specified in init\_params. Here, it is set to **None.**

# V Measure Score and Results Analysis

Results on training data were 0.4012, and on testing 0.5217.

V-measure score is a metric used to evaluate the quality of clustering results when the true class labels are known. It combines measures of homogeneity (whether clusters contain only data points from a single class) and completeness (whether all data points of a class are assigned to the same cluster) into a single score. The score ranges between 0.0 (worst) and 1.0 (best), with higher values indicating better clustering quality.

# References

* <https://scikit-learn.org/stable/modules/generated/sklearn.mixture.GaussianMixture.html>
* <https://scikit-learn.org/stable/modules/generated/sklearn.metrics.v_measure_score.html#v-measure-score>

Note: ChatGPT helped format some of the sentances above, we chose everything we wanted to say, he just made it sound better. ☺