Department of Biomedical Engineering

המחלקה להנדסה ביו-רפואית

Iby and Aladar Fleischman Faculty of Engineering Tel Aviv University

הפקולטה להנדסה ע"ש איבי ואלדר פליישמן אוניברסיטת תל-אביב

Developing Tools to Allow Finding "Sub-Types" of Diseases Using Machine Learning Methods

Final Report

Under-Graduate Final Project

Submitted toward the degree of

Bachelor of Science in Biomedical Engineering

By

Mor Zukin
(I.D. 311508352)

Supervised by Prof. Ran Gilad-Bachrach

August 2020

Table of Contents

1 Abstract

Some diseases may have sub-types that react differently to treatment. Therefore, finding a cure for such diseases requires identifying these subtypes. In addition, different disease sub-types are likely to appear in different ratios across populations.

In this work we are taking a method that has been presented theoretically in previous work and implement it for the purpose of finding sub-types of diseases, using machine learning tools. The aim of the current study is to take advantage of the differences between separate populations, using an additional observed signal which is corelated with the unobserved target variable (sub-type), to better recover the underlying structure of a disease. The performance of the algorithm is first validated using synthetic data, afterwards it is applied on dataset of verified patients for Coronavirus. By dividing the patients into two populations according to their age, a clear division into subgroups of patients was revealed. These results indicate that there may be clusters of people who response differently to the disease that can be identified in early stages of the disease.

2 Introduction

The division of diseases into subtypes can be significant both for diagnostic and for designated treatment. In this work we propose a method for finding sub-types of diseases using machine learning tools. To test the ability of this method, to identify sub-types, it is evaluated on data of Corona patients. This section will provide a brief review on machine learning which will be followed by concise background of the method's underlying assumptions and a clinical data regarding Coronavirus (COVID-19).

2.1 Machine Learning

Machine Learning (ML) is a method of Artificial Intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly specified using statistics and optimizations methods. The primary aim is to allow computers to learn automatically without human intervention and adjust actions accordingly. The process of learning begins with data provided to the computer including observations or examples in order to look for patterns and make better decisions in the future based on the examples that were provided. ML enables analysis of massive quantities of data and

generally delivers faster results compare to humans. The goal of the learning process is to perform as accurate as possible on new unseen examples based on the experience gained during the learning stage. ML algorithms are often categorized as supervised learning or unsupervised learning, the main differences between both lies in the nature of the data used for training as well as the approaches used to deal with it:

a. **Supervised learning –** in this form of learning the algorithm tries to learn a function $f: X \mapsto Y$ by observing input-output pairs (x, y). During the learning phase the algorithm is provided with examples of inputs (features) x and the expected output (label) y. This pairs of examples are drawn from a distribution over the $X \times Y$ space. The goal of the learning algorithm is to find a function f such that given a new pair (x, y) drawn from the same distribution, it will hold that $f(x)$ is close or similar to y. Hence, classification of the data to classes and assign new examples to the right class is a commonly method in use for this type of learning. Some of the known existing algorithms are Random Forest [1], Logistic Regression [2] and Support Vector Machine (SVM) [3].
b. **Un-supervised learning –** in this form of learning the algorithm has to find structures in the input. In this case, the algorithm is provided with a sample from a domain X and the goal is to find a way to partition the domain. Therefore, clustering the input into sub-groups is a common method in this kind of learning. There are many well-known existing algorithms, for example K-means [4] and Expectation – Maximization (EM) [5].

2.1.1 Classification

Classification is one of the most important aspects of supervised learning. Basically, classification is the task of labeling an input sample, identifying to which of a set of categories a new observation belongs, on the basis of a training set of data containing observations whose category membership is known. Usually, the data is divided into three sub-sets:

a. Training set – the largest sub-set, which undergoes pre-processing, contains the correct labels of each observation and it is used to fit the model.
b. Validation set – similar to the training set but is not used to training but for evaluation.
c. Test set – set of new data samples that was hidden from the algorithms and provides evaluation for the final model.

2.1.2 Clustering

Clustering is the most popular technique in unsupervised learning where data is grouped based on the similarity of the data-points. The basic principle behind cluster is the assignment of a given set of observations into subgroups or clusters such that observations present in the same cluster possess a degree of similarity. It is the implementation of the human cognitive ability to discern objects based on their nature.

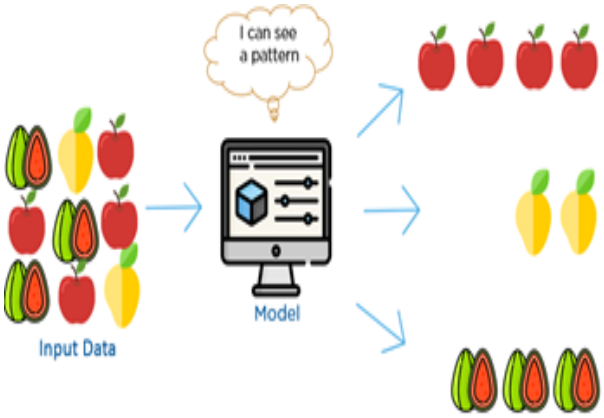


Figure 1: An example for the idea behind clustering. The algorithm gets unlabeled dataset and dived the samples into different groups according to their similarity. [6]

It is a method of unsupervised learning since there is no external label attached to the object, the machine has no access to examples of the expected outcome. The algorithm is able to extract inferences from the characteristics of the data objects and then divide the population into different groups such that each data point is similar to the data-points in the same group. On the basis of similarity and dissimilarity, it then assigns appropriate sub-group to the object. Figure 1 demonstrates this idea.

2.1.3 Mixture Models vs. Topic Models

A Mixture Model (MM) is a probabilistic model for representing the presence of subpopulations within an overall population [7]. Problems associated with MM are used to make statistical inferences about the properties of the subpopulations given only observations on the pooled population, without subpopulation identity information. The MM has been studied extensively [8], it is assumed that there is a single sample (population), that is a single collection of instances, and the goal is to associate instances with their generating distribution or to recover the parameter of the hidden distributions (subpopulations).

On the other hand, Topic modeling (TM) is a type of statistical modeling for discovering the abstract “topics” that occur in a collection of documents [9]. The meaning of “documents” in this context is multiple samples (populations) that are mixtures, with different weights of the underlying topics (distributions) over words. In this study, TM can be defined as an unsupervised technique to discover sub-types (topics) across various

populations (documents). The sub-types abstract in nature, i.e., symptoms (words) which are related to each other form a sub-type. Similarly, there can be multiple topics (sub-types) in an individual document (population).

