Analysis Pre and Post COVID-19 Pandemic Rorschach Test Data of Using EM Algorithms and GMM Models

Valerio Ponzi[1], Samuele Russo[2], Agata Wajda[3], Rafał Brociek[4] and Christian Napoli[1,5]

[1]Department of Computer, Control and Management Engineering, Sapienza University of Rome, Via Ariosto 25, Roma, 00185, Italy
[2]Department of Psychology, Sapienza University of Rome, Via dei Marsi 78, Roma, 00185, Italy
[3]Institute of Energy and Fuel Processing Technology, Zabrze, 41-803, Poland
[4]Department of Mathematics Applications and Methods for Artificial Intelligence, Faculty of Applied Mathematics, Silesian University of Technology, Gliwice, 44-100, Poland
[5]Institute for Systems Analysis and Computer Science, Italian National Research Council, Via dei Taurini 19, Roma, 00185, Italy

Abstract

The global spread of the COVID-19 virus has become one of the greatest challenges that humanity has faced in recent years. The unprecedented circumstances of forced isolation and uncertainty that it has imposed on us continue to impact our mental well-being, whether or not we have been directly affected by the virus. Over a period of nearly three years (2017-2020), data was collected from multiple administrations of the Rorschach test, one of the most renowned and extensively studied psychological tests. This study involved the clustering of data, collected through the RAP3 software, to analyze the distinctive trends in data recorded before and after the pandemic. This was achieved through the implementation of the well-established machine learning algorithm, Expectation-Maximization. The proposed solution effectively identifies the key variables that significantly influence the subject's score and provides a reliable solution. Additionally, the solution offers an intuitive visualization that can assist psychologists in accurately interpreting shifts in trends and response distributions within a large amount of data in the two periods.

Keywords

Rorschach test, Gaussian Mixture Model (GMM), Expectation Maximization (EM), Principal component analysis (PCA),

1. Introduction

Since the beginning of 2020, humanity had to battle against an invisible enemy: the COVID-19 virus. As well known nowadays, this virus can spread very easily and this has forced humans to dramatically change their common and usual behaviors: shaking hands, hugs, and kisses allow a fast transmission of this virus and they are now strictly forbidden unless you are sure that the person in front of you is not infected. This is not so easy to detect because this virus is mutated many times since March 2020 and many subjects may be infected without showing symptoms.

Regarding the symptoms that an infected subject could have, the most common ones are high fever, cough, sore throat, loss of smell and taste, and general tiredness. Some of these symptoms allow this virus to appear as simple flu and that's another reason why this virus is dangerous and hard to detect sometimes. In some fragile subjects (elders, and people with other serious illnesses), this virus may cause death. With the development of the vaccine, we have been able to restrict and occasionally weaken the virus's capacity to spread, with the hope that everyone will soon be able to resume their regular social activities in safety.

Since the scientific world didn't fully understand the COVID-19 virus at the time of its initial discovery and, more importantly, since individuals didn't know how to behave in the first place, staying at home was one of the hardest problems it presented to us. These factors compelled governments all over the world to relocate lockdowns in cities and states. For instance, the government of Italy has imposed two significant lockdowns, the first lasting from March 2020 to June 2020 and the second lasting from October 2020 to January 2021. These lockdowns may have had a large and dramatic impact on people's personalities and conceptions of reality, in addition to the disease, the spreading of uncertainty, and terror. The culmination of these unlucky occurrences may also have an impact on certain brain processes.

Despite all the negative events that occurred during this brief but intense historic moment, psychologists are now studying this topic in order to determine whether the virus can genuinely permanently alter brain function in addition to any physical harm that it may do. The Rorschach test is one of the most effective and sophisticated methods psychologists employ to examine mentality and personality. Hermann Rorschach, a Swiss psychologist who lived between the end of the 19th and the beginning of the 20th century, inspired the creation

SYSYEM 2022: 8th Scholar's Yearly Symposium of Technology, Engineering and Mathematics, Brunek, July 23, 2022
ponzi@diag.uniroma1.it (V. Ponzi); samuele.russo@uniroma1.it (S. Russo); awajda@ichpw.pl (A. Wajda); Rafal.Brociek@polsl.pl (R. Brociek); cnapoli@diag.uniroma1.it (C. Napoli)

