# KFUPM
# CISE483: FINAL REPORT

**28 November 2021**

# TITLE: CHRONIC KIDNEY DISEASE CLASSIFICATION

## 1. Introduction

The specific purpose of our project is to assess the use of machine learning in hospitals within Saudi Arabia. Specifically, by using sklearn and python programming to generate a solid program that will classify a dataset of patients wither they have chronic kidney disease or not. Using three different machine learning algorithms, we can reach to a very high level of accuracy in classifying Chronic Kidney Disease (CKD) patients.

## 2. Chronic Kidney Disease Background

CKD is a long-term disease that impacts the kidneys, damages them, and extends to the overall health by impacting other vital organs as complications. Those complications could be in the form of high blood pressure, low red cells count in the blood, or as known as anemia, nerve damage, and week bones. CKD can be treated to prevent such complications by early detection and treatment. In this project our aim is to help doctors confirm their physical diagnosis by the use of machine learning.

## 3. Dataset

Our dataset consists of 400 patients with 25 features and one target. The features are Age, Specific Gravity, Blood Pressure, Albumin, Sugar, Red Blood Cells, Pus Cell, Pus Cell Clumps, Bacteria, Blood Glucose Random, Blood Urea, Serum Creatinine, Sodium, Potassium, Hemoglobin, Packed Cell Volume, White Blood Cell Count, Red Blood Cell Count, Hypertension, Diabetes Mellitus, Coronary Artery Disease, Appetite, and Pedal Edema. And the goal is to classify a patient as CKD positive or negative given these features.

The dataset was found online from this website:
https://archive.ics.uci.edu/ml/datasets/Chronic_Kidney_Disease

And it can be found attached to this report.

## 4. Overview of the process

Our goal in this project is to classify a dataset of several patients into two classifications; Each patient is drawn to a conclusion of either she/he is CKD positive or negative. The process is well explained in the attached Python file, however let us take an overview of what we expect to happen.

Firstly, we observe the dataset and study it very well to draw a conclusion of what actions we must take next. In our dataset, the first thing we can observe is that the abbreviation is very much not clear and will cause a confusion to the ML programmer since it is medical terms and abbreviation. So, our first step was to take an action in this regard and rename the features into very clear names. Which leads us to our next step of changing the dataset features into numerical.

Secondly, we can notice that this dataset has many features that are written as object. For example, the hypertension feature in the dataset is written for all patients as either yes or no. This must be changed into numerical values so we can implement our classifiers into this dataset. The change in the type of the feature will not change the feature itself or affect the overall process as we will change it into binary; meaning, if it is yes, give it a value of 1, else, give it a value of 0. This will be significantly important once we reach the classification part of this project.

Thirdly, and most importantly, we notice that some of the features have missing value. In other words, the doctors and nurses did not take all the features for each and every patient. For some medical reasons, they thought that hypertension is not important for some patients while blood urea is not important for others and so on. This will not work for our project as we must have all the values of the dataset. To fix this challenge, we decided that if a feature has missing values of 5% or less, then we will remove the patient with the messing values from the feature. However, if the percentage of the missing values is more than 5%, then we will take the mean and the mode to fill the missing values of type numerical and object, respectively. Now, we will end up with a dataset of 355 instead of 400 patients but this new dataset is much more reliable and can be implemented into our program.

Finally, we are splitting our dataset into 20/80 testing/training ratio. This ratio is the most common and, apparently, the most efficient one used. Now we want to implement three different classifiers, the KNN algorithm, the Logistic Regression algorithm, and the Random Forest algorithm. These classifiers are chosen after excessive research and consulting with experts trying to find the best algorithm that fit our project.

1. **KNN algorithm**

The KNN algorithm is a clustering algorithm rather than it is a classification algorithm, as illustrated in *Figure 1*. However, since we want to classify our data into two categories only, we wanted to give it a try and say if it will give accurate results.

