Project

Valiullina Anna
Kaplun Igor
Wang Kunteng
Tekin Onat
Özdemir Ozan

Data Science in Practice
École Polytechnique Fédérale de Lausanne
Lausanne, Switzerland

13th May 2019

Contents

1 Introduction

1.1 Background

Heart disease is the disorder of the heart and is the state of being unhealthy. Heart disease can be very fatal. To understand better, a known example of heart disease is the hearth attack. Heart attack is a pathological condition which can result in death. It is often due to a congestion of coronary arteries, which causes the lack of oxygen in heart muscles (4). Heart disease is really important disease to be examined. Here are some surprising fact about heart disease (5):

- First of all heart disease can be very fatal. According to the Centers for Disease Control and Prevention (CDC) of U.S. Department of Health and Human Services, it the most frequent cause of humans.
- In 2015, the men who lost their life because of the heart disease is more than 50% in men.
- Each year more than half million people die from heart disease in the world.
- 25% of all the deaths is caused by heart disease. Within these heart disease, heart attack is the most common type which kills approximately 366,000 people in 2015.
- The rate of people having a heart attack is 40 seconds in United States.
- In United States, the rate of people dying by a heart disease related event is 1 minute.
- Also in United States, the financial cost of heart disease costs roughly $200 billion each year, including all the cost related to the health care services, pills and medications and the loss of productivity caused by disease.

The purpose of this study is to apply analytical data science tools on a heart disease data set provided by V.A. Medical Center, Long Beach and Cleveland Clinic Foundation: Robert Detrano, M.D., Ph.D. Since, the heart disease is health related subject, the project is driven by both ethical and financial motivation.

1.2 Problem Statement

Based on the facts provided above, the main goal of this project is to detect the features which significantly cause the heart disease through the machine learning

algorithms. There are some important challenges related to the detection of these causes. These challenges are :

- Based on the data set, problem needs to be formulated well. Otherwise, the methodology and the results can be good but useless. Therefore the expectation and and the goal should be clear. The result from this project should be valuable.
- Preprocessing the data is another important challenge. Does data need to be cleaned, or should some features to be extracted or selected. Otherwise, the result can be misleading.
- There are a lot of machine learning algorithms and tools. Which one is the most useful for this project. Again, this question is challenging. It is hard to decide which algorithm performs the best before understanding the data set. The right parameters and strategies should be chosen to tune the algorithms.
- Also, the balance between the accuracy and the velocity of the algorithm should be found. The model requires to be accurate and fast simultaneously. Otherwise, model is not adequately proper.
- After all is done, the causality and the correlation should be examined carefully to conclude the right results. All can be done fairly good, however still the project can miss the correct results.

To accomplish the main purpose of the project, firstly, the data set should be controlled if it is convenient for classification or regression. So that, accordingly supervised and/or unsupervised methods can be applied in order to explain the response variable. Secondly, the preprocess of data as preparation for the analysis should be taken care of. These preparation is mainly, data cleaning and maybe merging, extracting some data, if there is such a need. After this step the data is ready to be focused on. Therefore third step is understanding the data by conducting a detailed study on it. If there is a need, transform the data, the numerical variables into categorical or vice versa in terms of coding requirement. Then with the help of algorithms which will be analyzed later on this report, first exploratory analysis and machine learning analysis should be conducted. Finally, after interpreting the results there should be an evaluation and conclusion to examine the performances and to discuss further research that may be done after these results.

1.3 Value of Project

Early diagnose of the heart disease is particularly important (2). Therefore, the value of the project is first of all the opportunity of helping more people. On

the other hand, of course financially this project creates the value of decreasing the cost of health care, medication. Moreover, detecting the main features that cause heart diseases provide direction to follow for the scientific research to prevent heart disease or develop better diagnostic tools or medications. With all these benefits, this project is expected to be not only useful for one specific firm, but for humanity in general.

2 Data Description

2.1 General Information About the Data

The dataset to be analysed contains 303 observations which represents the number of patients for each 14 features. Within these features one of them is the response value, and the rest of them are the specifications of patients. The dataset contains both categorical information and numerical information, however the categorical information are given by numeric representations. For example, the feature "thal", thallium stress tests has three values: 1, 2, and 3. Here, 0 implies no defect, 1 implies normal, 2 implies fixed defect, and 3 implies reversable defect. Also, some of the variables are boolean variables. For example, the response value has two values: 1 implies the presence of heart disease in the patient, while 0 implies the absence of heart disease. The slope variable is also categorical. Value 1 shows the slop is upsloping, while value 2 shows that it is flat, and Value 3 shows downsloping. Table 1 shows the complete description of features.

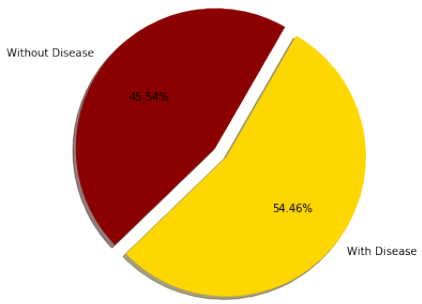


Figure 1: The Histograms of Numerical Values

Table 1: Description of the Features

Feature	Description	Variables
age	Age of the patient	numerical
sex	Sex of the patient	0, 1
cp	Presence of the chest pain	0, 1, 2, 3
trestbps	Blood pressure when patient rests	numerical
chol	Serum cholesterol of the patient	numerical
fbs	If fasting blood sugar larger 120mg/dl	0, 1
restecg	Electrocardiographic results when patient rests	0, 1, 2
thalach	Maximum achieved heart rate	numerical
exang	Exercise induced angina	0, 1
oldpeak	ST depression induced by exercise relative to rest	numerical
slope	Slope of the peak exercise ST segment	0, 1, 2
ca	Amount of colored major vessels by flourosopy	numerical
thal	Thallium stress tests	0, 1, 2, 3
target	Presence of heart disease	0, 1

As it can be seen from the table 1, data contains 6 numerical explanatory variables, 7 categorical explanatory variables and 1 categorical response variable. First of all, the proportion of the target value was examined. The 45.54% of the patients have heart disease, 54.46% of them do not have heart disease.

2.2 Summary Statistics

For the numerical values, the minimum and maximum values can give some ideas to understand how the features are. However, histograms are more useful to observe outliers, exceptions and general look for distributions. Therefore in Figure 2, it can be observed that age and thlach features are left skewed. On the other hand, chol and trestbps are right skewed. At first glance, the features ca and oldpeak look like chi-square distribution. Even before looking into more details and analysis, one can say the outliers looking into this histograms.