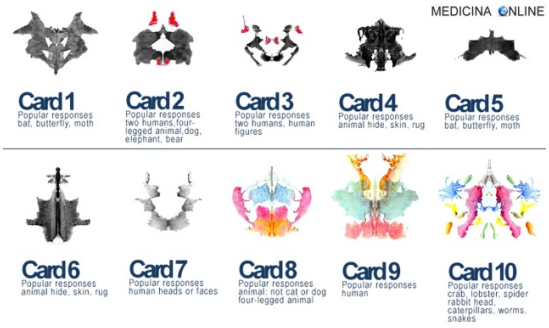

Figure 1: Cards used in the Rorschach test. Notice the symmetry of the inkblot for each card.

of this test and gave it his name. This test requires the participant to bring 10 cards (or tables) to their attention, each of which has a symmetric inkblot. The cards are shown in Fig. 1, where it can be seen that there are 5 monochromatic, 2 two-tone, and 3 colorized versions of them. One by one, these cards are given to the subject, who is then asked to comment on every aspect the card represents from its perspective. It's crucial to note that the subject may take any amount of time to respond and that there are no "correct" or "wrong" answers because the responses are all subjective. Also, notice there can be more than one response for each card: the psychiatrist/psychologist encourages the subject to give many original replies. Typically, the psychologist notes every answer the subject gives and it is related to a specific part of the inkblot. It's important to add that there is no specific submission order for the cards.

Many things we can say about this test, but this little jump into the psychology world is enough to understand the utility of this test for the task. In fact, a big amount of data was collected in a dataset called 'COVID-19 Rorschach test dataset' which contains several samples of protocols and responses, from 2017-01-01 to 2020-09-15, available online for free. It includes some demographics-related variables and the codes of the Comprehensive System (Exner, 2001). The dataset contains more than 500,000 coded responses to the test inkblots stimuli. The series of responses refer to the interpretation given by a certain subject. The data were collected by using the Rorschach Assistant Program (RAP3) software [1], which is one of the currently available examples of a system for online testing [2, 3, 4, 5, 6] and assessment [7, 8, 9, 10, 11, 12].

This dataset contains a significant amount of information for us to analyze. The final interpretation given by a psychiatrist depends on various factors, including the response's content, the visual stimuli, and the region of the inkblot that elicits the response. Our task in this paper is to identify which variables are responsible for eliciting the responses and examine how these values change in the data collected before and during the pandemic. It is worth noting that the dataset does not specify whether the subjects who underwent the test during the pandemic were infected or not.

The main idea is to perform a clustering of the data isolating some important features and sketching and understanding the behavior of the responses based on a probabilistic manner. For this purpose, clustering is a good choice since this is an unsupervised task, and to achieve that the Expectation-Maximization is the algorithm we are looking for. This algorithm will be implemented using a Gaussian Mixture Model (GMM) according to the number of clusters we assume to have, which is one of the results we'd like to achieve with this work.

2. Related works

Since the pandemic came up just a couple of years ago, the state-of-the-art research and literature about this specific task are quite poor. The most popular Rorschach test-related research involved the use of deep learning and neural networks for image classification [13], but this is closely related to inkblots for computer interpretation. As specified before, in this case, we are interested in what kind of feature is the most effective for the responses of the subjects and the difference in trends of the responses between tests done pre-pandemic and during the pandemic.

One of the major problems was (and is) the submission of the Rorschach test to infected people: as the AIP [14] [1] said the remote test submission introduces significant complications in some assessments where the physical presence of the subject was needed (cognitive, neurodevelopment, work stress). However, remote submission was strongly recommended in the other cases (must be done if the patient was currently affected by Covid-19). In detail, psychologists take into account lots of stimuli from a subject: during the remote sessions, they may find some little alterations in the verbal activities of the subject, but the non-verbal stimuli and handling manipulation may be dramatically affected, due to brightness and sharpness of the screen mainly. In-depth studies of the test results have not been published because the specialists need to preserve the privacy of the patients.

Referring to software applications created to help the psychiatrists in the analysis of the signatures, it's worth mentioning the PRALP3 [15] software that was made up by Pancheri, De Fidio and Corfiati from the University of Rome and the university of Bari and published in 1995,

[1] Associazione Italiana di Psicologia (Italian Association of Psychology, in English). Notice that Italy is the country with the highest submissions of the Rorschach test registered in the dataset.

the RIAP5 [16] scoring program that was developed by the PAR company, the RAP3 program cited before and the CHESSSS [17] program published in 2016. From this, it's clear that computer programs already helped industry specialists in a relevant manner.