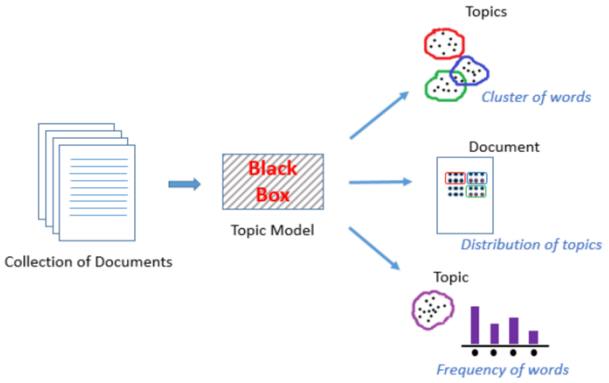


Figure 2: This black box (topic model) forms clusters of similar and related words which are called topics. These topics have a certain distribution in a document, and every topic is defined by the proportion of different words it contains [10].

Comparing these two models shows some similarities and some differences. The goal of both models is to recover information about the generative model. However, there are some key differences, such as the structure of the data. While in MM exists a single sample (population) to learn from, in TM there are some documents (populations) which are a mixture of the topics with different mixture weights. (see [8] for more on that). In this study, I try to close the gap between MM and TM, while solving a TM clustering problem using MM classification tools. If in the classical clustering settings (MM), a sample of instances is divided based on some similarity criteria into groups. In the method applied on this study, I assume that multiple samples are available, similar to TM, and use classification methods (MM) in order to get the underlying structure of groups in the data.

2.2 Bayesian Network

A Bayesian network is a probabilistic graphical model that represents a set of variables and their casual dependencies via graph. These networks are ideal for taking an event that occurred and predicting the likelihood that any one of several possible known causes was the contributing factor. Graphs play an essential role since they provide a vivid representation of the sets of variables that are relevant to each other in any given state of knowledge. The role of graphs in probabilistic and statistical modeling is [11]:

a. to provide convenient means of expressing substantive assumptions
b. to facilitate economical representation of joint probability functions
c. to facilitate efficient inferences from observations

Figure 3 illustrates a simple and typical Bayesian network, it describes relationships among some events, whether rain falls (R), whether the sprinkler is on (O), whether the pavement would get wet (W), and whether the pavement would be slippery (S). We can write the joint probability describes in figure 3 as a product (for further explanation see [11]):

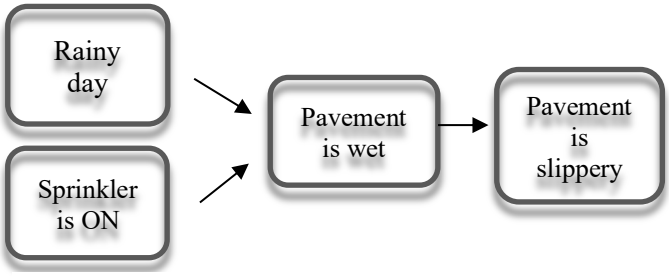


Figure 3: A Bayesian network representing dependencies among four variables.

$$P(S \mid R) = P(R) \cdot P(W \mid R) \cdot P(S \mid W, R) \quad (1)$$

$$P(S \mid O) = P(O) \cdot (W \mid O) \cdot P(S \mid W, O) \quad (2)$$

In the current study, I use Bayesian network to represent the probabilistic relationships between different population, sub-types of diseases and symptoms. Given symptoms and two known population, the network can be used to compute the probabilities of the presence of various sub-types.

2.2.1 Casual Markov Condition

The Markov condition is an assumption that every node in a Bayesian network is conditionally independent of its non-adjacent nodes, given its parents. In other words, it is assumed that a node has no bearing on nodes which do not descend from it. The related Causal Markov (CM) condition states that a node is independent of all variables which are not direct causes or direct effects of that node [12].

If the Bayesian network in Figure 3 describes correctly the real world, then according to Markov Condition, we aim to assert that the slippery pavement is independent of the rainy day and sprinklers once we know that the pavement is wet. This statement is defensible because we can easily translate the assertion into one involving causal relationships: that the influence of rain and sprinkler on slipperiness is mediated by the wetness of the pavement [12]. Now, taking into account this assumption, we can describe the relations in equations (1), (2) as follow:

$$P(S \mid R) = P(R) \cdot P(W \mid R) \cdot P(S \mid W) \quad (3)$$

$$P(S \mid O) = P(O) \cdot (W \mid O) \cdot P(S \mid W) \quad (4)$$

2.3 Coronavirus

An outbreak of 2019 novel coronavirus disease (COVID-19) in Wuhan, China has spread quickly nationwide [13]. For the third time in as many decades, a zoonotic coronavirus has crossed species to infect human populations [14]. As of 22 August 2020, more than 22.9 million cases of COVID-19 have been reported in more than 188 countries and territories, resulting in more than 799,000 deaths; more than 14.7 million people have recovered [15].

The virus is spread primarily via nose and mouth secretions including small droplets produced by coughing, sneezing, and talking. The transmission may also occur through smaller droplets that are able to stay suspended in the air for longer periods of time in enclosed spaces. It is most contagious during the first three days after the onset of symptoms, although spread is possible before symptoms appear, and from people who do not show symptoms [16]. Common symptoms include fever, cough, fatigue, shortness of breath, and loss of sense of smell. Complications may include pneumonia and acute respiratory distress syndrome [17].

Healthcare systems around the globe are being put to the test by Coronavirus. The current pandemic poses a variety of new challenges to all healthcare professionals and facilities. Finding inventive healthcare technologies to relieve the strain on overburdened healthcare systems may has a great importance. Early classification of patients with machine learning methods will significantly increase the effectiveness of tests [18]. In addition, the ability to detect unique features of sub-grouped of patients and understand the effect of the disease on each group, can facilitate the healthcare system and will allow us to utilize the resources for the more vulnerable groups. Clustering a population of patients into groups with common features will allow us to adapt the appropriate treatment to each group that will increase their chances of recovery.

2.3.1 Risk Factors

COVID-19 can affect anyone, and the disease can cause symptoms ranging from mild to very severe. For some other illnesses caused by respiratory viruses, such as influenza, some people may be more likely to have severe illness than others because they have characteristics or medical conditions that increase their risk. These are commonly called “risk factors” [19].

COVID-19 is a new disease and there is limited information regarding risk factors for severe disease. Based on currently available information and clinical expertise, older adults and people with underlying medical conditions are at higher risk for severe illness from COVID-19. People with risk factors may be more likely to need hospitalization or intensive care if they have COVID-19, or they may be more likely to die of the infection. More work is needed to better understand the risk factors for severe illness or complications of Coronavirus. Potential risk factors that have been identified to date include [19]:

- Age
- Some medical conditions
- Use of certain medications
- Certain occupations
- Gender
- Pregnancy

2.4 The Gap

Different disease sub-types are likely to appear in different ratios across populations. For example, if the triggers to the emergence of the different sub-types is influenced by genetics then the different sub-types will appear at different rates in different racial ethnicities. The aim of the current study is to take advantage of the differences between the populations to better recover the underlying structure of a disease. In terms of machine learning, in this project we are using an additional observed signal, for example ethnicity, which is corelated with the unobserved target variable (sub-type).