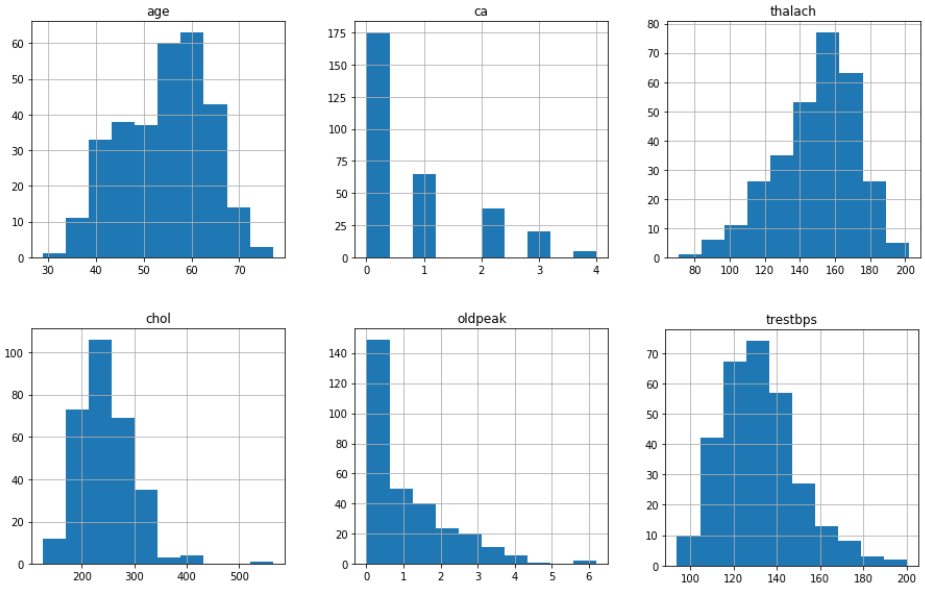


Figure 2: The Histograms of Numerical Values

To continue the analysis, it is good idea to have the basic statistical information, such as mean, variance, count, minimum and maximum values, and also some quantiles. These statistical information are only valuable and meaningful for numerical values. For categorical values of the data set, another measures will be applied.

	age		ca		thalach		chol		oldpeak		trestbps	
	0	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**
count	138	165	138	165	138	165	138	165	138	165	138	165
mean	56.60	52.50	1.17	0.36	139.1	158.47	251.09	242.23	1.59	0.58	134.40	129.30
std	7.96	9.55	1.04	0.85	22.6	19.17	49.45	53.55	1.30	0.78	18.73	16.17
min	35	29	0	0	71	96	131	126	0	0	100	94
25%	52	44	0	0	125	149	217.25	208	0.60	0	120	120
50%	59	52	1	0	142	161	249	234	1.40	0.20	130	130
75%	62	59	2	0	156	172	283	267	2.50	1	144.75	140
max	77	76	4	4	195	202	409	564	6.20	4.20	200	180

Figure 3: The Descriptive Statistics for Numerical Values

Also for the numerical analysis, the correlation among the values are important to examine. In order to do so, the correlation heat map can be used. The heat map is a colored table (i.e. map) in a scale from dark to light to show the correlations between the variables as a matrix. The cells on the diagonal usually are not shown,

since the correlation of a variable with itself is simply 1. In the heat map the values of the first variable is the first column of the table. Then it continues with second variable as second columns and so on. There is a legend column to show the color spectrum at the right. with this heat map the anomalies are expected to be detected easily.

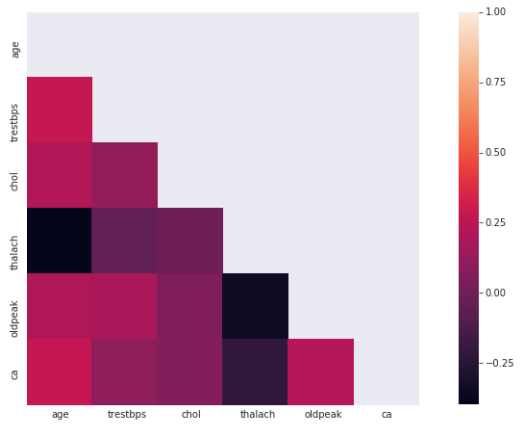


Figure 4: The Correlation Heat Map for Numerical Values

Based on the correlation heat map, it can be observed that there is no large correlation between variables. None of them are close to 1 nor -1. Based on the fact that the correlations are between -0.25 and 0.25, there is no necessity to exclude any variables, before starting further analysis.

Categorical values are the ones which belong one or more categories. It can be binary or boolean, so that there exist two categories 0 and 1. For example, the response variable with 0 and 1 or a judgment with good (1) and bad (0). Also, there can be many categories, for example, the evaluation surveys with strongly disagree, disagree, neutral, agree, strongly agree options. It's also possible to assign numerical representatives to these categories as in binary examples. Thus, in the first statistical anaylsis of these categorical variables, the most common value (top), the most common values frequency (freq) and the number of unique values (unique) are going to be examined.

	cp		fbs		sex		thal		exang		restecg		slope	
	0	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**
count	138	165	138	165	138	165	138	165	138	165	138	165	138	165
unique	4	4	2	2	2	2	4	4	2	2	3	3	3	3
top	0	2	0	0	1	1	3	2	1	0	0	1	1	2
freq	104	69	116	142	114	93	89	130	76	142	79	96	91	107

Figure 5: The Descriptive Statistics for Categorical Values

3 Methodology

3.1 Preprocess of the Data

The "data" can be defined as the new crude oil of 21st century, since it may help corporations to understand the behaviour of customers, extract some causal relationships, and focus on the right customer segment to increase efficiency by lowering costs. Therefore, corporations find themselves in a competition to collect the largest amount of data. For instance, Facebook, Amazon, and Google rent and build huge data warehouses although they may not discover the right method to use these data. This competition has been boosting the speed of data collection enormously and sometimes this data may not be well organized for the models. Because of this unstructured dirty data, people search for the methods to make the data ready for modelling, calling it as pre-processing. In a manner of speaking, the people of interest want to avoid "garbage-in-garbage-out" issue.