Some sort of clustering performed on this very same dataset was done by Surekha Ramireddy and published on the Kaggle platform[18]. In this work, the clustering has been performed by the KMeans algorithm [19, 20, 21, 22]. The choice of the KMeans algorithm is easy: it is one of the simplest and most intuitive algorithms in unsupervised ML. Assuming that the distribution of the samples in the dataset is generated by Gaussian distributions represented by a number of K means (priors and covariance matrices are the same for each Gaussian distribution), the algorithm works by starting with a random initialization of the K means, then assign each point to the closest mean (they represent the center of the clusters) by computing the distance between the point and all the means and taking the one with lower distance. Then, it repeats the process updating the means (when a sample is added to a cluster). The algorithm stops when there is no change of cluster between the current and previous step.

Convergence is always reached if the used distance function guarantees the sum of distances decreases from one iteration to the next (when the mean is moving to the center of the centroid).

Some problems might be encountered: the choice of K is fundamental (usually, from 1 to 10), and mostly depend on the choice of the distance function. Also, this algorithm doesn't consider priors and covariance matrix as parameters, so we rely on the computation of the means only and we do not have any information about the covariance matrices of the distributions, namely we cannot control the amplitude of the clusters. Moreover, the KMeans algorithm tends to cluster the data in a circular shape if in 2D (spherical if in 3D) and this may lose accuracy if the dataset is strongly unbalanced.

3. Dataset

Let's take a closer look at the dataset that will be used in this analysis. The dataset comprises 506,480 samples, each consisting of 24 features. The features are listed below for completeness:[18]:

- User: user ID number;
- PQlevel: professional qualifications levels
- Client: client ID number;
- Age: client age in years;
- Gender: client gender;
- Country: client country;
- Protocol: protocol ID number;
- Test Date: the date the RAP3 protocol was created;
- R: total number of responses in the protocol;
- ResponseOrder: the order of responses in the protocol;
- CardID: Rorschach card number, 1 to 10;
- Location: indicated to which area of the blot the responses referred to;
- LocationNumber: location normative number;
- Developmental Quality: quality of processing;
- Determinants: all the visual stimuli in the blot that shaped the reported objects in the response;
- Pair: two identical objects are reported, based on the symmetry of the blot;
- Form Quality: indicates how good is the fitness between the area of blot and the form of the object specified in the response;
- FQText: the form quality associated Normative Text;
- Contents: abbreviations for the category to which the responded object belongs;
- Popular: responses that occur with a frequency with a normative sample;
- ZCode: the relationship between distinct blot areas;
- ZScore: numerical value assigned to responses in which such organization activity occurs;
- Special Scores: indicate the presence of special features in the response;
- Rejection: number of card rejections in the protocol.

Not all of these features are relevant to our task; in fact, the features we will take into account to perform the clustering are the ones stored in the Location, Determinants, and Contents columns. All of these features store information about the test that may be useful for the interpretation of the inkblots provided by the subject. In addition to being unnecessary for this task, the other features can be eliminated because they will add to the payload's computational burden and processing time due to the large number of missing values they contain.

The choice to cluster the data using the Location, Contents, and Determinants features is easily understandable. However, rather than using the categorical Location feature, we opt for the numerical LocationNumber feature. Despite containing some missing values, this feature can still be used as the latent variable for the EM algorithm. Thus, we simplify the dataset by condensing it into three primary features.

Our goal is to cluster the data based on the determinants feature, but due to the limitations of clustering algorithms, it is not meant to consider more than 30 centroids. Therefore, we need to group similar values

together to reduce the number of possible centroids. To do this, we can categorize the values of the determinants into major groups based on their similar meanings:

- Determinants based exclusively on the form feature of the blot;
- Determinants based on the parts of the blot that either seems to reflect or are paired with other parts of the blot.
- Determinants based on movement features of the figure represented by the blot;
- Determinants based on the color features of the blot as the principal cause of response;
- Determinants based on the shading part in both achromatic and chromatic cards;

Our hypothesis is to create a model with five clusters, each corresponding to the categories we identified earlier. To ensure that our data is coherent, we need to take additional steps. One such step is to exclude protocols with less than seven responses, as they are considered non-useful for our purposes. This will result in a slight reduction in the size of our dataset.

Another step that we have to do is to split the values into Contents and Determinants because one sample may be collected in the dataset with more than one determinant and/or content. Moreover, every protocol has many responses associated with it, so we need to split this data in order to be considered as a single response. The splitting operation will increase the shape of the dataset significantly. The complete list of all the values recorded in the Contents column is shown in Table 1. These are still categorical values and we could encode them as one-hot vectors, but a number from 0 to 26 is associated with each content to let the program run on a local machine (0 to A, 1 to H, and so on).

Moreover, it's useful to see what these values represent from the point of view of the specialist. The values refer to some reference classes each answer belongs to. For example: content A refers to animal or animal parts; contents H, Hd, Hh refer to human parts; contents Sx and Sc refer to sexual responses etc.