3 Objectives

The main goal of the current study is to propose a method for finding sub-types of diseases using machine learning tools. Hence the objectives that need to be achieved in order to reach the main goal:

- Creating generic algorithm to identify sub-groups across two populations
- Validation of the performance of the algorithm using synthetic data
- Implementation of the model on real – life data in order to reveal sub-types

A secondary objective is to better understand machine learning and big-data analysis applications, and their potential use in today's vast growing biomedical world.

4 Materials and Methods

In this section, I will bring a description of the underlying assumptions leading me during the project's process, the data I will use in order to analyze the algorithm, and a model structure which will be applied in order to achieve the goals mentioned above. In addition, in the project we are taking a method, called Double Sample Clustering (DSC), that has been presented theoretically in previous work [8] and the main contribution is to test the effectiveness of this method on real data.

4.1 Assumptions and Model Guidelines

In fact, the machine learning problem we are dealing with in this project is closely related to an unsupervised clustering problem since we suspect that there is a division into groups (sub-types), but we don't know these clusters in advanced. One of the guidelines of this project is that if the chances of acquiring a certain sub-disease may be influenced by factors such as age, gender, genetics, behaviors or other measurable properties, we can divide the data into two samples such that we expect to see different proportions of each of the sub-diseases in these two samples. In fact, we get another piece of information about how the data is divided by the different clusters (sub-types).

One of the basic assumptions on which the model is based is that the world behaves in the form of a Bayesian network, similar to the one shown in Figure 3. That is, the relations among populations, sub-types, and symptoms behave in the form shown in Figure 4.

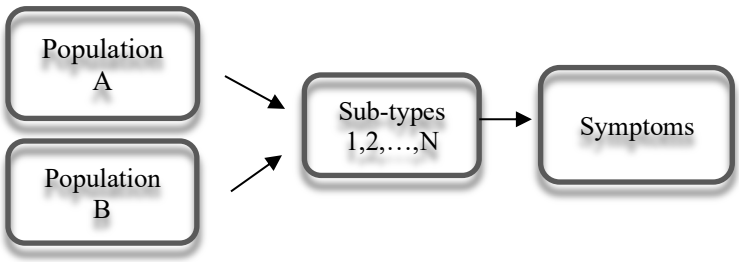


Figure 4: A Bayesian network representing the relationships according to which the world behaves. One of the main assumptions in the current study.

After describing these relations, others main assumptions according to this structure are:

a. Different disease sub-types are likely to appear in different ratios across populations

b. Across populations, the chances of an individual to get a specific sub-type are different

c. We assume that the different sub-types have disjoint supports, meaning the supports of the underlying distribution of each sub-type (cluster) are not overlapping [8]

In addition, if patients in two different populations acquired the same sub-type of a disease, parts of their medical records will be similar [8]. Hence, according to Markov condition discussed above, we can determine that conditioned on the sub-type the population and the symptoms are independent.

The process which will be done in this project is to identify the different clusters by building a tree of classifiers, such that each of the nodes in the tree is a classifier and a leaf represents a cluster. In each classifier, we will use the different symptoms in order to classify each patient as population A or population B, so that at the end of the process we will get clusters of the different groups, or at this case different sub-types, that composed the whole data. The question each classifier deals with is, conditioned symptom X, either the probability of a patient being part of population A is bigger than the probability for population B or smaller:

$$P(A \mid X) >\< P(B \mid X) \tag{5}$$

According to Bayes theorem [20] we can write:

$$\frac{P(X \mid A) \cdot P(A)}{P(X)} >\< \frac{P(X \mid B) \cdot P(B)}{P(X)} \tag{6}$$

Since we want to compare between these phrases, we can multiply both of them by $P(X)$ (which is always bigger than zero). In addition, at each step we make sure that the two populations will have the same weight, therefore $P(A) = P(B)$. We are ending up comparing between:

$$P(X \mid A) >\< P(X \mid B) \tag{7}$$

In accordance with the relationships presented in the equations (1), (2), and following Markov conditions we can determine that:

$$P(X \mid A) = P(A) \cdot P(sub - type_i \mid A) \cdot P(X \mid sub - type_i) \tag{8}$$

$$P(X \mid B) = P(B) \cdot P(sub - type_i \mid B) \cdot P(X \mid sub - type_i) \tag{9}$$

Comparing equations (8), (9) reveals that when population A or population B is given, classifying the symptoms by their probabilities is proportional (with the same proportion coefficient) to classifying the sub-type:

$$P(X \mid A) \propto P(sub - type_i \mid A) \tag{10}$$

$$P(X \mid B) \propto P(sub - type_i \mid B) \tag{11}$$

The outcome of such classification is a division of the data without breaking clusters. (For further mathematical background see [8]).

4.1.1 The Algorithm

In General, the algorithm builds a clustering tree assuming that the clusters are disjoint [8]. Each node in the tree is a classifier trained to separate between two populations (as the labels). At the beginning we take data of patients who having a known disease and create two samples from it, according to prior knowledge about the risk factors of the disease, reweight the patients such that the two samples will have the same cumulative weight [8], and train the classifier. The first classifier becomes the root of the tree and it splits to two sets, as the next step we take all of the patient in each set separately, reweigh them again and train another classifier to separate between the two samples. We keep going in the same fashion until all the patients associated with a leaf are from the same cluster in which case, no classifier can split the cluster any better than random. That is, under the assumption that our generalization error is half we will stop the classifier when our empirical error, calculated according to the classification of the training set, is equal to the generalization error with a predefined deviation. Another base case is the size of the group that the classifier receives. In order to avoid over-fitting, we do not want to use the classifier with too small group. If the classifier receives a sample size smaller than the minimum size defined the classifier will stop and the node will be defined as a leaf of the tree. The number of leaves at the final tree is equal to the number of clusters that composed the data.

Technically, the algorithm is written in Python with the 'Scikit- Learn' which is a library of machine learning for Python. The classifiers is implemented from the library. The model is generically structured so that the minimum number of records per leaf, the deviation from the generalization error, and the classifier itself can be chose according to the researcher's needs.

The classifier used in our model is a linear support vector machine (SVM). SVM is a supervised learning method used for classification and regression problems. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible [21]. Some of the advantages of SVM are [22]:

- Effective in high dimensional spaces
- Still effective in cases where number of dimensions is greater than the number of samples

The exact name of the 'Scikit- Learn' class is 'sklearn.svm.SVC' (Support Vector Classification). The parameters of the classifier that have been defined are described in the 'Data Set' section.

4.2 Data Set

In order to create the model and make sure that the algorithm works as intended, at first, we will use synthetic data which was create by us. This way we can monitor and follow the model steps and see if the results obtained are as expected. In the second part we will implement the model on real-life data.

4.2.1 Synthetic Data Set [1]

The synthetic data was generated from three 2D normal distributions with means at points (-300,0), (100,60) and (0, -200), and unit variance for all the dimensions. In fact, these three distributions represent the three sub-types we aim to reveal. The parameters of the distributions were chosen in order to make sure the supports will not be overlapping.