- **Feature scaling:** Although some of the algorithms are robust for scaling issues, some of them are not robust; therefore, those incapable methods may require scaling. Scaling can be defined as making the ranges of columns same level (preferably 0-1) roughly because if the range of one column is larger than the other column, the range difference between these columns may lead to large gradient updates in favor of the large scaled column by the optimizer or divergence from the optimal solution. The one can choose min-max scale or standardization as scaling method.
- **One-Hot-Encoding:** One-Hot-Encoding has primary importance to process the categorical variables for modelling. The tool essentially create n binary columns in which n is the number of unique categories in the corresponding feature. The one should drop one of the columns created, otherwise the created columns may lead to singularity (3).

3.2 Unsupervised Learning

3.2.1 Multidimensional Scaling (MDS)

When we are working with a large Data Set where each sample contains several properties it's getting harder and harder to see the patterns in the Data Set while increasing the number of properties. In this case there are several useful techniques which could help us to see the similarity among the samples. One of these useful tools is Multidimensional Scaling.

Describing in simple words, Multidimensional Scaling helps us to obtain the position of the points (samples) in the k-dimensional space having the Euclidean distances between the points in the n-dimensional space. What's important here is that the resulting position of the samples in the k-dimensional space doesn't make any difference, because what's important is the position of the samples relative to each other.

The great illustration of this concept would be an example from the "Python Data Science Handbook" by Jake VanderPlas (8). Lets consider that initially we were at the 3-dimensional space and we have reduce the dimension of the data up to 2-dimensions. Now think of the list of paper as a resulting 2-dimensional space. If we have any dots (or other information) written on that sheet then rotating, rolling or banding (or even scaling) that sheet wouldn't affect the information we get from the sheet.

Having X as a desired coordinate matrix lets denote matrix B as $B = XX^T$. Matrix B then is positive definite symmetric matrix.
That implies that B can be written as $B = V\Lambda V^T$ where Λ is a dioganal matrix of eigenvalues $\lambda_1 > \lambda_2 > ... > \lambda_n$ and V is a matrix of normalized eigenvectors. At this set up the resulting matrix X would have a form

$$X = V\Lambda^{\frac{1}{2}}$$

The only concern now is to how we actually define matrix B from the given data.(11) We are doing it connecting each element of matrix B with the input distance between 2 samples (dots) d_{ij}.

$$d_{ij} = b_{ii} + b_{jj} - 2b_{ij}$$

$$b_{ij} = -\frac{1}{2}(d^2_{ij} - d^2_{i\cdot} - d^2_{\cdot j} - d^2_{\cdot\cdot})$$

Advantages MDS

- In contrast to the PCA that will be discussed later, MDS can manage non-linear relationship in the data.

Disadvantages of MDS

- It cannot solve the issue of missing data.
- It cannot exactly determine the optimal number of output dimensions.

3.2.2 PCA

In unsupervised learning, Principal Component Analysis (PCA) is a widely applied techinique to analyze and simplify the data sets by reducing the dimensions and changing the multiple indicators problems into several indicators problems.

The main idea of the PCA is to do the decomposition on the covariance matrix of the data set in order to find its eigenvectors and eigenvalues. In other words, they are its principal components and weights. Based on these vectors and values, we can figure out which component has the largest influence on the total variance. In addition, PCA can provide a low-dimension graph when the original multivariate data set can be illustrated in the high-dimension coordinates. In other words, this graph is a projection of the original data set in the low-dimension coordinates. By using this technique, we can illustrate the original data set well with several principal components.

PCA also has a wide range of applications. For example, in the quantitative finance field, the APT models can be applied to find the effective factors from huge amount of the factors, which can provide highest R-squared value. The easiest trading strategy is to take the long positions on these stocks, which can be found based on the effective factors. However, PCA is just a tool for factor digging. It is hard for us to give an explanation why these chosen factors can perform well.

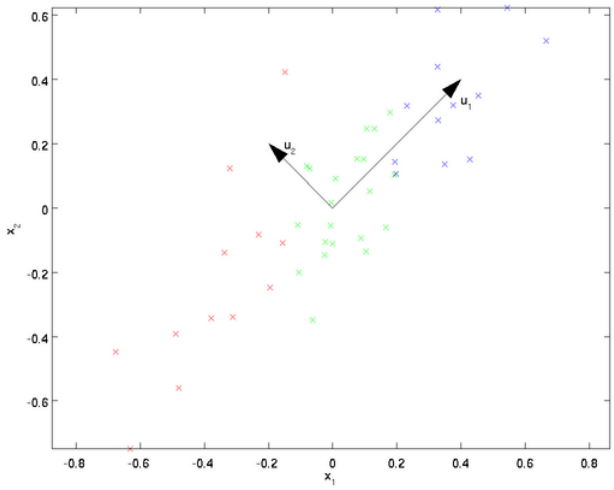


Figure 6: The demo of PCA

Advantages of PCA (10)

- **Effective for large datasets:** PCA is able to deal with large datasets in terms of objects and variables.
- **Reducing dimesionality:** PCA reduces dimensionality in size of data, therefore it becomes easier to learn Reduction of data.
- **Removing noise:** PCA removes the noise therefore the significant regularities becomes more obvious. These regularities are the uncorrelated components.
- **Fast computation:** PCA has a fast computation time.
- **No strong assumption:** PCA has not specific assumptions about the data, therefore it can be used on any dataset.
- **Suitable for high-dimensional data:** PCA is suitable to be used for estimation of probabilities in high-dimensional data.

Disadvantages of PCA (10)

- **Linearity Restriction:** PCA is suitable for linear models. It is difficult to model non-linear problems with PCA.
- **Complex Results:** The results may be hard to visualize and therefore sometimes they can be hard to be interpreted.

3.3 Supervised Learning

3.3.1 K-Nearest Neighbor

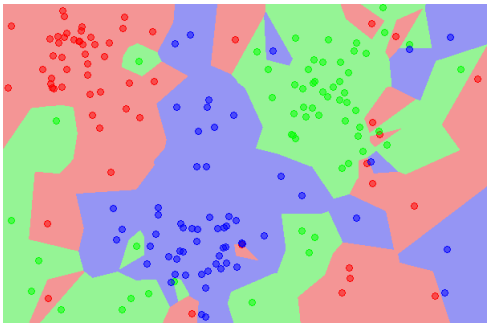
In machine learning literature, K-Nearest Neighbor is one of the most fundamental methods employed for both classification and regression analysis. The method can be identified a non-parametric approach meaning that it is not rigorously based on the group of parameters constructing a probability distribution. The main idea of the algorithm lies in the "k" closest training examples in the feature space.