Now we have a very augmented dataset to work with, but we need to make a couple of steps before constructing the models.

The next step is to split not a single category but the brand-new dataset into two subsets: one containing only data from tests submitted before the pandemic, and the other one containing only data from tests submitted after the pandemic. In the first one, we'll have tests done from Jan 1, 2017, to Feb 29, 2020, while in the second one, we'll have tests submitted from Mar 1, 2020, to Sep 15, 2020. From now on, we will refer to these two subsets as the pre-pandemic dataset for the first one, and the post-pandemic dataset for the second one. It's easy to notice that the pre-pandemic dataset contains way more samples than the post-pandemic dataset because it covers tests done in more than 3 years.

Table 1
Number of values for contents in the splitted dataset

Value	Pre-pandemic	Post pandemic
A	530729	59786
H	429648	46689
Cg	348894	36672
Sc	233454	24655
Hd	184813	18626
Hh	165845	18504
An	159584	16953
Na	144547	16731
(H)	160612	16325
Ad	148635	15936
Art	159168	15862
Hx	144663	15365
Ls	123480	13426
Bt	122926	13264
Fi	106805	11020
Ay	108365	10070
Bl	88084	10000
Sx	100066	9962
Id	87419	8583
Fd	58476	6547
(A)	59012	6434
(Hd)	54297	5814
Ex	34783	4024
Cl	34015	3593
(Ad)	18953	1893
Ge	12717	1301
Xy	14238	1285

The last step of the pre-processing is to deal with missing or null values for the LocationNumber feature. This is done by substituting the missing values with the mean value of the features using the `SimpleImputer` class of the Sci-Kit Learn library for Python (this library will help us for the whole development of this project). Next, we will encode the Determinants features using a 1-out-of-K encoding technique (not the one provided in the library), storing a 4-dimensional array for each determinant with a 1 in position i, where i = 0, 1, 2, or 3, according to the category the determinant belongs to (as shown earlier). Finally, we will perform a scaling operation on the data before fitting the model, transforming it to a standard normally distributed data ranging between [-1,1], with a mean of 0 and a standard deviation of 1.

After completing the pre-processing steps, the pre-pandemic dataset will contain 1,177,056 samples, and the post-pandemic dataset will contain 136,408 samples, with each dataset having 7 features (1 for location numbers, 1 for contents, and 5 for storing 5-dimensional one-hot encoding for the determinants). Since we are dealing

with a large number of samples, we will randomly select 10,000 samples from each dataset to evaluate clustering. However, we can confidently rely on this choice because of the scaling operation conducted just before fitting the model.

4. Implementation

In this paper, the EM[23] algorithm has been chosen to cluster the data. This algorithm provides the computations of the mean, the covariance matrix and the prior of each distribution (cluster) involved in the problem, given a dataset. This choice is reasonable: this algorithm is able to determine the cluster attributes in a smoother way, e.g. by considering the data to be distributed in an elliptical way. In this way, the clusters appear to be smoother than the KMeans clustering and allow us to overcome the drawbacks described before.

4.1. Gaussian Mixture Model

First, we have to make a strong assumption: the samples in the dataset we're dealing with are generated by a Gaussian distribution. Since we are dealing also with K types of clusters, we assume the samples are generated by K different Gaussian distributions. So the probability of having a specific sample x in the dataset is expressed as:

$$P(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x; \mu_k, \Sigma_k)$$

where x is a sample in the dataset $D = x_i$, i=1,...,N, π_k is the prior, μ_k is the mean and Σ_k is the covariance matrix of the k-th distribution.

The model is composed of a combination of Gaussian distributions because all K distributions are handled simultaneously, hence the term Gaussian Mixture Model (GMM).

In this case, a good way to express the data is by introducing the so-called latent variables $z_k \in \{(0, 1)\}$, with z = $(z_1, ..., z_N)$ and each z_k is the 1-out-of-K encoding where only one component is 1 (in the k-th position) and the other K-1 components are zeros. Using this representation, we are assigning each sample to one specific distribution and each sample has a prior probability of being assigned to the k-th distribution equal to

$$P(z_k = 1) = \pi_k$$

thus the probability of having a specific set of latent variables is given by

$$P(z) = \prod_{k=1}^{K} \pi_k^{z_k}$$

so only the k-th prior is selected for any sample.

