Work of EM2

```
creation_date: 2022-12-07 07:43
modification_date: Wednesday 7th December 2022 07:43:17
alias:
```

Dataset

- [x] ~~I presented the Time used per person per geography, life satisfaction per year per geography, life satisfaction per year per geography related to GPD ✅ 2022-12-07~~
- [ ] Shopping data to one store per customer

Questions:

- [ ] What can we learn when compared to more happy places? (TRY to see what factors/PC most affect happiness) How does life satisfaction vary per GDP?
- Time use, Happiness vs GDP
- [ ] What are the countries that are more similar in terms of time spent? (PCA)?
- Time use
- [ ] What are the countries that are more similar in terms of time spent? (Factor Analysis (FA))?
- Time use
- [ ] How does happiness evolve for Portugal over time? what what are the countries that most changed?
- Life satisfaction vs Geography

Tools:

- Overleaf
- Discord
- Google Slides (Maybe?)

Structure Overleaf

1. Introduction / Motivation for all this questions
2. Methodology (Brief introduction of every method that we will use)
   1. Clustering
   2. PCA
   3. Factor Analysis (FA)
3. Dataset (Presentation of the multiple datasets that we have)
4. Work Questions

5. Results
    1. Question 1
    2. Question 2
    3. Question 3
    4. Question 4
6. Conclusions
7. References
8. Contributions

How to divide the work?

> Suggested - 1 question per user, respective subsection of Results and 1 subsection of Methodology or Introduction + (Work Question)

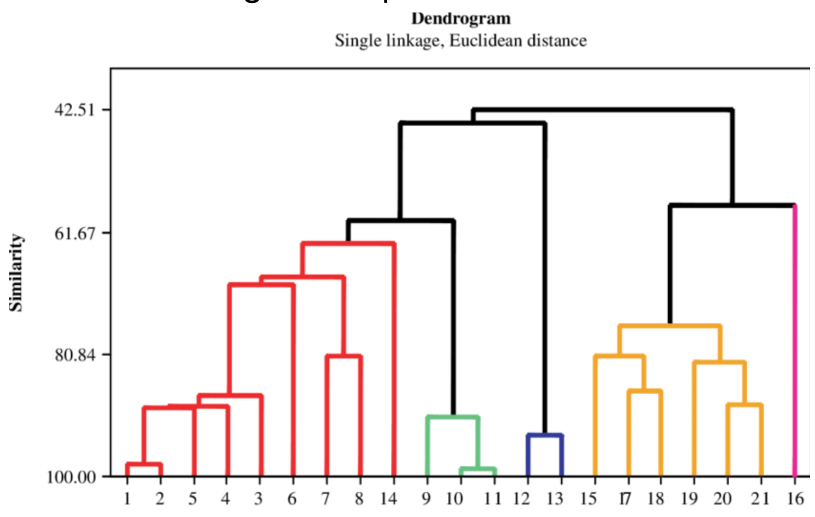
- Fred - PCA, 2nd question
- João - Factor Analysis (FA) , 3rd question
- Luís - 1 question, methodology regarding Clustering
- Tijan - 4 question, Introduction/Motivation , datasets explanation

2nd Meet between us for when?

> Suggested Day 19 or 20 Dec 2022, where it is suppose to deliver all the past work

Clustering

Clustering of the individuals, or features that share the same profile. Dendrograms like the following one help to elucidate the idea of Clustering.



Cluster for Natural Language Processing recommend the Cosine Distance. K nearest neighbours is also common approach. Mini batch also works, in fact it is faster but only supports euclidean distances.

The recommended approach to deciding the number of clusters involves usually the elbow technique, even though the silhouette_score is a better one, but also more computational intensive.

It also allows us to create a silhouette diagram, here we can see knifes shapes for the clusters.

This is deeply related to k-means.

In images we can do segmentation:

- color segmentation - similar color get the same segment
- semantic segmentation - all pixels of one object receive the same segment
- instance segmentation - all pixels of the same type of object receive the same segment

Can also be used in semi-supervised learning, for example

> **label propagation**
>
> It is a form of Clustering in semi-supervised learning where we first create the clusters and then we label the clusters that we have created.

.

Principal Component Analysis (PCA)

Dimensionality Reduction using the variables with the most variability of the data, all the components are now written as a linear combination of other components.

$$CP_1 = f(X_1, X_2, \ldots, X_p)$$

$$Var(PC_1 > Var(PC_2)$$

$$PC_i \perp PC_j(i.e. Cor(PC_i, PC_j)) = 0$$

It allows us to reduce the dimension, to only the variables that have the most information about our data (the ones that have the highest variance. For example, some variables might have a very high weight or very low on the 1PC

$$PC_1 = 0.9X_1 + 0.9X_2 + 0.001X_3 + 0.002X_4$$

This centers the data:

```
prcomp(x, center=TRUE, scale=False, ...) # This centers the data
summary(prcomp(...)) # This gives more information abou the prcomp
```

In Python to reach the same result we use a singular value decomposition

- The % explained Variance = var(PC_1)/total variance *100%*
- Total variance =
- $$\text{standart deviation}_1^2 + \text{standart deviation}_2^2 + \text{standart deviation}_3^2$$
- Using the example from class=

$$\frac{5.95^2}{5.95^2 + 1.76^2 + \ldots + 0.81^2}$$

- The rotation matrix give the coefficients for every PC
- Trace is the sum of the diagonal elements, this is the total variance if the B is the Var(X)

$$Tr(B) = \lambda_1 + \lambda_2 + \ldots + \lambda_p$$

Loading

Correlation between the PCA and the original values

$$cor(Y_j, X_k) = \frac{a_{kj}}{\sqrt{Var(X_k)}} \sqrt{\lambda_j}$$

The work presentation would be postponed one 1 week

Regression vs PCA

On the Regression we want a line that minimizes the error, on PCA we want a line that maximizes the Variance.

Normalisation yes or no?

By default we we should start by using the un-normalised because it is easier to interpret, yet we should normalize when:

- The var are a different order of magnitude
- When the Variability of the original data is very high This is equivalent to work with Correlation matrix

Unstanble

If you perturb the training set slightly and run the PCA again the unit vector might change or even rotate.

```python
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import RandomizedSearchCV
from sklearn.pipeline import make_pipeline

clf = make_pipeline(PCA(random_state=42),
                    RandomForestClassifier(random_state=42))
param_distrib = {
    "pca__n_components": np.arange(10, 80),
    "randomforestclassifier__n_estimators": np.arange(50, 500)
}
rnd_search = RandomizedSearchCV(clf, param_distrib, n_iter=10, cv=3,
                                random_state=42)
rnd_search.fit(X_train[:1000], y_train[:1000])
```

Factor Analysis (FA)

We find latent (hidden) variables/factores to explain our results. We assume that our variables are the result of hidden variables

$$X_1 = f(F_1, F_2, \dots, F_q)$$

Where q is inferior to the dimensionality of the population.

Example:

The quality of the Air in the forest, for a non allergic person this might be "good" for an allergic it might be "could be better".