KNN can be defined as an instance based learning algorithm or a lazy learner that are generic concepts in data science literature since it memorizes all the instances rather than setting the model of the relation between response variable and the explanatory variables. If we classify the whole computation time into two categories as training time of the model and prediction time of new instances, lazy learner algorithms assign very short portion of computation time to training time. This reduced training time feature comes up with a trade-off that each time the user wants to predict a new instance, the lazy learner seeks the closest neighbors in the whole training set. Consequently, it decreases the training time as well as it increases the prediction time.

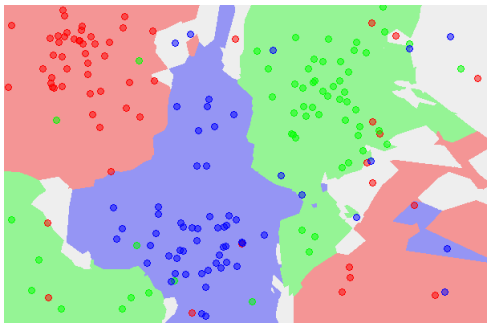
For KNN as a classification method, the algorithm follows a simple procedure. In this procedure, the response variable would be defined as the class of instance and the instance will be classified with respect to the the most popular class among its k-nearest-neighbor based on a pre-defined distance measure. The distance measure is based on the preference of user, being Euclidean, Manhattan etc.

The classification fields of knn may vary with respect to the value of k. When we increase the k value, we will increase the consistency and lose the sensitivity. The change in fields with respect to k values can be examined as below (6):

Figure 7



(a) The Fields of Classes with respect to 1 KNN



(b) The Fields of Classes with respect to 5 KNN

Advantages of K-Nearest Neighbor

- KNN is relatively intuitive and basic algorithm to understand and implement on both of regression and classification problems.
- There is no training time or pre-defined set of assumptions.
- Since each time a new instance appears, it updates the whole data set and improves the prediction evolutionary.
- Only one hyper-parameter exists to be tuned, being the number of nearest neighbor.
- It is easy to implement on multi-class problems.

Disadvantages of K-Nearest Neighbor

- Since it is instance based algorithm or lazy learner, the implementation speed is inversely proportional to the volume of dataset.
- It needs pre-processing such as scaling before implementation.
- It is easily affected by the curse of dimensionality and imbalanced data.
- It cannot propose a causal relation between response variable and explanatory variables.
- It is hard to find the optimal number of nearest neighbors since the algorithm evolves continuously.

3.3.2 Logistic Regression

Logistic regression is widely used classification algorithm to predict a binary response variable in its vanilla form. Although the algorithm is one type of regression models, it basically estimates the parameters of binary logistic model to restrict the response variable between 0 and 1 as the probability of being 1 for each instance. It can be classified as parameterized model and eager learning algorithm since it constructs a concrete model with certain parameters.

The technical set-up of logistic regression is based on transforming the linear predictions into a part of logistic odds and obtain a value between 0 and 1 and classifying the probabilities of instances with respect to a pre-defined threshold. The transformation can use the sigmoid function and the produced probability values can be represented as:

$$h_\theta(x) = \phi(\theta^\intercal x)$$
$$\phi(z) = \frac{1}{1+e^{-z}}$$

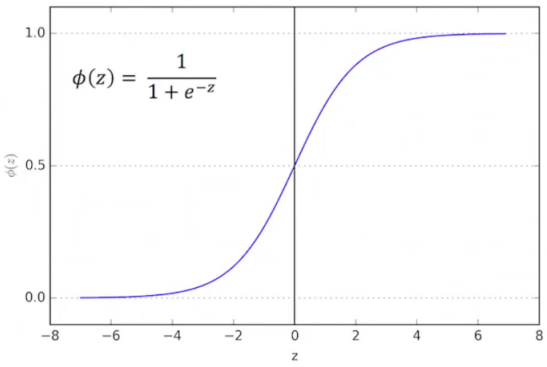


Figure 8: The Output of Sigmoid Transformation with respect to z values

After the transformation procedure with the help of sigmoid function, the optimization algorithm of parameters can be defined as (1):

$$min_\theta \quad J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost(h_\theta(x), y)^i$$

$$where$$

$$Cost(h_\theta(x), y)^i = y^i log(h_\theta(x^i)) + (1-y^i) log(1-h_\theta(x^i))$$

$$J(\theta) : \text{The Cost Function}$$

To predict the class of instances, a threshold should be defined as decision boundary. Based on this threshold, the probability values can be transformed into class values. Logistic regression models can be extended to be used for the multi-class problems, and the main approach for this reason would be ONE-vs-ALL approach.

Advantages of Logistic Regression

- The outputs of the model usually have a nice interpretation in terms of probability.
- Since the model has linear relationship between the response variable and explanatory variables, the parameters can be computed rapidly.
- The algorithm can be regularized to eliminate the overfitting issues.
- Logistic regression models can be improved with the updated new data using stochastic gradient descent.

Disadvantages of Logistic Regression

- It usually underperforms when there are more than one decision boundaries, XOR problems can be given as an example.
- It may not identify the complex relationships naturally.

3.3.3 Support Vector Machines

In machine learning field, SVM is a widely applied supervised learning method to analyze the classification and regression problems. The basic idea of SVM is to treat the data points as p-dimension vectors and try to find a hyperplane in p-1 dimension which can separate these data points as far as possible.This is the linear classifier. If such a hyperplane exists, the linear classifier that is defined by that is also called as maximum-margin classier(16). The corresponding plot(17) can illustrate it vividly.

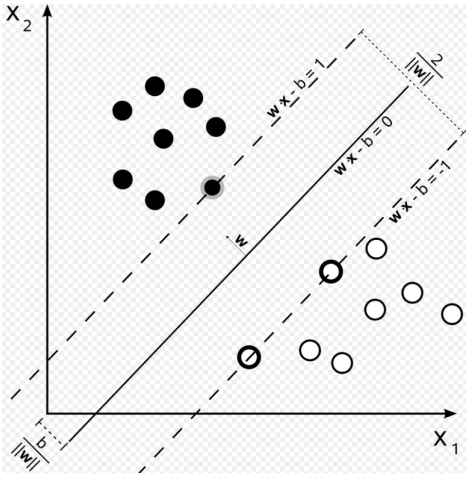


Figure 9: The demo of SVM

In the industry, SVM plays a role in the fields like text analysis and categorization because of their properties that they need less labeled training samples.In addition, the hand-written characters and images classification can achieve higher precision and accuracy, compared with the traditional methods.