Two sets of mixing coefficients were selected for the three distributions, each for each population, according to the pre-determined ratios of each distribution in the different populations. The two sets

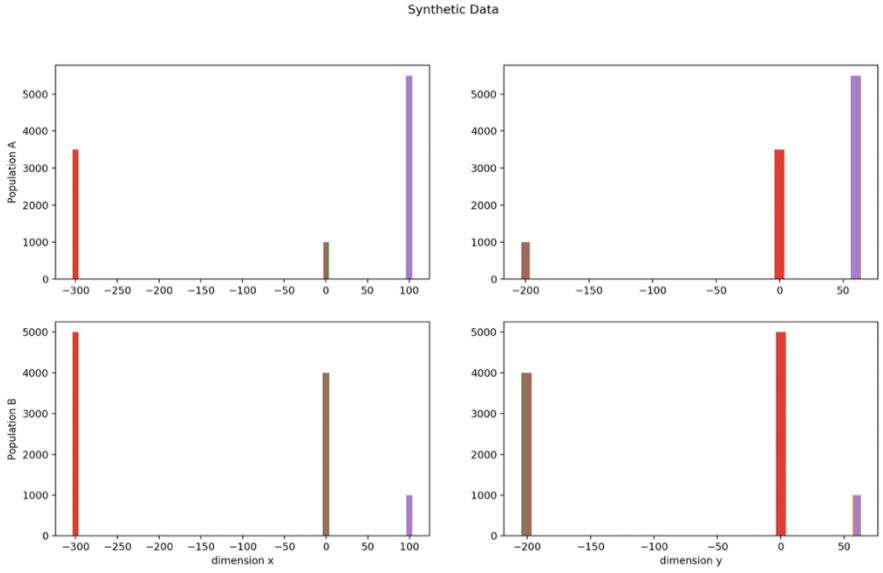


Figure 5: Division of the populations by the three generated distribution. We can see the three means of the distribution (-300,0), (100,60) and (0, -200) and across axis y of each plot we can see the different ratios of each distribution in each population.

[1] Relevant code ('Generate_Synthetic') at the following link - https://github.com/TAU-MLwell/sub-types

were normalized to sum to 1. For population A the ratios are [0.35,0.55, 0.1] and for population B [0.5, 0.1, 0.4], Each population contains overall 10,000 instances. Figure 5 shows the components of each of the populations.

At the end of the day, our aim is to distinguish between the different three "sub-types" (distributions) generated for this data.

First of all as mentioned above the model we use is a linear SVM hence the Kernel parameter, specifies the kernel type to be used in the algorithm, set to be 'linear'.

While applying the model on the synthetic data we create a parameters grid in order to choose the most compatible parameters for this problem. Because we solve a problem associated with unsupervised learning with classification method, we use tools from the world of supervised learning, such as accuracy, to select the most compatible parameters. Figure 6 shows the grid for this problem.

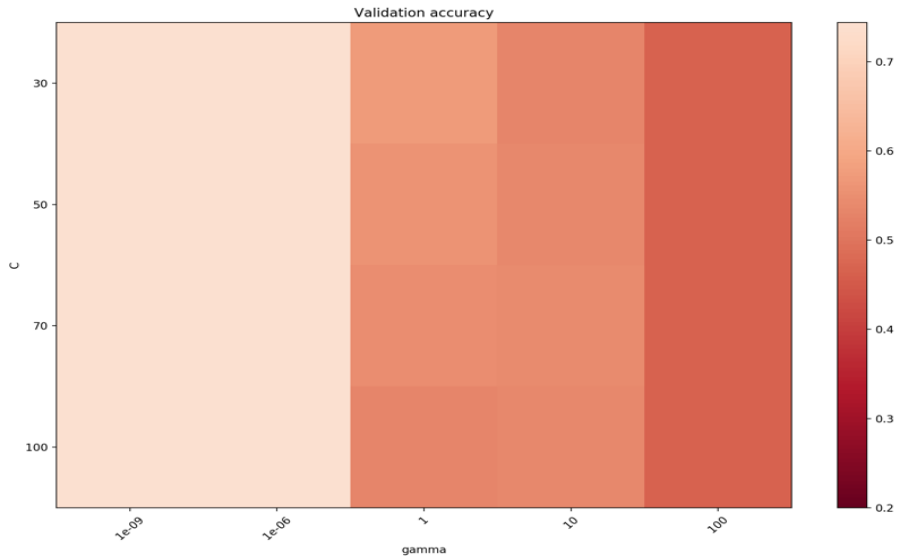


Figure 6: Parameters grid. Heatmap of the validation accuracy as a function of gamma and C.

As you can learn from the grid there is a range of values that lead to maximum accuracy (light color). According to the range we received, in the process of trial and error the final values were selected according to the most correct division into subgroups as defined in the synthetic data:

- C = 50.0: regularization parameter
- Gamma = 1e-9: kernel coefficient

The total number of records in the data is 20,000. In order to avoid over-fitting, the minimum records per leaf set to be half of that amount and the deviation from the error expressing the randomness of the classifier set to be 0.1.

For demonstration purposes [2]- if we take as example the synthetic data, we accept that the first classifier will separate between the purple distribution to the brown and red distributions (according to Figure 5), since we can see that the probability of an instance from the purple distribution to be part of population A is bigger than its chances to be part of population B. The tree we are accepted to get is demonstrate at Figure 7.

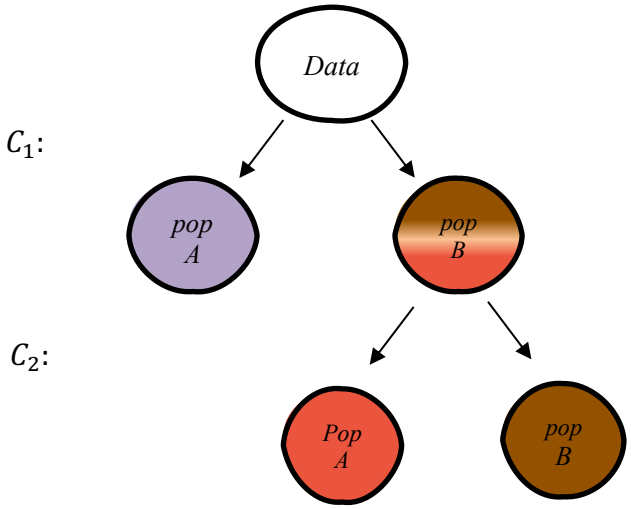


Figure 7: Classification tree for the synthetic data describes in figure 5. Each node in the tree is a classifier represented by C_i and the colorful circles represent the clusters.