For a given value of z, we have

$$P(x|z_k) = \mathcal{N}(x; \mu_k, \Sigma_k)$$

thus

$$P(x|z) = \prod_{k=1}^{K} \mathcal{N}(x; \mu_k, \Sigma_k)^{z_k}$$

so now we are able to compute the joint distribution of samples and latent variables as

$$P(x|z_k) = P(x)P(x|z).$$

In this case, the z variables have the 1-out-of-K encoding property and the probability of having the dataset generated by the defined model can be computed by

$$P(x) = \sum_{z} P(x)P(x|z) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x; \mu_k, \Sigma_k)$$

One can notice that the GMM distribution P(x) can be seen as a marginalization of a distribution P(x,z) over the variables z.

Given a dataset of observations D = $\left\{(x_n)_{n=1}^{N}\right\}$, each data point x_n is associated to the corresponding variable z_n which is unknown. The analysis of latent variables allows for a better understanding of input data (e.g., dimensionality reduction).

4.2. Expectation Maximization

The Expectation Maximization (EM) algorithm is an approach for maximum likelihood estimation in the presence of latent variables. It's a general technique for finding maximum likelihood estimators in latent variable models. In detail, given a dataset D = $\left\{(x_n)_{n=1}^{N}\right\}$ and a GMM defined as P(x), the algorithm determines the estimations of the mean μ_k, the covariance matrix Σ_k and the prior π_k.

The EM algorithm is based on the estimation of the maximum likelihood: let's define the posterior probability after observation of x as

$$\gamma(z_k) = P(z_k = 1|x) = \frac{P(z_k = 1)P(x|z_k = 1)}{P(x)}$$

$$\gamma(z_k) = \frac{\pi_k \mathcal{N}(x; \mu_k, \Sigma_k))}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x; \mu_j, \Sigma_j))}$$

thus the maximum likelihood is computed as

$$\underset{\pi,\Sigma,\mu}{\operatorname{argmax}}\, ln P(X|\pi, \Sigma, \mu)$$

where at maximum

$$\pi_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma\left(z_{nk}\right) x_n$$

$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma\left(z_{nk}\right) \left(x_n - \mu_k\right) \left(x_n - \mu_k\right)^T$$

$$\pi_k = \frac{N_k}{N}, N_k = \sum_{n=1}^{N} \gamma\left(z_{nk}\right)$$

for any k = 1,...,K.

The EM algorithm [23] is an iterative approach that cycles between two modes. The first mode attempts to estimate the missing or latent variables called the estimation step (or E-step). The second mode attempts to optimize the parameters of the model to best explain the data, called the maximization step (or M-step). At first, the algorithm takes a random initialization of the parameters as:

$$\pi_k^{(0)}, \mu_k^{(0)}, \Sigma_k^{(0)},$$

4.2.1. E-step

In this step, the model estimates the missing (latent) variables in the dataset. This is performed by the following:

$$\gamma\left(z_{nk}\right)^{(t+1)} = \frac{\pi_k^{(t)} \mathcal{N}(x; \mu_k^{(t)}, \Sigma_k^{(t)}))}{\sum_{j=1}^{K} \pi_j^{(t)} \mathcal{N}(x; \mu_j^{(t)}, \Sigma_j^{(t)}))}$$

4.2.2. M-step

In this step, the model maximizes the parameters of the model in the presence of the data and updates the parameters for further iterations. This is done by the following equations:

$$\pi_k^{(t+1)} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma\left(z_{nk}\right)^{(t+1)} x_n,$$

$$\Sigma_k^{(t+1)} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma\left(z_{nk}\right)^{(t+1)} d_{t+1}(x_n, \mu_k),$$

$$\pi_k^{(t+1)} = \frac{N_k}{N}, N_k = \sum_{n=1}^{N} \gamma\left(z_{nk}\right)^{(t+1)}$$

where

$$d_{t+1}(x_n, \mu_k) = \left(x_n - \mu_k^{(t+1)}\right) \left(x_n - \mu_k^{(t+1)}\right)^T$$

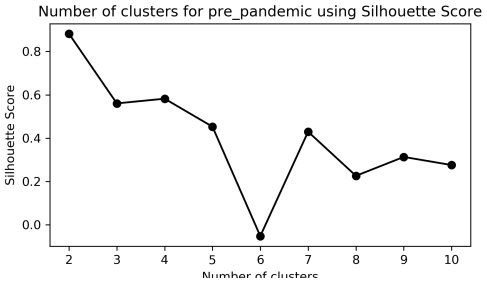

Figure 2: Sketch of the silhouette scores for the pre-pandemic data with the number of clusters for 2 to 10.

4.2.3. Convergence

This algorithm converges to the local maximum likelihood which provides the estimate of the latent variables z_{nk} that are the core of the computations of the parameters. It can be seen as an extended version of K-Means: indeed, if K=1 one can prove that these equations made up the K-Means algorithm itself, and it consists of a probabilistic assignment to a cluster z_{nk}.