Advantages of Support Vector Machine

- SVM can manage to deal with the complex nonlinear classification based on appropriate kernel functions.
- It can avoid the curse of the dimensionality when it deals with the high-dimension data.
- The SVM algorithm is much more general and robust, which means the risk of overfitting is less.
- SVM is a good choice when we have no idea how to deal with the data.

Disadvantages of Support Vector Machine

- Hard to explain and interpret the results.
- It takes long time to train the large datasets.
- It is hard for SVM algorithm to deal with the multi-classification problems, because the classic SVM is designed for two-classification.

3.3.4 Random Forest

Random forest is a classifier consisting of a collection of tree-structured classifiers, where each tree casts a unit vote for the most popular class at input (12). Tree-structured classifiers, or Random trees, are a family of classification algorithms based on division of the input dataset into subsets according to a certain set of rules which are chosen to minimize an error function

$$Q(X_m, j, t) \longrightarrow \min_{j,t},$$

where

X_m is the data subset in node m;

j, t are the parameters of the decision rule.

Error function can be expressed in the following way:

$$Q(X_m, j, t) = \frac{|X_l|}{|X_m|} H(X_l) + \frac{|X_r|}{|X_m|} H(X_r),$$

where

X_l and X_r are the subsets contained in the left and right leaves of node m correspondingly;

$H(X_i)$ is an information criteria function.

Information criteria accounts for the variance of classification results given by a certain subset. In present paper Gini information criteria is used:

$$H(X) = \sum_{k=1}^{K} p_k(1 - p_k), \; p_k = \frac{1}{|X|} \sum_{i \in X} [y_i = k],$$

where

K is the number of classes in the problem.

Therefore, the branches of the tree are constructed. Leaves of a Random tree contain indivisible subsets based on which class belongings the classification decision is made:

$$a_m = \text{argmax}_{y \in \mathcal{Y}} \sum_{i \in X_m} [y_i = y],$$

where

$\mathcal{Y}$ is a set of all classes (sets 0 and 1 in the case of this paper);

y_i are the indicators of $i-th$ dataset item to a certain class.

Random trees are powerful classification tools when it comes to non linear dependencies. Though they are also fast and easy to build and to fit, random trees have several significant drawbacks that harm their performance severely: random trees are easily overfitted and very unstable when the training dataset is slightly changed. In other words, Random forests tend to have low bias and large variance.

The reason for the overfitting feature lies in the tree-structure of the algorithm: the best fit is reached when each leave corresponds to one item of the dataset. Hence, some measures have to be taken in order to address potential overfitting. One of possible ways to mitigate the overfitting issue is to construct a problem such that the algorithm would build a tree of minimal size among all the trees which deliver minimum to the error function. Due to the fact that the aforementioned problem is an NP-complete problem, Random trees are always built on the greedy basis, as an alternative way of mitigating the overfitting problem. Practically, this implies construction of the tree step by step from the top to the bottom, following certain stopping rule, which would either limit the size of the leaves or the depth of the tree. In the present paper the latter approach is used and the depth of Random tree is taken as a hyper-parameter of the model.

The second drawback of Random trees, their tendency to abrupt changes when a slight changes are made to the training dataset, can be turned into their advantage in the process of the fight with overfitting. Such feature implies that even on a relatively small dataset a significant number of different Random trees can be constructed and each of them will give different solutions to the classification problem. In order to pick enough training datasets in present paper Bootstrap method is applied to the initial dataset. Moreover, we can construct an ensemble of Random trees, which is Random forest. The output of Random forest a formed by N different Random trees with outputs b_n can be expressed in the following way:

$$a(X) = sign\frac{1}{N}\sum_{n=1}^{N} b_n(X).$$

Such compositions address the problem of large variance in Random trees' output, significantly reducing it. The variance of a Random Forest also depends on correlation between Random trees - correlation is not affected by the number of trees and stays a significant source of variance. In order to mitigate the correlation

problem, each node of any tree inside Random forest has to be split with the use of some random dataset features only.

Advantages of Random Forest

- Handy, easy, fast to construct and to train;
- Though, in the present paper only classification Random forest was considered, regression counterpart can also be built;
- Is not affected by overfitting.

Disadvantages of Random Forest

- Though, Random forest is fast to train, in order to reach high precision, a significant number of trees is required which makes the model slow in real-life predictions and validation.
- Not cutting-edge algorithm.

3.3.5 Gradient Boosting

It is known that the most efficient way to get the prediction from the data is to build several machine learning algorithms rather then one, which helps us to reduce the variance and bias of the final prediction. There are two approaches on how to find a final prediction using outcomes from the several algorithms. The first one is called a Bagging technique. In this approach we collect the predictions from the algorithms and assign a certain weight to each prediction. The most importantly is that Bagging technique implies that the ML algorithms used for the predictions are independent. Whereas in the second approach called Boosting we construct the ML algorithms in a way, that they are not independent. In simple words, at each iteration we look at the result of the previous one and assign higher weight to the observations which we couldn't predict.

One might find the following picture useful to get the idea of the difference between Bagging and Boosting (13)

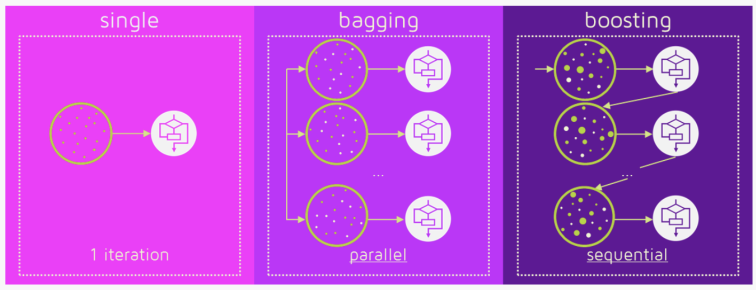


Figure 10: Different ensemble techniques

Similarities of Bagging and Boosting

- Both use the result of the N predictions
- Both helps to reduce variance and bias

Differences of Bagging and Boosting

- N predictions are built independently for Bagging and subsequently for Boosting
- Making a final prediction using the Bagging technique we assign equal weights to N predictions, while for Boosting we assign more weight for the predictions which have done the better prediction

Now having the idea of what is the Boosting lets find out the meaning of the first word in "Gradient Boosting". As we discussed earlier in this algorithm at each step we are trying to improve the last model. How are we doing so? Let F_m be a model we have developed at m-th stage. Then the next model would be

$$F_{m+1}(x) = F_m(x) + \alpha_{m+1} \sum_{i=0}^{n} \nabla_m L(y_i, F_m(x_i))$$

where $L(\cdot)$ is a loss function. So we see that the gradient decent method is used to minimize the loss function.