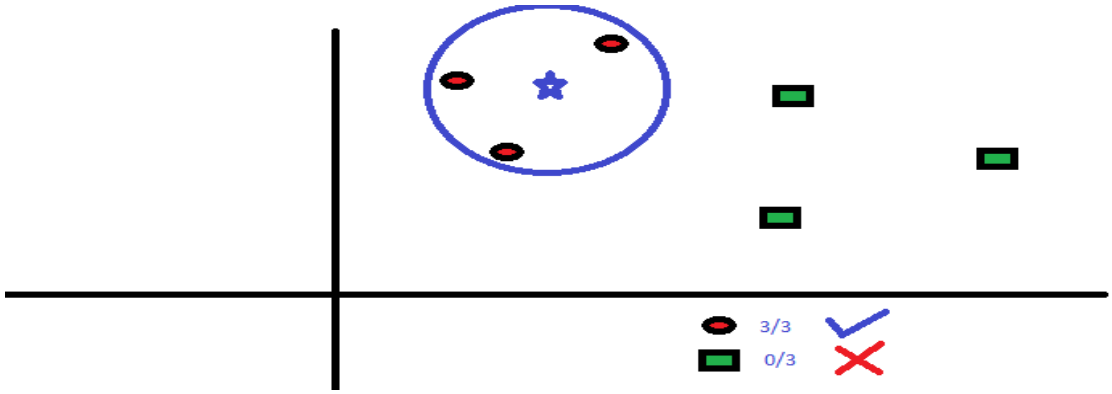



*Figure 1*: The KNN clustering algorithm trying to find the best fitting centroid of the clusters.

In order to implement this algorithm, we had to drop three features:

- ID because it is just a representation of the row where example is found on.
- Classification because this my target.
- Sugar because after a discussion with some medical students, we found that this feature has nothing to do with Kelly disease, and we decided not to put it in training so that the accuracy of the model's prediction is not affected.

After doing so and implementing the dataset to the algorithm, we reached an accuracy of 88.73% which is very not acceptable since we aim to help the medical community with something related to human lives. This conclusion led us to search for the next algorithm to implement.

2. **Logistic Regression algorithm**

Logistic Regression is a supervised learning classification algorithm used to predict the probability of a target variable, as shown in *Figure 2*. The nature of target or dependent variable is dichotomous, which means there would be only two possible classes. In simple words, the dependent variable is binary in nature having data coded as either 1 (stands for success/yes) or 0 (stands for failure/no). Mathematically, a logistic regression model predicts P(Y=1) as a function of X.

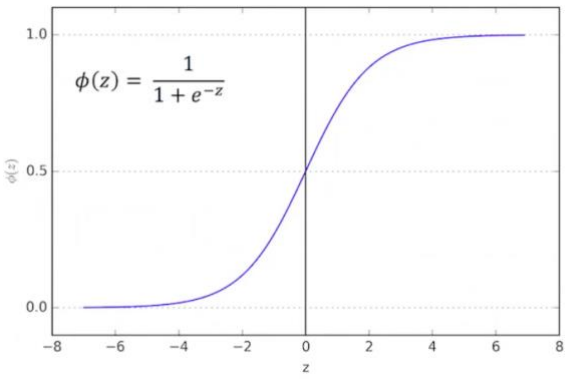
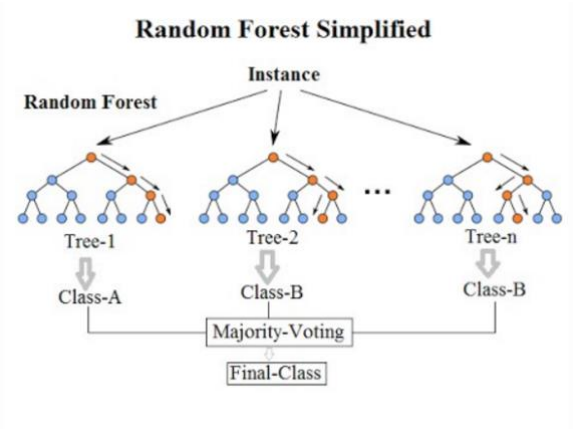


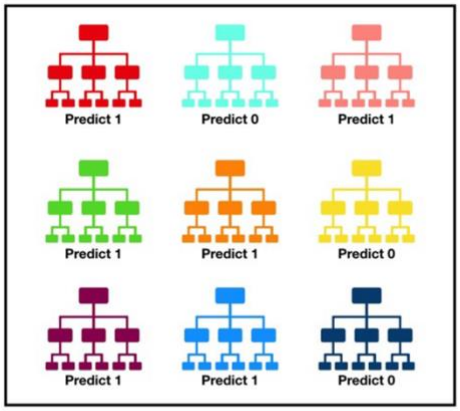

*Figure 2*: The Logistic Regression classification algorithm; graph of its function.