4.2.2 Real - Life Data Set - Coronavirus

With the outbreak of the Corona epidemic in Israel, the Ministry of Health began publishing a database that includes characteristics of people who undergo Corona tests [23]. The features the database includes are:

- Test date
- Cough – Yes / No
- Fever – Yes / No
- Sore throat – Yes / No
- Shortness of breath – Yes / No
- Headache – Yes / No
- Corona result – Positive / Negative
- Age 60 and above – Yes / No
- Gender – Female / Male
- Test indication – Abroad / Contact with confirmed / Other

[2] Relevant code ("Classifier_Tree_Synthetic") at the following link - https://github.com/TAU-MLwell/sub-types

The database is updated twice a week, so it is important to emphasize that the data used for this project includes the records between March 3 and April 11, 2020. For our work, only the positive subjects are taken, i.e. only those infected with Corona, in total 9937 patients. In Figure 8 we can see the database structure.

test_date	cough	fever	sore_throat	shortness_of_breath	head_ache	corona_result	age_60_and_above	gender	test_indication
2020-03-22	Yes	Yes	No	No	No	Yes	1	זכר	Abroad
2020-03-22	Yes	Yes	No	No	No	Yes	1	זכר	Contact with confirmed
2020-03-22	No	Yes	No	No	No	Yes	0	נקבה	Abroad
2020-03-22	No	Yes	No	No	No	Yes	0	נקבה	Abroad
2020-03-22	Yes	No	No	No	Yes	Yes	0	נקבה	Contact with confirmed
...	...	...	...	...	...	...	...	...	...
2020-04-11	Yes	No	No	No	No	Yes	1	זכר	Other
2020-04-11	No	Yes	No	No	No	Yes	1	זכר	Other
2020-04-11	No	No	No	No	No	Yes	0	זכר	Other
2020-04-11	No	No	No	No	No	Yes	0	נקבה	Other
2020-04-11	No	No	No	No	No	Yes	0	נקבה	Other

Figure 8: Corona database. Each line is a patient tested positive to Corona and the columns are the database features.

Preprocessing[3] of the data is needed. First of all, the table contents are converted to binary encoding (1 for 'Yes' and 0 for 'No') and then records with no age indication is removed from the data. Afterwards we make features selection for the model, we only select the clinical symptoms and the test indication as features, therefore only the following columns remain – cough, fever, sore throat, shortness of breath, headache and test indication. The next step is to perform one hot encoding, a process by which categorical variables are converted into a form that could be provided to ML algorithms to do a better job in prediction.

In addition, since we are using classification method the data is divided into two populations by one of the risk factors mentioned above. The chosen risk factor is age, so all patients are divided into two populations - above age 60 and below age 60. This division create the accepted labeling of the data. At Figure 9 we can examine the final database structure provided to the model.

[3] Relevant code ('Preprocessing_Corona') at the following link - https://github.com/TAU-MLwell/sub-types

cough_No	cough_Yes	fever_No	fever_Yes	sore_throat_No	sore_throat_Yes	shortness_of_breath_No	shortness_of_breath_Yes	head_ache_No
0	1	0	1	1	0	1	0	1
0	1	0	1	1	0	1	0	1
1	0	0	1	1	0	1	0	1
1	0	0	1	1	0	1	0	1
0	1	1	0	1	0	1	0	0
...	...	...	...	...	...	...	...	...
0	1	1	0	1	0	1	0	1
1	0	0	1	1	0	1	0	1
1	0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0	1

head_ache_Yes	test_indication_Abroad	test_indication_Contact with confirmed	test_indication_Other
0	1	0	0
0	0	1	0
0	1	0	0
0	1	0	0
1	0	1	0
...	...	...	...
0	0	0	1
0	0	0	1
0	0	0	1
0	0	0	1
0	0	0	1

Figure 9: The final database structure provided to the model. Each line is a patient tested positive to Corona and the columns are the selected features after applying one hot encoding, in total 13 features.

Before applying the classifier on the Corona dataset and based on the explanation in section 4.2.1, in the process of trial and error the final parameters' values are:

- C = 50.0: regularization parameter
- Gamma = 1e-9: kernel coefficient

The minimum records per leaf set to be 2000, about fifth of the initial number of records. The deviation from the error expressing the randomness set to be 0.03, we allow a very small deviation that is one order of magnitude less than the deviation in the case of the synthetic data.

In order to understand the behavior of the first classifier and according to which of the features the classification is mainly based, we examine the excepted coefficient of the linear model at the root of the clustering tree. Coefficient plays major role in ML as the prediction

of the machine is depend on the coefficient. Coefficient indicates the relationship between the different features and the outcome of the classifier. The model's coefficients are presented at Figure 10.

cough_No	cough_Yes	fever_No	fever_Yes	sore_throat_No	sore_throat_Yes	shortness_of_breath_No	shortness_of_breath_Yes	head_ache_No
-1.001128	-1.001132	0.001025	1.001015	0.000062	-0.000062	0.000012	-0.000012	0.499894

head_ache_Yes	test_indication_Abroad	test_indication_Contact with confirmed	test_indication_Other
-0.499894	0.333085	-0.666707	0.333622

Figure 10: A total of 13 features, each of which shows the coefficient value it received for the prediction

From looking at these values it can be seen that the most weight is given to patients who have a fever, in addition the model does not actually use the cough symptom because there is the same weight both for the presence of cough and its absence. Following these results, we decide not to use cough as a feature of the model. The rest of the data remain the same except for the first two columns in Figure 9 that were removed.

Finally, the predictions of the different models were performed using cross validation (CV). CV is a technique for assessing how the results of a statistical analysis will generalize to an independent data set, and how accurately a predictive model will perform in practice. The goal of CV is to test the model's ability to predict new data that was not used in estimating it, in order to flag problems like overfitting or selection bias [24]. We use 10 folds for the predications of our model.

The process described so far was done for the first classifier that defines the root of the tree. Later in the process the same attributes of a classifier are used at each node, but the difference is in the data itself which is determined by the classification to two populations at the previous node.

4.3 Comparison to Classic Models

K-means [4] and Expectation – Maximization (EM) [5] algorithms are an unsupervised machine learning method for clustering. We conducted several experiments with synthetic data to compare the different methods when clustering in high dimensional spaces. The synthetic data we use composed from the same synthetic data describes in section 4.2.1 in the first two dimensions. On top of that, we added additional noise dimensions generated from the standard normal distribution with costumed variance of 200.

Our model receives the two samples as inputs, containing 10,000 instances each, while the reference algorithms, which are not designed to use double samples, receive the combined set of 20,000 instances as input [8].

We run 6 trials each with different number of dimensions, and measure the percentage of the points associated with the true originating clusters, after making the best assignment of the inferred centers to the true clusters. The Scikit- Learn implantations for EM and K-means are used for this comparison.

At the end of the day, the aim of this model is to demonstrate that when multiple samples (division into populations) are available, often it is best not to pool the data into one large sample, but that the differences in the different populations can be leveraged to improve clustering power [8].

5 Results

Applying the algorithm [4] to the Corona dataset described in section 4.2.2 produced the clustering tree shown in Figure 11.