To conclude the analysis of the Gradient Boosting method let us look at and summarize prons and cons of this tool: (15)

Advantages of Gradient Boosting

- Can work with different Loss Functions
- Works with both numerical and categorical data
- Handles missing Data

- Obtain one of the best predictive accuracy

Disadvantages of Gradient Boosting

- Computationally GBM is a very costly technique
- GBM flexibility is obtained by using many parameters which in turn makes it very complicated for tuning
- Easily affected to overfitting since the GBM algorithm will continue to improve to minimize the errors

3.3.6 Naive Bayes

As it can be understood from the name, Naive Bayes algorithms are the machine learning algorithms based on Bayes Theorem. From the lecture notes of Data science in practice course from Christopher Bruffaerts and Omar Ballester, it is from Bayesian Algorithms Brach of machine learning algorithms. Naive Bayes algorithms are not a single, but many algorithms of simple probabilistic classification algorithms. The main and important assumption of the Naive Bayes algorithms is the independence assumption between the predictors (6). From the Summary Statistics part of this report above, the heat map shows that the correlation between the two variables are approximately between 0.25 and -0.25. Based on this fact, it is known that there is a correlation between the variables, however they are not really large. Therefore, the Naive Bayes algorithm is usable. Again, from the lecture notes, we know that Naive Bayesian model is a powerful and simple algorithm. It perform better than more complex classification methods. Especially, for the large data sets, this characteristics of this algorithm makes it more useful.

First of all, it is good to remember how the Bayes' theorem is:

$$\mathrm{P}(y|x) = \frac{\mathrm{P}(x|y) \cdot \mathrm{P}(y)}{\mathrm{P}(x)} \quad \text{where,}$$
$$\mathrm{P}(y|X) = \mathrm{P}(x_1|y) \cdot \mathrm{P}(x_2|y) \cdot \mathrm{P}(x_3|y) \cdot \cdots \cdot \mathrm{P}(x_n|y) \cdot \mathrm{P}(y)$$

From the formula above it is seen that the posterior probability equals to likelihood times class prior probability, divided by predictor prior probability.

After training the Naive Bayes classifier, the expectation is that new patients are automatically categorized. In order to do so, firstly we need to compute the prior probabilities. This is mainly the percentage of target values in target population. Secondly, we need the frequencies for each features. This is the denominator

part of the formula. Here, the frequencies are the percentage of the feature's values in the target population. Thirdly, in order to apply the Bayes' Formula, the conditional probability 'likelihood' should be calculated for each features. With these three steps, now one can calculate the probability that a new patient has a heart disease by substituting all the 3 equations into the Naive Bayes formula. (14)

Here are the steps in short to follow in Naive Bayes Algorithm:

- **Step 1:** Calculate the prior probabilities.
- **Step 2:** Calculate the frequencies.
- **Step 3:** Calculate the likelihoods.
- **Step 4:** Plug in all these three into Bayes' Formula, to get the probability of a new observation.

Advantages of Naive Bayes

- **It is for Categorical:** Naive Bayes suits well in categorical problems. For numerical variable, the required assumption that test data distributed normally is a strong assumption. (7)
- **Fast Computation:** It quickly predicts the class of data set.
- **Simple Implementation:** It is easy to implement.
- **Fast Computation:** It quickly predicts the class of data set. When the independence assumption holds, Naive Bayes outperforms in multi class prediction the other methods like logistic regression.

Disadvantages of Naive Bayes

- **Strong assumption:** The independence assumption is important and hard to get. Also, if the assumption fails the algorithm performs badly.
- **Zero Frequency:** There is the risk of not observing a categorical variable int the training set. If this happens, model gives zero probability and that prevents to predict correctly.

3.4 Evaluating the Results

The model constructed in the whole data may not be a well-designed relation if the analyst does not care about overfitting issues. Therefore, the analyst may use one of the model validation techniques to confirm and validate whether the model

can be generalized or not. The model validation process can be led by two types of data as train data and test data. Train data is the subject of model construction process, involving goodness of fit of the model and analysis of the residuals as well as the test data is the one to measure the predictive accuracy of the constructed model.

Cross-validation is one part of the model validation process to test the effectiveness of applied machine learning models by using out-of-sample dataset.In this case, we have used K-folds cross validation.

- **K-Folds Cross Validation:** K-fold cross validation is one of the most popular cross-validation techniques for recent years. Its popularity comes from the less biased structure compared to other models. It firstly splits the dataset into K-folds and each time it trains K-1 fold while estimating the Kth fold. This validation methodology tests each of the folds one time and ensures that all of observations appear one time in the test set.

Confusion matrix is a table summarizing the performance of the model with respect to the actual and predicted classes of instances on a set of data. Most of the evaluation metrics stem from the elements of confusion matrix.

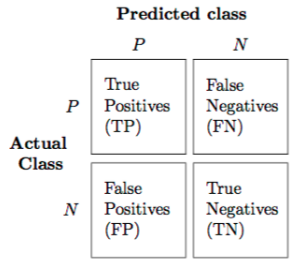


Figure 11: Confusion Matrix

Apart from the method we have used to validate our models, we need certain metrics to select the best machine learning algorithm. To evaluate the performances of these algorithms the metrics below can be used:

- **Classification Accuracy:** Classification accuracy is the overall performance of the model.

$$Accuracy = \frac{TP + TN}{Total}$$

For imbalanced datasets, we may use two main scoring metrics such as : F1 Score and ROC-AUC. However, we need to calculate pre-metrics to calculate the main metrics.

- **F1 Score:** F1 score is the harmonic mean of precision and recall scores.

$$Precision = \frac{TP}{TP + FP}$$
$$Recall = \frac{TP}{TP + FN}$$
$$F1\text{-}Score = 2 \cdot \frac{1}{\frac{1}{Precision} + \frac{1}{Recall}}$$

- **ROC-AUC:** ROC-AUC is the abbreviation of receiver operating characteristic-area under curve, being one of the most popular evaluation metrics for imbalanced datasets. It is basically created by plotting Recall (True Positive Rate) vs False Positive Rate at different threshold values.

$$TruePositiveTate = \frac{TP}{TP + FN}$$
$$FalsePositiveRate = \frac{FP}{FP + TN}$$

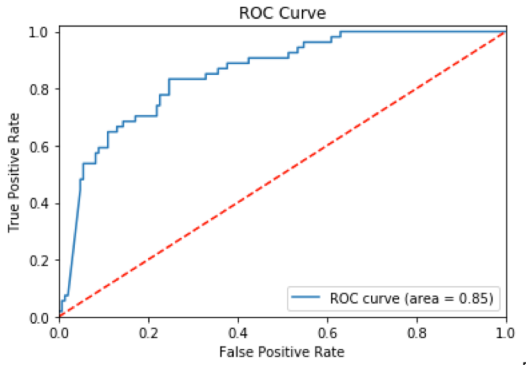


Figure 12: ROC Curve

- **Fitting Time:** Fitting time is one of the fundamental comparison metrics although it is not included directly. At some point of the process, the analyst may be confront a trade-off between the success of the model and the fitting time of that. If the fitting time takes too much time, the company may not be able to update its systems and renew the model easily.