After implementing this algorithm, we reached an accuracy rate of 100% which tells us that we have completed this project and found the best fitting algorithm for this process. However, we wanted to test another algorithm and see if we will get a more efficient algorithm.

### 3. Random Forest algorithm

Random Forest, like its name, consists of a large number of individual decision trees that operate as a group. Each individual tree in the random forest spits out a class prediction and the class with the most votes become our model's prediction, as Figure 3 and Figure 4 illustrates.




*Figures 3 & 4*: The Random Forest Algorithm illustrated.

Applying this algorithm gives us 100% accuracy rate as well. Which generate a question, if both logistic regression and random forest gives 100% accuracy rates, which one is more efficient to this project? Looking deeply into the two algorithms, we can conclude that the logistic regression gives a better result for this project since our aim is to classify the dataset into two classes. Even though the random forest was very successful for this dataset, by evaluating the algorithm itself it is safe to say that applying this algorithm into different dataset might not work as smoothly. In the end, it might generate classes that are not part of the goal we hope to reach.

### 5. Efficiency

This project comes with a very high efficiency. It will help save doctors time by giving fast information about their patients from a very reliable high-tech program that gives a perfect accuracy of 100% using different machine learning algorithms.

### 6. Accuracy

The accuracy rate differs between the three algorithms we used. The first one, KNN algorithm, gives an accuracy of about 88.73%. The second one, Logistic Regression algorithm, gave a perfect accuracy of 100% with a very good percussion rate. Even though not needed, we did the last one to explore the outcomes and it gave a good accuracy of about 100% as well.

## 7. Codes

The codes are found in the Python file attached to this report.

## 8. Evaluation

In order to evaluate this project, we must consider the computational time, the accuracy, and the future modifications.

8.1. The program of this project is a bit heavy. It needs a good computer with modern technologies implemented, fast processor for example, and a relatively very good internet connection for it to work as an optimal solution.

8.2. Giving the accuracy rates above and the evaluation section in the python file, the accuracy is very good and more than just reliable.

8.3. In implementing this project, we made sure that future modifications can be applied. Such modifications could be in using more advanced machine learning algorithms that could lead to better results in dealing with more complex datasets, commenting every line on the code so a future programmer can understand all the work, and implementing a similar program for a different, more complex, and time-consuming disease diagnosing.

## 9. Conclusion

To sum up, In this study, 26 data recording information of 355 people such as Age, Blood_Pressure, Specific_Gravity, Blood_Urea and others were used as attributes. Clinical records were examined to determine whether CKD was present or not and provided a high accuracy rate with machine learning methods. Moreover, three different classifiers were utilized in determining the targeted CKD and the best performing classifier was found by trial and error. These algorithms were compared on the basis of accuracy rate, sensitivity to testing datasets, recall and f1 score. When the results were evaluated with the data used in this study, it was seen that the Logistic Regression algorithm without cross validation performed better than other classification algorithms. Machine learning tools can be used for timely and accurate diagnosis of CKD, helping doctors confirm their diagnostic findings in a relatively short time, thereby helping a doctor to look and diagnose more patients in less time. In future studies, it may be possible to use different algorithms, such as deep learning methods, to predict CKD.

## 10. Used Resources


1. CKD:
   a. https://www.nhs.uk/conditions/kidney-disease/
   b. https://www.mayoclinic.org/diseases-conditions/chronic-kidney-disease/symptoms-causes/syc-20354521
2. Machine Learning and Deep Learning:
   c. https://iopscience.iop.org/article/10.1088/1742-6596/1624/2/022031/pdf
   d. https://www.researchgate.net/publication/321283578_Robot_Path_Training_and_Planning_Usign_LSTM_Network
   e. https://www.sciencedirect.com/topics/medicine-and-dentistry/logistic-regression-analysis
   f. https://www.statisticshowto.com/probability-and-statistics/regression-analysis/

3. A Data science and machine learning expert was a great help for us in introducing us to different machine learning algorithms.


| **PREPARED FOR:** | **TEAM MEMBERS:** | | |
|---|---|---|---|
| ***Dr. Khaled Alshehri*** | ***Bandar Alsuhaibani*** | **201443480** | **CISE – Elective** |
| | ***Meshari Alkhaldi*** | **201686420** | **CISE – CX** |