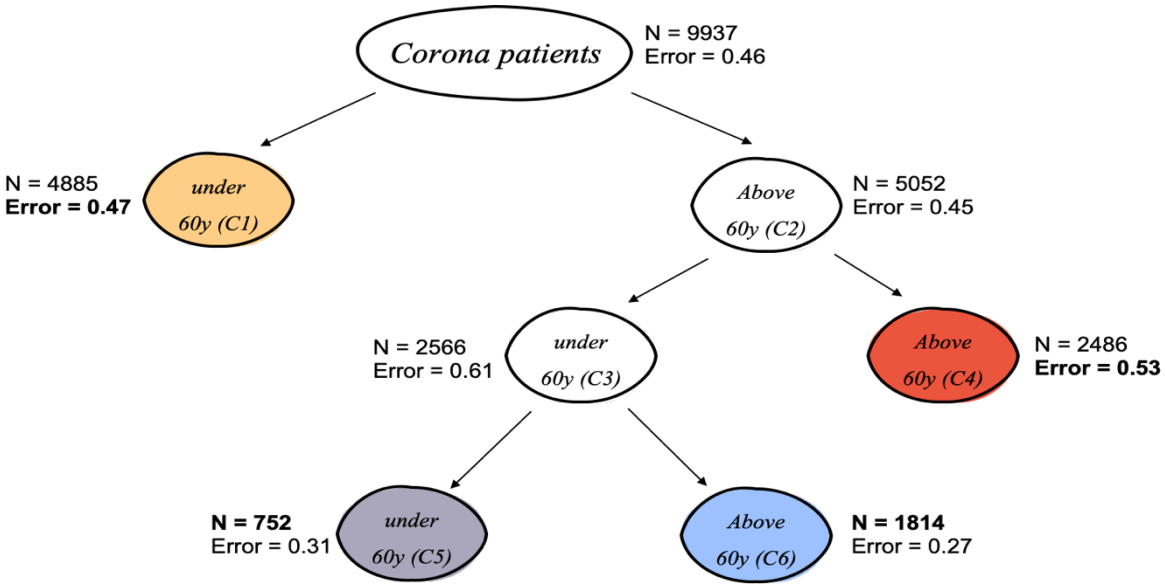
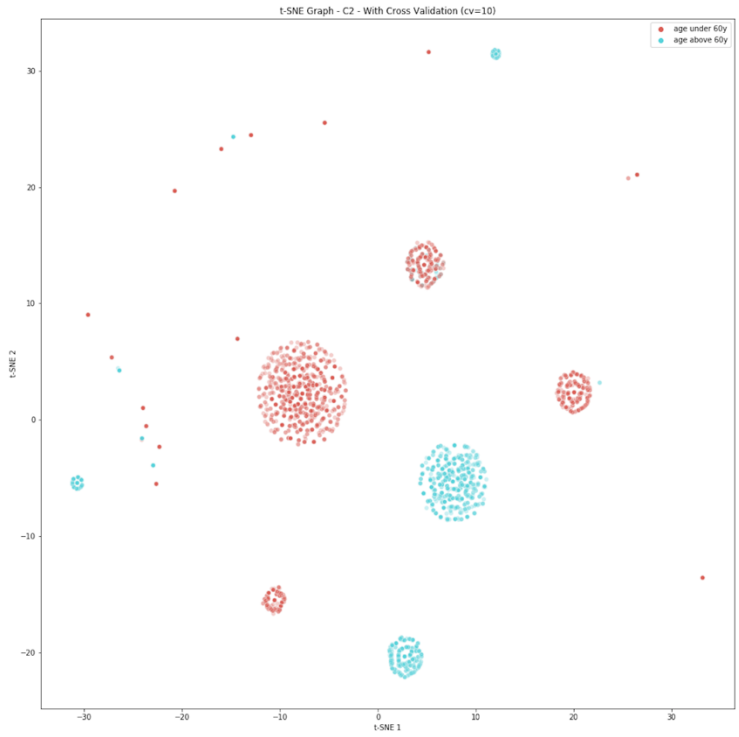


Figure 11: Clustering tree. Each node represents a classifier that divides the data into two populations according to age. Each of the four leaves represents a cluster. Next to each node and leaf we can see the amount of records and the empirical error of the classifier, next to each leaf is highlighted the base case leads to the creation of the terminal leaf according to section 4.2.2.

[4] Relevant code ('Classifier_tree_Corona') at the following link - https://github.com/TAU-MLwell/sub-types

We can learn from Figure 11 that in the process of creating the clusters, 3 different classifiers were trained. Each internal node is a classifier divides the data it received to two population, under the age of 60 or above.

In order to display the multidimensional space in a graph, we perform a dimension reduction using the t-SNE method. Figure 12 contains the division into populations of each of the three classifiers in the tree.



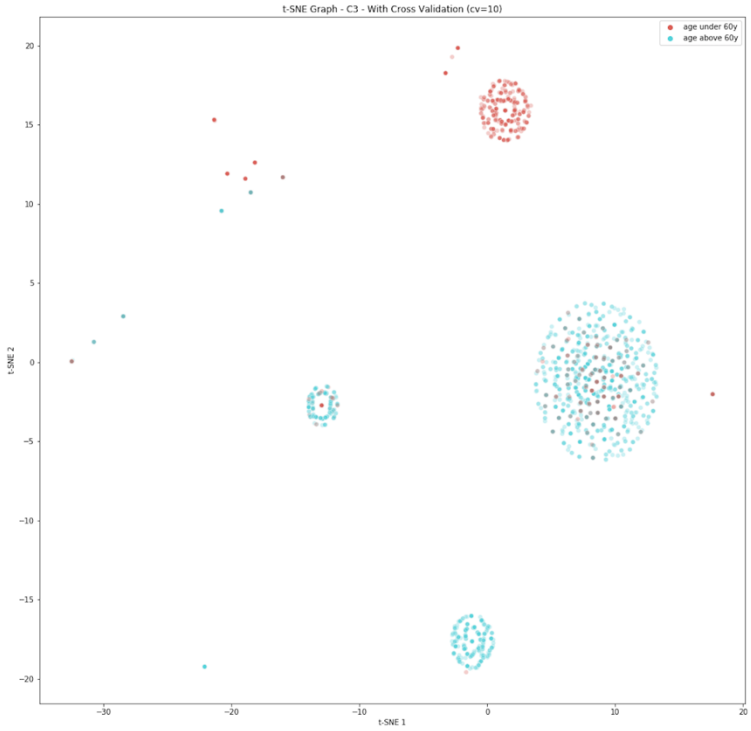


Figure 12: Each graph represents a classifier according the name at the its title. In the first step we performed a dimension reduction to the data in each node, the axes have no physiological significance other than spatial representation after the t-SNE process. The points were marked according to the prediction of each classifier as explained in the legend.

The spatial scattering of the points in Figure 12 is determined after the dimension reduction but the colors are determined according to the classification results of each of the internal classifiers in the tree shown in Figure 11. As can be seen the classification of the patients by age reveals a clear division of the data already in the early stages of the process.

The patients marked in red are classified by each of the models as patients under the age of 60. In other words, this is a group of patients has clinical characteristics that the model studied and on the basis of which it performs the classification. The same for the blue points which are classified as patients over the age of 60.