3.5 Hyper-parameter Tuning

Although the desire is to let the machine completely learn by itself, it may not be always the case because of the hyper-parameters. Hyper-parameter is a

parameter whose value is defined and controlled by the analyst. Unlike the other parameters learned in the model, these values are pre-determined. Since a human intervention plays an instrumental role to define the value of hyper-parameters, it becomes an optimization problem that yields an optimal model minimizing the cost function. Hyper-parameter tuning can be applied in a lot of ways, but we are going to apply grid search in this case. Grid search is basically a searching through a manually specified subset of the hyper-parameter space of the machine learning model, and it may be highly time inefficient if the hyper-parameters are not defined clearly.

4 Results

4.1 Unsupervised Learning

4.1.1 Multidimensional Scaling

Using the approach of the Multidimensional Scaling discussed above we were able to obtain the representation of the data in 2-dimensional space. The following graph shows our results:

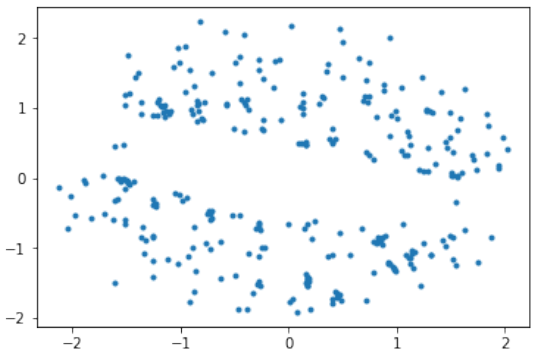


Figure 13: Multidimensional Scaling of the Data

As we can see from the graph our data in 2 dimensional space seems to be clustered in 2 groups. The first thought would be that these 2 groups represents the presence of the heart disease or it's absence. Let us check this assumption with the second graph which for each observation shows us the actual value for the target variable (= 1 if the person has a heart disease and 0 otherwise).

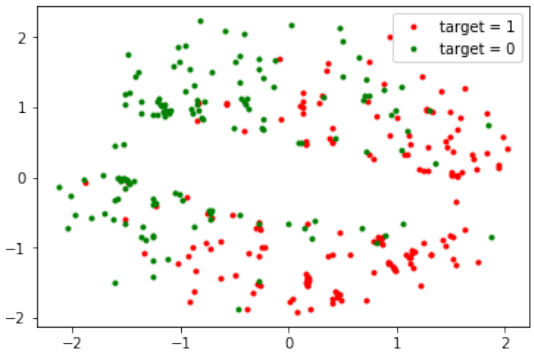


Figure 14: Multidimensional Scaling of the Data with target label

We see that the actual data is not exactly divided by the 2 groups we thought before. However, we still can see the pattern in the data. Mostly all the green dots are located in the left side of the graph, whereas the red ones are on the right.

4.1.2 PCA

Based on the PCA method, we can achieve the unsupervised three classifications and plot the scatter graph in 3D. The result plot is clear to find out the principal three components.These components can explain more than 50 percentage of the total variance in the data sample. The graph is shown as below:

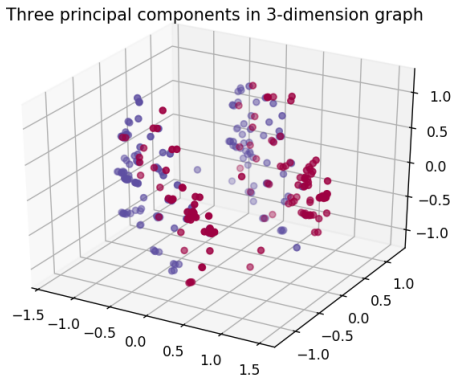


Figure 15: Three principal components in 3-dimension graph

In addition, we can also have the unsupervised two classifications in 2D and corresponding eigenvectors as below:

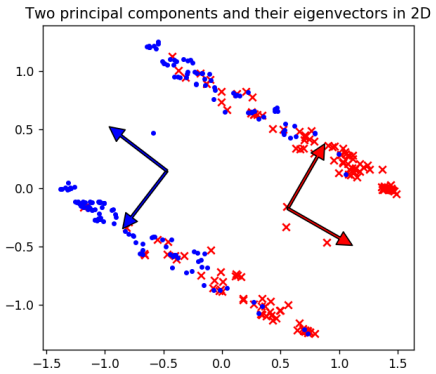


Figure 16: Two principal components and their eigenvectors in 2D

4.2 Supervised Learning

4.2.1 Before Hyper-parameter Tuning

	NB	KNN	LR	GB	RF	SVM
Fit Time	0.003	0.002	0.008	0.065	0.008	0.005
Test Accuracy	0.778	0.822	0.834	0.814	0.774	0.828
Test F1 Score	0.793	0.837	0.851	0.829	0.804	0.833
Test Precision	0.811	0.841	0.845	0.819	0.765	0.859
Test Recall	0.786	0.836	0.860	0.847	0.853	0.818
Test ROC AUC	0.858	0.902	0.909	0.897	0.857	0.903

Table 2: The 10-fold Cross Validation of Gradient Boosting, K-Nearest Neighbor, Logistic Regression, Random Forest, and Support Vector Machines Algorithms before Hyper-parameter Tuning

As we can see from the table, the best algorithms in terms of fitting time is KNN. It is not surprising since as we sad before, the KNN algorithm does not establish any connection between the samples but rather memorize the instances which allows it significantly reduce the fitting time. The longest algorithm in terms of time is Gradient Boosting, which is also make sense since this algorithm will fit the model until it minimizes the loss function so it basically continue to feed the sub-models.