4.2.4. Silhouette function

At this point, we need to compute some performance metrics to choose the right number of clusters. For this purpose, we are going to compute the silhouette score[24] which is calculated by taking into account the mean intra-cluster distance a and the mean nearest-cluster distance b for each data point. The silhouette score for a sample is $(b - a)$ / $max\,(a, b)$. The value is explained in the following:

- A silhouette score with a value near 1 means the data point is in the correct cluster;
- A silhouette score with a value near 0 means the data point might belong in some other cluster;
- A silhouette score with a value near -1 means, the data point is in the wrong cluster.

The silhouette score is computed for K = 2, ..., 10, and the number of clusters are chosen considering the highest silhouette score, in general.

5. Results

After going into the numerical results, we would like to underline that we compute the silhouette score to see what is the suggested number of clusters. Remember that we'd like to have k=4 clusters, so the silhouette scores will be used as a guide rather than a hard rule when deciding the number of clusters since the GMM is a probabilistic

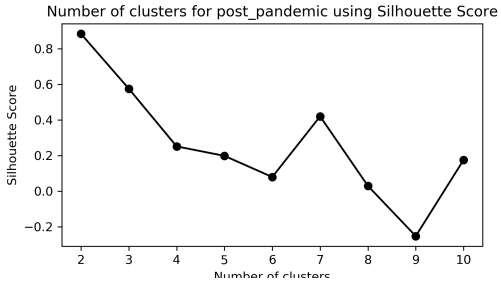

Figure 3: Sketch of the silhouette scores for the post-pandemic data with the number of clusters for 2 to 10.

Table 2
Silhouette scores

k	Pre-pandemic	Post-pandemic
2	0.8818	0.8846
3	0.5604	0.5757
4	0.5821	0.2513
5	0.4529	0.1981
6	-0.0523	0.0791
7	0.4297	0.4202
8	0.2259	0.0291
9	0.3132	-0.2531
10	0.27581	0.1751

model. The scores are sketched in Figure 2 for the pre-pandemic data and in Figure 3 for the post-pandemic data and the numerical values are listed in Table 2. They have been computed by using the `silhouette_score` function provided by the Sci-Kit Learn library [24]. Furthermore, the average Silhouette score for the pre-pandemic GMM model is 0.4077, while the average Silhouette score for the post-pandemic GMM model is 0.2622.

One may notice that the functions do have not similar shape, but the highest Silhouette score for both datasets is related to k=2: this should suggest that the best number of clusters should be 2 for this dataset, but the meaning of the silhouette score is to see that our choice k=5 is reasonable. Despite the poor score for the post-pandemic data, we can keep the initial choice because we are assuming that the distribution of entire data points is similar without any distinction of data submissions. The choice of k=5 is also acceptable because it's bigger than the average Silhouette score (for the pre-pandemic data).

From the plots, we can say the choice of either k=3 or k=4 could be acceptable, too. We may anticipate that the clusters might be distinct but some data points may overlap in space.

In Figure 4 and Figure 5, the data points randomly chosen for clustering are shown, whereas the GMMs are shown in Figure 6 and Figure 7. In particular, Figures 4

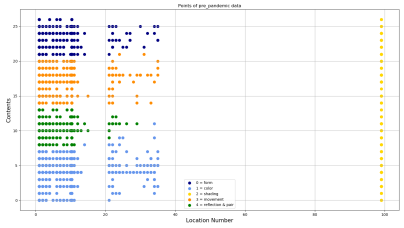

Figure 4: Random pre-pandemic samples, represented as 2D points.

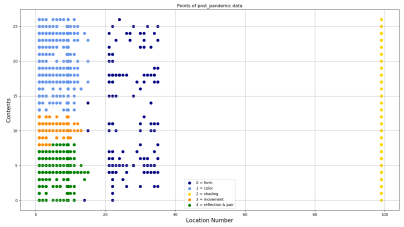

Figure 5: Random post-pandemic samples, represented as 2D points.

and 6 are related to the pre-pandemic data points and distributions, while Figures 5 and 7 are related to the post-pandemic data. Please notice that in both Figure 4 and Figure 5 plots, some points may overlap so you might see fewer points for a cluster.

It's also interesting to see that some points are very far away from the others and this happens for points that have been collected with a location number equal to 99 (the mean of the location numbers is 10.7) that have been predicted as belonging to the shading category (the yellow one). Those points may have an effective rule by enlarging and stretching the Gaussian distributions, especially the one related to the movement category.