The general division of the algorithm is shown in Figure 13. As you can see in Figure 11 there are 4 leaves in the clustering tree created by the algorithm, each of which represents a cluster of patients. Dimension reduction is first done to the initial dataset, a total of 9937 patients, then each point is marked according to the color of the leaf to which it belongs in the tree (Figure 11).

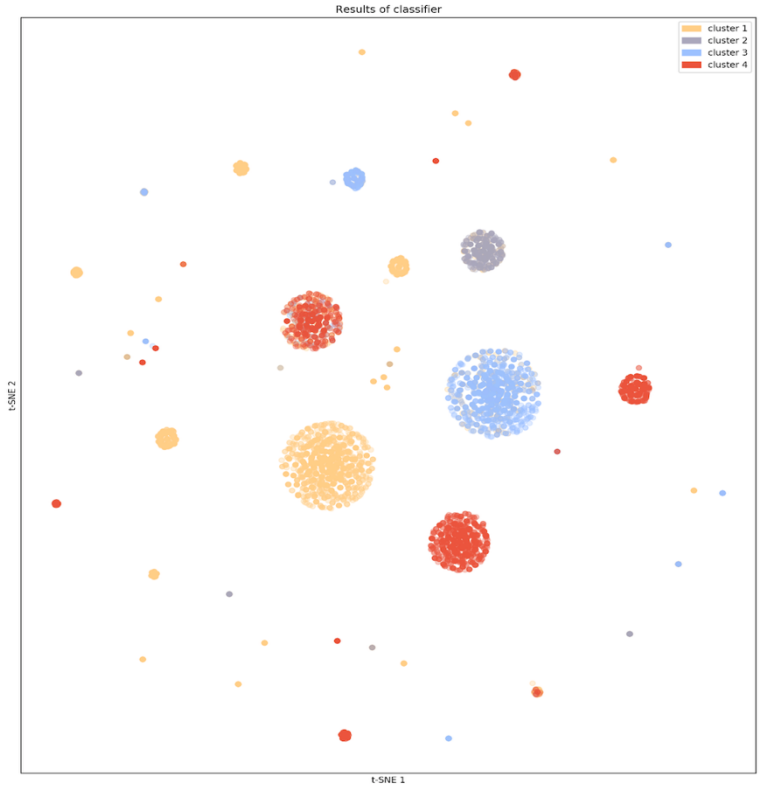


Figure 13: General division of the data. Each color represents a cluster ('sub-type') by the prediction of the algorithm.

As can be seen the spatial scattering of the points is the same as the scattering of the first graph in Figure 12, since the dataset at the root of the tree contains all the patients. The difference between the two graphs is the marking of the points, in the case of Figure 13 we witness the final result of the algorithm and the division into four clusters.

In addition, it can be noted that not all patients associated with the same leaf (points with the same color) are next to each other in terms of spatial scattering, this can be explained by the fact that the dimension reduction method gives the same weight to all dimensions (or in our case clinical features). According to the t-SNE method points that were adjacent to each other in the higher dimension, across all dimensions, will remain adjacent even when the number of dimensions is lower. The DSC algorithm, according to which we perform the coloring of the dots, is based on classifiers that study the various features and their association with the labels. In this case each of the features gets a different weight, with the help of the model's coefficients we can learn about the different weights and its effect on the classification. Therefore, the classification allows us to discover connections between patients who are not necessarily neighbors in the multidimensional space. Apparently, these

patients share common characteristics that have a higher impact on the classification problem.

The outcome of our algorithm is a division of the data without breaking clusters, or in other words, sub-types. Each of the subgroups we received represents a group of patients with common characteristics that may cause these groups of patients to respond differently to the disease. For example, the yellow clusters contain patients who have no symptoms of the disease, i.e. asymptomatic patients.

5.1 Comparison to Classic Models

Figure 14 shows the results of the comparison between the method we proposed (DSC) and two other existing clustering methods (K-means and EM). In this setting, the DSC method is expected to work well since the first two dimensions have a small variance while the variance in the dimensions of the noise is higher which creates a more challenging problem. The reason is that in these dimensions there is an overlap between the original clusters. Indeed we see that this is the case, when the number of dimensions is small we can see all three methods preform very well. However, when the number of dimensions is large, DSC outperforms the other methods; this corresponds to the difficult regime of low signal to noise ratio. Starting from 6 dimensions, DSC outperforms K-means and EM. While DSC continues to show performance with 100% accuracy, the other two methods, K-means and EM, drop to an accuracy level of 72% and 73% respectively. In 30 dimensions, the last two methods mentioned converge to an accuracy of 70%.

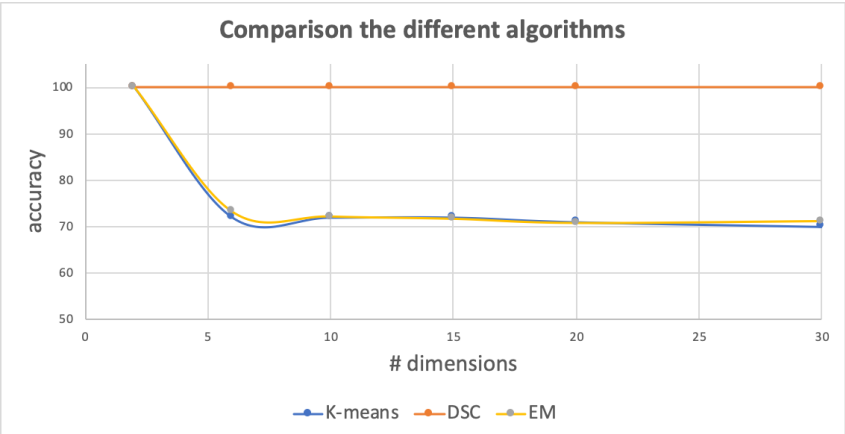


Figure 14: The dimension of the problem is presented in the X axis and the accuracy on the Y axis.

Another important advantage of the DSC method is that there is no need to provide the model with the number of clusters we want to find in the data. This principle is very important for the problem we want to solve because we only suspect the existence of different subgroups. On the other hand, the familiar methods require the user to define for the model the number of clusters it has to reveal.

6 Conclusions

The machine learning problem examined here is closely related to an unsupervised clustering problem. The clustering problem can be seen as the process of recovering the underlying structure of a date set. We present an algorithm that recover the underlying structure, under milder assumptions than the current clustering methods, by building a clustering tree.

Since different disease sub-types are likely to appear in different ratios across populations, we expect to see different proportions of each of the sub-diseases in two samples that we can generate from the data by a selected risk factor. The main principle leads us is that instead of pooling the data into one sample it is possible to use the differences between the samples to better recover the underlying structure. Meaning that when multiple samples are available, often it is best not to pool the data into one large sample, but that the structure in the different samples can be leveraged to improve clustering power.