Test Accuracy is another metrics which shows us how well we can predict the true target values. It turns out that the best it does Linear Regression algorithm,

while the worst performance in terms of the Test Accuracy has a Random Forest Algorithm.

The significant difference is observed comparing Naive Bayes approach and Linear Regression algorithm in terms of Test F1 Score. We see that the best LR has the highest average of the precision and recall.

The best algorithm in terms of precision terns out to be Support Vector Machines algorithm, whereas the Random Forest proved to be the worst one.

In terms of Test Recall Naive Bayes and Linear Regression stands out from the other algorithms with Naive Bayes being the one which has the lowest recall score. We suppose this metrics to be a better description of work of the algorithms than Test Precision, because dealing with heart diseases it's crucial to recognize a health problem. We believe that if the algorithm recognizes a healthy person as a one having a heart disease it doesn't cost as much as if the algorithm identify ill person as a healthy one.

As for the Test ROC AUC metrics, Linear Regression prove to be the best one once again. The poorest result showed Random Forest and Naive Bayes. This result is fairly predictable looking at the previous metrics.

4.2.2 After Hyper-parameter Tuning

Having performed the Hyper-parameter Tuning we found the following parameters of the models being the most optimal ones:

- **NB**: variance smoothing: 5e-06
- **KNN**: n: 5
- **LR**: penalty: l2 , C: 2.53
- **GB**: lr: 0.01, max_depth: 3, min_samples_leaf: 0.1, min_samples_split: 0.2, n_estimators: 300, subsample: 0.6
- **RF**: max depth: 80, max features: 3, min samples leaf: 3, min samples split: 8, n estimators: 200
- **SVM**: C: 12, kernel: rbf

	NB	KNN	LR	GB	RF	SVM
Fit Time	0.002	0.003	0.012	0.158	0.320	0.006
Test Accuracy	0.795	0.832	0.844	0.847	0.840	0.830
Test F1 Score	0.812	0.846	0.860	0.866	0.858	0.848
Test Precision	0.824	0.848	0.853	0.831	0.832	0.836
Test Recall	0.805	0.849	0.872	0.908	0.890	0.865
Test ROC AUC	0.858	0.881	0.909	0.921	0.919	0.916

Table 3: The 10-fold Cross Validation of Gradient Boosting, K-Nearest Neighbor, Logistic Regression, Random Forest, and Support Vector Machines Algorithms after Hyper-parameter Tuning

As we compare both Tables before and after Hyper-parameter tuning we see that for most of the algorithms the fitting time increased (but not significantly). The reason is that picking the optimal parameters for the algorithms we don't pick them thinking of the fitting time but rather about minimizing a loss function.

Looking at the results for Test Accuracy, F1 Score, Precision, Recall and ROC AUC for all algorithms in the table we have an improvements in scores after the hyper-parameter tuning. The only value which actually decreased is a Test Precision for the Support Vector Machines algorithm.

5 Conclusion

5.1 Value

With these models, now we have the algorithms which all have more than 85% receiver operating characteristic - area under curve (ROC-AUC) and more than 81% F1 score. If we observe the best model for our data set, which is Gradient Boosting, this means that we can diagnose the heart disease,which early diagnose is crucial, significantly fast. The value of the models which we provide is the better health care for people with more accurate, more precise and faster recognition. Decreasing the duration per patient in hospital means the opportunity of helping more people. Decreasing the false positive results decrease the cost of health care or medication and may create financial value. Moreover, decreasing false negative results may save people life, which is the most important value of this project. As it was mentioned in the introduction part, we believe we succeed with all these advantages, to be beneficial for the humanity in general.

5.2 Future Opportunities

There are several ways of further development of our project that are: expansion of the dataset and work with features; providing more rigorous analysis; construction of more advanced models. The first approach is an essential step for further improvement of the presented methodology in the sense that more extensive dataset allows to train the models better. Moreover, by incorporating more features such as information regarding the lifestyle, working and living conditions, medical history, etc. one can not only increase the precision of results but also achieve early and even preventive diagnostics of the disease therefore allowing patients to start mitigating health problems on the early stages. A crutial part of preparation of a better and more comprehensive dataset is feature selection process which helps to redue the dataset and clear it from irrelevant data and misleading correlations between features. Construction of more advanced models can address two different problems: low computational velocity and low accuracy of results. The former problem may be mitigated, for instance, with the use of parallel computing or, in case of a much more extensive dataset, wise application of sampling methods. The latter problem may be addressed by the use of more enhanced algorithms such as neural networks, which, though, require much more input data to work properly.

References

[1] NG, Andrew Machine Learning Notes by Stanford University on Coursera

[2] Cohn, J N. Screening for Early Detection of Cardiovascular Disease in Asymptomatic Individuals. American Heart Journal, Mosby, 8 Oct. 2003, Science Direct Article

[3] Hamza, Ali Becoming Human Blog Post

[4] U.S. Department of Health Human Services. (2019). *Ischemic Heart Disease.* Retrieved from: National Heart, Lung, and Blood Institute

[5] Centers for Disease Control and Prevention (CDC). (2017). *Heart Disease Fact Sheet.* Retrieved from: CDC Web-site

[6] Wikipedia contributors. (2019, April 30). Naive Bayes classifier. In Wikipedia, The Free Encyclopedia. Retrieved 13:32, May 4, 2019, from Wikipedia-Naive Bayes Classifier

[7] Sunil, Ray Analytics Vidhya Blog Post

[8] Jake VanderPlas Python Data Science Handbook: Essential Tools for Working with Data, 1st Edition

[9] Professor Dr. Tom M. Mitchell, Machine Learning 10-701. Carnegie Mellon University. January 25, 2010. https://www.cs.cmu.edu/ epxing/Class/10701-10s/Lecture/lecture5.pdf

[10] Professor Dr. O. Schulte, Deep Learning. Simon Fraser University https://coursys.sfu.ca/2019sp-cmpt-880-g1/pages/PCA/view

[11] Professor Dr. M. Kalisch, ETH University Lecture Notes

[12] Leo Breiman Random Forests

[13] Ana Porras Garrido Post on ensemble methods

[14] Prabhakaran, S. (2018). How Naive Bayes Algorithm Works? (with example and full code). Retrieved 14:30, May 4, 2019, from https://www.machinelearningplus.com/predictive-modeling/how-naive-bayes-algorithm-works-with-example-and-full-code/

[15] University of Cincinnati Notes on R Programming course

[16] Wikipedia contributors Wikipedia-Support-vector machine

[17] Wikipedia contributors Wikipedia-Support-vector machine