We plot the encoded location numbers on the horizontal axis and the encoded contents on the vertical axis. One can assert the following statements for these plots:

- For the pre-pandemic data, the plot is showing a similar distribution of samples predicted as the other 4 categories along the horizontal axis, while it is very diverse and distinct along the vertical axis: it's clear that we don't have many cases in which the distributions share same parts of the space;
- For the post-pandemic data, the plot is showing a very different scenario with respect to the previous one: we see a big increment of samples predicted as belonging to the form category (the dark blue ones), but one can see also see a decrease

in numbers of samples predicted as belonging to the movement category (the orange ones) and the reflection category (the green ones) and the color category (the light blue ones) samples change a little bit their trend;

- For both pre and post-pandemic models, some points are very far away from the others and this happens for points that have been collected with a location number equal to 99 that have been predicted as belonging to the shading category (the yellow ones). In this case, the location number is referring to this category only. We can understand this trend also by seeing at the model plots: indeed, the yellow Gaussian distribution is very thin (namely the minor semi-axis is really close to zero) and far away from the other Gaussian components.

Figure 6: Visualization of the GMM for the pre-pandemic data.

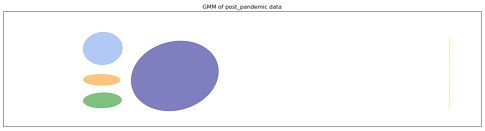

Figure 7: Visualization of the GMM for the post-pandemic data.

So, in general, we notice a significant change in distributions of almost all the categories. Because of that, we want to understand which feature is the most predominant one between the locations and the contents: to do this, we've performed the PCA (Principal Component Analysis)[25] by using the `decomposition.PCA` class of the Sci-Kit Learn library and we have got the following results: `[0.90091513 0.09491116]` for the pre-pandemic model, and `[0.88808373 0.10983465]` for the post-pandemic model, so in both cases, the most dominant component is the location number, and so the location of the inkblot which prompted the patient to respond in that way. One may conclude the location number can be seen as a kind of scaling factor for the data points.

6. Conclusions

The project highlights the significant impact that the numerical value assigned to the location of a region on a blot has on the psychologist's final interpretation. Furthermore, Figures 6 and 7 demonstrate that the results produced by the KMeans algorithm can be quite chaotic. The proximity of the means and the elliptical shapes of the distributions computed by the GMM indicate that the KMeans algorithm may not be able to accurately fit the data. This is likely due to the algorithm's constraint of computing circular-shaped distributions.[26].

Given the richness of the dataset and the subjectivity of the Rorschach test, other projects can be done. An interesting approach would be to cluster the data based on the location or contents of the blot. Another option would be to analyze the data distribution by grouping it according to the cards and determining the most frequent response provided by participants for each card.

Notwithstanding any potential future research, the main aim of this project is to facilitate psychologists in comprehending the variations in Rorschach tests conducted before and after the Covid-19 pandemic. The primary objective is to identify the most significant non-correlated variable that affects the overall subject choice in responses, thereby enabling an accurate interpretation of this phenomenon.

References

[1] J. E. Exner Jr, A rorschach workbook for the comprehensive system, in: Rorschach Workshops, 2001, pp. 171–187.

[2] S. Russo, C. Napoli, A comprehensive solution for psychological treatment and therapeutic path planning based on knowledge base and expertise sharing, volume 2472, 2019, pp. 41–47.

[3] G. Lo Sciuto, S. Russo, C. Napoli, A cloud-based flexible solution for psychometric tests validation, administration and evaluation, volume 2468, CEUR-WS, 2019, pp. 16–21.

[4] S. Pepe, S. Tedeschi, N. Brandizzi, S. Russo, L. Iocchi, C. Napoli, Human attention assessment using a machine learning approach with gan-based data augmentation technique trained using a custom dataset, OBM Neurobiology 6 (2022). doi:10.21926/obm.neurobiol.2204139.

[5] C. Napoli, G. Pappalardo, E. Tramontana, A hybrid neuro-wavelet predictor for qos control and stability, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 8249 LNAI (2013) 527–538. doi:10.1007/978-3-319-03524-6_45.

[6] N. Brandizzi, V. Bianco, G. Castro, S. Russo, A. Wajda, Automatic rgb inference based on facial emotion recognition, in: CEUR Workshop Proceedings, volume 3092, CEUR-WS, 2021, pp. 66–74.

[7] G. Capizzi, C. Napoli, S. Russo, M. Woźniak, Lessening stress and anxiety-related behaviors by means of ai-driven drones for aromatherapy, in: CEUR Workshop Proceedings, volume 2594, CEUR-WS, 2020, pp. 7–12.