Using data from Corona patients we examined the premise of our work that division into populations is sufficient to expose the sub types. By dividing the Corona patients into two populations according to their age, we exposed a clear division into subgroups of the patients. These results indicate that there may be clusters of people who response differently to the disease. In total we exposed 4 subgroups suspected as Corona sub-types, each group is composed of corona patients who may be affected by the disease differently. Early detection of these groups and classification of a new patient into its right group can contribute to predicting the effect of the disease on a patient and whether he will develop a severe illness. From the results of the study we can conclude that there is a great possibility for the correctness of our premise, but an unequivocal determination will be possible after further research of the disease. Validating this assertion and understanding its consequences is left for future study.

The comparison between the different clustering methods and the DSC algorithm is not straight forward. Clustering analysis is not based on a specific algorithm, it can be achieved by various algorithms that differ significantly in their understanding of what constitutes a cluster and how to efficiently find them. In addition, in our problem of finding sub-types of diseases it is barely possible to use the existing clustering methods since we only suspect the existence of sub-groups and more than that we do not know the number of groups underlie the structure of a disease, and for most of the currently known clustering methods it is required to predict in advance the number of subgroups. At the same time using synthetic data whose properties are well defined, we presented a comparison between different methods. We proved that DSC outperforms the current methods when either the dimensionality or the separation is high.

As mentioned above a follow-up study is necessary for an unequivocal determination about the division into sub-diseases. Today, after several months into the epidemic, there is more data on corona patients. First of all, there are more patients and therefore the algorithm can be run on much larger data. In addition, there is more information about the condition of patients after diagnosis. We can ask questions such as: whether the patient has mild or severe illness, whether he needed hospitalization, whether he needed respiratory assistance and, in some cases, whether the disease led to death. Additional information about how the disease effect the different patients forms the basis for further research of subtypes. For example, we can collect this data on patients from the subgroups we have exposed and characterize each group. In this way we can determine if these are indeed groups of patients who are affected by the disease in a different way.

It is also possible to use the algorithm in the field of machine learning. Dimension reduction methods, including t-SNE, embed the problem space from a high dimension to a low dimension. In these cases, the axes defining the problem lose their physical significance and therefore it is difficult to explain the new spatial scattering of the points. Using the algorithm to classify points in a data set for different samples, even multiple and not just double samples, can contribute to a better understanding the relationship between points after reducing the dimension and determine which properties each cluster of points has. For example, with the help of t-SNE we can learn about proximity between points across all dimensions, and with the addition of our algorithm we can also distinguish whether there is proximity between different points across some of the dimensions that have a higher influence in the classification problem.

To conclude, in this project we propose an algorithm that can be viewed as clustering algorithm. It is a generic algorithm that can be applied to different data sets to better recover the underlying structure of the data or as in our case, of a disease. Since some diseases may have subtypes that respond differently to treatment, it is very important to identify and characterize them early. We have seen that corona patients' data can be divided into four groups, each of which may consist of patients who are affected differently by the disease. In order to unequivocally determine the existence of the subtypes further research should be done on this topic.

7 References

[1] Ho, Tin Kam. “Random Decision Forests.” In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, 278-282. Montreal, Quebec, Canada, 1995.

[2] Kleinbaum, Klein, and Klein, Mitchel. *Logistic Regression : A Self-learning Text, 2nd ed.* New York: Springer, 2002.

[3] Guenther, Nick, and Matthias Schonlau. “Support Vector Machines.” *The Stata Journal 16*, 2016: 917-937.

[4] Kogan, Jacob. “k-means with Kullback–Leibler divergence.” In *Introduction to Clustering Large and High Dimensional Data*, 94-96. Cambridge: Cambridge University Press., 2007.

[5] McLachlan, Krishnan & Krishnan, T. *The EM Algorithm and Extensions.* New York: John Wiley, 1997.

[6] Puri, Ratik. *Exploring Machine Learning.* 18 Dec 2018. https://medium.com/datadriveninvestor/exploring-machine-learning-f1dc6f3ec902.

[7] J.McLachlan, Geoffrey. “Mixture Models in Statistics.” In *International Encyclopedia of the Social & Behavioral Sciences (Second Edition)*, 624-628. Brisbane, QLD, Australia: University of Queensland, 12 March 2015.

[8] Jason Lee, Ran Gilad-Bachrach and Rich Caruana. “Using Multiple Samples to Learn Mixture Models.” *Advances in Neural Information Processing Systems.* 28 Nov 2013.

[9] Blei, David M. “Probabilistic Topic Models.” *Communications of the ACM, vol. 55, no. 4*, 2012: 77–84.

[10] Pascual, Federico. *The Essential Guide to Topic Modeling.* 26 September 2019. https://monkeylearn.com/blog/introduction-to-topic-modeling/.

[11] Judea, Pearl. Causality : Models, Reasoning, and Inference. Cambridge: Cambridge UP, 2000.

[12] Daniel M. Hausman and James Woodward. “Independence, Invariance and the Causal Markov Condition.” *The British Journal for the Philosophy of Science Vol. 50 pp 521-583. 4* Dec 1999.

[13] Zhonghua Liu Xing Bing Xue Za Zhi. “Epidemiology Working Group for NCIP Epidemic Response, Chinese Center for Disease Control and Prevention”, 2020.

[14] Massachusetts Medical Society. “Another Decade, Another Coronavirus”. *The New England Journal of Medicine,* Feb 20, 2020.

[15] "COVID-19 Dashboard by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University (JHU)". ArcGIS. Johns Hopkins University. Retrieved 22 August 2020.

[16] U.S. Centers for Disease Control and Prevention (CDC). "How COVID-19 Spreads", 2 April 2020. https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/how-covid-spreads.html

[17] U.S. Centers for Disease Control and Prevention (CDC). "Symptoms of Coronavirus", 20 March 2020. https://www.cdc.gov/coronavirus/2019-ncov/symptoms-testing/symptoms.html

[18] Luzon, Muchnik, Koren. MAFAT Coronavirus Blog, 13 April 2020. https://blog.mafatchallenge.com/2020/04/13/covid-19-testing-with-ml/

[19] U.S. Centers for Disease Control and Prevention (CDC). "People at Increased Risk", 10 March 2020. https://www.cdc.gov/coronavirus/2019-ncov/need-extra-precautions/index.html

[20] Brownlee, Jason. "A Gentle Introduction to Bayes Theorem for Machine Learning". 4 Decmber 2019. https://machinelearningmastery.com/bayes-theorem-for-machine-learning/.

[21] Cortes, Vapnik. "Support-vector networks", Machine Learning. Vol. 20: 273–297. 1995

[22] Support vector machines, scikit-learn library for Python, https://scikit-learn.org/stable/modules/svm.html#svm-classification.

[23] Data collection and sharing was supported by the Israeli Ministry of Health. You can learn more about the database at: https://data.gov.il/dataset/covid-19/resource/d337959a-020a-4ed3-84f7-fca182292308

[24] Allen. "The Relationship between Variable Selection and Data Agumentation and a Method for Prediction". *Technometrics Vol 16 pp 125-127.* February 1974.