[8] V. Marcotrigiano, G. Stingi, S. Fregnan, P. Magarelli, P. Pasquale, S. Russo, G. Orsi, M. Montagna, C. Napoli, C. Napoli, An integrated control plan in primary schools: Results of a field investigation on nutritional and hygienic features in the apulia region (southern italy), Nutrients 13 (2021). doi:10.3390/nu13093006.

[9] N. Brandizzi, S. Russo, R. Brociek, A. Wajda, First studies to apply the theory of mind theory to green and smart mobility by using gaussian area clustering, volume 3118, CEUR-WS, 2021, pp. 71–76.

[10] S. Illari, S. Russo, R. Avanzato, C. Napoli, A cloud-oriented architecture for the remote assessment and follow-up of hospitalized patients, in: CEUR Workshop Proceedings, volume 2694, CEUR-WS, 2020, pp. 29–35.

[11] N. Dat, V. Ponzi, S. Russo, F. Vincelli, Supporting impaired people with a following robotic assistant by means of end-to-end visual target navigation and reinforcement learning approaches, in: CEUR Workshop Proceedings, volume 3118, CEUR-WS, 2021, pp. 51–63.

[12] S. Russo, S. Illari, R. Avanzato, C. Napoli, Reducing the psychological burden of isolated oncological patients by means of decision trees, in: CEUR Workshop Proceedings, volume 2768, CEUR-WS, 2020, pp. 46–53.

[13] A. S. Charles, Interpreting deep learning: The machine learning rorschach test?, stat.ML (2018) 1–4.

[14] G. Alessandri, F. Aschieri, A. Bobbio, R. Daini, A. Lis, M. Nucci, L. Parolin, Documento associazione italiana di psicologia (aip) sulle linee guida per l'assessment ai tempi del coronavirus (2020).

[15] D. De Fidio, P. Pancheri, L. Corfiati, The automated rorschach test: an assessment of the pralp3 three years after publication, Journal of Psychopathology (1998).

[16] J. Exner, I. Weiner, PAR, Riap5: Scoring program, ???? URL: https://www.parinc.com/Products/Pkey/363.

[17] J. M. Smith, E. E. Taylor, Chessss: An innovative rorschach scoring program, Journal of Personality Assessment 98 (2016) 660–662.

[18] S. Ramireddy, Covid-19 rorschach test dataset, 2021. URL: https://www.kaggle.com/surekharamireddy/covid-19-rorschach-test-dataset/data.

[19] K. P. Sinaga, M.-S. Yang, Unsupervised k-means clustering algorithm, IEEE Access 8 (2020) 80716–80727. doi:10.1109/ACCESS.2020.2988796.

[20] G. Capizzi, F. Bonanno, C. Napoli, Hybrid neural networks architectures for soc and voltage prediction of new generation batteries storage, in: 3rd International Conference on Clean Electrical Power: Renewable Energy Resources Impact, ICCEP 2011, 2011, pp. 341–344. doi:10.1109/ICCEP.2011.6036301.

[21] B. Nowak, R. Nowicki, M. Woźniak, C. Napoli, Multi-class nearest neighbour classifier for incomplete data handling, in: Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science), volume 9119, Springer Verlag, 2015, pp. 469–480. doi:10.1007/978-3-319-19324-3_42.

[22] G. Capizzi, F. Bonanno, C. Napoli, A wavelet based prediction of wind and solar energy for long-term simulation of integrated generation systems, in: SPEEDAM 2010 - International Symposium on Power Electronics, Electrical Drives, Automation and Motion, 2010, pp. 586–592. doi:10.1109/SPEEDAM.2010.5542259.

[23] T. Moon, The expectation-maximization algorithm, IEEE Signal Processing Magazine 13 (1996) 47–60. doi:10.1109/79.543975.

[24] K. R. Shahapure, C. Nicholas, Cluster quality analysis using silhouette score, 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA) (2020) 747–748. doi:10.1109/DSAA49011.2020.00096.

[25] F. Kherif, A. Latypova, Chapter 12 - principal component analysis, in: A. Mechelli, S. Vieira (Eds.), Machine Learning, Academic Press, 2020, pp. 209–225. URL: https://www.sciencedirect.com/science/article/pii/B9780128157398000122. doi:https://doi.org/10.1016/B978-0-12-815739-8.00012-2.

[26] Y. G. Jung, M. S. Kang, J. Heo, Clustering performance comparison using k-means and expectation maximization algorithms, Biotechnology & Biotechnological Equipment 28 (2014) S44–S48.